Efficient and High Performing Biometrics: Towards Enabling Recognition in Embedded Domains

Zur Erlangung des akademischen Grades Doktor-Ingenieur (Dr.-Ing.)
Genehmigte Dissertation von Fadi Boutros aus Idlib, Syria
Tag der Einreichung: April 22, 2022, Tag der Prüfung: June 14, 2022

1. Gutachten: Prof. Dr. Arjan Kuijper
2. Gutachten: Prof. Dr. Dieter W. Fellner
3. Gutachten: Prof. Dr. Kiran Raja
Darmstadt

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Computer Science
Department

Interactive Graphics
Systems Group

Efficient and High Performing Biometrics: Towards Enabling Recognition in Embedded Domains

Accepted doctoral thesis by Fadi Boutros

1. Review: Prof. Dr. Arjan Kuijper
2. Review: Prof. Dr. Dieter W. Fellner
3. Review: Prof. Dr. Kiran Raja

Date of submission: April 22, 2022
Date of thesis defense: June 14, 2022

Darmstadt

Erklärungen laut Promotionsordnung

§8 Abs. 1 lit. c PromO

Ich versichere hiermit, dass die elektronische Version meiner Dissertation mit der schriftlichen Version übereinstimmt.

§8 Abs. 1 lit. d PromO

Ich versichere hiermit, dass zu einem vorherigen Zeitpunkt noch keine Promotion versucht wurde. In diesem Fall sind nähere Angaben über Zeitpunkt, Hochschule, Dissertationsthema und Ergebnis dieses Versuchs mitzuteilen.

§9 Abs. 1 PromO

Ich versichere hiermit, dass die vorliegende Dissertation selbstständig und nur unter Verwendung der angegebenen Quellen verfasst wurde.

§9 Abs. 2 PromO

Die Arbeit hat bisher noch nicht zu Prüfungszwecken gedient.

Darmstadt, April 22, 2022

F. Boutros

Abstract

The growing need for reliable and accurate recognition solutions along with the recent innovations in deep learning methodologies has reshaped the research landscape of biometric recognition. Developing efficient biometric solutions is essential to minimize the required computational costs, especially when deployed on embedded and low-end devices. This drives the main contributions of this work, aiming at enabling wide application range of biometric technologies.

Towards enabling wider implementation of face recognition in use cases that are extremely limited by computational complexity constraints, this thesis presents a set of efficient models for accurate face verification, namely MixFaceNets. With a focus on automated network architecture design, this thesis is the first to utilize neural architecture search to successfully develop a family of lightweight face-specific architectures, namely PocketNets. Additionally, this thesis proposes a novel training paradigm based on knowledge distillation (KD), the multi-step KD, to enhance the verification performance of compact models. Towards enhancing face recognition accuracy, this thesis presents a novel margin-penalty softmax loss, ElasticFace, that relaxes the restriction of having a single fixed penalty margin.

Occluded faces by facial masks during the recent COVID-19 pandemic presents an emerging challenge for face recognition. This thesis presents a solution that mitigates the effects of wearing a mask and improves masked face recognition performance. This solution operates on top of existing face recognition models and thus avoids the high cost of retraining existing face recognition models or deploying a separate solution for masked face recognition.

Aiming at introducing biometric recognition to novel embedded domains, this thesis is the first to propose leveraging the existing hardware of head-mounted displays for identity verification of the users of virtual and augmented reality applications. This is additionally supported by proposing a compact ocular segmentation solution as a part of an iris and periocular recognition pipeline. Furthermore, an identity-preserving synthetic ocular image generation approach is designed to mitigate potential privacy concerns related to the accessibility to real biometric data and facilitate the further development of biometric recognition in new domains.

Zusammenfassung

Der wachsende Bedarf an verlässlichen und genauen Erkennungsmethoden zusammen mit den kürzlichen Fortschritten im Bereich des Deep Learning haben den Forschungsbereich der biometrischen Erkennung grundlegend verändert. Die Entwicklung von effizienten biometrischen Lösungen, die den benötigten Rechenaufwand minimieren ist wichtig, vor allem wenn die biometrischen Methoden auf eingebetteten Systemen oder Low-End-Geräten eingesetzt werden. Hintergrund dieser Arbeit ist daher, einen breiten Anwendungsbereich für biometrische Technologien zu erschließen.

Um eine breitere Anwendung von Gesichtserkennung in Szenarien mit starker Limitierung bezüglich des Rechenaufwands zu ermöglichen, präsentiert diese Thesis eine Reihe von effizienten Gesichtserkennungsmodellen namens MixFaceNets. Mit Fokus auf automatisiertem Netzwerkarchitektur-Design ist diese Thesis die erste, welche Neural Architecture Search einsetzt, um eine Reihe von kompakten Netzwerkarchitekturen, PocketNets, für die Gesichtserkennung zu entwickeln. Des Weiteren präsentiert diese Thesis ein neues auf Knowledge Distillation aufbauendes Trainingsparadigma namens multi-step KD, welches die Verifizierungsperformanz von kompakten Modellen verbessert. Um die Gesichtserkennungsgenauigkeit zu erhöhen, präsentiert diese Thesis zudem eine neuartige margin-penalty softmax loss Funktion, ElasticFace, welche die Einschränkung bezüglich einer festen penalty margin aufhebt.

Die Verdeckung von Teilen des Gesichts durch Gesichtsmasken während der jüngsten COVID-19 Pandemie stellt eine neue Herausforderung für Gesichtserkennungssysteme dar. Diese Thesis präsentiert einen Ansatz, welcher die Effekte der Maske auf die Erkennungsperformanz abschwächt und so die Performanz verbessert. Der vorgestellte Lösungsansatz setzt auf existierende Gesichtserkennungsmodelle auf und vermeidet so zusätzlichen Rechenaufwand aufgrund des erneuten Trainierens oder der Umsetzung eines separaten Lösungskonzepts für maskierte Gesichter.

Mit dem Ziel, biometrische Erkennung in neue eingebettete Domänen einzuführen, wird in dieser Thesis erstmals vorgeschlagen, Head-Mounted Displays für die Identitätsverifizierung von Benutzern von Virtual und Augmented Reality Anwendungen zu benutzen. Hierfür wird zudem eine kompakte Lösung zur Segmentierung der Augen als Teil der Erkennungspipeline vorgeschlagen. Darüber hinaus wird ein identitätserhaltender

Ansatz zur Erzeugung synthetischer Bilder von Augen entwickelt, um potenzielle Datenschutzbedenken im Zusammenhang mit dem Zugang zu echten biometrischen Daten zu entkräften und die Weiterentwicklung der biometrischen Erkennung in neuen Bereichen zu erleichtern.

Acknowledgement

Throughout the journey of my Ph.D. study, I received a great deal of support from my supervisors, colleagues, friends, and family.

I would like first to thank my supervisor Prof. Dr. Arjan Kuijper, for providing scientific guidance, continuous support, and great advice throughout my Ph.D. study. I would like to express my sincere gratitude to Prof. Dr. techn. Dieter W. Fellner for co-refereeing this work. I am also sincerely grateful to Prof. Dr. Kiran Raja for acting as co-referee.

I am beyond thankful to my friend and supervisor, Dr. Naser Damer, for his invaluable support of my Ph.D. study. His insightful feedback, motivation, and immense knowledge pushed me to sharpen my thinking, bringing this work to a higher level.

I would like to thank my friends and colleagues at the Competence Center for Smart Living & Biometric Technology (SLBT) of the Fraunhofer IGD. A Special thank goes to the head of the SLBT department Florian Kirchbuchner for his positive leadership and constant support. I appreciate all the support I received from my former colleague Saied Tazari from my first day at Fraunhofer IGD. My deepest thank extends to my friend and colleague Meiling Fang for the treasured working time spent together before deadlines, and for all fun we have had in the last three years. Special thanks go to Naser, Meiling, and Marco for their outstanding efforts in reviewing this thesis.

During my work on this thesis, I was thrilled to work with many students from whom I learned a lot. Thank you, Philipp, Salah, Andre, Marcel, Wei, Patrick, Malte, Sven, Ha Hai, Olga, and Tim, for your excellent work.

Most importantly, I would like to thank my dear mother and sisters for their unconditional love and encouragement. Finally, I would never have been able to get as far as I have without the tremendous understanding and encouragement of my fiancée, Dalia. You are always there for me.

Contents

List of abbreviations

AUC Area under the curve.

AutoML Automated Machine Learning.

CA-LFW Cross-Age LFW.

CFP-FP Celebrities in Frontal-Profile in the Wild.

CP-LFW Cross-Pose LFW.

DARTS Differential Architecture Search.

EER Equal Error Rate.

EUM Embedding Unmasking Model.

FAR False Acceptance Rate.

FLOP/S Floating Point Operations Per Second.

FLOPs Floating Point Operations.

FMR False Match Rate.

FNMR False Non-match Rate.

FR Face Recognition.

FRR False Reject Rate.

FTAR Failure to Acquire Rate.

FTCR Failure to Capture Rate.

FTXR Failure to Extract Rate.

HMD Head-Mounted Display.

IJB-B IARPA Janus Benchmark-B.

IJB-C IARPA Janus Benchmark-C.

IMR Iris Mask Ratio.

KD Knowledge Distillation.

LFW Labeled Faces in the Wild.

MACs Multiply Accumulate Operations.

NAS Neural Architecture Search.

PSNR Peak Signal to Noise Ratio.

RMSE Root Mean Square Error.

ROC Receiver Operating Characteristic.

SOTA State of the Art.

SRT Self-restrained Triplet Loss.

SSIM Mean Structural Similarity Index.

TAR True Acceptance Rate.

VR/AR Virtual Reality and Augmented Reality.

1. Introduction

Biometric recognition is the automated recognition of individuals based on their behavioral or biological characteristics [122]. Historically, applications using biometrics have been originated and used by law enforcement agencies within forensic investigations [44]. Nowadays, biometric recognition systems are an integral part of identity management systems with various application scenarios, such as automated border control [76], forensic application [44], and financial transaction [226]. Unlike knowledge-based (e.g. user-names and passwords) and ownership-based (e.g. tokens) authentication systems, biometric characteristics cannot be lost, forgotten, or delegated to a third party. These advantages enhance the usability and the security of traditional authentication systems, leading to an increase in the deployment of operational biometric systems. According to an extensive study by [40], the global biometrics market has been rapidly growing since 2016. The total global market revenue reached $18.78 billion in 2019 with an annual growth rate of 18.6% and an anticipated worth of $45.96 billion by 2024.

The advancements in biometric technology, equipped by the recent developments in computational intelligence, enable incorporating biometric technology into novel domains such as embedded environments. To perform biometric recognition, a biometric template that represents a biometric sample of an individual needs to be extracted. The inference of such template from a biometric sample can occur either on a back-end server (e.g. cloud) or on an edge device (e.g. mobile device, computer, or embedded device). Figure 1.1 shows an example of an on-device biometric system (embedded biometric). Embedded biometrics is a particular use of biometric recognition involving devices and use cases that targets minimizing the operational requirements (including hardware, energy consumption, etc.) and maximising the recognition performance, e.g. mobile devices. In embedded biometrics, data capture devices (e.g. cameras) are often built-in the devices themselves, biometric templates are locally stored [154], and biometric feature extraction and comparison operations are performed on edge devices [154, 74]. On-device inference enables a completely on the edge process, e.g. on a mobile device, which allows users to benefit from fast inference at the edge without sending sensitive biometric data to the server for biometric decision-making [154]. An example of on-device identity recognition systems is the Face ID powered by Apple [119, 216] and the Android FR powered by Google

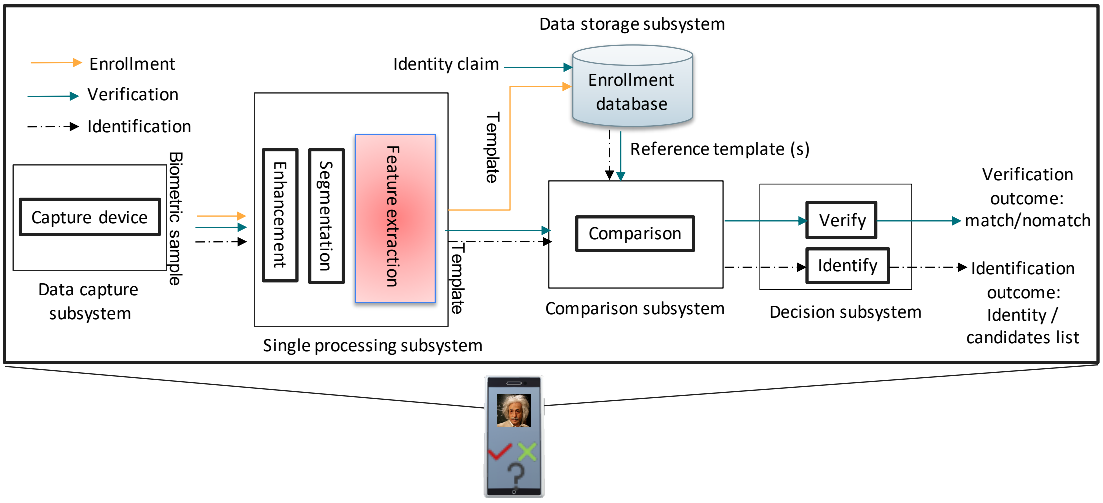


Figure 1.1.: General component of a biometric system in three operations modes: enrollment, verification and identification [123]. In this figure, all operation are performed on-device.

[95, 193]. These FR systems operate on the edge and can securely verify the identity of the individual. It should be noted that edge devices, such as mobile devices, usually have limited computational resources and power capacity. Thus, designing biometric systems for such devices requires carefully considering their computational limitations.

According to the recent report from the Juniper market research [182], FR will be integrated into 2.1 billion mobile devices by 2024, in comparison to an estimated 96 million in 2019. A study by Lovisotto et al. [166] pointed out the high demand for mobile biometric technology by stating that 93% of the MasterCard e-payment customers preferred mobile biometric authentication to traditional methods such as a password, and 92% of banks desire to adopt such technology.

In 2021, IDEMIA, a major biometric and identity solutions provider, announced the launch of Mobile ID technology in four states in the United-State [2]. This solution allows the residents to have a digitized version of their driver licenses or state-issued IDs on their mobile devices. Individuals can use the digitized ID as a legal form of identity verification where the associated information can only be accessed through biometric verification.

Another prominent use case of computationally restrained biometrics is the one in the automotive domain [262]. According to a study by Allied Market Research [1], the global automotive biometric market was valued at $476 million in 2017, and it is anticipated to be worth $1128 million by 2024. As hardware cost in the automotive domain is relatively high, the biometric solution provider should pay attention to the required deployment resources, especially when competing with other smart automotive applications. Growth in the demand for safe, secure, and convenient access control solutions is the primary aspect that drives the evolution of the automotive biometric market [245, 111]. Integrating biometric identity recognition, such as speech, face, and iris recognition, in automotive domains, provides secure vehicle access control, ignition switch, vehicle personalized, and health monitoring for safety purposes [282]. An application use-case of automotive biometrics is car-sharing, where a user can be automatically recognized, enabling vehicle customization and guaranteeing secure temporal access to vehicles [137].

Building efficient, convenient, and high-performing biometric solutions is essential to enable the spread of the technology in novel domains. This work tackles several research questions raised by the emerging deployments of biometrics in the embedded domain and presents solutions to address these research questions. These solutions targets minimizing computational costs, deployment in novel domains, as well as targeting emerging challenges in the biometric domain. This chapter presents the motivation towards developing biometrics in the embedded domain in Section 1.1, followed by the research questions posed in this work in Section 1.2. Finally, an outlook on the content of the rest of this dissertation is presented in Section 1.3.

1.1. Toward efficient biometric

One of the primary aspects of achieving an accurate biometric recognition system is to extract a discriminative biometric template from biometric samples. Biometric template refers to a compact representation of the captured characteristic of an individual so that this representation, i.e., template, is discriminated in comparison to other individuals. Recently, high-performing biometric solutions, especially FR ones [80, 185, 117, 268, 27], rely on deep neural networks for biometric template extraction due to their high learning capabilities. The rapid progress in deep learning research has been dramatically influenced by the advanced computational capabilities of the hardware accelerators such as Graphics Processing Unit (GPU) and Tensor processing unit (TPU), enabling training extremely deep neural networks [258]. However, the computational demands of training and deploying deep learning-based solutions have scaled up rapidly over the past years, especially with the increased depth and width of such network leading to a high number of trainable parameters [235, 258]. For example, the computational demands of training deep neural networks have increased 300,000 times from 2012 to 2018 [10]. Besides the high computational demand for training deep neural networks, deploying such solutions on use-cases constrained by the computational capabilities is challenging [180, 81], due to rapid inference requirements along with resources limitations on edge devices, e.g. memory footprint and power capacity.

These challenges raise the need for novel, accurate, yet efficient solutions. The efficiency in this perspective refers to deep learning solutions that can achieve high accuracy without increasing the computational demands and, ideally, decreasing them. From an application perspective, such an efficient biometric solution is needed for two main reasons: a) Computational resources is critical, and b) Minimize hardware cost. In use-cases constrained by computational resources such as mobile devices, Head-Mounted Display (HMD) devices, and internet-of-thing (IoT) devices, the available run-time memory, computational operations, and power capabilities are limited. Also, these resources are usually shared between several applications to enable simultaneous access to smart applications. Thus, enabling a biometric recognition in such an environment requires designing efficient and accurate solutions that can reliably operate with minimum resources.

The emerging deployment of biometric systems in new domains often requires additional investment into hardware. The hardware cost varies between different application scenarios. For example, the hardware cost is relatively high in the automotive domain. Automotive companies such as Daimler and HyundaiMotor announced that biometric technology would be integrated into their high-end luxury segmented cars [120, 53]. These limited deployments are mainly caused by the high electronics component cost in the automotive domain [240]. Reducing the computational cost of a biometric solution, and thus the required hardware cost will enable wider implementation of biometric solutions in the automotive domain. In other application scenarios, such as large-scale identity management, the cost comes from the scale of the operation rather than the specialty of the application domain. An example of a large-scale biometric system is the EU Entry/Exit (EES) System, aiming at registering information (name, travel document, biometric data, and place of entry and exit) of travelers from third countries each time they cross an EU external border [52]. Achieving an efficient biometric solution in large-scale application scenarios reduces the operation cost and thus, enables wider deployability of the biometric solution.

1.2. Research questions

This thesis aims at enabling a wider implementation of biometric technologies in the embedded domains and use-cases constrained by computational capabilities and operational limitations. This section presents three principal research questions posed by this thesis, followed by detailed research questions derived from the principal research questions to address each of them more granularly. The research questions fall within three research areas based on the identified challenges and targeted application.

In the first research area, this thesis concentrates on enabling FR in use-cases constrained by computational capabilities, which requires designing efficient and accurate FR models. This leads to the first principal research question posed in this thesis:

RQ1: Can efficient and high-performing FR approaches be successfully designed?

Achieving efficient and high-performing FR solutions leads to the broad integration of the technology in various applications from border control to logical access control on consumer end-devices. However, FR still faces several challenges such as pose variations [295], ageing [192] and occlusion [64]. One of the recent and emerging challenges for FR is the face mask occlusions presented during the recent COVID-19 pandemic. Several studies [64, 82, 201, 202] have evaluated the effect of wearing protective face masks on FR performance and concluded that such occlusion negatively affect the FR verification performance. The second research area covers this emerging challenge, leading to the second principal research question:

RQ2: Can the negative impact of face masks on FR verification performance be effectively reduced?

FR commonly requires capturing full faces of the subjects for the recognition process, which might be infeasible in some application scenarios due to the limited data capture setup. Virtual Reality and Augmented Reality (VR/AR) technologies utilize HMDs, which typically include eye-facing cameras that capture the ocular region and are used for eye tracking [96]. The possible introduction of biometric recognition to VR/AR using the existing setups is therefore limited to specific biometric modalities within the ocular region. The third research area focuses on introducing biometric recognition to VR/AR applications enabled by HMDs, leading to the third research question:

RQ3: Can existing VR/AR setups be leveraged for the biometric verification of their users identities?

This thesis dissects the stated principal research questions into detailed research questions and provides extensive responses to each of them. The rest of this section presents the sets of detailed research questions following the principal RQ1, RQ2, and RQ3, in Sections 1.2.1, 1.2.2 and 1.2.3, respectively.

1.2.1. Efficient and high-performing face recognition

Following the stated principal RQ1, this section presents detailed research questions. FR technologies are increasingly used to enhance the security and convenience of identity verification processes, such as border control and financial services. State of the Art (SOTA) FR models [80, 268, 156, 117] depend on learning to extract deep feature representation using a deep neural network that applies multiple convolutional layers. The design choice of SOTA deep learning-based FR models followed the common trend of other computer vision applications by utilizing an overparameterized deep neural network with high computational cost [107, 110, 241, 252]. Deploying such as an overparameterized model on use-cases constrained by a computational capability is challenging. This challenge has received increased attention in the literature in the past few years [180, 81]. The main focus of recent efficient FR models presented in the literature was reducing the memory footprint of the FR model [49, 179, 284, 150, 281]. Although the reduction of the memory footprint is important, the aspect of computational complexity is additionally critical for many use-cases, and it received relatively lower attention in FR literature. In use-cases that are extremely limited by computational complexity, achieving an accurate solution with low complexity is essential to enable FR. All these motivate the first research question raised in this work, stated as follows:

RQ1.1: Can a network based on multi-scale convolution operations be designed for accurate and yet low computationally complex FR? And can the accuracy be further improved by enabling information communication between various fractions of the network?

Efficient deep FR models proposed in the literature [49, 179, 284, 281, 150] are commonly adopted from the ones designed for common computer vision tasks [233, 173, 290, 144]. With the developments in Automated Machine Learning (AutoML), Neural Architecture Search (NAS), a technique for automating the design of neural network architectures, has shown SOTA performances in many computer vision tasks [155, 283]. The NAS solutions are commonly trained and evaluated on general image classification datasets such as CIFAR10 (animals, cars, etc.) [143] with the training objective of recognizing the main object in images [155, 283]. Unlike the common images classification task, the training objective of FR is to learn discriminative identity features from the face images, which might be more subtle. Thus, architectures designed for common computer vision tasks could be suboptimal for FR. This motivates the next question:

RQ1.2: Can NAS be successfully utilized to design a lightweight network specifically for FR? Does this architecture optimization over face identities enhance the learned architecture?

Knowledge Distillation (KD) is a common technique to improve the performance of compact models by transferring the knowledge learned by a deeper model (teacher) or assembly of models to a single small model (student) [108]. When the network size gap between the teacher and the student models is large, transforming the knowledge to a shallow student is challenging [284, 187], due to the different levels of model complexity. This affects the effectiveness of the KD process and thus might lead to less optimal performance of student models. This motivates the next research question tackled in this work:

RQ1.3: Can the difficulty of a substantial discrepancy between teacher and student model in KD paradigm be relaxed through KD process management? Does such a solution lead to a better-performing student model?

In addition to the evolution of deep network architectures, training losses are behind major advances in achieving accurate FR [150, 268, 117, 185, 27]. Early FR models such as FaceNet [234] proposed to utilize metric-based learning, e.g. triplet loss [234], to minimize the distance between face embeddings of the same identity while maximizing the distance between embeddings of different identities. An alternative to metric-based learning loss is the softmax classification loss and its variants. Margin-penalty softmax loss is the most adopted loss in the recent high-performing FR solution due to its SOTA performance on the main benchmarks [268, 80, 159]. Such a loss adds a fixed margin penalty between the feature embeddings and their respective class centers to encourage intra-class compactness and inter-class separability between the learned features. However, fixed margin penalty losses [268, 80, 159] assume that the samples can be equally pushed to their class center. This learning objective may lead to sub-optimal verification performance in a real training dataset with inconsistent inter-and intra-class variations. This motivates our next research question:

RQ1.4: Can the FR performance be enhanced by relaxing the fixed margin penalty during training through assigning a more flexible penalty margin?

Focusing on enabling FR on use-cases constrained by a computational capability, this thesis proposes a set of efficient and yet accurate FR models. Moreover, this work investigates designing a compact FR model by taking advantage of NAS to design a face-specific architecture. Such approaches consider the computational cost by design, and thus, their operations require minimum computational resources. Additionally, to enhance the accuracy of FR, a novel margin-penalty softmax loss that relaxes the restriction of having a single fixed penalty margin is presented.

1.2.2. The emerging challenge of masked face recognition

This section presents detailed research question derived from the principal RQ2. FR has been preferred as a contactless identity verification solution deployed in many application scenarios, such as automated border control [3]. Given the recent COVID-19 pandemic, wearing masks became an essential means to prevent the spread of contagious diseases, which presents a new challenge for FR. This motivates a number of studies to evaluate the effect of wearing a face mask on a FR performance [64, 82, 201, 202]. These studies concluded that FR performance, and thus the trust in contactless identity verification through FR, is affected by wearing a mask. Several works proposed to deal with this challenge by training a FR model with synthetically generated masked faces [118, 195]. However, deploying such solutions in a real-world scenario is not realistic and comes with a high cost as it requires replacing the current FR solution with new ones. Furthermore, the previous studies [64, 82, 201, 202] that evaluated the effect of the face mask on FR performance reported that the genuine score distribution, i.e., distribution of scores obtained by comparing samples belonging to the same identity, is significantly affected by masked probes. The studies [64, 61] also reported that the genuine score distribution strongly shifts towards the imposter score distributions. On the other hand, masked face probes do not seem to strongly affect the imposter score distribution, i.e., distribution of scores obtained by comparing samples belonging to the different identities. These motivate the next questions in this work.

RQ2.1: Can a compact model be designed on top of existing FR models to produce masked face templates that perform similarly to the ones from unmasked faces? Can such a solution be designed to take advantage of the deep insights into the effect of wearing a mask on FR verification?

To answer this question, this thesis designs an approach to reduce the negative impact of wearing a protective face mask on FR performance. The proposed approach processes an embedding of a masked face and outputs a new embedding that behaves similarly to the embedding of the unmasked faces of the same identity. Such a solution is designed to operate on top of existing FR models, and thus, it avoids the cost of retraining the base FR models or requiring an additional full scale dedicated masked FR model.

1.2.3. Biometrics in head-mounted displays

This section presents detailed research questions derived from the principal RQ3. VR/AR, enabled by HMDs, is being increasingly deployed in different applications such as healthcare, education, and law enforcement [259, 172, 246]. These applications often require accessing and processing sensitive information that should only be accessible to authorized users. HMD devices typically include internal cameras to facilitate gaze interaction with the virtual environment [96]. The identity verification of the HMD user using this internal camera is not yet explored. Such a solution is not only understudied but also has to consider the limited computational resources in HMDs. This led to the next research question posed in this thesis and stated as follows:

RQ3.1: Can images captured from the internal camera of HMD devices be successfully used for biometric authentication based on the suboptimal iris and periocular captures?

The performance of iris recognition depends on the precise segmentation of the iris area [294, 217]. Considering the minimalistic hardware specifications of an HMD device, the segmentation solution should be efficient yet accurate to enable continuous identity verification. Such segmentation solution is additionally needed to enable smoother interaction and eye-tracking in AR/VR environment [96]. These motivate the next research question that can be formulated as follows:

RQ3.2: Can a key eye-regions semantic segmentation solution successfully take advantage of multi-scale representations to perform accurately?

Enabling biometric solutions, especially in new domains, requires the availability of identity-specific biometric data, i.e., pairs of samples belonging to the same identity with large variations. However, it is not a trivial task, and it may not be feasible to collect biometric datasets for biometric processing due to privacy concerns [266]. Such concerns motivate the next investigation in this work.

RQ3.3: Can an identity preserving ocular image be successfully generated from an arbitrary semantic segmentation? How well would these images preserve the content and identity embedded in the iris and periocular characteristics?

In the effort to answer these questions, this thesis is the first work that proposes and investigates enabling biometric recognition in HMDs from their existing built-in cameras. The proposed biometric recognition solutions focus on both, the iris and periocular region. Moreover, a compact multi-label segmentation model is designed to serve essential preprocessing operations. Additionally, this thesis successfully proposes an identity-preserving synthetic ocular image generation approach.

1.3. This thesis

This section presents an overview of the rest of this thesis.

Chapter 2 provides essential background knowledge that enables a better comprehension of the contributions of this thesis and their motivations. It starts by discussing the components of biometric system and the performance evaluation metrics, including those measuring the performance of biometric recognition, as well as the computational costs.

A high-level overview of deep learning-based FR models is presented, including the main network architectures, loss functions, and mainstream benchmarks.

Chapter 3 proposes two sets of FR networks and a novel margin-penalty loss function. First, as a response to RQ1.1, this chapter presents a set of extremely low complex and high throughput models for accurate face verification, MixFaceNets. The proposed MixFaceNets are evaluated and compared, in terms of verification performance and computational complexity, with SOTA efficient FR models proposed in the literature. Second, as a response to RQ1.2, this thesis is the first to automate the design of FR network architecture by successfully utilizing NAS learned on a face dataset, resulting in a set of highly compact architectures, PocketNets. Chapter 3 will then present a novel training paradigm based on KD, the multi-step KD, in response to RQ1.3. Lately, a novel margin penalty softmax loss, ElasticFace, is proposed. ElasticFace aims at relaxing the fixed penalty margin constrain by proposing elastic penalty margin loss allowing flexibility in the push for class separability. This chapter responds to RQ1.4 by presenting, discussing, and evaluating ElasticFace in comparison to the state-of-the-art apporaches.

In response to RQ2.1, Chapter 4 successfully presents a novel solution to reduce the effect of wearing a face mask on FR verification performance. This chapter first investigates how the verification performances of high-performing and compact FR models are affected by wearing a face mask. This investigation is then used to theorize the learning process of the proposed unmasking embedding model and self-restrained triplet loss. Such a learning process aims at training the unmasking embedding model to learn to process a masked face embedding to behave more similarly to an embedding from an unmasked face of the same identity. The benefit of the proposed solution is successfully demonstrated on top of different FR models.

Chapter 5 focuses on designing and evaluating several approaches to enable biometrics in VR/AR applications. First, this chapter proposes to use the ocular images captured by eye-oriented cameras within HMD devices for biometric verification, taking into account the limited computational resources in HMD devices. Additionally, this chapter provides in-depth analyses on the effect of iris selection based on the amount of visible iris in the image on the biometric performance and presents a new methodology to select the suitable iris to enhance the biometric performance on HMD devices, responding an answer to RQ3.1. A compact semantic segmentation model for the ocular region is designed and evaluated as a response to RQ3.2. Later, this chapter proposes and validates an identity-preserving synthetic ocular image generation approach. The identity preservation of the generated images is validated by providing the biometric performance of the iris and periocular characteristics on the generated images and comparing its verification performances to the one on real data, responding to RQ3.3.

Chapter 6 provides a set of concluding remarks of this thesis and an outlook for future

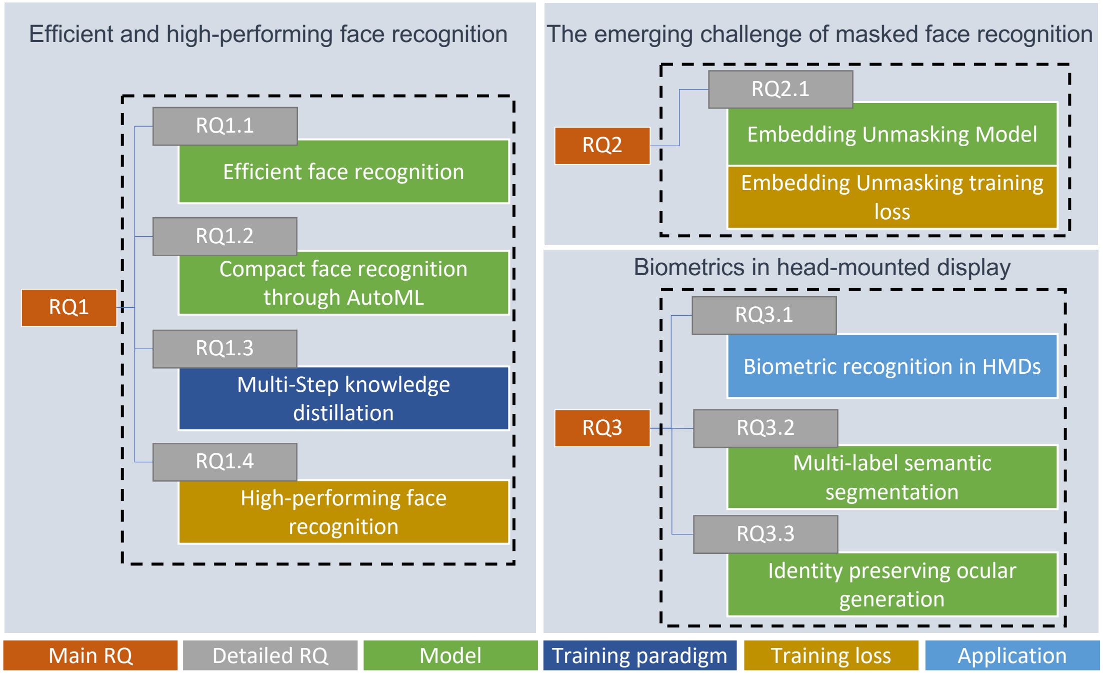

Efficient and high-performing face recognition
RQ1
RQ1.1
Efficient face recognition
RQ1.2
Compact face recognition through AutoML
RQ1.3
Multi-Step knowledge distillation
RQ1.4
High-performing face recognition
The emerging challenge of masked face recognition
RQ2
RQ2.1
Embedding Unmasking Model
Embedding Unmasking training loss
Biometrics in head-mounted display
RQ3
RQ3.1
Biometric recognition in HMDs
RQ3.2
Multi-label semantic segmentation
RQ3.3
Identity preserving ocular generation
Main RQ
Detailed RQ
Model
Training paradigm
Training loss
Application

Figure 1.2.: An overview of the principal and detailed research questions posed in this thesis grouped by the research area.

research. A summary of the main contributions of this thesis is also presented in this chapter.

1.4. Summary

This chapter provided the motivation and identified the challenges leading to the set of research questions posed in this thesis. These research questions are mainly motivated by enabling a wider deployment of biometric solutions in embedded domains and other use-cases constrained by computational resources. The research questions are grouped into three categories based on the targeted challenges, efficient and high-performing FR, the emerging challenge of masked FR, and biometrics in head-mounted displays (Figure 1.2).

The research questions concerned with enabling FR solutions in low computational capability domains stresses the need to propose novel yet accurate FR models that can be operated within such environments. The focus of the second group of research questions targets the emerging challenge of masked FR. Consequently, it raises the need for designing novel solutions to reduce the negative effect of a masked face on FR performance. The third group of research questions is concerned with introducing biometric recognition to VR/AR applications enabled by HMD devices.

2. Background

The previous chapter presented a motivation for the research questions posed in this thesis. This chapter provides the essential background information and definitions needed to comprehend the investigations in the following chapters. This chapter presents first the formal definitions for biometric systems and their main components. Then, this chapter presents biometric performance metrics commonly used in the literature, including biometric recognition performance metrics and computational cost metrics. Finally, this chapter presents a detailed insight into deep FR network architectures and the mainstream benchmarks.

2.1. Biometric systems

The growing demands for reliable and accurate identification and verification solutions in many government and commercial applications are the essential aspects that have driven the extraordinary evolution in biometric recognition technology over the past years [126, 3]. Biometric recognition is the automated recognition of individuals based on their biological or behavioral characteristics such as iris, periocular, fingerprint, and face [122]. This section presents first the biometric recognition systems components. Then, the biometric system operation modes are presented.

2.1.1. Components of biometric recognition system

This section presents the biometric recognition pipeline and main components based on the definition of biometric system components in ISO/IEC 19795-1 standard [123]. Biometric samples are acquired from a subject by a sensor, e.g. a camera (Figure 1.1). This process is part of the data capture subsystem. The output of the data capture subsystem (signal) is sent to the signal processing subsystem via the transition subsystem. The signal processing subsystem extracts the feature from the biometric sample. The input of the signal processing subsystem is a biometric sample, and the output is distinctive features. The signal processing subsystem usually involves preprocessing, feature extraction, and quality control modules. The prepossessing module refers to the enhancement and segmentation processes of the signal processing subsystem [123]. Feature extraction module derives repeatable and distinctive features from the captured biometric sample. Feature extraction module could be run on a local server, cloud, or embedded device (e.g. mobile device). The signal processing subsystem may involve a quality control module that assesses the suitability of a biometric sample for biometric recognition [122, 256]. Based on the operation mode, i.e. enrolment or the operation of verification and identification, the extracted features are sent to either the data storage subsystem or comparison subsystem. In the case of enrolment, the signal processing subsystem produces a biometric reference and sends it to the data storage subsystem. In the case of the operation of verification and identification, the signal processing subsystem produces a biometric probe and sends it to the comparison subsystem. The data storage subsystem stores the biometric reference in the enrolment database. Reference may store in portable devices such as a mobile device, local server, or could. The comparison subsystem involves a process that compares probe(s) against reference(s) and produces comparison scores. These comparison scores are then passed to the decision subsystem. In the verification operation, a single probe is compared to a single reference. In the identification operation, a probe is compared to all references or a subset of references. In verification and identification operations, the comparison scores indicate the similarity/dissimilarities between the compared pair(s). The decision subsystem processes a comparison score to produce a decision based on the verification or identification operation. In the verification scenario, a match or non-match decision is obtained based on a defined threshold, i.e. a comparison score is higher than a defined threshold. In the identification scenario, identification decision is produced based on 1) comparison score is higher than a defined threshold 2) and/or the comparison score is ranked within a predefined number of ranked values.

2.1.2. Operation modes

The biometric system involves three operation modes: enrolment, verification, and identification [123, 3]. In enrolment mode, the biometric sample of an individual is captured by the data capture subsystem and then processed by the signal processing subsystem to generate and store an enrolment template (reference) with the associated identity information in the data storage subsystem.

Biometric verification uses biometrics information to verify a positive identity claim. The biometric sample is captured and then processed by the single processing subsystem to generate a biometric template (probe). The biometric reference is usually associated with identity information, and the system uses this information to retrieve the corresponding reference template from the data storage subsystem. Then, the biometric system verifies that the claimed identity belongs to the individual by comparing the probe template with the stored reference one.

Biometric identification attempts to identify an individual based on the captured biometric characteristics. Unlike the verification mode where probe template is compared to one reference, biometric identification requires comparing the probe template to all reference ones in the enrolment database. Therefore, the verification is a 1:1 comparison process, and identification is a 1: N comparison process, where N is the number of the enrolled subjects.

2.2. Performance metrics

This section provides performance metrics for evaluating the biometric recognition system. This section presents first the performance metrics recommended by ISO/IEC 19795-1 [123]. Noting that most of the biometric recognition works presented in literature did not follow the ISO/IEC 19795-1 [123] terms for algorithmic level evaluation. For the sake of comparability and reproducibility, most of the biometric recognition works presented in the literature follow the evaluation metrics used in the utilized benchmarks and the previous works reporting on them. Therefore, this section also presents the main evaluation metrics used in the literature and link them to the evaluation metric based on ISO/IEC 19795-1 [123] terms. Lastly, this section presents metrics for estimating the computational cost of deep learning-based models. There is no definition for estimating the computational cost of deep learning-based systems in the international standard ISO/IEC 19795-1 [123]. Therefore, the derived computational cost metrics are based on practice and reported metrics in the literature.

2.2.1. Biometric recognition performance metrics

The ISO/IEC 19795-1 standard [123] provided a set of metrics for evaluating a biometric system. A subset of these metrics targets algorithmic level evaluation. The performance of the biometric acquisition process is reported as Failure to Acquire Rate (FTAR). FTAR is the proportion of verification or identification attempts for which the system fails to capture or locate a sample of sufficient quality. FTAR is a combination of failures of the capture process and failures of the feature extraction process. The failure of the capture process is reported as Failure to Capture Rate (FTCR), which is a proportion of failures of the biometric capture process to produce a captured biometric sample. The failure of the feature extraction process is reported as Failure to Extract Rate (FTXR), which is a proportion of failures of the feature extraction process to generate a template from the captured biometric sample.

Algorithmic level evaluation assumes that FTXR and FTCR are zeros. In the case of verification operation, the verification performance on algorithmic level evaluation depends on False Non-match Rate (FNMR) and False Match Rate (FMR) metrics. FNMR is the proportion of genuine attempt samples falsely declared not to match the template of the same characteristic from the same user supplying the sample. FMR is the proportion of zero effort impostor attempt samples falsely declared to match the compared non-self template. Both FMR and FNMR metrics are functions of the system operation threshold that control the trade-off between these metrics. Thus, it is common to report the verification performance in terms of FNMR at different decision thresholds by reporting the lowest FNMR at fixed FMR [94]. Another common metric to report the biometric verification performance is Equal Error Rate (EER). EER is the FNMR or the FMR at the operation point where they are equal. On the system level evaluation, the verification performance depends on False Reject Rate (FRR) and False Acceptance Rate (FAR) metrics. FRR and FAR correspond to the FNMR and FMR, respectively, on the system evaluation level. The FRR is given by:

$$FRR = FTAR + FNMR \times (1 - FTAR), \tag{2.1}$$

and the FAR is given by:

$$FAR = FMR \times (1 - FTAR). \tag{2.2}$$

Detection error trade-off (DET) and Receiver Operating Characteristic (ROC) curves are verification performance visualization plots that show the performance at all decision thresholds. DET curve plots false negative (y-axis) vs. false positive (x-axis), i.e. FNMR vs. FMR. ROC curve plots true positive (y-axis) vs. false positive (x-axis), i.e. 1- FNMR vs. FMR.

The biometric evaluation in the identification case varies between closed-set and open-set identification scenarios. The primary measure of closed-set identification performance is the cumulative match characteristic curve (CMC), in which the (true positive) identification rate at rank r is plotted as a function of r. In open-set identification, the main metric for reporting the identification performance is the false-negative identification error rate (FNIR) at fixed false-positive identification error rate (FPIR). These two errors can be estimated from the verification error metrics as:

$$FNIR = FTAR + (1 - FTAR) \times FNMR, \tag{2.3}$$

and

$$FPIR = (1 - FTAR \times (1 - (1 - FMR)^N))), \tag{2.4}$$

where N is the number of samples in the reference dataset.

The reported evaluation results in the literature on mainstream benchmarks do not always follow the ISO standard evaluation metrics. In general, several benchmarks opt to include the ROC curve as an evaluation metric [114, 296, 192]. In addition to the ROC curve, several benchmarks [114, 296, 192] additionally report the performance in terms of accuracy (Acc) as follows:

$$Acc = (TP + TN)/(N), \tag{2.5}$$

where true positive (TP) is the number of correctly match genuine pairs i.e. 1-FNMR, true negative (TN) is the number of non-match imposter pairs i.e. 1 - FMR, and N is the total number of comparison. Other benchmarks such as the IARPA Janus Benchmark-B IJB-B [274], report the verification performance for 1:1 verification protocol as True Acceptance Rate (TAR) i.e. 1- FRR at fixed FAR). Because FTAR is assumed to be zero in the IJB-B benchmark, the TAR and FAR, in this case, refer to the 1-FNMR and FMR in the international standard ISO/IEC 19795-1 [123].

2.2.2. Computational cost metrics

Recent state-of-the-art biometric solutions utilized deep neural networks as a feature extraction module. In resource-constrained environments by memory footprint and computational complexity such as edge deployments, estimating the required resources by deep neural networks is essential to enable biometric recognition systems in such environment. However, there are no standard metrics to estimate the computational cost of biometric solutions. Thus, the computational cost of deep learning models in this thesis is estimated as:

1. Required memory footprint: The required memory footprint of deep neural networks can be estimated by multiplying the number of parameters by b-bit precision used to represent each parameter.

2. Computational complexity: The computational complexity in this thesis is reported based on Floating Point Operations (FLOPs), i.e. the number of multiplication and addition in a single feed-forward phase.

These metrics are chosen based on: 1) These metrics are commonly reported by the recent efficient biometric apporaches in the literature [81, 179, 150, 284]. Therefore, for the sake of comparability with previous works, these metrics are reported in this thesis when it is feasible. 2) Unlike hardware-related metrics such as Floating Point Operations

Per Second (FLOP/S), the number of parameters and the FLOPs are independent of the underlying hardware.

Convolutional neural networks are one of the main classes of deep learning methods used in various computer vision tasks, including the biometric feature extraction model [42, 80, 159]. The following section presents details on the main building layers of convolutional neural networks along with computation cost estimation for each layer.

2.3. Convolutional neural networks

Convolutional neural networks are a class of deep neural networks (DNN). CNN has been tremendously applied to various computer vision tasks such as FR, image classification, object detection, and image segmentation. Three layers are commonly used to build CNN: Convolutional Layer, Pooling Layer, and Fully-Connected Layer. A typical CNN-based feature extraction network consists of several convolutional and pooling layers followed by fully connected layers. This section presents an overview of the main components of CNN and the computational cost of each of them.

Convolutional Layer: Convolutional Layer is the core building block of CNN. A basic convolutional layer consists of linear operation (convolution) and nonlinear operation (activation function). Convolutional layer (C_i) parameters consist of a set of kernels (k). i is the layer index. Each filter has size of $k_w \times k_h \times c_{i-1}$, where k_w, k_w and c_{i-1} are the kernel width, height and depth, respectively. The input of convolutions layer is an image or feature maps of size $w_{i-1} \times h_{i-1} \times c_{i-1}$ and the output is the extracted feature maps. The output of the convolutional layer is calculated by sliding each filter across the input volume. Then, the element-wise product between each filter and the input at any position is computed. Each filter produces feature activation maps. The final output of the convolutional layer is obtained by adding the bias factors to each feature activation map and then aggregating them. The size of the convolutional layer depends on three hyperparameters: number of filters (filters depth), padding, and stride. The stride hyperparameter s specifies the step size by which the kernel slides over the input volume. For example, when the stride is equal to one, the filter is moved by one pixel at each sliding. The padding hyperparameter p specifies whether the input volume is padded with a specific value around the border. The convolutional layer C_i process input of size $h_{i-1} \times w_{i-1} \times c_{i-1}$ to produce feature maps of size $h_i \times w_i \times c_i$, where $h_i = (h_{i-1} - k_h + 2p)/s + 1$ and $w_i = (w_{i-1} - k_w + 2p)/s + 1$.

Number of Parameters: The number of learnable parameters of convolutional layer C_i depends on the kernel size k, number of channel c_{i-1} of the input volume, and the number of kernel (depth) c_i . Given an input image/feature map with depth (channel) of c_{i-1} and convolution layer with $k_w \times k_h$ kernel and depth (number of kernels) of c_i, the number of learnable parameters of this layer is given by:

$$P(C_i) = k_w \times k_h \times c_{i-1} \times c_i + B, \tag{2.6}$$

where B is the bias factor and it is equal to the layer depth c_i.

FLOPs The FLOPs of convolutional layer C_i is defined as [189]:

$$FLOPs(C_i) = 2 \times h_{i-1} \times w_{i-1}(k_w \times k_h \times c_{i-1} + 1) \times c_i, \tag{2.7}$$

where c_{i-1}, k_w and k_h are the depth, width and height of the input volume and c_i is the depth of layer C_i.

Pooling layer: A Pooling layer reduces the spatial size of the input feature maps. Thus, it reduces the number of parameters and computation in the CNN. It is commonly inserted in-between successive convolutional layers. The common procedures for pooling layer are average (Avg-Pooling) and maximum (Max-Pooling) pooling. Avg-Pooling computes the average values over an $f \times f$ neighborhood in each feature map. Max-Pooling computes the maximum values over an $f \times f$ neighborhood in each feature map. Similar to the convolutional layer, pooling layer requires stride hyperparameter. The pooling layer processes input of size $h_{i-1} \times w_{i-1} \times c_{i-1}$ to produce output of size $h_i \times w_i \times c_i$, where $h_i = (h_{i-1} - f)/s + 1$, $w_i = (w_{i-1} - f)/s + 1$ and $c_i = c_{i-1}$.

Number of parameters The pooling layer is parameter-free as the pool size $f \times f$ and stride (s) are hyperparameters.

FLOPs: The FLOPs count of pooling layer PL_i can be computed as:

$$FLOPs(PL_i) = h_i \times w_i \times c_i \times f \times f. \tag{2.8}$$

Fully Connected Layer (FC): The FC layer connects each neuron to all activations in the previous layer. The FC is commonly used as the final layer of the feature extraction model to obtain the biometric template [234, 253, 80, 27]. The output of the FC layer is computed as a matrix multiplication between the input of FC and its weights and then adding a bias offset.

Number of parameters: The number of parameters of FC_i layer is given by:

$$P(FC) = (n_{i-1} + 1) \times m_i), \tag{2.9}$$

where n_{i-1} is the input size and m_i is the number of the output neurons.

FLOPs: The FLOPs count of FC_i layer can be calculated as:

$$FLOPs(FC_i) = 2 \times (n_{i-1} \times m_i) + m_i. \tag{2.10}$$

2.4. Face recognition

FR is one of the widely used biometric recognition systems due to its contactless nature and the high accuracy achieved by FR algorithms.

The conceptual design and principal components of FR systems are inherent from the biometric systems described in Section 2.1. The output of the data capture subsystem, face image, is sent to the signal processing subsystem. The preprocessing module of the signal preprocessing subsystem commonly contains face and landmark points detectors to align and crop the face inside the images. Face detector [289] is used to locate the signal of the subject's face within the received sample from the data capture subsystem. Then, a facial landmark detector [289] is used to allocate the facial landmark points. Once the face is aligned and cropped, feature extraction module is used to extract distinctive features from the face samples. Recently, high-performing FR models used deep neural networks as feature extraction modules, which is one of the main focuses of this thesis. The comparison subsystem of FR usually uses a cosine similarity [80] (an inverse of cosine distance) or euclidean distance [234] to obtain the comparison score between probe and reference. In normalized embedding space the euclidean distance is equivalent to cosine similarity.

The following section presents an overview of the main feature extraction architectures used in FR along with the training loss functions and the mainstream benchmarks.

2.4.1. Deep face recognition

Over the last years, a constant trend in deep learning-based FR models is towards deeper and larger convolutional neural networks [159, 234, 208] and commonly adopted network architectures from the ones designed for image classification [107, 252, 241].

Network architectures

The main network architectures of high-performing FR models are designed based on the common architectures used in image classification including AlexNet [144], GoogLeNet [252], VGGNet [241], ResNet [107], and SENet [110]. These network architectures are briefly described in the following with computational cost of the each of them when it is feasible.

AlexNet [144] was one of the earliest efforts that popularized CNN in computer vision and it was the winner of ImageNet large-scale competition (ILSVRC) 2012. AlexNet has 60M parameters, and it consists of five convolutional layers and three fully connected layers. DeepFace [253] and DeepID series [249, 248] were the pioneer works that proposed to use deep neural networks for FR. These works followed the design choice of AlexNet, i.e. using deeper and wider CNN than LeNet, ReLU activation function, and dropout as regularization methods to propose FR networks. DeepFace [253] architecture consists of nine-layer with more than 120 million parameters. DeepID [249] is based on the training of 60 ConvNets, each of which is trained on different patches (region) of face images. Each ConvNet consists of four convolutional layers (with max-pooling) followed by the fully connected to obtain two 160-D face embedding.

GoogleNet [252] was the winner of the ImageNet competition (ILSVRC) 2014. The main contribution of GoogleNet was the development of an Inception Module and utilizing global average pooling instead of FC layer at the top of the network that significantly reduced the number of parameters. GoogleNet contains 4M parameters in comparison to 60M paramters in AlexNet. The idea of an Inception Module is to utilize different convolution layers with different kernel sizes (1×1, 3×3 and 5×5) inside the convolutional block to capture spatial information at different scales. Additionally, GoogleNet adds a bottleneck layer of 1×1 kernel to regulate the computation cost before applying a convolutional layer with a large kernel (3×3 and 5×5). FaceNet [234] used GoogleNet [252] architecture to train FR model with triplet loss. The utilized architecture by FaceNet [234] is almost identical to GoogleNet [252] with slightly differences i.e. using L_2 pooling instead of max-pooling and adding an FC layer with 128-D at the top of the network to obtain the final face embedding. The presented architecture by FaceNet [234] contains 7M of trainable parameters with 1600 MFLOPs.

VGGNet [241] was runner-up in the ImageNet competition (ILSVRC) 2014. VGGNet proposed a very deep homogeneous architecture that only utilizes 3×3 convolutions and

2×2 pooling layers from the beginning to the end of the network. On the top of the network, VGGNet added two FC layers(4096-D). VGGFace [208] FR network architecture is based on VGGNet [241] with 145M parameters and 30967 MFLOPs.

ResNet [107] is one of the most widely used CNN architecture. It was the winner of the ILSVRC 2015 challenge. The network architecture is based on an identity shortcut connection (residual connection) that skips one or more layers. The skip connection adds the input of a residual block to its output and passes it to the following residual block. The powerful representation ability of ResNet has motivated several computer vision tasks other than classification, such as object detection and FR [80]. ResNet has been widely adopted for FR [80]. SphereFace [159], CosFace [268] and RingLos [297] used 64-Layer ResNet architecture to train FR model. ResNet-64 architecture contains 48.3M parameters with 12227 MFLOPs. 512-D FC layer is used on top of ResNet-64 to obtain the final face embedding. ResNet100 and ResNet50 are the most widely adopted CNN architecture for FR [11, 80, 139, 117, 156, 292, 250]. ResNet100 and ResNet50 contain 65.2M and 43.5M parameters with 24211 and 12639 MFLOPs, respectively. The major difference between the ResNet utilized for FR and the one proposed in the original work [107] is the use of an FC layer on the top of the network instead of using a global average pooling layer as a feature extraction layer.

Squeeze and Excitation Network (SENet) introduced a new building block, namely Squeeze-and-Excitation (SE) block. The main idea of SE-Block is to learn channel attention for each convolution block, which brings performance gain for various CNN architectures e.g. ResNet [107] and VGGNet [241]. SENet was first adopted for FR by VGGFace2 [42]. VGGFace2 used the last average pooling layer of SENet (2048-D) to obtain the final face representation. VGGFace2 contains 26M parameters with 7749 MFLOPs.

Loss functions

In addition to the evolution of deep network architectures, training losses are behind major advances in achieving accurate FR [150, 268, 117, 185]. The loss function used to train FR models can be categorized into metric-based learning [50, 234, 273] and classification loss [80, 27, 268, 159].

The training objective of metric-based learning loss is to guide the network to directly optimize the embedding space in which pairs of the same identity stay close to each other while the ones of different identities are far apart. Contrastive loss [50] was one of the earliest metric-based learning losses. Contrastive loss processes pairs of embeddings. The

Model	Year	Training Dataset	Architecture	Loss	Param. (M)	MFLOPs	LFW Acc (%)	AgeDB-30 Acc (%)
DeepFace[253]	2014	Facebook [253]	Alexnet [144]	classification	120	-	97.35	-
DeepID2[248]	2014	Celebfaces+ [248]	Alexnet [144]	classification	-	-	99.15	-
Facenet[234]	2015	Google [234]	GoogleNet [252]	metric-based learning	7.5	1600	99.63	-
VGGFace[208]	2015	VGGface [208]	VGGNet-16 [208]	classification + metric-based learning	145	30967	98.95	-
L-Softmax [160]	2016	CASIA [286]	VGGNet-16 [208]	classification	145	145	98.71	-
Center Loss[273]	2016	CASIA [286]	-	metric-based learning	-	-	98.8	-
SphereFace[159]	2017	CASIA [286]	Resnet-64 [107]	classification	48.3	12227	99.42	-
VGGFace2 [42]	2018	VGGFace2 [42]	SE-ResNet [110]	classification	26	7749	98.95	-
ArcFace [80]	2019	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.82	98.15
CosFace [268]	2018	Private	Resnet-64 [107]	classification	48.3	12227	99.73	-
Dynamic-AdaCos [292]	2019	MS-Celeb-1M [103]	Resnet-50 [107]	classification	43.5	12639	99.73	-
AdaptiveFace [156]	2019	MS-Celeb-1M [103, 281]	Resnet-50 [107]	classification	43.5	12639	99.62	-
GroupFace [139]	2020	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.85	98.28
CircleLoss [250]	2020	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.73	-
CurricularFace [117]	2020	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.80	98.32
Partial-FC [11]	2021	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.83	98.20
Dyn-arcFace [128]	2021	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.80	97.76
MagFace [185]	2021	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.83	98.17
ElasticFace [27]	2021	MS1MV2 [80, 103]	Resnet-100 [107]	classification	65.2	24211	99.82	98.35

Table 2.1.: Overview of the high-performing FR approaches. The recent SOTA FR models used MS1MV2 [80, 103] to train ResNet-100 [107] using classification loss i.e, softmax loss or its variants. The difference between models that used classification loss is the deployed margin-penalty on the feature embeddings and their corresponding class centers. The number of parameters (Param), the FLOPs, and the accuracy on Labeled Faces in the Wild (LFW) and AgeDB-30 are reported for each model when it is feasible.

training objective of contrastive loss is to minimize the distance between embeddings of the same identity and maximize the embedding distance when they are of different identities. Triplet loss is another metric-based learning loss proposed by [234]. Training with triplet loss requires a triplet of samples (anchor, positive and negative). An anchor and positive are two different samples of the same identity (genuine pair), while a negative is a sample belonging to a different identity (imposter). The learning objective of the triplet loss is that the distance between genuine pair embeddings with the addition of a fixed margin value (m) is smaller than the distance between the face embedding (anchor) and any face embedding of any other identities (imposter). Center loss [273] proposed to minimize the distance between feature embeddings of each sample and their class center by leaning the center of each identity and pushing each sample to be close to its class center.

The widely used multi-class classification loss, softmax loss [160], refers to applying cross-entropy loss between the output of the logistic function (softmax activation function) and the ground-truth. Margin-penalty softmax loss is the most widely adopted variant

Dataset	Identities	Images /(videos frames)	Metrics
LFW [114]	5.7K	13.2K	1:1 Accuracy, ROC
CA-LFW [296]	5.7K	11.6K	1:1 Accuracy, ROC
CP-LFW [295]	5.7K	12.1K	1:1 Accuracy, ROC
AgeDB-30 [192]	568	16.4K	1:1 Accuracy
CFP-FP [236]	500	7K	1:1 Accuracy, EER, ROC
IJB-B [274]	1.8K	76.7K	1:1 TAR at FAR1e-4
IJB-C [183]	3.5K	148.3K	1:1 TAR at FAR1e-4
MegaFace [136]	690K	1.1M	Rank-1, 1:1 TAR at FAR1e-6

Table 2.2.: Mainstream face recognition benchmarks.

of softmax loss functions for training FR models due to its state-of-the-art performance on mainstream benchmarks [27, 159, 268]. Margin-penalty softmax losses proposed to push the decision boundary of softmax, and thus enhance intra-class compactness and inter-class discrepancy by deploying a margin penalty between the feature embedding and their corresponding class centers.

Table 2.1 presents an overview of the high-performing FR approaches. The constant trend in these models is the use of overparameterized network. The majority of the recent high-performing FR models are trained with classification loss [80, 268, 11, 27] i.e. softmax loss and its variants, while earlier works such as FaceNet [234] and VGGFace [208] are trained using metric-based learning loss.

2.4.2. Evaluation benchmarks

This section presents the mainstream face benchmarks used in the literature as well as in this thesis.

Face recognition performance evaluation datasets

Table 2.2 summarizes the mainstream FR benchmarks proposed in the literature and used in this thesis to evaluate the FR verification performances. Each of these datasets (Table 2.2) is briefly described in the following.

LFW [114] : LFW is an unconstrained face verification dataset. The LFW contains 13,233 images of 5749 identities collected from the web [1].

[1] http://vis-www.cs.umass.edu/lfw/

AgeDB-30 [192]: AgeDB is an in-the-wild dataset for age-invariant face verification evaluation, containing 16,488 images of 568 identities. Every image is annotated with respect to the identity, age, and gender attribute. AgeDB-30 (30 years age gap) as is the most reported and challenging subset of AgeDB [2].

Celebrities in Frontal-Profile in the Wild (CFP-FP) [236]: CFP-FP [236] dataset addresses the comparison between frontal and profile faces. CFP-FP dataset contains 7,000 images across 500 identities, where 10 frontal and 4 profile image per identity [3].

Cross-Age LFW (CA-LFW) [296]: The CA-LFW dataset [296] is based on LFW with a focus on comparison pairs with the age gap, however not as large as AgeDB-30. Age gap distribution of the CA-LFW is provided in [296]. It contains 3000 genuine comparisons, and the negative pairs are selected of the same gender and race to reduce the effect of attributes [4].

Cross-Pose LFW (CP-LFW) [295]: The CP-LFW dataset [295] is based on LFW with a focus on comparison pairs with pose differences. CP-LFW contains 3000 genuine comparisons, while the negative pairs are selected of the same gender and race [5].

IJB-B [274]: The IARPA Janus Benchmark-B (IJB-B) dataset contains 21,798 still images and 55,026 frames from 7,011 videos of 1,845 identities [274]. The standard 1:1 verification protocol with 10,270 genuine and 8M impostor comparisons [274] [6].

IJB-C [183]: The IARPA Janus Benchmark-C (IJB-C) [183] is a video-based FR dataset provided by the Nation Institute for Standards and Technology (NIST). It is an extension of the IJB-B [274] dataset with a total of 31,334 still images and 117,542 frames of 11,779 videos across 3531 identities [7].

[2] https://ibug.doc.ic.ac.uk/resources/agedb/
[3] http://www.cfpw.io/
[4] http://whdeng.cn/CALFW/
[5] http://whdeng.cn/CPLFW/
[6] https://www.nist.gov/programs-projects/face-challenges
[7] https://www.nist.gov/programs-projects/face-challenges

MegaFace [136] and MegaFace(R) [80, 136]: The MegaFace benchmark [136] includes a gallery of 1m Flicker images (of 690K identities) and a probe of FaceScrub [199] images (100K images of 530 identities). The MegaFace benchmark reports the FR performance as Rank-1 correct identification rate and as TAR at FAR equal to 1e–6 verification accuracy [8]. The MegaFace (R) [80] benchmark is a refined version of MegaFace (refined in [80]) and reports the same evaluation metrics as MegaFace [9].

Ocular dataset

This work uses OpenEDs [96] to train and evaluate the ocular segmentation and recognition approachs presented in Chapter 5.

OpenEDs [96]: OpenEDs is large-scale ocular images dataset captured using a virtual-reality HMD device with two eye-facing cameras at a frame rate of 200 Hz under controlled illumination. . OpenEDs contains three different sub-datasets: generation, semantic segmentation and sequence sets. The semantic segmentation dataset included 12759 images of 152 individuals with pixel resolution of 640×400. The data is split into 8916 pairs of eye images for training, 2403 images for validation and 1440 images for test as described in [96]. The generation data includes 152 subjects and 12759 images of 640x400 pixel resolution. The data is split into identity-disjoint training, validation, and testing splits as described in [96]. The sequence dataset contains 91200 images from contiguous 1.5 second video snippets with pixel resolution of 640x400 [96].

Masked face datasets

Two masked face datasets, MRF2 [13] and MFR [64, 62], are used to evaluate the proposed approach in Chapter 4.

MRF2 [13]: The Masked Faces in Real World for Face Recognition (MRF2) dataset contains 269 images of 53 identities crawled from the internet to evaluate the masked face verification performance.

MFR [64, 62]: The Masked Face Recognition (MFR) simulates a collaborative yet varying scenario. Such as the situation in automatic border control gates or personal unlocking devices with FR, where the mask illumination and background can change. MFR contains

[8] http://megaface.cs.washington.edu/
[9] https://insightface.ai/

4320 images collected from 48 participants using their webcams on three different, not necessarily consecutive days (session). Each session contains masked and unmasked face image captures.

2.5. Summary

This chapter discussed the biometric systems and their main components. Moreover, it presented the biometric performance metrics including recognition performance and computational cost metrics. An overview of the high-performing deep FR architectures and training loss functions was discussed. A description of the mainstream evaluation benchmarks was presented. Next chapters will investigate in more details the response to the research questions stated in Chapter 1.

3. Efficient and high-performing face recognition

The previous chapter provided essential background knowledge for biometric recognition systems and their components, along with the performance evaluation metrics, including those measuring the performance of biometric recognition, as well as the computational costs. This chapter investigates designing efficient and yet accurate FR models. This chapter presents two sets of efficient FR networks as response to RQ1.1 (Section 3.3), RQ1.2 and RQ1.3 (Section 3.4). Then, this chapter presents a novel FR loss based on margin-penalty softmax as a response to RQ1.4 (Section 3.5). This chapter is based on [23, 38, 27].

3.1. Introduction

FR is an active research field, and it has benefited from the recent advancements in machine learning, especially the advancements in deep learning [107] and the novelty of margin-based Softmax losses [80, 268], achieving a notable recognition accuracy.

Recent SOTA FR models rely on deep learning models with an extremely large number of parameters [80, 185]. Deploying such models on embedded devices or in applications with limited memory specifications is a major challenge [180, 81], due to the limited resources in such environments. This challenge has received increased attention in the literature in the last few years [180, 81]. Over the past few years, several compact FR models have been proposed in the literature. MobileFaceNet [49] proposed an efficient FR model based on MobileNetV2 [233] with around 1M parameters. ShuffleFaceNet [179] and VarGFaceNet [284] model architectures adopted ShuffleNetV2 [173] and VarGNet [290], respectively, for the FR task. VarGFaceNet contains 5M parameters. ShuffleFaceNet presented three architectures with different width scales (0.5, 1.5 and 2) containing 0.5, 2.6, and 4.5M parameters, respectively. Martinez-Diaz et al. [180] evaluated the computational requirements and the verification performance of five compact model architectures including MobileFaceNet (2.0M parameters), VarGFaceNet [284]

(5M parameters), ShufeFaceNet [179] (2.6/M parameters), MobileFaceNetV1 (3.4M parameters), and ProxylessFaceNAS (3.2M parameters) [180]. The reported results by Martinez-Diaz et al. [180] demonstrated that compact FR models can still achieve high accuracies for FR.

Besides the tremendous advances in deep network architectures, training losses are behind the significant advances in achieving accurate FR. Learning discriminative features for FR models was the main focus of the recent SOTA FR solution proposed in the literature [27, 268, 159, 80]. Margin-penalty softmax loss and its variants are the most widely studied and adopted loss functions for training FR networks [27, 268, 159, 80]. This is mainly because of their SOTA performance on mainstream benchmarks.

With a focus on achieving efficient FR, this chapter presents a family of extremely efficient FR networks, MixFaceNets [23], for accurate face verification. Extensive experiment evaluations on mainstream benchmarks have shown the effectiveness of the proposed MixFaceNets for applications restricted by computational complexity. Under the same level of computation complexity ($\leq$ 500M FLOPs), MixFaceNets outperform recent efficient FR models proposed in the literature on all the evaluated datasets. With computational complexity between 500M and 1G FLOPs, MixFaceNets achieved results comparable to the top-ranked models while using significantly fewer FLOPs and less computation overhead, proving practical value MixFaceNets. The achieved results by MixFaceNets provide an answer to RQ1.1.

Previous compact FR models have been adopted from the ones designed for common computer vision tasks, and none of them designed a network specifically for FR. This chapter presents PocketNet, one of the earliest efforts proposed to utilize NAS to design a FR network. Additionally, PocketNet [38] proposes a novel training paradigm based on KD, the multi-step KD, where the knowledge is distilled from the teacher to the student model at different stages of the training maturity. A detailed ablation study is conducted in this work, proving both the sanity of using NAS for the specific task of FR rather than general object classification, which provides an answer to RQ1.2 and the benefits of the proposed multi-step KD, which provides an answer to RQ1.3. This chapter presents an extensive experimental evaluation and comparisons with the SOTA compact FR models on mainstream benchmarks. PocketNets have consistently advanced the SOTA FR performance on nine mainstream benchmarks when considering the same level of model compactness. With 0.92M parameters, the smallest network, PocketNetS-128, achieved very competitive results to recent SOTA compacted models that contain up to 4M parameters.

This chapter also presents a novel margin penalty-based softmax loss, namely ElasticFace [27]. The recent FR loss functions proposed incorporating a fixed penalty margin on commonly used classification loss function, softmax loss, in the normalized hypersphere to increase the discriminative power of FR models. Marginal penalty softmax losses, such as ArcFace [80], and CosFace [268], assume that the geodesic distance between and within the different identities can be equally learned using a fixed penalty margin. However, such a learning objective is not realistic for real data with inconsistent inter-and intra-class variation, which might limit the discriminative and generalizability of the FR model. The proposed ElasticFace loss in this chapter relaxes the fixed penalty margin constrain by proposing elastic penalty margin loss, allowing flexibility in the push for class separability. The presented EalsticFace loss is used to train ResNet100 [107] network to be compatible with the previous works proposed training loss function. ElasticFace loss outperformed ArcFace and CosFace losses [80, 268], using the same geometric transformation, on a large set of mainstream benchmarks, providing an answer RQ1.4. From a wider perspective, ElasticFace has advanced the SOTA FR performance on seven out of nine mainstream benchmarks.

This chapter is organized as follows: Section 3.2 presents a detailed look into related works to efficient FR models and training loss functions. Section 3.3 presents a set efficient FR networks, MixFaceNets. Section 3.4 presents lightweight and accurate FR models based on NAS, PocketNets. EalsticFace loss is presented in Section 3.5.

3.2. Related work

This section presents and discusses recent efficient FR approaches and FR training losses proposed in the literature.

3.2.1. Efficient face recognition architectures

This section lists out and discuss the recent efforts on designing an efficient deep learning model for FR. The computational cost of the presented approaches is based on the the number of trainable parameters and the FLOPs when it is feasible.

Light CNN [281] was one of the earliest works that presented 3 network architectures for learning compact representation on a large-scale database. Light CNN proposed three different architectures- Light CNN-4, Light CNN-9, and Light CNN-29. Light CNN-4 consists of four convolution layers followed by an FC layer. The Light CNN-4 architecture is based on AlexNet [144] and it contains 4M parameters with 1.5G FLOPs. Light CNN-9 is designed based on Network in Network (NIN) [152] and followed the designed choice of VGGNet [241] by utilizing a small kernel size. Light CNN-9 consists of 9 convolution layers followed by an FC layer with 5.5M parameters and 1G FLOPs. Light CNN-29 included residual block [107] to design a network with 29 convolution layers followed by an FC layer. Light CNN-29 contains 12.6M parameters with 3.9G FLOPs. Light CNN also proposed a new activation function Max-Feature-MAP (MFM), an extension of maxout activation, and incorporated it in all proposed architecture. Compared to the recent efficient FR models [180], Light CNN architecture is considered computational expensive.

ShiftNet [279] proposed shift-based modules as an alternative to spatial convolutions and it is adopted for the FR task. The presented face model contains 0.78M parameters.

MobileFaceNets [49] is popular network architecture that has been widely adopted for designing a compact FR solution [150, 179, 81]. MobileFaceNets [49] is based on MobileNetV2 [233]. MobileFaceNet contains around one million of trainable parameters with 443M FLOPs. MobileFaceNets model architecture is based on the residual bottlenecks proposed by MobileNetV2 [233] and depth-wise separable convolutions layer, which allows building CNN with a smaller set of parameters compared to standard CNNs.

Different from MobielNetV2 architecture, MobileFaceNet uses Parametric Rectified Linear Unit (PReLU) [106] as the non-linearity in all convolutional layers and replaces the last global average pooling with linear global depth-wise convolution layer as a feature output layer.

AirFace [150] proposed to increase the MobileFaceNet network width and the depth and adding attention module. The work also presented a loss function named Li-ArcFace which is based on ArcFace. Li-ArcFace demonstrates better converging and performance than ArcFace loss on low dimensional features embedding. The proposed model by AirFac has a computational cost of 1G FLOPs.

VarGFaceNet [284] deployed variable group convolutional network proposed by VarGNet [290] to design a compact FR model with 5M trainable parameters and 1G FLOPs. VarGNet [290] proposed to fix the number of input channels in each group convolution instead of fixing the total group numbers in an effort to balance the computational intensity inside the convolutional block. VarGFaceNet adds squeeze and excitation (SE) block on the VarGNet block, replaces ReLU with PReLU, and uses variable group convolution along with pointwise convolution as the feature output layer.

ShuffleFaceNet [179] is a compact FR model based on ShuffleNet-V2 [173]. ShuffleNetV2 utilizes a channel shuffle operation proposed by ShuffleNetV1 [291], achieving an acceptable trade-off between accuracy and computational efficiency. Channel shuffle operation enables information flowing between different groups of channels by shuffling a group of g channels of the convolution output (i.e. feature map). Channel shuffle operation is parameter-free and it reduces the computational cost by a factor of $1/g$. However, it affects to some degree the latency of the model [173]. Similar to MobileFaceNet [49], ShuffelFaceNet replaces the last global average pooling layer with a global depth-wise convolution layer and a Rectified Linear Unit (ReLU) with PReLU. ShuffleFaceNet presented three architectures with different width scales (0.5, 1.5 and 2) containing 0.5, 2.6, and 4.5M parameters and FLOPs of 66.9M, 577.5M and 1050M, respectively.

In a recent survey by Martinez-Diaz et al. [180], the computational requirements and the verification performance of five lightweight model architectures are analyzed and evaluated. The evaluated models are MobileFaceNet (0.9G FLOPs and 2.0M parameters), VarGFaceNet [284] (1G FLOPs and 5M parameters), ShuffleFaceNet [179] (577.5M FLOPs and 2.6M parameters), MobileFaceNetV1 (1.1G FLOPs and 3.4M parameters), and ProxylessFaceNAS (0.9G FLOPs and 3.2M parameters). MobileFaceNetV1, ProxylessFaceNAS and MobileFaceNet are extended versions of MobileNetV1 [109], ProxylessNAS [41], and MobileFaceNets [49], respectively. The reported evaluation results by [180] showed that ShufeFaceNet, VarGFaceNet, MobileFaceNet achieved very close accuracy on the considered evaluation datasets, while MobileFaceNetV1 and ProxylessFaceNAS achieved slightly lower accuracy.

Among the previous listed works, MobileFaceNets [49] is the only architecture that achieved high accuracy with less than 500M FLOPs. With the almost same number of FLOPs as in MobileFaceNets, MixFaceNets (Section 3.3) outperform MobileFaceNets and achieved competitive results to other models with fewer FLOPs using extremely efficient architecture.

All previous works utilized network architecture designed for common computer vision task. Section 3.4 presents PocketNets, which is one of the earliest efforts to automate the FR network architecture design.

3.2.2. Learning losses

Face recognition training losses can be categorized into metric-based learning and classification losses. Metric-based learning losses guide the model to directly optimize the embedding space by pushing the embeddings of same identity to have smaller distance than the ones of different identities [50] One of the main challenges for training with metric-based learning such as Triple [234], n-pair [244], or contrastive [50] losses, is training the model with a large-scale dataset as the number of possible triplets explodes with the number of samples. Alternatively, classification-based losses such as softmax loss can be easily adopted for training a FR model as it does not pose that issue. However, the softmax loss does not directly optimize the feature embedding needed for face verification. Liu et al. [160] proposed a large-margin softmax (L-Softmax) by incorporating angular margin constraints on softmax loss to encourage intra-class compactness and inter-class separability between learned features. SphereFace [159] extended L-Softmax by normalizing the weights of the last full-connected layer and deploying multiplicative angular penalty margin between the deep features and their corresponding weights. Different from SphereFace, CosFace [268] proposed additive cosine margin on the cosine angle between the deep features and their corresponding weights. CosFace also proposed to fix the norm of the deep features and their corresponding weights to 1, then scaling the deep feature norm to a constant s, achieving better performance on mainstream FR benchmarks. Later, ArcFace [80] proposed additive angular margin by deploying angular penalty margin on the angle between the deep features and their corresponding weights. The great success of softmax loss with penalty margin motivated several works to propose a novel variant of softmax loss [128, 156, 84, 139, 250, 117, 185, 11]. All these solutions achieved notable accuracies on mainstream benchmarks [114, 236, 274, 183] for FR. Huang et al. [117] proposed an Adaptive Curriculum Learning loss based on margin-based softmax loss. The proposed loss targets the easy samples at an early stage of training and the hard ones at a later stage of training. Jiao et al. [128] proposed Dyn-arcface based on ArcFace loss [80] by replacing the fixed margin value of ArcFace with an adaptive one. The margin value of Dyn-arcface is adjusted based on the distance between each class center and the other class centers. However, this might not reflect the real properties of the class separability, but rather their separability in the current stage of the model training. Kim et al. [139] proposed to enrich the feature representation learned by ArcFace loss with group-aware representations. UniformFace [84] suggested to equalize distances between all the classes centers by adding a new loss function to SphereFace loss [159]. A recent work by An et al. [11] presented an efficient distributed sampling algorithm (Partial-FC). The Partial-FC method is based on randomly sampling a small subset of the complete training set of classes for the softmax-based loss function. Thus, it enables the training of the FR model on a massive number of identities. The authors experimentally proved that training with only 10% of training samples using CosFace [268] and ArcFace[80] can achieve comparable results on mainstream benchmarks to the case when training is performed on a complete set of classes. MagFace [185] deployed magnitude-aware margin on ArcFace loss to enhance intra-class compactness by pulling high-quality samples close to class centers while pushing low-quality samples away. However, this is based on the weak assumption of optimal face quality (utility) estimation. Moreover, this might prevent the model from convergence when the most of training samples in the training dataset are of low quality.

The main challenge for the majority of the previously listed works is the fine selection of the ideal margin penalty value. Section 3.5 presents ElasticFace loss that relaxes the fixed single margin value by deploying a random margin drawn from a normal distribution.

3.3. Efficient face recognition

This section presents a set of extremely efficient architectures for accurate face verification and identification, namely the MixFaceNets [23]. The proposed MixFaceNets uses MixNets [254] as a baseline network structure. Additionally, this work carefully designed tailored head and embedding settings suitable for FR. The proposed MixFaceNets also extends the MixConv block with a channel shuffle operation aiming at increasing the discriminative ability of MixFaceNets. With computation complexity of 451M FLOPs, MixFaceNet-S and ShuffleMixFaceNet-S achieved 99.60 and 99.56 % accuracies on LFW [114] and 92.23 and 93.60 TAR (at FAR1e-6) on MegaFace [136] which are significantly higher than the ones achieved by MobileFaceNets [49] with a comparable level of computational complexity (99.55% accuracy on LFW and 90.16% TAR (at FAR1e-6) on MegaFace). Also, MixFaceNets achieve comparable results to the SOTA solutions that have computation complexity of thousands of MFLOPs.

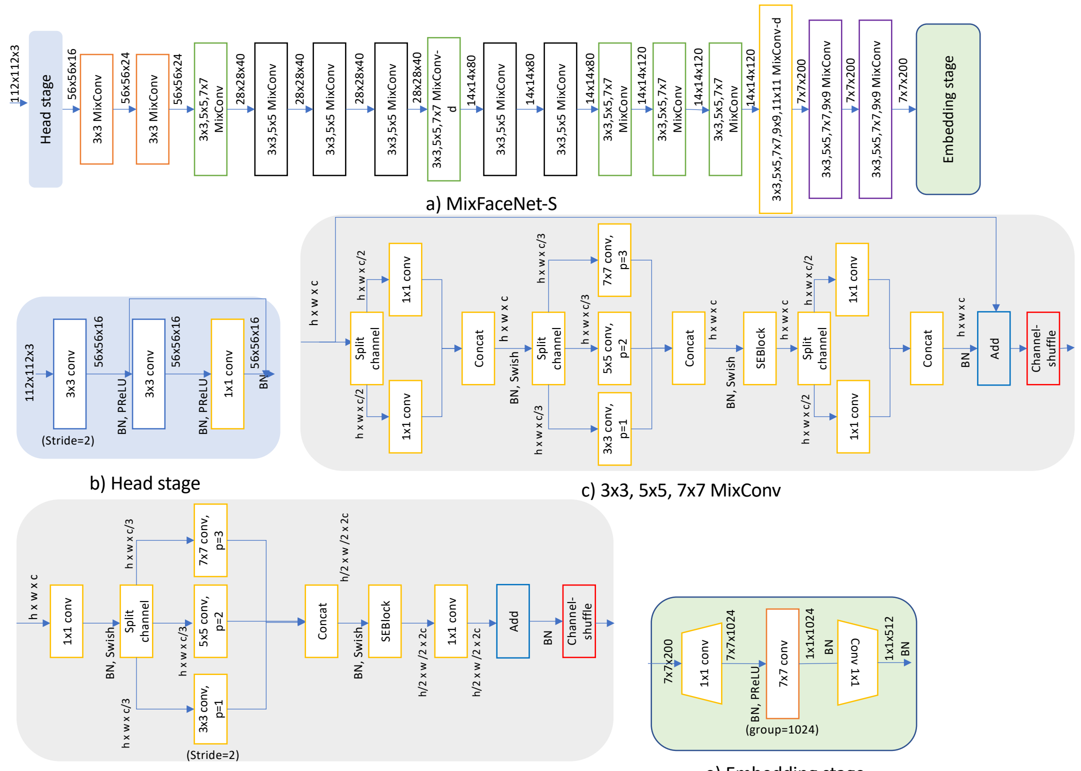
This section presents fist the architecture of MixFaceNets. Later on, the experimental setup along with evaluation details are introduced. This is followed by a detailed comparative discussion of the achieved results in terms of FR performance and computational complexity. Finally, a set of concluding remarks are drawn.

3.3.1. Methodology

This section presents the architecture of MixFaceNets designed for accurate face verification. Figure 3.1 illustrates the architecture of the MixFaceNet, partially inspired by MixNets [254]. To improve the accuracy and the discriminative ability of MixNet, we (a) implement different head settings, (b) introduce channel shuffle operation to the MixConv block, and (c) propose different embedding settings. This section discusses the MixConv as an inspiration to this work. Then, the detailed architecture of MixFaceNets is presented.

Mixed depthwise convolutional kernels

Depthwise Convolution is one of the most popular building block for mobile models [109, 233, 173]. Depthwise convolutional applies a single convolution filter over each channel of input. Thus it reduces the number of parameters and achieves computational efficiency while maintaining the discriminative ability of the convolution [109]. Mixed Depthwise Convolutional Kernels (MixConv) [254] extends vanilla depthwise convolution by using multiple kernel sizes in a single convolution. MixConv depends on mixing up multiple kernel sizes in a single convolution by splitting convolution input into groups and applying different kernel sizes to each group. Unlike vanilla depthwise convolution,



a) MixFaceNet-S

b) Head stage

c) 3x3, 5x5, 7x7 MixConv

d) 3x3, 5x5, 7x7 down sampling MixConv-d block

e) Embedding stage

Figure 3.1.: An overview of the proposed MixFaceNet-S network architecture inspired by MixNets [254]. The input of MixFaceNet-S has size of $112 \times 112 \times 3$ and the output is a face embedding of dimension $512 - d$. b) illustrates the head setting of MixFaceNet-S. The input on the first convolution (stride=2) is downsapled then one residual block is added. d) is the MixConv block with multiple kernel sizes ($[(3, 3,), (5, 5), (7, 7)]$) and channel shuffle operation. All MixConv blocks have the same structure as in (c) and the reduced blocks have the same structure as in (d). e) shows the embedding setting of MixFaceNet-S, where the channel is expanded from 200 to 1024, and then global depthwise convolution is applied to obtain a $512 - d$ embedding. The input and output size, kernel size, stride, and padding (*p*) are shown for each convolution layer.

MixConv can capture different patterns from convolution input at various resolutions. Also, it requires fewer parameters, and it is more computationally efficient than using a single kernel, e.g., using multiple kernels of size $[(3,3),(5,5),(7,7)]$ is more computation efficient than using a single kernel of size 7×7. For example, given a convolution input of size $w \times h \times c$ and multiple kernels of size $[(3,3),(5,5),(7,7)]$, MixConv split the input into 3 groups, each of them has a dimension of $w \times h \times c/3$. Then it uses different kernels for each of these groups. Finally, the three outputs are concatenated to produce the final convolution output. An example of MixConv and downsample-MicConv blocks are shown as part of Figure 3.1. Unlike manually designed mobile models [173, 179, 233], MixConv utilized neural architecture search to develop new series of MixConv-based networks, namely MixNets (MixNet-S, MixNet-M, and MixNet-L). MixNet-S and MixNet-M are developed using neural architecture search, while MixNet-L is obtained by scaling up the number of channels in each block by a factor of 1.3. For details about the network structure and search space, one can refer to the original work [254].

MixFaceNet architecture

This work deploys MixNets [254] as a baseline network structure to develop the proposed MixFaceNets. Figure 3.1 illustrates the network architecture for MixFaceNet-S. For the network head, fast down-sampling in the first 3×3 convolution (stride=2) followed by batch normalization [121] and PReLU non-linearity [106] is applied. Then, one residual block is used as shown in Figure 3.1.b. MixFaceNet-S network uses the same global structure as MixNet-S. However, different from MixNet-S, MixFaceNet-S did not apply down-sampling at the first convolution after the head stage to reserve as much information as possible at the earliest stage of the network. The presented architecture mixes up both channels and kernels to increase the discriminative ability of MixFaceNet and improve the model accuracy. This has been achieved by introducing shuffle operation to the MixConv block. The channel shuffle operation is proposed by [179] to enable information flowing between different groups of channels. A channel shuffle operation with a group value of 2 is applied after each MixConv block. Thus, MixFaceNet can capture high and low-resolution patterns at different scales, and it also enables information communication between various groups of channels. Figures 3.1.c and 3.1.d show the detailed structure of MixConv and downsampling MixConv blocks with the channel shuffle operation. All MiXConv blocks uses swish as an activation function [223] followed by batch normalization. MixConv includes also squeeze-and-excitation (SEBlock) [110] at the end of each block. Finally, to obtain the feature embedding of the input face image, the last global average pooling layer is replaced with global depth-wise convolution, as presented in the next section. This work proposes 3 network architectures, MixFaceNet-XS, MixFaceNet-S, and MixFaceNet-M. MixFaceNet-S network architecture is illustrated in Figure 3.1.a. MixFaceNet-XS is obtained by scaling up MixFaceNet-S with a depth multiplier of 0.5. MixFaceNet-M has the same global network architecture as MixNet-M [254] with the same strategies applied to MixFaceNet-S, i.e., head setting and embedding settings. All proposed MixFaceNets are trained and evaluated with/without channel shuffle operation. The models trained with channel shuffle operation will be noted as ShuffleMixFaceNet-XS, ShuffleMixFaceNet-S, and ShuffleMixFaceNet-M.

Embedding setting

MixNets use global average pooling before the classification layer as a feature output layer. This is common choice for most of the classical compact deep learning models [109, 233, 173]. However, previous works in [80, 49] observed that CNNs with a fully connected layer (FC) or global depthwise convolution are more accurate than the ones with global average pooling for FR. A fully connected layer has been used in many of the recent deep FR models to obtain face representations [80]. However, using FC on top of the last convolutional layer will add a large number of parameters to the model. And thus, it extremely increases the memory footprint and reduces the throughput. For example, giving the last convolutional layer of CNN with a kernel size of 7×7 (as in MixNet) and output feature maps of size 200, the output of this layer, in this case, has a size of $7 \times 7 \times 200$. Using FC of size $512 - d$ on top of the previous layer will add additional 5M parameters to the network ($7 \times 7 \times 200 \times 512$). Even for small FC, $128 - d$, the number of additional parameters caused by FC will be 1.2M. Thus, using FC is not the optimal choice for an efficient FR model. Using global depthwise convolution is a common choice for most of the previous works proposing efficient FR models as it contains fewer parameters than FC, and it can lead to higher verification performance than using global average pooling [49]. Therefore, the global average pooling is replaced with global depthwise convolution. Specifically, we first add 1×1 convolutional layer (Conv1) with stride=1 and zero paddings followed by batch normalization [121] and PReLU none-linearity [106]. In Conv1, the channel is expanded from 200 to 1024. Then, 7×7 convolution layer (stride=1, padding=0 and grouping=1024) followed by batch normalization is used. Finally, 1×1 convolution with 512 output channels followed by batch normalization is added to obtain the final feature embedding which is of size $512 - d$, as shown in Figure 3.1.e.

3.3.2. Experimental setup

Dataset: The MS1MV2 dataset [80] is used to train MixFaceNet models. The MS1MV2 is a refined version of the MS-Celeb-1M [103] by [80] and it contains 5.8M images of 85K identities. The Multi-task Cascaded Convolutional Networks (MTCNN) solution [289] is

Method	FLOPs (M)	# Params. (M)	LFW (%)	AgeDB-30 (%)
ArcFace (LResNet100E-IR) [80]	24211	65.2	99.83	98.15
AirFace [150]	1000	-	99.27	-
ShuffleFaceNet 2× [179]	1050	4.5	99.62	97.28
ShuffleFaceNet 1.5× [179]	577.5	2.6	99.67	97.32
VarGFaceNet [284]	1022	5	99.68	98.10
MobileFaceNet [180]	933	2.0	99.7	97.6
MobileFaceNetV1 [180]	1100	3.4	99.4	96.4
ProxylessFaceNAS [180]	900	3.2	99.2	94.4
MixFaceNet-M (ours)	626.1	3.95	99.68	97.05
ShuffleMixFaceNet-M (ours)	626.1	3.95	99.60	96.98
MobileFaceNets [49]	439.8	0.99	99.55	96.07
ShuffleFaceNet 0.5× [179]	66.9	0.5	99.23	93.22
MixFaceNet-S (ours)	451.7	3.07	99.60	96.63
ShuffleMixFaceNet-S (ours)	451.7	3.07	99.58	97.05
MixFaceNet-XS (ours)	161.9	1.04	99.60	95.85
ShuffleMixFaceNet-XS (ours)	161.9	1.04	99.53	95.62

Table 3.1.: MixFaceNets verification accuracies on LFW and AgeDB-30 datasets. The first row of the table shows the achieved result by the current SOTA ResNet100 models. The table is divided into two parts. The first part of the table shows the results achieved by models with computational complexity between 500 and 1000M FLOPs. The second part of the table shows the achieved by models that have computational complexity less than 500M FLOPs. The number of decimal points is reported as in the related works.

used to detect and align face images. The MixFaceNet models process an aligned and cropped face image of the size $112 \times 112 \times 3$ to produce $512 - d$ feature embeddings. MixFaceNets are evaluated on the widely used LFW [114] and on the AgeDB-30 [192] datasets. Also,the performance of the MixFaceNets is reported on large scale evaluation datasets including MegaFace [136], IJB-B [274] and IJB-C [183].

MixFaceNets training setup: The proposed models in this work are implemented using Pytorch. All models are trained using ArcFace loss [80]. The margin value of ArcFace loss is set to 0.5 and the feature scale to 64. The batch size is set to 512 and trained MixFaceNets using distributed Partial-FC algorithm [11] on one machine with 4 Nvidia GeForce RTX 6000 GPUs to enable faster training on a single node. All models are trained with Stochastic Gradient Descent (SGD) optimizer with an initial learning rate of 1e-1.

The momentum is set to 0.9 and the weight decay to 5e-4. The learning rate is divided by 10 at 80k, 140k, 210k, and 280k training iterations. During the training, we evaluate the model on LFW and AgeDB after each 5650 training iterations. The training is stopped after 300k iterations. During the testing phase, the feature embedding is obtained from the last layer of the size $512 - d$. The Euclidean distance between feature vectors is used in all experiments for comparison.

Method	MFLOPs	Params (M)	MegaFace		MegaFace (R)		IJB	
			Rank-1 (%)	TAR at FAR1e–6	Rank-1 (%)	TAR at FAR1e–6	IJB-B	IJB-C
ArcFace (LResNet100E-IR) [80]	24211	65.2	81.03	96.98	98.35	98.48	94.2	95.6
AirFace [150]	1000	-	80.80	96.52	98.04	97.93	-	-
MobileFaceNet [180]	933	2.0	79.3	95.2	95.8	96.8	92.8	94.7
ShuffleFaceNet [180, 179]	577.5	2.6	77.4	93.0	94.1	94.6	92.3	94.3
MobileFaceNetV1 [180]	1100	3.4	76.0	91.3	91.7	93.0	92.0	93.9
VarGFaceNet [180, 284]	1022	5.0	78.20	93.9	94.9	95.6	92.9	94.7
ProxylessFaceNAS [180]	900	3.2	69.7	82.8	82.1	84.8	87.1	89.7
MixFaceNet-M (ours)	626.1	3.95	78.2	94.26	94.95	95.83	91.55	93.42
ShuffleMixFaceNet-M (ours)	626.1	3.95	78.13	94.24	94.64	95.22	91.47	93.5
MobileFaceNets [49]	439.8	0.99	-	90.16	-	92.59	-	-
MixFaceNet-S (ours)	451.7	3.07	76.49	92.23	92.67	93.79	90.17	92.30
ShuffleMixFaceNet-S (ours)	451.7	3.07	77.41	93.60	94.07	95.19	90.94	93.08
MixFaceNet-XS	161.9	1.04	74.18	89.40	89.35	91.04	88.48	90.73
ShuffleMixFaceNet-XS (ours)	161.9	1.04	73.85	89.24	88.823	91.03	87.86	90.43

Table 3.2.: The achieved results on large-scale evaluation datasets- MegaFace, IJB-B, and IJB-C. The results on MegaFace and MegaFace (R) [80] using FaceScrube as probe set are reported as face identification (Rank-1 %) and verification (TAR at FAR1e–6) for different lightweight models. The last two columns of the table show 1:1 verification TAR (at FAR=1e-4) on IJB-B and IJB-C. The first row reports the evaluation result using the SOTA FR model- ArcFace (LResNet100E-IR), which contains 65.2M parameters and 24211M FLOPs. The rest of the table is organized into two parts: models with computational complexity between 500 and 1000M FLOPs and models with less than 500M FLOPs. The number of decimal points is reported as in the related works. Considering the computation complexity, MixFaceNet models are evaluated as ones of top-ranked models.

3.3.3. Results

This section presents the achieved result by the MixFaceNets on different benchmarks. We acknowledge the evaluation metrics in the ISO/IEC 19795-1 [178] standard. However, for the sake of comparability and reproducibility, we follow the evaluation metrics used in the utilized benchmarks and the previous works reporting on them.

Result on LFW and AgeDB-30

LFW [114] is one of the widely used datasets for unconstrained face verification. The dataset contains 13,233 images of 5749 different identities. The result on LFW is reported as verification accuracy (as defined in [114]) following the unrestricted with labeled outside data protocol using the standard 6000 comparison pairs defined in [114]. AgeDB [192] is common used in-the-wild dataset for evaluating age-invariant face verification. It contains 16,488 images of 568 different identities. The performance as verification accuracy is reported for AgeDB-30 (years gap 30) as it is the most challenging subset of AgeDB. Also, it is the commonly reported set of AgeDB by the recent SOTA FR models. Similar to the LFW, MixFaceNets is evaluated on AgeDB-30 following the standard protocol provided by AgeDB [192]. Table 3.1 shows the achieved result on LFW and AgeDB-30. The result is reported first for one of the top-ranked FR models, ArcFace (LResNet100E-IR) [80], to give an indication of the current SOTA performance on LFW (99.83 %) and AgeDB-30(98.15%). Although, the ArcFace (LResNet100E-IR) model [80] is far from being considered an efficient model, in comparison to lightweight models, with 24211M FLOPs and 65.2m parameters. Then, the second section of Table 3.1 presents the achieved result by the recent lightweight models that have computational complexity between 500 and 1000M FLOPs. The best-reported result on LFW (99.70% accuracy) is achieved by the MobileFaceNet [180] (933M FLOPs). MixFaceNet-M achieved a competitive result on LFW (99.68% accuracy) using 38% fewer FLOPs (626M). A similar result has been achieved on AgeDB-30. MixFaceNets achieved very close accuracy to the current SOTA models using a more efficient model architecture with almost the same number of parameters. Among all models that have computational complexity less than 500M FLOPs, MixFaseNet models outperform all listed models, including MobileFaceNets [49] on LFW and AgeDB-30. Similar conclusion can be seen in the Figure 3.2a and 3.2b. It can be clearly noticed that MixFaceNets achieved the highest accuracies on LFW and AgeDB-30 when considering the same level of computational complexity.

Result on IJB-B and IJB-C

The IJB-B face dataset consists of 1,845 subjects of 21,798 still images and 55,026 frames from 7,011 videos [274]. The IJB-B verification protocol provides a list of 10,270 genuine comparisons and 8M impostor comparisons. The IJB-C face dataset is an extension of IJB-B by increasing the database variability and size with additional 1,661 new subjects [183]. The IJB-C consists of 31,334 still images and 117,542 frames from 11,779 videos of 3531 subjects. The IJB-C verification protocol provides a list of 19,557 genuine comparisons and 15,638,932 impostor comparisons. The result on IJB-C and IJB-B is reported in terms of TAR at FAR (as defined in [274]) equal to 1e-4 to provide a comparable result with the previous works evaluated on these datasets. The achieved verification performances on IJB-B and IJB-C by MixFaceNet models are reported as part of the Table 3.2. The proposed MixFaceNet-M and ShuffleMixFaceNet-M models achieved close results to the top-ranked models using significantly fewer FLOPs.

Result on MegaFace

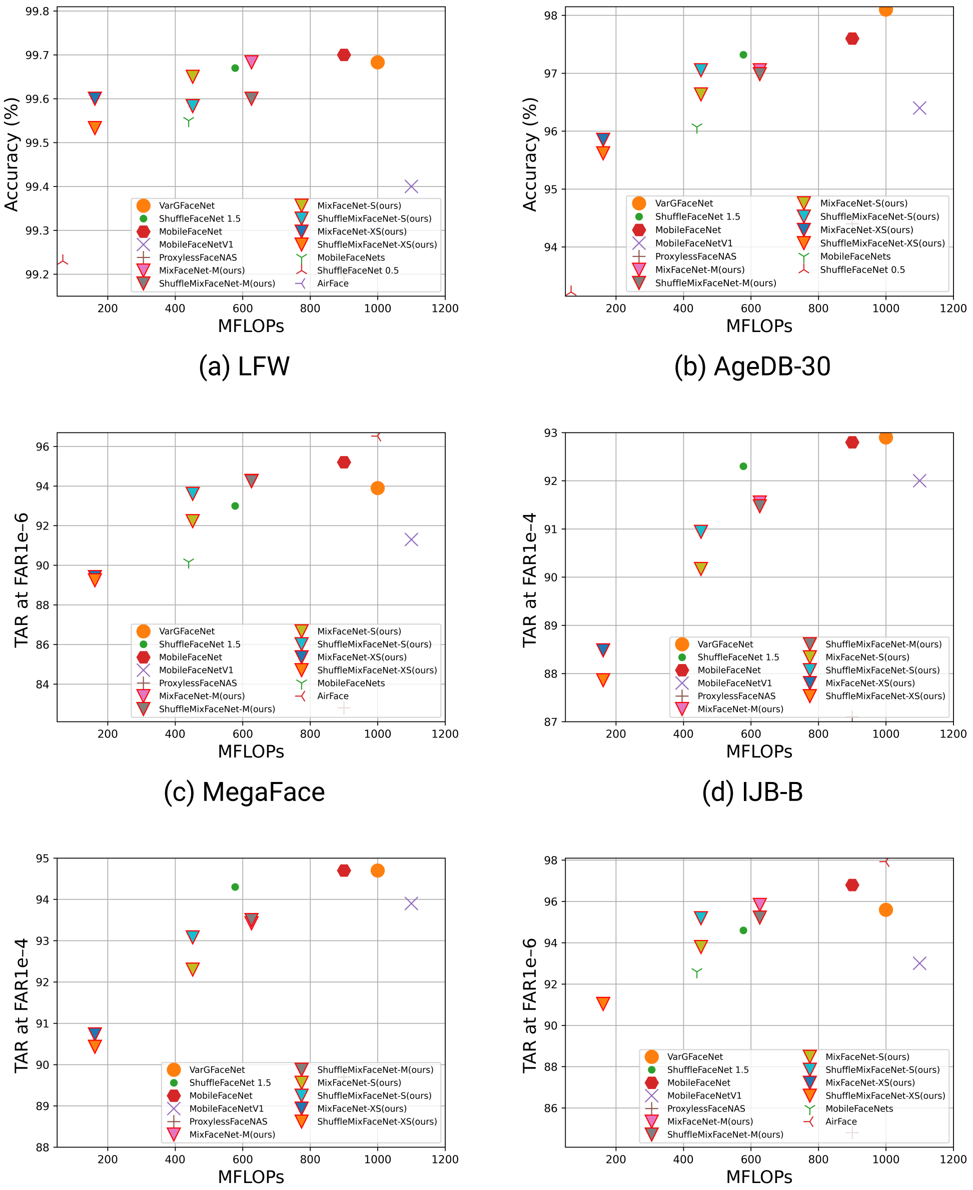
The evaluation protocol of MegaFace includes gallery (1m images from Flickr) and probe (FaceScrub and FGNe) sets. In this work, MegaFace [136] is used as a gallery set, and FaceScrub [199] as the probe set to provide a comparable result with the previous works evaluated on this dataset. The MegaFace [136] contains 1m images of 690K different identities and the FaceScrub contains 100K images of 530 identities [199]. The result on MegaFace is reported as identification (Rank-1) and verification (TAR at FAR=1e–6) to be compatible with the previous works evaluated on this dataset [180]. Also, the result on the refined version of the MegaFace is reported [80]. The face verification and identification results on the MegaFace and the refined version of MegaFace (noted as MegaFace (R)) are presented in Table 3.2. For all evaluated models that have computational complexity between 500 and 1000M FLOPs, the proposed MixFaceNet-M outperformed ProxylessFace-NAS [180], VarGFaceNet [180], MobileFaceNetV1 [180], and ShuffleFaceNet [180, 179]. And it achieved very close verification, and identification results to the top-ranked models-AirFace [150], and MobileFaceNet [180] using less than half the number of FLOPs. Also, when the considered computational cost is less than 500M FLOPs, in the third section of Table 3.2, ShuffleMixFaceNet-S achieved the highest verification and identification performances.

Performance vs. Computational complexity

To present the achieved results in terms of the trade-off between the verification performance and the computation complexity (represented by the number of FLOPs), the number of FLOPs vs. the verification performance of the proposed MixFaceNets and the SOTA solutions are plotted. The plots for the comparisons on the LFW, AgeDB-30, MegaFace, IJB-B, IJB-C and MegaFace(R) benchmarks are presented in Figures 3.2 (a), (b), (c) ,(d), (e) and (f). Each of the reported models is presented by an indicator on the plot, where an ideal model will tend to be placed on the top left corner (high performance and low complexity). In most ranges of the number of FLOPs and on the six benchmarks, different versions of MixFaceNets achieved the highest verification performance. Similar conclusions can be made by analyzing the presented values in Table 3.2.

3.3.4. Discussion

This section presented accurate and extremely efficient FR models, MixFaceNets. Extensive experiments on popular, publicly available datasets, including LFW, AgeDB-30, MegaFace, IJB-B, and IJB-C, have been conducted in this work. The overall evaluation results demonstrate the effectiveness of the proposed MixFaceNets for applications associated with low computational complexity requirements. MixFaceNet-S and ShuffleMixFaceNet-S outperformed MobileFaceNets [49] under the same level of computation complexity ($\leq$500M FLOPs). Also, MixFaceNet-M is shown to be one of the top-ranked performing models while using significantly fewer FLOPs than the SOTA models.



(a) LFW

(b) AgeDB-30

(c) MegaFace

(d) IJB-B

(e) IJB-C

(f) MegaFace (R)

Figure 3.2.: FLOPs vs. performance on LFW (accuracy), AgeDB-30 (accuracy), MegaFace (TAR at FAR1e-6), IJB-B (TAR at FAR1e-4), IJB-C (TAR at FAR1e-4) and refined version of MegaFace, noted as MegaFace (R), (TAR at FAR1e-6). The proposed MixFaceNets are highlighted with triangle marker and red edge color.

3.4. Compact face recognition through AutoML

This work successfully aims at intelligently designing and training a family of lightweight FR models, namely the PocketNets, that offer the SOTA trade-off between model compactness and performance [38]. To achieve that, we focus on two aspects, the first is the use of a NAS algorithm to learn an FR-specific lightweight network architecture, and the second is to design a novel KD paradigm to relax training difficulties raised by the substantial discrepancy between teacher and student models. We use CASIA-WebFace (500K images) [286] to learn the optimal architecture using Differential Architecture Search (DARTS) [155]. We additionally propose a novel training paradigm based on KD, namely multi-step KD, to enable transferring the knowledge of the teacher network at different stages of the training process, and thus enhance the verification performance of the compact student model. We prove our face-specific NAS-based architecture and the proposed multi-step KD in two detailed ablation studies. First, we experimentally evaluate the impact of the NAS training dataset source (face vs. general image classes) on the FR performance of the learned architecture. Second, we experimentally proved and analyzed the competence of our proposed multi-step KD on improving FR performance in comparison to the baseline KD solutions, as well as training without KD. To experimentally demonstrate the competence of our proposed PocketNets, we report their FR performance on nine different benchmarks, in comparison to the recent SOTA compact models, in terms of FR performance and model compactness. In a detailed comparison, different versions of our PocketNets scored SOTA performances in both, under 1M parameters and under 2M parameters, FR model categories. Moreover, PocketNets achieved very competitive results to much larger FR models, and even outperformed them in many cases.

3.4.1. Methodology

This section presents the methodology leading to our proposed PocketNets solution, both the architecture design and the training paradigm. We first present the NAS process leading to the architecture of our proposed PocketNets. Then, we present our proposed multi-step knowledge distillation training paradigm.

Towards PocketNet architecture

Neural architecture search (NAS) automates the network design by learning the network architecture that achieves the best performance for a specific task. NAS has proved to be a robust method in discovering and optimizing neural network architecture. Previous works [41, 155] demonstrate that the discovered network architectures by NAS do outperform handcraft-designed network architectures for different computer vision tasks. For our PocketNets, we opt to use DARTS algorithm [155] to search for two types of building blocks (cell) i.e. normal cell and reduce cell, which can be stacked to form the final architecture. Our choice for DARTS is based on: a) it achieved a competitive result to the SOTA NAS solutions on different image classification tasks [155], and b) the search time for DARTS is feasible in comparison to other search methods [298, 299] and thus, it can be adapted to a large-scale dataset. Unlike common NAS algorithms that are applied on a small image size of a small dataset, our NAS will be learned on a large-scale face image dataset with relatively high resolution. In the following, we briefly present the DARTS algorithm. Our goal here is not only to build an optimal architecture, but also to analyze the FR performance implications when optimizing such an architecture on a different learning task, as will be clarified later in this work.

DARTS aims at learning two types of cells: normal cell and reduce cell. Each cell is a direct acyclic graph (DAG) that consist of N nodes. Each node x_i is a latent representation, where $i \in [0, N]$. The operation space O is a set of candidate operation e.g. convolutional layer, skip-connection, pooling layer etc. Each edge (i, j) between node x_i and x_j is a candidate operation $o^{(i,j)} \in O$ that applies a particular transformation on x_i. Each candidate operation o is weighted by the architecture parameter $a(i, j)$. An intermediate node x_j is calculated as $x_j = \sum_{i<j,i\in[0,N]} o^{(i,j)}(x_i)$. Each cell (DAG) has two input nodes and a single output node. The two input nodes are the output of the previous two cells of the network. The output of the last node x_{N-1} i.e. the cell output, is a concatenation of all nodes in the DAG excluding the input nodes. The candidate operation applied to $x^{(i)}$ is represented as a function $o(.)$. The choice of a candidate operation is formulated by applying a Softmax function over the weights of all possible operations O:

$$\bar{o}^{(i,j)}(x) = \sum_{o\in O} \frac{\exp(\alpha_o^{(i,j)})}{\sum_{o'\in O} \exp(\alpha_{o'(i,j)})} o(x), \tag{3.1}$$

where $\alpha_o^{(i,j)}$ is a network architecture weight parameter of a candidate operation o. Therefore, the architecture search becomes a task of learning a set of parameters $\alpha = \{\alpha^{(i,j)}\}$. The learning procedure of DARTS is based on jointly learning the network architecture represented by α and the network weights w. Given L_{train} and L_{val} as the train and validation loss, respectively. The learning objective of DARTS is to find the optimal architecture represented by α^* that minimizes the validation loss $L_{val}(w^*, \alpha^*)$ with $w^* = \arg\min_w L_{train}(w, \alpha^*)$ as the best performing network weights on the training set. The architecture parameters are learned using a bi-level optimization problem with α as the upper-level and w the lower level variable:

$$\begin{aligned} \min_{\alpha} L_{val}(w^*(\alpha), \alpha) \\ s.t. w^*(\alpha) = \arg\min_{w} L_{train}(w, \alpha). \end{aligned} \tag{3.2}$$

The final discrete architecture is derived by setting $o^{(i,j)} = argmax_{o \in O} \alpha_o^{(i,j)}$. Given an input of the shape $w \times h \times c$, the output of the reduction cell is $w/2 \times h/2 \times 2c$ and the output of the normal cell is $w \times h \times c$. The first two nodes of cell k represent the output of the two previous cells $k-1$ and $k-2$.

Search space: PocketNet search space includes the following operations: 1) 3×3, 5×5, 7×7 depthwise separable convolutions [109] with kernel size of $\{3 \times 3, 5 \times 5, 7 \times 7\}$, padding of $\{1, 2, 3\}$ to preserve the spatial resolution, and they have a stride of one (if applicable). 2) 1×1 Conv, a convolution layer with kernel size of 1×1 and zero padding. 3) max pooling layer with kernel size of 3×3. 4) average pooling layer with a kernel size of 3×3. 5) identity. 6) zero. A zero operation indicates that there is no connection between nodes. The max and average pooling layers are followed by batch noramlization (BN) [121]. We use Parametric Rectified Linear Unit (PReLU) [106] as the non-linearity in all convolutional operation.

PocketNet architecture: We followed [155] by setting the number of nodes in all cells to $N = 7$. We apply fast down-sampling in the beginning of the network using 3×3 convolution (stride=2) followed by BN [121]. To obtain the feature embedding of the input face image, we use global depthwise convolution [109] rather than using average pooling or fully connected layer directly before the classification layer. Our choice of using the global depthwise convolution for the embedding stage is based on: a) it contains fewer parameters than a fully connected layer, b) convolutional neural network (CNN) with global depth-wise convolution is more accurate than the one with average pooling for FR, as reported in previous works [49, 23]. The rest of the network architecture is constructed by stacking M normal cells and 3 reduction cells at 1/3 and 2/3 of the network depth, and after the last normal cell. We trained the NAS to optimize α_{normal} and $\alpha_{reduction}$ used to construct the normal and reduction cells, respectively.

We trained the search algorithm to learn from the CASIA-WebFace dataset [286]. Training details are presented later in Section 3.4.2. The best discovered normal and reduction cells by DARTS are shown in Figures 3.3a and 3.3b, respectively. In this work, we present four architectures based on the learned cells: PocketNetS-128, PocketNetS-256, PocketNetM-128, and PocketNetM-256. The architecture of PocketNetS-128 and

Operation	Output size	R	Param.
Conv2d(k=3,s=2,p=1),BN	[64 x 56 x 56]	1	1,856
Normal-Cell 1-6	[64 x 56 x 56]	6	33,792
Reduction-Cell 1	[128 x 28 x 28]	1	10,688
Normal-Cell 7-11	[128 x 28 x 28]	5	92,608
Reduction-Cell 2	[256 x 14 x 14]	1	35,712
Normal-Cell 12-15	[256 x 14 x 14]	4	60,493
Reduction-Cell 3	[512 x 7 x 7]	1	128,768
PReLU, Conv2d(k=1), BN, PReLU	[512 x 7 x 7]	1	264,192
Conv2d(k=7,g=512), BN	[512 x 1 x 1]	1	26,112
Conv2d(k=1), BN	[128 x 1 x 1]	1	65,792

Table 3.3.: Architecture of PocketNetS-128. Normal and reduction cells are the cells learned by DARTS on CASIA-WebFace. The table shows the number of parameters for each operation. If the operation contains a set of sub-operations (e.g. Conv2d, BN), the number of parameters is presented as the sum of parameters for all these sub-operations and multiplied by R. Column R indicates how many times the operation is repeated. The k of the convolution layer (Conv2d) refers to the kernel size, s is the stride, p is the padding, and g is the group parameter.

PocketNetS-256 (PocketNet small) are identical. Each of them contains 18 cells i.e 15 normal cells and 3 reduction cells. The number of feature maps (out channel) of the first layer is 64. The only difference is the embedding size, where the embedding in PocketNetS-128 is of size 128-D and in PocketNetS-256 is of size 256-D. Table 3.3 presents the overall architecture of PocketNetS-128. PocketNetS-128 contains in total 925,632 trainable parameters and setting the embedding size to 256 increases the number of parameters in PocketNetS-256 to 991,424. All networks use floating-point 32 and the required memory footprints are 3.7 and 3.9 MB by PocketNetS-128 and PocketNetS-256, respectively. The main motivation for using different embedding sizes is to evaluate the effect of embedding size on the network performance and memory footprint. We also investigate a wider architecture of PocketNet by doubling the number of feature maps of the network and reducing the number of cells from 18 to 9. This result in two networks: PocketNetM-128 and PocketNetM-256 (PocketNet medium) with embedding size of 128-D and 256-D, respectively. The architecture of PocketNetM-128 is presented in Table 3.4. PocketNetM-128 contains 1,686,656 parameters and PocketNetM-256 contains 1,752,448 parameters.

Operation	Output size	R	Param
Conv2d(k=3,s=2,p=1),BN	[128 x 56 x 56]	1	3712
Normal-Cell1-6	[128 x 56 x 56]	3	56,832
Reduction-Cell 1	[256 x 28 x 28]	1	35,712
Normal-Cell 7-11	[256 x 28 x 28]	2	128,896
Reduction-Cell 2	[512 x 14 x 14]	1	128,768
Normal-Cell 12-15	[512 x 14 x 14]	1	227,072
Reduction-Cell 3	[1024 x 7 x 7]	1	486,912
PReLU, Conv2d(k=1), BN, PReLU	[512 x 7 x 7]	1	526,848
Conv2d(k=7,g=512), BN	[512 x 1 x 1]	1	26,112
Conv2d(k=1), BN	[128 x 1 x 1]	1	65,792

Table 3.4.: Architecture of PocketNetM-128. Normal and reduction cells are the cells learned by DARTS on CASIA-WebFace. The table shows the number of parameters for each operation. If the operation contains a set of sub-operations (e.g. Conv2d, BN), the number of parameters is presented as the sum of parameters for all these sub-operations and multiplied by R. Column R indicates how many times the operation is repeated. The k of the convolution layer (Conv2d) refers to the kernel size, s is the stride, p is the padding, and g is the group parameter.

PocketNet training paradigm

Towards the PocketNet training paradigm that incorporates our proposed multi-Step KD, we start by formulating the margin-based Softmax loss and knowledge distillation concept. Margin-Based Softmax loss has been widely deployed in recent FR solutions [80, 268, 185]. It achieved SOTA accuracy on major benchmarks [80, 180, 185]. In this work, we utilize the ArcFace loss [80] to train our PocketNets. ArcFace loss extends over the softmax loss by manipulating the decision boundary between the classes by deploying an additive angular margin penalty on the angle between the weights of the last fully connected layer and the feature representation. Formally, ArcFace loss is defined as follow:

$$L_{Arc} = \frac{1}{M}\sum_{i\in M} -log \frac{e^{s(cos(\theta_{y_i}+m))}}{e^{s(cos(\theta_{y_i}+m))} + \sum\limits_{j=1,j\neq y_i}^{C} e^{s(cos(\theta_j))}}, \tag{3.3}$$

where θ_{yi} is the angle between the feature f_i and $i-th$ class center, $y_i \in [1, C]$ (C is the number of classes), M is batch size, m is the margin penalty value and s is scale parameter.

Knowledge distillation (KD): KD is a technique to improve the performance and generalizability of smaller models by transferring the knowledge learned by a cumbersome model (teacher) to a single small model (student) [108]. The idea is to guide the student model to learn the relationship between different classes discovered by the teacher model that contains more complex information beyond the ground truth labels [108]. The KD is originally proposed to improve the performance of a small backbone trained with SoftMax loss for a classification task [108]. However, the learning objective of the FR model is to optimize feature representations needed for face verification. In this work, as a step towards our proposed multi-step KD, we train our PocketNet model to learn feature representations that are similar to the ones learned by the teacher model. We achieve that by introducing an additional loss function (Mean squared error (MSE)) to ArcFace loss operated on the embedding layer. Formally,the l_{mse} loss is defined as follows:

$$l_{mse} = \frac{1}{M} \sum_{i \in M} 1 - \frac{1}{D} \Sigma_{h=1}^{D} \left(\Phi_t^S(x)_h - \Phi_t^T(x)_h \right)^2, \tag{3.4}$$

where Φ_t^S and Φ_t^T are the feature representations obtained from the last fully connected layer of student and teacher models, respectively, and D is the size of the feature representation. The final training loss function is defined as follow:

$$l_{mse} = l_{Arc} + \lambda l_{mse}, \tag{3.5}$$

where λ is a weight parameter. The feature representations learned by the ArcFace loss are normalized. Thus, the value range of l_{mse} is much small i.e. $\leq$ 0.007. This value is very small in comparison to the ArcFace loss value (around 60 at the beginning of the training phase.) We set the λ value to 100. Thus, the l_{mse} contributes to the model training.

Multi-Step knowledge distillation: Previous works [187, 284] observed that transforming the knowledge from a very deep teacher model to a small student model is difficult when the gap in terms of network size between the teacher and the student model is large.

In this work, we present a novel concept by relaxing this difficulty of a substantial discrepancy between teacher model and student by synchronizing the student and the teacher model during the training, without the need for transforming the knowledge to intermediate networks [187, 284]. Our solution is designed to transfer the knowledge learned by a teacher model in a step-wise manner after each x number of iterations, i.e.

multi-step KD. The key idea is that the information learned by a teacher at different steps of the training phase is different from the one learned when the teacher is fully converged. Thus, transferring the knowledge learned by a teacher at an early stage of training is easier for a student to learn. Thus, at a later point when the student is converged to some degree, it can learn more complex patterns from the teacher. To achieve that, we first train the teacher for I iterations. This teacher model is noted as $T1$. Then, we train the student model for the same number of iteration I with the assistance of the teacher $T1$. In this case, Φ^T (Equation 3.4) is $T1$ obtained after the first I iterations. We choose to train the teacher for one epoch each time. This will give the teacher a chance to learn from the whole training dataset. We repeated these two steps until the teacher and student models are converged. To simplify the implementation, we train first the teacher model until it is converged and save the model weights after each epoch. Then, we train the student model with the assistance of the teacher models. During the student training, we load the teacher weights that correspond to the same training epoch.

3.4.2. Experimental setups

Neural architecture search

We train the DARTS to learn the normal and reduction cells on the CASIA-Webface dataset [286]. CASIA-Webface consists of 494,141 face images from 10,757 different identities. We split the dataset equally into two parts used for training and validation. The images are pre-aligned and cropped to 120×120 for the training subset and to 112×112 for the validation subset using the Multi-task Cascaded Convolutional Networks (MTCNN) solution [289]. During the training phase, the training images are randomly cropped to have a fixed size of 112×112 and then randomly horizontally flipped to make the search more robust, following common practice in FR research [80, 185]. All the training and validation images are normalized to have pixel values between -1 and 1. We followed DARTS training setup [155] by using Stochastic Gradient Descent with the momentum of 0.9 and weight decay of $3e-4$ to optimize the DARTS weight w. We utilize a cosine annealing strategy [165] to decrease the learning rate after each epoch with a minimum learning rate of 0.004. We set the batch size to 128 and the initial learning rate to 0.1. For α optimization, we use similar setup to DARTS [155] by using Adam optimizer with momentum $\beta = (0.5, 0.999)$ and weight decay of $1e-3$. We set the initial learning rate for Adam optimizer to 0.0012. The initial channel size is set to 64 and the number of nodes in each cell is set to 8. We use a batch size of 128 and train DARTS for 50 epochs. These configurations are chosen to enable DARTS training on available GPUs. All training codes are implemented in Pytorch [210] and trained on 6 NVIDIA GeForce RTX

2080 Ti (11GB) GPUs. The training lasted 2274 hours. We additionally conducted an additional experiment on CIFAR-10 [143] as a NAS domain ablation study for this work. The CIFAR-10 is a commonly used dataset for object detection and image classification tasks consisting of 60000 images (of the size 32×32) of 10 classes. We split CIFAR-10 equally into two parts: training and validation subsets. We run the DARTS search using the exact configurations described previously in this section to learn on the CIFAR-10 dataset. The training lasted around 30 hours on 6 NVIDIA GeForce RTX 2080 Ti (11GB) GPUs.

Face Recognition models and training

Based on the normal and reduction cells learned by DARTS on CASIA-WebFace [286], we trained three instances of PocketNetS-128. The first instance (noted as PocketNetS-128 (no KD)) is only trained with ArcFace loss described in Section 3.4.1. The second instance (noted as PocketNetS-128 (KD)) is trained with ArcFace loss with KD. The third instance is trained with ArcFace loss along with our proposed multi-step KD (noted as PocketNetS-128 (multi-step KD)). These three instances are used in our ablation study towards the proposed multi-step KD. On the other hand, based on the normal and reduction cells learned on CIFAR-10 [143] (object classification domain), we train another model based on these cells, noted as DartFaceNet-128 (no KD). This training is used as an ablation study to analyze the effect of training dataset sources on the NAS algorithm by comparing its FR performance to its direct counterpart PocketNetS-128 (no KD).

Additionally, as detailed earlier, we trained four instances of PocketNets: PocketNetS-128, PocketNetS-256, PocketNetM-128, and PocketNetM-256 to compare our proposed PocketNets with the recent compact FR models proposed in the literature on different levels of compactness. All these models are trained with ArcFace loss along with our proposed multi-step KD. To enable KD multi-step solutions, we trained two instances of the ResNet-100 model with embedding sizes of $128-D$ and $256-D$. The ResNet-100(128) is used as a teacher for PocketNetS-128 and PocketNetM-128, while ResNet-100(256) is used as a teacher for PocketNetS-256 and PocketNetM-256.

We use the MS1MV2 dataset [80] to train all the investigated FR models in this work. The MS1MV2 is a refined version [80] of the MS-Celeb-1M [103] containing 5.8M images of 85K identities. We follow the common setting [80] to set the scale parameter s to 64 and margin value of ArcFace loss to 0.5. We set the mini-batch size to 512 and train our models on a single Linux machine (Ubuntu 20.04.2 LTS) with Intel(R) Xeon(R) Gold 5218 CPU 2.30GHz, 512 G RAM, and 4 Nvidia GeForce RTX 6000 GPUs. The proposed models in this work are implemented using Pytorch [210]. All FR models are trained with Stochastic Gradient Descent (SGD) optimizer with an initial learning rate of 1e-1. We set the momentum to 0.9 and the weight decay to 5e-4. The learning rate is divided by 10 at 80k, 140k, 210k, and 280k training iterations. The total number of training iteration is 295K. During the training, we use random horizontal flipping with a probability of 0.5 for data augmentation. The networks are trained (and evaluated) on images of the size $112 \times 112 \times 3$, with pixel values between -1 and 1. These images are aligned and cropped using the Multi-task Cascaded Convolutional Networks (MTCNN) [289], following [80].

Evaluation benchmarks and metrics

We evaluate our PocketNets and build a comparison to SOTA based on 9 benchmarks detailed in this section. The considered evaluation benchmarks are LFW [114], CA-LFW [296], CP-LFW [295], CFP-FP [236], AgeDB-30 [192], IJB-B [274], IJB-C [183], MegaFace [136], and MegaFace (R) [80, 136].

We acknowledge the evaluation metrics in the ISO/IEC 19795-1 [178] standard, however, for comparability, we follow the evaluation metrics defined in the utilized benchmarks as follows: LFW (accuracy), CA-LFW (accuracy), CP-LFW (accuracy), CFP-FP (accuracy), AgeDB-30 (accuracy), MegaFace (Rank-1 identification rate and TAR at FAR) of 1e-6), IJB-B (TAR at FAR1e-4), IJB-C (TAR at FAR1e-4) and MegaFace (R), (Rank-1 identification rate and TAR at FAR1e-6). A detailed description of the benchmarks is provided in the supplementary material.

3.4.3. Ablation study

This section presents two ablation studies addressing the two main aspects of our design of the PocketNets solution.

Ablation study on NAS training dataset source: We trained two different instances of DARTS search algorithm to learn from CASIA-WebFace [286] (face images) and CIFAR-10 [143] (animals, cars, etc.), respectively. Figure 3.3 presents the normal and reduction cells learned on CASIA-WebFace and CIFAR-10, used to build our PocketNetS-128 (no KD) and the DartFaceNetS-129 (no KD), respectively. These networks share the same structure including the embedding stage and the number of cells. These networks are trained using the exact training setup described in Section 3.4.2. DartFaceNetS-128 (no KD) contains 885,184 parameters with 620.9286 MFLOPs. PocketNetS-128 (no KD) contains 925,632 parameters with 587.11 MFLOPs. Table 3.5 presents the achieved performance by PocketNetS-128 (no KD) and DartFaceNetS-128 (no KD) on nine different benchmarks. It can be clearly noticed that PocketNetS-128 (no KD) outperformed DartFaceNetS-128 (no KD) with an obvious margin on all considered benchmarks. The demonstrates that utilizing neural network architecture designed for common computer vision tasks leads to sub-optimal performance when it is used for the FR. It also supports our choice for training NAS to learn from a face image dataset and points out that FR does require face-specific architecture design.

Ablation study on multi-step KD: Here, we prove the benefit of introducing our multi-step KD training process on the PocketNet FR performance. This step-wise ablation study first looks into the advances provided by the KD training in comparison to training with no KD, proving the advancement achieved by our multi-step KD in comparison to KD. Introducing KD to the PocketNet training phase improved the verification performances on all evaluation benchmarks by comparing PocketNetS-128 (no KD) to PocketNetS-128 (KD), ass observed in Table 3.5. PocketNetS-128 (no KD) is trained only with ArcFace loss, while PocketNetS-128 (KD) is trained with ArcFace along with KD from the ResNet-100 model. When PocketNetS-128 is trained with ArcFace along with our multi-step KD (i.e. PocketNetS-128 (multi-step KD)), the achieved verification performance improved in eight out of nine different benchmarks in comparison to PocketNetS-128 (KD) (Table 3.5), empirically proving the benefit of our multi-step KD. We also investigated the competence of our proposed multi-step KD on improving the model convergence. Figure 3.4a presents a comparison between ArcFace loss values of PocketNetS-128 (KD) and PocketNetS-128 (multi-step KD). It can be noticed that multi-step KD improved the model convergence. Also, our multi-step KD enhanced the similarity between the feature representation of the teacher model and the student model. This observation is seen in Figure 3.4b where the MSE values of PocketNetS-128 (multi-step KD) is smaller than the one of PocketNetS-128 (KD).

3.4.4. Experimental results

Table 3.6 presents the achieved FR results by our PocketNets on all evaluation benchmarks. It also presents a comparison between our proposed PocksetNets and the recent compact models proposed in the literature. The presented models are ordered in groups based on the number of parameters (compactness). The first part of Table 3.6 presents the achieved result by the models that have between 2 and 5M trainable parameters, while the second and third parts present the results for the models with less than 2M and less than 1M trainable parameters, respectively.

Our PocketNetS-128 (0.92M parameters) and PocketNetS-256 (0.99M parameters) outperformed all models that have less than 1M parameters. With 10% less parameter than MobileFaceNets [49], PocketNetS-128 outperformed MobileFaceNets on all considered

Model	Param. (M)	MFLOPs	LFW (%)	CA-LFW (%)	CP-LFW (%)	CFP-FP (%)	AgeDB-30 (%)	IJB-B (%)	IJB-C (%)	MegaFace		MegaFace (R)	
										Rank-1(%)	Ver.(%)	Rank-1(%)	Ver.(%)
ResNet100-128 - Teacher	55.52	24192.51	99.83	96.16	93.1	98.64	98.3	94.72	96.08	80.55	97.13	98.36	98.66
DartFaceNetS-128 (no KD)	0.89	620.9	99.26	94.98	88.5	93.18	95.23	87.89	90.5	73.44	87.65	87.99	89.42
PocketNetS-128 (no KD)	0.925	587.11	99.5	95.01	88.93	93.78	95.88	88.29	90.79	74.42	88.99	89.46	90.67
PocketNetS-128 - KD	0.925	587.11	99.55	95.15	89.13	93.82	**96.50**	89.23	91.47	75.22	90.21	90.72	92.04
PocketNetS-128 - multi-step KD	0.925	587.11	**99.58**	**95.48**	**89.63**	**94.21**	96.10	**89.44**	**91.62**	**75.81**	**90.54**	**91.22**	**92.23**

Table 3.5.: Comparative evaluation results of ResNet100-128, DartFaceNetS-128 (no KD), PocketNetS-128 (no KD), PocketNetS-128 KD, and PocketNetS-128 multi-step KD on different evaluation benchmarks. The results are reported based on the evaluation metric described in Section 3.4.2. ResNet100-128, DartFaceNetS-128 (no KD) and PocketNetS-128 (no KD) are trained with ArcFace loss. PocketNetS-128 KD is trained with ArcFace loss with KD from teacher model (ResNet100-128). PocketNetS-128 multi-step KD is trained with ArcFace loss with multi-step KD from teacher model (ResNet100-128). PocketNetS-128 (no KD) performed better than the DartFaceNetS-128 (no KD), proving the sanity of designing FR-specific architecture. PocketNetS-128 multi-step KD performes better than PocketNetS-128 (no KD) and PocketNetS-128 KD, proving the benefits of the proposed multi-step KD.

benchmarks. Also, PocketNetS-128 and PocketNetS-256 achieved competitive results to other deeper models that contain 4 or 5 times more parameters than PocketNets. For example, PocketNetS-128 outperformed VarGFaceNet (5M parameters) on the challenging CA-LFW and CP-LFW benchmarks where the achieved accuracies by PocketNetS-128 are 95.48% on CA-LFW and 89.63% on CP-LFW in comparison to 95.15% on CA-LFW and 88.55% CP-LFW achieved by VarGFaceNet [284].

The proposed PocketNetM-128 (1.68M parameters) and PocketNetM-256 (1.75M parameters) outperformed all models proposed in the literature that have less than 2M parameters. They also achieved competitive results to the models that have between 2 and 5M parameters, even outperforming them in many cases. For example, our PocketNetM-128 achieved SOTA accuracies on the challenging CA-LFW and CP-LFW among all models that have less than 5M of trainable parameters. On the large-scale evaluation benchmarks, IJB-B and IJB-C, our PocketNetM achieved competitive performance to many of the larger models. For example, on IJB-C, our PocketNetM-128 (1.68M parameters) achieved verification performance of 92.63% TAR at FAR 1e-6 and the best verification performance is 94.7% achieved by MobileFaceNet [180] (2M parameters) and VarGFaceNet [284] (5M parameters). On MegaFace and the refined version of MegaFace, our PocketNetM outperfomred all the models than have less than 2M of trainable parameters and they achieved a competitive results in term of identification and verification accuracies to the models that have between 2 and 5M parameters. For example, our PocketNetM-258 (1.75M

Model	Params.(M)	MFLOPs	LFW (%)	CA-LFW (%)	CP-LFW (%)	CFP-FP (%)	AgeDB-30 (%)
VarGFaceNet [284, 180]	5.0	1022	99.85	95.15	88.55	98.50	98.15
ShuffleFaceNet 2× [179]	4.5	1050	99.62	-	-	97.56	97.28
MixFaceNet-M [23]	3.95	626.1	99.68	-	-	-	97.05
ShuffleMixFaceNet-M [23]	3.95	626.1	99.60	-	-	-	96.98
MobileFaceNetV1 [180]	3.4	1100	99.4	94.47	87.17	95.8	96.4
ProxylessFaceNAS [180]	3.2	900	99.2	92.55	84.17	94.7	94.4
MixFaceNet-S [23]	3.07	451.7	99.6	-	-	-	96.63
ShuffleMixFaceNet-S [23]	3.07	451.7	99.58	-	-	-	97.05
ShuffleFaceNet 1.5x [179, 180]	2.6	577.5	99.7	95.05	88.50	96.9	97.3
MobileFaceNet [180]	2.0	933	99.7	95.2	89.22	96.9	97.6
PocketNetM-256 (Ours)	1.75	1099.15	99.58	95.63	90.03	95.66	97.17
PocketNetM-128 (Ours)	1.68	1099.02	99.65	95.67	90.00	95.07	96.78
Distill-DSE-LSE [161]	1.35	-	99.67	95.63	89.68	94.19	96.83
MixFaceNet-XS [23]	1.04	161.9	99.60	-	-	-	95.85
ShuffleMixFaceNet-XS [23]	1.04	161.9	99.53	-	-	-	95.62
MobileFaceNets [49]	0.99	439.8	99.55	-	-	-	96.07
PocketNetS-256 (Ours)	0.99	587.24	99.66	95.50	88.93	93.34	96.35
PocketNetS-128 (Ours)	0.92	587.11	99.58	95.48	89.63	94.21	96.10
ShuffleFaceNet 0.5x [179]	0.5	66.9	99.23	-	-	92.59	93.22

Table 3.6.: The achieved results on LFW, CA-LFW, CP-LFW, CFP-LFW and AgeDB benchmarks. The results are reported in % based on the evaluation metric described in Section 3.4.2. The models are ordered based on the number of parameters. Our PoacketNetS-128 and PocketNetS-256 consistently extend the SOTA performance on all evaluation benchmarks for the models that have less than 1M parameters. Our PoacketNetM-128 and PocketNetM-256 also achieved SOTA performances for models that have less than 2M parameters. Additionally, they achieved very competitive results to larger models that have between 2 and 5M parameters. All decimal points are provided as reported in the respective works.

parameters) outperformed MixFaceNet-S [23] (3.07M parameters), ProxylessFaceNAS [180] (3.2M parameters) and MobileFaceNetV1 [180] (3.4M parameters) on MegaFace and MegaFace (R).

To visually illustrate the competence of our PocketNet, we plot the number of parameters vs. the achieved verification performance of our PocketNet and the recent compact models proposed in the literature (all numbers provided in Table 3.6). Figure 3.5 presents a trade-off between the number of parameters and the achieved verification performance. Each of the presented solutions is marked with a point(x,y) in the plot, where x is the

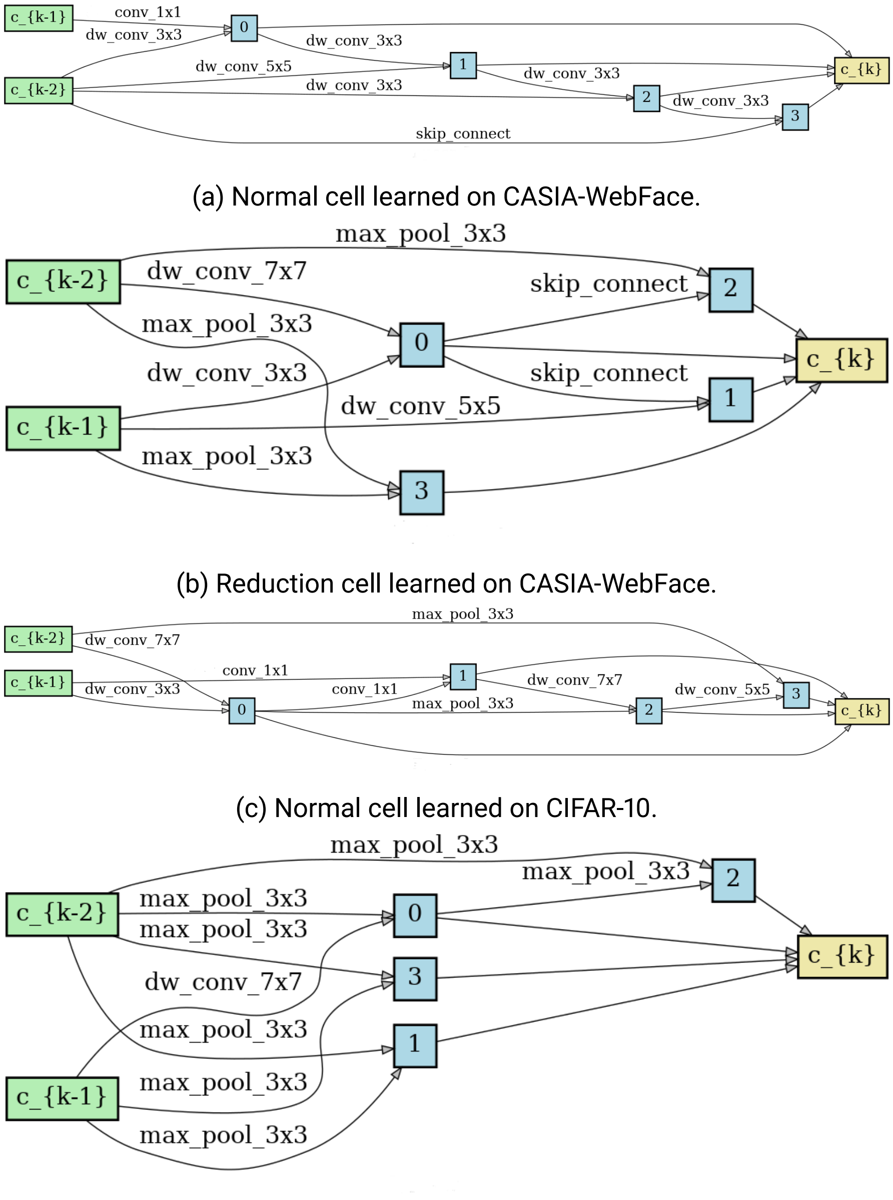
Model	Params.(M)	MFLOPs	IJB-B (%)	IJB-C (%)	MegaFace		MegaFace(R)	
					Rank-1 (%)	Ver. (%)	Rank-1 (%)	Ver. (%)
VarGFaceNet [284, 180]	5.0	1022	92.9	94.7	78.2	93.9	94.9	95.6
ShuffleFaceNet 2× [179]	4.5	1050	-	-	-	-	-	-
MixFaceNet-M [23]	3.95	626.1	91.55	93.42	78.20	94.26	94.95	95.83
ShuffleMixFaceNet-M [23]	3.95	626.1	91.47	91.47	78.13	94.24	94.64	95.22
MobileFaceNetV1 [180]	3.4	1100	92.0	93.9	76.0	91.3	91.7	93.0
ProxylessFaceNAS [180]	3.2	900	87.1	89.7	69.7	82.8	82.1	84.8
MixFaceNet-S [23]	3.07	451.7	90.17	92.30	76.49	92.23	92.67	93.79
ShuffleMixFaceNet-S [23]	3.07	451.7	90.94	93.08	77.41	93.60	94.07	95.19
ShuffleFaceNet 1.5x [179, 180]	2.6	577.5	92.3	94.3	77.4	93.0	94.1	94.6
MobileFaceNet [180]	2.0	933	92.8	94.7	79.3	95.2	95.8	96.8
PocketNetM-256 (Ours)	1.75	1099.15	90.74	92.70	78.23	92.75	94.13	94.40
PocketNetM-128 (Ours)	1.68	1099.02	90.63	92.63	76.49	92.45	92.77	94.17
Distill-DSE-LSE [161]	1.35	-	-	-	-	-	-	
MixFaceNet-XS [23]	1.04	161.9	88.48	90.73	74.18	89.40	89.35	91.04
ShuffleMixFaceNet-XS [23]	1.04	161.9	87.86	90.43	73.85	89.24	88.823	91.03
MobileFaceNets [49]	0.99	439.8	-	-	-	90.16	-	92.59
PocketNetS-256 (Ours)	0.99	587.24	89.31	91.33	76.53	91.77	92.29	93.5
PocketNetS-128 (Ours)	0.92	587.11	89.44	91.62	75.81	90.54	91.22	92.23
ShuffleFaceNet 0.5x [179]	0.5	66.9	-	-	-	-	-	-

Table 3.7.: The achieved results on IJB-B, IJB-C, MegaFace, and MegaFace (R) benchmarks. The results are reported in % based on the evaluation metric described in Section 3.4.2. The models are ordered based on the number of parameters. Our PoacketNetS-128 and PocketNetS-256 consistently extend the SOTA performance on all evaluation benchmarks for the models that have less than 1M parameters. Our PoacketNetM-128 and PocketNetM-256 also achieved SOTA performances for models that have less than 2M parameters. Additionally, they achieved very competitive results to larger models that have between 2 and 5M parameters. All decimal points are provided as reported in the respective works.

number of parameters in millions and y is the achieved verification performance. The model that tends to be placed on the top-left corner (small x and large y) of the plot has the best trade-off between the model compactness and the achieved verification performance. It can be observed, in Figure 3.5, that our PocketNets are always in the top left corner in comparison to other methods, proving to achieve SOTA trade-off between model compactness and FR performance. It must be noted that all the reported PocketNets in this section are trained with our proposed multi-step KD.

3.4.5. Discussion

This section presented a family of extremely lightweight FR models, namely PocketNets [38]. This is one of the first efforts proposing to utilize NAS to learn to design a compact yet accurate FR model. We additionally presented a novel training paradigm based on knowledge distillation, namely mulit-step KD, where the knowledge distillation is performed at multiple stages of the teacher training maturity. Extensive step-wise ablation studies proved the benefits of both, designing a face-specific architecture, as well as, the enhanced performance of the lightweight model when trained with the proposed multi-step KD. Through extensive experimental evaluations on nine FR benchmarks, we demonstrated the high verification performance achieved by our compact PocketNet models and our proposed mulit-step KD. Under the same level of model compactness, our PocketNets consistently scored SOTA performances in comparison to the compact models proposed in the literature.



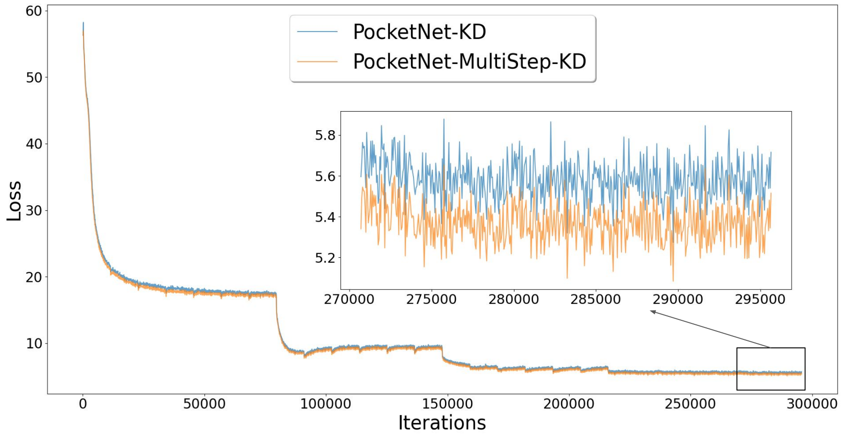
(a) Normal cell learned on CASIA-WebFace.

(b) Reduction cell learned on CASIA-WebFace.

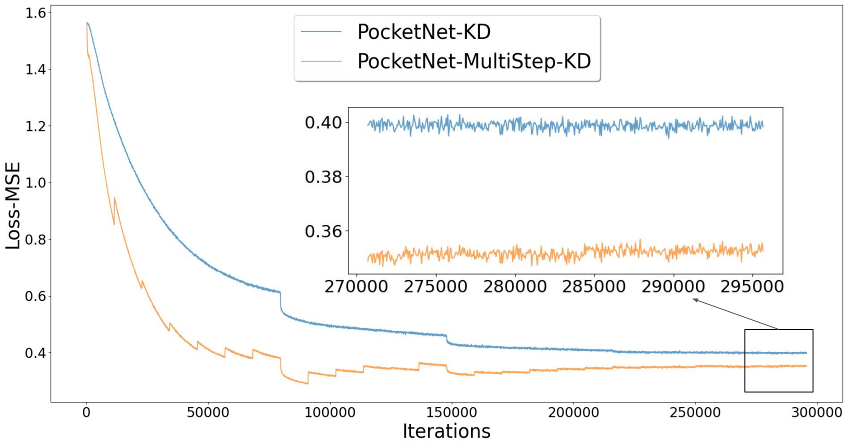
(c) Normal cell learned on CIFAR-10.

(d) Reduction cell learned on CIFAR-10.

Figure 3.3.: Normal and reduction cells learned by DARTS on CASIA-WebFace and CIFAR-10 datasets.

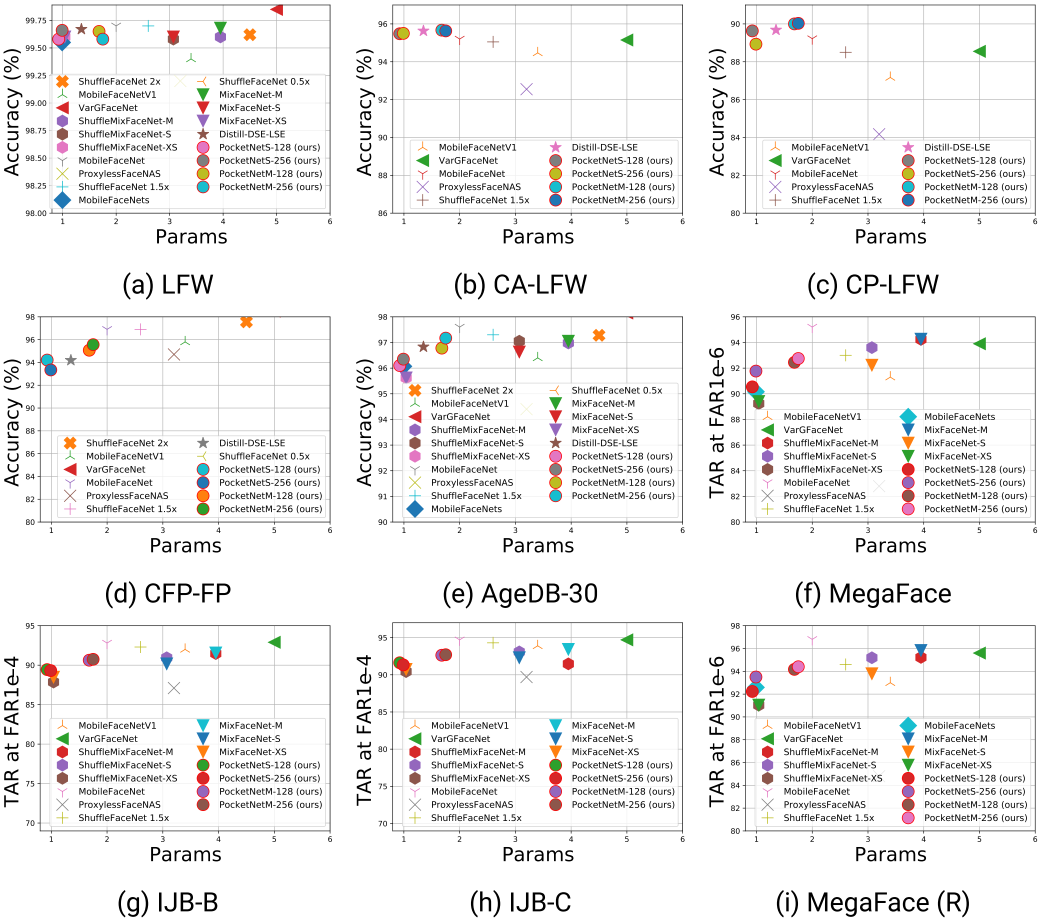


(a) ArcFace loss value of the model trained with KD vs. the model trained with multi-step KD over training iterations.



(b) KD vs. multi-step KD loss values over training iterations.

Figure 3.4.: Effect of multi-step KD on the student model convergence. It can be noticed that multi-step KD enables the model trained with ArcFace and multi-step KD losses to better converges in comparison to the case where the model is trained with ArcFace and KD losses (Figure 3.4a). Also, it can be observed that training with multi-step KD guides the model to learn feature representations that are more similar (in comparison to KD) to the teacher ones (Figure 3.4b). These figures are based on training the PocketNetS-128 network.



(a) LFW

(b) CA-LFW

(c) CP-LFW

(d) CFP-FP

(e) AgeDB-30

(f) MegaFace

(g) IJB-B

(h) IJB-C

(i) MegaFace (R)

Figure 3.5.: Number of parameters (in millions) vs. performance on LFW (accuracy), CA-LFW (accuracy), CP-LFW (accuracy), CFP-FP (accuracy), AgeDB-30 (accuracy), MegaFace (TAR at FAR1e-6), IJB-B (TAR at FAR1e-4), IJB-C (TAR at FAR1e-4) and MegaFace (R), (TAR at FAR1e-6). Our PocketNets are marked with circle marker and red edge color and are placed repeatedly in the top left corner, proving a SOTA trade-off between FR performance and compactness.

3.5. High-performing face recognition

This section presents the ElasticFace loss [27] aiming at enhancing the discriminative learning ability of FR. ElasticFace relaxes the fixed single margin value of margin-penalty softmax loss by deploying a random margin drawn from a normal distribution. The randomized margin assignment allows flexibility in the push for class separability, demonstrated by proving the superiority of ElasticFace loss over fixed-margin penalty losses on the mainstream benchmarks, using the same geometric transformation. We additionally extended this concept by guiding the assignment of the drawn margin values to put more attention on hardly classified samples. We provided a simple toy example with an 8-class classification problem to demonstrate the enhanced separability and robustness induced by our ElasticFace loss. To experimentally demonstrate the effect of our ElasticFace loss on face recognition accuracy, we report the results on nine different benchmarks. The achieved results are compared to the results reported in the recent SOTA. In a detailed comparison, compared to fixed margin penalties and recent SOTA, our ElasticFace loss enhanced the face recognition accuracy on most of the considered benchmarks, consequently extending SOTA face recognition performance on seven out of nine benchmarks and scoring close to the SOTA in the remaining two. This is especially the case in the benchmarks where the intra-class variation is extremely high, such as frontal-to-profile face verification (CFP-FP [236]) and large age gap face verification (AgeDB-30 [192]), which points to the generalizability induced by the proposed ElasticFace.

In the rest of this section, we will first introduce our proposed ElasticFace loss by building up to its definition starting from the basic softmax loss. This rationalization will include an experimental toy example demonstrating the effect of the proposed loss. Later on, the experimental setup and implementation details are introduced. This is followed by a detailed comparative discussion of the achieved results and a final conclusion.

3.5.1. ElasticFace

We propose in this work a novel learning loss strategy, ElasticFace loss, aiming at improving the accuracy of face recognition by targeting enhanced intra-class compactness and inter-class discrepancy in a flexible manner. Unlike previous works [80, 159, 268] that utilize a fixed penalty margin value, our proposed ElasticFace loss accommodates flexibility through relaxing this constraint by randomly drawing the margin value from a Gaussian distribution. Our proposed ElasticFace loss does not only target giving the model flexibility in optimizing the separability between and within the classes but also avoiding overfitting (thus enhancing generalizability) the model as it incorporates random margin values for each sample in each training iteration. The randomized margin penalty can be easily integrated into any of the angular margin-based softmax losses, which we demonstrate on two SOTA margin-based softmax losses. The angular margin-based losses and our ElasticFace loss extend over the softmax loss by manipulating the decision boundary to enhance intra-class compactness and inter-class discrepancy. Therefore, in this section, we first revisit the conventional softmax loss. Then, we present the modified version of softmax loss and the angular margin-based softmax loss. This leads up to presenting our proposed ElasticFace loss and an extended definition, the ElasticFace+, where the assignment of the drawn margins to training samples is linked to their proximity to their class centers.

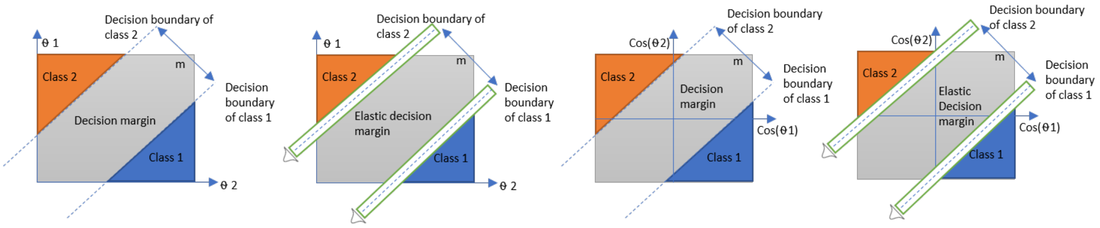
Softmax loss The widely used multi-class classification loss, softmax loss [160], refers to applying cross-entropy loss between the output of the logistic function (softmax activation function) and the ground-truth. Assume $x_i \in R^d$ is a feature representation of the i-th sample z_i and y_i is its corresponding class label (y_i integer in the range $[1, c]$). Given that c is the number of classes (identities), the output of the softmax activation function is defined as follows:

$$softmax(x_i, y_i) = \frac{e^{f_{y_i}}}{\sum\limits_{j=1}^{c} e^{f_j}} = \frac{e^{x_i W_{y_i}^T + b_{y_i}}}{\sum\limits_{j=1}^{c} e^{x_i W_j^T + b_j}}, \tag{3.6}$$

where f_{y_i} is the activation of the last fully-connected layer with weight vector W_{y_i} and bias b_{y_i}. W_{y_i} is the y_i-th column of weights $W \in R_c^d$ and b_{y_i} is the corresponding bias offset. The output of the softmax activation function is the probability of x_i being correctly classified as y_i. Given a mini-batch of size N, the cross-entropy loss function that measures the divergence between the model output and the ground-truth labels can be defined as follows:

$$L_{CE} = \frac{1}{N} \sum_{i \in N} -log \frac{e^{x_i W_{y_i}^T + b_{y_i}}}{\sum\limits_{j=1}^{c} e^{x_i W_j^T + b_j}}. \tag{3.7}$$

In a simple binary class classification, assuming that the input z_i belong to class 1, the model will correctly classify z_i if $W_1^T x_i + b1 > W_2^T x_i + b2$ and z_i will be classified as class 2 if $W_2^T x_i + b2 > W_1^T x_i + b1$. Therefore, the decision boundary of softmax loss is $x(W_1^T - W_2^T) + b1 - b2 = 0$. One of the main limitations of using softmax loss for learning face embeddings is that softmax loss does not explicitly optimize the feature representation needed for face verification as there is no restriction on the minimum distance between the class centers. Thus, training with softmax loss is less than optimal for achieving the maximum inter-class distances and the minimum intra-class distances. To mitigate this limitation, margin penalty-based cosine softmax loss was proposed as an alternative to the conventional softmax loss and it became a popular loss function for training face recognition models [80, 268, 159]. To get there, [159] has proposed a modified softmax loss (Cosine softmax loss) that optimized the angle cosine between features and the weights $cos(\theta)$ and then, incorporates a margin penalty on $cos(\theta)$.


Decision boundary of class 2
θ 1
Class 2
m
Decision margin
Decision boundary of class 1
Class 1
θ 2
Decision boundary of class 2
θ 1
Class 2
m
Elastic decision margin
Decision boundary of class 1
Class 1
θ 2
Decision boundary of class 2
Cos(θ2)
Class 2
m
Decision margin
Decision boundary of class 1
Cos(θ1)
Class 1
Decision boundary of class 2
Cos(θ2)
Class 2
m
Elastic Decision margin
Decision boundary of class 1
Cos(θ1)
Class 1

(a) ArcFace (b) ElasticFace-Arc (c) CosFace (d) ElasticFace-Cos

Figure 3.6.: Decision boundary of (a) ArcFace, (b) ElasticFace-Arc, (c) CosFace, and (d) ElasticFace-Cos for binary classification. The dashed blue line is the decision boundary. The gray area illustrates the decision margin.

Cosine softmax loss Following [80, 268, 159, 160], the bias offset, for simplicity, can be fixed to $b_{y_i} = 0$. The logit f_{y_i}, in this case, can be reformulated as: $x_i W_{y_i}^T = \|x_i\| \|W_{y_i}\| cos(\theta_{y_i})$, where θ_{y_i} is the angle between the weights of the last fully-connected layer W_{y_i} and the feature representation x_i. By fixing the weights norm and the feature norm to $\|W_{y_i}\| = 1$ and $\|x_i\| = 1$, respectively, and rescaling the $\|x_i\|$ to the constant s [268], the output of the softmax activation function is subject to the cosine of the angle θ_{y_i}. The modified softmax loss (L_{ML}) can be defined, as stated in [268, 159], as follows:

$$L_{ML} = \frac{1}{N} \sum_{i \in N} -log \frac{e^{s(cos(\theta_{y_i}))}}{e^{s(cos(\theta_{y_i}))} + \sum\limits_{j=1, j \neq y_i}^{c} e^{s(cos(\theta_j))}}. \tag{3.8}$$

In the previous binary example, assume that the input z_i belong to the class 1, z_i will be correctly classified if $cos(\theta 1) > cos(\theta 2)$. The decision boundary, in this case, is $cos(\theta 1) - cos(\theta 2) = 0$. Therefore, training with the modified (cosine) softmax loss emphasizes that the model prediction depends on the angle cosine between the features and the weights. However, and similar to conventional softmax loss, modified softmax loss does not explicitly optimize the feature representation needed for face verification. This motivated the introduction of the angular margin penalty-based losses [80, 268, 159].

Angular margin penalty-based loss In recent works [80, 268, 159], different types of margin penalty are proposed to push the decision boundary of softmax, and thus enhance intra-class compactness and inter-class discrepancy aiming at improving the accuracy of face recognition. The general angular margin penalty-based loss (L_{AML}) is defined as follows:

$$L_{AML} = \frac{1}{N}\sum_{i\in N} -log\frac{e^{s(cos(m_1\theta_{y_i}+m_2)-m_3)}}{e^{s(cos(m_1\theta_{y_i}+m_2)-m_3)} + \sum\limits_{j=1,j\neq y_i}^{c} e^{s(cos(\theta_j))}}, \tag{3.9}$$

where m_1, m_2 and $m3$ are the margin penalty parameters proposed by SphereFace [159], ArcFace [80] and CosFace [268], respectively. In SphereFace [159], multiplicative angular margin penalty is deployed by multiplying θ by $m_1 = \alpha$ and setting $m_2 = 0$ and $m_3 = 0$ ($\alpha > 1.0$). The decision boundary of SphereFace is then $cos(m_1\theta_{y_i}) - cos(\theta_j) = 0$. Differently, CosFace [268] proposed additive cosine margin penalty by setting $m_1 = 1$, $m_2 = 0$ and $m_3 = \alpha$ $(0 < \alpha < 1 - cos(\frac{\pi}{4}))$. The decision boundary of CosFace is then $cos(\theta_{y_i}) - cos(\theta_j) - m3 = 0$. Later, ArcFace [80] proposed additive angular margin penalty by setting up $m_1 = 1$, $m_2 = \alpha$ and $m_3 = 0$ $(0 < \alpha < 1.0)$. The decision boundary of ArcFace is then $cos(\theta_{y_i} + m2) - cos(\theta_j) = 0$.

Even though, ArcFace [80], CosFace [268] and SphereFace [159] introduced the important concept of angular margin penalty on softmax loss, selecting a single optimal margin value (α) is a critical issue in these works. By setting up $m_1 = 1$, $m_2 = 0$ and $m_3 = 0$, ArcFace, CosFace and SphereFace are equivalent to the modified softmax loss. A reasonable choice could be selecting a large margin value that is close to the margin upper bound to enable higher separability between the classes. However, when the margin value is too large, the model fails to converge, as stated in [268]. ArcFace, CosFace, and SphereFace selected the margin value through trial and error assuming that the samples are equally distributed in geodesic space around the class centers. However, this assumption could not be held when there are largely different intra-class variations leading to less than optimal discriminative feature learning, especially when there are large variations between the samples/classes in the training dataset. This motivated us to propose ElasitcFace loss by utilizing random margin penalty values drawn from a Gaussian distribution aiming at giving the model space for flexible class separability learning.

Elastic angular margin penalty-based loss (ElasticFace) The proposed ElasticFace loss is extended over the angular margin penalty-based loss by deploying random margin penalty values drawn from a Gaussian distribution. Formally, the probability density function of a normal distribution is defined as follows:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}, \tag{3.10}$$

where μ is the mean of the distribution and σ is its standard deviation. To demonstrate and prove our proposed elastic margin, we chose to integrate the randomized margin penalty in ArcFace (noted as ElasticFace-Arc) and CosFace (noted as ElasticFace-Cos) as they proved to have clearer geometric interpretation and achieved higher accuracy on mainstream benchmarks than the earlier SphereFace. ElasticFace-Arc (L_{EArc}) can be defined as follows:

$$L_{EArc} = \frac{1}{N} \sum_{i \in N} -log \frac{e^{s(cos(\theta_{y_i} + E(m,\sigma)))}}{e^{s(cos(\theta_{y_i} + E(m,\sigma)))} + \sum_{j=1, j \neq y_i}^{c} e^{s(cos(\theta_j))}}, \tag{3.11}$$

and ElasticFace-Cos (L_{ECos}) can be defined as follows:

$$L_{ECos} = \frac{1}{N} \sum_{i \in N} -log \frac{e^{s(cos(\theta_{y_i}) - E(m,\sigma))}}{e^{s(cos(\theta_{y_i}) - E(m,\sigma))} + \sum_{j=1, j \neq y_i}^{c} e^{s(cos(\theta_j))}}, \tag{3.12}$$

where $E(m, \sigma)$ is a normal function that return a random value from a Gaussian distribution with the mean m and the standard deviation σ.

The decision boundaries of ElasticFace-Arc and ElasticFace-Cos are $cos(\theta_{y_i} + E(m, \sigma)) - cos(\theta_j) = 0$ and $cos(\theta_{y_i}) - cos(\theta_j) - E(m, \sigma) = 0$, respectively. Figure 3.6 illustrates the decision boundary of ArcFace, ElasticFace-Arc, CosFace and ElasticFace-Cos. The sample push towards its center during training using ElasticFace-Arc and ElasticFace-Cos varies between training samples, based on the margin penalty drawn from $E(m, \sigma)$. During the training phase, a new random margin is generated for each sample in each training iteration. This aims at giving the model flexibility in the push for class separability. When σ is 0, our ElasticFace-Arc and ElasticFace-Cos are equivalent to ArcFace and CosFace, respectively.

ElasticFace+ We propose an extension to our ElasticFace, the ElasticFace+, that observes the intra-class variation during each training iteration and use this observation to assign a margin value to each sample based on its proximity to its class center. This causes the samples that are relatively far from their class center to be pushed with a larger penalty margin to their class center. This aims at giving the model space to push the samples that are relatively far from their class center to be closer to their centers while giving less penalty attention to the samples that are already close to their center. To achieve that, the output of the Gaussian distribution function (Equation 3.10) is sorted (descending) based on $cos(\theta_{y_i})$ value. Thus, the sample with small $cos(\theta_{y_i})$ will be pushed with large value from $E(m, \sigma)$ function, and vice versa.

Loss	LFW		AgeDB-30		CALFW		CPLFW		CFP-FP		-
	Acc (%)	BC	Acc(%)	BC	Acc(%)	BC	Acc(%)	BC	Acc(%)	BC	Sum BC
ArcFace (m=0.55)	**99.52**	3	94.58	1	93.82	2	89.05	1	95.24	1	8
ArcFace (m=0.5)	99.46	2	**94.83**	3	**93.88**	3	**89.72**	3	95.36	2	**13**
ArcFace(m=0.45)	99.43	1	94.66	2	93.80	1	89.42	2	**95.53**	3	9
ElasticFace-Arc(m=0.5, σ=0.0125)	**99.53**	4	94.80	1	93.68	2	89.72	3	95.43	1	11
ElasticFace-Arc(m=0.5, σ=0.0175)	99.47	1	**95.13**	4	93.67	1	89.53	2	95.54	3	11
ElasitcFace-Arc(m=0.5,σ=0.025)	99.52	3	94.95	3	93.78	3	89.50	1	95.44	2	12
ElasitcFace-Arc(m=0.5,σ=0.05)	99.52	3	94.82	2	**93.90**	4	**89.79**	4	**95.59**	4	**17**
ElasitcFace-Arc+ (m=0.5,σ=0.0125)	**99.53**	4	95.00	2	93.68	1	**89.58**	4	95.40	2	13
ElasitcFace-Arc+ (m=0.5, σ=0.0175)	**99.53**	4	95.07	3	93.95	3	89.37	1	**95.67**	4	**15**
ElasitcFace-Arc+ (m=0.5, σ=0.025)	99.42	1	**95.15**	4	93.73	2	89.48	2	95.36	1	10
ElasitcFace-Arc+ (m=0.5,σ=0.05)	99.45	2	94.83	1	**94.00**	4	89.50	3	95.56	3	13

Table 3.8.: Parameter selection for ElasticFace-Arc and ElasticFace-Arc+. The Borda count (BC) is reported separately for each of training settings (ArcFace, ElasticFace-Arc and ElasticFace-Arc+) and each of the evaluation benchmarks. The final σ and m parameters are selected based on the highest BC sum. In all settings, the used architecture is ResNet-50 trained on CASIA [286].

Parameter selection The probability density function has its peak around m [211]. Thus, when ElasticFace is integrated into ArcFace [80], we select the best margin value (as a single value) by training three instances of ResNet-50 [107] on CASIA [286] with ArcFace loss using margins equal to 0.45, 0.50 and 0.55, respectively, to assure the advised margin in [80]. Then, based on the sum of the performance ranking Borda count on LFW [114], AgeDB-30 [192], CA-LFW [296], CP-LFW [295], and CFP-FP [236], we select the margin that achieved the highest Borda count sum and set it as m for $E(m, \sigma)$ function, where our goal is to use the most optimal margin. The best margin observed in our experiment, in this case, is 0.5 (Table 3.8). To select the σ value for $E(m, \sigma)$ function, we conducted additional experiments on four instances of ResNet-50 trained on CASIA [286] with our proposed ElasticFace-Arc by setting up the σ to one of these values 0.0125, 0.015, 0.025 and 0.05. Then, we rank these models based on the sum of the performance

Loss	LFW		AgeDB-30		CALFW		CPLFW		CFP-FP		-
	Acc (%)	BC	Acc (%)	BC	Acc (%)	BC	Acc (%)	BC	Acc (%)	BC	Sum BC
CosFace (m=0.4)	99.42	1	**94.65**	3	93.45	1	**90.38**	3	95.30	1	9
CosFace (m=0.35)	**99.55**	3	94.55	2	**93.78**	3	89.95	1	95.31	2	**11**
CosFace (m=0.3)	99.45	2	94.45	1	93.46	2	90.12	2	**95.39**	3	10
ElasticFace-Cos (m=0.35,σ=0.0125)	99.45	2	94.72	1	93.83	1	90.12	2	95.47	3	9
ElasticFace-Cos (m=0.35,σ=0.0175)	99.50	3	94.77	3	**93.97**	4	90.10	1	95.30	2	13
ElasticFace-Cos (m=0.35,σ=0.025)	99.42	1	**94.85**	4	93.88	2	90.20	3	95.21	1	11
ElasticFace-Cos (m=0.35,σ=0.05)	**99.52**	4	94.77	3	93.93	3	**90.38**	4	**95.52**	4	**18**
ElasticFace-Cos+(m=035, σ=0.0125	99.38	1	94.50	2	93.67	3	89.85	1	95.20	1	8
ElasticFace-Cos+(m=035, σ=0.0175)	99.45	2	**94.97**	4	93.48	1	89.98	2	95.23	2	11
ElasticFace-Cos+(m=035, σ=0.025)	**99.55**	4	94.63	3	93.65	2	**90.28**	4	**95.47**	4	**17**
ElasticFace-Cos+(m=035, σ=0.05)	99.48	3	94.45	1	**93.77**	4	90.01	3	95.26	3	14

Table 3.9.: Parameter selection for ElasticFace-Cos and ElasticFace-Cos+. The Borda count (BC) is reported separately for each of training settings (ArcFace, ElasticFace-Cos and ElasticFace-Cos+) and each of the evaluation benchmarks. The final σ and m parameters are selected based on the highest BC sum. In all settings, the used architecture is ResNet-50 trained on CASIA [286]

ranking Borda count across all datasets. Finally, the σ value is chosen based on the highest Borda count sum. The best σ observed in our experiment, in this case, is 0.05 (Table 3.8). Similarly, we follow the same procedure to select the parameters (m and σ) for ElasticFace-Cos. We first choose the best margin value by training three different instances of ResNet-50 on CASIA [286] with CosFace using a margin equal to 0.3, 0.35, and 0.40. The best m observed in our experiment based on the sum of the performance ranking Borda count across all evaluated datasets, in this case, is 0.035 (Table 3.9). Similar to σ selection approach of ElasticFace-Arc, we train four instance of ElasticFace-Cos to choose the best σ for $E(m, \sigma)$ function. The best observed σ in our experiment, in this case, is 0.05 (Table 3.9). For ElasticFace-Cos+ and ElasticFace-Arc+, we followed the exact approach of parameter selection for ElasticFace-Arc and ElasticFace-Cos. The best observed σ for ElasticFace-Arc+ is 0.0175 and the best observed one for ElasticFace-Cos+ is 0.025 (Table 3.8 and 3.9). These selected parameters are used to train our solutions (training details in Section 4.4) evaluated in Section 5.5.

Toy example To demonstrate the robustness and the class separability induced by our proposed ElasticFace and ElasticFace+, we present a simple toy example by training three ResNet-18 networks [107] to classify eight different identities and produce 2-D feature embeddings. All the networks are trained with a small batch size of 128 for 11200 iterations with stochastic gradient descent (SGD) and an initial learning rate of 0.1. The learning rate is reduced by a factor of 10 after 1680, 2800, 3360, and 8400 training iterations. To demonstrate a classification case where the classes are not identically varied, these eight identities are selected to have four identities with small intra-class variation and another four identities with a large intra-class variation (measured as the average of all intra-class comparison scores for each identity). These identities were chosen from all the identities with more than 400 images per identity in the MS1MV2 dataset [80], we note this selected subset as MS1MV2-400. From these identities, we select the four identities with the highest intra-class variation and the four with the lowest intra-class variation. The features for this selection were extracted using an open-source [1] ResNet-100 [107] model trained with ArcFace loss [80], and the comparison is performed by a cosine similarity. The set of the selected eight identities is noted as MS1MV2-8. We use MS1MV2-8 to train the toy networks with ArcFace (m=0.5), ElasticArcFace (m=0.5, σ=0.05), and ElasticArcFace+ (m=0.5, σ=0.0175), based on our parameter selection. Figure 3.7 shows the classification of MS1MV2-8 for each of the experimental settings. In each of the plots in Figure 3.7a, 3.7b and 3.7c, we calculate the angle between each consecutive identities to demonstrate the separability between the identities in the arc space (inter-class discrepancy). The optimal inter-class discrepancy may be achieved if the angle, in degree, between each of consecutive identities is close to 45 degrees i.e. 360 / 8. Also, we calculate the mean of the standard deviation of each class feature embeddings to illustrate intra-class compactness induced by ArcFace, ElasticFace, and ElasticFace+. The smaller standard deviation (shown at the edge of each class in Figure 3.7), in this case, indicates higher intra-class compactness. It can be noticed that our EalsticFace and EalsticFace+ achieved better intra-class compactness and inter-class discrepancy than ArcFace, while the differences in inter-class variation between EalsticFace and EalsticFace+ are minor (Figures 3.7a 3.7c, and 3.7b).

3.5.2. Experimental setup

Training settings: The network architecture we used to demonstrate our ElasticFace is the ReseNet-100 [107]. This was motivated by the wide use of this architecture in the SOTA FR solutions [80, 11, 84, 250, 117]. We follow the common setting [80, 11, 117] to set the scale parameter s to 64. We set the mini-batch size to 512 and train our model on one Linux machine (Ubuntu 20.04.2 LTS) with Intel(R) Xeon(R) Gold 5218 CPU 2.30GHz, 512 G RAM, and 4 Nvidia GeForce RTX 6000 GPUs. The proposed ElasticFace models are implemented using Pytorch [210]. All models are trained with Stochastic Gradient Descent (SGD) optimizer with an initial learning rate of 1e-1. We set the momentum to 0.9 and the weight decay to 5e-4. The learning rate is divided by 10 at 80k, 140k, 210k, and

[1] https://github.com/deepinsight/insightface

Method	Training Dataset	LFW	AgeDB-30	CALFW	CPLFW	CFP-FP
		Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
ArcFace[80] (CVPR2019)	MS1MV2 [103, 80]	99.82 (3)	98.15	95.45	92.08	98.27
CosFace[268] (CVPR2018)	private	99.73	-	-	-	-
Dynamic-AdaCos[292] (CVPR2019)	clean MS1M [103, 292] + CASIA [286]	99.73	-	-	-	-
AdaptiveFace[156] (CVPR2019)	clean MS1M [103, 281]	99.62	-	-	-	-
UniformFace[84] (CVPR2019)	clean MS1M [103, 80] + VGGFace2 [42]	99.8	-	-	-	-
GroupFace[139] (CVPR2020)	clean MS1M [103, 80]	**99.85** (1)	98.28 (3)	**96.20** (1)	93.17	98.63
CircleLoss[250] (CVPR2020)	clean MS1M [103, 250]	99.73	-	-	-	96.02
CurricularFace[117] (CVPR2020)	MS1MV2 [103, 80]	99.80	98.32 (2)	**96.20** (1)	93.13	98.37
Dyn-arcFace [128] (MTAP2021)	clean MS1M [103, 80]	99.80	97.76	-	-	94.25
MagFace[185] (CVPR2021)	MS1MV2 [103, 80]	99.83 (2)	98.17	96.15	92.87	98.46
Partial-FC-ArcFace [11] (ICCVW2021)	MS1MV2 [103, 80]	99.83 (2)	98.20	96.18 (2)	93.00	98.45
Partial-FC-CosFace [11] (ICCVW2021)	MS1MV2 [103, 80]	99.83 (2)	98.03	**96.20** (1)	93.10	98.51
ElasticFace-Arc (ours)	MS1MV2 [103, 80]	99.80	**98.35** (1)	96.17 (3)	93.27 (2)	98.67 (2)
ElasticFace-Cos (ours)	MS1MV2 [103, 80]	99.82 (3)	98.27	96.03	93.17	98.61 (3)
ElasticFace-Arc+ (ours)	MS1MV2 [103, 80]	99.82 (3)	**98.35** (1)	96.17 (3)	**93.28** (1)	98.60
ElasticFace-Cos+ (ours)	MS1MV2 [103, 80]	99.80	98.28 (3)	96.18 (2)	93.23 (3)	**98.73** (1)

Table 3.10.: The achieved results on the LFW, AgeDB-30, CALFW, CPLFW, and CFP-FP benchmarks. On large age gape (AgeDB-30) and frontal-to-profile face comparisons (CFP-FP), the ElasticFace solutions consistently extend state-of-the-art performances. ElasticFace scores very close to the state-of-the-art on LFW and CALFW. All decimal points are provided as reported in the respective works. The top performance in each benchmark is in bold. The top three performances in each benchmark are noted with rank number between parentheses (1,2 or 3).

280k training iterations. The total number of training iteration is 295K, which corresponds to the number of margin sampling from the normal distribution. During the training, we use random horizontal flipping with a probability of 0.5 for data augmentation. The networks are trained (and evaluated) on images of the size $112 \times 112 \times 3$ to produce $512 - d$ feature embeddings. These images are aligned and cropped using the Multi-task Cascaded Convolutional Networks (MTCNN) [289] following [80]. All the training and testing images are normalized to have pixel values between -1 and 1.

Training dataset: We follow the trend in recent works [80, 11, 117, 185] in using the MS1MV2 dataset [80] to train the investigated models with the proposed ElasticFace loss. This enables a direct comparison with the SOTA as will be shown in Section 5.5. The MS1MV2 is a refined version [80] of the MS-Celeb-1M [103] containing 5.8M images of 85K identities.

Evaluation benchmarks and metrics: To demonstrate the effect of our proposed ElasticFace on FR accuracy and enable a wide comparison to SOTA, we report the achieved results on nine benchmarks. These benchmarks are of a diverse nature, where some represent a special vulnerabilities of FR. The nine benchmarks are 1) LFW [114], 2) AgeDB-30 [192], 3) CA-LFW [296], 4) CP-LFW [295], 5) CFP-FP [236], 6) IJB-B [274], 7) IJB-C [183], 8) MegaFace [136], and 9) MegaFace (R) [80]. The FR performance on LFW, AgeDB-30, CA-LFW, CP-LFW, and CFP-FP is reported as verification accuracy, following their evaluation protocol. The performance on IJB-C and IJB-B is reported (as defined in [274, 183]) as TAR at FAR of 1e-4. The MegaFace and MegaFace(R) benchmarks report the FR performance as Rank-1 correct identification rate and as TAR at FAR=1e–6 verification accuracy.

We acknowledge the verification and identification performance evaluation metrics reported in ISO/IEC 19795-1 [123]. However, to enhance the reproducibility and comparability, we follow the evaluation protocols and metrics used in each of the benchmarks as listed above.

Method	Training Dataset	IJB-B	IJB-C	MegaFace (R)		MegaFace	
		TAR at FAR1e–4 (%)	TAR at FAR1e–4 (%)	Rank-1 (%)	TAR at FAR1e–6 (%)	Rank-1 (%)	TAR at FAR1e–6 (%)
ArcFace[80] (CVPR2019)	MS1MV2 [103, 80]	94.2	95.6	98.35	98.48	81.03	96.98
CosFace[268] (CVPR2018)	private	-	-	-	-	**82.72** (1)	96.65
Dynamic-AdaCos[292] (CVPR2019)	clean MS1M [103, 292] + CASIA [286]	-	92.40	97.41	-	-	-
AdaptiveFace[156] (CVPR2019)	clean MS1M [103, 281]	-	-	95.02	95.61	-	-
UniformFace[84] (CVPR2019)	clean MS1M [103, 80] + VGGFace2 [42]	-	-	-	-	79.98	95.36
GroupFace[139] (CVPR2020)	clean MS1M [103, 80]	94.93	96.26	98.74 (3)	98.79	81.31 (2)	97.35 (2)
CircleLoss[250] (CVPR2020)	clean MS1M [103, 250]	-	93.95	98.50	98.73	-	-
CurricularFace[117] (CVPR2020)	MS1MV2 [103, 80]	94.8	96.1	98.71	98.64	81.26 (3)	97.26
Dyn-arcFace [128] (MTAP2021)	clean MS1M [103, 80]	-	-	-	-	-	-
MagFace[185] (CVPR2021)	MS1MV2 [103, 80]	94.51	95.97	-	-	-	-
Partial-FC-ArcFace [11] (ICCVW2021)	MS1MV2 [103, 80]	94.8	96.2	98.31	98.59	-	-
Partial-FC-CosFace [11] (ICCVW2021)	MS1MV2 [103, 80]	95.0	96.4	98.36	98.58	-	-
ElasticFace-Arc (ours)	MS1MV2 [103, 80]	95.22 (3)	96.49 (3)	**98.81** (1)	**98.92** (1)	80.76	97.30
ElasticFace-Cos (ours)	MS1MV2 [103, 80]	95.30 (2)	96.57 (2)	98.70	98.75	81.01	97.31 (3)
ElasticFace-Arc+ (ours)	MS1MV2 [103, 80]	95.09	96.40	98.80 (2)	98.83 (3)	80.68	**97.44** (1)
ElasticFace-Cos+ (ours)	MS1MV2 [103, 80]	**95.43** (1)	**96.65** (1)	98.62	98.85 (2)	80.08	97.29

Table 3.11.: The achieved results on the IJB-B, IJB-C, MegaFace (R), and MegaFace benchmarks. On the earlier three, and the verification accuracy of the fourth, the ElasticFace solutions consistently extend state-of-the-art performances. ElasticFace scores very close to the state-of-the-art on MegaFace. MegaFace has been refined in [80] to MegaFace (R) as it contains many face images with wrong labels. All decimal points are provided as reported in the respective works. The top performance in each benchmark is in bold. The top three performances in each benchmark are noted with rank number between parentheses (1,2 or 3).

3.5.3. Results

Tables 3.10 and 3.11 presents the achieved results on the nine considered benchmarks. The main observation is that our proposed ElasticFace solutions scored beyond the SOTA in seven out of the nine benchmarks, and very close to the SOTA in the remaining two. When possible, and to build a fair comparison, the results of previous works are reported when trained on the MS1MV2 [103, 80] (or a refined variant of MS1M [103]) as the ElasticFace results are based on training on this dataset. The proposed ElasticFace ranked first in comparison to the SOTA on the benchmarks AgeDB-30, CP-LFW, CFP-FP, IJB-B, IJB-C, MegaFace (R), and MegaFace (verification). In the remaining benchmarks, ElasticFace solutions ranked second on CA-LFW, third on LFW, and fourth on MegaFace (identification).

A main outcome of the evaluation is concerning the databases with very large intra-user variations. These are the large age gape benchmark (AgeDB-30) and the frontal-to-profile face verification benchmark (CFP-FP). On AgeDB-30, our ElasticFace-Arc solution scored an accuracy of 98.35%, while the top SOTA performance was 98.32% scored by the CurricularFace [117]. On CFP-FP, our ElasticFace-Arc+ solution scored an accuracy of 98.73% and our ElasticFace-Arc scored an accuracy of 98.67%, while the top SOTA performances were 98.51% scored by the Partial-FC-CosFace [11] solution and 98.46% scored by the MagFace [185]. This significantly enhanced performance in the extreme intra-class variation scenarios points out the generalizability induced by the ElasticFace loss. CA-LFW and CP-LFW also considered age gaps and pose variation, however, with a lower variation than AgeDB-30 and CFP-FP. In CA-LFW, ElasticFace-Cos+ scored a close second with 96.18% accuracy, with the lead going to the CurricularFace [117] with 96.20% accuracy. In CP-LFW, our ElasticFace-Arc+ is ranked first with 93.28% accuracy, while the top SOTA performance was 93.17% accuracy scored by the GroupFace [139]. On the LFW benchmark [114], which is one of the oldest and nearly saturated benchmarks reported in the recent works, our ElasticFace-Cos and ElasticFace-Arc+ solutions scored an accuracy of 98.82%, very close behind the GroupFace [139] with 99.85%.

In Table 3.11, on IJB-B benchmark, our ElasticFace-Cos+ scored a TAR at FAR1e–4 of 95.43%, far ahead of the Partial-FC-CosFace [11] and the GroupFace [139] with 95.0% and 94.93%, respectively. Similarly, on the IJB-C benchmark, our ElasticFace-Cos+ scored a TAR at FAR1e–4 of 96.65%, ahead of the Partial-FC-CosFace [11] and the GroupFace [139] with 96.4% and 96.36% respectively. On the MegaFace (R), our ElasticFace-Arc scored 98.81% Rank-1 identification rate and 98.92% TAR at FAR1e–6, ahead of the previous lead solution, the GroupFace [139] with 98.74% and 98.79%, respectively. On the MegaFace benchmark, our ElasticFace-Cos scored Rank-1 identification rate of 81.01%, close to the SOTA 82.72% score by CosFace [268], noting that CosFace was trained on a private dataset. On the same benchmark (MegaFace), our ElasticFace-Arc+ ranked first with 97.44% TAR at FAR1e–6, while the top SOTA performances were 97.35% scored by the GroupFace [139]. It must be mentioned that the MegaFace benchmark has been refined in [80] to MegaFace (R) as it contains many face images with wrong labels as reported in [80].

In comparison to the closely defined losses in ArcFace [80], CosFace [268], and Partial-FC [11] solutions, our ElasticFace models did prove to provide a strong performance edge by scoring higher recognition performance on most benchmarks. When it comes to comparing ElasticFace and ElasticFace+, the ElasticFace-Arc and ElasticFace-Arc+ did achieve very close performances when considering all benchmarks. On the other hand, the ElasticFace-Cos+ did outperform ElasticFace-Cos on most benchmarks.

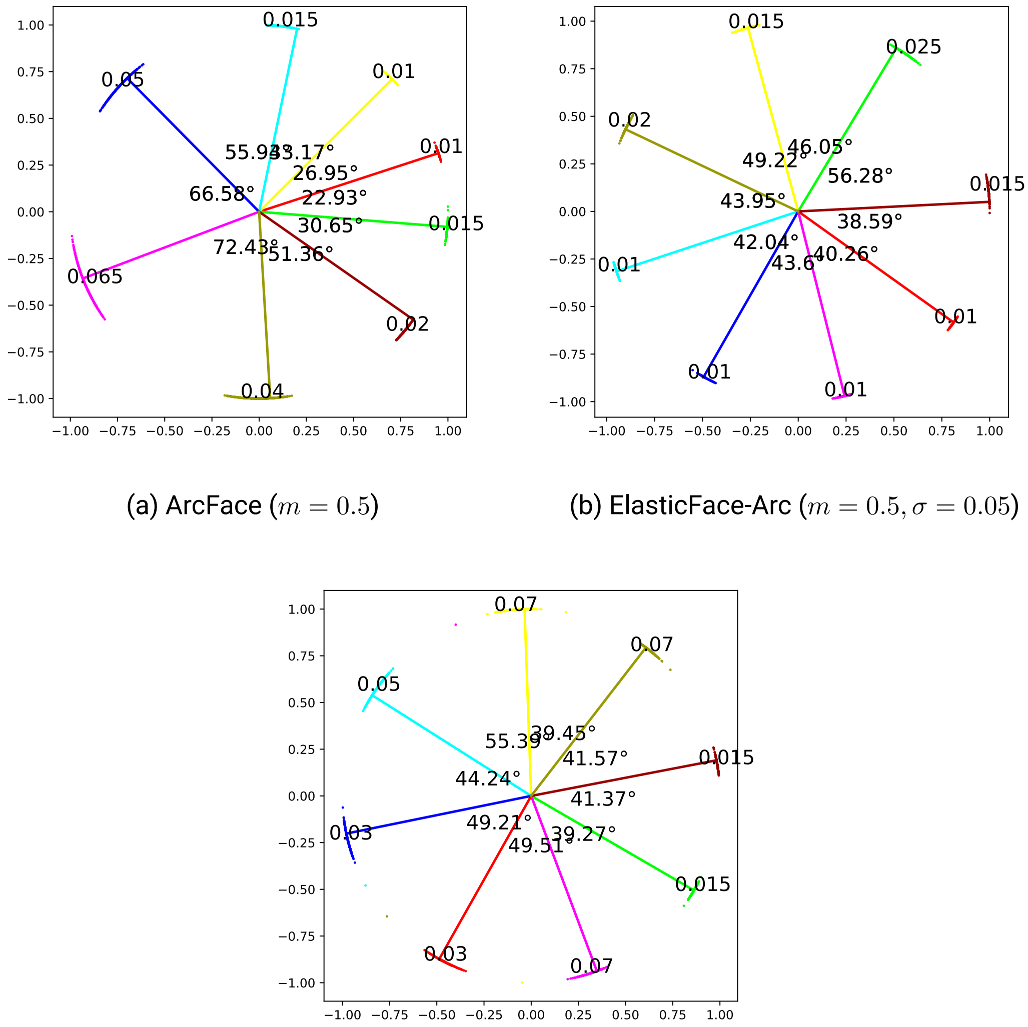
We acknowledge that the Partial-FC [11] solution reported additional performance rates when trained on their new collected database, the Glint360K [11]. However, this database could not be acquired as it requires an account on a cloud platform, that in itself requires a SIM card registered in a specific country, which is very restrictive and we do not have access to. Therefore, and for a fair comparison, we opted to compare our results with the Partial-FC results when trained on the same dataset that our ElasticFace solution is using, the MS1MV2 [103, 80] dataset.

The slightly increased training computational cost is a minor limitation of our proposed ElasticFace. Training the ResNet-100 model on MS1MV2 dataset with CosFace or ArcFace using the specified machine and training details described in Section 4.4 requires around 57 hours. This training time is increased by around one minute for ElasticFace and by 11 hours for ElasticFace+. The minor increase in the ElasticFace training time is caused by the sampling of the margin values, while the larger increase in ElasticFace+ training time is additionally caused by the sorting algorithms.

On a less technical note, we stress that our efforts in the advancement of FR are aimed at enhancing the security, convenience, and life quality of the members of society, e.g. enabling convenient access to financial and health services [85] and enhancing the security of border checks within clear legal frameworks and users consent. We acknowledge and reject the possible malicious or illegal use of this and other technologies.

3.5.4. Discussion

This section presented an elastic margin penalty loss (ElasticFace) that avoids setting a single constant penalty margin. Our motivation considers that real training data is inconsistent in terms of inter and intra-class variation, and thus the assumption made by many margin softmax losses that the geodesic distance between and within the different identities can be equally learned using a fixed margin is less than optimal. We, therefore, relax this fixed margin constrain by using a random margin value drawn from a normal distribution in each training iteration. In an extended definition, the assignment of these margin values to training samples corresponds to their proximity to their class centers. We evaluated our ElasticFace loss, in comparison to SOTA FR approaches, on nine different benchmarks. This evaluation demonstrated that our ElasticFace solution consistently extended SOTA FR performance on most benchmarks (seven out of nine). This was specifically apparent in the challenging benchmarks with large intra-class variations, such as large age gaps and frontal-to-profile face comparisons.



(a) ArcFace ($m = 0.5$)

(b) ElasticFace-Arc ($m = 0.5, \sigma = 0.05$)

(c) ElasticFace+ ($m = 0.5, \sigma = 0.0175$)

Figure 3.7.: Toy example of 3 ResNet-18 networks trained under different experimental settings. The 2-D features are normalized. Thus, the feature embeddings are allocated around the class centers in the arc space with a fixed radius. The numbers next to each class center indicate the mean of the standard deviation of each class feature embeddings. The angle in degree are calculated between each two consecutive classes to illustrate the decision margin between the classes. One can noticed that feature produced by ElasticFace and ElasticFace+ are more equally distributed around the class centers than ArcFace, in the arc space. Same colors always indicates same class across plots.

3.6. Summary

This chapter proposed efficient and accurate FR models.

With a focus on enabling FR in use-cases that are extremely limited by computational complexity, this chapter presented efficient FR architectures, MixFaceNet, for accurate face verification and identification [23]. This chapter presented three variants of MixFaceNet with different levels of computational complexity. Additionally, MixFaceNets were extended with channel shuffle operation, aiming at increasing the discriminative learning ability of MixFaceNet. Under the same level of computational complexity, MixFaceNets outperformed the recent efficient models proposed in the literature on the mainstream benchmarks. Section 3.3 provided an answer to RQ1.1 by designing extremely efficient FR architecture, MixFaceNet, and proving its practical value by presenting its high performance on mainstream benchmarks.

This chapter then focused on two aspects. First, it presented a novel approach to automate FR architecture design. The presented approach utilized the NAS algorithm to learn an FR-specific lightweight network, namely PocketNets [38]. Extensive experiment evaluations on mainstream benchmarks have shown that PocketNets offer a SOTA trade-off between model compactness and verification performance. This chapter responded to RQ1.2 by providing an ablation study on the NAS training dataset source and an extensive experiment evaluations of PocketNets on mainstream benchmarks. Second, this chapter presented a multi-step KD training paradigm in which the knowledge learned by a teacher model is transferred to the student model in a step-wise manner. The conducted ablation study and achieved results by the proposed multi-step KD proved the effectiveness of multi-step KD in achieving higher verification performance compared to conventional KD, providing an answer to RQ1.3.

Lately, this chapter presented a novel elastic margin-penalty softmax loss, namely ElasticFace [27]. The proposed ElasticFace deployed random margin penalty values to give the model space for flexible class separability learning, thus enhancing intra-class compactness and inter-class separability. This chapter provided an answer to RQ1.4 by empirically proving the superiority of ElasticFace over over fixed-margin penalty losses on the mainstream benchmarks, using the same geometric transformation. The next chapter focuses on a different aspect in this work, efficiently reducing the effect of the masked face on FR performance.

4. The emerging challenge of masked face recognition

Chapter 3 was concerned with designing efficient and high-performing FR models. This chapter proposes a solution to deal with emerging and unusual challenge for FR posed by wearing a facial masked during COVID-19 pandemic, as response to RQ2.1. Also, this chapter present a summary of Masked Face Recognition (MFR) competition designed to motivate solutions aiming at enhancing the FR accuracy of masked faces. This chapter is based on the published papers [28, 29].

4.1. Introduction

FR is one of the preferable biometric recognition solutions due to its contactless nature and the high accuracy achieved by FR algorithms. FR systems have been widely deployed in many application scenarios such as automated border control, surveillance, as well as convenience applications [101, 166, 14]. However, these systems are mostly designed to operate on none occluded faces. After the current COVID-19 pandemic, wearing a protective face mask has been imposed in public places by many governments to reduce the rate of COVID-19 spread. This new situation raises a serious unusually challenge for the current FR systems. Recently, several studies have evaluated the effect of wearing a face mask on FR accuracy [64, 82, 201, 202]. These studies have reported the negative impact of masked faces on FR performance. The main conclusion of these studies [64, 82, 201, 202] is that the accuracy of FR algorithm with a masked face is significantly degraded, in comparison to unmasked face.

Motivated by this new circumstance this work propose a new approach to reduce the negative impact of wearing a facial mask on FR performance. The presented solution is designed to operate on top of existing FR models and thus, avoid retraining existing solutions developed for unmasked FR. Recent works either proposed to train FR models with simulated masked faces [13] or to train a model to learn the periocular area of the face images exclusively [151]. Unlike these, our proposed solution does not require any modification or training of the existing FR model. This goal is achieved by proposing the Embedding Unmasking Model (EUM) operated on the embedding space. The input for EUM is feature embedding extracted from the masked face, and its output is new feature embedding similar to an embedding of an unmasked face of the same identity, whereas, it is dissimilar from any other embedding of other identities. To achieve that through our EUM, a novel loss function, SRT is proposed to guide the EUM during the training phase. The SRT shares the same learning objective with the triplet loss i.e. it enables the model to minimize the distance between genuine pairs and maximize the distance between imposter pairs. Nonetheless, unlike triplet loss, the SRT can dynamically self-adjust its learning objective by focusing on minimizing the distance between the genuine pairs when the distance between the imposter pairs is deemed to be sufficient.

The presented approach is evaluated on top of three FR models, ResNet-100 [107], ResNet-50 [107] and MobileFaceNet [49] trained with the loss function, Arcface loss [80], to validate the feasibility of adopting our solution on top of different deep neural network architectures. With a detailed evaluation of the proposed EUM and SRT, The verification performance gain by the proposed approach is reported on two real masked face datasets [13, 64] and two synthetically generated masked face datasets. We further experimentally supported our theoretical motivation behind our SRT loss by comparing its performance with the conventional triplet loss. The overall verification result showed that our proposed approach improved the performance in most of the experimental settings. For example, when the probes are masked, the achieved FMR100 measures (the lowest FNMR for FMR ≤ 1.0 %) by our approach on top of MobileFaceNet are reduced by $\sim$ 28% and 26% on the two real masked face evaluation datasets.

In the rest of the chapter, the related works focusing on masked FR are discussed in in Section 4.2. Then, Section 4.3 presents the proposed EUM architecture and our SRT loss. Section 4.4 presents the experimental setups and implementation details. Section 4.5.3 presents a summary of MFR competition. Section 4.5 presents and discuss the achieved results. Finally, a set of conclusions are drawn in Section 4.7.

4.2. Related work

In recent years, significant progress has been made to improve FR verification performance with essentially non-occluded faces. Several previous works [205, 205, 127, 285] addressed general face occlusion e.g. wearing sunglasses or a scarf. Nonetheless, they did not directly address facial mask occlusion (before the current COVID-19 situation).

After the current COVID-19 situation, four major studies evaluated the effect of wearing a facial mask on FR performance [64, 82, 201, 202]. The National Institute of Standards and Technology (NIST) has published two specific studies on the effect of masked faces on the performance of FR solutions submitted by vendors using pre-COVID-19 [201] and post-COVID-19 [202] algorithms. These studies are part of the ongoing FR Vendor Test (FRVT). The studies by the NIST concluded that wearing a face mask has a negative effect on FR performance. However, the evaluation by NIST is conducted using synthetically generated masks, which may not fully reflect the actual effect of wearing a protective face mask on the FR performance. The recent study by Damer et al. [64] has tackled this limitation by evaluating the effect of wearing a mask on two academic FR algorithms and one commercial solution using a specific collected dataset for this purpose from 24 participants over three collaborative sessions. The study indicates the significant effect of wearing a face mask on FR performance. A similar study was carried out by the Department of Homeland Security (DHS) [82]. In this study, several FR systems (using six face acquisition systems and 10 matching algorithms) were evaluated on a specifically collected dataset of 582 individuals. The main conclusion from this study is that the accuracy of most best-performing FR systems is degraded from 100% to 96% when the subject is wearing a facial mask.

Li et al. [151] proposed to use an attention-based method to train a FR model to learn from the periocular area of masked faces. The presented method showed improvement in the masked FR performance. However, the proposed approach is only tested on simulated masked face datasets, and it essentially only maps the problem into a periocular recognition problem. A recent preprint by [13] presented a small dataset of 269 unmasked and masked face images of 53 identities crawled from the internet. The work proposed to fine-tune FaceNet model [234] using simulated masked face images to improve the recognition accuracy. However, the proposed solution is only tested using a small dataset (269 images). Recently, a rapid number of researches are published to address the detection of wearing a face mask [163, 220]. These studies did not directly address the effect of wearing a mask on FR performance or presenting a solution to improve masked FR.

Motivated by the recent evaluations efforts on the negative effect of wearing a facial mask on the FR performance [64, 82, 201, 202] and driven by the need for exclusively developing an effective solution to improve masked FR, this work presents a novel approach to improve masked FR performance. The proposed solution is designed to run on top of existing FR models. Thus, it does not require any retraining of the existing FR models as presented in next Section 4.3.

4.3. Methodology

This section presents our proposed approach to improve the verification performance of masked FR. The proposed solution is designed to operate on top of existing FR models. To achieve this goal, we propose an EUM. The input to our proposed model is a face embedding extracted from a masked face image, and the output is a so-called ”unmasked face embedding”, which is intended to be more similar to the embedding of the same identity without wearing a mask. Therefore, the proposed solution does not require any modification or training of the existing FR solution. Figure 4.1 shows an overview of the workflow of the proposed approach in training and operational modes.

Furthermore, we propose the SRT to control the model during the training phase. Similar to the well-known triplet-based learning, the SRT loss has two learning objectives: 1) Minimizing the intra-class variation, i.e., minimizing the distance between genuine pairs of unmasked and masked face embeddings. 2) Maximizing the inter-class variation, i.e., maximizing the distance between imposter pairs of masked face embeddings. However, unlike the traditional triplet loss, the proposed SRT loss function can self-adjust its learning objective by only focusing on optimizing the intra-class variation when the inter-class variation is deemed sufficient. When the gap in inter-class variation is violated, our proposed loss behaves like a conventional triple loss. The theoretical motivation behind our SRT-loss is presented along with the functional formulation later in this section. In the following, this section presents our proposed EUM architecture and the SRT loss.

4.3.1. Embedding unmasking model architecture

The EUM architecture is based on a fully connected neural network (FCNN). Having an FCNN architecture, where all neurons are connected in two consecutive layers, we can demonstrate a generalized EUM design. This is the case because this structure can be easily adapted to different input shapes, and thus can be adapted on the top of different FR models, motivating our decision to use FCNN. The model input is a masked feature embedding (i.e., resulting from a masked face image) of size D (D depends on the FR network used), and the model output is a feature vector of the same size D. The proposed model consists of four fully connected layers (FC): an input layer, two hidden layers, and an output layer. The input size for all FC layers is of size d. Each of the input and the hidden layers is followed by batch normalization (BN) [121] and Leaky ReLU non-linearity activation function [174]. The last FC layer is followed by BN.

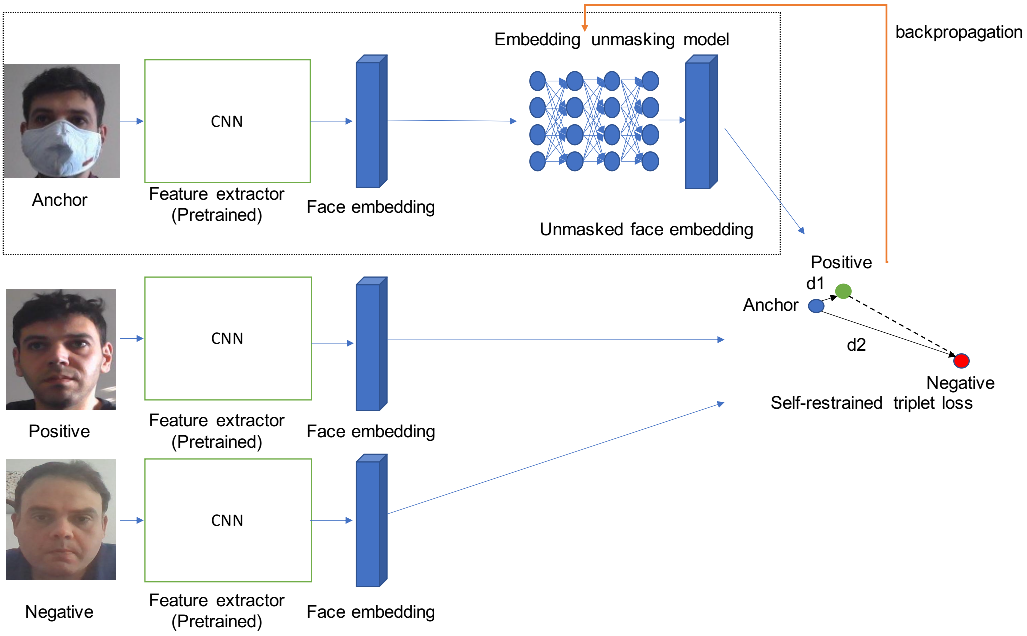
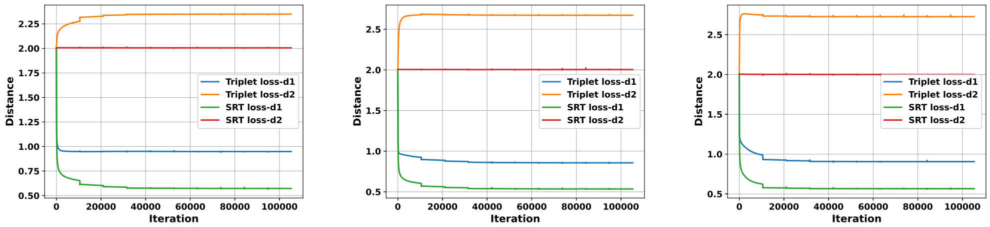

backpropagation
Embedding unmasking model
CNN
Anchor
Feature extractor (Pretrained)
Face embedding
Unmasked face embedding
Positive
d1
Anchor
d2
Negative
Self-restrained triplet loss
CNN
Positive
Feature extractor (Pretrained)
Face embedding
CNN
Negative
Feature extractor (Pretrained)
Face embedding

Figure 4.1.: EUM on top of FR model. Given a masked face embedding (anchor), the proposed SRT aims at guiding EUM to learn to output an embedding similar to the one of the same identity (positive) and dissimilar from one of the different identities (negative).

4.3.2. Unmasked face embedding learning

The learning objective of our model is to reduce the FNMR of genuine unmasked-masked pairs. The main motivation behind this learning goal is inspired by the latest reports on evaluating the effect of the masked faces on FR performance by the National Institute of Standards and Technology (NIST) [201] and the recent work by Damer et al. [64]. The NIST report [201] stated that the FNMR are increased in all evaluated algorithms when the probes are masked. For the most accurate algorithms, the FNMR increased from 0.3% to 5% at FMR of 0.001% when the probes are masked. On the other hand, the NIST report concluded that FMR appeared to be less affected when probes are masked. A similar observation comes from the study by Damer et al. [64]. This work reported that the genuine score distributions are significantly affected by masked probes [64]. The study also reported that the genuine score distribution strongly shifts towards the imposter score distributions. On the other hand, the imposter score distributions do not seem to be strongly affected by masked face probes. One of the main observations of the previous studies in [64, 201], is that wearing a face mask significantly increase the FNMR, whereas the FMR seem to be less affected by wearing a mask. Similar remarks have been also reported in our result (see Section 4.5). Based on these observations, we motivate our proposed SRT loss function to focus on increasing the similarity between genuine pairs of unmasked and masked face embeddings, while maintaining the imposter distance at an acceptable level. In the following, we briefly present the naive triplet loss followed by our proposed SRT loss.



(a) ResNet-100

(b) ResNet-50

(c) MobileFaceNet

Figure 4.2.: Naive triplet loss vs. SRT loss distance learning over training iterations. The plots show the learned d1 (distance between genuine pairs) and d2 (distance between imposter pairs) by each loss over training iterations. It can be clearly noticed that the anchor (model output) of the model trained with SRT loss is more similar to the positive than the anchor of the model trained with naive triple loss.

Self-restrained triplet loss

Previous works [234, 93] indicated that utilizing triplet-based learning is beneficial for learning discriminative face embeddings. Let $x \in X$ represents a batch of training samples, and $f(x)$ is the face embeddings obtained from the FR model. Training with triplet loss requires a triplet of samples in the form $\{x_i^a, x_i^p, x_i^n\} \in X$, where x_i^a, the anchor, and x_i^p, the positive, are two different samples of the same identity, and x_i^n, the negative, is a sample belonging to a different identity. The learning objective of the triplet loss is that the distance between $f(x_i^a)$ and $f(x_i^p)$ (genuine pairs) with the addition of a fixed margin value (m) is smaller than the distance between $f(x_i^a)$ and any face embedding $f(x_i^p)$ of any other identities (imposter pairs). In FaceNet [234], triplet loss is proposed to learn face embeddings by applying the Euclidean distance to normalized face embeddings.

Formally, the triplet loss ℓ_t for a mini-batch of N samples is defined as follow:

$$\ell_t = \frac{1}{N}\sum_{i}^{N} \max\{d(f(x_i^a), f(x_i^p)) - d(f(x_i^a), f(x_i^n)) + \mathrm{m}, 0\}, \tag{4.1}$$

where m is a margin applied to impose the separability between genuine and imposter pairs. An d is the euclidean distance applied on normalized features and it is given by:

$$d(x_i, y_i) = \|\mathbf{x}_i - \mathbf{y}_i\|_2^2 \,. \tag{4.2}$$

Figure 4.3 visualize two triplet loss learning scenarios. Figure 4.3.a shows the initial training triplet, and Figure 4.3.b and 4.3.c illustrate two scenarios that can be learnt using triplet loss. In both scenarios, the goal of the triplet loss is achieved i.e. $d(f(x_i^a), f(x_i^n)) > d(f(x_i^a), f(x_i^p)) + m$. In Figure 4.3.b (scenario 1), both distances are optimized. However, in this scenario, the optimization of d2 distance is greater than the optimization of d1 distance. Whereas, in Figure 4.3.c (scenario 2), the triplet loss enforces the model to focus on minimizing the distance between the anchor and the positive. The optimal state for the triplet loss is achieved when both distance are fully optimized i.e. $d(f(x_i^a), f(x_i^p))$ is equal to zero and $d(f(x_i^a), f(x_i^n))$ is greater than the predefined margin. However, achieving such a state may not be feasible, and it requires a huge number of training triplets with large computational resources for selecting the optimal triplets for training. Given a masked face embedding, our model is trained to generate a new embedding such as it is similar to the unmasked face embedding of the same identity and dissimilar from other face embeddings of any other identities. As discussed earlier in this section, the distance between imposter pairs is less affected by wearing a mask [64, 201]. Thus, we aim to ensure that our proposed loss focuses on minimizing the distance between the genuine pairs (similar to scenario 2) while maintaining the distance between imposter pairs.

Training EUM with SRT loss requires a triple to be defined as follows: $f(x_i^a)$ is an anchor of masked face embedding, $EUM(f(x_i^a))$ is the anchor given as an output of the EUM, $f(x_i^p)$ is a positive of unmasked embedding, and $f(x_i^n)$ is a negative embedding of a different identity than anchor and positive. This triplet is illustrated in Figure 4.1. We want to ensure that the distance (d1) between $EUM(f(x_i^a))$ and $f(x_i^p)$ in addition to a predefined margin is smaller than the distance (d2) between $EUM(f(x_i^a))$ and $f(x_i^n)$. Our goal is to train EUM to focus on minimizing d1, as d2 is less affected by the mask.

Under the assumption that the distance between the positive and the negative embeddings (d3) is close to optimal and it does not contribute to the back-propagation of the EUM model, we propose to use this distance as a reference to control the triplet loss. The main idea is to train the model as a naive triplet loss when d2 (anchor-negative distance) is smaller than d3 (positive-negative distance). In this case, the SRT guides the model to maximize d2 distance and to minimize d1 distance. When d2 is equal or greater than d3, we replace d2 with d3 in the loss calculation. This distance swapping allows the SRT to learn only, at this point, to minimize d1 distance. At any point of the training, when the condition on d2 is violated i.e $d(d2) < d(d3)$, the SRT behave again as a naive triplet loss. We opt to compare the d2 and d3 distances on the batch level to avoid swapping the distance on every minor update on the distance between the imposter pairs. In this case, we want to ensure that the d1 distance, with the addition of a margin m, is smaller than the mean of the d3 distances calculated on the mini-batch of triplets. Thus, our loss is less sensitive to the outliers resulting from comparing imposter pairs. We define our SRT loss for a mini-batch of the size N as follow:

$$\ell_{SRT} = \begin{cases} \frac{1}{N}\sum_i^N \max\{d(f(x_i^a), f(x_i^p)) - d(f(x_i^a), f(x_i^n)) + \mathrm{m}, 0\} & \text{if } \mu(d2) < \mu(d3) \\ \frac{1}{N}\sum_i^N \max\{d(f(x_i^a), f(x_i^p)) - \mu(d3) + \mathrm{m}, 0\} & \text{otherwise,} \end{cases} \tag{4.3}$$

where $\mu(d2)$ is the mean of the distances between the anchor and the negative pairs calculated on the mini-batch level, given as $\frac{1}{N}\sum_i^N(d(f(x_i^a), f(x_i^n)))$. $\mu(d3)$ is the mean of the distances between the positive and the negative pairs calculated on the mini-batch level, given as $\frac{1}{N}\sum_i^N(d(f(x_i^p), f(x_i^n)))$. An d is the euclidean distance computed on normalized feature embedding (Equation 4.2).

Figure 4.2 illustrates the optimization of d1 (distance between genuine pairs) and d2 (distance between imposter pairs) by naive triplet loss and SRT loss over the training iterations of three EUM models on top of ResNet-100 (Figure 4.2a), ResNet-50 (Figure 4.2b) and MobileFaceNet (Figure 4.2c). Details on the training settings are presented in Section 4.4. It can be clearly noticed that the d1 distance (anchor-positive distance) learned by SRT is significantly smaller than the one learned by naive triplet loss. This indicates that the output embedding of the EUM trained with SRT is more similar to the embedding of the same identity (the positive) than the output embedding of EUM trained with triplet loss.

4.4. Experimental setup

This section presents the experimental setups and the implementation details applied in the work.

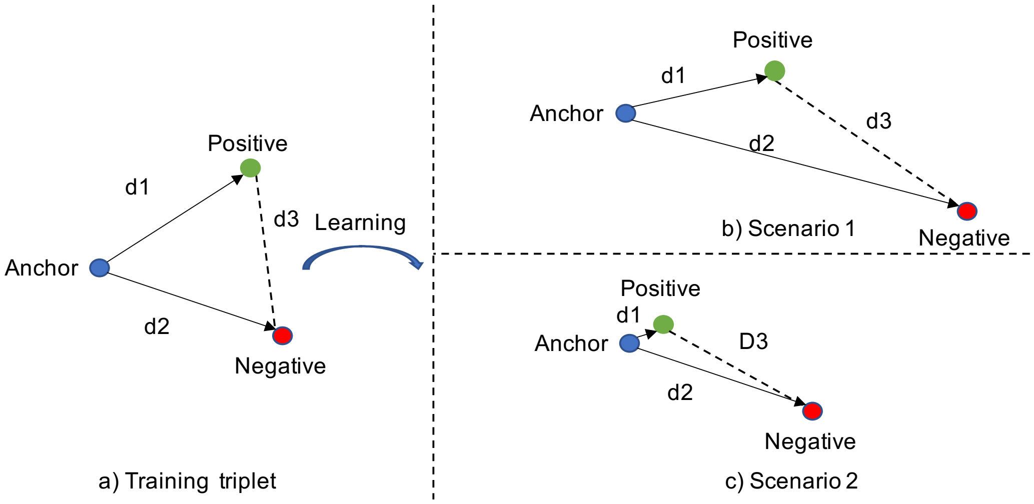
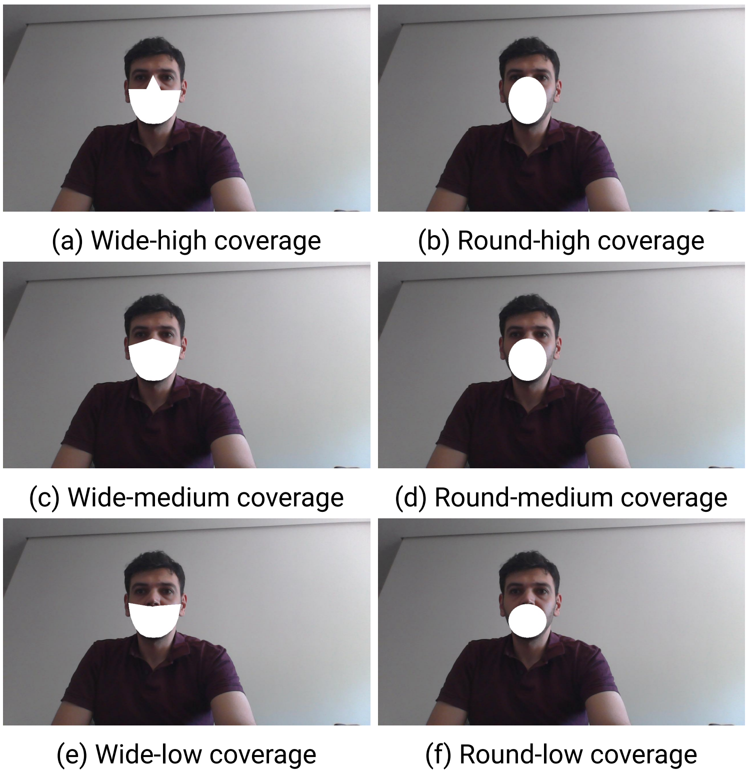


Figure 4.3.: Triplet loss guides the model to minimize the distance d1 between the anchor and positive (genuine pair) to be smaller than the distance d2 between the anchor and positive (imposter pair). In Scenario 1 (Figure a) and Scenario 2 (Figure b), the learning goal of triple loss is achieved, where the d1 is smaller than d2 in both scenarios. One can be clearly noticed that d1 is larger in scenario 1 than scenario 2. The proposed SRT aims at achieving a close learning objective to scenario 2, i.e., focusing on optimizing d1 when d2 is sufficient.

4.4.1. Face recognition model

To provide a deep evaluation of the performance of the proposed solution, we evaluated our proposed solution on top of three FR models - ResNet-100 [107], ResNet-50 [107] and MobileFaceNet [49]. ResNet is one of the widely used Convolutional Neural Network (CNN) architecture used by several FR models, e.g. ArcFace [80] and VGGFace2 [42].

MobileFaceNet is a compact model designed for low computational powered devices. MobileFaceNet model architecture is based on residual bottlenecks proposed by MobileNetV2 [233] and depth-wise separable convolutions layer, which allows building a CNN model with a much smaller set of parameters in comparison to standard CNNs. To provide fair and comparable evaluation results, ResNet-50 and MobileFaceNet are trained using the same loss function, the Arcface loss [80], and the same training dataset, MS1MV2 [80]. The MS1MV2 is a refined version of the MS-Celeb-1M [103] dataset. For ResNet-100, we use the pretrained model released by [80]. ResNet-100 is trained with ArcFace loss on MS1MV2 [80]. Our choice is to employ Arcface loss as it achieved state-of-the-art performance of several FR testing datasets such as Labeled Face in the Wild (LFW) [114]. The achieved accuracy on LFW by ResNet-100, ResNet-50 and MobileFaceNet trained with Arcface loss using MS1MV2 dataset are 99.83%, 99.80%, and 99.55%, respectively. The considered FR models are evaluated with cosine-distance for comparison. The Multi-task Cascaded Convolutional Networks (MTCNN) solution [289] is employed to detect and align the input face image. All models process aligned and cropped face image of size 112×112 pixels to produce $512 - D$ embedding feature by ResNet-100 and ResNet-50 and $128 - D$ embedding feature by MobileFaceNet.



(a) Wide-high coverage

(b) Round-high coverage

(c) Wide-medium coverage

(d) Round-medium coverage

(e) Wide-low coverage

(f) Round-low coverage

Figure 4.4.: Samples of the synthetically generated face masks of different shape and coverage.

4.4.2. Synthetic mask generation

As there is no large-scale dataset with pairs of unmasked and masked face images, we opted to use a synthetically generated mask to train our proposed approach. Specifically, we use the synthetic mask generation method proposed by NIST [201]. The synthetic mask generation method depends on the Dlib [140] Toolkit for detecting and extracting 68 facial landmarks from a face image. Based on the extracted landmark points, a face mask of different shapes, heights and colors can be drawn on the face images. The detailed implementation of the synthetic mask generation method is described in [201]. The synthetic mask generation method provided six mask types with different heights and coverage: A) wide-high coverage, B) round-high coverage, C) wide-medium coverage, D) round-medium coverage, E) wide-low coverage, and F) round-low coverage. Figure 4.4 shows examples of the simulated face mask with different types and coverage levels. To synthetically generate a masked face image, we first extract the facial landmark points of the input face image. Then, a mask with a specific color and type can be drawn on the face image using the x, y coordinates of the facial landmarks points.

4.4.3. Dataset

MS1MV2 dataset [80] is used to train our proposed approach. The MS1MV2 is a refined version of MS-Celeb-1M [103] dataset. The MS1MV2 contains $58m$ images of $85k$ different identities. We generated a masked version of the MS1MV2 noted as MS1MV2-Masked as described in Section 4.4.2. The mask type (as described in Section 4.4.2) and color are randomly selected for each image to add more diversity of mask color and coverage to the training dataset. The Dlib failed in extracting the facial landmarks from $426k$ images. These images are neglected from the training dataset. A subset of $5k$ images are randomly selected from MS1MV2-Masked to validate the model during the training phase.

The proposed solution is evaluated using two real masked face datasets: Masked Faces in Real World for Face Recognition (MRF2) [13] and Masked face recognition (MFR) [64, 62]. We also evaluated our solution on two larger-scale datasets with synthetically generated masks. We use the synthetic mask generation method described in Section 4.4.2 (proposed by NIST [201]) to synthetically generate masked faces from the Labeled Faces in the Wild (LFW) [114] and IARPA Janus Benchmark -C (IJB-C) [183]. The mask type and color are randomly selected for each image in the LFW and IJB-C datasets to achieve a greater variety of mask types and colors [1]. The evaluation datasets were described

[1] The SRT implementation, training and evaluation codes, pretrained models and the list of mask types and colors applied on IJB-C and LFW are publicly released for reproducibility of the result `https://github.com/fdbtrs/Self-restrained-Triplet-Loss`

in Chapter 2. In the following, we briefly describe each of the evaluation datasets for convenience of the reading and to clarify the base for the presented evaluation setting in Section 4.4.4.

Masked Faces in Real World for Face Recognition (MRF2) MFR2 [13] contains 269 images of 53 identities crawled from the internet. Therefore, the images of the MRF2 dataset can be considered as captured under in-the-wild conditions. The dataset contains images of real masked and unmasked faces with an average of 5 images per identity.

Masked Face Recognition (MFR) We deploy an extended version of the MFR dataset [64, 62]. The extended version of MFR is collected from 48 participants using their webcams under three different sessions- session 1 (reference) and session 2 and 3 (probes). The sessions are captured on three different days. Each session contains data captured using three videos. In each session, the first video is recorded when the subject is not wearing a facial mask in the daylight without additional electric lighting. The second and third videos are recorded when the subject is wearing a facial mask and with no additional electric lighting in the second video and with electric lighting in the third video (room light is turned on). The first session (reference) contains 480 unmasked images and 960 masked images. The second and the third sessions (probe) contain 960 unmasked images and 1920 masked images.

Labeled Faces in the Wild (LFW) LFW [114] is an unconstrained face verification benchmark. It contains 13,233 images of 5,749 identities. The number of comparison pairs in unrestricted with labeled outside data protocol [114] of LFW is 6000 (3000 genuine and 3000 imposter comparisons).

IARPA Janus Benchmark–C (IJB-C) IJB-C dataset [183] is one of the largest face verification benchmark. IJB-C consists of 31,334 still images and 117,542 frames of 11,779 videos of 3531 identities. The 1:1 mixed verification protocol [183] of IJB-C contains 19,557 genuine and 15,638,932 impostor comparisons.

4.4.4. Evaluation settings

The verification performances are reported for each of the evaluation datasets under seven experimental settings. Also, for each of the conducted experiments, we report the failure to extract rate (FTX) to capture the effect of wearing a face mask on face detection. FTX measure is the proportion of comparisons where the feature extraction was not possible.

For IJB-C and LFW, we used the bounding box provided by the datasets to align and crop the face. Therefore, the FTX for LFW and IJB-C are 0.0% in all experimental settings. For MFR and MRF2 datasets, we reported the FTX for each of the experiment settings. The conducted experiments are defined as follow:

Unmasked reference-unmasked probe (UMR-UMP) The unmasked references are compared to unmasked probes. For LFW and IJB-C, we followed the evaluation protocol given by each of these datasets and evaluated them based on the provided comparison pairs. The number of genuine comparisons is 3000 in LFW and 19,557 in IJB-C. The number of imposter comparisons is 3000 in LFW and 15,638,932 in IJB-C. The evaluation of UMR-UMP on the MFR2 dataset is done by performing N:N comparisons between all unmasked faces resulting in 90 genuine and 9,416 imposter comparisons. For the MFR dataset, we performed N:M comparisons between the unmasked reference of the first session (reference session) and unmasked probe of the second and the third sessions (probe sessions) resulting in 9,600 genuine and 451,200 imposter comparisons. The FTXs of MFR and MRF2 when the probes and the references are unmasked are 0.0%.

Unmasked reference-masked probe (UMR-MP) The unmasked references, in this case, are compared to masked probes. For LFW and IJB-C datasets, we utilized the exact comparison pairs defined in UMR-UMP experimental settings. Different from UMR-UMP, the probes, in this case, are synthetically masked (as described in section 4.4.2). We considered the first image in defined pairs as a reference and the second image is considered as a probe. For the MRF2 dataset, we performed N:M comparisons between unmasked and masked sets resulting in 296 genuine and 15090 imposter comparisons. The FTX of MRF2, in this setting, is 0.9497%. For the MFR dataset, we performed N:M comparisons between unmasked references of the first session and masked probes of the second and the third sessions. The FXT, in this case, is 4.4237%, and the number of comparisons is 16,490 genuine and 86,4341 imposter comparisons.

Unmasked reference-masked probe (UMR-MP(T)) The unmasked references are compared to masked probes. Different from UMR-MP, the masked probes are processed by EUM model trained with conventional triplet loss (T).

Unmasked reference-masked probe (UMR-MP(SRT)) The unmasked references are compared to masked probes processed by EUM model trained with SRT loss. In UMR-MP(T) and UMR-MP(SRT), the number of genuine and impostor pairs and the FTXs are identical to UMR-MP experimental setting.

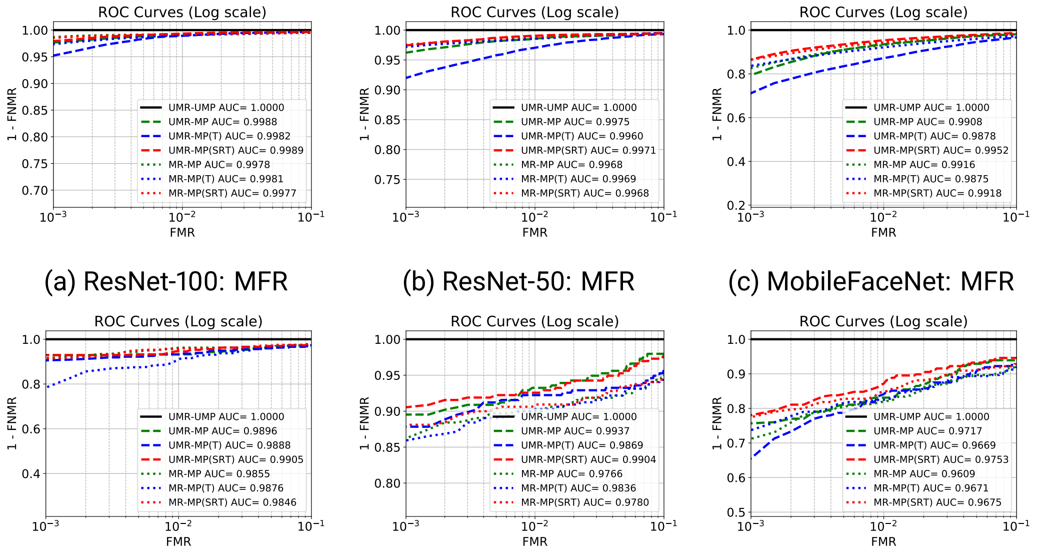
Masked reference-masked probe (MR-MP) The masked references are compared to masked probes. For LFW and IJB-C, we utilized the same comparison pairs described in UMR-UMP experimental setting. Both reference and probe are synthetically masked. The number of genuine and imposter comparisons, in this case, is the same as in UMR-UMP experimental setting. For the MRF2 dataset, we performed N:N comparisons between masked faces resulting in 639 genuine and 24,010 imposter comparisons. The FTX, in this case, is 1.2030% for the MRF2 dataset. For the MFR dataset, we performed N:M comparisons between the masked faces of the first session and the masked faces of the second and the third sessions resulting in 31,318 genuine and 1,729,424 imposter comparisons. The FTX, in this case, is 4.4736% for the MFR dataset.

Masked reference-masked probe (MR-MP(T)) The masked references are compared to masked probes. Both masked references and probes are processed by EUM trained with conventional triplet loss (T). The comparison pairs and the FTX are the same as in the MR-MP experimental setting.

Masked reference-masked probe (MR-MP(SRT)) The masked references are compared to masked probes. Masked references and probes are processed by EUM trained with SRT loss. The comparison pairs and the FTX for all evaluation datasets, in this experimental case, are identical to the MR-MP case.

4.4.5. Model training setup

We trained six instances of the EUM model. The first, second, and the third instances, ResNet-100 EUM(SRT), ResNet-50 EUM(SRT) and MobileFaceNet EUM(SRT), are trained with SRT loss using feature embeddings obtained from ResNet-100, ResNet-50 and MobileFaceNet, respectively. The fourth, fifth and sixth instances, ResNet-100 EUM(T), ResNet-50 EUM(T) and MobileFaceNet EUM(T), are trained with triplet loss using feature embeddings obtained from ResNet-100, ResNet-50 and MobileFaceNet, respectively as an ablation study to our proposed SRT. The proposed EUM models are implemented by Pytorch and trained on Nvidia GeForce RTX 2080 GPU. All models are trained using an SGD optimizer with an initial learning rate of 1e-1 and batch size of 512. The learning rate is divided by 10 at $30k, 60k, 90k$ training iterations. The early-stopping patience parameter is set to 3 (around 30k training iteration) causing ResNet-100 EUM(SRT), ResNet-50 EUM(SRT), MobileFaceNet EUM(SRT), ResNet-100 EUM(T), ResNet-50 EUM(T) and MobileFaceNet EUM(T) to stop after 10k, 80k, 70k, 10k, 60k, 10k training iterations, respectively.



(a) ResNet-100: MFR

(b) ResNet-50: MFR

(c) MobileFaceNet: MFR

(d) ResNet-100: MRF2

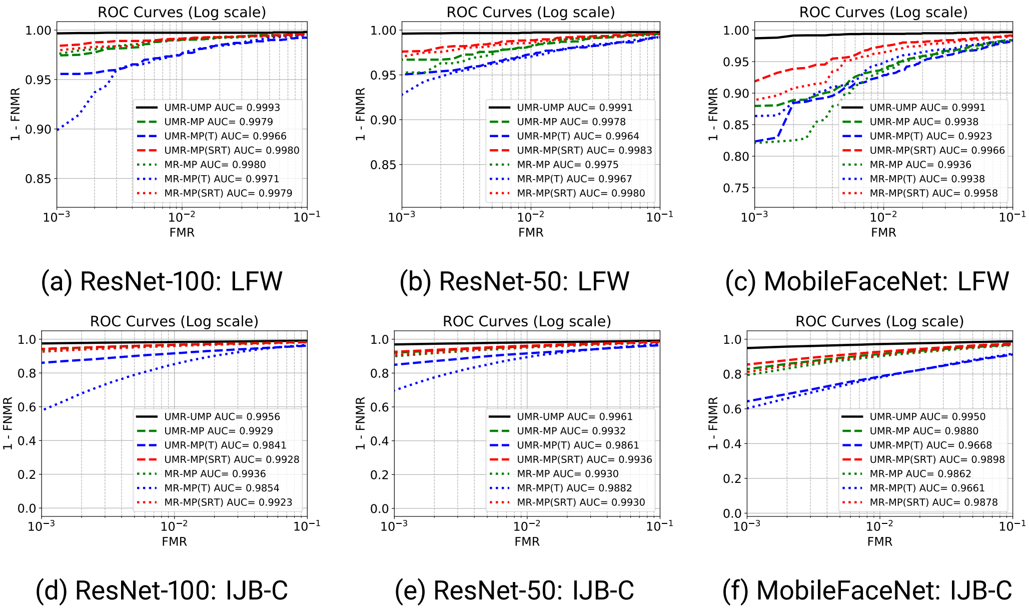
(e) ResNet-50: MRF2

(f) MobileFaceNet: MRF2

Figure 4.5.: The achieved log-scale ROC curves by different experimental settings using MFR and MRF2 datasets. The ROC curves achieved by EUM trained with SRT in all plots are in red. The ROC curves achieved by EUM trained with the naive triplet in all plots are in blue. The ROC curves of the considered models without EUM are in green color. In each plot, the curves of UMR-MP, UMR-MP(T), and UMR-MP(SRT) cases are marked with a dashed line. The curves of MR-MP, MR-MP(T), and MR-MP(SRT) cases are marked with a dotted line. For each ROC curve, the area under the curve (AUC) is listed inside the plot.

4.4.6. Evaluation metric

The verification performance is reported as Equal Error Rate (EER), as well as, FMR100, and FMR1000, which are the lowest FNMR for a FMR$\leq$1.0% and $\leq$0.1%, respectively. Additionally, we calculate and report the operation thresholds at FMR100 (FMR100_Th) and FMR1000 (FMR1000_Th) for each of the evaluated models and each of the benchmarks based on UMR-UMP experimental setting (unmasked reference - unmasked probe). Based on FMR100_Th and FMR1000_Th thresholds, we report the FMR, the FNMR, and the average of the FMR and FNMR (Avg) at these thresholds for all experimental settings. This aims to estimate a realistic scenario where the operational threshold is decided on the conventional UMR-UMP performance. We also report the mean of the genuine scores



(a) ResNet-100: LFW

(b) ResNet-50: LFW

(c) MobileFaceNet: LFW

(d) ResNet-100: IJB-C

(e) ResNet-50: IJB-C

(f) MobileFaceNet: IJB-C

Figure 4.6.: The achieved log-scale ROC curves by different experimental settings using LFW and IJB-C datasets. The ROC curves achieved by EUM trained with SRT in all plots are in red. The ROC curves achieved by EUM trained with the naive triplet in all plots are in blue. The ROC curves of the considered models without EUM are in green color. In each plot, the curves of UMR-MP, UMR-MP(T), and UMR-MP(SRT) cases are marked with a dashed line. The curves of MR-MP, MR-MP(T), and MR-MP(SRT) cases are marked with a dotted line. For each ROC curve, the area under the curve (AUC) is listed inside the plot.

(G-mean) and the mean of imposter scores (I-mean) to analysis the shifts in genuine and imposter scores distributions induced by wearing a face mask and to demonstrate the improvement in the verification performance achieved by our proposed solution. For each of the evaluation settings, we plot the ROC, showing FMR100 and FMR1000 clearly by providing a log-scale FMR axis. Further, we enrich our reported evaluation results by reporting the Fisher Discriminant Ratio (FDR) to provide an in-depth analysis of the separability of genuine and imposters scores for different experimental settings. FDR is a class separability criterion described in [214], and it is given by:

$$FDR = \frac{(\mu_G - \mu_I)^2}{(\sigma_G)^2 + (\sigma_I)^2}, \tag{4.4}$$

where μ_G and μ_I are the genuine and imposter scores mean values and σ_G and σ_I are their standard deviations values. The larger the FDR value, the higher is the separation between the genuine and imposters scores and thus better expected verification performance.

MFR	Setting	EER%	FMR100%	FMR1000%	FMR100_Th$^{\text{UMR-UMP}}$			FMR1000_Th$^{\text{UMR-UMP}}$			G-mean	I-mean	FDR
					FMR%	FNMR%	Avg.%	FMR%	FNMR%	Avg.%			
ResNet-100	UMR-UMP	0.0000	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.8534	0.0252	70.7159
	UMR-MP	0.8914	0.8793	2.3347	0.4829	1.1886	0.8358	0.0082	6.0461	3.0272	0.5271	0.0203	15.0316
	UMR-MP(T)	1.0430	1.0794	4.7726	0.3084	2.4257	1.3670	0.0000	17.4773	8.7386	0.4331	0.0188	12.0587
	UMR-MP(SRT)	**0.7702**	0.6610	2.0558	0.4717	0.9460	**0.7089**	0.0108	4.8029	**2.4068**	0.5379	0.0221	15.9027
	MR-MP	**0.8014**	0.7695	1.3155	4.3230	0.5971	2.4601	0.4031	0.8685	**0.6358**	0.7314	0.0560	18.7469
	MR-MP(T)	0.9598	0.9471	2.6348	16.0855	0.4513	8.2684	2.4656	0.7660	1.6158	0.7415	0.1185	15.2544
	MR-MP(SRT)	0.8270	0.8015	1.4433	3.6616	0.6482	**2.1549**	0.3083	0.9994	0.6539	0.7248	0.0486	18.3184
ResNet-50	UMR-UMP	0.0000	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.8538	0.0349	55.9594
	UMR-MP	1.2492	1.4251	3.7780	0.4308	1.9709	1.2008	0.0007	10.6246	5.3126	0.5254	0.0251	12.6189
	UMR-MP(T)	1.9789	2.9533	7.9988	0.5626	4.0206	2.2916	0.0000	30.6549	15.3275	0.4401	0.0392	9.4412
	UMR-MP(SRT)	**0.9611**	0.9460	2.5652	0.5595	1.2129	**0.8862**	0.0030	7.4591	**3.7310**	0.5447	0.0272	13.4045
	MR-MP	1.2963	1.4145	2.6311	3.7683	0.8302	2.2993	0.2222	2.0467	1.1345	0.7232	0.0675	15.1356
	MR-MP(T)	1.3091	1.4560	2.8259	96.3681	0.0000	48.1840	62.1757	0.1980	31.1868	0.8269	0.4169	13.0528
	MR-MP(SRT)	**1.1207**	1.1367	2.4523	3.2837	0.8717	**2.0777**	0.2227	1.8775	**1.0501**	0.7189	0.0557	15.1666
MobileFaceNet	UMR-UMP	0.0000	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.8432	0.0488	37.3820
	UMR-MP	3.4939	6.5070	20.5640	0.2723	12.3833	6.3278	0.0088	40.4063	20.2075	0.4680	0.0307	7.1499
	UMR-MP(T)	5.2759	12.7835	28.8175	0.2151	21.7829	10.9990	0.0149	66.7192	33.3671	0.3991	0.0501	5.9623
	UMR-MP(SRT)	**2.8805**	4.6331	13.4384	0.3746	7.3802	**3.8774**	0.0097	30.1516	**15.0807**	0.5013	0.0383	8.6322
	MR-MP	3.5060	6.8842	17.3479	4.6039	2.8674	3.7357	0.5465	8.6723	**4.6094**	0.6769	0.1097	7.9614
	MR-MP(T)	4.2947	7.9124	16.3772	94.0982	0.0064	47.0523	61.3860	0.6354	31.0107	0.8082	0.4716	6.6455
	MR-MP(SRT)	**3.1866**	5.6166	13.5290	3.1906	3.1867	**3.1886**	0.2658	9.4802	4.8730	0.6636	0.0837	8.0905

Table 4.1.: The achieved verification performance of different experimental settings by ResNet-100, ResNet-50 and MobileFaceNet models along with EUM trained with triplet loss and EUM trained with SRT loss. The result is reported on MFR dataset. The FMR100_Th$^{\text{UMR-UMP}}$ are equal to 0.2307, 0.2652 and 0.3246 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The FMR1000_Th$^{\text{UMR-UMP}}$ are equal to 0.3482, 0.3926 and 0.4476 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The lowest EER and the lowest average error of FMR100 and FMR1000 at the defined threshold for each of the evaluation cases are marked in bold. One can notice the significant improvement in the verification performance induced by our proposed approach (SRT) in most evaluation cases.

4.5. Result

In this section, we present and discuss our achieved results. First of all, we experimentally present the negative impact of wearing a face mask on FR performance. Then, we present and discuss the impact of our EUM trained with SRT on enhancing the separability between the genuine and imposter comparison scores. Then, we present the gain in the masked face verification performance achieved by our proposed EUM trained with SRT on the collaborative and in-the-wild masked FR. Finally, we present an ablation study on SRT to experimentally support our theoretical motivation behind the SRT loss by comparing its performance with the triplet loss.

4.5.1. Impact of masked face on the FR verification performance

Tables 4.1, 4.2, 4.3, and 4.5 present a comparison between the baseline evaluation where reference and probe are unmasked (UMR-UMP), the case where only the probe is masked (UMR-MP), and the case where reference and probe are masked (MR-MP). On UMR-UMP case, the considered FR models, ResNet-100, ResNet-50, and MobileFaceNet, achieve a very high verification performance. This is demonstrated by scoring 0.0%, 0.0% and 0.0% EER on the MFR dataset (Table 4.1), 0.0%, 0.0% and 0.0106% on the MRF2 dataset (Table 4.2), 0.2660%, 0.3333% and 0.6333% EER on the LFW (Table 4.3) and 1.5340%, 1.6881% and 2.2396% EER on IJB-C dataset (Table 4.5), respectively, by model ResNet-100, ResNet-50 and MobileFaceNet.

The verification performances of the considered models are substantially degraded when evaluated on real and synthetically generated masked face images. This is indicated by the degradation in verification performance measures and FDR values, in comparison to the case where probe and reference are unmasked. MobileFaceNet achieved lower verification performance on MR-MP than UMR-MP evaluation setting, as seen in Tables 4.1, 4.2, 4.3, and 4.5. Furthermore, ResNet-50 achieved lower verification performance in the MR-MP than the UMR-MP setting when it is evaluated on MFR, MRF2, and IJB-C datasets. For example, on the MFR dataset, the achieved EER by ResNet-50 model is 1.2492% (UMR-MP). This error rate is raised to 1.2963% for the MR-MP evaluation setting, as seen in Table 4.1. On LFW, the ResNet-50 model achieved very close performance for the MR-MP and the UMR-MP evaluation setting. In this case, the achieved EER by ResNet-50 are 1.4667% for the UMR-MP evaluation setting and 1.3667% for the MR-MP evaluation setting. Furthermore, ResNet-100 achieved lower verification performance for the MR-MP evaluation setting than the UMR-MP evaluation setting when it is evaluated on MRF2 and IJB-C datasets. On LFW and MFR, the ResNet-100 model achieved very close performance for the MR-MP and the UMR-MP evaluation settings. When the FMR and FNMR measures are calculated based on FMR100_Th$^{\text{UMR-UMP}}$, the achieved FMR and FNMR are higher on the MR-MP than the UMR-MP case in most of the settings. When the threshold is set to FMR1000_Th$^{\text{UMR-UMP}}$, the achieved FMR and FNMR are lower when both reference and probe are masked (MR-MP) than in the case where only probes are masked (UMR-MP) in most of the evaluation settings. Also, one can be noticed that wearing a face mask (UMR-MP and MR-MP cases) has a higher effect on the FNMR than FMR when these measures are calculated based on FMR100_Th$^{\text{UMR-UMP}}$ or FMR1000_Th$^{\text{UMR-UMP}}$. These results are also supported by the G-mean, I-mean, and FDR shown in Tables 4.1, 4.2, 4.3 and 4.5.

We also make four general observations: 1) The compact model, MobileFaceNet, achieved lower verification performance than the ResNet-100 and ResNet-50 model. One of the reasons for this performance degradation might be due to the smaller embedding size of MobileFaceNet (128-D), in comparison to the embedding size of 512-D in ResNet-100 and ResNet-50. Moreover, the size of the MobileFaceNet network (1m parameters) is extremely smaller than ResNet-100 (65m parameters) and ResNet-50 (36m parameters), which might affect the generalization ability of the MobileFaceNet model. 2) The considered models achieved lower performance when evaluated on the MRF2 dataset than the case when evaluated on the MFR dataset. This result was expected as the images in the MRF2 dataset are crawled from the internet with high variations in facial expression, pose, illumination. On the other hand, the images in the MFR dataset are collected in a collaborative environment. 3) The considered models achieved lower performance on LFW and IJB-C datasets in comparison to MFR and MRF2 as they are larger scale. The considered models achieved lower performance when evaluated on IJB-C than the case when evaluated on LFW. This result was expected as the evaluation protocol of LFW is simpler than the IJB-C, and the IJB-C has shown to be more challenging than LFW in multiple studies [80, 42]. 4) The considered models achieved relatively higher G-mean scores on the UMR-MP than the MR-MP experimental setting. This indicates a higher similarity between genuine pairs in the MR-MP than the UMR-MP. However, the achieved verification performances by UMR-MP cases on most of the evaluated datasets are higher than the achieved ones by MR-MP. One of the contributing factors for the difference in the performance is that the imposter distribution is shifted more toward genuine distribution in the MR-MP cases than the UMR-MP ones, i.e. masked face pairs are more similar (in comparison to unmasked-masked pairs) even if the identities are different. This statement can be clearly observed from the achieved I-mean values shown in Tables 4.1, 4.2, 4.3, and 4.5. This shifting in imposter distribution strongly affects the verification performance of the considered models.

To summarize, wearing a face mask has a negative effect on FR performance. This observation is experimentally proved by evaluating the verification performance of three FR models, ResNet-100, ResNet-50, and MobileFaceNet, on two real masked datasets (MFR and MRF2) and two synthetically generated masked face datasets (LFW and IJB-C). This result supports and complements the previous findings in the studies in [64, 82, 201, 202] evaluating the impact of wearing a mask on FR performance.

4.5.2. Impact of EUM with SRT on the separability between genuine and imposter comparison scores

The proposed approach significantly enhanced the separability between the genuine and imposter comparison scores in the considered FR models and both evaluated datasets. This improvement can be seen in the increase in the FDR separability measure achieved by our proposed EUM trained withSRT in comparison to the achieved FDR measures by the considered FR models, as shown in Table4.1, 4.2, 4.3, and 4.5. This indicates a general improvement in the verification performance of FR and thus enhancing the general trust in the verification decision. For example, when the ResNet-50 model is evaluated on the MFR dataset and the probe is masked, the FDR increases from 12.6189 (UMR-MP) to 13.4045 (UMR-MP(SRT)) using our proposed approach, as shown in Table 4.1. Similar observations can be made when the evaluation dataset is synthetically masked. For example, when ResNet-100 is evaluated on the synthetically generated masked face of IJB-C, the FDR increases from 9.7516 (UMR-MP) to 9.9005 (UMR-MP(SRT)) using our proposed approach, as shown in Table 4.5. This improvement in the separability between the genuine and the imposter samples by our proposed approach is achieved in most of the evaluation settings, where the FDR increased in 20 out of 24 experimental settings.

4.5.3. Impact of EUM with SRT solution on the collaborative masked FR

When the considered models are evaluated on the MFR dataset, it can be observed that our proposed approach significantly enhanced the masked face verification performance, as shown in Table 4.1. The achieved EER by ResNet-100 is 0.8912% on the UMR-MP case. This error is reduced to 0.7702% using our approach (UMR-MP(SRT)). The achieved EER by the ResNet-100 is 0.8014% on MR-MP experimental setting. The achieved EER using our approach on top of the ResNet-100 is 0.8270% (MR-MP(SRT)). However, this is the only case that we did not observe improvement in EER when the considered models are evaluated on the MFR dataset. The achieved EER by ResNet-50 model is 1.2492% based on UMR-MP experimental setting. This error rate is decreased to 0.9611% by our proposed approach (UMR-MP(SRT)) indicating a clear improvement in the verification performance induced by our proposed approach, as shown in Table 4.1. A similar enhancement in the verification performance is observed by our approach for the MR-MP setting. In this case, the EER is decreased from 1.2963% (MR-MP) to 1.1207% (MR-MP(SRT)). The achieved EER by the MobileFaceNet model is 3.4939% (UMR-MP). This error is reduced to 2.8805% using our proposed approach (UMR-MP(SRT)). Considering the MR-MP setting, the EER is decreased from 3.506% (MR-MP) to 3.1866% (MR-MP(SRT)) by our approach. The improvement in the masked FR verification performance is also noticeable from the improvement in FMR100 and FMR1000 measures. When the considered models are evaluated on masked data (UMR-MP and MR-MP) based on FMR100_Th$^{\text{UMR-MP}}$, the average of FMR and FNMR was significantly reduced by our proposed approach in all evaluation settings (UMR-MP(SRT) and MR-MP(SRT)), as shown in Table 4.1. When the operation threshold is calculated at FMR1000 (FMR1000_Th$^{\text{UMR-MP}}$), a significant reduction in the average of FMR and FNMR with our proposed approach is notable in most evaluation settings. This result is also supported by ROC curves shown in Figures 4.5a, 4.5b and 4.5c.

MRF2	Setting	EER%	FMR100%	FMR1000%	FMR100_Th$^{\text{UMR-UMP}}$			FMR1000_Th$^{\text{UMR-UMP}}$			G-mean	I-mean	FDR
					FMR%	FNMR%	Avg.%	FMR%	FNMR%	Avg.%			
ResNet-100	UMR-UMP	0.0000	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.7605	0.0019	46.4218
	UMR-MP	4.0515	6.7568	7.0946	0.9079	6.7568	3.8323	0.1127	7.0946	**3.6036**	0.4454	-0.0000	9.3458
	UMR-MP(T)	4.0515	6.7568	9.4595	0.7820	6.7568	3.7694	0.0530	11.1486	5.6008	0.3677	-0.0012	8.3377
	UMR-MP(SRT)	**3.3757**	5.4054	7.0946	0.9145	5.7432	**3.3289**	0.1127	7.0946	**3.6036**	0.4587	-0.0003	9.8264
	MR-MP	3.7522	3.7559	8.4507	4.3648	3.4429	3.9039	1.0079	3.7559	**2.3819**	0.6757	0.0183	6.4714
	MR-MP(T)	4.3817	9.0767	21.5962	20.6247	2.5039	11.5643	9.3461	3.1299	6.2380	0.6947	0.0834	5.8089
	MR-MP(SRT)	**3.4416**	4.3818	8.4507	3.8651	3.1299	**3.4975**	0.8247	4.3818	2.6033	0.6738	0.0099	6.4496
ResNet-50	UMR-UMP	0.0000	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.7477	0.0038	37.9345
	UMR-MP	**4.3895**	6.7568	10.4730	0.7025	8.4459	4.5742	0.0795	10.8108	5.4452	0.4263	0.0005	8.2432
	UMR-MP(T)	6.4169	7.7703	12.1622	0.4241	8.7838	4.6040	0.0000	17.9054	8.9527	0.3567	-0.0066	6.8853
	UMR-MP(SRT)	4.7274	7.4324	9.4595	0.8748	7.4324	**4.1536**	0.1193	**9.1216**	4.6205	0.4553	0.0014	8.4507
	MR-MP	6.8831	10.0156	13.7715	4.2316	7.8247	6.0281	1.1662	9.7027	5.4344	0.6496	0.0301	4.7924
	MR-MP(T)	6.8831	9.7027	14.0845	97.8759	0.0000	48.9379	90.7622	0.0000	45.3811	0.7759	0.3663	4.8791
	MR-MP(SRT)	**6.2578**	9.0767	11.8936	2.9738	8.1377	**5.5557**	0.8413	**9.3897**	5.1155	0.6488	0.0144	4.9381
MobileFaceNet	UMR-UMP	0.0106	0.0000	0.0000	1.0000	0.0000	0.5000	0.1000	0.0000	0.0500	0.7318	0.0078	26.4276
	UMR-MP	6.4169	16.8919	24.3243	0.9874	16.8919	8.9397	0.0663	27.3649	13.7156	0.3803	-0.0019	4.6457
	UMR-MP(T)	7.7685	15.8784	34.4595	0.6759	18.9189	9.7974	0.0596	37.1622	18.6109	0.3304	-0.0027	4.2067
	UMR-MP(SRT)	**6.079**	12.5000	21.9595	0.9675	13.1757	**7.0716**	0.0928	22.2973	**11.1950**	0.4157	-0.0018	5.2918
	MR-MP	8.4777	18.1534	28.7950	6.5056	10.3286	8.4171	1.9908	14.0845	8.0377	0.6087	0.0509	3.2505
	MR-MP(T)	8.7634	17.5274	26.2911	95.9683	0.0000	47.9842	84.9896	0.0000	42.4948	0.7638	0.3966	3.5408
	MR-MP(SRT)	**7.8232**	15.0235	22.5352	3.9733	9.0767	**6.525**	1.1745	14.3975	**7.7860**	0.6087	0.0241	3.5815

Table 4.2.: The achieved verification performance of different experimental settings by ResNet-100, ResNet-50 and MobileFaceNet models along with EUM trained with triplet loss and EUM trained with SRT loss. The result is reported using MRF2 dataset. The FMR100_Th$^{\text{UMR-UMP}}$ are equal to 0.1711, 0.2038 and 0.2351 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The FMR1000_Th$^{\text{UMR-UMP}}$ are equal to 0.2316, 0.2639 and 0.3041 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The lowest EER and the lowest average error of FMR100 and FMR1000 at the defined threshold for each of the evaluation cases and each of the evaluated models are marked in bold. One can notice the significant improvement in the verification performance induced by our proposed approach (SRT) in most evaluation cases.

4.5.4. Impact of EUM with SRT on in-the-wild masked FR

The achieved evaluation results on the MRF2 dataset by ResNet-100, ResNet-50, and MobileFaceNet models are presented in Tables 4.2. When probes are masked, the ERR achieved by the ResNet-100 model is reduced from 4.0515% (UMR-MP) to 3.3757% by our proposed approach (UMR-MP(SRT)). A similar improvement in the verification performance is achieved by our solution (MR-MP(SRT)) in the MR-MP evaluation setting, as shown in Table 4.2.

Using masked probes, the achieved EER by ResNet-50 model is 4.3895% (UMR-MP). Only in this case, the EER did not improve by our proposed approach (UMR-MP(SRT)). The achieved EER, in this case, by our proposed approach is 4.7274%. Nonetheless, a notable improvement in the FMR1000 and the FDR separability measures can be observed from the reported result. The increase in FDR points out the possibility that given larger and more representative evaluation data, the consistent enhancement in verification accuracy will be apparent here as well. A significant improvement in the verification performance is achieved by our approach when comparing masked probes to masked references. In this case, the achieved EER is decreased from 6.8831% (MR-MP) to 6.2578% (MR-MP(SRT)). A similar conclusion can be made from the improvements on the other performance verification measures and the FDR measure.

Using masked probes, the achieved verification performance by MobileFaceNet is significantly enhanced by our proposed approach (UMR-MP(SRT)). A similar improvement in the verification performance is achieved on MR-MP(SRT) case as shown in Table 4.2. For example, the achieved EER by MobileFaceNet is 8.4777% on the MR-MP case. This error rate is reduced to 7.8232% using our proposed approach (MR-MP(SRT)).

Considering the FMR100_Th$^{\text{UMR-UMP}}$ and the FMR1000_Th$^{\text{UMR-UMP}}$, the achieved FMR and FNMR improved is by our proposed solution (UMR-MP(SRT) and MR-MP(SRT)) in most evaluation cases, especially when the considered operation threshold is FMR100_Th$^{\text{UMR-UMP}}$. This result is also supported by ROC curves shown in Figures 4.5d, 4.5e and 4.5f.

4.5.5. Impact of EUM with SRT on simulated masked FR

In addition to the evaluation of the real masked face dataset presented in Section 4.5.3 and 4.5.4, we evaluated our proposed solution on two large synthetically generated masked faces datasets: LFW and IJB-C. The achieved verification performance on the synthetically generated masked face of LFW is presented in Table 4.3. The improvement in verification performance induced by our proposed solution on the synthetic masked face of LFW is observable for all evaluation cases.

Table 4.5 presents the achieved verification performance by the considered models on

LFW	Setting	EER%	FMR100%	FMR1000%	FMR100_Th$^{\text{UMR-UMP}}$			FMR1000_Th$^{\text{UMR-UMP}}$			G-mean	I-mean	FDR
					FMR%	FNMR%	Avg.%	FMR%	FNMR%	Avg.%			
ResNet-100	UMR-UMP	0.2660	0.2667	0.3333	1.0000	0.2667	0.6333	0.1000	0.3333	0.2167	0.7157	0.0026	33.0630
	UMR-MP	1.0000	0.9667	2.5667	1.0667	0.9667	**1.0167**	0.0667	2.9333	1.5000	0.5220	0.0019	13.1746
	UMR-MP(T)	1.7000	2.3667	4.4333	0.9667	2.5333	1.7500	0.0333	5.9667	3.0000	0.4115	0.0029	11.0452
	UMR-MP(SRT)	**0.8667**	0.8667	1.6000	1.2667	0.7667	**1.0167**	0.1000	1.7333	**0.9167**	0.5380	0.0024	15.0505
	MR-MP	**0.9667**	0.9667	2.4333	3.1000	0.7000	1.9000	0.5000	1.2000	**0.8500**	0.5996	0.0110	14.2278
	MR-MP(T)	1.7333	2.3333	10.1333	19.4667	0.3667	9.9167	6.7667	0.6667	3.7167	0.6290	0.0808	10.8161
	MR-MP(SRT)	**0.9667**	0.9667	2.0667	3.0000	0.6667	**1.8333**	0.4667	1.5333	1.0000	0.6035	0.0053	14.6018
ResNet-50	UMR-UMP	0.3333	0.3000	0.4000	1.0000	0.3000	0.6500	0.1000	0.4000	0.2500	0.7023	0.0029	26.5107
	UMR-MP	1.4667	1.8333	3.3000	1.0000	1.8333	1.4167	0.1000	3.5667	1.8333	0.5117	0.0014	11.8522
	UMR-MP(T)	2.0000	2.7000	4.9667	0.6333	3.3333	1.9833	0.0667	6.6333	3.3500	0.4278	0.0020	10.5553
	UMR-MP(SRT)	**1.1000**	1.1333	2.4000	0.9667	1.1333	**1.0500**	0.2000	2.2000	**1.2000**	0.5427	0.0016	14.5079
	MR-MP	1.3667	1.7333	4.7333	3.0000	0.8333	1.9167	0.9000	1.9333	1.4167	0.5893	0.0158	12.2339
	MR-MP(T)	2.0333	2.9667	7.2000	10.8667	0.7667	5.8167	4.0333	1.5333	2.7833	0.6256	0.0525	10.2560
	MR-MP(SRT)	**1.2333**	1.4333	2.9667	2.2333	0.9333	**1.5833**	0.6333	1.5333	**1.0833**	0.6051	0.0053	13.4416
MobileFaceNet	UMR-UMP	0.6333	0.6000	1.3000	1.0000	0.6000	0.8000	0.1000	1.3000	0.7000	0.6742	0.0051	18.2460
	UMR-MP	3.2333	5.9333	12.0333	0.7667	6.7333	3.7500	0.0000	18.2667	9.1333	0.4641	-0.0011	7.5840
	UMR-MP(T)	3.6667	7.1333	17.6667	0.6000	8.7667	4.6833	0.0000	27.4333	13.7167	0.4023	0.0013	7.2341
	UMR-MP(SRT)	**1.8667**	2.4667	8.1333	0.8333	2.8667	**1.8500**	0.1000	9.3667	**4.7333**	0.5144	0.0006	10.2266
	MR-MP	3.3333	6.4667	17.9000	5.7667	2.6333	4.2000	0.8333	7.1333	3.9833	0.5688	0.0505	7.7096
	MR-MP(T)	3.0667	5.2000	13.6333	93.9000	0.0000	46.9500	72.1333	0.0667	36.1000	0.7495	0.3970	7.7594
	MR-MP(SRT)	**2.2667**	3.5333	11.1000	2.3000	2.2333	**2.2667**	0.4667	5.9667	**3.2167**	0.5872	0.0091	9.6183

Table 4.3.: The achieved verification performance of different experimental settings by ResNet-100, ResNet-50, and MobileFaceNet models along with EUM trained with triplet loss and EUM trained with SRT loss. The result is reported using synthetically generated masked faces of the LFW dataset. The FMR100_Th$^{\text{UMR-UMP}}$ are equal to 0.1736, 0.2052 and 0.2449 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The FMR1000_Th$^{\text{UMR-UMP}}$ are equal to 0.2451, 0.2617 and 0.3450 are for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The lowest EER and the lowest average error of FMR100 and FMR1000 at the defined threshold are marked in bold. It can be noticed the significant improvement in the verification performance induced by our proposed approach (SRT) in most evaluation cases.

the synthetically generated masked face of IJB-C. When the reference and the probes are synthetically masked, the achieved EER by ResNet-100 is 2.7356% (MR-MP). Only in this case for synthetically masked face dataset, the EER did not improve by our proposed approach, where the EER achieved by our approach is 2.9197% (MR-MP(SRT)). However, when the operation threshold is set to FMR100_Th$^{\text{UMR-UMP}}$, a notable reduction in the average of FMR and FNMR can be observed for all evaluation cases. These reported results on synthetically generated masked face datasets support our achievement on real masked face datasets. Also, it points out the competence of our proposed solution in improving the masked face verification performance. A similar observation can be noticed in the ROC curves in Figures 4.6a, 4.6b, 4.6c, 4.6d, 4.6e, 4.6f.

4.5.6. Ablation study on self-restrained triplet loss

In this section, we experimentally prove and theoretically discuss the advantage of our proposed SRT solution over the common naive triplet loss. We explore first the validity of training the EUM model with triplet loss using masked face datasets. It is noticeable that training EUM with naive triplet is inefficient for learning from masked face embedding as presented in in Tables 4.1, 4.2, 4.3 and 4.5. For example, when the probe is masked, the achieved EER by EUM with triplet loss on top of ResNet-50 is 1.9789% (UMR-MP(T)), in comparison to 0.9611% EER achieved by EUM with our SRT (UMR-MP(SRT)), as shown in Table 4.1. It is crucial for learning with triplet loss that the input triplet violate the condition $d(f(x_i^a), f(x_i^n)) > d(f(x_i^a), f(x_i^p)) + m$. Thus, the model can learn to minimize the distance between the genuine pairs and maximize the distance between the imposter pairs. When the previous condition is not violated, the loss value will be close to zero and the model will not be able to further optimize the distances of the genuine pairs and imposter pairs. This motivated our SRT solution.

Given that our proposed EUM solution is built on top of a pre-trained FR model, the feature embeddings of the genuine pairs are similar (to a large degree), and the ones of imposter pairs are dissimilar. However, this similarity is affected (to some degree) when the faces are masked. The learning objective of our approach is to reduce this effect. This statement can be observed from the achieved results presented in Tables 4.1, 4.2, 4.3 and 4.5. For example, using the MFR dataset, the achieved G-mean and I-mean by ResNet-50 is 0.8538 and 0.0349 (UMR-UMP), respectively. When the probe is masked (UMR-MP), the achieved G-mean and I-mean shift to 0.5254 and 0.0251, respectively, as shown in Table 4.1. The shifting in the G-mean points out that the similarity between the genuine pairs is reduced (to some degree) when the probe is masked. Training EUM with naive triplet loss requires selecting a triplet of embeddings that violated the triplet condition. As we discussed earlier, the masked anchor is similar (to some degree) to the positive (unmasked embedding), and it is dissimilar (to some degree) from the negative. Therefore, finding triplets that violate the triplet condition is not trivia. Also, it could not be possible for many triplets in the training dataset. This explains the poor result achieved when the EUM model is trained with triplet loss, as there are only a few triplets violating the triplet loss condition. One can assume that using a larger margin value allows the EUM model to further optimize the distance between genuine pairs and imposter pairs, as the triplet condition can be violated by increasing the margin value. However, by increasing the margin value, we increase the upper bound of the loss function. Thus, we ignore the fact that the distance between imposter pairs is sufficient with respect to the distance between genuine pairs in the embedding space. For example, using unmasked data, the mean of the imposter scores achieved by ResNet-50 on the MFR dataset is 0.0349.

When the probe is masked, the mean value of the imposter scores is 0.0251, as shown in Table 4.1. Therefore, any further optimization on the distance between the imposter pairs will affect the discriminative features learned by the base FR model. Also, there is no restriction in the learning process ensured that the model will maintain the distance between the imposter pairs. Alternatively, training the EUM model with our SRT loss achieved significant improvement in minimizing the distance between the genuine pairs. Simultaneously, it maintains the distance between the imposter pairs to be closer to the one learned by the base FR model. It is noticeable from the reported result that the I-mean achieved by our SRT is closer to the I-mean achieved when the model is evaluated on unmasked data, in comparison to the one achieved by naive triplet loss, as shown in Tables 4.1, 4.2, 4.3 and 4.5. The achieved result points out the efficiency of our proposed EUM trained with SRT in improving the masked FR, in comparison to the considered face recondition models. Also, it supported our theoretical motivation behind SRT where training the EUM with SRT significantly outperformed the EUM trained with naive triplet loss.

The proposed solution is designed and trained to manipulate masked face embedding and not to manipulate unmasked one. Based on this workflow, our EUM solution will not be used on unmasked faces. This is based on the assumption that the existence of the mask is known, e.g., by the automatic detection of wearing a face mask that can be relatively easily detected where most of mask face detection methods proposed in the literature achieved very high accuracy in detecting masked face (more than 99% [163]). Despite the fact that our workflow does not assume processing unmasked faces, and for the sake of experiment completeness, we apply our solution on UMR-UMP cases. The achieved results showed slight degradation in face verification performance in a number of the experimental settings. However, this result was expected as the proposed solution is designed and trained to operate on masked face embedding rather than processing an unmasked face embedding. In the following, we present the achieved results by our proposed approach when it is applied to the UMR-UMP case. When ResNet-100 and ResNet-50 are evaluated on the MFR and MRF2 datasets, and the unmasked face embeddings (UMR-UMP) are processed by EUM with the SRT solution, the achieved EER and FMR100 are 0.0% and 0.0%, respectively. When MobileFaceNet is evaluated on the MFR or MRF2 datasets and the unmasked face embedding (UMR-UMP) are processed by EUM with the SRT solution, the verification performances are slightly degraded. In this case, the EER increases from 0.0% to 0.0112% EER, when MobileFaceNet is evaluated on the MFR dataset. When MobileFaceNet is evaluated on the MRF2 dataset, the EER value increases from 0.0106% to 0.2124%. The achieved FMR100, in this case, is 0.0%. When the considered models are evaluated on the LFW and the unmasked face embeddings (UMR-UMP) are processed by EUM with the SRT solution, the verification performances obtained by the considered models slightly deteriorate. In this case, the EER and the FMR100 by the ResNet-100 model decrease from 0.2660% and 0.2667% to 0.3% and 0.2667%, respectively. When the considered model is ResNet-50, the achieved EER and FMR100 are degraded from 0.3333% and 0.3000% to 0.5333% and 0.5000%, respectively. For the MobileFaceNet model, the achieved EER and FMR100 are degraded from 0.6333% and 0.6000% to 1.1667% and 1.2000%, respectively. By applying our approach on UMR-UMP cases of the IJB-C dataset, the achieved verification performance is degraded from 1.5340% to 1.5595% EER and from 1.6362% to 1.7027% FMR on top of the ResNet-100 model. For the ResNet-50 model, the achieved EER and FMR100 are degraded from 1.6881% to 2.0709% EER and from 1.8663% to 2.4857% FMR. For the MobileFaceNet model, the achieved EER and FMR are degraded from 2.396% to 2.8379% EER and from 2.7918% to 4.1417%. This performance trend in the UMR-UMP setting is expected as processing unmasked face embedding by EUM with SRT is not the aim of our proposed solution and do not match its operational concept, where unmasked faces will not be processed by the EUM. The conducted experiments are thus only included for the sake of experiment completeness.

Model	Param. (M)	MFLOPs
ResNet100	65.16	24211.78
EUM	1.10	2.10
ResNet100+ EUM	66.26	24213.88
ResNet50	43.59	12639.29
EUM	1.10	2.10
ResNet50+EUM	44.60	12641.30
MobileFaceNet	0.99	474.84
EUM	0.07	0.13
MobileFaceNet+EUM	1.06	474.97

Table 4.4.: The computational cost of the base FR models and the proposed EUMs in terms of number of parameters (Param) and computational complexity (FLOPs). The proposed EUM adds minor computational cost to the base FR model.

4.5.7. Discussion

In summary, the reported results in this work illustrate how the verification performance of current FR models proposed in the literature is affected by wearing a face mask and how this can be improved by learning to process the masked face embedding to behave more similarly to an embedding from an unmasked face. This has been demonstrated through extensive experimental evaluations of three FR models and four masked face datasets. The evaluation datasets include two real masked datasets captured under different scenarios: in the wild (MRF2) and collaborative (MFR), and two synthetically generated masked faces of large-scale datasets: LFW and IJB-C. We have also theoretically and experimentally demonstrated the competence of our proposed EUM together with SRT in reducing the negative influence of the masked face on the FR performance. The competence of our solution in improving the masked face verification performance has been demonstrated on real masked datasets captured under different scenarios (in the wild and collaborative) and on synthetically generated masked faces of large-scale datasets.

The proposed EUM model does not require retraining existing FR models or deploying separate solutions for masked face recognition. Table 4.4 presents the computational cost of the base FR models and EUMs. It can be noticed that EUMs only add little computational cost to base FR (around 1%).

From research to industry perspective, the developers of commercial FR solutions could use our proposed concept to improve the performance of their algorithms when processing masked face images. Many commercial FR solutions produce face templates to enable template storage instead of images in large-scale datasets. The advantages of storing face templates are to enable faster identification searches and matching, by avoiding the re-generation of embeddings in every search. As our solution operates on embedding space, the commercial models can benefit from our solution to improve the performance of their algorithms when facing a masked face image. Examples of such commercial solutions are Neurotechnology [198] and Cognitec [51] (achieved high accuracies in NIST Ongoing FR Vendor Test (FRVT)[102]). Such solutions produce face templates to populate the biometric datasets to enable efficient biometric searches.

4.6. Masked face recognition competition

This section presents a summary of the MFR competition held within the 2021 International Joint Conference on Biometrics (IJCB 2021) [29]. This competition is designed to motivate worldwide technical solutions from academia and industry aiming at enhancing the accuracy of masked FR on real face masks and in a collaborative face verification scenario. The competition attracted a total of 10 participating teams with valid submissions. The affiliations of these teams are diverse and associated with academia and industry in nine different countries. These teams successfully submitted 18 valid solutions. A private dataset representing a collaborative, multi-session, real masked, capture scenario is used

IJB-C	Setting	EER%	FMR100%	FMR1000%	FMR100_Th$^{\text{UMR-UMP}}$			FMR1000_Th$^{\text{UMR-UMP}}$			G-mean	I-mean	FDR
					FMR%	FNMR%	Avg.%	FMR%	FNMT	Avg.%			
ResNet-100	UMR-UMP	1.5340	1.6362	2.4799	1.0000	1.6362	1.3181	0.1000	2.4799	1.2900	0.7460	0.0034	15.7436
	UMR-MP	2.6026	3.3492	5.8751	1.0684	3.3032	2.1858	0.1133	5.6502	2.8817	0.5593	0.0050	9.7516
	UMR-MP(T)	5.0724	8.3500	13.9643	0.6437	9.2857	4.9647	0.0333	17.0374	8.5353	0.3966	0.0004	6.0168
	UMR-MP(SRT)	**2.5476**	3.2520	5.7575	1.0563	3.2214	**2.1388**	0.1112	5.5837	**2.8474**	0.5667	0.0038	9.9005
	MR-MP	**2.7356**	3.7685	6.9643	4.3300	2.3163	3.3232	0.9488	3.7992	**2.3740**	0.6751	0.0228	10.2180
	MR-MP(T)	5.2834	14.7415	42.2509	52.8218	0.4909	26.6563	31.5749	1.1403	16.3576	0.7273	0.1981	6.6082
	MR-MP(SRT)	2.9197	3.9781	7.3631	3.4202	2.7663	**3.0932**	0.6837	4.4588	2.5712	0.6604	0.0129	9.4975
ResNet-50	UMR-UMP	1.6881	1.8663	3.0782	1.0000	1.8663	1.4332	0.1000	3.0782	1.5891	0.7370	0.0061	14.6355
	UMR-MP	2.8634	4.2593	7.9971	1.0257	4.2389	2.6323	0.1045	7.9051	4.0048	0.5505	0.0091	9.1274
	UMR-MP(T)	4.9547	8.3602	15.0995	0.5824	9.6436	5.1130	0.0317	19.0878	9.5597	0.4227	0.0005	6.3770
	UMR-MP(SRT)	**2.7221**	3.8401	7.4142	1.0675	3.7685	**2.4180**	0.1162	7.1944	**3.6553**	0.5731	0.0061	9.5896
	MR-MP	3.2418	4.9138	10.0680	5.1556	2.6026	3.8791	1.1855	4.6275	**2.9065**	0.6698	0.0395	9.2267
	MR-MP(T)	4.8065	10.6202	30.2398	28.2029	1.1556	14.6793	12.5160	2.5055	7.5107	0.7126	0.1396	7.1907
	MR-MP(SRT)	**3.0833**	4.6940	9.4186	3.2926	3.0373	**3.1649**	0.6722	5.3485	3.0103	0.6585	0.0175	9.0671
MobileFaceNet	UMR-UMP	2.2396	2.7918	5.0826	1.0000	2.7918	1.8959	0.1000	5.0826	2.5913	0.7150	0.0075	11.6725
	UMR-MP	4.6539	8.5698	17.1908	0.9843	8.6056	4.7949	0.0910	17.6305	8.8608	0.4997	0.0121	6.5141
	UMR-MP(T)	9.1834	21.4297	35.7315	0.2993	29.0229	14.6611	0.0086	51.6950	25.8518	0.3273	-0.0117	3.7926
	UMR-MP(SRT)	**4.0548**	7.1995	14.5421	0.9831	7.2506	**4.1169**	0.0974	14.6495	**7.3734**	0.5295	0.0056	7.2243
	MR-MP	5.0339	9.7305	20.6064	10.1750	3.4003	6.7877	2.6584	6.7137	**4.6860**	0.6624	0.0939	6.6892
	MR-MP(T)	8.9175	21.9972	39.7454	99.6336	0.0205	49.8270	96.8945	0.0818	48.4881	0.8281	0.5465	3.9353
	MR-MP(SRT)	**4.6837**	9.0249	18.8782	4.2937	4.9241	**4.6089**	0.9800	9.1016	5.0408	0.6353	0.0301	6.9284

Table 4.5.: The achieved verification performance of different experimental settings by ResNet-100, ResNet-50, and MobileFaceNet models along with EUM trained with triplet loss and EUM trained with SRT loss. The result is reported using synthetically generated masked faces of the IJB-C dataset. The FMR100_Th$^{\text{UMR-UMP}}$ are equal to 0.1804, 0.2143 and 0.2546 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The FMR1000_Th$^{\text{UMR-UMP}}$ are equal to 0.2557, 0.2990 and 0.3493 for ResNet-100, ResNet-50 and MobileFaceNet, respectively. The lowest EER and the lowest average error of FMR100 and FMR1000 at the defined threshold for each of the evaluation cases and each of the evaluated models are marked in bold. One can notice the significant improvement in the verification performance induced by our proposed approach (SRT) in most of the evaluation cases.

to evaluate the submitted solutions [29]. The evaluation dataset contains 3290 images of 47 identities.

The verification performance of submitted solutions is evaluated under two settings: 1) Masked vs. not-masked verification pairs (noted as BLR-MP). 2) Masked vs. masked verification pairs (noted as MR-MP). Moreover, the competition considered the deployability of the proposed solutions by taking the compactness of the FR models into account. The model compactness is reported as the number of trainable parameters. A summary of the submitted solutions and the achieved results is provided in the following. Details on the MFR competition are provided in the appendix of this thesis.

Algorithms Most of the submitted algorithms used ResNet [107] and SEResNet [110] architectures as a main backbone for the proposed solution (17 out of 18 models). One solution opted to use FaceNet [234]. 16 solutions used softmax and margin penalty softmax losses [80, 78] to train the proposed models. Two solutions used triplet loss [234]. Most of the submitted solutions (16 out of 18) opted to augment the training dataset with masked face data. One team submitted two solutions trained using the periocular area, i.e., the upper region of a face image. The smallest model has 23.8M of parameters, and the largest one contains 108.9M parameters. None of the submitted solutions propose a solution that could be applied on top of the existing FR model, as the proposed EUM with the SRT approach presented in this chapter.

Verification performance The MFR competition used ResNet-100 [107] architecture pretrained on MS1MV2 [103, 80] as baseline. Most of the presented solutions achieved a competitive verification performance compared to the baseline. Ten out of 18 solutions achieved higher verification performance than the baseline solution for BLR-MP and MR-MP evaluation settings. The achieved verification performances in terms of FMR100 by baseline solution were 0.06009 and 0.05925 for BLR-MP and MR-MP evaluation settings, respectively. The best verification performances by the top-performing submitted models in terms of FMR100 were 0.05095 and 0.04489 for BLR-MP and MR-MP evaluation settings, respectively.

4.7. Summary

This chapter presented and evaluated a novel solution to reduce the negative impact of wearing a protective face mask on FR performance. This work was motivated by the recent evaluation efforts on the effect of masked faces on FR performance. The presented solution is designed to operate on top of existing FR models, thus avoiding the need for retraining existing FR solutions used for unmasked faces. This goal has been accomplished by proposing the EUM operated on the embedding space. The learning objective of our EUM is to increase the similarity between genuine unmasked-masked pairs and decrease the similarity between imposter pairs. We achieved this learning objective by proposing a novel loss function, the SRT which, unlike triplet loss, dynamically self-adjust its learning objective by concentrating on optimizing the distance between the genuine pairs only when the distance between the imposter pairs is deemed to be sufficient. Through ablation study and experiments on four masked face datasets and three FR models, we demonstrated that our proposed EUM with SRT significantly improved the masked face verification performance in most experimental settings, providing an answer to RQ2.1. This chapter also presented a summary of MFR competition designed to motivate solutions aiming at enhancing the FR accuracy of masked faces.

The next chapter will present the contribution of this thesis aiming at enabling biometric recognition in virtual and augmented reality applications enabled by HMDs.

5. Biometrics in head-mounted displays

The previous chapter presented a novel approach to reduce the effect of wearing a face mask on face recognition verification performance. This chapter focuses on introducing biometric recognition to VR/AR applications enabled by HMD devices. This chapter is based on the publications [25, 32, 33, 34, 35].

5.1. Introduction

An essential aspect of security-related and penalization-driven access control applications is linked to the accurate identification of individuals. With the growing interest of VR/AR applications such as entertainment applications [259], manufacturing [194], healthcare [172], law enforcement [246] and education [280], identifying the users within the associated headset is becoming a critical challenge for VR/AR systems for secure access. Identification of VR/AR users can further be used to prevent unauthorized access to the system, enhance user privacy, and guarantee the safety of using the system in multi-user environments. For example, in the domain of field policing and crime scene investigation, Poelman et el. [213] utilized an AR system to enable crime scene investigators to access remote support from the experts, enabling collaborative spatial analysis of location. This type of information should only be accessible to authorized users. Enabling VR/AR technologies in real application scenarios allows sensitive information to be accessible to the user and, if not carefully handled, can cause a considerable amount of damage. This sensitive information should be properly and continuously authenticated to prevent anonymous access to private and sensitive data. These, among many other application scenarios, raise a question regarding the security mechanism in such headsets.

The current security mechanism in VR and VR use-cases depends on pattern matching or Personal Identification Number (PIN) to authenticate the user in the VR environments [98]. Such authentication mechanism is limited to individual knowledge and does not allow continuous authentication while the device is being used without interruption. Kupin et al. [147] proposed a security approach for VR devices based on biometric data collected by tracking the behavior of users, achieving 90% identification accuracy. However, this approach is limited to specific application scenarios where the user performs a predefined physical task through a high degree of user cooperation. By design, HMD for AR/VR applications are commonly built with a camera to enable gaze interaction with the virtual environment. We assert that such a camera can be used to verify the user identity by relying on biometric characteristics captured during regular use. Using the HMD camera, two biometric characteristics - iris and periocular region, can be captured continuously, enabling us to employ them for biometric authentication of the users.

While we note this is an inherent advantage, we also draw attention to one of the main challenges for deploying such solutions, i.e., the limited computational and storage power of HMD devices. Large segmentation or feature extraction models are not realistically deployable in such low computational powered and mobile devices. As noted in earlier works, the performance of iris recognition depends on the precise segmentation of the iris area, especially in uncooperative capture scenarios [294, 217]. As a second factor, it has to be noted that common iris recognition approaches require significant cooperation of the user to capture high-quality iris images with relatively widely open eyes to enable high segmentation accuracy. Within the AR/VR scenarios, the user should not be required to continuously and intentionally collaborate with the identity verification sub-system impeding the use of AR/VR, rather the authentication system should run in the background. Such an interaction results in sub-optimal iris capture, unlike the traditional iris recognition systems. Considering the minimalistic hardware specifications available in such applications and the need for reliable segmentation accuracy, one of the main challenges for iris recognition in HMD is creating an accurate yet efficient (from a model size perspective) segmentation approach. We also assert that the periocular region captured during the interaction can supplement and mitigate the lower performance due to iris alone [207]. As noted in earlier works, especially for non-collaborative biometrics where the need for user collaboration can be relaxed, the periocular region has proven to improve the authentication performance when used along with iris [218].

Motivated by such arguments, this chapter investigates and evaluates the feasibility of using captures from HMD internal cameras for biometric recognition. In the efforts seeking to answer RQ3.1, RQ3.2, and RQ3.3, this chapter presents three main contributions:

- As a response to RQ3.1, this chapter provides a comprehensive evaluation of three widely used iris features extractions methodologies with variations in the comparison process. Furthermore, this chapter evaluates the verification performance of four well-used periocular feature extraction approaches. This work also analyses the relative utility of the iris images captured under the non-collaborative environment by studying the relative low iris area available in the captured images. Based on this, this chapter provides in-depth analyses on the effect of iris selection on the general performance and the expected gap in the sequence of eye captures when a threshold based on the amount of visible iris is applied. Also, this chapter presents a new methodology to select the suitable iris for better biometric performance on HMD devices.

- As a response to RQ3.2, this chapter presents a multi-scale segmentation network based on a cascade framework that considers the image information at multiple scales. The largest segmentation model has 6574k parameters. We propose to reduce the number of parameters to 216K parameters in the second model (Eye-MMS216) and 80K in the third model (Eye-MMS80) by taking advantage of the fact that segmentation takes an image from a highly detailed space to a space with a small number of discrete labels. Both Eye-MMS216 and Eye-MMS80 models achieved over 91% mean Intersection over Union (IoU) on the four label regions. Despite the small number of parameters, the compact model performs similarly to the initially proposed model with 6574k parameters (two percentages lower than the accuracy of the larger model).

- As a response to RQ3.3, this chapter presents a two-stage image generation network (D-ID-Net) for generating synthetic images that can be used for large-scale training data generation or presentation attacks (aka. spoofing attacks) in future works. The first stage (network) is the domain network (D-Net), which transfers the semantic labels into an eye-like image without specifically addressing the identity issue. The second stage (network) is the identity-specific network (ID-Net) which induces identity-related information into the output of the D-NET and generates a realistic image that corresponds to the initial semantic label and possesses the appearance of a specified identity.

In the rest of the chapter, Section 5.2 discusses the related work focusing on issues related to biometrics in HMD, ocular semantic segmentation, iris, and periocular recognition, as well as identity preserving image generation. This is followed by a detailed description of the algorithmic methodology proposed and deployed for the various tasks addressed in this work. Section 5.4 presents the implementation and experimental details needed to assure the reproducibility of the work and to enable in-depth comprehension of the achieved results. Section 5.5 presents and discusses the achieved results for the segmentation, generation, and the different investigated verification setups. A set of conclusions are drawn in Section 5.7 to motivate future work on HMD-based biometrics.

5.2. Related work

This section discusses the related work focusing on biometrics in HMD, ocular semantic segmentation, iris, and periocular recognition, as well as identity-preserving image generation.

Biometrics in HMD A growing number of applications are integrating AR/VR headsets. Verifying and maintaining the trust in the identity of the user might be necessary, especially in security-related applications. An example of that is the foreseen use of AR headsets for border guards in both crowded border-crossing points and remote locations. This use case is being developed by the H2020 EU-funded project ARESIBO [5]. In such a scenario, the information processed and displayed to the user is of highly-secure nature, and the user identity should be verified without disturbance under controlled settings. A previous work by Bastias et al. [20] proposed a method for iris reconstruction from several 2D near-infrared iris images and designed a sensor for 2D image capturing, which is mounted on a wearable headset. However, the work did not target wearable headsets or the cameras within the HMD devices, rather it used the setup to create a capturing mechanism and proposed a consequent verification approach.

Olade et al.[204] discussed multimodal facial biometric authentication and mentioned the practical need for mapping such approaches to authentication within the HMD, however, without providing any experimental study that includes HMDs. Kim and Lee [138] discussed periocular biometric verification in HMDs without providing any details on the captured device and database structure. Their best-performing comparison approach was a naive L1 distance between the pixel values of the images, which raises many concerns on the scalability and generalizability of the solution.

Very recently, the OpenEDS database was released [96], which is a large-scale iris image dataset captured using a virtual-reality HMD with two eye-facing cameras. Based on the OpenEDS, Facebook hosted a competition for two main challenges, semantic segmentation, and synthetic eye generation [6]. The main goal of the competition is to address different eye-tracking solutions for VR and AR. Although, the main goal of OpenEDS is to enable eye gaze for VR/AR application, it opens a new opportunity to evaluate iris and periocular biometric recognition in HMD setups for VR/AR application.

Eye semantic segmentation Previous works that addressed semantic image segmentation for the ocular region mainly focused on iris or sclera segmentation. Sclera can be a biometric characteristic, but its segmentation also acts as a way to detect the outer boundaries of the iris. This had been motivated mainly by the high interest in iris recognition, as one of the most accurate biometrics characteristics [126]. The localization (segmentation) accuracy of the iris significantly affects the iris recognition performance [168]. Earlier works have suggested segmenting the iris region by defining its boundaries, e.g., by Hough transforms [275]. More recent works followed the trend in generic segmentation and detected the iris region by utilizing Fully Convolutional Network (FCN) [157, 22] or U-Net [167].

Since 2015 a series of competitions have addressed sclera segmentation in the last five years [70]. The latest competition [69] focused on variations in the capture angle and the use of mobile devices. The winning team employed U-Net structure [227] modified by a channel attention module as described by Yu et al. [287].

Eye-tracking can benefit greatly from multiple region semantic segmentation of the ocular area. However, only recent activities have targeted this problem and provided appropriate research databases. One of these is the iBUG Eye Segmentation Dataset [170] where relatively low-resolution ocular regions are segmented into two labels, iris and pupil as one class, and sclera as the second class. The work also proposed a segmentation solution based on a convolution neural network followed by refinement using a conditional random field. Rot et al. [230] also addressed the multi-region (iris, sclera, pupil, periocular, eyelashes, and canthus) segmentation issue by building a convolutional encoder-decoder solution, however, with a database of a limited size. OpenEDS [96] derived a large scale iris images dataset captured using a virtual-reality HMD with two eye-facing cameras.

Luo et al. [169] have recently created the Eye Segmentation in the Wild dataset, which is a large scale eye images dataset collected and manually annotated from several datasets [148, 238, 232, 212, 231, 186, 242]. However, the released dataset does not contain identity labels. Therefore it could not be used for identity verification evaluation. The work also proposed a multi-region semantic segmentation method based on SegNet [17] followed by a pre-trained encoder and discriminator to regularise the model during the training phase. However, the SegNet model contains a relatively large number of trainable parameters (29.46 million), which is very large for devices with limited computational power.

Generic image segmentation solutions have achieved increasingly impressive performances since the rise of deep learning. Main advances in this regard are segmentation based on FCN [164], U-Net [227], Feature Pyramid Network [153], Mask R-CNN [105], DeepLabv3+ [47], Path Aggregation Network [158], and most recently the Context Encoding Network [288]. However, few works have addressed segmentation solutions constrained by very limited computational resources (e.g. embedded systems). The latest of such works is the Fast-SCNN [215] that resulted in a model with 1.1 million parameters, which is still large for some embedded devices. Other works such as ENet [209] achieved a significantly small model size (0.37 million parameters). Nevertheless, this reduction in the model size notably affected the prediction accuracy.

Iris recognition Iris recognition is one of the most reliable and highly accurate biometric verification methods [73]. One of the most accurate and widely deployed approaches to extract iris features was proposed by Daugman [73]. Further, iris recognition approaches have been proposed in the literature, whether they are derivative of Daugman's iris features or based on deep learning techniques. For example, Sun and Tan [251] presented ordinal measures(OM) as a new type of iris features. Miyazawa [188] proposed an approach based on Discrete Fourier Transforms (DFT). More recently, an iris recognition approach was presented by Chen et al. [46], where the authors built a new set of iris features based on Human-interpreted Crypt Features.

This advancement in iris recognition techniques has driven many security-related applications to implement biometric identification based on iris recognition algorithms. In most application scenarios, the iris images are acquired in a collaborative/cooperative environment under ideal conditions to achieve maximum performance of iris recognition. In the collaborative application scenario, the acquisition of an iris image requires user collaboration, where the distance between user and camera is limited, and the user is looking directly into the camera with open eyes. This has traditionally allowed the systems to capture less noisy images that also contain a maximized portion of the iris area.An example of such applications is iris recognition in the border-crossing processes deployed in many airports [74] e.g., United Kingdom, Canada, and United Arab Emirates. Despite the fact that these solutions are scalable and efficient in terms of performance [72], the limitation to collaborative environment and restriction in using Near Infra-Red (NIR) cameras make these solutions not directly suitable to use-cases where such conditions are not necessarily feasible, e.g., mobile phones, VR/AR headsets, or in the automotive domain.

The non-collaborative iris recognition has recently gained interest from different applications. In such application scenarios, the user is not expected to make an effort and intentionally present an ideal iris sample to the capture device. An example of that is the adoption of iris recognition in the automotive domain, where identity information can help provide security and personalization. For example, Gentex, in collaboration with Delta ID, developed a biometric identification system based on iris recognition to be used inside vehicles [97].

Periocular recognition Periocular recognition provides a trade-off between using the iris or the entire face for identity verification by considering a small area around the eye, including eyelids, lashes, and eyebrows, as biometric trait [207].

Periocular recognition is particularly appropriate for a non-collaborative real-world scenario [132] where the need for user cooperation can be relaxed in comparison to iris recognition[207]. The performance of iris recognition greatly depends on the precise isolation of the iris area [200]. Therefore, using periocular area for person identification avoids the need for highly accurate iris segmentation, which can be a challenge for iris recognition in less controlled scenarios [129]. Another advantage is that the iris and face capture typically contain the periocular region, thus it can be easily obtained using the existing capture setups and can potentially improve the overall performance by fusing periocular, and iris/face information [277, 9, 221].

Moreover, periocular recognition is highly robust to expression variations [8], and aging [131, 146] in comparison to face recognition. Also, it enables matching partial captured face images [133].

Traditionally, periocular recognition methods utilized handcrafted features extracted from periocular images. Park et al. [207] proposed one of the first works to use the periocular area as a biometric trait captured under a controlled environment. The work depends on handcrafted features obtained by three descriptors, Local Binary Patterns (LBP), Histograms of Oriented Gradients (HOG), and Scale-Invariant Feature Transform (SIFT), followed by score fusion to build a verification decision. Following the works of Park et al. [207], several subsequent works were proposed in the literature: Juefei-Xu et al. [133] utilized LBP to encode discrete transforms enabling translation-robust descriptor. Mahalingam and Ricanek Jr [175] proposed the use of multi-scale patch-based LBP feature descriptors. Ross et al. [229] presented a fusion-based scheme to handle the variability in input periocular images. The introduced method is based on three descriptors: HoG to model global information, SIFT to extract local edge anomalies, and probabilistic deformation models to handle non-linear deformations. Woodard et al. [278] proposed a method based on local appearance-based feature representation by adding the color histogram to LBP features. Joshi et .al [130] proposed using both eyes for the periocular recognition task and computing feature representation by calculating the mean of a bank of complex Gabor filters and then normalizing it using zero-mean and unit variance normalization.

With the increased interest in deep learning techniques, many methods explored the use of deep learning techniques in periocular recognition. [218] trained convolution neural networks (CNNs) to implicitly learn the region of interest (periocular area) while ignoring the ocular region of the input image. Therefore, the proposed method does not require an explicit segmentation of the eye image during the recognition phase. Zhao and Kumar [294] proposed a Semantics-Assisted CNN (SCNN) model that consists of several CNN models. The main CNN model is trained based on the identity information while the other models are trained with semantic information such as gender and ethnicity and then joint the output of the models at the end to produce a feature representation or perform score fusion.

Identity preserving image generation The statistical properties of a generated image should be similar to those of a natural image. The parametric density estimation of natural image distribution is a function that maximizes the probability of producing an output similar to the observed data. Traditionally, this challenging problem has been handled using Restricted Boltzmann Machines [243] and vanilla autoencoders [263]. Due to the latest developments in deep learning techniques, generative models benefited from deep architectures and have achieved very promising results. This has lead to several pivotal techniques based on deep generative models, including Variational Auto Encoder [141], Generative Adversarial Network (GAN) [100], and Auto-regressive models [260, 261].

Image synthesis from semantic segmentation is a specific application of image-to-image translation task. The goal of image-to-image translation is to generate a new image conditioning on certain input. Major works in this direction utilized conditional GAN architectures (e.g. pix2pix [124]) and CNN architectures [48]. Several works propose to train GAN in an unsupervised fashion using cycle consistency GAN to handle the absence of unpaired training data [135, 116, 45]. Chen et al. [48] presented a cascade framework for high-resolution photographic image synthesizing from semantic layouts.

Generating identity-preserving images was commonly studied in the context of face images. Most solutions based their works on GAN, with some works targeting pose-specific generation [43] and others enhancing the identity-preservation by leveraging a three-player GAN [239]. Antipov et al. [12] proposed a method for identity-preserving face aging synthesis based on conditional GAN. Li et al. [149] proposed a method for generating face images from a predefined set of attributes while preserving the input face identity. Similarly, Bao et al. [19] proposed a method for identity-preserving face generation by combining identity representation from an initial input image with attribute representation from a second input image, then synthesizing a new face image from the combined representation. Huang et al. [115] proposed Two-Pathway GAN (TP-GAN) to synthesize frontal faces by simultaneously perceiving global structures and local details. Based on this work, Wang et al. [269] proposed multiple discriminators to synthesize high-resolution images. Generating an image that would preserve multiple identities was also addressed in the context of face morphing attacks [67, 56].

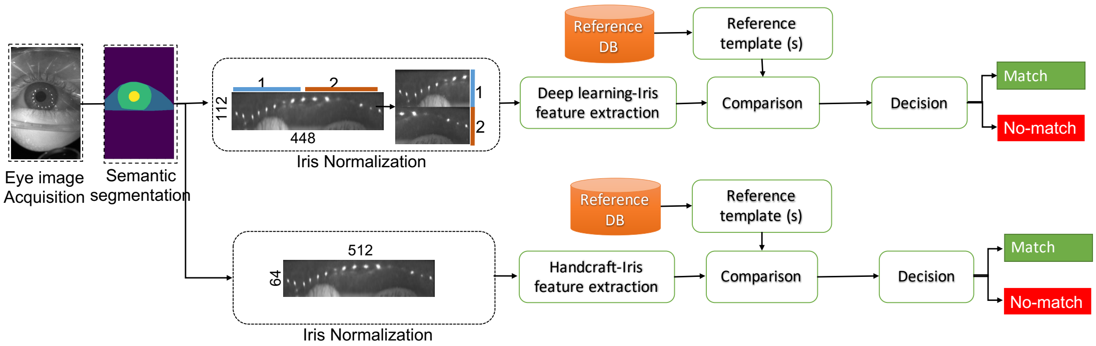


Figure 5.1.: An overview of iris recognition workflow from eye image acquisition to feature extraction. The image is segmented then the iris area is normalized and masked for non-iris areas. The mask and normalized image is processed by a feature extraction process. An extracted feature vector can be compared to another vector to perform identity verification.

5.3. Methodology

This chapter investigates the possibility of using HMD internal cameras for iris and periocular biometrics within HMDs where the ocular image is acquired using a camera mounted within the headset without requiring the user's collaboration. The performance of iris recognition methods depends on the accurate segmentation of the iris region. Therefore, this section starts by presenting a novel compact model for semantic segmentation of eye regions that aims at enabling deployment in low computationally powered devices. Then this section provides a comprehensive study on iris recognition approaches within AR/VR environment (HMD). The generic iris recognition pipeline consists of four main steps: a) iris image acquisition, b)segmentation and normalization, d) feature extraction, e) comparison and decision making. These steps are illustrated in Figure 5.1. This is followed by presenting multiple algorithms that we evaluate for periocular recognition. This aims at extending the biometric information source from the iris into the whole eye region to compare its feasibility to HMD biometrics in comparison to iris recognition alone. Finally, a method for generating realistic and identity-specific eye images from semantic segmentation labels is presented. The proposed synthesized solution consists of two-stage networks. The first stage network aims to generate a generic eye image corresponding to a given semantic segmentation label. The second stage network uses the output of the first stage network to generate identity-specific eye images. The identity preservation of the generated images is later evaluated within the presented iris and periocular recognition evaluation to gauge the suitability for biometric applications.

5.3.1. Ocular segmentation

The goal of the proposed segmentation solution is to create an accurate segmentation approach for a given eye region image despite appearance variations. The created model should be of small size (around or below 1MB) to enable application in the embedded environments, such as AR/VR applications [96]. In this section, three segmentation models are presented. The first is built to demonstrate the idea of multi-scale segmentation, while the second and third aim at maintaining (to various degrees) the performance of the first model while being significantly smaller (smaller number of learned parameters).

Multi-scale segmentation solutions (Eye-MS) This model aims at extracting more general information at lower image scales and thus minimizing the model size required to extract such information. It also processes the image at higher scales to analyze detailed image information. The presented model architecture is influenced by the cascaded refinement network introduced by Chen and Koltun [48] as an image synthesis tool. The proposed architecture is a convolutional neural network that consists of inter-connected refinement modules. Each module consists of only two convolutional layers (last module contains 3 convolutional layers), each followed by layer normalization [16] and a leaky rectified linear function (LReLU) [174]. The first module considers the lowest resolution space (40x25 in our model). This resolution is increased in the successor modules until the last module (640x400 in our case), matching the target image resolution. The input of each module is the output of the previous module bilinearly up-sampled to the proper input size of the current module, concatenated with the source image down-sampled using pixel area relation (area interpolation) to the proper input size of the current module. Our Eye-MS model uses 4 convolutions and a feature map (FM) of the size 256 for the first three modules and 128 for the last two modules. A summary of the network details is presented in Table 5.1. Our Eye-MS model has 6574k parameters and 457.3 GFLOPs, making it relatively smaller than conventional solutions such as the real-time ICNet (6680k) [293] and SegNet (29460k) [17]. However, such a model size might be quite large for embedded applications such as in HMD devices.

As we aim at producing an accurate segmentation model, however with a much smaller size, we point out that we are moving from a higher detailed space (captured eye image) to a space with lower variation (segmentation of four classes corresponding to ocular region, sclera, iris, and pupil). Thus, we neglect minor details in the image and focus on major changes across the image space. This can help us reduce the less important (for

Module	Input size	layer	Output size
Module 0	40x25x1	conv1 (kernel:(4,4), FM:256)	40x25x256
		conv2 (kernel:(4,4), FM:256)	
Module 1	80x50x257	conv3 (kernel:(4,4), FM:256)	80x50x256
		conv4 (kernel:(4,4), FM:256)	
Module 2	160x100x257	conv5 (kernel:(4,4), FM:256)	160x100x256
		conv6 (kernel:(4,4), FM:256)	
Module 3	320x200x257	conv7 (kernel:(4,4), FM:128)	320x200x128
		conv8 (kernel:(4,4), FM:128)	
Module 4	640x400x129	conv9 (kernel:(4,4), FM:128)	640x400x1
		conv10 (kernel:(4,4), FM:128)	
		conv11 (kernel:(1,1), FM:1)	

Table 5.1.: The detailed structure of the multi-scale segmentation network Eye-MS (6574k parameters). The input of module 0 is the source image down-sampled to its input size. The input of each of the modules 1, 2, 3, and 4 is the source image down-sampled (using area interpolation) to the proper input size of the current module concatenated with the output of the previous module bilinearly up-sampled to the proper input size of the current module. Each of the layers 1 to 10 is followed by LReLU activation and layer normalization (LN).

segmentation) learned parameters. We induce this notion by reducing the feature map size of the convolutional layers of the Eye-MS model. The proposed compact segmentation networks are described in the following paragraph.

Miniature multi-scale segmentation networks (Eye-MMS) In the first network, the feature maps size is set to 32 for the first two modules and 16 for the last three modules. In the second network, the feature maps are set to size to 64 for the first module and 32 for the last four modules. This reduction in the feature map size leads to a reduction in the size of the subsequent convolutional layers, therefore, a significant reduction in the number of learned parameters. The feature map produced by the convolutional layer represents how strongly the kernel responds to the layer input. Therefore, we design our architecture so that the feature map size goes from high to low as the network progress since we are moving from a higher detailed space (captured eye image) to a space with lower variation (segmentation of four classes), reducing the feature map can still maintain the lower variation of segmentation information while reducing the number of parameters significantly at the last layers of the network. At the same time, the larger feature map size can capture the high variations of the image at the initial layers. The feature map size at later layers of the network has more effect on the number of parameters as the input of these layers is up-sampled to higher resolutions. The first Eye-MMS model contains 80081 learned parameters and 6.2 GFLOPs. The second Eye-MMS model contains 216865 learned parameters and 23.09 GFLOPs. Thus, they will be noted as Eye-MMS80 and Eye-MMS216. The model architecture is provided in Table 5.3 for Eye-MMS80 and Table 5.2 for Eye-MMS216.

Module	Input size	layer	Output size
Module 0	40x25x1	conv1 (kernel(4,4), FM:64)	40x25x64
		conv2 (kernel(4,4), FM:64)	
Module 1	80x50x65	conv3 (kernel(4,4), FM:32)	80x50x32
		conv4 (kernel(4,4), FM:32)	
Module 2	160x100x33	conv5 (kernel(4,4), FM:32)	160x100x32
		conv6 (kernel(4,4), FM:32)	
Module 3	320x200x33	conv7 (fkernel(4,4), FM:32)	320x200x32
		conv8 (kernel(4,4), FM:32)	
Module 4	640x400x33	conv9 (kernel(4,4), FM:32)	640x400x1
		conv10 (kernel(4,4), FM:32)	
		conv11 (kernel(1,1), FM:1)	

Table 5.2.: The detailed structure of the miniature multi-scale segmentation network Eye-MMS216 (216K parameters). The input module 0 is the source image down-sampled to its input size. The input of each of the modules 1, 2, 3, and 4 is the source image down-sampled (using area interpolation) to the proper input size of the current module concatenated with the output of the previous module bilinearly up-sampled to the proper input size of the current module. Each of the layers 1 to 10 is followed by LReLU activation and layer normalization (LN).

All the networks (Eye-MS, Eye-MMS216, and Eye-MMS80) are trained using a $L2$ loss at the pixel level between the produced segmentation and the ground-truth label. The networks were trained with a batch size of one and a learning rate of 10e-4. The output layer produced a 2-D array of float numbers to enable a smooth learning conversion. The predicted segmentation is rounded to the nearest integer values to represent the discrete labels. To neglect any irregularly labeled pixels, we post-process the semantic segmentation by finding the largest contours around each of the considered labels then fitting a convex hull around these contours. These hulls represent the borders of each

Module	Input size	layer	Output size
Module 0	40x25x1	conv1(kernel(4,4), FM:32)	40x25x32
		conv2 (kernel(4,4), FM:32)	
Module 1	80x50x33	conv3 (kernel(4,4), FM:32)	80x50x32
		conv4(kernel(4,4), FM:32)	
Module 2	160x100x33	conv5 (kernel(4,4), FM:16)	160x100x16
		conv6 (kernel(4,4), FM:16)	
Module 3	320x200x17	conv7 (kernel(4,4), FM:16)	320x200x16
		conv8 (kernel(4,4), FM:16)	
Module 4	640x400x17	conv9 (kernel(4,4), FM:16)	640x400x1
		conv10 (kernel(4,4), FM:16),	
		conv11 (kernel(1,1), FM:1)	

Table 5.3.: The detailed structure of the miniature multi-scale segmentation network Eye-MMS80 (80K parameters). The input of module 0 is the source image down-sampled to its input size. The input of each of the modules 1, 2, 3, and 4 is the source image down-sampled (using area interpolation) to the proper input size of the current module concatenated with the output of the previous module bilinearly up-sampled to the proper input size of the current module. Each of the layers 1 to 10 is followed by LReLU activation and layer normalization (LN).

label.

5.3.2. Iris recognition

This section presents the iris normalization method and the different iris feature extraction solutions used for our benchmarking, along with their comparison methodology.

Iris normalization: To normalize the iris region, we start by defining a general circular border that contains the pupil and the iris. The pupil circular region is defined around its moment center. The circular region has the radius of the closest (from the center of the moment) iris labeled pixel. The circular border between the iris and the sclera is also defined as centered around the pupil center of the moment and has the radius of the distance between this center and the furthest (from the center of the moment) iris labeled pixel. The iris normalization follows the rubber sheet approach defined in [72], where the area is unrolled to perform a rectangular image. This normalized image is paired with a mask map containing the value zero for each pixel not labeled as iris in the semantic segmentation results and one for each pixel labeled as iris.

Iris feature extraction and comparison: In order to study the backward compatibility of the iris images captured using an AR/VR headset, we employ two feature extraction approaches- handcrafted and deep learning feature extraction approaches. For deep learning approach, we evaluate a deep representation extracted from iris modality using DenseNet-201 and DensNetBC-100 models [112]. For the handcrafted feature extraction approach, we employ three well-established and complementary iris feature extraction methods owing to the robustness and time-tested applicability for iris recognition in various constrained, and unconstrained iris recognition [39, 237, 145, 206, 217, 228]. The iris-codes in the first method are extracted using the classical Gabor features as proposed by Daugman [72] and we employ the generalized version of the same by employing 1D Log-Gabor features [181]. In the second approach, we employ Discrete Cosine Transform (DCT) coefficients of overlapped angular patches from normalized iris images to derive the iris-codes [190]. Further, in the third approach, we extract the iris-codes using the Cumulative-Sum-Based Change Analysis [142]. Each of the approaches is explained in brevity for the convenience of the reader.

1D Log-Gabor Filtered Iriscodes (LG): 1D Log-Gabor Filters [181] extend the idea of original 2D Gabor feature encoding proposed by Daugman [72] where they can extract the signal representation jointly in space and spatial frequency. The signal/iris image is decomposed using a quadrature pair of Gabor filters, with a real part specified by a cosine modulated by a Gaussian and an imaginary part specified by a sine modulated by a Gaussian filter. This decomposition yields the real and imaginary filters corresponding to even symmetric and odd symmetric components, respectively. Further, only the phase component is retained for encoding the discriminative information in the iris as the amplitude information, which mainly corresponds to illumination, is discarded. In order to represent the information in a compact form, the phase information is further encoded in four distinct levels in a four-quadrant principle. However, noting a zero DC component can be obtained for any bandwidth by using a Gabor filter which is Gaussian on a logarithmic scale, Masek et al. [181] proposed to employ the Log-Gabor filters and established the superior performance. In the lines of Daugman's approach[72], Masek et al.[181] also encoded the phase using four-quadrant quantization to be invariant to illumination. Further, the approach leads to a smaller template of the iris, which is space-efficient without compromising biometric performance [39, 237, 145, 206].

DCT Coefficients based Iriscodes (DCT): In the lines of the previous approach detailed above, DCT coefficients extract iris features using a similar principle, specifically in a non-semantic manner. The DCT based feature extraction is robust not only to illumination but also against focus-blur [39, 237, 145, 206] which is typically seen in unconstrained iris capture as in the case of AR/VR headset based iris capture. In this approach of

DCT based iris-code extraction feature vectors are derived from the zero crossings of the differences between 1D DCT coefficients calculated in rectangular image patches of the iris image [190]. Further, averaging across the width of the chosen patches with appropriate windowing smoothens the data, which helps in mitigating the effects of noise and other image artifacts such as motion blur, enabling to use of a 1D DCT to code along the length of each patch along [190]. Given such a formulation, the approach results in low-computational cost and an optimal noise-robust template/iris-code.

Cumulative-Sum Based Change Analysis for Iriscodes (CSBCA): Unlike the previous two approaches, Cumulative-Sum Based Change Analysis [142] employs the image directly without any specific filtering. This approach is based on dividing the entire image into a number of blocks with a size of $3\ pixels \times 10\ pixels$ representing a cell on a normalized iris image. Each cell is further represented by an average intensity of the gray value, and the cumulative sum over the 5 cells is used to obtain the binary code simply by thresholding the zero-crossing values [142]. The binary code obtained is robust against the illumination changes, while the rotation invariance is not well-accounted. Despite the simplicity, the approach has demonstrated superior performance in many applications of constrained, unconstrained iris recognition and, including the template protection [224, 225].

Feature C comparison using hamming distance (HD): Given the binary nature of the iris codes, we employ the Hamming Distance to measure the similarity between the iris codes. Further, we employ the segmentation masks to obtain robust comparison scores to account for the noisy part from the iris images that constitute the eye-lashes, eye-lids, specular reflections, and ambient reflection. For a reference iris-code ($\mathrm{IC}_{\mathrm{reference}}$) and probe iris-code ($\mathrm{IC}_{\mathrm{probe}}$) with corresponding iris segmentation masks represented by $\mathrm{mask}_{\mathrm{reference}}$ and $\mathrm{mask}_{\mathrm{probe}}$ respectively, the Hamming Distance score is computed as given below:

$$\mathrm{HD} = \frac{||(\mathrm{IC}_{\mathrm{reference}} \otimes \mathrm{IC}_{\mathrm{probe}}) \cap \mathrm{mask}_{\mathrm{reference}} \cap \mathrm{mask}_{\mathrm{probe}}||}{||\mathrm{mask}_{\mathrm{reference}} \cap \mathrm{mask}_{\mathrm{probe}}||}$$

Shifted hamming distance (SHD): As noted from the feature encoding methods employed in this work, the approaches do not account for factors such as rotation invariance and dilation/contraction of pupils. To account for any adverse impacts of rotation of iris region as observed in the iris images captured from AR/VR headsets, we also employ the shifted version of the Hamming Distance comparison as proposed in earlier works [224, 225]. Specifically, we shift the iris codes by 8 bits in both positive and negative directions to obtain the scores, following which the minimum of the scores is considered for reporting the performance.

Off-the-shelf CNN features: Along with the set of hand-crafted features, we investigate the performance of a state-of-the-art pre-trained CNN model on ImageNet [144] database for iris recognition. Specifically, we apply transfer learning on pre-trained DenseNet-201 [112] model. Our choice of DenseNet-201 was based on the promising reported accuracy for iris recognition [257]. DenseNet is a convolutional neural network designed for image classification to achieve low classification error rates while having fewer parameters than the ILSVRC 2015 winner, ResNet model [107]. The architecture is based on connecting each convolutional layer to every other layer in a feed-forward fashion. Thus, each layer ℓ^{th} receives collective knowledge from all preceding layers $x_0, x_1, ..., x_{\ell-1}$ and passes on its knowledge to all subsequent layers. Given that each layer produces k feature maps, the input feature map for ℓ^{th} layer is $k_0 + k \times (\ell - 1)$ where k_0 is the number of channels in the input layer and k refers to the growth rate of the network. The DenseNet-201 contains 201 layers (network depth) and the growth rate is $k = 32$. In order to apply transfer learning, we replace the classification layer (1000 neurons) with a new classification layer to fit the number of classes in our training dataset (95 identities). The normalized iris images are resized to 224×224 to match the input layer size as illustrated in Figure 5.1. During the training phase, we did not freeze any weights from the pre-trained layers. Rather, we fine-tuned the entire model with Softmax classifier on training data from OpenEDs database [96]. In the testing phase, we removed the classification layer from the model, and the rest of the network was used as a feature extractor. The features f are extracted from the last convolutional layer which is of dimension $7 \times 7 \times 1920$. The comparison between features extracted by this method uses Cosine distance.

Features learning using compact model- DenseNet-BC: Considering the limited computational resources and storage capacity of HMD devices, we explore feature learning of the DenseNet-BC [112] model. The employed DenseNet-BC model (depth= 100 and $k = 12$) contains only 0.8m trainable parameters and 1.8 GLOPs, compared to the 18.5m parameters and 8.6 GFLOPs of the DenseNet-201 model. Thus, it is more realistic to deploy on low computational power devices. DenseNet-BC has the same main architecture as DenseNet-201 but with 100 (instead of 201) layers and a growth rate of $k = 12$ (instead of 32). Besides, DenseNet-BC added bottleneck(B) and compression(C) layers to improve the computational efficiency of the DeneNet model. For implementation details of DenseNet and DenseNet-BC, one can refer to the original work in [112]. To adapt DenseNet-BC for iris images from HMD devices, we trained the model from scratch on training data of OpenEDs database [96] with Softmax classifier. Similar to the DenseNet-201 model, we resized the normalized iris to 224×224 and set the number of classes in the classification layer to 95 classes. The classification layer is removed during the testing phase, and the feature f is extracted from the last convolutional layer, which is of the dimension $14 \times 14 \times 342$. The comparison between features extracted by this method uses Cosine distance.

5.3.3. Periocular recognition

As illustrated in Figure 5.2, the periocular region captured from the HMD devices does not correspond to cooperative periocular captures. Under such highly uncooperative captures, the periocular recognition is expected to result in sub-optimal biometric performance as noted in earlier works for unconstrained periocular recognition[217]. Thus, we design a pipeline for periocular recognition by first aligning the images captured from the HMD device. With the alignment of periocular images, we account for multiple distortions arising out of the non-ideal gaze of the user. Following the alignment of the images, we extract the features using a selected set of feature extraction approaches, as detailed in this section. Further, we employ a simple distance metric to measure the similarity of the features to obtain the biometric performance.

Image alignment: In order to align periocular images, we first define the binarized mask of the eye image simply by using the coarse segmentation label of the eye image. The segmentation mask is derived on the basis of the pupil, the iris, and the sclera as one region (label value is 1) and the background as the second region (label value is 0). We further calculate the moment of the binary area with label values 1 and x, y coordinate of the moment center. By considering the moment center as the center of the eye, we transform the ocular image into a new image where the center of the new output image corresponds to the center of the eye (moment center). The aligned image is further resized to the original size (640×400 pixels) by padding with zero values. Samples of the aligned and not aligned periocular images are presented in Figure 5.2 for the sake of illustration.

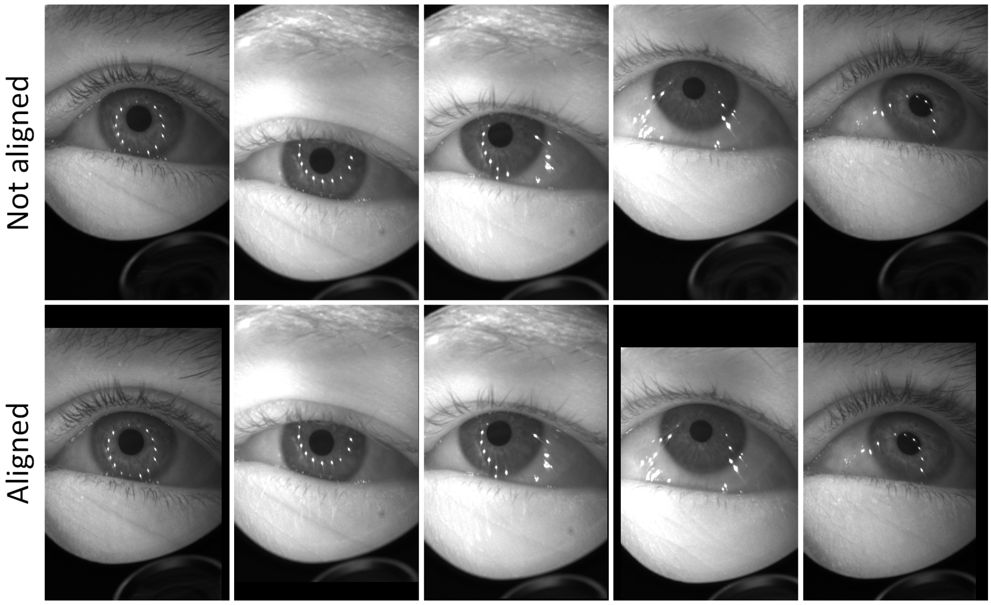


Figure 5.2.: Corresponding samples of images before (top) and after (bottom) alignment. The centralization of the pupil, iris and sclera combined region can be clearly noticed in the aligned images.

Feature extraction: Texture information is characterized by a set of patterns or local variations within images due to structural and intensity changes. The texture feature extraction banks on mathematical calculations on the pixel intensities of the images in a specific manner. The texture of the image also captures the description of gradients, orientation, local and global statistical features based on mathematical formulations in either local or global neighborhoods. Owing to such properties of extracting useful information from the images, a number of earlier works have employed texture features for many biometric modalities such as the face and periocular region

[203, 134, 278]. Motivated by these earlier works exemplifying the use of textural features, we employ state-of-the-art feature extraction schemes based on the texture descriptors to extract features from periocular images. Specifically, we explore two category of the texture descriptors - handcrafted texture [203, 134] and deep feature texture extraction approaches [112]. Given the aligned periocular image (I_p), we extract the texture features using the following approaches as detailed below:

Local Binary Pattern (LBP): LBP [203] descriptor works by thresholding intensity values of a pixel around a specified neighborhood in an image. The threshold is computed based on the intensity of central pixel intensity in a chosen window or selected pixels. The new binary value of the neighborhood is computed in a circular symmetric manner by interpolating the locations and checking against the value of the central pixel. If a particular value in the neighborhood is greater than the chosen central value, 1 is assigned and 0 otherwise. The set of values in a particular chosen block is encoded to form the compact pixel value f in the range of $0 - 255$ by using a simple binary to decimal conversion strategy as given by Eqn. 5.1.

$$f = \sum_{j=1}^{8} (Q(i) - Q(c)) * (2^{(j)}) \tag{5.1}$$

where Q represents the quantized values corresponding to central pixel $Q(c)$ and considered pixel $Q(i)$ in a neighbourhood. The set of f obtained from LBP is further used as the feature representation for the periocular image recognition.

Tree Local Binary Patterns (TreeLBP): Tree-Shaped Sampling Based Hybrid Multi-Scale Feature [171] is a variant of the LBP where a number of different configurations are employed to extract noise-resistant features are extracted. While in LBP, multiple radius can be explored to extract the features, different pixel radius $r = 2, 6, 8$ configurations and Tree-Shaped Sampling radius $R = 2, 6, 8$ can be used in TreeLBP to extract the multi-scale features.

$$f = \sum_{r,R} \sum_{j=1}^{8} (Q(i) - Q(c)) * (2^{(j)}) \tag{5.2}$$

where Q represents the quantized values corresponding to central pixel $Q(c)$ and considered pixel $Q(i)$ in a neighbourhood for a radius of r and sampling radius R. The set of f obtained from TreeLBP is further used as the feature representation for the periocular image recognition.

Binarized Statistical Independent Features: BSIF is another texture extraction method similar to LBP [134]. BSIF automatically learns a fixed set of filters from a set of natural images, unlike the hand-configured approach in LBP. The BSIF based technique consists of applying learned textural filters to obtain a statistically meaningful representation of the image data, which enables efficient information encoding using binary quantization. A set of filters of patch size $l \times l$ are learned using natural images, and independent component analysis (ICA) [134] where the patch size l is defined as :

$$l = (2 * n + 1)$$

such that n ranges from $\{1, 2...8\}$. The set of pre-learned filters from natural images are used to extract the texture features from periocular images. If a periocular image is represented using

$I(x,y)$ and the filter is represented by $H_i(x,y)$ where i represents the basis of the filter, the linear response of the filter s_i can be given as [134]:

$$s_i = \sum_{x,y} I(x,y) H_i(x,y) \tag{5.3}$$

where x, y represents the dimension of image and filter. The response is further binarized based on the obtained response value. If the linear filter response is greater than the threshold, a binarized value of 1 is assigned as given by [134]:

$$b_i = \begin{cases} 1, & \text{if } s_i > 0 \\ 0, & \text{otherwise} \end{cases} \tag{5.4}$$

The obtained responses b are encoded to form the compact pixel value f in the range of $0 - 255$ by using binary to decimal conversion as provided by Eqn. 5.5. The set of f is used as the feature representation for the perioclar image recognition.

$$f = \sum_{j=1}^{k} b_j \times 2^{(j-1)}; \tag{5.5}$$

where $k = 5, 6, \ldots 12$. We employ a BSIF filter of size 9×9 with $k = 8$ for extracting features in this work.

Histogram of Oriented Gradients (HOG): The idea behind HOG is to model the local object appearance and shape through characterizing the local intensity gradients or edge directions. HOG features are extracted by dividing the image into small spatial regions (also referred to as cells), and for each cell, a set of a local l-D histogram of gradient directions or edge orientations over the pixels of the window are accumulated. After normalization, the combined histogram entries from the entire image constitute the final feature vector of HOG. We employ a cell size of 32×32 after experimenting with a range of cell sizes by considering the entire image to extract features in this work.

Deep feature extraction: We evaluated the deep feature representation extracted from the periocular region by utilizing the same models employed for deep iris recognition- DenseNet-201 and DenseNetBC-100. The models details are described in Section 5.3.2. To adapt these models for periocular modality, we fine-tuned DenseNet-201 with periocular images from our training dataset and trained the compact model DenseNetBC-100 from scratch. For both models, we modified the number of classes in the classification layer to fit the number of identities in our training dataset (95 identities) and resized the periocular images to 224×224 to match the model input layer size. The classification layer is removed during the testing phase, and the features f are extracted from the last layer.

Periocular feature comparison: The set of handcrafted features extracted from the periocular images is compared to obtain the biometric performance using a simple distance metric of χ^2

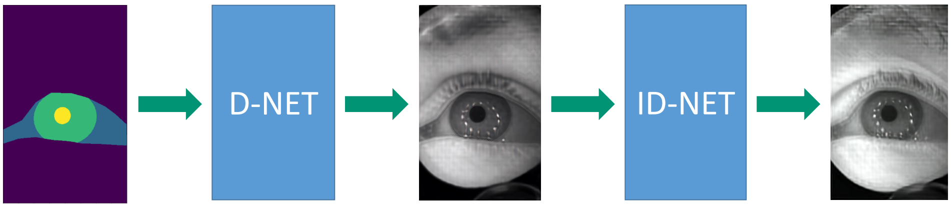


Figure 5.3.: Eye image generation approach using D-Net and ID-Net models. One can notice the identity-related information induced by the ID-Net on the output of the more generic output of the D-Net.

distance metric in order to align this work with earlier works in this direction of periocular recognition. The features extracted by deep feature extraction are compared using Cosine distance.

5.3.4. Computational cost

The computational cost of the deep learning approaches depends on the number of trainable parameters, computational complexity (FLOPs), and inference latency. DenseNetBC-100 contains 0.801 million (m) of trainable parameters and 1.8 GLOPs. The inference time for a single image is 3.25 milliseconds (ms). DenseNet-201 contains 18.275M trainable parameters and has 8.6 GFLOPs. The inference time for a single image is 6.91ms. Eye-MMS216 contains 0.216m of trainable parameters and the inference time is 4ms. All evaluations are performed using Tensorflow framework (Version 1.14) running on Linux OS with Intel(R) Xeon(R) Gold 6130 CPU 2.10GHz processor. Each extracted feature is stored as a four-byte floating-point resulting in templates of 376.2 kilobytes (KB) and 268.1 KB for DenseNet-201, and DenseNetBC-100, respectively. The computational efficiency of the deep learning methods is identical for both iris and periocular verification.

The chosen handcrafted iris verification methods are computationally efficient and optimally represented for storage purposes. Each of the handcrafted feature extraction is completed within 3ms, and the templates result in 915 bytes, 1022 bytes, and 336 bytes for 1D Log-Gabor, DCT Coefficient iris-code, and Cumulative-Sum-Based Change Analysis based iris-code, respectively, when stored in lossless Portable Graphics Format (png) format. The comparison of masked iris codes using Hamming Distance takes around 2ms, while the shifted version of the same takes around 6ms.

The computational cost of the methods employed for periocular feature extraction are 0.02 seconds, 0.52 seconds, and 0.072 seconds for HOG, TreeLBP, and BSIF, respectively, and the comparison of two sets of features for each of them takes 0.02, 0.019, and 0.002 seconds. Further, the template sizes correspond to 135Kb, 8.97Kb, and 16 bytes for HOG, TreeLBP, and BSIF.

5.3.5. Identity preserving image generation

The proposed image generation approach aims at generating realistic and identity-specific eye images from semantic segmentation for potential future applications, such as generating large-scale training data or generating presentation attack samples. The identity information is generally available for a limited set of images of a specific identity. Our proposed solution is designed as a two-stage network. The first network transfers the semantic label to the eye-domain (a more realistic eye image) that maintains the structure of the label, which we refer to as *D-Net*. The second network (*ID-Net*) induces the identity information by transforming the output of the D-Net into an image containing identity-specific details. Together, these networks compose our D-ID-Net solution for identity preserving image generation as shown in Figure 5.3.

Both D-Net and ID-Net share the same architecture (as detailed in Table 5.4) but differ in the training protocol. Our architecture is a convolutional neural network that consists of interconnected refinement modules. Each module consists of only two convolutional layers (the last module contains 3 convolutional layers), each followed by layer normalization [16] and a LReLU with non-linearity [174]. The first module considers the lowest resolution space (5x3 in our model). This resolution is increased in the successor modules until the last module (640x400 in our case), matching the target image resolution. The input of each module is the output of the previous module up-sampled to the proper input size of the current module, concatenated with the source image down-sampled to the proper input size of the current module. Our architecture uses 3x3 convolutions and a feature map (FM) of the size 512 for the first five modules and 256 for the last three modules. The presented solution is influenced by the cascaded refinement network introduced by Chen and Koltun [48]. This model aims at extracting more general information at lower image scales and processes the image at higher scales to analyze detailed image information.

The D-Net input (source) images are semantic segmentation of the eye-regions (pupil, iris, sclera, and background) that we represent in the source image with the corresponding pixel values (20, 90, 160, and 230) to avoid extreme (0 or 256) values in the training process. The target is the corresponding real images to the semantic labels (for all identities in training). To achieve the style transformation to the natural image space, we use the contextual loss (CL) [184] as a training loss. This is accompanied with a pixel-level Euclidean (L2) loss to smooth the training convergence.

The CL is calculated between image embeddings extracted by a pre-trained VGG19 [241] network trained on the ImageNet database [77]. The CL is given as:

$$\begin{aligned} CL_{CX}(t, g, l_1, l_2) = &-log(CX(\Phi^{l_1}(g), \Phi^{l_1}(t))) \\ &-log(CX(\Phi^{l_2}(g), \Phi^{l_2}(t))) \end{aligned} \tag{5.6}$$

where t and g are the target and generated images respectively. CX is the rotation and scale invariant contextual similarity [184]. Φ is a perceptual network which is VGG19 in our work. $\Phi^{l_1}(x)$ and $\Phi^{l_2}(x)$ are the embedding vectors extracted from the image x at layers `conv3_2`, and `conv4_2`, respectively. The L2 loss is given as:

$$L2(t, g) = \frac{1}{I_X I_Y} \Sigma_{i=1}^{I_X} \Sigma_{j=1}^{I_Y} (t_{ij} - g_{ij})^2, \tag{5.7}$$

Module	Input size	layer	Output size
Module 0	5x3x3	conv1 (kernel (3,3), FM:512) conv2 (kernel (3,3), FM:512)	5x3x512
Module 1	10x6x515	conv3 (kernel (3,3), FM:512) conv4 (kernel (3,3), FM:512)	10x6x512
Module 2	20x12x515	conv5 (kernel (3,3), FM:512) conv6 (kernel (3,3), FM:512)	20x12x512
Module 3	40x25x515	conv7 (kernel (3,3), FM:512) conv8 (kernel (3,3), FM:512)	40x25x512
Module 4	80x50x515	conv9 (kernel (3,3), FM:512) conv10 (kernel (3,3), FM:512)	80x50x512
Module 5	160x100x515	conv11 (kernel (3,3), FM:256) conv12 (kernel (3,3), FM:256)	160x100x256
Module 6	320x200x259	conv13 (kernel (3,3), FM:256) conv14 (kernel (3,3), FM:256)	320x200x256
Module 7	640x400x259	conv15 (kernel (3,3), FM:256) conv16 (kernel (3,3), FM:256) conv17 (kernel (1,1), FM:3)	640x400x3

Table 5.4.: The detailed structure of the D-Net and ID-Net. Both networks share the same structure with different training strategies. The input of each of the 7 modules is the source image and the output of the previous module (not for Module 0), down-sampled and up-sampled subsequently to the input size of the current module. Each of the layers 1 to 17 is followed by LReLU activation and layer normalization (LN).

where t_{ij} and g_{ij} are the ground-truth target and generated pixel values respectively at position (i, j), with value range of $[0, 255]$. I_X and I_Y are the height and width of the generated image (and ground-truth) in pixels. The total loss (TL_{D-Net}) function of the D-Net is:

$$TL_{D-Net}(t, g) = CL_{CX}(t, g, l_1, l_2) + \lambda * L2(t, g), \tag{5.8}$$

with $\lambda = 1e - 4$.

The second stage ID-Net uses the outputs of the D-Net as source images and corresponding real images as target images. Every ID-Net is trained separately for each identity. The source of the ID-Net already has the properties of the natural eye image and requires the induction of identity information. We only use the contextual loss as defined in Equation 5.6 between the source (D-Net output) and the target (real images) to achieve this. As a result, passing a semantic label through

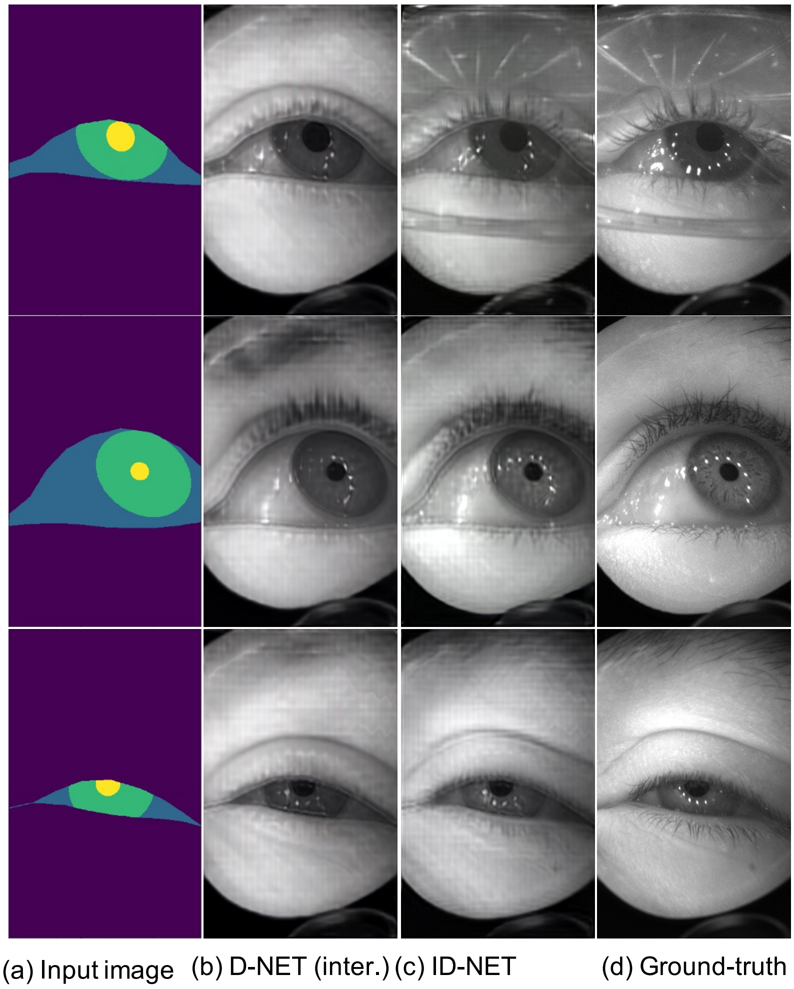
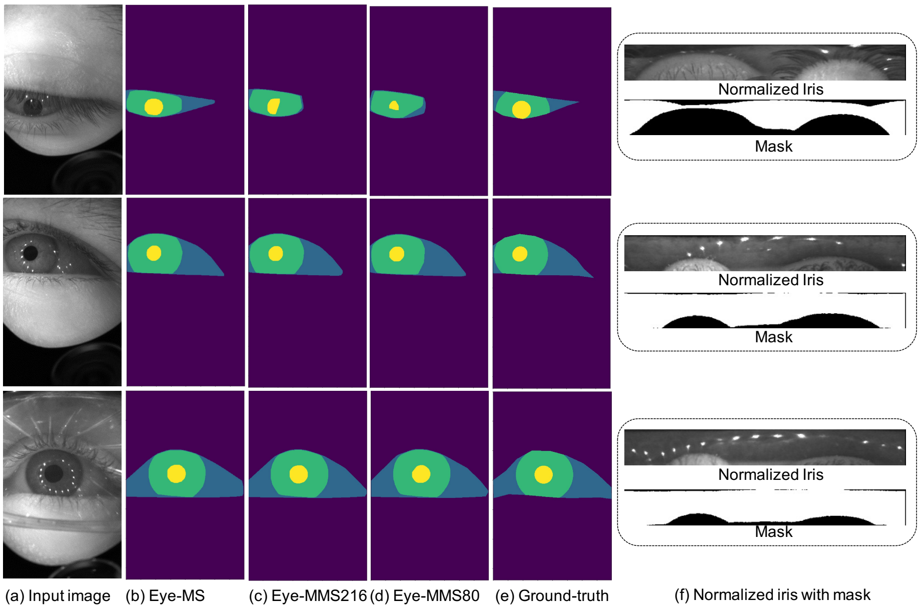

(a) Input image (b) D-NET (inter.) (c) ID-NET (d) Ground-truth

Figure 5.4.: Samples of the input semantic segmentation, the intermediate generation by the D-Net, the final generated image by the D-ID-Net, and the ground-truth images. These samples are selected from different identities to have eyeglasses with reflection, extreme gaze directions, and eyes opening variations. The identity-specific details can be noticed when moving from the D-Net output to the final D-ID-Net output.

the generic D-Net, then the D-Net output through identity-specific (ID-Net) results in the targeted realistic and identity-specific eye image as demonstrated in Figure 5.4.



(a) Input image (b) Eye-MS (c) Eye-MMS216 (d) Eye-MMS80 (e) Ground-truth (f) Normalized iris with mask

Figure 5.5.: (a) Samples of input images, (b) segmentation produced by our Eye-MS, (c) segmentation produced by our Eye-MMS216, (d) segmentation produced by our Eye-MMS80, (e) the ground-truth segmentation, and (f) normalized iris with using based on selected segmentation model Eye-MMS216. The images are selected to represent different challenging conditions such as relatively closed eye, non-central gaze, and an image with glasses. The images from top to bottom achieved an Iris Mask Ratio (IMR) of 0.47, 0.78, 0.85 respectively, which reflects in the visible proportion of the iris.

5.4. Experimental setup

This section presents the implementation details, evaluation benchmarks and evaluation settings used in this chapter.

5.4.1. Database

This work uses the OpenEDs dataset acquired using a virtual-reality HMD with two synchronized eye-facing cameras. OpenEDs dataset contains three different datasets, generation, semantic segmentation, and sequence dataset.

Segmentation Dataset: The semantic segmentation dataset includes 12759 images of 152 individuals with a pixel resolution of 640x400. The data is split into 8916 images for training, 2403 images for validation, and 1440 images for test as described in [96]. The test split is not available publicly yet and thus, is not used in this work. Since the semantic segmentation labels are available only for training and validation splits, the segmentation model is trained on training split and evaluated on validation split.

	Eye-MMS80		Eye-MMS216		Eye-MS	
Region	15 epochs	40 epochs	15 epochs	40 epochs	15 epochs	40 epochs
IoU(BG)	0.9857	0.9860	0.9874	0.9898	0.9896	0.9905
IoU(Sclera)	0.8084	0.8201	0.8249	0.8542	0.8519	0.8628
IoU(Iris)	0.9223	0.9273	0.9289	0.9412	0.9408	0.9443
IoU(Pupil)	0.9105	0.9159	0.9181	0.9302	0.9276	0.9346
IoU-mean	0.9068	0.9125	0.9148	0.9289	0.9275	0.9330

Table 5.5.: The performance, given as IoU, on different ocular regions and a mean IoU to represent general performance of our proposed models, Eye-MS, Eye-MMS216 and Eye-MMS80, at two different stages of the training process. It is noticed that despite the significant reduction in the model size the performance is only slightly effected. BG refers to the background region.

Image synthesis dataset: The generation data includes 152 subjects and 12759 images of 640x400 pixel resolution. The data is split into identity-disjoint training, validation, and testing splits as described in [96]. As the test split is not publicly available yet and we have not used it in this work. Training the D-Net in this work uses the segmentation subset of the training split, containing 8916 images (all with labels) of 95 identities. The D-Net training used the labels as a source and eye images as a target. The ID-Net aims at inducing identity information from a set of identity-known images. An ID-Net was trained for each of the 28 identities in the validation split. This used the generation subset of the validation split, containing 2048 images per identity. To generate the segmentation labels of these images (required for training), the images were segmented using the Eye-MS segmentation network. The D-Net processed these labels to produce the ID-Net source images (with the initial images as targets). Evaluating the generation performance used the semantic segmentation subset of the validation split with its labels. This contains the same identities used to train the ID-Nets, but different images. The segmentation labels are used as input to the D-ID-Net, and the corresponding images are used as a ground-truth (GT) for the evaluation. Each of the 28 validation identities contained between 28 to 138 labels/images (avg. of 86 per identity). The resulting images are referred to as the generated images (Gen).

Iris recognition dataset: The deep learning methods are trained on the training split of the semantic segmentation dataset. The training split includes 8916 images of 95 identities that are disjoint from the validation set used to report the verification results. We randomly selected a subset of 190 images (two per identity) of the training split to validate the model for early stopping to avoid over-fitting the model during training. The handcrafted iris recognition methods used in this work do not require training. Deep learning and handcrafted feature extraction methods are

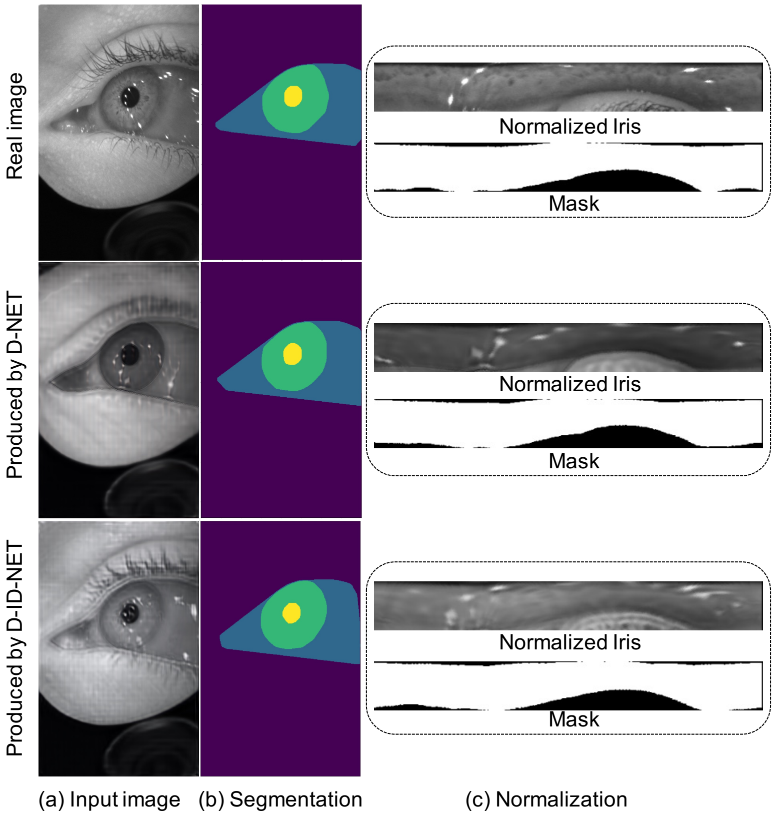


Figure 5.6.: An illustration of normalized iris of different eye images (original, D-NET generated, and ID-NET generated of the same identity as the original image on the top). Samples of an input image (a), segmentation produced by our Eye-MMS216 (b), and the normalized iris images. The first image is a real image selected from the validation dataset. The second image is produced by D-NET from the real images segmentation label. The third image is produced from ID-NET using D-NET output and the corresponding ID-NET model (of the identity of the original image on the top). One can notice similar iris properties between the original iris of the same identity and the ID-NET generated iris, but not the D-NET generated image as expected.

tested and reported on three datasets: a) The first dataset is the validation split of the semantic segmentation dataset (TDS1). The validation data contains 2403 images of 28 identities. Each of the 28 validation identities contained between 37 and 128 images captured consecutively and on average 86 per identity. b) The second dataset is the synthesized images using D-NET. The segmentation labels from the validation split of the semantic segmentation dataset (TDS1) are used as input to D-NET, and the output of D-NET is used as a second testing dataset (TDS2). c) The third dataset is the synthesized images using ID-NET, where the previous output images of D-NET are passed to corresponding ID-NET, and the synthesized output images are used as a third testing dataset (TDS3). The image set of each identity is split into reference images and probe images. For the three test datasets, the reference images are selected only from the validation split of the semantic segmentation dataset. The first ten images for each identity are considered as a reference pool. One of these ten images is considered as the reference image based on the proportion of the visible iris region, as will be defined later in this section. The consequent five images are neglected to create a time gap between the reference and probe images. All the consequent images for each identity are considered as probe images.

	Generation quality		
Metric	RMSE	PSNR	SSIM
D-Net	9.621	15.486	0.591
D-ID-Net	**7.235**	**23.347**	**0.678**

Table 5.6.: The generation performance given as RMSE, PSNR, and SSIM with respect to the ground-truth. The improvement induced by the our two-stage D-ID-Net is demonstrated in all metrics.

Preiocular recognition dataset: The deep learning periocular recognition methods are trained on the same iris recognition training dataset (prior to segmenting the iris). As the handcrafted feature extraction approaches do not require training, the periocular recognition methods are also tested on the same iris recognition testing datasets (prior to segmenting the iris), including TDS1, TDS2, and TDS3.

5.4.2. Segmentation

Eye-MS, Eye-MMS216, and Eye-MMS80 models were trained on the training split, containing 8916 pairs of eye images and corresponding ground-truth labels. The results are post-processed as described in Section 5.3.1. The segmentation performance is evaluated here as the Intersection over Union (IoU) of each of the four segmented regions i (pupil, iris, sclera, background) between the predicted segmentation (P) and the ground-truth label (L) and is given by

$$IoU_i = \frac{L_i \cap P_i}{L_i \cup P_i}. \tag{5.9}$$

To get an overall performance measure, we also report the IoU_{mean}, the unweighted mean of the four IoU_i values. The results are reported for the model Eye-MS and the miniature model Eye-MMS80 and Eye-MMS216 after 15 epochs (reached loss: 0.012074 for Eye-MMS80, 0.011019 for Eye-MMS216, and 0.008486 for Eye-MS) of training and after 40 epochs of training (reached loss: 0.010709 for Eye-MMS80, 0.008320 for Eye-MMS216 and 0.007085 for Eye-MS).

5.4.3. Image generation

The D-Net and each ID-Net training used a batch size of one, a learning rate of 1e-4, and ran for 10 and 20 epochs for D-Net and ID-Nets, respectively. The generation performance is measured as the similarity of the generated image to the ground-truth by calculating the Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{I_X I_Y} \Sigma_{i=1}^{I_X} \Sigma_{j=1}^{I_Y} (t_{ij} - g_{ij})^2}, \tag{5.10}$$

A lower RMSE indicates a high similarity to the ground-truth. We also measure two generation quality metrics, the Peak Signal to Noise Ratio (PSNR) and the Mean Structural Similarity Index (SSIM). PSNR indicates the ratio of the maximum pixel intensity to the power of the distortion. SSIM [272] combines local image structure, luminance, and contrast into a single local quality score. A higher value of SSIM and PSNR indicates that the generated image is more similar to the ground-truth.

To analyze the degree of identity preservation, we evaluate the data under a verification scenario using handcrafted and CNN-based features. This is described in detail in the periocular recognition methodology in Section 5.3.3 and experimental setup in Section 5.4.6.

5.4.4. Iris selection

To analyze the non-cooperative nature of the image acquisition, we analyze the results based on the amount of visible iris in the image. In order to achieve this, we introduce the IMR as a ratio of the actual iris area (mask neglected) size to the whole normalized image size. A higher IMR indicates that a larger proportion of the iris is visible in the image. The IMR is used to select the reference image from the reference images pool for each identity, i.e., the image with the highest IMR is selected from each reference pool to be the reference image.

The non-collaborative nature of the process has further motivated us to analyze the iris selection strategy for the probe images, i.e., selecting iris images to be used for verification from the series of iris images. We, therefore, threshold the IMR value to neglect images with low IMR. We analyze the verification performance at different IMR thresholds. However, this thresholding will create a time (samples) gap in the verification process, which is significant if the verification is performed in a continuous nature. Therefore, we analyze the amount of gap (measured by the number of images under threshold between accepted images) in the consequent probe samples introduced by different IMR thresholds. We also present a histogram of the IMR values in the probe data, the generated probe samples using D-NET, the generated probe samples using ID-NET, the reference pool data, and the selected reference samples. Eight IMR thresholds (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6 ,0.7) are selected to perform our analyses.

5.4.5. Iris verification

We evaluate the verification performance using three handcrafted feature extraction approaches and two deep learning models. The DenseNet-201 and DenseNetBC-100 models are evaluated with cosine-distance for comparison. The LG and the DCT approaches are evaluated with HD and SHD distance for comparison, resulting in four evaluation settings, noted as LG-HD, LG-SHD, DCT-HD, and DCT-SHD. A fifth evaluation setup uses the CSBCA features with the HD distance for comparison, as the nature of the feature extraction does not benefit from the more computationally expensive SHD distance calculation. Each of these settings is evaluated with each of the IMR thresholds computed on the probe data. The verification performance is illustrated and reported as ROC, Area under the curve (AUC), and FMR at fixed FNMR (FMR10, the lowest FNMR for FMR$\leq$10%). Moreover, a general indication of the performance is reported as the Equal Error Rate (ERR), which is the common value of FMR and FNMR at the decision threshold where they are equal. The results are reported on the three test datasets, validation split of the semantic segmentation dataset, synthesized images using D-NET, and ID-NET.

5.4.6. Periocular verification

The verification performance of periocular recognition is evaluated using four handcrafted feature extraction approaches and two deep learning models. The HOG, LBP, TreeLBP, and BSIF as handcrafted feature extraction methods, and they are evaluated with χ^2 distance (as recommended in [75]) for comparison. The DenseNet-201 and DenseNetBC-100 deep learning models are evaluated with cosine-distance for comparison. The verification performance is illustrated and reported using common biometric evaluation metric as ROC curves, AUC, FMR at fixed FNMR (FMR10, the lowest FNMR for FMR$\leq$10%), and Equal Error Rate (ERR). Similar to the iris verification, the result is reported on the three test datasets, validation split of the semantic segmentation dataset, synthesized images using D-NET, and ID-NET to also prove the identity preserving nature of the proposed generation approach.

5.4.7. Deep learning models training setup

The investigated models are trained using SGD optimizer with Nesterov momentum 0.9, batch size of 16, and initial learning rate of $\gamma = 0.1$. The learning rate is reduced by a factor of 0.1 when the accuracy on the validation dataset does not improve by a value of 0.1 for five consequent epochs. The early-stopping patience parameter is set to 10. When the models are trained on normalized iris, the training process of DenseNet-201 and DenseNetBC-100 models stopped after 22 and 26 epochs, respectively, and after 16 and 23 epochs when they are trained on the periocular region.

5.5. Result

5.5.1. Iris segmentation

Figure 5.5 shows samples of the validation images along with the segmentation result obtained by our Eye-MS, Eye-MMS216, and Eye-MMS80 models (all three models trained for 40 epochs), the segmentation ground-truth, and normalized iris with a mask using segmentation produced by Eye-MMS216 model. One can notice the relatively accurate segmentation of the iris region even when the eye is relatively closed, with non-central gaze, or with eyeglasses. To point out the visual relationship between the iris images and their achieved IMR, the sample images in Figure 5.5 are normalized using segmentation produced by the Eye-MMS216 model and achieved the following IMR values from top to bottom: 0.47, 0.78, 0.82. This corresponds to the eye image in the top being relatively closed and thus contains a smaller visible portion of the iris.

Table 5.5 lists the performances, given as IoU, for each individual region (label) and as a mean over the four regions. This performance comparison is given for Eye-MS, Eye-MMS216, and Eye-MMS80 and at two different points of the training process. It is noticeable from Table 5.5, and in all experimental settings that the IoU(background) achieves the highest value. The reason for this is based on the relatively large area of the background and thus the lower probable ratio of non-intersection to the union area, between the ground truth and prediction. The IoU(iris) and IoU(pupil) achieve closer values, with the IoU(iris) slightly overperforming the later. The IoU(sclera) scores are significantly lower than the other eye regions. The reason for this is the potential confusion between the sclera and background, especially with images containing highly reflective glasses.

Table 5.5 also shows that increasing the training to fourteen epochs improves the performance of all models. The Eye-MMS216 generally performs better than Eye-MMS80 and only slightly worse than the Eye-MS model while having less than 1/30 of Eye-MS model parameters. The Eye-MMS80 is the smallest model where the number of its parameters is less than 1/80 of Eye-MS model parameters and 1/3 of Eye-MMS216 model parameters. However, both models, Eye-MMS80 and Eye-MMS216, have less than 1 MB model size, and they can be run on devices with limited memory footprint.

IMR	DCT-SHD		LG-SHD		CSBCA		DCT-HD		LG-HD		DenseNet-201		DenseNetBC-100	
	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10
IMR 0.0	0.3438	0.5359	0.3474	0.5435	0.3952	0.7174	0.3817	0.6739	0.3675	0.5870	**0.1063**	**0.1112**	0.1451	0.1769
IMR 0.1	0.3438	0.5359	0.3474	0.5435	0.3952	0.7174	0.3817	0.6739	0.3675	0.5870	**0.1042**	**0.1074**	0.1377	0.1627
IMR 0.2	0.3438	0.5359	0.3474	0.5435	0.3952	0.7174	0.3817	0.6739	0.3675	0.5870	**0.1043**	**0.1069**	0.1372	0.1622
IMR 0.3	0.3436	0.5355	0.3475	0.5441	0.3949	0.7168	0.3814	0.6727	0.3673	0.5866	**0.1038**	**0.1064**	0.1369	0.1618
IMR 0.4	0.3404	0.5291	0.3443	0.5383	0.3933	0.7123	0.3789	0.6701	0.3654	0.5826	**0.0983**	**0.0961**	0.1311	0.1527
IMR 0.5	0.3375	0.5210	0.3415	0.5285	0.3887	0.7022	0.3778	0.6633	0.3639	0.5770	**0.0899**	**0.0793**	0.1184	0.1312
IMR 0.6	0.3341	0.5071	0.3335	0.5142	0.3731	0.6972	0.3744	0.6529	0.3571	0.5643	**0.0783**	**0.0548**	0.0943	0.0937
IMR 0.7	0.3113	0.4735	0.3178	0.4743	0.3434	0.6487	0.3644	0.6267	0.3480	0.5387	**0.0635**	**0.0309**	0.0725	0.0566

Table 5.7.: The iris verification performance in terms of EER and FMR10 for the different experimental settings and different IMR thresholds. The lowest FMR10 and EER are in bold for each IMR threshold. One can notice the lower errors achieved by the DenseNet-201 and DenseNetBC-100 settings.

5.5.2. Identity preserving image generation

Figure 5.4 presents samples of the input semantic labels, the intermediate results by the D-Net, our D-ID-Net generated images, and the ground-truth. One can notice the high similarity in the results provided by the D-ID-Net and the ground-truth under various conditions. The intermediate D-Net output does not contain detailed information that might relate to an identity but has general eye characteristics and the structure of the semantic label. The introduction of these details by the ID-Net is noticeable when comparing the D-Net output with the final D-ID-Net results. Figure 5.6 shows an example of the segmentation and normalization alongside masking for the real sample image, sample image produced by D-NET, and sample image produced by ID-NET. It can be clearly noticed the similar identity-related iris properties between the original iris of the same identity and the ID-NET generated iris, but not the D-NET generated image as expected.

The quality of the D-ID-Net generated images is presented as RMSE, PSNR, and SSIM values in Table 5.6 all of which are calculated with the ground-truth images as a reference. The D-ID-Net achieved, as desired, relatively high PSNR and SSIM and low RMSE values, indicating high similarity to the ground-truth images. The intermediate result of the D-net is also analyzed and scored worse values of all metrics, indicating the importance of the identity information induced by the second stage network (ID-Net). It can also be noted that the results of iris and periocular recognition (presented later in this section) utilizing these generated images allow to concretely confirm the identity preservation within D-ID-Net.

IMR	DCT-SHD		LG-SHD		CSBCA		DCT-HD		LG-HD		DenseNet-201		DenseNetBC-100	
	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10
IMR 0.0	0.4970	0.9046	0.4925	0.9077	0.4665	0.8385	0.4991	0.9072	0.4854	0.8956	0.4499	0.8535	**0.4366**	**0.8282**
IMR 0.1	0.4970	0.9046	0.4925	0.9077	0.4665	0.8385	0.4991	0.9072	0.4854	0.8956	0.4527	0.8640	**0.4357**	**0.8262**
IMR 0.2	0.4970	0.9046	0.4925	0.9077	0.4665	0.8385	0.4991	0.9072	0.4854	0.8956	0.4524	0.8639	**0.4355**	**0.8262**
IMR 0.3	0.4972	0.9034	0.4923	0.9084	0.4664	0.8381	0.4992	0.9074	0.4852	0.8963	0.4527	0.8638	**0.4356**	**0.8261**
IMR 0.4	0.4959	0.9030	0.4912	0.9082	0.4644	0.8406	0.5001	0.9066	0.4856	0.8973	0.4525	0.8646	**0.4337**	**0.8242**
IMR 0.5	0.4959	0.9024	0.4909	0.9088	0.4643	0.8342	0.5024	0.9088	0.4861	0.8975	0.4523	0.8643	**0.4322**	**0.8241**
IMR 0.6	0.4995	0.9054	0.4919	0.9072	0.4583	0.8363	0.5041	0.9096	0.4846	0.9025	0.4547	0.8675	**0.4252**	**0.8201**
IMR 0.7	0.5032	0.9111	0.4952	0.9032	0.4525	0.8333	0.5119	0.9079	0.4881	0.8968	0.4581	0.8740	**0.4166**	**0.8166**

Table 5.8.: The iris verification performance for the different experimental settings and different IMR thresholds calculated from different feature extraction methods on D-NET synthesized images. As expected, the verification results are random as the D-NET images contains no identity information.

5.5.3. Iris selection

Figure 5.7.a presents the histogram of the IMR values scores by the images in the reference pool. A set of samples scored lower than 0.4 IMR, indicating a low proportion of visible iris. When the sample with the highest IMR is selected for each validation identity, the lowest IMR value becomes above 0.7, as seen in Figure 5.7.b. On the other hand, Figures 5.7.c, 5.7.d, 5.7.e show the histogram of the IMR values achieved by the probe samples, D-NET probe samples, and ID-NET probe samples, respectively. One can notice that the probe samples contained some images where the iris was not visible at all, i.e., close eyes. The plots also show that most probe samples had an IMR between 0.6 and 0.9, where the mean IMR value is 0.710 for the probe samples, 0.709 for the probe samples produced by ID-NET, and 0.713 for the probe samples produced by D-NET. This points out the high similarity from the IMR perspective between the generated image of both networks and the original images. Figures 5.7.f shows the histogram of the IMR values achieved by the training dataset. Considering that the testing and training data are acquired under the same none-collaborative capturing condition, it can be noticed that some of the training samples also contained partially closed eyes, and most of them had IMR values between 0.6 and 0.9. An indication of the IMR visible interpretation is illustrated in Figure 5.5 where the images from top to bottom achieved an IMR of 0.47, 0.78 and 0.85, respectively.

When samples with low IMR values are neglected, this will produce a sample gap (SG) between consecutive frames. Having a large SG might affect the applicability to continuous authentication or, if a large SG is allowed, it will give an attacker the time frame to gain access. Therefore, it is important to study the amount and frequency of gaps induced by neglecting captures with low IMR. To do that, we present a thorough analyses in Figure 5.8 for probe samples images, Figure 5.9 for probe samples produced by D-NET, and Figure 5.10 for probe samples produced by ID-NET. The figures present the occurrences of different gaps in each IMR thresholding setup and for each testing dataset. In these figures, SG0 indicates the occurrences of two consecutive captures having no neglected captures between them, SG1 indicates the occurrences of two consecutive captures having one neglected capture between them, and so on for SG2, SG3, etc., while the SG>10 indicates the total number of occurrences of SG equal to 10 or more. The number on each block in the figure represents the occurrences of gaps at a certain IMR threshold. For example, Figure 5.8 shows that at certain 0.6 IMR threshold results in 143 single sample gaps (SG1), 38 two consecutive gaps, 11 three consecutive gaps, etc. Figures 5.8, 5.9 and 5.10 thus show that increasing the IMR threshold might result in an unwanted sample gaps. However, an IMR threshold of 0.5 will only result in a few sample gaps above three and a maximum gap of five. Original and generated samples produced, as expected, similar sample gaps distributions.

5.5.4. Iris recognition

The verification performances of the different evaluated algorithms applied on the three evaluated datasets are presented as ROC curves in Figure 5.11, Figure 5.12 and Figure 5.13 alongside EER and FMR10 values in Table 5.7, Table 5.8 ,and Table 5.9. In all presented results, the references are selected from the real images. In the result shown in Figure 5.11 and Table 5.7, the probes are selected from the real images, while in the Figure 5.12 and Table 5.8 the probes are selected from

IMR	DCT-SHD		LG-SHD		CSBCA		DCT-HD		LG-HD		DenseNet-201		DenseNetBC-100	
	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10	EER	FMR10
IMR 0.0	0.4553	0.8519	0.4489	0.8454	0.4159	0.7645	0.4664	0.8555	0.4528	0.8434	**0.2210**	**0.4110**	0.2261	0.3595
IMR 0.1	0.4553	0.8519	0.4489	0.8454	0.4159	0.7645	0.4664	0.8555	0.4528	0.8434	**0.2201**	**0.4094**	0.2255	0.3573
IMR 0.2	0.4553	0.8519	0.4489	0.8454	0.4159	0.7645	0.4664	0.8555	0.4528	0.8434	**0.2199**	**0.4096**	0.2257	0.3569
IMR 0.3	0.4552	0.8522	0.4491	0.8456	0.4157	0.7641	0.4663	0.8557	0.4527	0.8435	**0.2199**	**0.4103**	0.2257	0.3566
IMR 0.4	0.4534	0.8507	0.4490	0.8419	0.4127	0.7703	0.4660	0.8548	0.4521	0.8424	**0.2171**	**0.4042**	0.2194	0.3502
IMR 0.5	0.4537	0.8542	0.4494	0.8397	0.4048	0.7618	0.4647	0.8558	0.4530	0.8419	0.2125	0.3905	**0.2103**	**0.3346**
IMR 0.6	0.4503	0.8565	0.4460	0.8357	0.3901	0.7542	0.4625	0.8518	0.4508	0.8333	0.2135	0.3792	**0.1973**	**0.3144**
IMR 0.6	0.4532	0.8511	0.4412	0.8308	0.3739	0.7201	0.4589	0.8495	0.4412	0.8234	0.2176	0.3872	**0.1834**	**0.2921**

Table 5.9.: The iris verification performance for the different experimental settings and different IMR thresholds calculated from different feature extraction methods on ID-NET synthesized images.

generated images by D-NET and from ID-NET in Figure 5.13 and Table 5.9. Each of the Figures 5.11.a-g, 5.12.a-g, and 5.13.a-g shows the ROC achieved when the processed probe captures with an IMR below a certain threshold are neglected. It can be clearly noticed that the highest performance of all the evaluated algorithms is achieved when the probes are selected from the real images as shown in the Figures 5.11.a-g.

As expected, when the probes are from generated images by D-NET where the output does not contain identity information, the results of all the evaluated algorithms are almost random as expected and shown in Figure 5.12. Same conclusion can be made by looking at ERR (around 0.5) in the Table 5.8. However, when the identity information is introduced to the synthesized images (ID-NET), the performance of all the evaluated algorithms is improved as shown in Figure 5.13 in comparison to the case when the probes are from generated images by D-NET.

Method	TDS1		TDS2 (D-NET)		TDS3 (ID-NET)	
	ERR	FMR10	ERR	FMR10	ERR	FMR10
BSIF	0.3477	0.8452	0.4635	0.8645	0.4012	0.8285
LBP	0.3558	0.8471	0.4991	0.8891	0.4872	0.8896
TreeLBP	0.3127	0.8246	0.4387	0.8514	0.3284	0.8085
HOG	0.2851	0.4279	0.3688	0.7119	0.2975	0.4630
DenseNet-201	**0.0586**	**0.0298**	0.3719	0.7266	0.0824	0.0581
DenseNetBC-100	0.1233	0.1390	0.4088	0.7261	0.1189	0.1309

Table 5.10.: The periocular verification performance for different experimental settings and different periocular feature extraction methods on ID-NET synthesized images, D-NET synthesized images, and the original real validation images.

Further, one can notice clearly that neglecting captures with low IMR enhances the performance of all the evaluated algorithms. This can be clearly explained by the complete information included in an iris image with high IMR and thus the accurate verification result. The same conclusion can be made when looking at the EER, and FMR10 values in Table 5.7, where increasing the IMR threshold reduces the different error rates consistently.

Each of the Figures 5.11.h and 5.13.h compares the performance of the different recognition algorithms under the most strict IMR threshold (IMR>0.7). When the probes are from real images, it is noticeable from this set of ROC curves and the error rate in Table 5.7 that the deep learning model significantly better than the rest of the algorithms. Moreover, as expected, the verification performance of the handcrafted approaches was lower than the deep learning approaches. It is noticeable that the LG-SHD and DCT-SHD perform significantly better than the rest of the algorithms, and using the SHD distance to compare iris codes generally achieves better verification performance than the simple HD distance that does not consider any rotational shifts. When the probes are from generated images by ID-NET, the CSBCA performs better than the other algorithms.

In general, when probe images are from real images, the best EER achieved with deep learning methods was 10.63% when no probe images are neglected and 6.35% when iris images with IMR lower than 0.7 are neglected. These results indicate that the employed deep learning models were able to learn the discriminative features in the iris images using a small number of training samples (8916 training images). Moreover, it can be clearly noticed that the compact model, DenseNetBC-100 achieved a close verification performance to the DenseNet-201 model. In the case where the probe images are produced by ID-NET, the best EER achieved with deep learning methods was 22.10% when no probe images are neglected and 18.34% when iris images with IMR lower than 0.7 are neglected. For handcrafted feature approaches evaluated on probe images from real images, the best EER achieved was 34.38% when no probe images are neglected and 31.13% when iris images with IMR lower than 0.7 are neglected. In the case where the probe images are produced by ID-NET, the best EER achieved was 41.59% when no probe images are neglected and 37.39% when iris images with IMR lower than 0.7 are neglected. Such EER value is considered high, which motivates future works on developing application-specific solutions for iris recognition with HMD considering the computational limitations. These results also show significant detail preservation of the generated images by the ID-NET, even at the detailed iris level.

5.5.5. Periocular recognition

The verification performances of periocular recognition using different feature extraction approaches are presented in Figure 5.16a-f, Figure 5.15a-f, and Figure 5.14a-f. In all presented results, the references are selected from the real images. In the first experiment, the probes are selected from the real image pool as shown in Figure 5.16. In the second and third experiments, the probe are selected from the synthesised images as shown in Figure 5.15 for D-NET probe images and Figure 5.14 for ID-NET probe images. It can be clearly noticed that the highest performance is achieved by DenseNet-201 when the probes are selected from the real images as shown in Figure 5.16a-f. The result of intermediate D-NET achieved, as expected, is an almost random verification decision. However, the results of HOG, DenseNet-201 and DenseNetBC-100 approaches were slightly better than random. This can be explained by the fact that the deep learning feature extraction and HOG features, unlike the locally calculated handcrafted features, analyze the image globally and thus partially describe the global shape. Such results point out that part of the identity information of the periocular is embedded in its global shape and not only in the detailed local information. In comparison to iris recognition on images generated by the D-Net (Table 5.8), one can notice that the deeply learned features on D-Net generated iris images produced close to random decisions, as the iris shape is rather consistent over different identities. The improvement in the performance can be noticed when the probes are from generated images by ID-NET as shown in the Figure 5.14 in comparison to the case where the probes are from generated images by D-NET. The results clearly indicate the success of ID-NET in generating identity-specific ocular images, which can further be explored for large-scale training data generation or presentation attack generation.

Table 5.10 illustrates the performance of the periocular recognition with EER and FMR10. In the case where the probes are selected from the real images, the best achieved EER was 5.86% with the fine-tuned DenseNet-201, followed by DenseNetBC-100 with 12.33% EER. When the probes are from generated images by ID-NET, the performance is slightly degraded, and the best achieved EER was 8.24% by DenseNet-201, which indicates the high level of the identity-preservation. On the other hand, when the probes are from generated images by D-NET, all methods achieve random verification decision, and the best achieved EER was 36.88% by HOG. The overall periocular verification results provide a baseline and motivate further work on ocular biometrics within HMD, especially with many emerging VR/AR applications.

5.6. Discussion

With the detailed investigations and analyses, this chapter provided answers to RQ3.1, RQ3.2 and RQ3.3.

- With the set of extensive experiments conducted in this chapter, it can be concluded that the iris and ocular images from HMD devices can be used for biometric recognition. This work demonstrated that the current state-of-art approaches could be used for both iris and periocular regions. Specifically, deep feature extraction methods achieved promising verification performances for both iris and periocular images, even when the computationally light DenseNetBC-100 model is used, where the best achieved EER was 6.35% for iris verification and 5.86% for periocular verification,

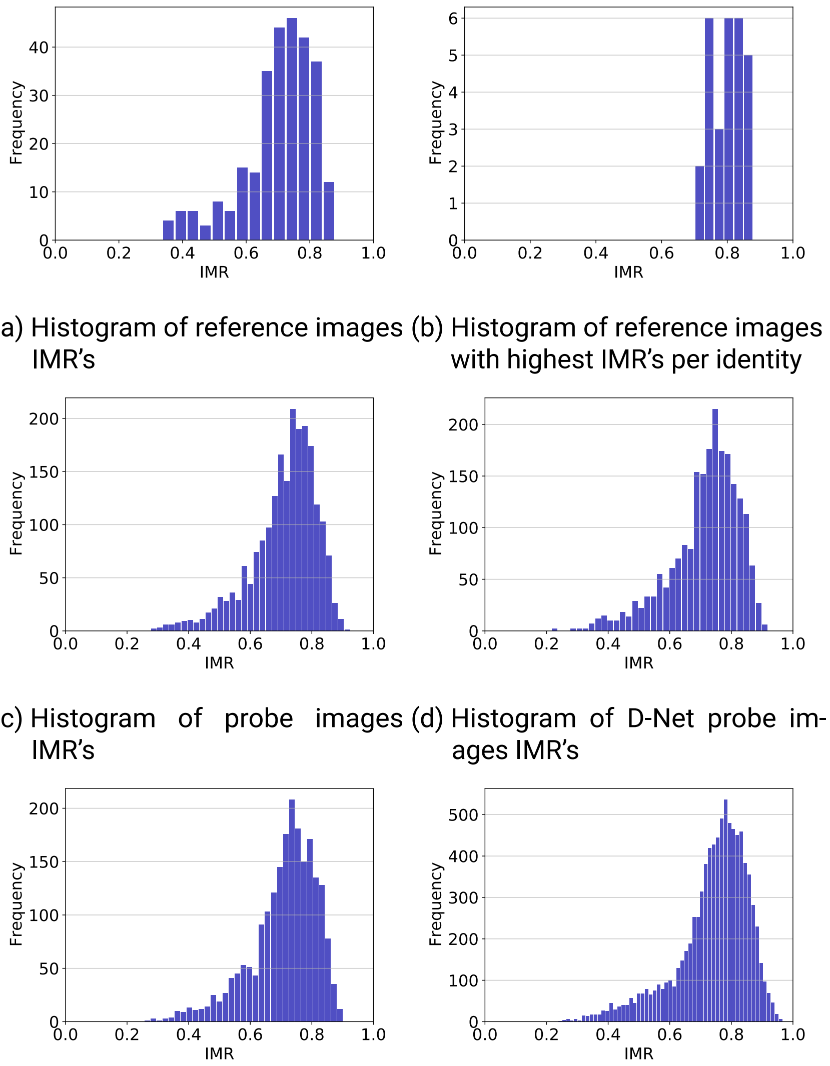
- With the proposed compact models having 216k (Eye-MMS261) and 80k (Eye-MMS80) parameters, this work has demonstrated the possibility of using compact models for near accurate segmentation of sub-optimal data captured from HMD devices. The achieved IoU(mean) was 90.68% by Eye-MMS80 and 91.48% by Eye-MMS261.

- This work presented a two-stages approach for generating identity-preserving ocular images directly from semantic segmentation. The realistic nature of generated images has been established through various quality metrics evaluated on the generated images. To empirically complement the observation from quality metrics and prove the identity-preserving nature of the generation, this work also provided and compared the biometric performance obtained on generated images for both iris and periocular.

5.7. Summary

Motivated by the recent developments in VR/AR technologies and driven by the security needs within the applications of this technology, this chapter investigates the possibility of using captures from the internal HMD camera for biometric verification with considerations to low computation resources. New applications supported by AR/VR technology may need to access critical information or resources, which requires maintaining a high level of trust in the user's identity. Such a goal should be achieved by a verification process that does not require the intentional collaboration of the user. To conduct the investigations in this chapter, it introduced and investigated several iris and periocular verification methodologies on the targeted use-case scenario and provided a detailed evaluation that includes a comprehensive study on iris sample selection and its effect on verification accuracy, providing an answer to RQ3.1. Moreover, a lightweight segmentation model that minimizes the computational need of larger networks while maintaining a very close accuracy is proposed and evaluated, providing an answer to RQ3.2. Variations of Eye-MMS solution performed very competitively in various segmentation challenges [267, 265]. This chapter additionally presented an identity-preserving synthetic ocular image (captured within HMD) generation approach, which produces identity-specific images from an arbitrary ocular semantic label. This chapter has also demonstrated that the images produced by the proposed approach maintain to a large degree the identity, in both iris and periocular modalities, in comparison to the original real data, which provides an answer to RQ3.3.

The next chapter will conclude the thesis and provide a brief outlook on future work.



(a) Histogram of reference images IMR’s

(b) Histogram of reference images with highest IMR’s per identity

(c) Histogram of probe images IMR’s

(d) Histogram of D-Net probe images IMR’s

(e) Histogram of ID-Net probe images IMR’s

(f) Histogram of training images IMR’s

Figure 5.7.: Plots (a) to (f) show the histogram of the IMR values scored by the iris images from the reference pool, the selected references, all probe samples, all D-Net probe samples, all ID-Net probe samples, and the training dataset, respectively.

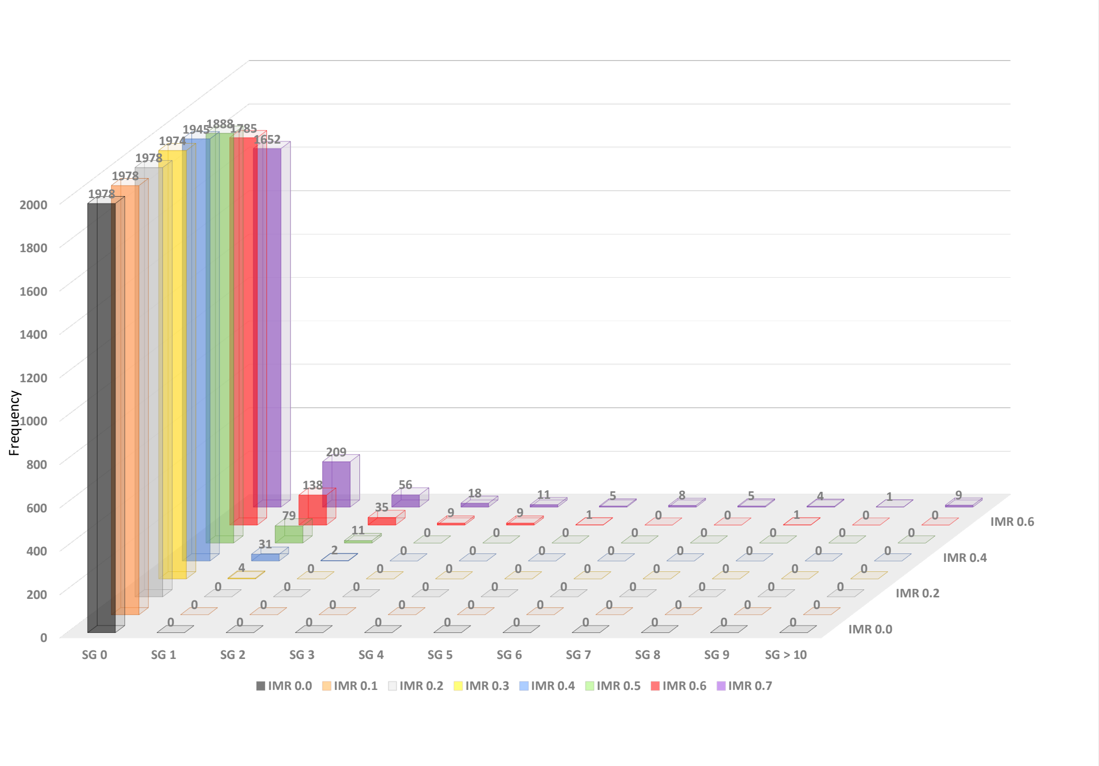


Figure 5.8.: Statistics of the size and amount of sequence sample gaps (SG) induced by neglecting probe images that score an IMR value below a certain threshold. Each block represents the occurrences of a certain gap with a certain IMR threshold setting. e.g., SG0 is the case where two consecutive captures do not have any neglected capture between them, and SG1 is the case where two consecutive captures do have one neglected capture between them. With a higher IMR threshold, the higher sample gaps occur more often.

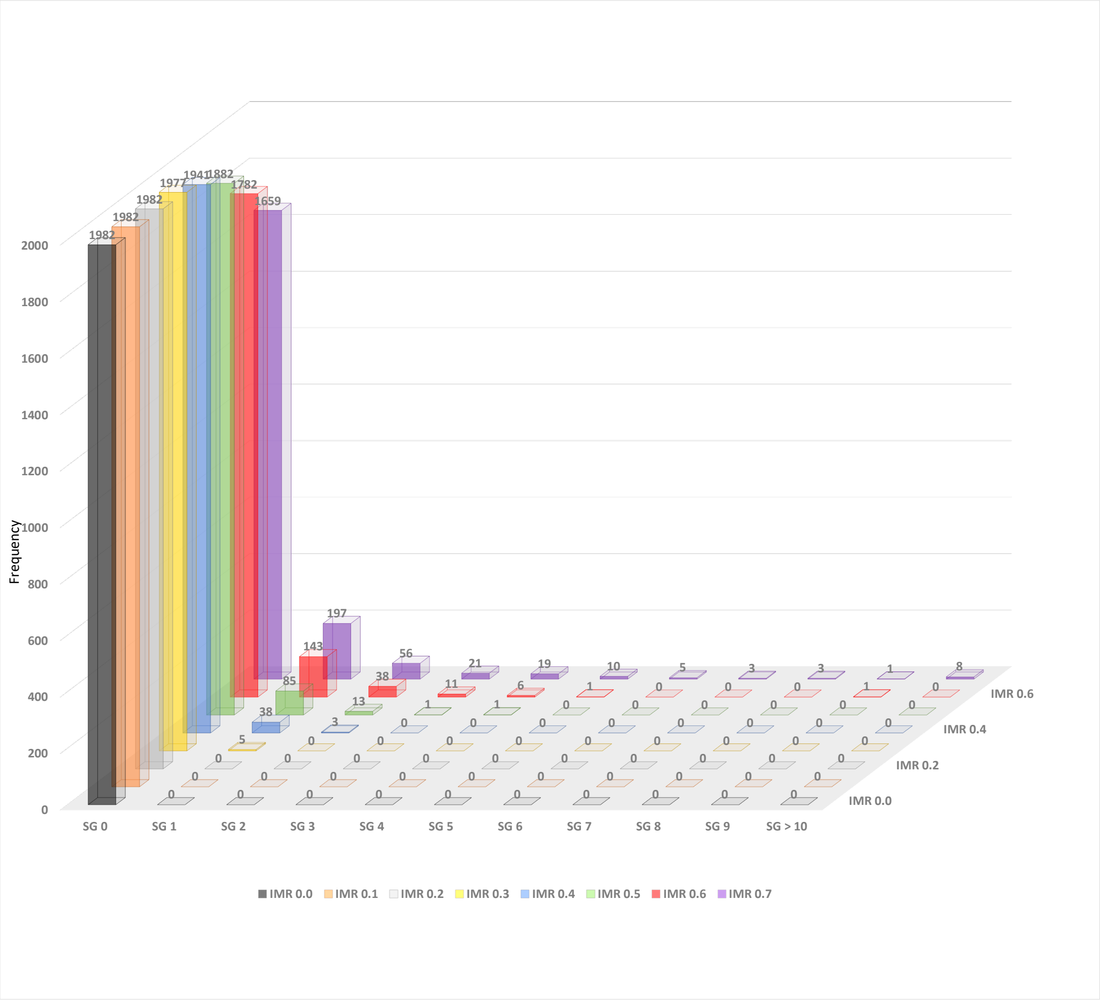


Figure 5.9.: Similar to Figure 5.8, this figure shows the size and amount of sequence sample gaps (SG) induced by neglecting probe images that scores an IMR value below a certain threshold. The probe images are selected from D-NET model.

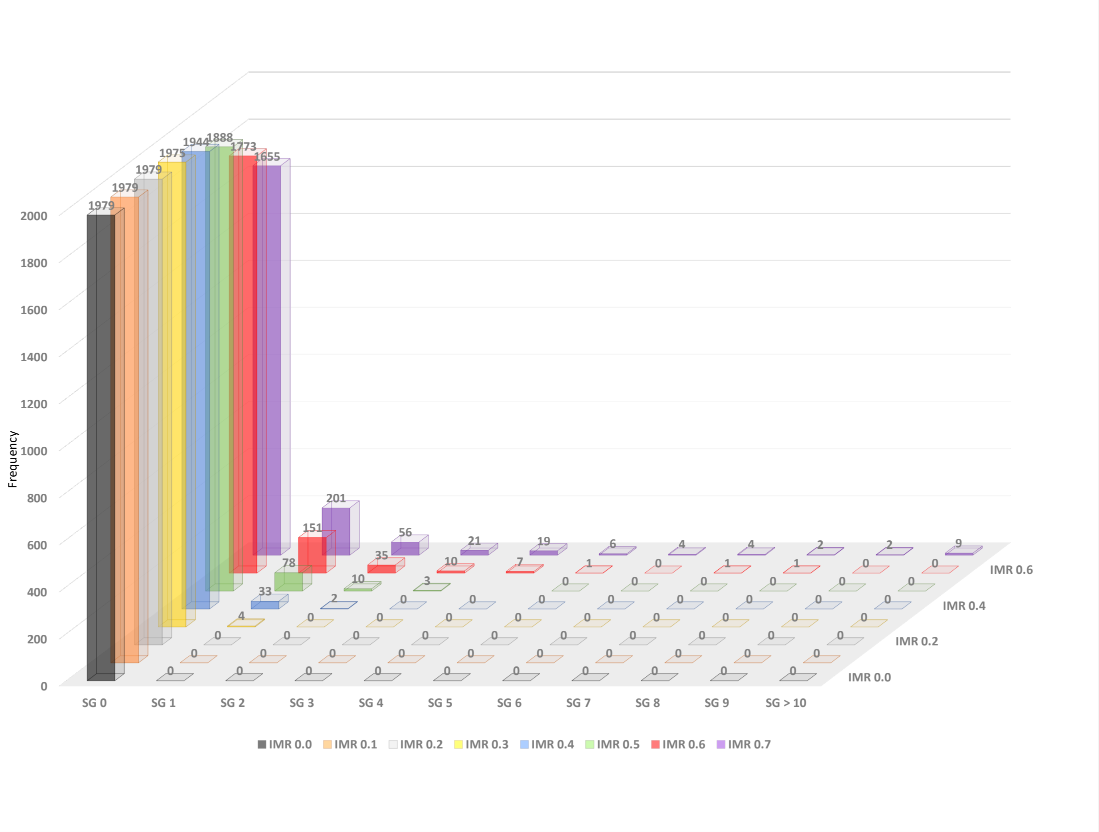
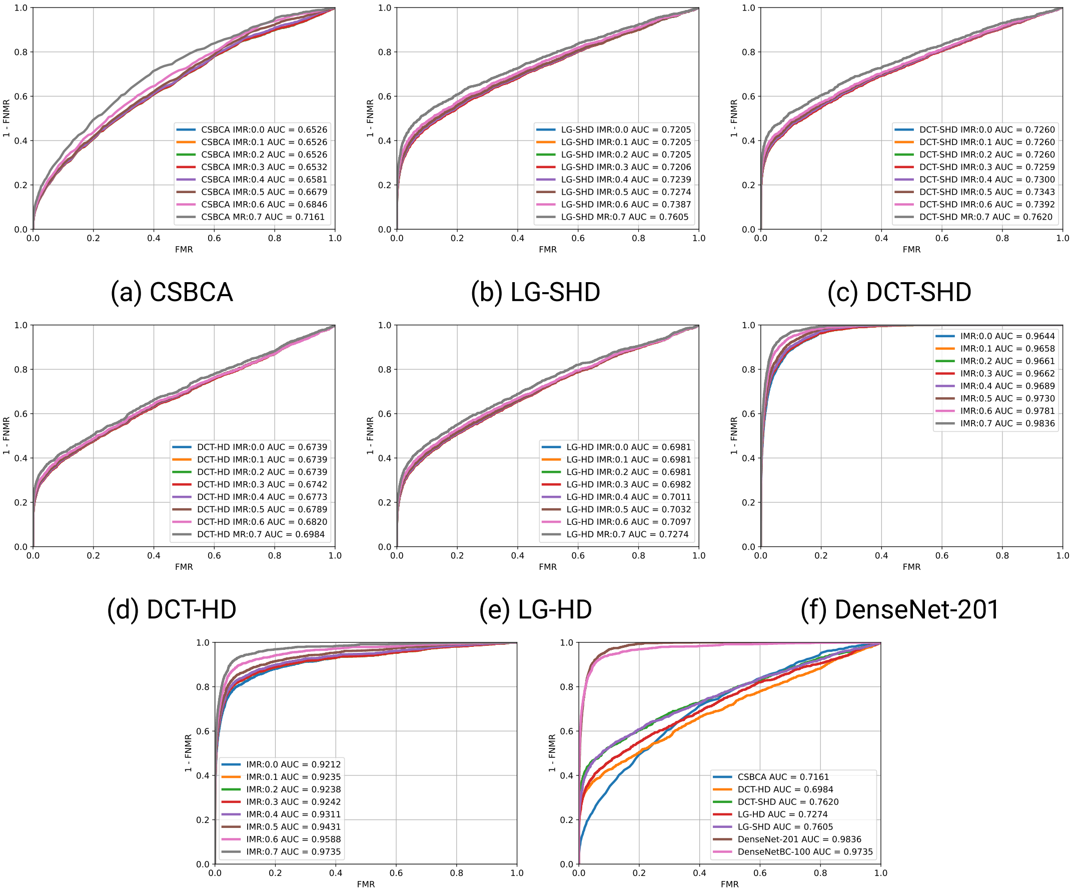


Figure 5.10.: Similar to Figure 5.8, this figure shows the size and amount of sequence sample gaps (SG) induced by neglecting probe images that scores an IMR value below a certain threshold. The probe images are selected from ID-NET model.



(a) CSBCA

(b) LG-SHD

(c) DCT-SHD

(d) DCT-HD

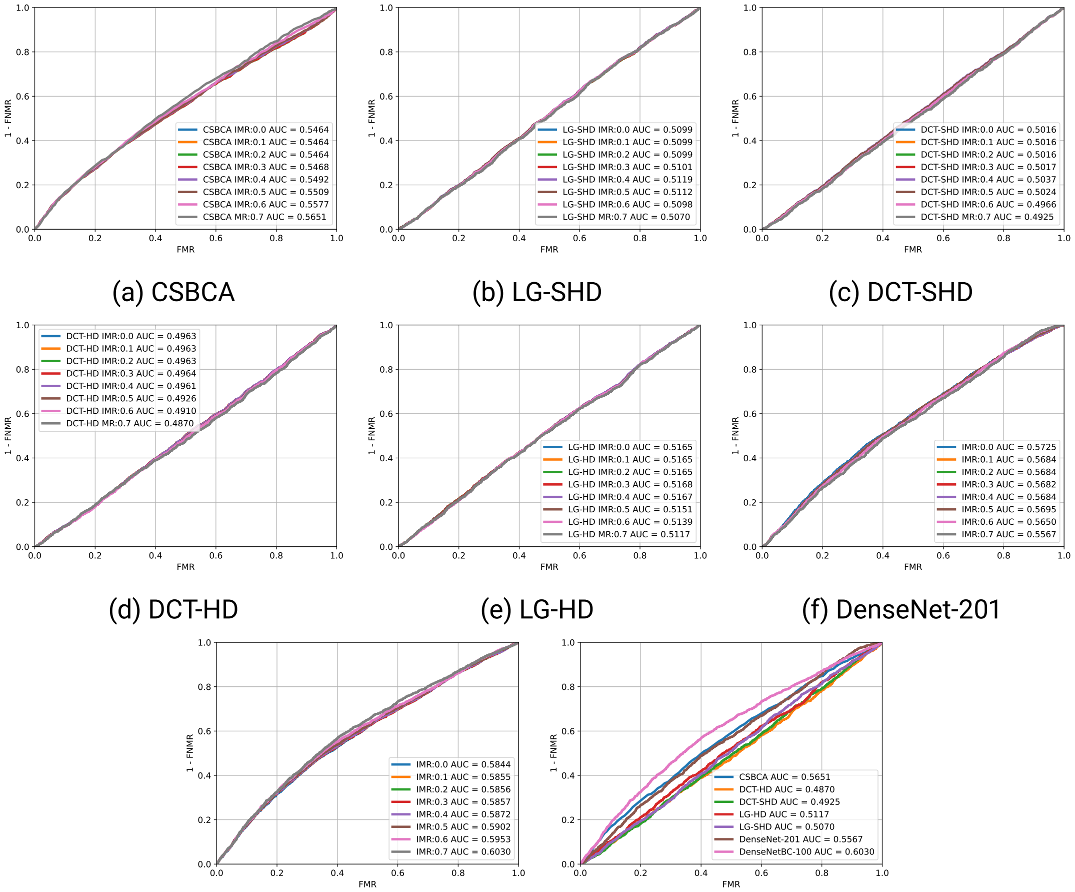
(e) LG-HD

(f) DenseNet-201

(g) DenseNetBC-100

(h) All-IMR 0.7

Figure 5.11.: The achieved ROC curves for the different experimental settings of iris recognition and different IMR thresholds. Each of the plots (a) to (g) shows the ROC curves achieved by one of the benchmarking settings with different levels of IMR rejection threshold. Plot (h) shows a comparison of the different algorithms at the most strict IMR threshold (0.7). Notice the increased performance when rejecting samples with low IMR values.



(a) CSBCA

(b) LG-SHD

(c) DCT-SHD

(d) DCT-HD

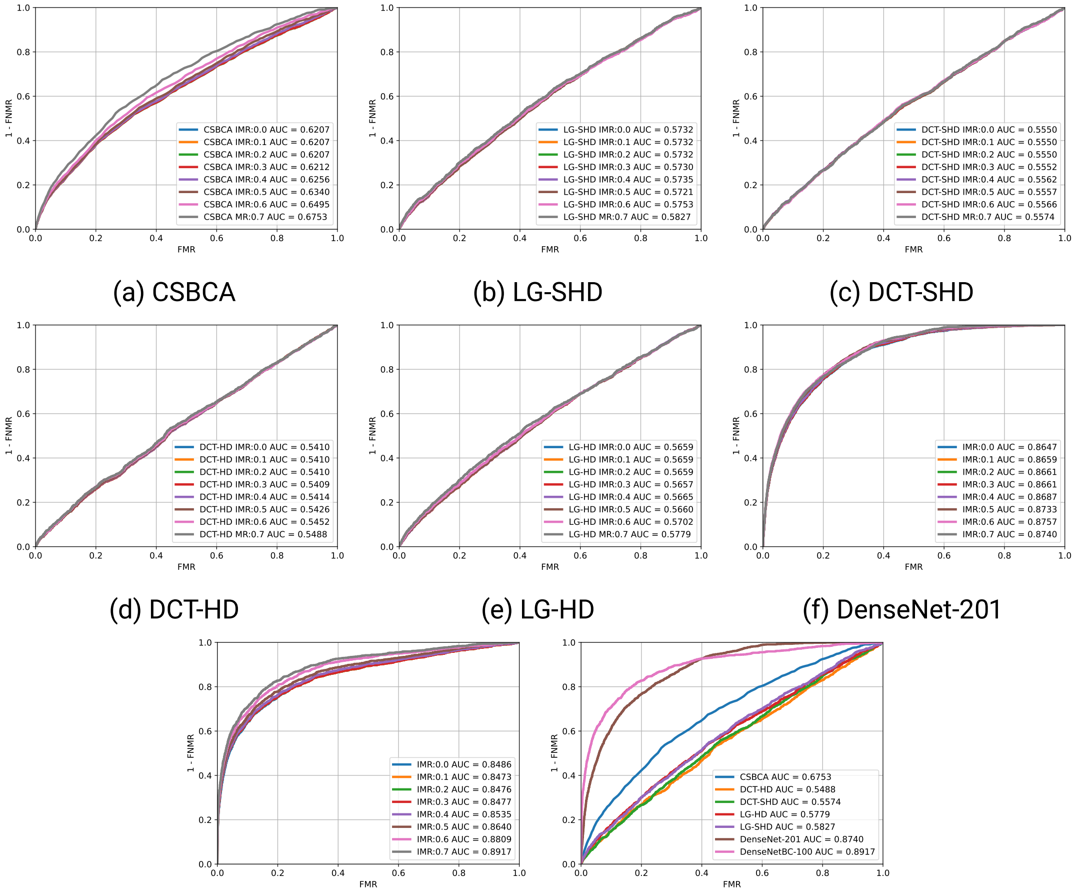
(e) LG-HD

(f) DenseNet-201

(g) DenseNetBC-100

(h) All-IMR 0.7

Figure 5.12.: The achieved ROC curves for the different experimental settings of iris recognition and different IMR thresholds. The results are shown based on D-NET synthesized images . Each of the plots (a) to (g) shows the ROC curves achieved by one of the benchmarking settings with different levels of IMR rejection threshold. Plot (h) shows a comparison of the different algorithms at the most strict IMR threshold (0.7).



(a) CSBCA

(b) LG-SHD

(c) DCT-SHD

(d) DCT-HD

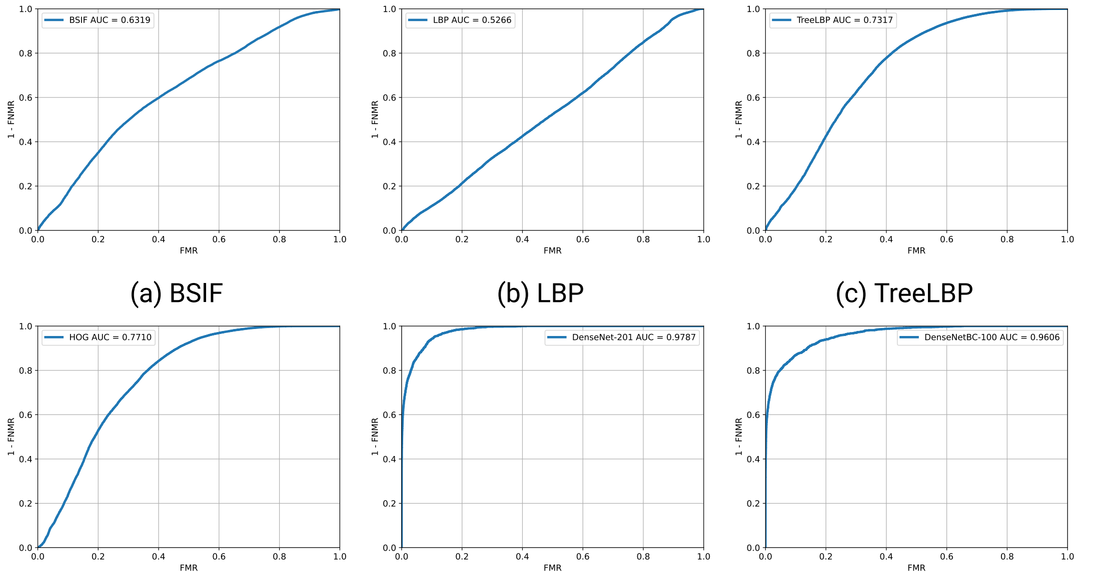
(e) LG-HD

(f) DenseNet-201

(g) DenseNetBC-100

(h) All-IMR 0.7

Figure 5.13.: The achieved ROC curves for the different experimental settings of iris recognition and different IMR thresholds. In this experiment, synthesized images from ID-NET are used. Each of the plots (a) to (g) shows the ROC curves achieved by one of the benchmark settings with different levels of IMR threshold. Plot (h) shows a comparison of the different algorithms at the most strict IMR threshold (0.7).



(a) BSIF

(b) LBP

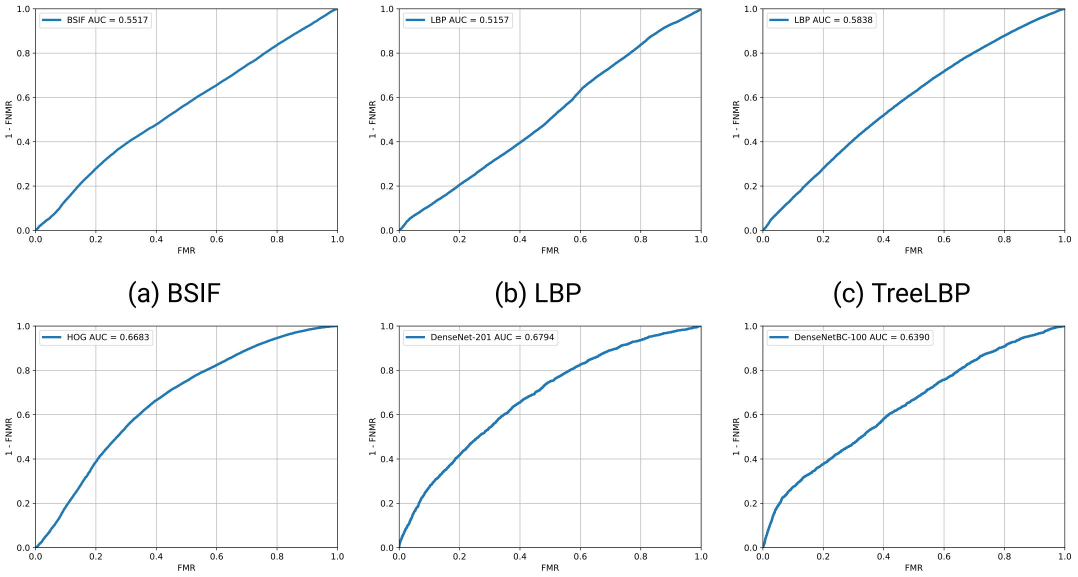
(c) TreeLBP

(d) HOG

(e) DenseNet-201

(f) DenseNetBC-100

Figure 5.14.: The achieved ROC curves for different experimental settings for periocular recognition. In this experiment, synthesized images from ID-NET are used. Each of the plots (a) to (e) shows the ROC curves obtained by one of the benchmark settings.



(a) BSIF

(b) LBP

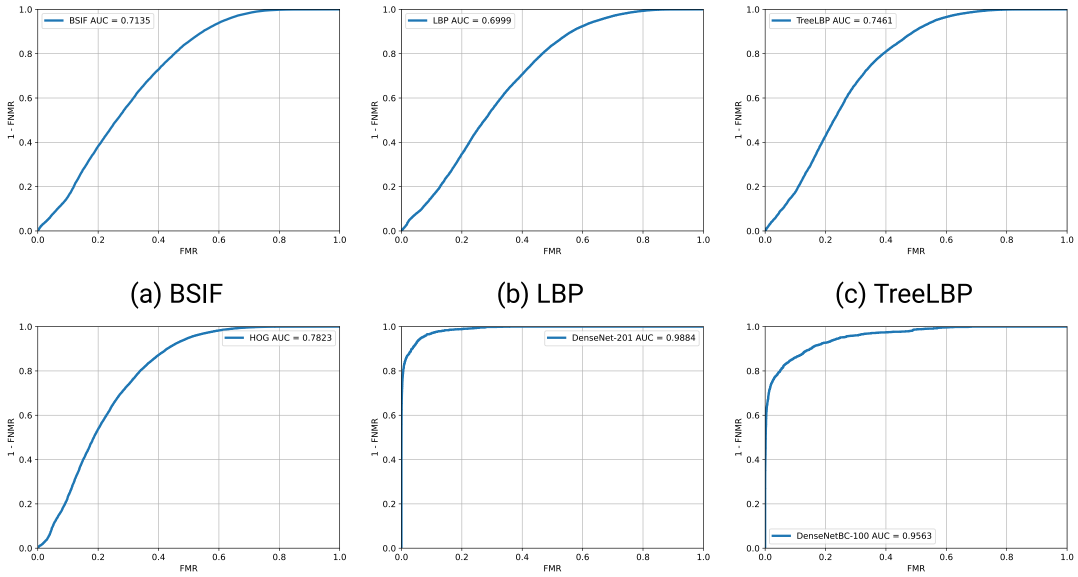
(c) TreeLBP

(d) HOG

(e) DenseNet-201

(f) DenseNetBC-100

Figure 5.15.: The achieved ROC curves for different experimental settings of periocular recognition. In this experiment, synthesized images from D-NET are used. Each of the plots (a) to (e) shows the ROC curves obtained by one of the benchmark settings.



(a) BSIF

(b) LBP

(c) TreeLBP

(d) HOG

(e) DenseNet-201

(f) DenseNetBC-100

Figure 5.16.: The achieved ROC curves for the different experimental settings of periocular recognition. Each of the plots (a) to (e) shows the ROC curves achieved by one of the benchmark settings.

6. Conclusion and Future Work

The previous chapters (3, 4 and 5) provided detailed responses to the research questions presented in Chapter 1. This chapter provides a set of concluding remarks of this thesis and an outlook for future research.

6.1. Conclusion

This thesis offered theoretical and practical contributions towards the development of efficient biometrics, which are deemed essential to enabling a wider deployment of biometric technology. These contributions were motivated by the identified challenges in Chapter 1 and targeted addressing the research questions posed in this thesis. The research questions were assembled into three categories based on the targeted challenges, efficient and high-performing face recognition, the emerging challenge of masked face recognition, and biometrics in head-mounted displays.

Efficient and high-performing face recognition: The first principal research question, RQ1, was addressed by providing responses to the detailed research questions, RQ1.1, RQ1.2, RQ1.3 and RQ1.4. Towards answering these detailed research questions, this thesis proposed extremely efficient FR networks, intelligently developed a new family of lightweight face-specific architectures, designed a step-wise KD approach, and proposed elastic margin-penalty softmax loss.

As a response to RQ1.1, this thesis proposed a set of extremely efficient face recognition models, MixFaceNets, for accurate face verification on low-end devices. The proposed MixFaceNets utilized MixConv as the main building block to capture different patterns from convolution input at various resolutions. Furthermore, MixFaceNets extended the MixConv block with a channel shuffle operation, aiming at increasing the discriminative ability of MixFaceNets. Through extensive experimental evaluations on mainstream benchmarks and comparison to the recent SOTA efficient models, the reported results in Chapter 3 proved the superiority of MixFaceNets over recent efficient models. In a detailed comparison, MixFaceNets outperformed all efficient models that require less than 500M FLOPs on all the evaluated datasets, achieving 99.60% accuracy on LFW, 97.05% accuracy on AgeDB-30, and 93.08 TAR (at FAR1e-4) on IJB-C. With computational complexity between 500M and 1G FLOPs, MixFaceNets achieved competitive results to the top-ranked models while using significantly fewer FLOPs and less computation overhead, which proves the practical value of the proposed MixFaceNets.

With a focus on automated network architecture design, this thesis successfully utilized NAS to develop a new family of lightweight face-specific architectures, PocketNets. The sanity of using

NAS for the specific task of FR rather than general object classification was proven in the conducted ablation study in Chapter 3. In this ablation study, the effect of training dataset sources on the NAS algorithm was studied and analyzed by comparing the network architectures and the verification performances of NAS instance learned from a face-specific dataset (PocketNet) to NAS instance learned from a dataset with general image classes (DartFaceNet). The reported verification results on nine different benchmarks showed that PocketNet outperformed DartFaceNet with a noticeable margin on all considered benchmarks, proving the design choice of PocketNets. Additionally, this thesis proposed a novel training paradigm, the multi-step KD, where the knowledge is distilled from the teacher model to the student model at different stages of the training maturity to fill disparity gap in terms of network size between the student and the teacher model. The benefit of the proposed multi-step KD was empirically proven in a step-wise ablation study, where the verification performance and model convergence of PocketNets trained with multi-step KD are compared to PocketNets trained with conventional KD and PocketNets trained without KD. The reported evaluation results proved the benefit of the proposed multi-step KD in improving the PocketNets verification performance on nine mainstream benchmarks compared to conventional KD, answering RQ1.3. In comparison to the recent compact FR models, PocketNets consistently scored SOTA performances in comparison to the compact models proposed in the literature. For example, PocketNetS-128 (0.92M parameters) achieved 96.10% accuracy on AgeDB-30, outperforming all proposed models in the literature that have less than 1M parameters.

This thesis additionally proposed a novel margin penalty softmax loss, namely ElasticFace, aiming at developing high-performing FR models. The proposed ElasticFace relaxed the fixed margin penalty restrictions by deploying random margin values drawn from a normal distribution in each training iteration. Such flexible margin penalty aimed at giving the decision boundary, i.e., the boundary between the class embedding in normalized hypersphere, chances to extract and retract to allow space for flexible class separability learning. Additionally, the concept of flexible margin penalty in ElasticFace was extended in the proposed ElasticFace+, where the assignment of the drawn margins to training samples was linked to their proximity to their class centers. The presented verification results on nine mainstream benchmarks demonstrated the superiority of ElasticFace loss over its counterparts, ArcFace and CosFace losses, that deployed a fixed margin penalty, providing an answer to RQ1.4. In comprasion to the recent SOTA high-performing FR models, ElasticFace has advanced the SOTA FR performance on seven out of nine mainstream benchmarks. For example, on large age gape benchmark (AgeDB-30), ElasticFace scored an SOTA accuracy of 98.35%.

The emerging challenge of masked face recognition: The focus of the second category of research questions targeted the emerging challenge of masked FR. RQ2 was answered by providing a detailed response to the RQ2.1. Toward this end, this thesis proposed a solution to reduce the negative impact of masked faces on FR performance. The proposed solution was based on designing EUM operated on the top (embedding space) of existing FR and proposing SRT loss to guide the EUM during the training phase to learn to produce embeddings of masked faces similar to unmasked face embeddings of the same identities. The EUM with SRT approach was theoretically motivated by the reported evaluation studies on masked FR, stating that wearing face masked negatively affects the FR performance, and the genuine score distributions are significantly affected by masked probes. In contrast, the imposter score distributions seem to be less affected by masked probes. Motivated by such observations, SRT, unlike triplet loss, is designed in a way that it dynamically self-adjusts its learning objective by optimizing the distance between the genuine pairs only when the distance between the imposter pairs is deemed to be sufficient. The effectiveness of the EUM with SRT approach in reducing the effect of masked faces on FR verification performance was empirically proven through an ablation study and extensive experiment evaluations on four masked face datasets and three FR models and four masked face datasets.

Biometrics in head-mounted display: Considering the ever-increasing use of AR/VR technologies in new fields and the associated developments in HMD devices, this thesis is the first to introduce biometric recognition to AR/VR enabled by HMD. RQ3 was answered by the detailed responses to the research questions RQ3.1, RQ3.2, and RQ3.3. These three research questions were answered by investigating and evaluating the verification performance and computational efficiency of several iris and periocular verification methodologies on the targeted use-case scenario, developing multi-label semantic segmentation models, and proposing a two-stage identity-preserving synthetic ocular image generation approach.

The verification performances of iris and periocular methodologies were reported on a realistic database captured using HMD internal eye-facing cameras. The overall verification result showed that the deep learning approaches achieved better performance than handcrafted approaches, where the best-achieved iris and periocular verification EERs were 10.63% and 5.86%, respectively, by DenseNet-201. Moreover, a multi-scale eye regions semantic segmentation approach was proposed as an essential processing step of iris and periocular recognition pipelines. Three different variants of the segmentation approach were presented Eye-MMS80, Eye-MMS216, and Eye-MS, containing 80K, 216K, and 6574k parameters, respectively. The achieved IoU (mean) by Eye-MMS80, Eye-MMS216 and Eye-MS were 0.9125, 0.9289, and 0.9330, respectively. Variations of these solutions performed very competitively in various segmentation challenges. This thesis also proposed a two-stage D-ID-Net approach to generate realistic and identity-specific images from semantic segmentation labels. The first stage of the D-ID-Net approach generates a generic eye image that corresponds to a given semantic label. The second identity-specific stage induces identity information into that generic image. The generated images largely maintained the targeted identity information confirmed by analyzing and reporting their biometric verification performances compared to the original real data.

Summary To sum up, the previously presented contributions were driven by the research questions discussed in Chapter 1 and motivated by the need for accurate and efficient biometric solutions to enable the spread of the technology in embedded domains. In response to the RQ1 ”Can efficient and high-performing FR approaches be successfully designed?”, this thesis confirmed that by successfully designing a family of efficient and yet accurate FR models, intelligently utilizing NAS to design a compact FR model, and proposing a novel margin-penalty softmax loss. To answer the second principal research question, RQ2 ”Can the negative impact of face masks on FR verification performance be effectively reduced?”, this thesis confirmed that by proposing the EUM trained with SRT approach to reduce the negative impact of wearing a protective face mask on FR performance. Towards introducing biometrics to VR/AR applications enabled by HMDs and as a response to RQ3 ”Can existing VR/AR setups be leveraged for the biometric verification of their users identities?”, this thesis presented three key contributions. This thesis was the first to introduce and investigate the biometric verification performance of iris and periocular characteristics on a realistic database captured from HMDs. Moreover, this thesis successfully designed a compact and accurate multi-label semantic segmentation model as an essential part of an iris and periocular recognition pipeline. Furthermore, this thesis proposed a novel identity-preserving synthetic ocular image generation approach to promote further development of biometric recognition in new domains.

6.2. Future Work

The development of efficient biometrics aims at reducing the computational cost of deploying biometric recognition, which enables wider implementation of biometric recognition in the use-cases constrained by computational capabilities and operational limitations. This prompts several future research directions that can build up on the contributions of this thesis. These research directions can be summarized as follows:

Green AI in biometrics Since 2012, deep learning has become a popular method to solve several machine learning problems, achieving exceptional accuracy in many computer vision tasks [107]. Three main aspects drive the rapid progress in deep learning: 1) the innovation in deep neural network architectures and training algorithms, 2) the availability of large-scale training datasets, and 3) the massive computational resources available for model training. The last two points, along with the utilization of high-computational capabilities platforms, enable training extremely deep architecture and clearly benefit the model accuracy [10]. However, the environmental and economic costs of training deep learning models have received less attention and were rarely discussed and reported along with the model accuracy. In general, deep learning approaches presented in the literature mainly focused on reporting a single metric, the accuracy. While niche solutions targeting computational efficiency additionally reported the computational needs to deploy the model, ignoring the environmental (CO2 emissions) and the financial costs of the model training. Training deep neural models is responsible for tons of CO2 emissions. A study by Strubell et al. [247] quantified the approximate financial and environmental costs (CO2 emission) of deep learning-based Natural Language Processing (NLP) models. This study [247] reported that training one machine translation model that uses NAS was responsible for an estimated 626,155 tons of CO2 emissions (274,120 hours of training) and its training costs between $942,973 and $3,201,722 on eight P100 GPUs on a cloud compute (the lower and upper bounds of GPUs cost varied between the providers), which reflects energy consumption level. Such high CO2 emissions are around 17 times more than the CO2 emissions caused by an average American per year (estimated at 36,156 tons of CO2 emissions per year) [247]. Schwartz et al. [235] proposed the concept of Green AI that considers the price tag, i.e., the cost of training and deploying deep learning solutions, as an additional metric to the model accuracy. This opens new research directions for environment-friendly deep learning-based solutions that do not only focus on the deployment cost but also on the environmental and economic costs of training deep learning models. Following this motivation, future work can extend the deployments of ”on-the-top” solutions, in which a small model is designed and trained to operate on the top of an existing pre-trained model. Thus, it does not require retraining of existing base models or the use of additional full-scale domain-specific models. An example of ”on-the-top” solution is the the EUM with SRT presented in Chapter 4. This solution can be adapted to solve other biometric problems, e.g. reducing the performance variation across demographic groups.

Additionally, the concept of Green AI and efficient deep learning models prompts the need for defining a set of standard evaluation metrics to measure and report the cost of model development, training, and deployment.

Standardization of computational cost evaluation and reporting The reported computational cost metrics in the literature varied between different approaches. Commonly reported metrics for deploying deep learning-based model are FLOPs, FLOP/S, Multiply Accumulate Operations (MACs), number of trainable parameters (memory footprint), and inference time (latency) [49, 23, 284, 179, 150]. For model training computational cost, the commonly reported metric is the required GPU hours [155, 41]. However, there is no standardized evaluation and reporting protocol, including metrics, to measure the computational cost of AI solutions in a comparative manner, similarly to the standards that define biometric performance evaluation and reporting protocols [123]. A suitable standardization body to work on such direction would be the ISO/IEC JTC 1/SC 42 committee on Artificial intelligence that commenced operation in 2017.

On a technical note, some of the current metrics that are used to estimate the cost of deploying a model, such as FLOP/S and inference time, are hardware-dependant and deployment platform-dependent metrics, making it difficult to compare different approaches based on these metrics [173]. Other metrics such as FLOPs and the number of trainable parameters are independent of the deployment hardware, making them more suitable than hardware-related ones to estimate the computational cost of model deployment [49, 23, 38, 233]. However, the actual operation cost may depends on the application scenario. For example, some application scenarios may require continuous authentication e.g. VR/AR applications enabled by HMDs [34], leading to high energy consumption. Other application scenarios, such as logical access control, operate on demand and consume less energy than continuous authentication. Moreover, the GPU hours required for training might not be sufficient to estimate the financial and environmental costs of model training, as this depends on the training hardware and training settings. Additionally, reporting the expected CO2 emissions resulting from a model training is highly dependent on the energy source, e.g. green energy or fossil fuels. This raises the need for a set of evaluation protocols and metrics for the model development, training, and deployment computational cost to promote developing environment-friendly and efficient solutions.

Hardware-aware biometrics The design and development choices of biometric systems typically depend on the targeted use case, the deployment environment, and the available hardware resources. Application scenarios, such as automated border control (ABC), use dedicated hardware for biometric recognition [74]. Other use-case scenarios such as verification by mobile and HMD devices use built-in and multi-purpose hardware for biometric recognition [34]. The models designed to operate on a specific platform, e.g. mobile devices, may lead to sub-optimal performances, in terms of latency, when deployed to another platform, e.g. HMD devices. This promotes the need for designing hardware-aware biometric solutions. Hardware-aware NAS approaches have been growing in popularity to design efficient models tailored for specific hardware platforms [155, 41]. However, such hardware-aware architecture design approaches are rarely utilized to design biometric recognition approaches.

Multi-task modules The signal processing subsystem of a biometric system consists of several modules including, feature extraction, quality control, presentation attack detection (PAD), and segmentation models [123] as described in Chapter 2. These models commonly use different approaches for each tasks. The deployment and operation cost of all models is high compared to each model separately. One possible future research direction that might help to reduce the overall cost is to design a multi-task model that can preform all signal processing subsystem operations using a single efficient model. The proposed efficient face recognition model, MixFaceNet [23], in this thesis (Chapter 3) has been recently used successfully as a backbone for a PAD solution [87]. Such duplicate use of network architectures can be combined in a single multi-purpose model to reduce the overall computational costs.

Adaptive biometric solutions The recognition performance of biometric solutions depends on the quality of the captured sample. The reference samples are usually acquired under a controlled capture environment. However, this might not always be the case for probe samples, especially when the probe samples are acquired under uncontrolled scenarios, leading to several sample degradations such as physical distortion, occlusion, and low resolution. The concept of the proposed EUM with SRT presented in this thesis could be extended to solve other biometric challenges such as learning to produce biometric templates of low-quality samples that behave similarly to the ones of high quality. Additionally, such approach could be further automated to handle less-granularly defined reasons of biometric performance degradation.

Biometric model compression Model compression techniques such as parameter pruning and model quantization are established methods to reduce the required computational cost of deep learning models. Neural network parameter pruning creates sparse neural networks from dense ones by removing neurons that have a relatively low effect on model accuracy [255]. Model quantization approaches compress deep neural networks by reducing the number of bits required to represent each weight [18, 125]. These techniques were not sufficiently studied and applied to regulate the computational cost of biometric recognition models, which promotes several future research directions aiming at reducing the computational cost of biometric recognition models with a minimal reduction in the recognition performance.

A. Masked Face Recognition Competitions

Section 4.6 briefly presented the Masked Face Recognition Competitions (MFR) held within the 2021 International Joint Conference on Biometrics (IJCB 2021) [29]. To provide detailed insights into the recent solutions from academia and industry that are designed to perform well with a masked face, this chapter provides a thorough description of MFR competition, including the competition dataset along with the evaluation criteria, the participant teams, the submitted solutions, and the achieved results.

The MFR competition attracted ten participating teams from nine different countries that submitted 18 valid submissions. This section provides a detailed description of MFR competition, starting with a motivation for MFR competition in Section A.1. This is followed by detailed descriptions of the competition evaluation database, the evaluation criteria, and the participating teams. Then, in Section A.3, short descriptions of the submitted solutions are listed. Section A.4 presents and discusses the achieved results along with listing the winning submissions. A set of conclusions are drawn in Section A.5 with a final general conclusion.

A.1. Introduction

Given the current COVID-19 pandemic, it is essential to enable contactless and smooth-running operations, especially in contact-sensitive facilities like airports. With the ever-enhancing performance of face recognition, the technology has been preferred as a contactless means of verifying identities in applications ranging from border control to logical access control on consumer electronics. However, wearing masks is now essential to prevent the spread of contagious diseases and has been currently forced in public places in many countries. The performance, and thus the trust in contactless identity verification through face recognition can be impacted by the presence of a mask [99]. The effect of wearing a mask on face recognition in a collaborative environment is currently a sensitive issue. This competition is the first to attract and present technical solutions that enhance the accuracy of masked face recognition on real face masks and in a collaborative verification scenario.

In a recent study, the National Institute of Standards and Technology (NIST), as a part of the ongoing Face Recognition Vendor Test (FRVT), has published a specific study (FRVT -Part 6A) on the effect of face masks on the performance on face recognition systems provided by vendors [201]. The NIST study concluded that the algorithm accuracy with masked faces declined substantially. One of the main study limitations is the use of simulated masked images under the questioned assumption that their effect represents that of real face masks. The Department of Homeland Security has conducted an evaluation with similar goals, however on more realistic data [15]. They also concluded with the significant negative effect of wearing masks on the accuracy of automatic face recognition solutions. A study by Damer et al. [64] evaluated the verification performance drop in 3 face biometric systems when verifying masked vs. not-masked faces, in comparison to verifying not-masked faces to each other. The authors presented limited data (24 subjects), however, with real masks and multiple capture sessions. They concluded by noting the bigger effect of masks on genuine pairs decisions in comparison to imposter pairs decisions. This study has been extended [62] with a larger database and evaluation on both synthetic and real masks, pointing out the questionable use of simulated masks to represent the real mask effect on face recognition. Recent work has evaluated the human performance in recognizing masked faces in comparison to automatic face recognition solutions [58]. The study concluded with a set of take-home messages that pointed to the correlated effect of wearing masks on both human recognizers and automatic face recognition. Beyond recognition, facial masks showed to affect both the vulnerability of face recognition to presentations attacks and the detectability of these attacks [92].

There were only a few works that addressed enhancing the recognition performance of masked faces. Li et al. [151] proposed to use an attention-based method to train a face recognition model on the periocular area of masked faces. This presented improvement in the masked face recognition performance, however, in a limited evaluation. Moreover, the proposed approach essentially only maps the problem into a periocular recognition problem. A recent preprint by [13] presented a relatively small dataset of 53 identities crawled from the internet. The work proposed to fine-tune FacenNet model [234] using simulated masked face images to improve the recognition accuracy. Wang et al. [271] presented three datasets crawled from the internet for face recognition, detection, and simulated masked faces. The authors claim to improve the verification accuracy from 50% to 95% on masked faces. However, they did not provide any information about the evaluation protocol, proposed solution, or implementation details. Moreover, the published part of the dataset does not contain pairs of not-masked vs. masked images for the majority of identities. A work by Montero et al. [191] proposed to combine ArcFace loss with a specially designed mask-usage classification loss to enhance masked face recognition performance. Boutros et al. [26] proposed a template unmasking approach that can be adapted on the top of any face recognition network. This approach aims to create unmasked-like templates from masked faces. This goal was achieved on top of multiple networks by the proposed self-restrained triplet loss [26]. On a related matter, a rapid number of works are published to address the detection of wearing a face mask [21, 163, 270, 220]. These studies did not address the effect of wearing a mask on the performance of face recognition or present solution to improve masked face recognition.

Besides the exclusive interest in face recognition accuracy, there is a growing interest in compact face recognition models [180]. This interest is driven by the demand for face recognition deployment on consumer devices and the need to enhance the throughput of face recognition processes. A major challenge has been organized in ICCV 2019 to motivate researchers to build lightweight face recognition models [81]. MobileFaceNets are an example of such face recognition models [49]. MixeFaceNet [23] is a recent example where mixed depthwise convolutional kernels, with a tailored head and embedding design and a shuffle operation, are utilized to achieve high recognition accuracies with extremely light models.

Motivated by (a) the hygiene-driven wide use of facial masks, (b) the proven performance decay of existing face recognition solutions when processing masked faces, (c) the need to motivate novel research in the direction of enhancing masked face recognition accuracy, and (d) the requirement of lightweight models by various applications, we conducted the IJCB Masked Face Recognition Competition 2021 (IJCB-MFR-2021). The competition attracted submissions from academic and industry teams with a wide international representation. The final participation toll was 10 teams with valid submissions. These teams submitted 18 valid solutions. The solutions were evaluated on a database collected to represent a collaborative face verification scenario with individuals wearing real face masks. This chapter summarises the MFR competition with a detailed presentation of the submitted solutions and the achieved results in terms of masked vs. masked face verification accuracy, masked vs. not-masked face verification accuracy, and the compactness of the recognition models.

A.2. Database, evaluation criteria, and participants

A.2.1. Database

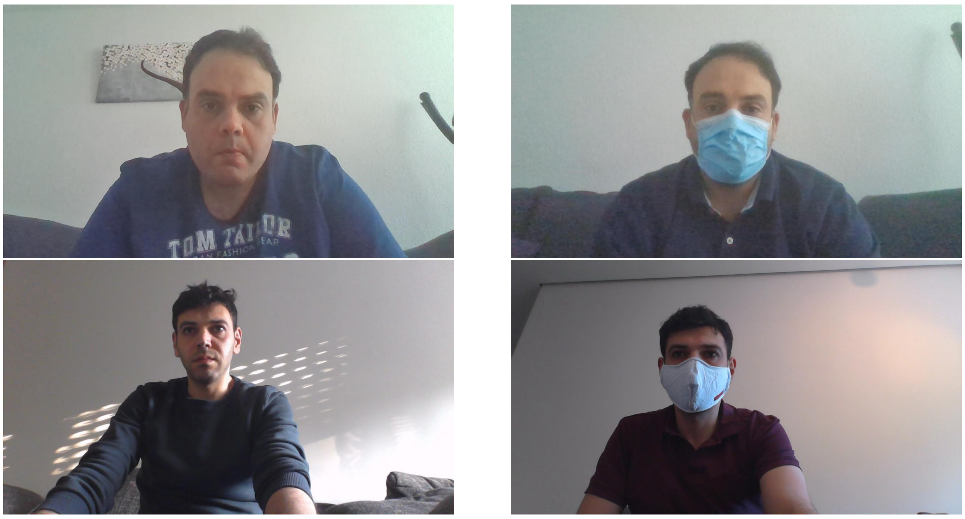
The evaluation data, the masked face recognition competition data (MFRC-21), simulates a collaborative yet varying scenario. Such as the situation in automatic border control gates or unlocking personal devices with face recognition, where the mask, illumination, and background can change. The database is collected by the hosting institute and not available publicly. The data is collected on three different, not necessarily consecutive days. Each of these days is considered as one session. On each day, the subjects have collected three videos, each of a minimum length of 5 seconds (used as single image frames). The videos are collected from static webcams (not handheld), while the subjects are requested to look at the camera, simulating a login scenario. The data is collected by subjects at their residences during the pandemic-induced home-office period. The first session is considered a reference session, while the other two were considered probe sessions. Each day contained three types of captures, no mask, masked with natural illumination, masked with additional illumination. The database participants were asked to remove eyeglasses only when the frame was considered very thick. No other restrictions were imposed, such as background or mask type and its consistency over days, to simulate realistic scenarios. The first second of each video was neglected to avoid possible biases related to the subject interaction with the capture device. After the neglected one second, three seconds were considered. From these three seconds, 10 frames are extracted with a gap of 9 frames between each consecutive frame, knowing that all videos are captured at a frame rate of 30 frames per second.

The final considered portions of the database in the competition are (a) the not-masked baseline reference from the first session (noted as BLR), (b) the masked reference from the first session (noted as MR), and (c) the masked face probes from the second and third sessions under both illumination scenarios (noted as MP). A summary of the used database is presented in Table A.1 and samples of the database are presented in Figure A.1. The database contained 47 subjects, all of them participated in all the sessions. All the subject provided their informed consent to use the data for research purposes.

Two evaluation setups are considered, (a) not-masked vs. masked, where all images in BLR are compared to all images in MP (noted as BLR-MP), and (b) masked vs. masked, where all images in MR are compared to all images in MP (noted as MR-MP).

Session	Session 1: References		Session 2 and 3: Probes
Data split	BLR	MR	MP
Number of Captures	470	940	1880

Table A.1.: An overview of the MFRC-21 database structure.



(a) Not-masked baseline faces (BLR)

(b) Masked faces (MR/MP)

Figure A.1.: Samples of the MFRC-21 database from the two capture types (BLR and MR/MP). MR and MP have similar capture settings, MR on the first setting and MP on the second and third session.

A.2.2. Evaluation criteria

The solutions evaluation will be based on both, the verification performance and the compactness of the used mode/models. The verification evaluation will be based on the verification performance of masked vs. not-masked verification pairs, as this is the common scenario, where the reference is not-masked, while the probe is masked, e.g. in entry to a secure access area. This scenario will be noted as BLR-MP. However, the performance of masked vs. masked verification pairs is also be reported. This scenario is noted as MR-MP.

The verification performance is evaluated and reported as the false non-match rate (FNMR) at different operation points FMR100 and FMR1000, which are the lowest FNMR for a false match rate (FMR) $< 1.0\%$ and $< 0.1\%$, respectively. The verification performance evaluation of the submitted solutions is based on FMR100. To get an indication of generalizability, a separability measure between the genuine and imposter comparison scores is also reported. This is measured by the Fisher Discriminant Ratio (FDR) as formulated in [214].

To consider the deployability of the participating solutions, we will also consider the compactness of the model (represented by the number of trainable parameters [83]) in the final ranking. The participants are asked to report the number of trainable parameters and can be asked to provide their solutions to validate this number.

The final teams ranking is be based on a weighted Borda count, where the participants will be ranked by (a) the verification metric as mentioned above (noted as Rank-a), and (b) by the number of trainable parameters in their model/models (notes as Rank-b). For Rank-a, the solutions with lower FMR100 are ranked first, and for Rank-b, the solutions with the lower number of trainable parameters are ranked first. In the final ranking, Rank-a will have 75% weight, and Rank-b will have 25% weight. Each participant is given a Borda count (BC) for each ranking criteria (BC-a and BC-b). For example, if solution X is ranked first out of 10 participants in the verification performance rank-a (BC-a =9) and third out of 10 solutions in model compactness Rank-b (BC-b = 7) (this corresponds to BC = total number of solutions – rank). Then the weighted Borda count w-BC = 0.75x9+0.25x7= 8.5. Therefore, the final score of solution X is 8.5, and higher indicates a better solution. The solutions are ranked from the highest w-BC to the lowest w-BC.

A.2.3. Submission and evaluation process

Each of the teams was requested to submit their solutions as Win32 or Linux console applications. These applications should be able to accept three parameters, evaluation-list (text file), landmarks (text file), and an output path. The evaluation-list contains pairs of the path to the reference and probe images and a label for each of the compared images, indicating if the image is masked or not. The landmarks provided a bounding box and five landmark locations of the images as detected by the MTCNN solution [289]. Only the pairs of images with valid detected faces are provided to the solutions in the evaluation list. From the initial considered data, the face detector [289] did not provide valid face detections. For the BLR-MP pairs, 4.42% of the pairs contained invalid detections of faces and thus were not considered in the evaluation. For the MR-MP pairs, 4.75% of the pairs contained invalid detections of faces and thus not considered in the evaluation. The output of the solution application script is a text file containing comparison scores for each pair in the evaluation list.

A.2.4. Competition participants

The competition aimed at attracting participants with a high geographic and activity variation. The call for participation was shared on the International Joint Conference on Biometrics (IJCB

2021) website, on the competition own website [1], on public computer vision mailing lists (e.g. CVML e-Mailing List), and through private e-Mailing lists. The call for participation has attracted 12 registered teams. Out of these, 10 teams have submitted valid solutions. These 10 teams have affiliations based in nine different countries. Seven of the 10 teams are affiliated with academic institutions, two are affiliated with the industry, and one team has both academic and industry affiliations. Only one of the participating teams has chosen to be anonymous. Each team was allowed to submit up to two solutions. The total number of validly submitted solutions is 18. A summary of the participating teams is presented in Table A.2.

Solution	Team members	Affiliations	Type of institution
A1_Simple	Asaki Kataoka, Kohei Ichikawa, Shizuma Kubo	ACES, Inc, Japan	Industry
TYAI	Pengcheng Fang, Chao Zhang, Fei Wang	TYAI, China	Industry
MaskedArcFace	David Montero, Naiara Aginako Basilio Sierra, Marcos Nieto	Vicomtech, Spain - University of the Basque Country, Spain	Academic
MTArcFace			
MFR-NMRE-F	Klemen Grm, Vitomir Štruc	University of Ljubljana, Slovenia	Academic
MFR-NMRE-B			
MUFM Net	Sachith Seneviratne, Nuran Kasthuriarachchi, Sanka Rasnayaka	University of Melbourne, Australia - National University of Singapore, Singapore - University of Moratuwa, Sri Lanka	Academic
EMUFM Net			
VIPLFACE-M	Jie Zhang , Mingjie He, Dan Han, Shiguang Shan	Institute of Computing Technology, Chinese Academy of Sciences, China, University of Chinese Academy of Sciences, China	Academic
VIPLFACE-G			
SMT-MFR-1	Mustafa Ekrem Erakın, Uğur Demir, Hazım Kemal Ekenel	Smart Interaction and Machine Intelligence Lab (SiMiT Lab), Istanbul Technical University, Turkey	Academic
SMT-MFR-2			
LMI-SMT-MFR-1	Mustafa Ekrem Erakın, Uğur Demir, Hazım Kemal Ekenel, Klemen Grm, Vitomir Štruc	Istanbul Technical University, Turkey - University of Ljubljana, Slovenia	Academic
LMI-SMT-MFR-2			
IM-MFR	Pedro C. Neto, Ana F. Sequeira, João Ribeiro Pinto, Mohsen Saffari, Jaime S. Cardoso	INESC TEC, Portugal - University of Porto, Faculty of Engineering (FEUP), Portugal	Academic
IM-AMFR			
Anonymous-1	Anonymous	Anonymous	mix
Anonymous-2			

Table A.2.: A summary of the submitted solutions, participant team members, affiliations, and type of institutions (Industry, Academic, or mix). The table lists the abbreviations of each submitted solution. Details of the submitted algorithms are in Section A.3.

A.3. Submitted solutions

Ten teams have been registered for MFR 2021 competition and submitted 18 valid solutions. Table A.2 presents a summary of the registered team members and their affiliation, submitted solutions, and type of institution of each registered team (Academic, Industry, or mix of both academic and industry). In the following, a brief description of the valid submitted solutions is provided:

A1_Simple employed ArcFace [80] to train a ResNet model. A1_Simple applied MaskTheFace [13] method to synthetically generate masked face images in the training dataset- MS1MV2. A1_Simple is trained with cosine annealing LR scheduling to adjust the learning rate. In the evaluation phase, A1_Simple used the provided landmark facial point and bounding box in the MFRC-21 to align and crop the face image to 112×112. The feature embedding of the presented solution is of size 512-D. The model is trained with ArcFace loss. During the training phase, three

[1] https://sites.google.com/view/ijcb-mfr-2021/home

Solution	Verification performance					Compactness			Joint	
	FMR100	FMR1000	FDR	Rank-a	BC-a	number of parameters	Rank-b	BC-b	w-BC	Rank
Baseline	0.06009	0.07154	8.6899	-	-	65155648	-	-	-	-
TYAI	0.05095	0.05503	11.2005	1	17	70737600	14	4	13.75	**1**
MaskedArcFace	0.05687	0.05963	10.4484	5	13	43589824	6	12	12.75	**2**
SMT-MFR-2	0.05584	0.06268	11.2025	3	15	65131000	12	6	12.75	**2**
A1_Simple	0.05538	0.06113	8.5147	2	16	87389138	16	2	12.5	**3**
VIPLFACE-M	0.05681	0.06279	8.2371	4	14	65128768	10	8	12.5	**3**
MTArcFace	0.05699	0.05860	10.7497	6	12	43640002	7	11	11.75	4
SMT-MFR-1	0.05704	0.06003	10.6824	7	11	65131000	12	6	9.75	5
VIPLFACE-G	0.05750	0.07269	8.1693	9	9	65128768	10	8	8.75	6
MFR-NMRE-B	0.05819	0.08344	7.9504	10	8	43723943	8	10	8.5	7
LMI-SMT-MFR-1	0.05722	0.06205	9.7384	8	10	108854000	17	1	7.75	8
MFR-NMRE-F	0.08125	0.17660	5.3876	12	6	43723943	8	10	7	9
MUFM Net	0.17579	0.40489	4.4640	14	4	25636712	3	15	6.75	10
IM-AMFR	0.28252	0.47608	3.7414	15	3	36898792	4	14	5.75	11
LMI-SMT-MFR-2	0.05848	0.07096	8.5278	11	7	108854000	17	1	5.5	12
Anonymous-1	0.92536	0.96596	0.1011	17	1	23777281	1	17	5	13
IM-MFR	0.28447	0.47430	3.7369	16	2	36898792	4	14	5	13
EMUFM Net	0.16239	0.35681	4.5445	13	5	76910136	15	3	4.5	14
Anonymous-2	0.97125	0.99517	0.0426	18	0	23777281	1	17	4.25	15

Table A.3.: The comparative evaluation of the submitted solutions on the MFRC-21 dataset. The results are presented in terms of verification performance including FMR100, FMR1000, and FDR, and the model compactness in terms of the number of trainable parameters. The FMR100 and FMR1000 are given as absolute values. The rank of the verification performance (Rank-a) is based on FMR100 and the rank of the solution compactness (Rank-b) is based on the number of parameters. Rank-a has 75% weight and Rank-b has 25% weight. The results are ordered based on weighted Borda count (w-BC).

data augmentation methods are used- random resized crops, random horizontal flip, and color jittering.

TYAI solution uses Sub-center ArcFace [78] and ir-ResNet152 model to train a masked face recognition model on Glint360K dataset [11]. The proposed solution randomly augmented half of the training dataset with a synthetic generated mask using five types of transparent masks. The input image size of the proposed model is 112 × 112 and the size of the output feature embedding is 512-D. During the training, additional four data augmentation methods are used: random crop by resizing the image to 128 × 128 and then randomly cropping it to 112 × 112, random horizontal flip, random rotation, and random affine. The model uses a Sub-center ArcFace loss to train the proposed solution.

Mask aware ArcFace (MaskedArcFace) opts to generate a masked twin dataset from MS1MV2 [103, 80] dataset and to combine them during the training process. Both datasets are shuffled separately using the same seed and, for every new face image selected for the input

Solution	Verification performance			
	FMR100	FMR1000	FDR	Rank
Baseline	0.05925	0.06504	9.68640	-
TYAI	0.04489	0.05961	12.36306	1
VIPLFACE-M	0.05759	0.06788	8.98593	2
A1_Simple	0.05771	0.06368	10.48611	3
SMT-MFR-2	0.05792	0.06172	11.30901	4
MaskedArcFace	0.05825	0.06245	10.57307	5
SMT-MFR-1	0.05825	0.06012	11.03444	6
VIPLFACE-G	0.05843	0.06359	9.41466	7
MTArcFace	0.0585	0.06390	10.16996	8
LMI-SMT-MFR-1	0.05856	0.06061	9.90914	9
LMI-SMT-MFR-2	0.05916	0.06586	8.87424	10
MFR-NMRE-B	0.05970	0.12903	8.11963	11
MFR-NMRE-F	0.09630	0.1989	4.73224	12
EMUFM Net	0.15045	0.31945	4.45317	13
MUFM Net	0.16354	0.37607	4.43278	14
IM-AMFR	0.23507	0.40265	3.94744	15
IM-MFR	0.23661	0.40373	3.94905	16
Anonymous-1	0.89481	0.97584	0.19968	17
Anonymous-2	0.9114	0.98102	0.16569	18

Table A.4.: The comparative evaluation results of the submitted solutions. The verification evaluation is based on the verification performance of masked vs. masked verification pairs where references and probes are masked. The performances are reported in terms of FMR-100, FMR-1000 and FDR. The FMR100 and FMR1000 are given as absolute values. The reported results are ordered based on FMR-100.

batch, MaskedArcFace decides whether the image is taken from the original (not-masked) or the masked dataset with a probability of 50%. MaskedArcFace use ArcFace [80] as the baseline work. MaskedArcFace selects the dataset recommended by ArcFace (MS1MV2) [103, 80] as the training dataset, which contains 5.8M images and 85,000 identities. MaskedArcFace uses IResNet-50 as the backbone among all the network architectures tested in the ArcFace repository as it is it offers good trade-off between the accuracy and the number of parameters. For the generation of the masked version of the dataset, MaskedArcFace uses MaskTheFace [13]. The types of masks considered are surgical, surgical green, surgical blue, N95, cloth, and KN95. The mask type is selected randomly

Solution	Input size	FM	Loss function	RM	SM
A1_Simple	112 x 112	512	ArcFace	No	Yes
TYAI	112 x 112	512	Sub-center ArcFace	No	Yes
MaskedArcFace	112 x 112	512	ArcFace	No	Yes
MTArcFace	112 x 112	512	ArcFace	No	Yes
MFR-NMRE-F	96 x 192	2048	CE	No	No
MFR-NMRE-B	112 x 224	2048	CE	No	No
MUFM Net	224 x 224	2048	CE	No	Yes
EMUFM Net	224 x 224	2048	CE	No	Yes
VIPLFACE-M	112 x 112	512	ArcFace	No	Yes
VIPLFACE-G	112 x 112	512	ArcFace	No	No
SMT-MFR-1	112 x 112	512	ArcFace	Yes	No
SMT-MFR-2	112 x 112	512	ArcFace	Yes	No
LMI-SMT-MFR-1	96 x 192	2048	CE	No	No
	112 x 112	512	ArcFace	Yes	No
LMI-SMT-MFR-2	112 x 224	2048	CE	No	No
	112 x 112	512	ArcFace	Yes	No
IM-MFR	224 x 224	512	CE, triplet loss and MSE	No	Yes
IM-AMFR	224 x 224	512	CE, triplet loss and MSE	No	Yes
Anonymous-1	160 x 160	512	CE	No	Yes
Anonymous-2	160 x 160	512	CE	No	Yes

Table A.5.: Basic details of the submitted solutions including, the input image size, the feature embedding size (FM), the loss function used for training, the use of real masked faces (RM), and simulated masked faces (SM) in the training process. The solutions in bold are the ones ranked top in the competition. Note that all the top-ranked solutions used a version of the ArcFace loss [80, 78].

with a 50% probability of applying a random color and a 50% probability of applying a random texture. During the evaluation phase, MaskTheFace uses the provided landmark points and the bounding box provided by the competition to align and crop face images. The feature embedding produced by MaskedArcFace solution is of the size 512-D and the input face image is of the size 112×112 pixels.

Multi-task ArcFace (MTArcFace) utilized the same training dataset, loss function, backbone, and mask generation method as in MaskedArcFace. MTArcFace adds another dense layer in parallel to the one used to generate the feature vector by IResNet-50, just after the dropout layer. The new dense layer generates an output with two floats, which correspond to the scores related to the probability that the face is masked or not, respectively. This way, MTArcFace aims to force the network to learn when a face is wearing a mask. This information will also be used by the layer that generates the feature vector. The data preprocessing steps and the size of the feature embedding are identical to the MaskedArcFace.

Masked face recognition using non-masked region extraction and fine-tuned recognition model (MFR-NMRE-F) Based on the 5-point face landmark detections, the proposed approach identifies a crop that corresponds to the upper facial region where masks are not visible. Then, MFR-NMRE-F fine-tuned a VGG2-SE-ResNet-50 face recognition model for the classification task on these crops using the VGGFace2 [42] training dataset processed with the RetinaFace [79] detector. For the evaluation, MFR-NMRE-F uses the provided face landmarks provided by MFRC-21, since they correspond closely to the RetinaFace results obtained on the training dataset. Using the landmark coordinates, the MFR-NMRE-F solution extracts the upper face region, extracts feature vectors using the fine-tuned VGG2-SE-ResNet-50 model, and compares features using the cosine similarity measure. The proposed method is trained using cross-entropy (CE) loss. The input size of the proposed model is 96×192, and the feature embedding size is 2048-D.

Masked face recognition using non-masked region extraction and pre-trained recognition model (MFR-NMRE-B) identifies a crop that corresponds to the upper facial region where masks are not visible based on the 5-point face landmark. MFR-NMRE-B utilized a VGG2-SE-ResNet-50 model pre-trained for the classification task using the VGGFace2 [42] training dataset. Different from MFR-NMRE-F, the MFR-NMRE-B solution did not fine-tune the feature extraction model with cropped images. For the evaluation, the proposed method uses the provided face landmarks provided by MFRC-21. Using the landmark coordinates, the proposed method crops the upper face region, extracts feature vectors using the VGG2-SE-ResNet-50 model, and compares features using the cosine similarity measure. MFR-NMRE-B is trained using Softmax cross-entropy loss. The input size of the proposed model is 112×224, and the feature embedding size is 2048-D.

Masked-Unmasked Face Matching Net (MUFM Net) utilizes Momentum Contrast (MoCo) [104] to create an initial embedding using a ResNet-50 model trained on CelebA dataset [162]. Then, synthetic masked versions of CelebA, Spectacles on Faces [7], Youtube Faces [276] and LFW [114] are created as defined in [201]. The initial model is fine-tuned using these dataset. For fine-tuning, MUFM Net uses a siamese network with shared weights with absolute differences taken at the last bottleneck layer. This difference is fed into a 512 fully connected layer followed by a single softmax node.The model is fine-tuned with binary cross-entropy loss with 50% of layers frozen. The input size of the presented model is 224×224 pixels.

Ensemble MUFM Net (EMUFM Net) builds upon MUFM to create an ensemble. First, the best-performing MUFM models are selected based on the validation accuracy. The selected models are M1 (obtained after 695K iterations) and M2 (obtained after 885K iteration) These models are fine-tuned on hard examples drawn from the training set. Three models are fine-tuned- E1 and E2 builds on M1 where 90% and 80% of the layers are frozen, respectively, and E3 builds on M2 where 50% of the layers are frozen. All these models have an input of size 224×224 and an output embedding of size 2048-D. During the testing phase, the similarity scores of these three models (E1-3) are averaged to provide the final similarity score.

VIPLFACE-M adopted ResNet-100 [107] and ArcFace loss [80] for face recognition. The proposed solution uses a refined version of the MS1M dataset [103] for training the proposed solution. The number of face images in the training dataset is 3.8M of 50K identities. VIPLFACE-M uses the synthetic mask creation method defined in [2] to add synthetic masks on part of the training dataset. The number of synthetically masked face images used in training is 500K, and the number of synthetically masked identities is 50K. During the training phase, the proposed solution uses random flipping as a data augmentation method. The input size of the presented solution is 112×112, and the output feature embedding size is 512-D.

VIPLFACE-G is based on training ResNet-100 model [107] with ArcFace loss [80]. The input size of the presented solution is 112×112, and the feature embedding size is 512-D. The model is trained on a clean version of MS1M [103] that contains 5.8M of 80K identities. The presented solution uses random flip to augment the dataset during training.

SiMiT Lab – Masked Face Recognition–1 (SMT-MFR-1) employs LResNet-100E-IR model [107] trained with ArcFace loss function [80]. The model is originally trained on MS1MV2 dataset [103, 80]. SMT-MFR-1 solution depends on fine-tuning LResNet100E-IR using two real world masked face datasets- Real World Occluded Faces (ROF) [3] and MFR2 dataset [13]. MFR2 contains 296 images of 53 identities. ROF dataset is crawled from the internet and contains 678 masked face images and 1853 not-masked face images of 123 identities. The proposed solution is fine-tuned using the ROF dataset and a part of the MFR2 dataset (35 identities). The model process input image of size 112×112 to produce feature embedding of size 512-D. During the training, the training dataset is augmented using a horizontal flip augmentation method.

SiMiT Lab – Masked Face Recognition–2 (SMT-MFR-2) is conceptually identical to SMT-MFR-1. Different from SMT-MFR-1, the SMT-MFR-2 model is fine-tuned using the ROF dataset and the entire MFR2 dataset.

[2] https://github.com/JDAI-CV/FaceX-Zoo/blob/main/addition_module/face_mask_adding/FMA-3D/README.md

[3] https://github.com/ekremerakin/RealWorldOccludedFaces

LMI - SiMiT Lab - Masked Face Recognition - 1 (LMI-SMT-MFR-1) is a combination of two solutions- MFR-NMRE-F and SMT-MFR-1. First, the features are extracted separately by each of the solutions- MFR-NMRE-F and SMT-MFR-1. Then, the comparison scores are calculated for each solution. To combine the scores, cosine similarity measures are converted to euclidean distance in MFR-NMRE-F. The output of SMT-MFR-2 is euclidean distance. After this, the scores are normalized separately for each solution. Then, both scores are multiplied to generate the ensemble score.

LMI - SiMiT Lab - Masked Face Recognition - 2 (LMI-SMT-MFR-2) is also a combination of two solutions-MFR-NMRE-B and SMT-MFR-1. LMI-SMT-MFR-2 follows the same scores fusion method described in the LMI-SMT-MFR-1 solution.

Ignoring masks for accurate masked face recognition (IM-MFR) approach consists of two different training processes. The first, which aims to build a classification model, uses 6000 training identities from the VGGFace2 dataset [42] to minimize the cross-entropy while classifying these images. Each image had a probability of 65% of being masked. All training images are randomly resized and cropped to 224×224 In this solution, the masked creation method [201] uses the open implementation [4] by Boutros [26]. After achieving above 96% accuracy in the classification on the validation set, the last fully-connected layer was replaced with a fully connected layer with 512 outputs units. All the layers, except the newest one, are now frozen. The last layer is trained with and joint Triplet Loss and MSE for metric learning. The backbone network is a ResNet-50 [107]. The model is trained for 65k iterations.

Ignoring masks for accurate masked face recognition (IM-AMFR) follows the same training procedure, architecture, and loss function as in IM-MFR. The only difference is the number of training iterations where the IM-AMFR model is trained for 32k training iterations.

anonymous-1 and anonymous-2 employed FaceNet [234] as base architecture pre-trained on VGGFace2 [42]. MaskTheFace [13] is used to augment the LFW [114, 113] dataset and create a masked-face dataset. A masked version of each image in LFW is created. The FaceNet model is then fine-tuned using the augmented dataset. In the anonymous-1 solution, the model is fine-tuned using only masked face images. In the anonymous-2 solution, the model is fine-tuned using pairs of unmasked and masked images. For inference, the last layer of FaceNet consists of 512-dimensional embeddings, while the input size for both solutions is 160×160 pixels. One must note that the presented approach is reasonable, however, the verification accuracy presented in Section A.4 is extremely low, which might indicate an implementation error in the submission.

[4] `https://github.com/fdbtrs/MFR/blob/master/FaceMasked.py`

Baseline The baseline is chosen to put the submitted approaches in perspective of state-of-the-art face recognition model performance. The considered baseline is the ArcFace, which scored state-of-the-art performance on several face recognition evaluation benchmarks such as LFW 99.83% and YTF 98.02% by using Additive Angular Margin loss (ArcFace) to improve the discriminative ability of the face recognition model. We considered ArcFace based on ResNet-100 [107] architecture pretrained on refined version of the MS-Celeb-1M dataset [103] (MS1MV2).

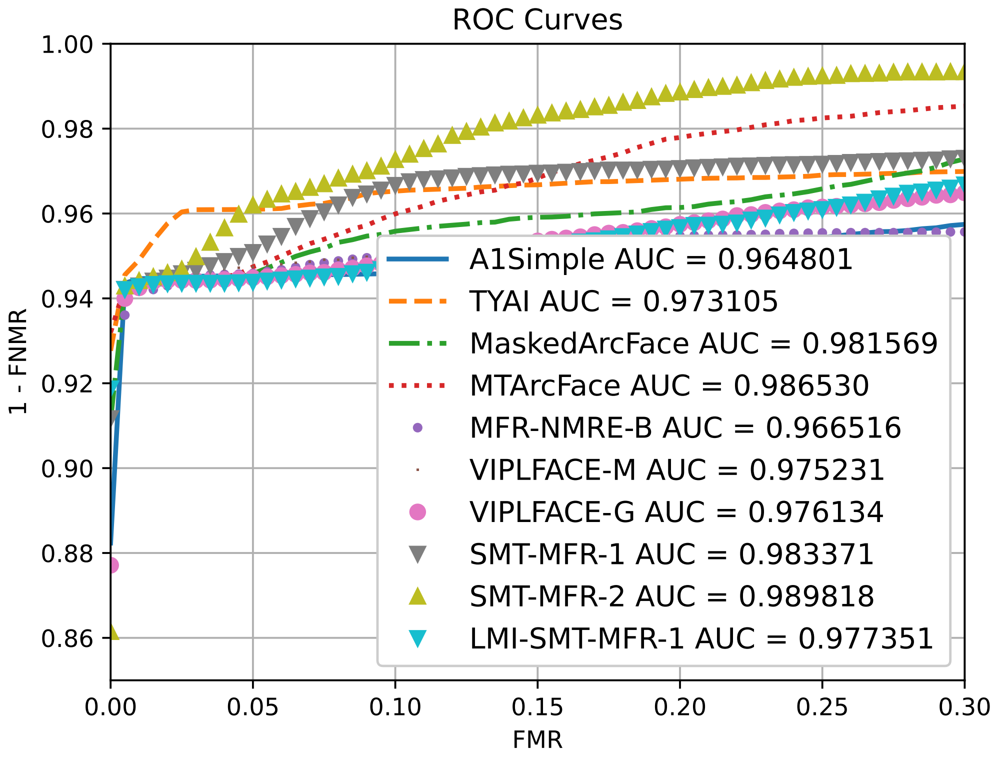
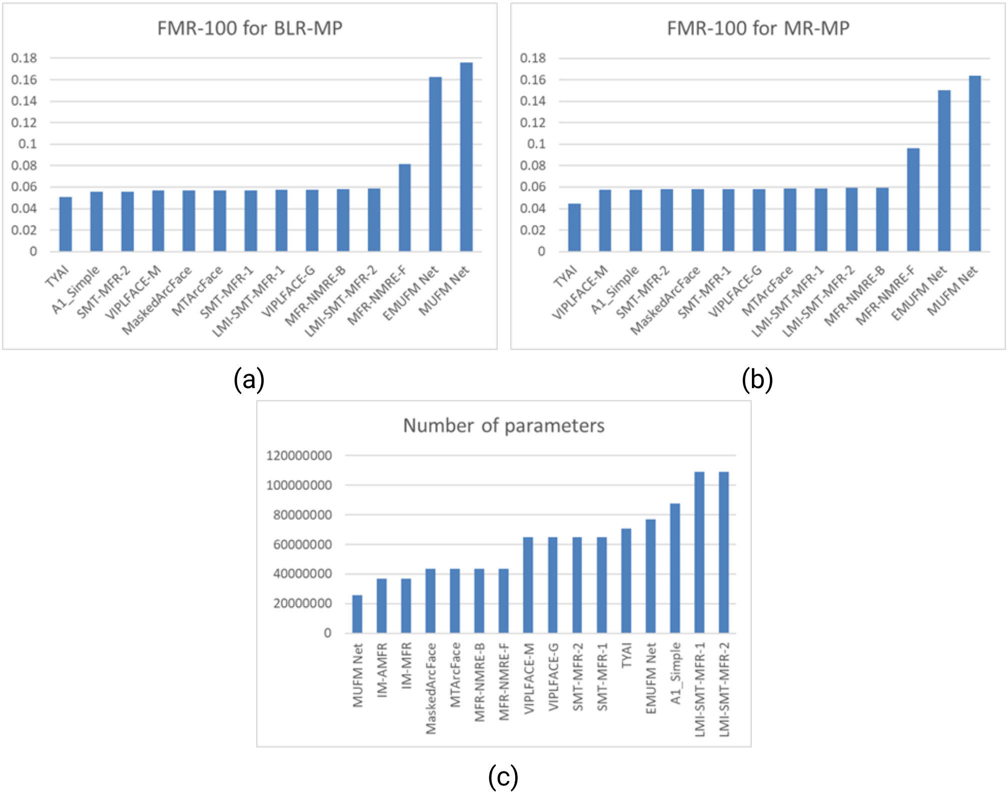


Figure A.2.: The ROC curve scored by the top 10 solutions in the BLR-MP experimental setting.

A.4. Results and analyses

This section presents comparative evaluation results of the submitted solution. We present first the achieved results on the BLR-MP evaluation setting and the model compactness. Then, the



(a) (b)

(c)

Figure A.3.: (a) The FMR100 scored by the top 14 solutions in the BLR-MP experimental setting. (b) The FMR100 scored by the top 14 solutions in the MR-MP experimental setting. (c) The number of trainable parameters in the top 16 solutions.

achieved results on the MR-MP evaluation setting are presented.

A.4.1. Not-masked vs. masked (BLR-MP)

Table A.3 presents comparative evaluation results achieved by the submitted solutions for BLR-MP evaluation setting and the model compactness. The results are reported and ranked based on the evaluation criteria described in Section A.2.2. From the reported results in Table A.3, the following observations can be made:

- Based on the defined evaluation criteria in Section A.2.2, the top-ranked solution based on the weighted Borda count is TYAI (rank 1), followed by MaskedArcFace and SMT-MFR-2 (rank 2) and then A1_Simple and VIPLFACE-M (rank 3).

- Most of the presented solutions achieved a competitive verification performance, in comparison to the baseline. Ten out of 18 solutions achieved higher verification performance than the baseline solution for the BLR-MP evaluation setting as reported in Table A.3 and Figure A.2. Figure A.2 presented the achieved verification performances in term of Receiver operating characteristic (ROC) curves by the top 10 solution on the BLR-MP experimental setting. The best verification performance in terms of FMR100 is achieved by the TYAI solution, where the achieved FMR100 was 0.05095 (Table A.3 and Figure A.3a).
- By comparing the verification performances reported in Table A.3 and the loss function utilized by each of the solutions reported in Table A.5, it is noted that the models trained with margin-based softmax loss (ArcFace or Sub-center ArcFace loss) achieved higher verification performance than the models trained with other loss functions including cross-entropy and triplet loss. This points out the generalizability brought by the nature of the marginal penalty that forces a better separability between classes (identities) and better compactness within classes.
- The solutions that achieved competitive FMR100 to the baseline solution have relatively higher separability between genuine and imposter scores (FDR) than other solutions that achieved relatively lower verification performance.
- Regarding model compactness, all solutions contain between 23M and 108M parameters as shown in Table A.3 and Figure A.3c. The top 3 ranked solutions have less than 87M parameters. This indicates that utilizing a larger and deeper deep learning model does not necessarily and solely lead to higher verification performance.
- The common strategy to improve the masked face recognition verification performance by the submitted solutions is to augment the training dataset with a simulated mask. All submitted solutions depended on training or fine-tuning face recognition model with masked face images (real or simulated). However, none of the presented solutions propose a solution that could be applied on top of the existing face recognition model, as in [26]. Furthermore, none of the presented solutions has clearly benefited from the mask labels included in the evaluation list. Four of the five top-ranked solutions utilized synthetically generated masks to augment the training dataset with simulated masked images. Utilizing such a method is usually easier than other solutions, such as using a real masked training dataset. Collecting a large-scale training dataset with pairs of not-masked/masked face images is, however, not a trivial task.

A.4.2. Masked vs. masked (MP-MR)

The verification performance of the experimental setting MR-MP for all submitted solutions is presented in Table A.4. The achieved verification performance is reported in terms of FMR100, FMR1000, and FDR. The presented results are ordered and ranked based on the achieved FMR100. It can be noted from the reported verification performance in Table A.4 that ten out of 18 solutions achieved better verification performance than the baseline solution when comparing masked reference to masked probe (MR-MP). TYAI solution achieved the best verification performance, followed by VIPLFACE-M and A1_Simple. By comparing the reported verification performance of BLR-MP evaluation setting (Table A.3) and the reported one of MR-MP (Table A.4), we can observe the following: a) Most of the solutions have higher separability between genuine and imposter scores (higher FDR) when both reference and probe are masked (MR-MP) than the case where only the probe are masked (BLR-MP). b) The top-ranked solutions in the MR-MP evaluation setting are also ranked among the top solutions in the BLR-MP evaluation setting.

A.5. Summary

Motivated by the pandemic-driven use of facial masks, the Masked Face Recognition Competitions (MFR 2021) was organized to motivate and evaluate face recognition solutions specially designed to perform well with masked faces. A total of 10 teams from 11 affiliations participated in the competition and contributed 18 solutions for the evaluation. The evaluation focused on not-masked vs. masked face verification accuracy, the masked vs. masked face verification accuracy, and the face recognition model compactness. Out of the 18 submitted solutions, 10 achieved lower verification error (FMR100) than the considered baseline. Most of the top-performing solutions used variations of the ArcFace loss and either real or simulated masked face databases in their training process. The lowest achieved FMR100 for the not-masked vs. masked evaluation was 5.1%, in comparison to an FMR100 of 6.0% scored by the baseline.

B. Publications

The author published 13 scientific publications as a first author, contributed additionally to 27 scientific publications and participated in four international competitions.

B.1. Publications

The thesis is partially based on the following publications:

1. Fadi Boutros, Patrick Siebke, Marcel Klemt, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Pocketnet: Extreme lightweight face recognition network using neural architecture search and multi-step knowledge distillation. *IEEE Access*, 2022

2. Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Elasticface: Elastic margin loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, Louisiana, USA, June 19-24, 2022*. Computer Vision Foundation / IEEE, 2022

3. Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Self-restrained triplet loss for accurate masked face recognition. *Pattern Recognition*, 124:108473, 2022

4. Fadi Boutros, Naser Damer, Kiran Raja, Florian Kirchbuchner, and Arjan Kuijper. Template-driven knowledge distillation for compact and accurate periocular biometrics deep-learning models. *Sensors*, 22(5), 2022

5. Fadi Boutros, Naser Damer, Meiling Fang, Florian Kirchbuchner, and Arjan Kuijper. Mixfacenets: Extremely efficient face recognition networks. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–8. IEEE, 2021

6. Fadi Boutros, Naser Damer, Jan Niklas Kolf, Kiran B. Raja, Florian Kirchbuchner, Raghavendra Ramachandra, Arjan Kuijper, Pengcheng Fang, Chao Zhang, Fei Wang, David Montero, Naiara Aginako, Basilio Sierra, Marcos Nieto, Mustafa Ekrem Erakin, Ugur Demir, Hazim Kemal Ekenel, Asaki Kataoka, Kohei Ichikawa, Shizuma Kubo, Jie Zhang, Mingjie He, Dan Han, Shiguang Shan, Klemen Grm, Vitomir Struc, Sachith Seneviratne, Nuran Kasthuriarachchi, Sanka Rasnayaka, Pedro C. Neto, Ana F. Sequeira, João Ribeiro Pinto, Mohsen Saffari, and Jaime S. Cardoso. MFR 2021: Masked face recognition competition. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021

7. Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Iris and periocular biometrics for head mounted displays: Segmentation, recognition, and synthetic data generation. *Image Vis. Comput.*, 104:104007, 2020

8. Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Fusing iris and periocular region for user verification in head mounted displays. In *IEEE 23rd International Conference on Information Fusion, FUSION 2020, Rustenburg, South Africa, July 6-9, 2020*, pages 1–8. IEEE, 2020

9. Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Periocular biometrics in head-mounted displays: A sample selection approach for better recognition. In *8th International Workshop on Biometrics and Forensics, IWBF 2020, Porto, Portugal, April 29-30, 2020*, pages 1–6. IEEE, 2020

10. Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. On benchmarking iris recognition within a head-mounted display for AR/VR applications. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–10. IEEE, 2020

11. Fadi Boutros, Naser Damer, Meiling Fang, Kiran B. Raja, Florian Kirchbuchner, and Arjan Kuijper. Compact models for periocular verification through knowledge distillation. In *BIOSIG 2020 - Proceedings of the 19th International Conference of the Biometrics Special Interest Group, online, 16.-18. September 2020*, volume P-306 of *LNI*, pages 291–298. Gesellschaft für Informatik e.V., 2020

12. Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Eye-mms: Miniature multi-scale segmentation network of key eye-regions in embedded applications. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3665–3670. IEEE, 2019

13. Fadi Boutros, Naser Damer, Philipp Terhörst, Florian Kirchbuchner, and Arjan Kuijper. Exploring the channels of multiple color spaces for age and gender estimation from face images. In *22th International Conference on Information Fusion, FUSION 2019, Ottawa, ON, Canada, July 2-5, 2019*, pages 1–8. IEEE, 2019

B.2. Co-author publications

1. Meiling Fang, Fadi Boutros, and Naser Damer. Intra and cross-spectrum iris presentation attack detection in the nir and visible domains using attention-based and pixel-wise supervised learning. In *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection and Vulnerability Assessment, Third Edition*, Advances in Computer Vision and Pattern Recognition. Springer, 2022

2. Naser Damer, César Augusto Fontanillo López, Meiling Fang, Noémie Spiller, Minh Vu Pham, and Fadi Boutros. Privacy-friendly synthetic data for the development of face morphing

attack detectors. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, Louisiana, USA, June 19-24, 2022*. Computer Vision Foundation / IEEE, 2022

3. Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. The overlapping effect and fusion protocols of data augmentation techniques in iris PAD. *Mach. Vis. Appl.*, 33(1):8, 2022

4. Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. An extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biometrics*, 2021

5. Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. Extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biom.*, 10(5):548–561, 2021

6. Pedro C. Neto, Fadi Boutros, João Ribeiro Pinto, Mohsen Saffari, Naser Damer, Ana F. Sequeira, and Jaime S. Cardoso. My eyes are up here: Promoting focus on uncovered regions in masked face recognition. In *Proceedings of the 20th International Conference of the Biometrics Special Interest Group, BIOSIG 2021, Digital Conference, September 15-17, 2021*, volume P-315 of *LNI*, pages 21–30. Gesellschaft für Informatik e.V., 2021

7. Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Cross-database and cross-attack iris presentation attack detection using micro stripes analyses. *Image Vis. Comput.*, 105:104057, 2021

8. Pedro C. Neto, Fadi Boutros, João Ribeiro Pinto, Mohsen Saffari, Naser Damer, Ana F. Sequeira, and Jaime S. Cardoso. My eyes are up here: Promoting focus on uncovered regions in masked face recognition. In *Proceedings of the 20th International Conference of the Biometrics Special Interest Group, BIOSIG 2021, Digital Conference, September 15-17, 2021*, volume P-315 of *LNI*, pages 21–30. Gesellschaft für Informatik e.V., 2021

9. Meiling Fang, Fadi Boutros, Arjan Kuijper, and Naser Damer. Partial attack supervision and regional weighted inference for masked face presentation attack detection. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021

10. Marco Huber, Fadi Boutros, Florian Kirchbuchner, and Naser Damer. Mask-invariant face recognition through template-level knowledge distillation. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021

11. Pedro C. Neto, Fadi Boutros, João Ribeiro Pinto, Naser Darner, Ana F. Sequeira, and Jaime S. Cardoso. Focusface: Multi-task contrastive learning for masked face recognition. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021

12. Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Iris presentation attack detection by attention-based and deep pixel-wise binary supervision network. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–8. IEEE, 2021

13. Naser Damer, Kiran B. Raja, Marius Süßmilch, Sushma Venkatesh, Fadi Boutros, Meiling Fang, Florian Kirchbuchner, Raghavendra Ramachandra, and Arjan Kuijper. Regenmorph: Visibly realistic GAN generated face morphing attacks by attack re-generation. In *Advances in Visual Computing - 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part I*, volume 13017 of *Lecture Notes in Computer Science*, pages 251–264. Springer, 2021

14. Naser Damer, Noémie Spiller, Meiling Fang, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. PW-MAD: pixel-wise supervision for generalized face morphing attack detection. In *Advances in Visual Computing - 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part I*, volume 13017 of *Lecture Notes in Computer Science*, pages 291–304. Springer, 2021

15. Sandip Purnapatra, Nic Smalt, Keivan Bahmani, Priyanka Das, David Yambay, Amir Mohammadi, Anjith George, Thirimachos Bourlai, Sébastien Marcel, Stephanie Schuckers, Meiling Fang, Naser Damer, Fadi Boutros, Arjan Kuijper, Alperen Kantarci, Basar Demir, Zafer Yildiz, Zabi Ghafoory, Hasan Dertli, Hazim Kemal Ekenel, Son Vu, Vassilis Christophides, Dashuang Liang, Guanghao Zhang, Zhanlong Hao, Junfu Liu, Yufeng Jin, Samo Liu, Samuel Huang, Salieri Kuei, Jag Mohan Singh, and Raghavendra Ramachandra. Face liveness detection competition (livdet-face) - 2021. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021

16. Caiyong Wang, Yunlong Wang, Kunbo Zhang, Jawad Muhammad, Tianhao Lu, Qi Zhang, Qichuan Tian, Zhaofeng He, Zhenan Sun, Yiwen Zhang, Tianbao Liu, Wei Yang, Dongliang Wu, Yingfeng Liu, Ruiye Zhou, Huihai Wu, Hao Zhang, Junbao Wang, Jiayi Wang, Wantong Xiong, Xueyu Shi, Shao Zeng, Peihua Li, Haodong Sun, Jing Wang, Jiale Zhang, Qi Wang, Huijie Wu, Xinhui Zhang, Haiqing Li, Yu Chen, Liang Chen, Menghan Zhang, Ye Sun, Zhiyong Zhou, Fadi Boutros, Naser Damer, Arjan Kuijper, Juan E. Tapia, Andres Valenzuela, Christoph Busch, Gourav Gupta, Kiran B. Raja, Xi Wu, Xiaojie Li, Jingfu Yang, Hongyan Jing, Xin Wang, Bin Kong, Youbing Yin, Qi Song, Siwei Lyu, Shu Hu, Leon Premk, Matej Vitek, Vitomir Struc, Peter Peer, Jalil Nourmohammadi-Khiarak, Farhang Jaryani, Samaneh Salehi Nasab, Seyed Naeim Moafinejad, Yasin Amini, and Morteza Noshad. NIR iris challenge evaluation in non-cooperative environments: Segmentation and localization. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021

17. Priyanka Das, Joseph McGrath, Zhaoyuan Fang, Aidan Boyd, Ganghee Jang, Amir Mohammadi, Sandip Purnapatra, David Yambay, Sébastien Marcel, Mateusz Trokielewicz, Piotr Maciejewicz, Kevin W. Bowyer, Adam Czajka, Stephanie Schuckers, Juan E. Tapia, Sebastian Gonzalez, Meiling Fang, Naser Damer, Fadi Boutros, Arjan Kuijper, Renu Sharma, Cunjian

Chen, and Arun Ross. Iris liveness detection competition (livdet-iris) - the 2020 edition. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–9. IEEE, 2020

18. Matej Vitek, Abhijit Das, Y. Pourcenoux, A. Missler, C. Paumier, Sumanta Das, Ishita De Ghosh, Diego Rafael Lucio, Luiz A. Zanlorensi, David Menotti, Fadi Boutros, Naser Damer, Jonas Henry Grebe, Arjan Kuijper, J. Hu, Y. He, C. Wang, H. Liu, Y. Wang, Z. Sun, Dailé Osorio Roig, Christian Rathgeb, Christoph Busch, Juan E. Tapia, Andres Valenzuela, G. Zampoukis, Lazaros T. Tsochatzidis, Ioannis Pratikakis, S. Nathan, R. Suganya, Vineet Mehta, Abhinav Dhall, Kiran B. Raja, G. Gupta, Jalil Nourmohammadi-Khiarak, M. Akbari-Shahper, Farhang Jaryani, Meysam Asgari-Chenaghlu, R. Vyas, S. Dakshit, Peter Peer, Umapada Pal, and Vitomir Struc. SSBC 2020: Sclera segmentation benchmarking competition in the mobile environment. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–10. IEEE, 2020
19. Naser Damer, Jonas Henry Grebe, Cong Chen, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. The effect of wearing a mask on face recognition performance: an exploratory study. In *BIOSIG 2020 - Proceedings of the 19th International Conference of the Biometrics Special Interest Group, online, 16.-18. September 2020*, volume P-306 of *LNI*, pages 1–10. Gesellschaft für Informatik e.V., 2020
20. Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Deep learning multi-layer fusion for an accurate iris presentation attack detection. In *IEEE 23rd International Conference on Information Fusion, FUSION 2020, Rustenburg, South Africa, July 6-9, 2020*, pages 1–8. IEEE, 2020
21. Naser Damer, Fadi Boutros, Alexandra Mosegui Saladie, Florian Kirchbuchner, and Arjan Kuijper. Realistic dreams: Cascaded enhancement of gan-generated images with an example in face morphing attacks. In *10th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2019, Tampa, FL, USA, September 23-26, 2019*, pages 1–10. IEEE, 2019
22. Khawla Mallat, Naser Damer, Fadi Boutros, and Jean-Luc Dugelay. Robust face authentication based on dynamic quality-weighted comparison of visible and thermal-to-visible images to visible enrollments. In *22th International Conference on Information Fusion, FUSION 2019, Ottawa, ON, Canada, July 2-5, 2019*, pages 1–8. IEEE, 2019
23. Khawla Mallat, Naser Damer, Fadi Boutros, Arjan Kuijper, and Jean-Luc Dugelay. Cross-spectrum thermal to visible face recognition based on cascaded image synthesis. In *2019 International Conference on Biometrics, ICB 2019, Crete, Greece, June 4-7, 2019*, pages 1–8. IEEE, 2019
24. Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. D-id-net: Two-stage domain and identity learning for identity-preserving image generation from semantic segmentation. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3677–3682. IEEE, 2019

25. Kiran B. Raja, Naser Damer, Raghavendra Ramachandra, Fadi Boutros, and Christoph Busch. Cross-spectral periocular recognition by cascaded spectral image transformation. In *2019 IEEE International Conference on Imaging Systems and Techniques, IST 2019, Abu Dhabi, United Arab Emirates, December 9-10, 2019*, pages 1–7. IEEE, 2019

26. Naser Damer, Viola Boller, Yaza Wainakh, Fadi Boutros, Philipp Terhörst, Andreas Braun, and Arjan Kuijper. Detecting face morphing attacks by analyzing the directed distances of facial landmarks shifts. In *Pattern Recognition - 40th German Conference, GCPR 2018, Stuttgart, Germany, October 9-12, 2018, Proceedings*, volume 11269 of *Lecture Notes in Computer Science*, pages 518–534. Springer, 2018

27. Naser Damer, Fadi Boutros, Philipp Terhörst, Andreas Braun, and Arjan Kuijper. P-score: Performance aligned normalization and an evaluation in score-level multi-biometric fusion. In *26th European Signal Processing Conference, EUSIPCO 2018, Roma, Italy, September 3-7, 2018*, pages 1402–1406. IEEE, 2018

B.3. Under review papers

1. Fadi Boutros, Naser Damer, and Arjan Kuijper. Quantface: Towards lightweight face recognition by synthetic data low-bit quantization. 2022 (under review - ICPR 2022)

2. Fadi Boutros, Meiling Fang, Marcel Klemt, Biying Fu, and Naser Damer. CR-FIQA: face image quality assessment by learning sample relative classifiability. *CoRR*, abs/2112.06592, 2021 (under review - ECCV 2022)

3. Naser Damer, Fadi Boutros, Marius Süßmilch, Meiling Fang, Florian Kirchbuchner, and Arjan Kuijper. Masked face recognition: Human vs. machine. *CoRR*, abs/2103.01924, 2021 (minor revision - IET Biometrics)

4. Pedro C. Neto, João David Pinto, Fadi Boutros, Naser Damer, Ana Sequeira, and Jaime Cardoso. Beyond masks: On the generalization of masked face recognition models to occluded face recognition. 2022 (under review - IET Biometrics)

5. Matej Vitek, Abhijit Das, Diego Rafael Lucio, Luiz Antonio Zanlorensi Jr., David Menotti, Jalil Nourmohammadi Khiarak, Mohsen Akbari Shahpar, Meysam Asgari-Chenaghlu, Farhang Jaryani, Juan E. Tapia, Andres Valenzuela, Caiyong Wang, Yunlong Wang, Zhaofeng He, Zhenan Sun, Fadi Boutros, Naser Damer, Jonas Henry Grebe, Arjan Kuijper, Kiran Raja, Gourav Gupta, Georgios Zampoukis, Lazaros Tsochatzidis, Ioannis Pratikakis, S. V. Aruna Kumar, B. S. Harish, Umapada Pal, Peter Peer, and and Vitomir Štruc. Exploring bias in sclera segmentation models: A group evaluation approach. 2022 (under review - TIFS)

B.4. Competitions

The author participated in four international competitions [71, 265, 219, 267].

1. The first place in image category and the second place in video category at Face Liveness Detection Competition (LivDet-Face) - 2021 [219]
2. The twenty-third place at NIR Iris Challenge Evaluation in Non-cooperative Environments: Segmentation and Localization [267]
3. The second place at Iris Liveness Detection Competition (LivDet-Iris) - 2020 [71]
4. The thrid and fourth places at SBC 2020: Sclera segmentation benchmarking competition in the mobile environment [265]

C. Supervising thesis

The following bachelor and master theses were supervised by the author, and the results of these works were partially used as an input into this thesis.

1. Ha Hai Tu, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Reducing Deep Face Recognition Model Memory Footprint via Quantization. Bachelor Thesis, TU Darmstadt, 2022.

2. Patrick Siebke, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Neural Architecture Search for Mobile Face Recognition. Bachelor Thesis, TU Darmstadt, 2021.

3. Marcel Klemt, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Reducing Deep Face Recognition Model Size by Knowledge Distillation. Bachelor Thesis, TU Darmstadt, 2020.

4. Wei-Hung Hsu, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Learned Data Augmentation for Model Optimization. Master Thesis, TU Darmstadt, 2020.

5. Philipp lukas kubon, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Person ubiquitous identify in smart living environment. Bachelor Thesis, TU Darmstadt, 2020.

6. Salahedine Youssef, Fadi Boutros (supervisor) and Prof. Dr. Arjan Kuijper (supervisor). Investigate the use of ultrasonic sensor for human pose estimation in smart environments. Bachelor Thesis, TU Darmstadt, 2019.

D. Curriculum vitae

Personal data

Name	Fadi Boutros
Birth date	27.05.1984
Email	`fadi.boutros@igd.fraunhofer.de`
Goolge scholar	`https://scholar.google.com/citations?user=C-zewBgAAAAJ&hl=en`
Computer science bibliography	`https://dblp.org/pid/231/2501.html`

Education

2016-2019	Master of Science in Distributed Software System, Technical University of Darmstadt, Darmstadt, Germany
2009-2012	Master in Web Science, Syrian Virtual University, Damascus, Syria
2001-2007	Bachelor of Science in Software and Information System, Faculty of Informatics Engineering, Aleppo University, Aleppo, Syria

Work experience

Sep 2018 - current	Researcher, Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany

Projects

2019 - current	ATHENE 2.0 EmBiom: Mission Next Generation Biometric Systems Embedded Biometrics

2019 - 2020	ACTIVAGE: ACTivating InnoVative IoT smart living environments for AGEing well

Oct 2016 - Sep 2018 Student research assistant - Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany

Projects

2017 - 2019	ACTIVAGE: ACTivating InnoVative IoT smart living environments for AGEing well
2016 - 2017	Semantisches Smart Living: Semantisches Smart Living Anwendung auf Basis openWRT

Aug 2018 - Feb 2019 Master thesis work, Reducing Ethnic Bias of Faces Recognition by Ethnic Augmentation, Fraunhofer Institute for Computer Graphics Research IGD, Darmstadt, Germany

Academic activities

Awards

1. Best master thesis award, Computer Graphics Night 2019 Thesis title: “Reducing Ethnic Bias of Faces Recognition by Ethnic Augmentation” Authors: Fadi Boutros, Naser Damer (supervisor), Arjan Kuijper (supervisor)
2. The 5th award in master thesis category in the CAST-Förderpreis IT-Sicherheit 2019 Thesis title: “Reducing Ethnic Bias of Faces Recognition by Ethnic Augmentation” Authors: Fadi Boutros, Naser Damer (supervisor), Arjan Kuijper (supervisor)
3. Best poster award in the 19th International Conference of the Biometrics Special Interest Group (BIOSIG-2020). Authors: Fadi Boutros, Naser Damer, Meiling Fang, Kiran B. Raja, Florian Kirchbuchner, Arjan Kuijper Paper title: Compact Models for Periocular Verification Through Knowledge Distillation.
4. Best paper award in the 19th International Conference of the Biometrics Special Interest Group (BIOSIG-2020) Authors: Naser Damer, Jonas Henry Grebe, Cong Chen, Fadi Boutros, Florian Kirchbuchner, Arjan Kuijper Paper title: The Effect of Wearing a Mask on Face Recognition Performance: an Exploratory Study
5. Best paper award, Impact on Society, Computer Graphics Night 2020 Paper title: Detecting Face Morphing Attacks by Analyzing the Directed Distances of Facial Landmarks ShiftsAuthors: Naser Damer,

Viola Boller, Yaza Wainakh, Fadi Boutros, Phlipp Terhörst, Andreas Braun, Arjan Kuijper

6. Best paper award, Impact on Business, Computer Graphics Night 2021 Paper title: Iris and Periocular Biometrics for Head Mounted Displays: Segmentation, Recognition, and Synthetic Data Generation Authors: Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, Arjan Kuijper

Honor

Scholarship for 17th international summer school for advanced studies on "Biometrics Forensics and Identity Science for Human-Centered Applications", 7 -12 July 2020

Teaching

1. Teaching at the lecture of "Biometric recognition and security" 2021 at TU Darmstadt
2. Supervisor for a number of practical course works within "Visual Computing Lab", GRIS, TU Darmstadt 2021/2022

Bibliography

[1] Automotive biometric market. https://www.alliedmarketresearch.com/automotive-biometric-market. Accessed: 2017-05.

[2] Mobile id. https://na.idemia.com/dmv/mobile-id/. Accessed: 2022.

[3] *Handbook of Face Recognition, 2nd Edition*. Springer, 2011.

[4] *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 2014.

[5] ARESIBO project: http://aresibo.eu/, 2019.

[6] OpenEDS Challenge: https://research.fb.com/programs/openeds-challenge/, 2019.

[7] Mahmoud Afifi and Abdelrahman Abdelhamed. Afif^4: Deep gender classification based on adaboost-based fusion of isolated facial features and foggy faces. *J. Vis. Commun. Image Represent.*, 62:77–86, 2019.

[8] Faisal AlGashaam, Kien Nguyen, Wageeh Boles, and Vinod Chandran. Periocular recognition under expression variation using higher order spectral features. In *2015 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6. IEEE, 2015.

[9] Fernando Alonso-Fernandez, Anna Mikaelyan, and Josef Bigun. Comparison and fusion of multiple iris and periocular matchers using near-infrared and visible images. In *3rd International Workshop on Biometrics and Forensics (IWBF 2015)*, pages 1–6. IEEE, 2015.

[10] Dario Amodei and Danny Hernandez. Ai and compute. https://openai.com/blog/ai-and-compute/. Accessed: 2018-05-16.

[11] Xiang An, Xuhan Zhu, Yuan Gao, Yang Xiao, Yongle Zhao, Ziyong Feng, Lan Wu, Bin Qin, Ming Zhang, Debing Zhang, and Ying Fu. Partial FC: training 10 million identities on a single machine. In *ICCVW*, pages 1445–1449. IEEE, 2021.

[12] Grigory Antipov, Moez Baccouche, and Jean-Luc Dugelay. Face aging with conditional generative adversarial networks. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 2089–2093. IEEE, 2017.

[13] Aqeel Anwar and Arijit Raychowdhury. Masked face recognition for secure authentication. 2020.

[14] Ognjen Arandjelović. Reimagining the central challenge of face recognition: Turning a problem into an advantage. *Pattern Recognition*, 83:388–400, 2018.

[15] Arun Vemury and ake Hasselgren and John Howard and Yevgeniy Sirotin. 2020 biometric rally results - face masks face recognition performance. `https://mdtf.org/Rally2020/Results2020`, 2020. Last accessed: June 28, 2022.

[16] Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

[17] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017.

[18] Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7948–7956, 2019.

[19] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Towards open-set identity preserving face synthesis. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6713–6722. Computer Vision Foundation / IEEE Computer Society, 2018.

[20] Diego Bastias, Claudio A Perez, Daniel P Benalcazar, and Kevin W Bowyer. A method for 3d iris reconstruction from multiple 2d near-infrared images. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 503–509. IEEE, 2017.

[21] Borut Batagelj, Peter Peer, Vitomir Štruc, and Simon Dobrišek. How to correctly detect face-masks for covid-19 from visual information? *Applied Sciences*, 11(5), 2021.

[22] Shabab Bazrafkan, Shejin Thavalengal, and Peter Corcoran. An end to end deep neural network for iris segmentation in unconstrained scenarios. *Neural Networks*, 106:79–95, 2018.

[23] Fadi Boutros, Naser Damer, Meiling Fang, Florian Kirchbuchner, and Arjan Kuijper. Mixfacenets: Extremely efficient face recognition networks. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–8. IEEE, 2021.

[24] Fadi Boutros, Naser Damer, Meiling Fang, Kiran B. Raja, Florian Kirchbuchner, and Arjan Kuijper. Compact models for periocular verification through knowledge distillation. In *BIOSIG 2020 - Proceedings of the 19th International Conference of the Biometrics Special*

Interest Group, online, 16.-18. September 2020, volume P-306 of *LNI*, pages 291–298. Gesellschaft für Informatik e.V., 2020.

[25] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Eye-mms: Miniature multi-scale segmentation network of key eye-regions in embedded applications. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3665–3670. IEEE, 2019.

[26] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Unmasking face embeddings by self-restrained triplet loss for accurate masked face recognition. *CoRR*, abs/2103.01716, 2021.

[27] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Elasticface: Elastic margin loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, Louisiana, USA, June 19-24, 2022*. Computer Vision Foundation / IEEE, 2022.

[28] Fadi Boutros, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Self-restrained triplet loss for accurate masked face recognition. *Pattern Recognition*, 124:108473, 2022.

[29] Fadi Boutros, Naser Damer, Jan Niklas Kolf, Kiran B. Raja, Florian Kirchbuchner, Raghavendra Ramachandra, Arjan Kuijper, Pengcheng Fang, Chao Zhang, Fei Wang, David Montero, Naiara Aginako, Basilio Sierra, Marcos Nieto, Mustafa Ekrem Erakin, Ugur Demir, Hazim Kemal Ekenel, Asaki Kataoka, Kohei Ichikawa, Shizuma Kubo, Jie Zhang, Mingjie He, Dan Han, Shiguang Shan, Klemen Grm, Vitomir Struc, Sachith Seneviratne, Nuran Kasthuriarachchi, Sanka Rasnayaka, Pedro C. Neto, Ana F. Sequeira, João Ribeiro Pinto, Mohsen Saffari, and Jaime S. Cardoso. MFR 2021: Masked face recognition competition. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021.

[30] Fadi Boutros, Naser Damer, and Arjan Kuijper. Quantface: Towards lightweight face recognition by synthetic data low-bit quantization. 2022.

[31] Fadi Boutros, Naser Damer, Kiran Raja, Florian Kirchbuchner, and Arjan Kuijper. Template-driven knowledge distillation for compact and accurate periocular biometrics deep-learning models. *Sensors*, 22(5), 2022.

[32] Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Fusing iris and periocular region for user verification in head mounted displays. In *IEEE 23rd International Conference on Information Fusion, FUSION 2020, Rustenburg, South Africa, July 6-9, 2020*, pages 1–8. IEEE, 2020.

[33] Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Iris and periocular biometrics for head mounted displays: Segmentation, recognition, and synthetic data generation. *Image Vis. Comput.*, 104:104007, 2020.

[34] Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. On benchmarking iris recognition within a head-mounted display for AR/VR applications. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–10. IEEE, 2020.

[35] Fadi Boutros, Naser Damer, Kiran B. Raja, Raghavendra Ramachandra, Florian Kirchbuchner, and Arjan Kuijper. Periocular biometrics in head-mounted displays: A sample selection approach for better recognition. In *8th International Workshop on Biometrics and Forensics, IWBF 2020, Porto, Portugal, April 29-30, 2020*, pages 1–6. IEEE, 2020.

[36] Fadi Boutros, Naser Damer, Philipp Terhörst, Florian Kirchbuchner, and Arjan Kuijper. Exploring the channels of multiple color spaces for age and gender estimation from face images. In *22th International Conference on Information Fusion, FUSION 2019, Ottawa, ON, Canada, July 2-5, 2019*, pages 1–8. IEEE, 2019.

[37] Fadi Boutros, Meiling Fang, Marcel Klemt, Biying Fu, and Naser Damer. CR-FIQA: face image quality assessment by learning sample relative classifiability. *CoRR*, abs/2112.06592, 2021.

[38] Fadi Boutros, Patrick Siebke, Marcel Klemt, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Pocketnet: Extreme lightweight face recognition network using neural architecture search and multi-step knowledge distillation. *IEEE Access*, 2022.

[39] Kevin W Bowyer, Karen Hollingsworth, and Patrick J Flynn. Image understanding for iris biometrics: A survey. *Computer vision and image understanding*, 110(2):281–307, 2008.

[40] Jaylon Brinkley. Global biometrics market, forecast to 2024. `https://www.frost.com/news/press-releases/global-biometrics-to-reach-45-96-billion-as-the-implementation-of-egovernance-by-the-public-sector-rises/`. Accessed: 2020-03-19.

[41] Han Cai, Ligeng Zhu, and Song Han. Proxylessnas: Direct neural architecture search on target task and hardware. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[42] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *13th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2018, Xi'an, China, May 15-19, 2018*, pages 67–74. IEEE Computer Society, 2018.

[43] Weilong Chai, Weihong Deng, and Haifeng Shen. Cross-generating GAN for facial identity preserving. In *13th IEEE International Conference on Automatic Face & Gesture Recognition, FG 2018, Xi'an, China, May 15-19, 2018*, pages 130–134. IEEE Computer Society, 2018.

[44] Christophe Champod and Massimo Tistarelli. Biometric technologies for forensic science and policing: State of the art. In *Handbook of Biometrics for Forensic Science*, Advances in Computer Vision and Pattern Recognition, pages 1–15. Springer, 2017.

[45] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. PairedCycleGAN: Asymmetric style transfer for applying and removing makeup. In *CVPR 2018*, June 2018.

[46] Jianxu Chen, Feng Shen, Danny Ziyi Chen, and Patrick J Flynn. Iris recognition based on human-interpretable features. *IEEE Transactions on Information Forensics and Security*, 11(7):1476–1485, 2016.

[47] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII*, volume 11211 of *Lecture Notes in Computer Science*, pages 833–851. Springer, 2018.

[48] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 1520–1529. IEEE Computer Society, 2017.

[49] Sheng Chen, Yang Liu, Xiang Gao, and Zhen Han. Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices. In *CCBR 2018, Urumqi, China, August 11-12, 2018, Proceedings*, volume 10996 of *Lecture Notes in Computer Science*, pages 428–438. Springer, 2018.

[50] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, pages 539–546. IEEE Computer Society, 2005.

[51] Cognitec. Cognitec: The face recognition company. "https://cognitec.com/", 2021.

[52] EUROPEAN COMMISSION. Technical study on smart borders. https://ec.europa.eu/home-affairs/system/files/2016-12/smart_borders_executive_summary_en.pdf, 2014.

[53] Daimler. Mercedes-benz strategy update 2020. https://www.daimler.com/dokumente/investoren/praesentationen/daimler-ir-mercedes-benz-strategy-update-2020-presentation.pdf. Accessed: 2020-10-06.

[54] Naser Damer, Viola Boller, Yaza Wainakh, Fadi Boutros, Philipp Terhörst, Andreas Braun, and Arjan Kuijper. Detecting face morphing attacks by analyzing the directed distances of facial landmarks shifts. In *Pattern Recognition - 40th German Conference, GCPR 2018,*

Stuttgart, Germany, October 9-12, 2018, Proceedings, volume 11269 of *Lecture Notes in Computer Science*, pages 518–534. Springer, 2018.

[55] Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. D-id-net: Two-stage domain and identity learning for identity-preserving image generation from semantic segmentation. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3677–3682. IEEE, 2019.

[56] Naser Damer, Fadi Boutros, Alexandra Mosegui Saladie, Florian Kirchbuchner, and Arjan Kuijper. Realistic dreams: Cascaded enhancement of gan-generated images with an example in face morphing attacks. In *10th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2019, Tampa, Florida, USA, September 23-26, 2019*. IEEE, 2019.

[57] Naser Damer, Fadi Boutros, Alexandra Mosegui Saladie, Florian Kirchbuchner, and Arjan Kuijper. Realistic dreams: Cascaded enhancement of gan-generated images with an example in face morphing attacks. In *10th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2019, Tampa, FL, USA, September 23-26, 2019*, pages 1–10. IEEE, 2019.

[58] Naser Damer, Fadi Boutros, Marius Süßmilch, Meiling Fang, Florian Kirchbuchner, and Arjan Kuijper. Masked face recognition: Human vs. machine. *CoRR*, abs/2103.01924, 2021.

[59] Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. An extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biometrics*, 2021.

[60] Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. Extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biom.*, 10(5):548–561, 2021.

[61] Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. Extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biometrics*, n/a(n/a).

[62] Naser Damer, Fadi Boutros, Marius Süßmilch, Florian Kirchbuchner, and Arjan Kuijper. Extended evaluation of the effect of real and simulated masks on face recognition performance. *IET Biometrics*, 10(5):548–561, 2021.

[63] Naser Damer, Fadi Boutros, Philipp Terhörst, Andreas Braun, and Arjan Kuijper. P-score: Performance aligned normalization and an evaluation in score-level multi-biometric fusion. In *26th European Signal Processing Conference, EUSIPCO 2018, Roma, Italy, September 3-7, 2018*, pages 1402–1406. IEEE, 2018.

[64] Naser Damer, Jonas Henry Grebe, Cong Chen, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. The effect of wearing a mask on face recognition performance: an exploratory study. In *BIOSIG 2020 - Proceedings of the 19th International Conference of the Biometrics Special Interest Group, online, 16.-18. September 2020*, volume P-306 of *LNI*, pages 1–10. Gesellschaft für Informatik e.V., 2020.

[65] Naser Damer, César Augusto Fontanillo López, Meiling Fang, Noémie Spiller, Minh Vu Pham, and Fadi Boutros. Privacy-friendly synthetic data for the development of face morphing attack detectors. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, Louisiana, USA, June 19-24, 2022*. Computer Vision Foundation / IEEE, 2022.

[66] Naser Damer, Kiran B. Raja, Marius Süßmilch, Sushma Venkatesh, Fadi Boutros, Meiling Fang, Florian Kirchbuchner, Raghavendra Ramachandra, and Arjan Kuijper. Regenmorph: Visibly realistic GAN generated face morphing attacks by attack re-generation. In *Advances in Visual Computing - 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part I*, volume 13017 of *Lecture Notes in Computer Science*, pages 251–264. Springer, 2021.

[67] Naser Damer, Alexandra Mosegui Saladie, Andreas Braun, and Arjan Kuijper. Morgan: Recognition vulnerability and attack detectability of face morphing attacks created by generative adversarial network. In *9th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2018, Redondo Beach, CA, USA, October 22-25, 2018*, pages 1–10. IEEE, 2018.

[68] Naser Damer, Noémie Spiller, Meiling Fang, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. PW-MAD: pixel-wise supervision for generalized face morphing attack detection. In *Advances in Visual Computing - 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part I*, volume 13017 of *Lecture Notes in Computer Science*, pages 291–304. Springer, 2021.

[69] Abhijit Das, Umapada Pal, Michael Blumenstein, and zhun sun. Sclera segmentation benchmarking competition in cross-resolution environment. In *International Conference on Biometrics, ICB 2019, 4-7 June, 2019, Crete, Greece*. IEEE, 2019.

[70] Abhijit Das, Umapada Pal, Miguel A. Ferrer, and Michael Blumenstein. SSBC 2015: Sclera segmentation benchmarking competition. In *IEEE 7th International Conference on Biometrics Theory, Applications and Systems, BTAS 2015, Arlington, VA, USA, September 8-11, 2015*, pages 1–6. IEEE, 2015.

[71] Priyanka Das, Joseph McGrath, Zhaoyuan Fang, Aidan Boyd, Ganghee Jang, Amir Mohammadi, Sandip Purnapatra, David Yambay, Sébastien Marcel, Mateusz Trokielewicz, Piotr Maciejewicz, Kevin W. Bowyer, Adam Czajka, Stephanie Schuckers, Juan E. Tapia, Sebastian Gonzalez, Meiling Fang, Naser Damer, Fadi Boutros, Arjan Kuijper, Renu Sharma, Cunjian Chen, and Arun Ross. Iris liveness detection competition (livdet-iris) - the 2020

edition. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–9. IEEE, 2020.

[72] John Daugman. Iris recognition border-crossing system in the uae. *International Airport Review*, 8(2), 2004.

[73] John Daugman. How iris recognition works. In *The essential guide to image processing*, pages 715–739. Elsevier, 2009.

[74] John Daugman. Iris recognition at airports and border crossings. *Encyclopedia of Biometrics*, pages 998–1004, 2015.

[75] Tiago de Freitas Pereira and Sébastien Marcel. Periocular biometrics in mobile environment. In *IEEE 7th International Conference on Biometrics Theory, Applications and Systems, BTAS 2015, Arlington, VA, USA, September 8-11, 2015*, pages 1–7. IEEE, 2015.

[76] Jose Sanchez del Rio Saez, Daniela Moctezuma, Cristina Conde, Isaac Martín de Diego, and Enrique Cabello. Automated border control e-gates and facial recognition systems. *Comput. Secur.*, 62:49–72, 2016.

[77] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR 2009, 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009.

[78] Jiankang Deng, Jia Guo, Tongliang Liu, Mingming Gong, and Stefanos Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In *ECCV (11)*, volume 12356 of *Lecture Notes in Computer Science*, pages 741–757. Springer, 2020.

[79] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *CVPR*, pages 5202–5211. IEEE, 2020.

[80] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *IEEE CVPR, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4690–4699, 2019.

[81] Jiankang Deng, Jia Guo, Debing Zhang, Yafeng Deng, Xiangju Lu, and Song Shi. Lightweight face recognition challenge. In *2019 IEEE/CVF ICCV, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 2638–2646. IEEE, 2019.

[82] Department of Homeland Security. Biometric Technology Rally at MDTF. Technical report, 2020. Last accessed: June 28, 2022.

[83] Piotr Dollár, Mannat Singh, and Ross B. Girshick. Fast and accurate model scaling. *CoRR*, abs/2103.06877, 2021.

[84] Yueqi Duan, Jiwen Lu, and Jie Zhou. Uniformface: Learning deep equidistributed representation for face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3415–3424. Computer Vision Foundation / IEEE, 2019.

[85] e-Aadhaar - Unique Identification Authority of India. https://eaadhaar.uidai.gov.in/, 2015.

[86] Meiling Fang, Fadi Boutros, and Naser Damer. Intra and cross-spectrum iris presentation attack detection in the nir and visible domains using attention-based and pixel-wise supervised learning. In *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection and Vulnerability Assessment, Third Edition*, Advances in Computer Vision and Pattern Recognition. Springer, 2022.

[87] Meiling Fang, Fadi Boutros, Arjan Kuijper, and Naser Damer. Partial attack supervision and regional weighted inference for masked face presentation attack detection. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021.

[88] Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Deep learning multi-layer fusion for an accurate iris presentation attack detection. In *IEEE 23rd International Conference on Information Fusion, FUSION 2020, Rustenburg, South Africa, July 6-9, 2020*, pages 1–8. IEEE, 2020.

[89] Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Cross-database and cross-attack iris presentation attack detection using micro stripes analyses. *Image Vis. Comput.*, 105:104057, 2021.

[90] Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. Iris presentation attack detection by attention-based and deep pixel-wise binary supervision network. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–8. IEEE, 2021.

[91] Meiling Fang, Naser Damer, Fadi Boutros, Florian Kirchbuchner, and Arjan Kuijper. The overlapping effect and fusion protocols of data augmentation techniques in iris PAD. *Mach. Vis. Appl.*, 33(1):8, 2022.

[92] Meiling Fang, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. Real masks and fake faces: On the masked face presentation attack detection. *CoRR*, abs/2103.01546, 2021.

[93] Yushu Feng, Huan Wang, Haoji Roland Hu, Lu Yu, Wei Wang, and Shiyan Wang. Triplet distillation for deep face recognition. In *IEEE International Conference on Image Processing, ICIP 2020, Abu Dhabi, United Arab Emirates, October 25-28, 2020*, pages 808–812. IEEE, 2020.

[94] Frontex. Best practice technical guidelines for automated border control (abc) systems, 2015.

[95] Ray Ganong, Donald Craig Waugh, Yong Man Ro, Konstantinos Plataniotis, and Chris Studholme. Face detection and recognition, May 2 2017. US Patent 9,639,740.

[96] Stephan J. Garbin, Yiru Shen, Immo Schuetz, Robert Cavin, Gregory Hughes, and Sachin S. Talathi. Openeds: Open eye dataset. *CoRR*, abs/1905.03702, 2019.

[97] Gentex-Corporation. Gentex introduces biometric authentication system for automotive use. *Gentex-Corporation: https://ir.gentex.com/news-releases/news-release-details/gentex-introduces-biometric-authentication-system-automotive-use*.

[98] Ceenu George, Mohamed Khamis, Emanuel von Zezschwitz, Marinus Burger, Henri Schmidt, Florian Alt, and Heinrich Hussmann. Seamless and secure vr: Adapting and evaluating established authentication systems for virtual reality. NDSS, 2017.

[99] Marta Gomez-Barrero, Pawel Drozdowski, Christian Rathgeb, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Naser Damer, Jannis Priesnitz, Nicholas W. D. Evans, and Christoph Busch. Biometrics in the era of COVID-19: challenges and opportunities. *CoRR*, abs/2102.09258, 2021.

[100] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[101] Dmitry O. Gorodnichy, Svetlana N. Yanushkevich, and Vlad P. Shmerko. Automated border control: Problem formalization. In *2014 IEEE Symposium on Computational Intelligence in Biometrics and Identity Management, CIBIM 2014, Orlando, FL, USA, December 9-12, 2014*, pages 118–125. IEEE, 2014.

[102] Patrick Grother, Mei Ngan, and Kayee Hanaoka. Ongoing face recognition vendor test (frvt) part 1: Verification. Technical report, 2018.

[103] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *ECCV 2016, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, volume 9907 of *Lecture Notes in Computer Science*, pages 87–102. Springer, 2016.

[104] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF CVPR, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. IEEE, 2020.

[105] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society, 2017.

[106] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE ICCV, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1026–1034. IEEE Computer Society, 2015.

[107] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016.

[108] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[109] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861, 2017.

[110] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7132–7141, 2018.

[111] Zhang Hua and Wei Lieh Ng. Speech recognition interface design for in-vehicle system. In *Proceedings of 2nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI 2010, Pittsburgh, PA, USA, November 11-12, 2010*, pages 29–33. ACM, 2010.

[112] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2261–2269. IEEE Computer Society, 2017.

[113] Gary B. Huang, Marwan A. Mattar, Honglak Lee, and Erik G. Learned-Miller. Learning to align from scratch. In *NIPS*, pages 773–781, 2012.

[114] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 11 2007.

[115] Rui Huang, Shu Zhang, Tianyu Li, and Ran He. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2439–2448, 2017.

[116] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018.

[117] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: Adaptive curriculum learning loss for deep face recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5900–5909. IEEE, 2020.

[118] Marco Huber, Fadi Boutros, Florian Kirchbuchner, and Naser Damer. Mask-invariant face recognition through template-level knowledge distillation. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021.

[119] Apple Inc. About face id advanced technology. `https://support.apple.com/en-us/HT208108`. Accessed: 2021-09-14.

[120] Motor Intelligence. Biometric in the automotive market. `https://www.mordorintelligence.com/industry-reports/biometric-in-the-automotive-market`. Accessed: 2021.

[121] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448–456. JMLR.org, 2015.

[122] ISO/IEC JTC1 SC37 Biometrics. ISO/IEC 2382-37:2017 Information technology - Vocabulary - Part 37: Biometrics. International Organization for Standardization, 2017.

[123] ISO/IEC JTC1 SC37 Biometrics. ISO/IEC 19795-1:2021 Information technology — Biometric performance testing and reporting — Part 1: Principles and framework. International Organization for Standardization, 2021.

[124] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5967–5976. IEEE Computer Society, 2017.

[125] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018.

[126] Anil K. Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Trans. Circuits Syst. Video Technol.*, 14(1):4–20, 2004.

[127] Govind Jeevan, Geevar C. Zacharias, Madhu S. Nair, and Jeny Rajan. An empirical study of the impact of masks on face recognition. *Pattern Recognition*, 122:108308, 2022.

[128] Jichao Jiao, Weilun Liu, Yaokai Moand Jian Jiao, Zhongliang Deng, and Xinping Chen. Dyn-arcface: dynamic additive angular margin loss for deep face recognition. *Multim. Tools Appl.*, 2021.

[129] Raghavender Jillela, Arun A Ross, Vishnu Naresh Boddeti, BVK Vijaya Kumar, Xiaofei Hu, Robert Plemmons, and Paúl Pauca. Iris segmentation for challenging periocular images. In *Handbook of iris recognition*, pages 281–308. Springer, 2013.

[130] Akanksha Joshi, Abhishek Gangwar, Renu Sharma, Ashutosh Singh, and Zia Saquib. Periocular recognition based on gabor and parzen pnn. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4977–4981. IEEE, 2014.

[131] Felix Juefei-Xu, Khoa Luu, Marios Savvides, Tien D Bui, and Ching Y Suen. Investigating age invariant face recognition based on periocular biometrics. In *2011 International Joint Conference on Biometrics (IJCB)*, pages 1–7. IEEE, 2011.

[132] Felix Juefei-Xu and Marios Savvides. Unconstrained periocular biometric acquisition and recognition using cots ptz camera for uncooperative and non-cooperative subjects. In *2012 IEEE Workshop on the Applications of Computer Vision (WACV)*, pages 201–208. IEEE, 2012.

[133] Felix Juefei-Xu and Marios Savvides. Subspace-based discrete transform encoded local binary patterns representations for robust periocular matching on nist's face recognition grand challenge. *IEEE transactions on image processing*, 23(8):3490–3505, 2014.

[134] Juho Kannala and Esa Rahtu. Bsif: Binarized statistical image features. In *21st International Conference on Pattern Recognition (ICPR) 2012*, pages 1363–1366. IEEE, 2012.

[135] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[136] Ira Kemelmacher-Shlizerman, Steven M. Seitz, Daniel Miller, and Evan Brossard. The megaface benchmark: 1 million faces for recognition at scale. In *2016 IEEE CVPR, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4873–4882. IEEE Computer Society, 2016.

[137] H. Khalid, S. Hashim, S. Ahmad, F. Hashim, and M. Chaudary. New and simple offline authentication approach using time-based one-time password with biometric for car sharing vehicles. In *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, pages 1–7, Los Alamitos, CA, USA, dec 2020. IEEE Computer Society.

[138] Sehee Kim and EuiChul Lee. Periocular biometric authentication methods in head mounted display device. *International Journal of Engineering & Technology*, 7(3.24), 2018.

[139] Yonghyun Kim, Wonpyo Park, Myung-Cheol Roh, and Jongju Shin. Groupface: Learning latent groups and constructing group-based representations for face recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5620–5629. IEEE, 2020.

[140] Davis E. King. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.*, 10:1755–1758, 2009.

[141] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[142] Jong-Gook Ko, Youn-Hee Gil, Jang-Hee Yoo, and Kyo-IL Chung. A novel and efficient feature extraction method for iris recognition. *ETRI journal*, 29(3):399–401, 2007.

[143] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, 2009.

[144] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114, 2012.

[145] Ajay Kumar and Arun Passi. Comparison and combination of iris matchers for reliable personal authentication. *Pattern recognition*, 43(3):1016–1026, 2010.

[146] K Kishore Kumar and Movva Pavani. Periocular region-based age-invariant face recognition using local binary pattern. In *Microelectronics, Electromagnetics and Telecommunications*, pages 713–720. Springer, 2019.

[147] Alexander Kupin, Benjamin Moeller, Yijun Jiang, Natasha Kholgade Banerjee, and Sean Banerjee. Task-driven biometric authentication of users in virtual reality (vr) environments. In *International Conference on Multimedia Modeling*, pages 55–67. Springer, 2019.

[148] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir D. Bourdev, and Thomas S. Huang. Interactive facial feature localization. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part III*, volume 7574 of *Lecture Notes in Computer Science*, pages 679–692. Springer, 2012.

[149] Mu Li, Wangmeng Zuo, and David Zhang. Convolutional network for attribute-driven and identity-preserving human face generation. *arXiv preprint arXiv:1608.06434*, 2016.

[150] Xianyang Li, Feng Wang, Qinghao Hu, and Cong Leng. Airface: Lightweight and efficient model for face recognition. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 2678–2682. IEEE, 2019.

[151] Yande Li, Kun Guo, Yonggang Lu, and Li Liu. Cropping and attention based approach for masked face recognition. *Applied Intelligence*, 51(5):3012–3025, 2021.

[152] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[153] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 936–944. IEEE Computer Society, 2017.

[154] Rolf Lindemann, Davit Baghdasaryan, Eric Tiffany, and Fido Alliance. Fido uaf protocol specification v1. 0. *FIDO Alliance Proposed Standard*, 2014.

[155] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: differentiable architecture search. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[156] Hao Liu, Xiangyu Zhu, Zhen Lei, and Stan Z. Li. Adaptiveface: Adaptive margin and sampling for face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 11947–11956. Computer Vision Foundation / IEEE, 2019.

[157] N. Liu, H. Li, M. Zhang, Jing Liu, Z. Sun, and T. Tan. Accurate iris segmentation in non-cooperative environments using fully convolutional networks. In *2016 International Conference on Biometrics (ICB)*, pages 1–8, June 2016.

[158] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8759–8768. IEEE Computer Society, 2018.

[159] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6738–6746. IEEE Computer Society, 2017.

[160] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. In *Proceedings of the 33nd International Conference on*

Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 507–516. JMLR.org, 2016.

[161] Wenting Liu, Li Zhou, and Jie Chen. Face recognition based on lightweight convolutional neural networks. *Inf.*, 12(5):191, 2021.

[162] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738. IEEE Computer Society, 2015.

[163] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N. Taha, and Nour Eldeen M. Khalifa. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the covid-19 pandemic. *Measurement*, 167:108288, 2021.

[164] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3431–3440. IEEE Computer Society, 2015.

[165] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[166] Giulio Lovisotto, Raghav Malik, Ivo Sluganovic, Marc Roeschlin, Paul Trueman, and Ivan Martinovic. Mobile biometrics in financial services: A five factor framework. Technical report, 2017.

[167] J. Lozej, B. Meden, V. Struc, and P. Peer. End-to-end iris segmentation using u-net. In *2018 IEEE International Work Conference on Bioinspired Intelligence (IWOBI)*, pages 1–6, July 2018.

[168] Jus Lozej, Dejan Stepec, Vitomir Struc, and Peter Peer. Influence of segmentation on deep iris recognition performance. In *7th International Workshop on Biometrics and Forensics, IWBF 2019, Cancun, Mexico, May 2-3, 2019*, pages 1–6. IEEE, 2019.

[169] Bingnan Luo, Jie Shen, Shiyang Cheng, Yujiang Wang, and Maja Pantic. Shape constrained network for eye segmentation in the wild. *CoRR*, abs/1910.05283, 2019.

[170] Bingnan Luo, Jie Shen, Yujiang Wang, and Maja Pantic. The ibug eye segmentation dataset. In Edoardo Pirovano and Eva Graversen, editors, *2018 Imperial College Computing Student Workshop, ICCSW 2018, September 20-21, 2018, London, United Kingdom*, volume 66 of *OASICS*, pages 7:1–7:9. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

[171] Dongdong Ma, Ziqin Chen, and Qingmin Liao. Tree-shaped sampling based hybrid multi-scale feature extraction for texture classification. In *ICIP*, pages 2087–2091. IEEE, 2018.

[172] Minhua Ma, Lakhmi C Jain, Paul Anderson, et al. *Virtual, augmented reality and serious games for healthcare 1*, volume 1. Springer, 2014.

[173] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet V2: practical guidelines for efficient CNN architecture design. In *ECCV 2018, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*, volume 11218 of *Lecture Notes in Computer Science*, pages 122–138. Springer, 2018.

[174] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

[175] Gayathri Mahalingam and Karl Ricanek. Lbp-based periocular recognition on challenging face datasets. *EURASIP Journal on Image and Video processing*, 2013(1):36, 2013.

[176] Khawla Mallat, Naser Damer, Fadi Boutros, and Jean-Luc Dugelay. Robust face authentication based on dynamic quality-weighted comparison of visible and thermal-to-visible images to visible enrollments. In *22th International Conference on Information Fusion, FUSION 2019, Ottawa, ON, Canada, July 2-5, 2019*, pages 1–8. IEEE, 2019.

[177] Khawla Mallat, Naser Damer, Fadi Boutros, Arjan Kuijper, and Jean-Luc Dugelay. Cross-spectrum thermal to visible face recognition based on cascaded image synthesis. In *2019 International Conference on Biometrics, ICB 2019, Crete, Greece, June 4-7, 2019*, pages 1–8. IEEE, 2019.

[178] A Mansfield. Information technology–biometric performance testing and reporting–part 1: Principles and framework. *ISO/IEC*, pages 19795–1, 2006.

[179] Yoanna Martínez-Díaz, Luis S. Luevano, Heydi Mendez Vazquez, Miguel Nicolás-Díaz, Leonardo Chang, and Miguel González-Mendoza. Shufflefacenet: A lightweight face architecture for efficient and highly-accurate face recognition. In *2019 IEEE/CVF ICCVW, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 2721–2728. IEEE, 2019.

[180] Yoanna Martínez-Díaz, Miguel Nicolás-Díaz, Heydi Méndez-Vázquez, Luis S Luevano, Leonardo Chang, Miguel Gonzalez-Mendoza, and Luis Enrique Sucar. Benchmarking lightweight face architectures on specific face recognition scenarios. *Artificial Intelligence Review*, pages 1–44, 2021.

[181] Libor Masek. Recognition of human iris patterns for biometric identification. Technical report, 2003.

[182] Nick Maynard. Mobile payment authentication. `https://www.juniperresearch.com/researchstore/fintech-payments/mobile-payment-authentication-market-research`. Accessed: 2021-04-12.

[183] Brianna Maze, Jocelyn C. Adams, James A. Duncan, Nathan D. Kalka, Tim Miller, Charles Otto, Anil K. Jain, W. Tyler Niggel, Janet Anderson, Jordan Cheney, and Patrick Grother. IARPA janus benchmark - C: face dataset and protocol. In *2018 ICB, ICB 2018, Gold Coast, Australia, February 20-23, 2018*, pages 158–165. IEEE, 2018.

[184] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings*, volume 11218 of *Lecture Notes in Computer Science*, pages 800–815. Springer, 2018.

[185] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. Magface: A universal representation for face recognition and quality assessment. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Virtual, June 19-25, 2020*. IEEE, 2021.

[186] Meredith Minear and Denise C Park. A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments, & Computers*, 36(4):630–633, 2004.

[187] Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 5191–5198. AAAI Press, 2020.

[188] Kazuyuki Miyazawa, Koichi Ito, Takafumi Aoki, Koji Kobayashi, and Hiroshi Nakajima. An effective approach for iris recognition using phase-based image matching. *IEEE transactions on pattern analysis and machine intelligence*, 30(10):1741–1756, 2008.

[189] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[190] Donald M Monro, Soumyadip Rakshit, and Dexin Zhang. Dct-based iris recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (4):586–595, 2007.

[191] David Montero, Marcos Nieto, Peter Leskovský, and Naiara Aginako. Boosting masked face recognition with multi-task arcface. *CoRR*, abs/2104.09874, 2021.

[192] Stylianos Moschoglou, Athanasios Papaioannou, Christos Sagonas, Jiankang Deng, Irene Kotsia, and Stefanos Zafeiriou. Agedb: The first manually collected, in-the-wild age database. In *2017 IEEE CVPRW, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1997–2005. IEEE Computer Society, 2017.

[193] Michael Christian Nechyba, Henry Will Schneiderman, and Michael Andrew Sipe. Facial recognition, 2012.

[194] Andrew YC Nee, SK Ong, George Chryssolouris, and Dimitris Mourtzis. Augmented reality applications in design and manufacturing. *CIRP annals*, 61(2):657–679, 2012.

[195] Pedro C. Neto, Fadi Boutros, João Ribeiro Pinto, Naser Darner, Ana F. Sequeira, and Jaime S. Cardoso. Focusface: Multi-task contrastive learning for masked face recognition. In *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*, pages 1–8. IEEE, 2021.

[196] Pedro C. Neto, Fadi Boutros, João Ribeiro Pinto, Mohsen Saffari, Naser Damer, Ana F. Sequeira, and Jaime S. Cardoso. My eyes are up here: Promoting focus on uncovered regions in masked face recognition. In *Proceedings of the 20th International Conference of the Biometrics Special Interest Group, BIOSIG 2021, Digital Conference, September 15-17, 2021*, volume P-315 of *LNI*, pages 21–30. Gesellschaft für Informatik e.V., 2021.

[197] Pedro C. Neto, João David Pinto, Fadi Boutros, Naser Damer, Ana Sequeira, and Jaime Cardoso. Beyond masks: On the generalization of masked face recognition models to occluded face recognition. 2022.

[198] Neurotechnology. Neurotechnology: Fingerprint, face, eye iris, voice and palm print identification, speaker and object recognition software. `"https://www.neurotechnology.com/"`, 2021.

[199] Hongwei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *2014 IEEE ICIP, ICIP 2014, Paris, France, October 27-30, 2014*, pages 343–347. IEEE, 2014.

[200] Richard Yew Fatt Ng, Yong Haur Tay, and Kai Ming Mok. An effective segmentation method for iris recognition system. 2008.

[201] Mei Ngan, Patrick Grother, and Kayee Hanaoka. Ongoing face recognition vendor test (frvt) part 6a: Face recognition accuracy with masks using pre- covid-19 algorithms. Technical report, 2020-07-24 2020.

[202] Mei Ngan, Patrick Grother, and Kayee Hanaoka. Ongoing face recognition vendor test (frvt) part 6b: Face recognition accuracy with face masks using post-covid-19 algorithms. Technical report, 2020-11-30 2020.

[203] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[204] Ilesanmi Olade, Hai-Ning Liang, and Charles Fleming. A review of multimodal facial biometric authentication methods in mobile devices and their application in head mounted displays. In Guojun Wang, Qi Han, Md. Zakirul Alam Bhuiyan, Xiaoxing Ma, Frédéric Loulergue, Peng Li, Manuel Roveri, and Lei Chen, editors, *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing &*

Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI 2018, Guangzhou, China, October 8-12, 2018, pages 1997–2004. IEEE, 2018.

[205] Michael Opitz, Georg Waltner, Georg Poier, Horst Possegger, and Horst Bischof. Grid loss: Detecting occluded faces. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, volume 9907 of *Lecture Notes in Computer Science*, pages 386–402. Springer, 2016.

[206] Javier Ortega-Garcia, Julian Fierrez, Fernando Alonso-Fernandez, Javier Galbally, Manuel R Freire, Joaquin Gonzalez-Rodriguez, Carmen Garcia-Mateo, Jose-Luis Alba-Castro, Elisardo Gonzalez-Agulla, Enrique Otero-Muras, et al. The multiscenario multienvironment biosecure multimodal database (bmdb). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1097–1111, 2009.

[207] Unsang Park, Arun Ross, and Anil K Jain. Periocular biometrics in the visible spectrum: A feasibility study. In *2009 IEEE 3rd Int. Conference on Biometrics: Theory, Applications, and Systems*, pages 1–6. IEEE, 2009.

[208] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *Proceedings of the British Machine Vision Conference 2015, BMVC 2015, Swansea, UK, September 7-10, 2015*, pages 41.1–41.12. BMVA Press, 2015.

[209] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016.

[210] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[211] Peyton Z Peebles. *Probability, random variables, and random signal principles*. McGraw Hill, 1987.

[212] Peter Peer. Cvl face database. *Computer vision lab., faculty of computer and information science, University of Ljubljana, Slovenia. Available at http://www. lrv. fri. uni-lj. si/facedb. html*, 2005.

[213] Ronald Poelman, Oytun Akman, Stephan Lukosch, and Pieter Jonker. As if being there: mediated reality for crime scene investigation. In *Proceedings of the ACM 2012 conference on computer supported cooperative work*, pages 1267–1276. ACM, 2012.

[214] Norman Poh and Samy Bengio. A study of the effects of score normalisation prior to fusion in biometric authentication tasks. Technical report, IDIAP, 2004.

[215] Rudra P. K. Poudel, Stephan Liwicki, and Roberto Cipolla. Fast-scnn: Fast semantic segmentation network. *CoRR*, abs/1902.04502, 2019.

[216] Deepti S Prakash, Lucia E Ballard, Jerrold V Hauck, Feng Tang, Etai Littwin, Pavan Kumar Ansosalu Vasu, Gideon Littwin, Thorsten Gernoth, Lucie Kucerova, Petr Kostka, et al. Biometric authentication techniques, February 23 2021. US Patent 10,929,515.

[217] Hugo Proença. Unconstrained iris recognition in visible wavelengths. In *Handbook of Iris Recognition*, pages 321–358. Springer, 2016.

[218] Hugo Proença and João C Neves. Deep-prwis: Periocular recognition without the iris and sclera using deep learning frameworks. *IEEE Transactions on Information Forensics and Security*, 13(4):888–896, 2017.

[219] Sandip Purnapatra, Nic Smalt, Keivan Bahmani, Priyanka Das, David Yambay, Amir Mohammadi, Anjith George, Thirimachos Bourlai, Sébastien Marcel, Stephanie Schuckers, Meiling Fang, Naser Damer, Fadi Boutros, Arjan Kuijper, Alperen Kantarci, Basar Demir, Zafer Yildiz, Zabi Ghafoory, Hasan Dertli, Hazim Kemal Ekenel, Son Vu, Vassilis Christophides, Dashuang Liang, Guanghao Zhang, Zhanlong Hao, Junfu Liu, Yufeng Jin, Samo Liu, Samuel Huang, Salieri Kuei, Jag Mohan Singh, and Raghavendra Ramachandra. Face liveness detection competition (livdet-face) - 2021. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021.

[220] Bosheng Qin and Dongxiao Li. Identifying facemask-wearing condition using image super-resolution with classification network to prevent covid-19. *Sensors*, 20(18):5236, 2020.

[221] Ramachandra Raghavendra, Kiran B. Raja, Bian Yang, and Christoph Busch. Combining iris and periocular recognition using light field camera. In *2nd IAPR Asian Conference on Pattern Recognition, ACPR 2013, Naha, Japan, November 5-8, 2013*, pages 155–159. IEEE, 2013.

[222] Kiran B. Raja, Naser Damer, Raghavendra Ramachandra, Fadi Boutros, and Christoph Busch. Cross-spectral periocular recognition by cascaded spectral image transformation. In *2019 IEEE International Conference on Imaging Systems and Techniques, IST 2019, Abu Dhabi, United Arab Emirates, December 9-10, 2019*, pages 1–7. IEEE, 2019.

[223] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. Searching for activation functions. In *6th ICLRs, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net, 2018.

[224] Christian Rathgeb and Andreas Uhl. Secure iris recognition based on local intensity variations. In *International Conference Image Analysis and Recognition*, pages 266–275. Springer, 2010.

[225] Christian Rathgeb, Andreas Uhl, and Peter Wild. Shifting score fusion: On exploiting shifting variation in iris recognition. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, pages 3–7. ACM, 2011.

[226] Narsi Reddy, Ajita Rattani, and Reza Derakhshani. Comparison of deep learning models for biometric-based mobile user authentication. In *9th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2018, Redondo Beach, CA, USA, October 22-25, 2018*, pages 1–6. IEEE, 2018.

[227] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells III, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241. Springer, 2015.

[228] Arun Ross. Iris recognition: The path forward. *Computer*, 43(2):30–35, 2010.

[229] Arun Ross, Raghavender Jillela, Jonathon M Smereka, Vishnu Naresh Boddeti, BVK Vijaya Kumar, Ryan Barnard, Xiaofei Hu, Paul Pauca, and Robert Plemmons. Matching highly non-ideal ocular images: An information fusion approach. In *2012 5th IAPR International Conference on Biometrics (ICB)*, pages 446–453. IEEE, 2012.

[230] P. Rot, Ž. Emeršič, V. Struc, and P. Peer. Deep multi-class eye segmentation for ocular biometrics. In *2018 IEEE International Work Conference on Bioinspired Intelligence (IWOBI)*, pages 1–8, July 2018.

[231] Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126(2-4):144–157, 2018.

[232] Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: Database and results. *Image and vision computing*, 47:3–18, 2016.

[233] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 4510–4520. IEEE Computer Society, 2018.

[234] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 815–823. IEEE Computer Society, 2015.

[235] Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. Green AI. *Commun. ACM*, 63(12):54–63, 2020.

[236] Soumyadip Sengupta, Jun-Cheng Chen, Carlos Domingo Castillo, Vishal M. Patel, Rama Chellappa, and David W. Jacobs. Frontal to profile face verification in the wild. In *2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016*, pages 1–9. IEEE Computer Society, 2016.

[237] Samir Shah and Arun Ross. Iris segmentation using geodesic active contours. *IEEE Transactions on Information Forensics and Security*, 4(4):824–836, 2009.

[238] Jie Shen, Stefanos Zafeiriou, Grigoris G Chrysos, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 50–58, 2015.

[239] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang. Faceid-gan: Learning a symmetry three-player GAN for identity-preserving face synthesis. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 821–830. IEEE Computer Society, 2018.

[240] Konstantin Shirokinskiy. Gtime to shift gear? a new role for tier 1 automotive suppliers in software-enabled vehicles. `https://www.rolandberger.com/en/Insights/Publications/Computer-on-Wheels-A-new-role-for-Tier-1-suppliers.html`. Accessed: 2021-02-22.

[241] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[242] Brian A Smith, Qi Yin, Steven K Feiner, and Shree K Nayar. Gaze locking: passive eye contact detection for human-object interaction. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, pages 271–280. ACM, 2013.

[243] Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. Technical report, University of Colorado Boulder, Computer Science Department, 1986.

[244] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1849–1857, 2016.

[245] Andreas Strasser, Philipp Stelzer, Christian Steger, and Norbert Druml. Live state-of-health safety monitoring for safety-critical automotive systems. In *22nd Euromicro Conference on Digital System Design, DSD 2019, Kallithea, Greece, August 28-30, 2019*, pages 102–107. IEEE, 2019.

[246] Jan Willem Streefkerk, Mark Houben, Pjotr van Amerongen, Frank ter Haar, and Judith Dijk. The art of csi: An augmented reality tool (art) to annotate crime scenes in forensic investigation. In *International Conference on Virtual, Augmented and Mixed Reality*, pages 330–339. Springer, 2013.

[247] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3645–3650. Association for Computational Linguistics, 2019.

[248] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 1988–1996, 2014.

[249] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation from predicting 10, 000 classes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014* [4], pages 1891–1898.

[250] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 6397–6406. IEEE, 2020.

[251] Zhenan Sun and Tieniu Tan. Ordinal measures for iris recognition. *IEEE Transactions on pattern analysis and machine intelligence*, 31(12):2211–2226, 2008.

[252] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 1–9. IEEE Computer Society, 2015.

[253] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014* [4], pages 1701–1708.

[254] Mingxing Tan and Quoc V. Le. Mixconv: Mixed depthwise convolutional kernels. In *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, page 74. BMVA Press, 2019.

[255] Yehui Tang, Yunhe Wang, Yixing Xu, Yiping Deng, Chao Xu, Dacheng Tao, and Chang Xu. Manifold regularized dynamic network pruning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 5018–5028. Computer Vision Foundation / IEEE, 2021.

[256] Philipp Terhörst, Jan Niklas Kolf, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. SER-FIQ: unsupervised estimation of face image quality based on stochastic embedding robustness. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5650–5659. Computer Vision Foundation / IEEE, 2020.

[257] Kien Nguyen Thanh, Clinton Fookes, Arun Ross, and Sridha Sridharan. Iris recognition with off-the-shelf CNN features: A deep learning perspective. *IEEE Access*, 6:18848–18855, 2018.

[258] Neil C. Thompson, Kristjan H. Greenewald, Keeheon Lee, and Gabriel F. Manso. The computational limits of deep learning. *CoRR*, abs/2007.05558, 2020.

[259] Azfar Bin Tomi and Dayang Rohaya Awang Rambli. A conceptual design for augmented reality games using motion detection as user interface and interaction. In *International Visual Informatics Conference*, pages 305–315. Springer, 2011.

[260] Aäron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Koray Kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4790–4798, 2016.

[261] Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1747–1756. JMLR.org, 2016.

[262] Maria Villa, Mikhail Gofman, and Sinjini Mitra. Survey of biometric techniques for automotive applications. In *Information Technology-New Generations*, pages 475–481. Springer, 2018.

[263] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In William W. Cohen, Andrew McCallum, and Sam T. Roweis, editors, *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, pages 1096–1103. ACM, 2008.

[264] Matej Vitek, Abhijit Das, Diego Rafael Lucio, Luiz Antonio Zanlorensi Jr., David Menotti, Jalil Nourmohammadi Khiarak, Mohsen Akbari Shahpar, Meysam Asgari-Chenaghlu, Farhang Jaryani, Juan E. Tapia, Andres Valenzuela, Caiyong Wang, Yunlong Wang, Zhaofeng He, Zhenan Sun, Fadi Boutros, Naser Damer, Jonas Henry Grebe, Arjan Kuijper, Kiran Raja, Gourav Gupta, Georgios Zampoukis, Lazaros Tsochatzidis, Ioannis Pratikakis, S. V. Aruna Kumar, B. S. Harish, Umapada Pal, Peter Peer, and and Vitomir Štruc. Exploring bias in sclera segmentation models: A group evaluation approach. 2022.

[265] Matej Vitek, Abhijit Das, Y. Pourcenoux, A. Missler, C. Paumier, Sumanta Das, Ishita De Ghosh, Diego Rafael Lucio, Luiz A. Zanlorensi, David Menotti, Fadi Boutros, Naser Damer, Jonas Henry Grebe, Arjan Kuijper, J. Hu, Y. He, C. Wang, H. Liu, Y. Wang, Z. Sun, Dailé Osorio Roig, Christian Rathgeb, Christoph Busch, Juan E. Tapia, Andres Valenzuela, G. Zampoukis, Lazaros T. Tsochatzidis, Ioannis Pratikakis, S. Nathan, R. Suganya, Vineet Mehta, Abhinav Dhall, Kiran B. Raja, G. Gupta, Jalil Nourmohammadi-Khiarak, M. Akbari-Shahper, Farhang Jaryani, Meysam Asgari-Chenaghlu, R. Vyas, S. Dakshit, Peter Peer, Umapada Pal, and Vitomir Struc. SSBC 2020: Sclera segmentation benchmarking competition in the mobile environment. In *2020 IEEE International Joint Conference on Biometrics, IJCB 2020, Houston, TX, USA, September 28 - October 1, 2020*, pages 1–10. IEEE, 2020.

[266] Paul Voigt and Axel von dem Bussche. *The EU General Data Protection Regulation (GDPR): A Practical Guide*. Springer, 1st edition, 2017.

[267] Caiyong Wang, Yunlong Wang, Kunbo Zhang, Jawad Muhammad, Tianhao Lu, Qi Zhang, Qichuan Tian, Zhaofeng He, Zhenan Sun, Yiwen Zhang, Tianbao Liu, Wei Yang, Dongliang Wu, Yingfeng Liu, Ruiye Zhou, Huihai Wu, Hao Zhang, Junbao Wang, Jiayi Wang, Wantong Xiong, Xueyu Shi, Shao Zeng, Peihua Li, Haodong Sun, Jing Wang, Jiale Zhang, Qi Wang, Huijie Wu, Xinhui Zhang, Haiqing Li, Yu Chen, Liang Chen, Menghan Zhang, Ye Sun, Zhiyong Zhou, Fadi Boutros, Naser Damer, Arjan Kuijper, Juan E. Tapia, Andres Valenzuela, Christoph Busch, Gourav Gupta, Kiran B. Raja, Xi Wu, Xiaojie Li, Jingfu Yang, Hongyan Jing, Xin Wang, Bin Kong, Youbing Yin, Qi Song, Siwei Lyu, Shu Hu, Leon Premk, Matej Vitek, Vitomir Struc, Peter Peer, Jalil Nourmohammadi-Khiarak, Farhang Jaryani, Samaneh Salehi Nasab, Seyed Naeim Moafinejad, Yasin Amini, and Morteza Noshad. NIR iris challenge evaluation in non-cooperative environments: Segmentation and localization. In *International IEEE Joint Conference on Biometrics, IJCB 2021, Shenzhen, China, August 4-7, 2021*, pages 1–10. IEEE, 2021.

[268] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5265–5274. Computer Vision Foundation / IEEE Computer Society, 2018.

[269] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[270] Zekun Wang, Pengwei Wang, Peter C Louis, Lee E Wheless, and Yuankai Huo. Wearmask: Fast in-browser face mask detection with serverless edge computing for covid-19. *arXiv preprint arXiv:2101.00784*, 2021.

[271] Zhongyuan Wang, Guangcheng Wang, Baojin Huang, Zhangyang Xiong, Qi Hong, Hao Wu, Peng Yi, Kui Jiang, Nanxi Wang, Yingjiao Pei, Heling Chen, Yu Miao, Zhibing Huang,

and Jinbi Liang. Masked face recognition dataset and application. *CoRR*, abs/2003.09093, 2020.

[272] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Processing*, 13(4):600–612, 2004.

[273] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VII*, volume 9911 of *Lecture Notes in Computer Science*, pages 499–515. Springer, 2016.

[274] Cameron Whitelam, Emma Taborsky, Austin Blanton, Brianna Maze, Jocelyn C. Adams, Tim Miller, Nathan D. Kalka, Anil K. Jain, James A. Duncan, Kristen Allen, Jordan Cheney, and Patrick Grother. IARPA janus benchmark-b face dataset. In *2017 IEEE CVPRW, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 592–600. IEEE Computer Society, 2017.

[275] R. P. Wildes. Iris recognition: an emerging biometric technology. *Proceedings of the IEEE*, 85(9):1348–1363, Sep. 1997.

[276] Lior Wolf, Tal Hassner, and Itay Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR 2011*, pages 529–534. IEEE, 2011.

[277] Damon L Woodard, Shrinivas Pundlik, Philip Miller, Raghavender Jillela, and Arun Ross. On the fusion of periocular and iris biometrics in non-ideal imagery. In *2010 20th International Conference on Pattern Recognition*, pages 201–204. IEEE, 2010.

[278] Damon L Woodard, Shrinivas J Pundlik, Jamie R Lyle, and Philip E Miller. Periocular region appearance cues for biometric identification. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 162–169. IEEE, 2010.

[279] Bichen Wu, Alvin Wan, Xiangyu Yue, Peter H. Jin, Sicheng Zhao, Noah Golmant, Amir Gholaminejad, Joseph Gonzalez, and Kurt Keutzer. Shift: A zero flop, zero parameter alternative to spatial convolutions. In *2018 IEEE CVPR, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 9127–9135, 2018.

[280] Hsin-Kai Wu, Silvia Wen-Yu Lee, Hsin-Yi Chang, and Jyh-Chong Liang. Current status, opportunities and challenges of augmented reality in education. *Computers & education*, 62:41–49, 2013.

[281] Xiang Wu, Ran He, Zhenan Sun, and Tieniu Tan. A light CNN for deep face representation with noisy labels. *IEEE Trans. Inf. Forensics Secur.*, 13(11):2884–2896, 2018.

[282] xiaodi Fu, Jiang Lu, Xin Zhang, Xiaokun Yang, and Ishaq Unwala. Intelligent in-vehicle safety and security monitoring system with face recognition. In *2019 IEEE International Conference on Computational Science and Engineering, CSE 2019, and IEEE International Conference on Embedded and Ubiquitous Computing, EUC 2019, New York, NY, USA, August 1-3, 2019*, pages 225–229. IEEE, 2019.

[283] Lumin Xu, Yingda Guan, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Vipnas: Efficient video pose estimation via neural architecture search. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 16072–16081. Computer Vision Foundation / IEEE, 2021.

[284] Mengjia Yan, Mengao Zhao, Zining Xu, Qian Zhang, Guoli Wang, and Zhizhong Su. Vargfacenet: An efficient variable group convolutional neural network for lightweight face recognition. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 2647–2654. IEEE, 2019.

[285] Jian Yang, Lei Luo, Jianjun Qian, Ying Tai, Fanlong Zhang, and Yong Xu. Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(1):156–171, 2017.

[286] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z. Li. Learning face representation from scratch. *CoRR*, abs/1411.7923, 2014.

[287] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Learning a discriminative feature network for semantic segmentation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1857–1866. IEEE Computer Society, 2018.

[288] Hang Zhang, Kristin J. Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7151–7160. IEEE Computer Society, 2018.

[289] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE SPL*, 23(10):1499–1503, 2016.

[290] Qian Zhang, Jianjun Li, Meng Yao, Liangchen Song, Helong Zhou, Zhichao Li, Wenming Meng, Xuezhi Zhang, and Guoli Wang. Vargnet: Variable group convolutional neural network for efficient embedded computing. *CoRR*, abs/1907.05653, 2019.

[291] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6848–6856. Computer Vision Foundation / IEEE Computer Society, 2018.

[292] Xiao Zhang, Rui Zhao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. Adacos: Adaptively scaling cosine logits for effectively learning deep face representations. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 10823–10832. Computer Vision Foundation / IEEE, 2019.

[293] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part III*, volume 11207 of *Lecture Notes in Computer Science*, pages 418–434. Springer, 2018.

[294] Zijing Zhao and Ajay Kumar. Accurate periocular recognition under less constrained environment using semantics-assisted convolutional neural network. *IEEE Transactions on Information Forensics and Security*, 12(5):1017–1030, 2016.

[295] T. Zheng and W. Deng. Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. Technical Report 18-01, Beijing University of Posts and Telecommunications, February 2018.

[296] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-age LFW: A database for studying cross-age face recognition in unconstrained environments. *CoRR*, abs/1708.08197, 2017.

[297] Yutong Zheng, Dipan K. Pal, and Marios Savvides. Ring loss: Convex feature normalization for face recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5089–5097. Computer Vision Foundation / IEEE Computer Society, 2018.

[298] Barret Zoph and Quoc V. Le. Neural architecture search with reinforcement learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[299] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8697–8710. IEEE Computer Society, 2018.