



VCU

Virginia Commonwealth University
VCU Scholars Compass

Theses and Dissertations

Graduate School

2023

Face Anti-Spoofing and Deep Learning Based Unsupervised Image Recognition Systems

Enoch Solomon
Virginia Commonwealth University

Follow this and additional works at: <https://scholarscompass.vcu.edu/etd>

 Part of the Artificial Intelligence and Robotics Commons, Data Science Commons, Software Engineering Commons, and the Theory and Algorithms Commons

© The Author

Downloaded from
<https://scholarscompass.vcu.edu/etd/7482>

This Dissertation is brought to you for free and open access by the Graduate School at VCU Scholars Compass. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of VCU Scholars Compass. For more information, please contact libcompass@vcu.edu.

©Enoch Solomon, August 2023

All Rights Reserved.



VCU

College of Engineering

Face Anti-Spoofing and Deep Learning Based Unsupervised Image Recognition Systems

*A Dissertation submitted in partial fulfillment of the requirements
for the degree of*

Engineering,

Doctor of Philosophy

with a concentration in

Computer Science

at Virginia Commonwealth University

by

Enoch Solomon

Director: Dr. Krzysztof J. Cios

Professor, Department of Computer Science

Dr. Kostadin Damevski, Virginia Commonwealth University

Dr. Hongsheng Zhou, Virginia Commonwealth University

Dr. Amita Chin, Virginia Commonwealth University

Dr. Yaohang Li, Old Dominion University

Virginia Commonwealth University

Richmond, Virginia

August, 2023

May God bless you!!

Dedication

To my beloved

Wife: Sebli

Sons: Noah, Matthias and Elon

Dad: Solomon

Mommy: Berhane

*Brothers and sisters: Ruth (Seife), Bini
(Abigi), Berni (Eda), Emuye (Aman), Kiduye
(Abelo), Eyuye (Sis) and Honey*

Best friend: Dr. Abraham Woubie (Etsube)

Acknowledgements

First and above all, I give thanks with all my heart to the Almighty God, Creator of this world, the light of my path, the lamp of my life, strength and protection. His name be praised. I thank and praise the Almighty God for providing me this opportunity and capability to complete my PhD successfully.

Then, this dissertation would not be possible without the supervision and ideas of my advisor and chair of my committee, Dr. Krzysztof J. Cios. I would like to express my deepest gratitude and thank him for his sincere guidance, support, encouragement, hard-work, dedication and cooperation throughout this long journey. Thank you again!!

I would like to express my gratitude to my PhD dissertation committee: Dr. Kostadin Damevski, Dr. Hongsheng Zhou, Dr. Amita Goyal Chin and Dr. Yaohang Li for serving in my dissertation committee and for their valuable feedback.

I am immensely grateful to my *best* friend Dr. Abraham Woubie for his invaluable guidance and unwavering integrity. Dr. Abraham has always guided me whenever I needed to. He is the same every time, in spite of the circumstances. Etsube, love you!

I am extremely thankful to my number-one dad, Solomon; my hero mother, Birhane; my beloved brothers and sisters: Ruth (Seife), Bini (Abigi), Berni (Eda), Emuye (Aman), Kiduye (Abelo), Eyuye (Sis) and Honey; Pastor Tilaye Habtemariam and friends for their immense support during my academic journey. They have always supported me and prayed for my success. Without their praying, I will not be where I am now.

My gorgeous wife, Sebli, who has been my backbone, I express my deep love and appreciation for her tireless support and togetherness. She has been always with me throughout this PhD work. She deserves a very special thanks. It is only because of her moral support that this dissertation has come to completion. I am wholeheartedly grateful to my sons: Noah, Matthias and Elon since I didn't give them enough time because of my PhD work, but I promise to pay back by spending decent amount of time together.

I also appreciate the financial support offered by the Virginia Commonwealth University.

“I can do all things through Christ which strengtheneth me.”

Philippians 4:13

Abstract

**Face Anti-Spoofing and Deep Learning Based
Unsupervised Image Recognition Systems**

By Enoch Solomon

**A Dissertation submitted in partial fulfillment of
the requirements for the degree of Engineering,
Doctor of Philosophy with a concentration in
Computer Science at Virginia Commonwealth
University.**

**Virginia Commonwealth University
August, 2023.**

**Director: Dr. Krzysztof J. Cios
Professor, Department of Computer Science**

One of the main problems of a supervised deep learning approach is that it requires large amounts of labeled training data, which are not always easily available. This PhD dissertation addresses the above-mentioned problem by using a novel unsupervised deep learning face verification system called UFace, that does not require labeled training data as it automatically, in an unsupervised way, generates training data from even a relatively small size of data. The method starts by selecting, in unsupervised way, k-most similar and k-most dissimilar images for a given face image. Moreover, this PhD dissertation proposes a new loss function to make it work with the proposed method. Specifically, the method computes loss function k times for both similar and dissimilar images for each input image in order to increase the discriminative power of feature vectors to learn the inter-class and intra-class face variability. The training is carried out based on the similar and dissimilar input face image vector rather than the same training input face image vector in order to extract face embeddings.

The UFace is evaluated on four benchmark face verification datasets: Labeled Faces in the Wild dataset (LFW), YouTube Faces dataset (YTF), Cross-age LFW (CALFW) and Celebrities in Frontal Profile in the Wild (CFP-FP) datasets. The results show that we gain an accuracy of 99.40%, 96.04%, 95.12% and 97.89% respectively. The achieved results, despite being unsupervised, is on par to a similar but fully supervised methods.

Another, related to face verification, area of research is on face anti-spoofing systems. State-of-the-art face anti-spoofing systems use either deep learning, or manually extracted image quality features.

However, many of the existing image quality features used in face anti-spoofing systems are not well discriminating spoofed and genuine faces. Additionally, State-of-the-art face anti-spoofing systems that use deep learning approaches do not generalize well.

Thus, to address the above problem, this PhD dissertation proposes hybrid face anti-spoofing system that considers the best from image quality feature and deep learning approaches. This work selects and proposes a set of seven novel no-reference image quality features measurement, that discriminate well between spoofed and genuine faces, to complement the deep learning approach. It then, proposes two approaches: In the first approach, the scores from the image quality features are fused with the deep learning classifier scores in a weighted fashion. The combined scores are used to determine whether a given input face image is genuine or spoofed. In the second approach, the image quality features are concatenated with the deep learning features. Then, the concatenated features vector is fed to the classifier to improve the performance and generalization of anti-spoofing system.

Extensive evaluations are conducted to evaluate their performance on five benchmark face anti-spoofing datasets: Replay-Attack, CASIA-MFSD, MSU-MFSD, OULU-NPU and SiW. Experiments on these datasets show that it gives better results than several of the state-of-the-art anti-spoofing systems in many scenarios.

Contents

Dedication	iii
Acknowledgements	iv
Abstract	vi
List of Tables	xii
List of Figures	xiv
1 Introduction	1
1.1 Motivation	2
1.1.1 Face Recognition	2
1.1.2 Face Anti-Spoofing	3
1.2 Objectives and Proposed Contributions	5
1.2.1 The Proposed Novel Unsupervised Deep Learning System for Face Verification System	5
1.2.2 The Proposed Novel Face Anti-Spoofing System	7
1.3 Organization of the PhD Dissertation	9
2 Background and Related Works	11
2.1 Face Recognition	12
2.1.1 Face Identification vs. Verification	13
2.1.2 Preprocessing	14
2.1.3 Face Detection	16
2.1.4 Feature Extraction	16
2.2 Face Anti-spoofing	23
2.2.1 Spoofing Attacks	24
2.2.2 Hardware Based Methods	27

2.2.3	Image Quality Feature Based Methods	28
2.2.4	Deep Learning Based Methods	30
3	An Unsupervised Deep Learning Face Verification System	35
3.1	The Proposed Preprocessing Method	39
3.2	The Proposed Autoencoder Training Method	41
3.3	The Proposed Siamese Network Training Method . .	46
3.4	Evaluation Method	48
3.5	Datasets	49
3.6	Experimental Setup	50
3.7	Experimental Results of the Proposed Autoencoder vs Classical Autoencoder	52
3.8	Experimental Results of the Proposed Siamese Network	54
3.8.1	Labeled Faces in the Wild dataset (LFW) Dataset	54
3.8.2	YouTube Faces dataset (YTF) Dataset	55
3.8.3	Cross-age LFW (CALFW) and Celebrities in Frontal Profile in the Wild (CFP-FP) Datasets	56
3.9	Summary	57
4	Face Anti-Spoofing System Using Image Quality Features and Deep Learning Approach	59
4.1	The Proposed Method	64
4.1.1	The Proposed Image Quality Feature Measurements	65
4.2	Score Level Fusion	70
4.3	Feature Level Fusion	72
4.4	Experimental Setup	73
4.5	Datasets	73
4.6	Evaluation metrics	75
4.7	Experimental Results of the Proposed Method (FASS)	76
4.7.1	The Proposed Image Quality Feature Measurements on Replay-Attack, CASIA- MFSD and MSU-MFSD Datasets	76
4.7.2	OULU-NPU Dataset	77
4.7.3	SiW Dataset	79

4.7.4	Cross-Dataset Testing between CASIA-MFSD and Replay-Attack Datasets	80
4.8	Experimental Results of the Proposed Method (HDLHC)	81
4.8.1	Oulu-NPU Dataset	81
4.8.2	SiW dataset	83
4.9	Summary	84
5	Conclusions	86
5.1	Conclusions	86
5.2	Future Research Lines	88
	Bibliography	90

List of Tables

3.1	Accuracy of the classical and the two proposed autoencoder training method.	53
3.2	Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on LFW dataset.	54
3.3	Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on YTF test dataset.	56
3.4	Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on CALFW test dataset.	56
3.5	Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on CFP-FP dataset.	57
4.1	List of the twelve No-Reference (NR) Image Quality (IQ) feature measurements.	65
4.2	Comparison of the proposed Image Quality (IQ) feature measurements with other image quality feature measurement based methods on Replay-Attack, CASIA-MFSD and MSU-MFSD datasets.	76
4.3	Comparison of the proposed method with the state-of-the-art methods using four protocols on the OULU-NPU dataset.	78
4.4	Comparison of the proposed method with the state-of-the-art methods using three protocols on the SiW dataset.	79
4.5	Comparison of the proposed method with the state-of-the-art methods using cross-dataset between CASIA-MFSD and Replay-Attack.	80

4.6	Comparison of the proposed method (HDLHC) with the state-of-the-art methods on Oulu-NPU dataset.	82
4.7	Comparison of the proposed method (HDLHC) with the state-of-the-art methods on SiW dataset.	83

List of Figures

2.1	Example of face identification vs. verification	14
2.2	Different points of attacking a face recognition system.	23
2.3	The left top four images represent the low quality videos, the left bottom are the normal quality videos, and the right are the high quality videos. For each quality, from left to right are genuine, warped photo attack, cut photo attack and video attack.	25
2.4	Sample face images of 3D mask attacks	27
3.1	Sample images after MTCNN was used for face detection.	39
3.2	The proposed architectures used for training with autoencoder a) and with Siamese network b).	45
3.3	UFace preprocessing steps.	46
3.4	Architecture used for the proposed method evaluation.	48
4.1	The proposed FASS system.	63
4.2	ACER value as it changes with adding additional features for the Replay-Attack dataset.	67
4.3	ACER value as it changes with adding additional features for the CASIA-MFSD dataset.	68
4.4	ACER value as it changes with adding additional features for the MSU-MFSD dataset.	69
4.5	The proposed HDLHC system.	72

Chapter 1

Introduction

Face recognition has been an active area of research in the past several decades. Initially, it was a branch of artificial intelligence endowing robots with visual perception. Now, it became a part of a general discipline of computer vision [1].

In contrast to general computer vision, face recognition is confined to the narrow band of visible light for which surveillance and biometrics authentication can be performed [2]. Biometrics is the term used to describe human characteristics metrics such as iris, fingerprint or face [3]. These metrics are used for identification and verification of individuals, and access control of individuals. Face has become the preferred metric simply because it is a natural characteristic of identity, and its non-intrusive nature provides convenience and ease of verification [3]. For example, using a fingerprinting system, the individual is required to interact with the system by placing a finger under a fingerprint reader. Thus, this PhD dissertation focuses on face recognition, since in contrast, using the individual's face as a metric does not require any physical intervention.

The development of a face recognition system has different stages. Firstly, all images must be captured by a camera and then be given

to a face recognition application for further processing. Compared to the human visual system, the camera is the eye, and the processing software is the brain of the application. To acquire the image, the camera uses light reflecting off an object and transmits the light intensity to its built-in sensors. The sensors then convert each of their cell intensities to a value in the range of 0-255, where a grid of numbers in this range becomes the final representation of the captured image [4].

1.1 Motivation

1.1.1 Face Recognition

The human visual system interprets the object as a human face effortlessly [5]. It has no problem interpreting the subtle variation of translucency and correctly segmenting the object as a human face from its background. The human eye and brain can extract detailed information from the image using an existing pattern of recognition from years of experience and evolution. Furthermore, the human vision system captures objects in three dimensions with contextual properties such as depth, color, shape, and appearance. However, these properties are all lost when the camera captures an image, and its data reach a face recognition system. Given camera data as a two-dimensional grid of numbers, a face recognition system must recover the lost contextual information by inverting the camera acquisition process from unknown and insufficient information. The recovery of lost contextual properties, the visual reconstruction of an image, and its interpretation from insufficient information are the reasons that make face recognition challenging.

Recently, several face recognition systems have been proposed and some major advances have been achieved. But, the state-of-the-art face recognition systems such as [6–19] are supervised ones, which requires labeled training data. But, in practice, it is costly to get labeled training data.

1.1.2 Face Anti-Spoofing

The other challenge we address in this PhD dissertation is the spoofing attack issue in face recognition systems. As long as there are face verification mechanisms in place, fraudsters will always find a way to get around them. One such method is face spoofing, in which a fraudster attempts to deceive a facial recognition system by displaying a spoof face to the camera. Face spoofing means using a target person’s fake face and simulating facial biometrics to steal their identity.

Thus, the need for reliable identification and verification methods is a fundamental requirement in many applications such as border control, financial transactions, and computer security. The authentication methods such as tokens, ID cards, passwords, and Personal Identification Numbers (PINs) are the most widely used tools to authenticate people and protect data and systems. These methods provide an adequate level of security but, unfortunately, they suffer from different drawbacks. For instance, the tokens and the ID cards can be easily stolen or lost, and the passwords can be forgotten, guessed or hacked by the attackers.

Another alternative for recognizing and authenticating people is the use of biometric information. Biometric systems aim to recognize people based on their physiological characteristics such as face, voice,

fingerprint, and iris [20, 21]. Because these biometric traits are unique for each individual, they ensure a good level of security. With system and hardware prices dropping and reliability and convenience going up, many biometric systems started to be deployed in real-world applications such as mobile device authentication, identity card management, and security portal verification.

But, the deployment of biometric systems has arisen new challenges. Among these different issues and challenges, the vulnerability against spoofing attacks has drawn a significant level of attention. Many studies (e.g., [22–24]) have shown that most of the existing biometric systems are vulnerable to spoofing attacks. In [25], six commercial face authentication systems were successfully spoofed using face images downloaded from social media websites. Compared to other modalities, face recognition systems are more susceptible to spoofing attacks. Because of the explosion of the social media websites and the improvement of the camera resolutions and print quality, it is easy to get face images or videos of a target person. Using these images and videos, someone can easily gain an illegitimate access to the systems by presenting them either on printed paper or replayed a video clip on display devices in front of the face recognition camera.

Indeed, recent studies have revealed that the performance of the state-of-the-art methods degrades drastically under the real-world variations (e.g., illumination and camera device variations) [26–30], which indicates that more robust face anti-spoofing methods are needed to reach the deployment levels of the face biometric systems.

1.2 Objectives and Proposed Contributions

This PhD dissertation includes five main objectives: the first three are for the case of face recognition system as described below in 1.2.1 and the remaining two are for the case face anti-spoofing system as described below in 1.2.2, which are organized in a sequential manner, where one objective flows into and motivates the next one.

The five main objectives are as follows.

1.2.1 The Proposed Novel Unsupervised Deep Learning System for Face Verification System

Here, the three main research objectives are motivated by the drawback and limitations of existing supervised systems. Most of the existing state-of-the-art face recognition systems often rely on a large amount of labeled training data, which are not always available. To address this problem, an unsupervised deep learning face verification system, called UFace, is proposed.

Objective 1: The proposed novel system using K most similar images using autoencoder network.

In the first objective, we propose a new loss function to make use of k most similar face images for face verification and applied it to an autoencoder network. First in unsupervised way, it generates the k most similar images to each input image from large unlabeled data. Then, using k most similar images, it trains the autoencoder. During the training, autoencoder tries to reconstruct similar face images instead of the same training face images. In this way, the network learns face variability in an implicit way. Further details are stated in chapter 3, section 3.2.

Objective 2: The Proposed novel system using K most similar and K most dissimilar images using autoencoder network.

In the second objective, to improve the performance of the first objective, we propose the use of k dissimilar face images in addition to k most similar face images and applied to autoencoder network. Firstly, in unsupervised way, it generates the k most similar and k dissimilar images for each input image from large unlabeled data. Then, using k most similar and dissimilar images, it trains the autoencoder. In addition, this work proposes new loss function. Further details are outlined in chapter 3, section 3.2.

Objective 3: The Proposed novel system using Siamese network.

In the third objective, again, to improve the performance of the above two objectives, we make use of the k most similar face images along with the dissimilar face images and applied it to the triple-branch Siamese network. Using these training pairs, we train a triple-branch Siamese network using triplet loss, which is aimed to extract unsupervised face embeddings. Unlike in typical deep neural network training, it computes the loss function k times for similar images and k times for dissimilar images for each input image. In the testing phase, we extract face embeddings and then score them using cosine scoring. Their performance is evaluated using four benchmark face verification datasets. One of the biggest advantages of the proposed system is that it uses much less training data and does not require labeled data. Further details are described in chapter 3, section 3.3.

1.2.2 The Proposed Novel Face Anti-Spoofing System

Face spoofing attack's goal is to obtain fraudulent access to a biometric system. Most of the state-of-the-art face anti-spoofing systems use either deep learning, or image quality feature measurements or hand-crafted features extracted from face images as input to a classical classifier.

Many of the existing image quality feature measurements used in face anti-spoofing systems are not well discriminating between spoofed and genuine faces. Likewise, the state-of-the-art deep learning-based face anti-spoofing system do not generalize well.

Therefore, we felt that there is a potential improvement can be achieved by considering the best out of those two approaches (i.e., image quality feature measurements and deep learning).

Thus, two additional main research objectives are proposed to improve the poor generalization and performance issues that most of state-of-the-art face anti-spoofing system suffers.

Objective 4: A novel face anti-spoofing system by fusing the proposed image quality features measurement with deep learning method at the classification score level.

In the fourth objective, we identify and introduce the most significant set of seven no-reference image quality features measurement. Thus, this PhD dissertation proposes a novel system, called **FASS**, that uses the identified image quality features measurement as input to random forest classifier, which results are fused in a weighted fashion with the results of a deep learning classifier to improve the performance and generalization of anti-spoofing system. Further details are stated in chapter 4, section 4.2. We perform extensive

experiments comparing the proposed system with state-of-the-art anti-spoofing systems on five benchmark face anti-spoofing datasets: Replay-Attack, CASIA-FASD, MSU-MFSD, Oulu-NPU and SiW.

Objective 5: A novel face anti-spoofing system by fusing the proposed image quality features measurement with deep learning method at the feature level.

In the last and fifth objective, we address the issue of face spoofing attack by proposing a novel system that concatenate the proposed image quality features measurement with the deep features obtained after removing the last layer of VGG network. The combined feature vector is fed into the classifier to get the decision score to determine whether a given input face image is genuine or spoofed. Further details are discussed in chapter 4, section 4.3. We perform extensive experiments comparing the proposed method with state-of-the-art anti-spoofing systems on two benchmark face anti-spoofing datasets: Oulu-NPU and SiW.

Some of the main contributions are as follows:

- Proposes a novel training method that does not explicitly require a labeled training dataset.
- Proposes a novel an unsupervised face recognition system called UFace, that uses the k most similar and k most dissimilar images of the original input face image.
- Proposes a novel algorithm to select similar and dissimilar images for each dataset in an unsupervised manner.
- Proposes a novel technique to significantly increase a size of training data for the applications where only small datasets exist. For example, having only 100 training images with $k =$

10 similar/dissimilar images, results in $100*K + 100*K$ training images.

- Proposes a novel autoencoder and Siamese training methods that tries to reconstruct similar face images instead of using the same original input face images.
- Proposes new loss functions called UFace_MSE and UFace_Loss, to make use of the most similar/dissimilar images.
- Introduces a novel set of seven no-reference image quality features measurement, to be used in face anti-spoofing system, that discriminate well between spoofed and genuine faces.
- Introduces a novel hybrid face anti-spoofing systems, called FASS and HDLHC.
- Introduces a novel system that fuses the scores in a weighted fashion from classifier that uses the proposed set of seven image quality features measurement and the classifier that uses the deep features based on ResNet-50 network.
- Introduces a novel system that concatenates the features extracted from the proposed seven image quality features measurement and deep learning features extracted by VGG network.
- Improves the generalization of anti-spoofing system.

1.3 Organization of the PhD Dissertation

This section provides a brief outline to the flow of the upcoming chapters starting from chapter one.

Chapter 1 presents the background of the topic, research problems, objectives, and contributions of this PhD dissertation.

Chapter 2 provides background concerning the concepts and methods that are used in this research. It briefly reviews face recognition and spoofing attacks. It also reviews the state-of-the-art approaches. After this, some recent developments and trends in face recognition and face spoofing attacks are briefly discussed.

In Chapter 3, we avoid the supervised DNN training by using only unsupervised autoencoder and Siamese network. In this chapter, we explain our proposed unsupervised autoencoder: firstly, using only the k most similar images, then, secondly also using both k most similar and dissimilar images approach. The unsupervised selection of k most similar and dissimilar images algorithm is also discussed in this chapter. We also describe our CNN based Siamese network which is trained using the k most similar and dissimilar face images pairs. The performance of the proposed approach is also evaluated on face verification benchmark datasets.

Chapters 4 focus on the poor generalization and performance issue of face anti-spoofing methods and investigate the importance of image quality features in addition to deep learning approaches for improving the performance of the face anti-spoofing system. It also describes how the image quality features are selected.

Finally, Chapters 5 provides conclusion remarks and future work.

Chapter 2

Background and Related Works

The humans can easily and successfully perform face recognition task using their eyes. However, the automatic human face recognition still far from optimal. In this chapter, a detailed view of the human face recognition methods is presented. Researchers introduced variant algorithms will be described. It also provides an overview of the face recognition process such as preprocessing, face detection, feature extractions and classification. The main purpose of the features extraction is to reduce the image dimension by selecting the most significant features with retaining the relevant information and should be diverse enough among classes for good classification performance. However, the strength of the features extraction methods relies on strong preprocessing approaches. The extracted features can be used to classify and to recognize patterns that are present in the source images. Therefore, the preprocessing and feature extraction processes are the key point of the classification performance.

In this chapter, we also provide a general introduction to the different attacks that can be launched against face recognition systems. Then, we focus on the spoofing attacks and types of spoof attack detection

systems to such kind of attacks. We also give an overview on the state-of-the art face anti-spoofing methods.

Finally, this chapter also briefly explains the different evaluation metrics used in face verification and anti-spoofing detection systems for assessing the performance of the countermeasure. At the end it also explains the datasets used in this PhD dissertation.

2.1 Face Recognition

Among all biometric methods for human recognition, face recognition has been used more frequently. Face recognition is the main biometric used by human beings. When two people meet each other, their brains run a variety of biometrics based on height, age, hair color and style, skin color, etc. However, the final decision of the other person's identity is made mainly based on his/her face; hence face is assumed to be the part of the body that carries more information than other parts.

Automatic face recognition has many commercial and security applications in identity recognition and has become one of the hottest topics in image processing and pattern recognition since 1990. Availability of feasible technologies as well as the increasing request for reliable security systems in today's world has been a motivation for many researchers to develop new methods for face recognition. Automatic recognition of human faces continues to attract researchers from different areas such as computer vision, image processing, pattern recognition, neural networks, and psychology and has been increasingly accepted by the general public to be used in authentication, security and law enforcement.

An automatic face recognition system is usually a procedure of four main stages. In most cases these four stages are namely: preprocessing, face detection, feature extraction and finally classification. The input images obtained from image acquisition devices e.g., cameras, might not be suitable for recognition due to noise or illumination conditions. Therefore, first step is the preprocessing stage. Then faces should be detected in input images. Some face detection methods are presented in this chapter. Next step would be to extract features in order to make a feature vector. These features must include distinctive information about each person to recognize the individual based on these features. And finally, the last stage is the classifier where we intend to recognize an unknown person by assigning a class to its feature vector.

2.1.1 Face Identification vs. Verification

Biometric methods for face are used for either one of two the functions: face identification or face verification, as shown in Figure 2.1. In face identification, the goal is to identify an individual against a database of previously collected individuals. In other words, systems which are designed for the purpose of identification will answer the question: “Who the person is?”, as shown in Figure 2.1. It often takes place when users are not aware that it is happening; they don’t participate in the process or directly benefit from it, and their privacy is not protected. For example, a camera in a public place could be matching faces that it spots on the street with a database of criminals.

In the verification applications, on the other hand, we desire to verify whether the individual is the person that they claim to be. Biometric

methods designed for verification purposes answer the question: “Is the person who he says he is?”, as shown in Figure 2.1 (image was taken from [1]). It takes place when a user needs to verify their identity or authenticate themselves. For example, if you want to apply for a driver’s license or a credit card online, you need to prove that it is genuinely you that is accessing your account and that you are not an imposter that is attempting to impersonate you.

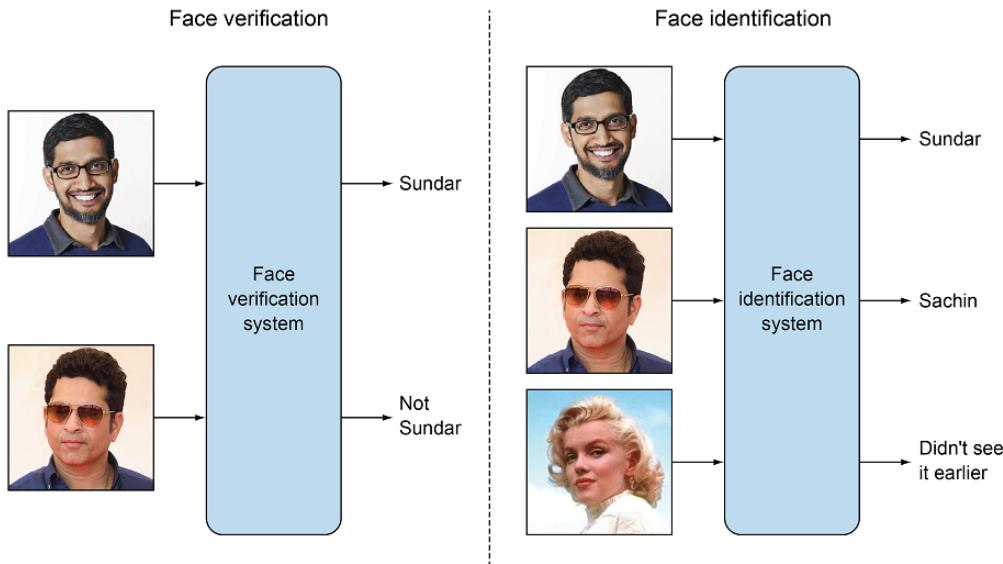


Figure 2.1: Example of face identification vs. verification

2.1.2 Preprocessing

The first step in most face recognition systems is preprocessing. The input images acquired via still or video cameras might have noise. Histogram equalization is the most common method used for image enhancement when images have illumination variations [31, 32]. Even for images under controlled illumination, histogram equalization improves the recognition results by flattening the histogram of pixel intensities of the images. The proposed Gamma intensity correction method in [33] used it for illumination normalization along with histogram equalization. They also presented a region-based

method for equalizing histogram and gamma locally in small portions of an image in their work.

Face images often contain background clutter that reduces the accuracy of face detection and facial recognition systems. To improve this, preprocessing methods are needed to remove the unneeded data from an input image before image recognition. Image cropping removes unnecessary surrounding material from the images for some specific reason. Image post-processing can help to extract relevant data. For example, many extraction methods are used in face detection systems to ensure the face in the image crop is in the most suitable position. Image filtering algorithms reduce the effect of noise on the image. As a result, image filtering improves the gray-level coherence, background white-noise, and smoothness. In addition, the regularized inverse auto-regressive (RIR) filter also results in a sharpened output image. Image de-noising is the process of detecting meaningful features in an image and enhancing them while suppressing background noise. Examples of useful features to improve include lines and edges and operations that can fill in gaps to create a complete representation of an prove that it is genuinely you that is accessing your account in the target image. Image filtering is any modification that alters some characteristic of an input image to obtain an altered output image. For example, spatial filtering modifies the intensities of pixels. Spectral filtering changes properties like hue and saturation. Other filters can refine temporal characteristics, like motion blurriness.

2.1.3 Face Detection

Face detection is the process of locating a face in an image. Detecting faces in a photograph is easily solved by humans, although has historically been challenging for computers given the dynamic nature of faces. For example, faces must be detected regardless of orientation or angle they are facing, light levels, hair color, facial hair, makeup, age, and so on. Although many face detection algorithms existed before 2001, a major breakthrough in face detection appeared with the Viola-Jones face detection method [34, 35]. Unlike previous face detection methods that relied on pixel analysis, Viola-Jones devised an algorithm called “Haar-classifier” that relied on Haar-like features. The Haar classifier is a machine learning algorithm that is trained with many positive and negative samples to detect objects in images. More recently deep learning methods such as Multi-task Cascaded Convolutional Networks (MTCNN) [36] have achieved state-of-the-art results on standard benchmark face detection datasets. MTCNN is a framework developed as a solution for both face detection and face alignment. The process consists of three stages of convolutional networks that are able to recognize faces and landmark location such as eyes, nose, and mouth.

2.1.4 Feature Extraction

The main function of this step is to extract the features of the face image which is detected in the detection step. This step represents a face with a set of features that describes the prominent features of the face image such as mouth, nose, and eyes with their geometry distribution [37, 38]. Each face is characterized by its structure, size, and shape, which allow it to be identified. Several techniques involve

extracting the shape of the mouth, eyes, or nose to identify the face using the size and distance [39]. The below techniques are widely used to extract the face features.

Histogram of oriented gradients (HOG) [40] is one of the best descriptors used for shape and edge description. The HOG technique can describe the face shape using the distribution of edge direction or light intensity gradient. The process of this technique is done by dividing the whole face image into small regions; a histogram of pixel edge direction is generated for each small region; finally, the histograms of the small regions are combined to extract the feature of the face image, [41, 42]. The magnitude of the gradient and the orientation of each pixel in the small region are voted in nine bins with the tri-linear interpolation. [40] proposed a combination of different histograms of oriented gradients (HOG) to perform a robust face recognition system. This technique is named “multi-HOG”. The authors create a vector of distances between the target and the reference face images for identification. [43] proposed a novel face recognition system based on the Laplacian filter and the pyramid histogram of gradient (PHOG) descriptor.

Eigenface [44] is one of the popular methods used to extract feature points of the face image. This approach is based on the principal component analysis (PCA) technique. The principal components created by the PCA technique are used as a face templates. The PCA technique transforms several possibly correlated variables into a small number of incorrect variables called “principal components”. The purpose of PCA is for reducing the dimensionality of datasets, increasing interpretability but at the same time minimizing information loss. It does so by creating new uncorrelated variables that successively maximize variance. PCA calculates the Eigenvectors of

the covariance matrix and projects the original data onto a lower dimensional feature space, which are defined by Eigenvectors with large Eigenvalues. PCA has been used in face recognition, where the Eigenvectors calculated are referred to as Eigenfaces.

Independent component analysis (ICA) [45] is a statistical and computational technique used in machine learning to separate a multivariate feature into its independent non-Gaussian components. ICA assumes that the observed data is a linear combination of independent, non-Gaussian features. The goal of ICA is to find a linear transformation of the data that results in a set of independent components, which allows the analysis of independent components. It is determined that they are not orthogonal to each other. In addition, the acquisition of images from different sources is sought in uncorrelated variables, which makes it possible to obtain greater efficiency, because ICA acquires images within statistically independent variables.

The authors in [46] proposed a hybrid approach which is combining Gabor wavelet and linear discriminant analysis (HGWLDA) for face recognition. The grayscale face image is approximated and reduced in dimension. The authors have convolved the grayscale face image with a bank of Gabor filters with varying orientations and scales. After that, a subspace technique 2D-LDA is used to maximize the inter-class space and reduce the intra-class space. To classify and recognize the test face image, the k-nearest neighbor (k-NN) classifier is used.

Scale invariant feature transform (SIFT) [47, 48] is an algorithm used to detect and describe the local features of an image. This algorithm is widely used to link two images by their local descriptors,

which contain information to make a match between them. The main idea of the SIFT descriptor is to convert the image into a representation composed of points of interest. These points contain the characteristic information of the face image. The four steps of the algorithm is: (1) detection of the maximum and minimum points in the space-scale, (2) location of characteristic points, (3) assignment of orientation, and (4) a descriptor of the characteristic point. A framework to detect the key-points based on the SIFT descriptor was proposed by [47], where they use the SIFT technique in combination with a Kepenekci approach for the face recognition.

The authors in [49] propose Gabor filters are spatial sinusoids located by a Gaussian window that allows for extracting the features from images by selecting their frequency, orientation, and scale. To enhance the performance under unconstrained environments for face recognition, Gabor filters are transformed according to the shape and pose to extract the feature vectors of face image combined with the PCA in the work of [49]. The PCA is applied to the Gabor features to remove the redundancies and to get the best face images description. Finally, the cosine metric is used to evaluate the similarity.

The authors in [50] propose Local binary pattern (LBP) is a great general texture technique used to extract features from any object. It has widely performed in many applications such as face recognition [39], facial expression recognition, texture segmentation, and texture classification. The LBP technique first divides the facial image in spatial arrays. Next, within each array square, pixel matrix is mapped across the square. The pixel of this matrix is a threshold with the value of the center pixel as a reference for thresholding to produce the binary code. If a neighbor pixel's value is lower than the

center pixel value, it is given a zero; otherwise, it is given one. The binary code contains information about the local texture. Finally, for each array square, a histogram of these codes is built, and the histograms are concatenated to form the feature vector.

The authors in [51] propose a fast face recognition system based on LBP, pyramid of local binary pattern (PLBP), a rotation invariant local binary pattern (RI-LBP). [52] have introduced a deep learning based technique, called local binary pattern network (LBPNet), to extract hierarchical representations of data. The LBPNet maintains the same topology as the convolutional neural network (CNN). [53] have implemented a method that helps to solve face recognition issues with large variations of parameters such as expression, illumination, and different poses. This method is based on two techniques: LBP and K-NN techniques. Owing to its invariance to the rotation of the target image, LBP become one of the important techniques used for face recognition. [54] proposed a variant of the LBP technique named “multiscale local binary pattern (MLBP)” for features’ extraction. Another LBP extension is the local ternary pattern (LTP) technique [55], which is less sensitive to the noise than the original LBP technique. This technique uses three steps to compute the differences between the neighboring ones and the central pixel. [56] develop a local quantized pattern (LQP) technique for face representation. LQP is a generalization of local pattern features and is intrinsically robust to illumination conditions. The LQP features use the disk layout to sample pixels from the local neighborhood and obtain a pair of binary codes using ternary split coding. These codes are quantized, with each one using a separately learned codebook.

FaceNet [6] is a unified system for face verification, recognition, and clustering. This method aims to extract an embedding vector for

each input image using a trained CNN network. The network is trained so that the square L2 distance of all embedding vectors in the embedding space simulates the similarity between the inputs; that is, faces of the same identity have a small distance while faces of different identities have a large distance. To achieve this kind of discriminative feature, this approach employs the triplet loss function. The FaceNet model is trained using 200M training faces of 8M different persons. In 2014, Facebook introduced DeepFace model [7]. DeepFace model was based on the Softmax loss function.

Softmax based models calculate the distance between the distribution of the output (ground truth) and the original distribution. Then, it normalizes a vector of logits (output of last FC layer) to be a probability distribution. The problem with these models is that they do not enforce separation between classes. To solve this issue, [57] introduced a discriminative feature learning approach for deep face recognition, this approach was based on a new loss function based on Softmax loss function called the center loss.

The center loss aims to calculate the center of each class and penalizes the distance between the feature and its corresponding class center. This can achieve intra-class compactness and inter-class disparity. However, measuring the center point for each class is computationally expensive because we need to calculate the distance between all features to find the center.

The authors in [58] introduced SphereFace, a deep Hypersphere Embedding for face recognition; this model is based on the Softmax loss function with some modifications. The author of SphereFace claims that features learned by Softmax loss have an intrinsic angular distribution. SphereFace utilizes this angular distribution by imposing

a discriminative constrain in a hypersphere manifold, allowing the intra-class and inter-class feature to be controlled by a parameter m . This method is called Angular Softmax "A-Softmax". The decision boundaries can significantly affect the feature distribution, so the basic idea is to manipulate decision boundaries to produce an angular margin.

The authors in [15] introduced CosFace, which is like SphereFace. CosFace adopted the idea of utilizing softmax natural angular distribution but introduced another angular margin technique called Large Margin Cosine Loss that can better maximize inter-class variance and minimize intra-class variance. This model only emphasizes correct classification but does not enforce discriminative features. To introduce the margin, CosFace implements the same idea of manipulating decision boundaries to produce angular margin.

The authors in [13] introduced ArcFace, an Additive Angular Margin Loss for Deep Face Recognition. Similar to SphereFace and CosFace, ArcFace utilizes Softmax natural angular distribution but introduces another angular margin technique. The authors of ArcFace proposed an Additive Angular Margin Loss function further to improve the face recognition model's discriminative power and stabilize the training process. Now the prediction depends only on the angle between the weight and the feature. The learned embedding features are distributed on a hypersphere of a radius s . And to intensify the intra-class compactness and inter-class disparity, an angular margin penalty m is added.

2.2 Face Anti-spoofing

Since deployment of face recognition systems is growing year after year, people are becoming more familiar with their use in daily life. Consequently, security weaknesses of face recognition systems are getting better known to the general public. As shown in Figure 2.2, (image was taken from [59]), there are nine different point where someone can compromise the security of a face recognition system.

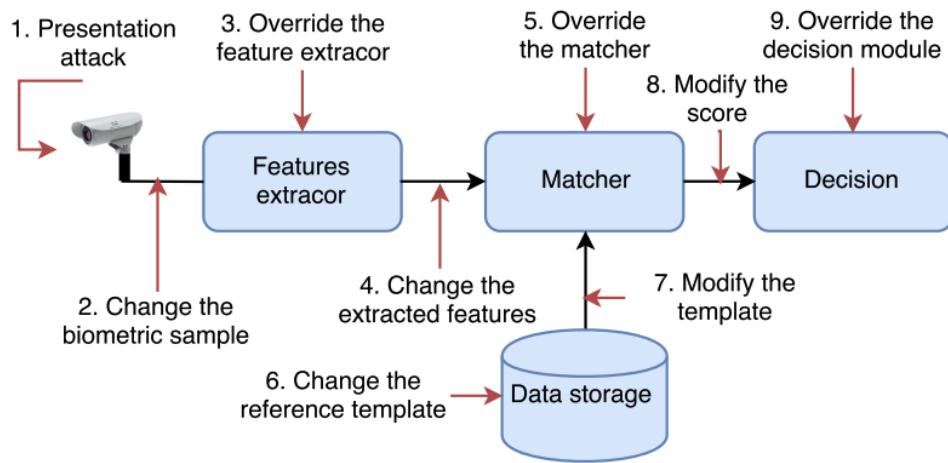


Figure 2.2: Different points of attacking a face recognition system.

These attacks can be divided into direct attacks and indirect attacks [59]. The direct attacks are performed outside the face recognition system (point 1) and they consist of presenting face artifacts in front of the sensor. According to the ISO/IEC JTC1 SC37 standard [60] an artifact or Presentation Attack Instrument (PAI) is "an artificial object or representation presenting a copy of biometric characteristics or synthetic biometric patterns". The direct attacks are also known as spoofing attacks or Presentation Attacks (PA) [60]. Contrary to the direct attacks, the indirect attacks are performed inside the biometric systems. These attacks can be done by intercepting the biometric sample captured by the sensor and replacing it with a fake

sample (point 2), overriding the feature extractor module by changing their functionality (point 3), replacing the extracted features of the captured face with pre-computed features of the target face (point 4), overriding the matcher to output a required score (point 5), replacing the reference template with the attacker template (point 6), modifying the template in the communication channel (point 7), changing the output score (point 8), or finally, by overriding the decision module to output the intended decision (point 9). Among these different attacks, the spoofing attacks have gained a wide interest from the biometric community because: it is easy to create an artifact and present it in front of a face recognition camera; it does not require any knowledge about the operational details of the biometric system; and it does not need any hacking or advanced programming skills. Therefore, we focus on direct attacks. For the indirect attacks, the security can be increased using different measures [61] that include but are not limited to firewalls, anti-virus, intrusion detection and encryption.

2.2.1 Spoofing Attacks

The biometric data used in the creation of the face spoofing attacks can be 2D cut photo, 3D images, or video sequences as shown in Figures 2.3 (image was taken from [62]) and 2.4 (image was taken from [63]). Face recognition systems rely on data which are personal in nature but, nevertheless, are already public. Using good camera devices, someone can easily capture face images or video sequences of a target person from distance without his/her permission. Moreover, with the increase of the internet utilization, many people are sharing their pictures and videos in the social media websites, such as Facebook and Twitter, and personal or professional web-pages.

Thus, it is easy to download these images or videos and use them to create face artifacts.

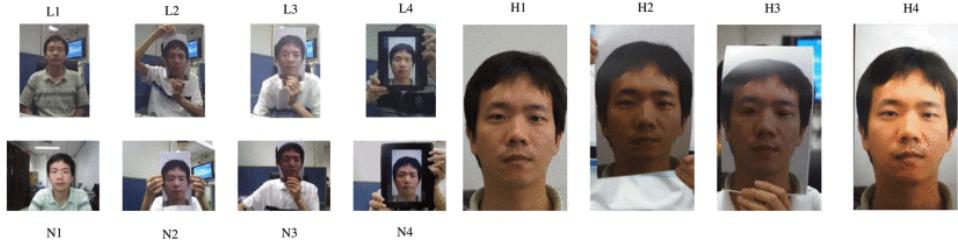


Figure 2.3: The left top four images represent the low quality videos, the left bottom are the normal quality videos, and the right are the high quality videos. For each quality, from left to right are genuine, warped photo attack, cut photo attack and video attack.

The 2D face images can be printed using a high-quality printers on glossy photo papers to create print attacks [28, 62, 64] or displayed on electronic display devices to generate photo display attacks [64]. The attacks based on the 2D images retain only the face appearance and they have no sign of liveness. To give some level of liveness to these attacks, the attacker may hold the spoofed face image in his/her hand and try to move it in a way to simulate the real face movements (e.g., translating, rotating, or warping) [62, 64]. Furthermore, to simulate the eye blinking, the eye regions can be cut, and the attacker hides behind the cutting pictures and exhibits eye-blinking through the holes [62]. In the case of photo display attacks, the 2D images can be animated using image processing software then replayed on the display devices. It is obvious that the real face movements are different from these artificial movements. However, the motion detection methods are not perfect, and these artificial movements can increase the error of the face spoofing attack methods based on motion analysis.

To include both the appearance and the liveness of the real faces, in a more sophisticated way, video sequences of the targeted faces

are used. These videos are first replayed on display devices, such as laptops, tablets, smart-phones, then presented in front of the face recognition camera. The quality of display attacks depends mainly on two factors: the quality of face images or videos and the quality of the display devices.

The print and the display attack instruments can be presented in front of the camera using fixed or hand support. Fixed support is more appropriate for presenting the video display attacks as it prevents from the creation of other motions different from the motions of the real faces. On the other hand, hand support is suitable for presenting print attacks and photo display attacks as it gives them some level of liveness.

The use of print and display attacks is only restricted to the face recognition systems operating in an unsupervised scenario (i.e., the recognition process is not assisted by an agent). Thus, to spoof the recognition systems operating on a supervised scenario (e.g. the face recognition systems in the airports), more sophisticated attacks are needed. The best choice for spoofing this kind of systems is the use of 3D mask attacks, Figure 2.4.

Nowadays, with the advancement in the 3D printing technology, it is easy to create 3D mask of a targeted face from its 3D images. Getting images of this kind is quite hard compared to the 2D images as more advanced devices (e.g., 3D scanner) and user cooperation are needed. However, it is easy to get 3D masks by just getting a set of 2D face images (usually, two images, frontal and profile images). The materials used in the creation of these 3D masks have a big impact on the quality of the attacks.



Figure 2.4: Sample face images of 3D mask attacks

There are several ways of detecting spoofing attack. Below we will review three of them namely hardware based, image quality based, and deep learning based.

2.2.2 Hardware Based Methods

The hardware-based methods use advanced materials to differentiate between the real and the fake face samples. Using a 3D cameras [22] or multispectral cameras [65, 66], we can get additional useful information about the depth and the reflectance properties of the observed faces. Thermal cameras can also be used to detect the print attacks, replay attacks, and even some plastic surgery [65, 66]. Surgical operations usually cause alteration in blood vessel flow that can be seen as cold spots in the thermal domain. Recently, light field cameras capable of rendering multiple depth images in a single capture [67], and optical filter systems providing horizontal and vertical

light polarization measurements [68] have shown promising results in print and video-replay attack detection.

In addition to the difficulty of integrating these additional hardware devices into existing face recognition systems and the high cost of some advanced materials, the hardware-based methods are powerless for detecting some kind of attacks under some circumstances. For instance, the depth and the thermal cameras are useless against the 3D masks attacks. It is known that thermal radiation can pass through materials, which causes problems when thermal information is used against wearable mask attacks [65]. The hardware-based techniques have also difficulties in capturing the reflectance disparities between genuine faces and 3D masks due to the 3D shape and a variety of artificial materials [65].

2.2.3 Image Quality Feature Based Methods

There are several ways of extracting hand-crafted features, one of them is using image quality features. Below we will review the most known image quality feature-based methods for face anti-spoofing detection.

Presenting a spoofed human face requires plastic, photo paper, printing paper, and other media with qualities that differ from a real face's facial features and skin materials. There is a variance in the reflection quality of materials, such as picture paper and display screens. Both of which exhibit specular reflections but no living faces. Most of the picture quality after spoofed face differs from that of a living face, such as color distribution distortion and blurring of the spoofed face image, even though the spoofed of the face manufacturing process is high. Image quality-based techniques use the variance among

reflection and image distortion qualities to distinguish genuine and spoofed faces.

The authors in [69, 70] introduced a new facial spoof detection method based on Image Quality Assessment (IQA), assuming that a spoofing image captured in a photo or video replay attack should have a different quality than a genuine sample, as it was captured twice instead of once for genuine faces, [71]. Eighteen and twenty-five image quality measures were adopted in [69, 70], respectively, to assess the image quality using scores extracted from single images. Then, the image-quality scores were combined as a single feature vector and fed into a Quadratic Discriminant Analysis (QDA) classifier to perform facial spoofing attack detection. The major advantage of the IQA-based methods is that it is not a application specific method, so this is a “multi-biometric” method that can also be employed for iris or fingerprint-based anti-spoof detection.

The authors in [28] also proposed an IQA-based method, using analysis of image distortion, for facial anti-spoof detection. [28] method analyzes the image chromaticity and the color diversity distortion in the HSV (Hue, Saturation and Value) space. The idea here is to detect imperfect/limited color rendering of a printer or LCD screen. The image distortion feature (which consists of a specular reflection feature [72], blurriness feature [73, 74], a chromatic moment feature [75] and a color diversity feature) is fed into SVM. The proposed method has shown a promising generalization performance when compared with other image quality based anti-spoof detection methods.

2.2.4 Deep Learning Based Methods

Deep learning-based algorithms have been effectively applied to various disciplines, including video, speech recognition, medical imaging applications, security, anomaly, and so on.

Deep learning in the machine learning field achieved numerous performances in the computer vision and the processing of human language applications [76–80]. Deep learning is driven by understanding how the human brain processes information. The brain is organized as a deep architecture with several layers that process the information among many levels of non-linear transformation and representation [81]. Deep learning learns the hierarchy, structure, and pattern of the features from the lower level features using multilevel of hidden layers of non-linear transformations [79]. Very complex functions can be learned with enough such transformations. The higher layers of representation increase aspects of the inputs that are important for discrimination and suppress irrelative variation for any object recognition. For human face recognition, higher layers of representation amplify features of the inputs that are significant for discrimination and subdue irrelative features [82]. The first layer learns the low-level features such as curves, edges, and point from the image pixels. The low-level features are combined in the following layers to produce higher features; for example, points and combined into lines and curves then they combined into shapes and more complex shapes. Once this is done, the deep neural network delivers a probability that these high-level features contain a particular object or scene. The main goal of deep learning is to automatically learn the most discriminative features from the raw data without human involvement. Convolutional Neural Networks (CNN), Auto-encoder

(SA), Recurrent Neural Network (RNN), and Deep Belief Network (DBN) are the popular models for deep learning. [83, 84].

More recently, deep learning based methods are used to detect spoofing attack. Researchers studying these methods focus on designing an appropriate neural network so as to learn the best features rather than to design the features themselves (as is the case with most hand-crafted features presented above).

The first attempt to use Convolutional Neural Networks (CNNs) for detecting spoofing attacks was claimed in [27]. In this method, AlexNet [85] is used for learning the features that best discriminate spoofing attacks. It was the first time that CNNs were proven to be effective for automatically learning features for face anti-spoofing. This method has surpassed almost all the existing state-of-the-art methods for photo and video replay attacks. It showed the potential of deep CNNs for face anti-spoofing. Later, more and more CNN-based methods were explored for facial anti-spoofing.

The authors in, [86] proposed an end-to-end framework based on AlexNet [85], namely CaffeNet, for facial anti-spoofing. The proposed CNN was pretrained on ImageNet [79] and WebFace [87] to provide a reasonable initialization and fine-tuned using the existing face anti-spoofing databases. More specifically, two separate CNNs are trained, respectively from aligned face images and enlarged images including some background. Finally, a voting fusion is used to generate a final decision. Just like Yang et al.’s method [27], the proposed CNN-based method has surpassed the state-of-the-art methods in face anti-spoofing.

The authors in, [88] proposed to train a deep CNN based on VGG-Face [8] for facial anti-spoofing. As in [86], the CNN was pretrained

on massive datasets and fine-tuned on the facial spoofing database. Furthermore, the features extracted from the different layers of the CNN were fused to a single feature and fed into an SVM for facial anti-spoofing. Principal component analysis (PCA) and the so-called part features are used to reduce the feature dimension. To obtain part features, the mean feature map in a given layer is firstly calculated. Then, the critical positions in the mean feature map are selected, in which the values are higher than 0.9 times the maximum value in the mean feature map. Finally, the values of the critical positions on each feature map are selected to generate the part feature. The concatenation of all part features of all feature maps is used as the global part feature. Then, PCA is applied on the global part feature to further reduce the dimension. Finally, the condensed part feature is fed into an SVM to discriminate between genuine and spoofed faces. Benefiting from using a deeper CNN based on VGG-Face, the proposed method has achieved state-of-the-art performances in both intra-dataset and cross-dataset scenarios for detecting face spoofing attacks.

The authors in, [89] proposed to estimate the noise of a given spoof face image to detect photo/video replay attacks. In this work, the spoof image was regarded as the summation of the genuine image and image-dependent noise introduced while generating the spoof image. Since the noise of a genuine image was assumed as zero in this work, a spoof image can be detected by thresholding the estimated noise. A GAN framework based on CNNs, De-Spoof Net (DS Net), was proposed to estimate the noise. However, as there is no noise ground-truth, instead of assessing the quality of noise estimation, the authors de-noise the spoof images and assess the quality of the recovered (de-noised) image using Discriminative Quality Net (DQ

Net) and Visual Quality Net (VQ Net). Besides, by fusing different losses for modelling different noise patterns in DS Net, the proposed method has shown a superior performance compared to other state-of-the-art deep facial anti-spoofing methods such as in [90].

The authors in, [91] proposed Deep Pixelwise Binary Supervision (DeepPixBiS), based on DenseNet [92], for facial anti-spoofing. Instead of only using the binary cross-entropy loss of the final output as in [86], DeepPixBiS also uses during training a pixel-wise binary cross-entropy loss based on the last feature map. Each pixel in the feature map is annotated as 1 for a genuine face input and as 0 for a spoof face input. In the evaluation/test phase, only the mean value of pixels in the feature map is used as the score for facial anti-spoofing. DenseNet and the proposed pixel-wise loss forcing the network to learn the patch-wise feature, DeepPixBiS showed a promising anti-spoofing performance for face spoofing attacks.

The authors in, [93] proposed to use NAS to design a neural network for estimating the depth map of a given RGB image for facial anti-spoofing. The gradient-based DARTS [94] and Pc-DARTS [95] search methods were adopted to search the architecture of cells forming the network backbone for facial anti-spoofing. Three levels of cells (low-level, mid-level and high-level) from the three blocks of CNNs in [90] were used for the search space. Each block has four layers, including three convolutional layers and one max-pooling layer, and is represented as a Directed Acyclic Graph, with each layer as a node.

The authors in, the image depth information is crucial for determining the face’s validity as the face in real life is three-dimensional, whereas the face attacked by photographs and screens is flat. The

depth map differs from the real face, even if the face is deformed. A two-channel CNN-based face anti-spoofing method was proposed in this study [96]. Spatial characteristics of faces, such as texture and depth, are essential, but temporal factors are even more critical for anti-spoofing. Examining a human face from a time and space viewpoint can provide more helpful information and enhance classification performance.

The authors in, A unique approach is presented [97] that reformulates the Generalized Presentation Attack Detection (GPAD) problem from the standpoint of anomaly detection. A deep metric learning model was provided. A triplet focal loss is a regularization for a novel loss called ‘metric-SoftMax.’ It guides the learning process towards more discriminating feature representations in an embedding space. Finally, the benefits of deep anomaly detection architecture are proven by introducing a few-shot posterior probability calculation that does not require any classifier to be trained on the learned features.

Chapter 3

An Unsupervised Deep Learning Face Verification System

Face recognition is a technology that identifies or verifies a person from an image or video [98]. Generally, face verification is used to access an application, system or service. The task is to compare a given face to another face and verify whether it is a match. In other words, given any two face images, the face verification algorithm decides if they are of the same person or not. Unlike other verification methods such as using passwords or fingerprints, biometric face verification uses dynamic patterns that make this approach one of the safest and most effective ones. Face recognition is also used in forensics and transaction authentication.

Deep neural networks have been successfully used in different applications such as speaker verification [99, 100] and image recognition [101, 102]. In addition to artificial neural networks, spiking neural networks have been successfully used for image recognition [103, 104].

It was shown that using deep neural networks for face verification [6–19] significantly improved accuracy when compared with other face verification systems [105–110]. The Facenet [6] face verification

system was developed by Google; it used a Siamese network [111] trained on a labeled dataset with 200M faces. It achieved an accuracy of over 98% on LFW [112] and over 95% on YTF [113], two benchmark face verification datasets. To achieve that result, it used a huge, labeled dataset, with 200M faces, for training. DeepFace [7] was developed by Meta. It used 3D face modeling and a nine-layer network with about 120 million parameters and was trained on 4.4M labeled face images. On the LFW dataset, it achieved an accuracy of over 97%. DeepFace was extended in [9] and, by using much more training data—over 500M faces—improved its performance on LFW to over 98%. Another face verification system, VGG Face, was developed at Oxford [8], used 37 convolutional layers and was trained on 2.6M labeled face images. It achieved accuracies comparable to Facenet and DeepFace on LFW, and over 97% on the YTF dataset. In [10], another face verification system was proposed using marginal loss, which was trained on a 4M labeled dataset, and achieved an accuracy of over 99% on LFW and over 95% on YTF. ArcFace [13] used an additive angular margin loss and obtained over 99% accuracy on LFW, over 98% on both YTF and CFP-FP, and over 95% on CALFW. GroupFace [14] used multiple group-aware representations and achieved over 99%, 97%, 96% and 98% on the LFW, YTF, CALFW and CFP-FP datasets, respectively. However, both ArcFace and GroupFace required labeled training data of 5.8M samples. MegaFace [114] deployed a magnitude-aware margin on ArcFace loss to improve intra-class compactness and achieved over 96% and 98% on CALFW and CFP-FP datasets, respectively. CurricularFace [16] used an adaptive curriculum learning loss and achieved over 99% on LFW, over 96% on CALFW and 98% on CFP-FP datasets. Both CurricularFace and MegaFace required about 3.8M labeled training

data. MDCNN [115] is composed of two advanced deep learning neural network models and achieved over 99% and 94% on the LFW and YTF datasets, respectively, using a 1M labeled training dataset. PSO AlexNet TL [116] used transfer learning and achieved an accuracy of over 99% on the LFW dataset. Ref. [117] used data augmentation and achieved over 99% and 96% on the LFW and YTF datasets, respectively.

Semi-supervised learning methods with deep neural networks use two main approaches: (1) consistency regularization-based methods [118] and (2) proxy label-based methods [119]. The consistency regularization-based methods use a regularization term in the objective function to enable consistency while training on a large amount of unlabeled data; this constrains model predictions to be invariant to input noise. Ref. [118] developed an Unsupervised Domain Adaptation method with advanced data augmentation methods such as rand-augment and back-translation. The proxy label-based methods first assign proxy labels to unlabeled data (pseudo-labels) and then train unlabeled and labeled data based on proxy and ground-truth labels. Ref. [119] introduced a FixMatch method that first generates pseudo-labels using the model’s predictions on weakly augmented unlabeled images.

Several methods were proposed to learn features from unlabeled data, which can significantly reduce the high cost of annotating large-scale data. For example, Ref. [120] introduced DeepCluster, a clustering method that jointly learns the parameters of a neural network and the cluster assignments of the resulting features. Ref. [121] proposed learning image features by training ConvNets to recognize the 2D rotation that is applied to the image it receives as input. Ref. [122] proposed Spatial-Semantic Patch Learning, which involves

two stages in training. First, three auxiliary tasks, consisting of a Patch Rotation Task, a Patch Segmentation Task and a Patch Classification Task, are jointly developed to learn the spatial-semantic relationship from large-scale unlabeled facial data. Ref. [123] proposed to enhance face recognition with a bypass of self-supervised 3D reconstruction. Ref. [124] proposed a face frontalization framework combined with 3DMorphableModel that only adopts front images for training. The authors in [125] proposed a fully trained generative adversarial network to generate realistic and natural images. In [126, 127], the authors proposed face synthesis and pose-invariant face recognition using generative adversarial network. PCA feature transform, Correlation Alignment [128] and Unsupervised Domain Adaptation for Face Recognition in Unlabeled video [117] methods were proposed to extract features using RFNet. The adaptation was achieved by distilling knowledge from the network to a video adaptation network through feature matching, performing feature restoration through synthetic data augmentation and learning a domain-invariant feature through a domain adversarial discriminator.

All the above-described methods, as is true for most other deep neural networks, require large amounts of labeled training data, which are not available in many domains. Moreover, in many real-world applications, sufficient labels can be difficult to collect. As a result, the performance of these methods greatly degrades.

To address this problem, we propose an unsupervised deep learning face verification system using k most similar and k most dissimilar images, called UFace. The k most similar and k most dissimilar images is calculated for a given face image. UFace does not require labeled data and, importantly, uses only about 200 K unlabeled face

images. However, based on the experimental result, UFace substantially improves the results of unsupervised methods because it takes into account the similar and dissimilar face images to extract distinct features.

UFace was evaluated on four benchmark face recognition datasets: LFW, YTF, CALFW and CFP-FP. The experimental results of UFace provide accuracies that are comparable with state-of-the-art methods such as ArcFace, GroupFace, MegaFace, Marginal Loss and VGG Face.

3.1 The Proposed Preprocessing Method

UFace first performs two preprocessing tasks, as shown in Figure 3.3. The first processing step is to detect a face from a given image using Multi-Task Cascaded Convolutional Neural Network (MTCNN) [36], which locates a face in a given image and draws a bounding box around it (see Figure 3.1b). It provides coordinates of the lower left corner of the bounding box plus its width and height then resizes the image size to 112 by 112 pixels.

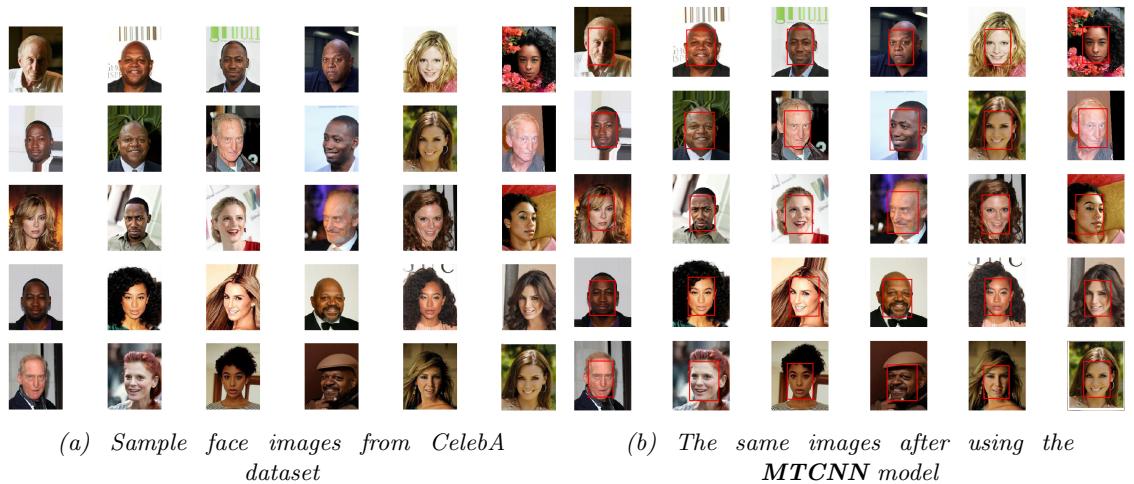


Figure 3.1: Sample images after MTCNN was used for face detection.

Secondly, it generates embedding vectors using the pre-trained Facenet model [6]. Then, we find the k most similar and k most dissimilar images for each image in the preprocessing phase. Note that Facenet is used here just to help calculate the cosine similarity/dissimilarity between images during the preprocessing stage, i.e., we did not use Facenet to train our models.

Algorithm 1 calculates the cosine similarity between a given image and all other remaining images in a dataset. Next, a threshold is used to select the k most similar and k most dissimilar images for each input image from the training dataset. To select the k most similar and k most dissimilar images, we experimented with different threshold values on validation set and empirically decided to use the optimal threshold (i.e., one that resulted in the highest accuracy). The optimal threshold value was found to be 0.6 for the most similar images and 0.2 for the most dissimilar images. In this way, we make sure any of the similar images are not the same as the dissimilar images.

Note that the value of k varies from image to image since a face can have a different number of most similar images. On average, however, we discovered that there are about 11 similar and dissimilar images for each image. In total, we created about 4M training pairs (both for the similar and dissimilar pairs) for all images in the CelebA dataset (which has only about 200k images). The selection of the threshold value that is used to select the similar and dissimilar images is described in detail in the experimental section.

Note that we used the Facenet pre-trained model only to calculate the cosine similarity between images during the pre-processing phase. However, the UFace training methods do not require to use Facenet

Algorithm 1 To select the k most similar and k most dissimilar images for each image in a dataset.

Require: The thresholds ths and thd , and m training images x
Ensure: k most similar (\tilde{x}_{is}) and k most dissimilar images (\tilde{x}_{id}) for each image in a dataset, $1 \leq i \leq m$, $1 \leq p \leq k$ and $1 \leq n \leq k$

```

for  $i \leftarrow 1$  to  $N$ ,  $N \leftarrow \text{length}(m)$  do
    for  $j \leftarrow 1$  to  $N$ ,  $N \leftarrow \text{length}(m)$  do
        if  $i \neq j$ 
             $\tilde{x}_{ij} = \text{cosine}(x_i, x_j)$ 
        end
    end
    Select  $k$  most similar images above the  $ths=0.6$ ,  $\tilde{x}_{is}$  and randomly select  $k$  most dissimilar images below the  $thd=0.2$ ,  $\tilde{x}_{id}$ 
end

```

and do not require explicitly labeled training data, as described in the training section.

3.2 The Proposed Autoencoder Training Method

Note that the preprocessing and evaluation modules for both the autoencoder and Siamese networks are the same.

The state-of-the-art methods such as ArcFace [13], Facenet [6], GroupFace [14], CosFace [15], MegaFace [114], DeepFace [7], VGG Face [8] and Marginal Loss [10] require a very large amount of labeled data, which are difficult to obtain in many applications other than face images. For example, Facenet used about 200M training images.

To address this problem, we propose an unsupervised deep learning face verification system using k most similar and k most dissimilar images, called UFace. To demonstrate the performance of UFace, we started using only k most similar images and the autoencoder network for verification. Next, we used both the k most similar and the k most dissimilar images with autoencoder. Since the latter gave

better results than just using k most similar images, in the Siamese network we used both k most similar and k most dissimilar images.

Classical Autoencoder Training: An autoencoder is an unsupervised neural network used in situations when no labeled data are available [129]. It is a feedforward neural network where the output (the compressed version of the input) is trained to be almost the same as the input. Autoencoders were successfully used in feature extraction [130], dimensionality reduction [131], image denoising [132] and image inpainting [133]. Autoencoder compresses high-dimensional input data, such as an image, into a lower-dimensional (compressed) representation and is trained to recreate the original input from its output. The difference between the reconstructed and the input image is the reconstruction error. The network is trained to minimize this error to find the best lower-dimensional representation, called the embedded vector. The autoencoder (AE) consists of (see Figure 3.2a) an encoder and decoder.

Encoder: The encoder part of the network maps the original input image into its lower-dimensional representation h .

$$h = g((w * x) + b) \quad (3.1)$$

where w is a weight matrix between the input x and hidden layers, b is the bias and g is a nonlinear activation function.

Decoder: The decoder reconstructs the original input data from its encoded representation. In the decoding process, the AE maps h back to the original input approximation \hat{x} .

$$\hat{x} = f((\hat{w} * h) + \hat{b}) \quad (3.2)$$

where \hat{w} is a weight matrix between the output of the encoder and hidden layers, \hat{x} is the output data, \hat{b} is bias and f is a nonlinear activation function.

The Mean Square Error (MSE) measures the reconstruction error [134, 135]. The classical training is carried out by minimizing the average squared difference between the output value and the input value, as shown in Equation (3.3):

$$\text{MeanSquaredError}(MSE) = \frac{1}{m} \sum_{t=1}^m (\hat{x} - x)^2 \quad (3.3)$$

where x is the original input and \hat{x} is the predicated value.

To make a fair comparison of the classical AE system with UFace, we developed our own classical AE system. Both systems are developed exactly in the same way except how the reconstruction error is computed. The classical AE system computes the reconstruction error with one original input image, whereas UFace computes the reconstruction error with k most similar and k most dissimilar images.

UFace Autoencoder Training: The UFace method is first demonstrated using only similar images. It trains the autoencoder to reconstruct k most similar images of the input image. Then, UFace is demonstrated using both similar and dissimilar images. It trains the autoencoder to reconstruct the k most similar and k most dissimilar images of the input image rather than the single input image, as is the case with classical autoencoder training. UFace uses the k most similar and k most dissimilar images of the input image during calculation of the reconstruction error, which is backpropagated to update the network weights.

State-of-the-art methods such as Facenet [6], Fusion [9], DeepFace [7], VGG Face [8] and Marginal Loss [10] require a very large amount of labeled data, which may be hard to obtain in many applications other than face images. For example, Facenet used about 200M labeled training images.

To address this problem, we propose a novel training method that does not explicitly require a labeled training dataset. It trains the autoencoder to reconstruct the k most similar images of the input image rather than the single input image, as is the case with the classical autoencoder training. The new method uses the k most similar and k most dissimilar images of the input image during the calculation of the reconstruction error, which is backpropagated to update the network weights.

The autoencoder is trained by minimizing the loss function between the reconstructed image \hat{x} and the k most similar and k most dissimilar images of the original input image x for all images in the dataset.

The used training mechanism consider intra-person and inter-person face variabilities (k number of times), while in the classical autoencoder training mechanism, the loss function is computed only once. The value of k varies from image to image. In the first iteration, as shown in Equation (3.4), once the first input image is reconstructed it calculates the mean square error between the reconstructed image and the first k th most similar/dissimilar images (for the case of dissimilar images, it takes the negative value of the MSE). After calculating the error, it backpropagates the error to update the network parameters. In the second iteration, it continues training the same first input image and computes the mean square error with the

second k th most similar/dissimilar images, and it continues training in the same way using the remaining k th most similar/dissimilar images. Once training for the first input image is completed, it starts training for the second input image in the same way and continues for all images in the dataset. UFace calculation of the error is shown in Equation (3.4). The total number of training images is calculated as the sum of $f(j)$, where $f(j)$ is the function that outputs the total number of k most similar and k most dissimilar images in the training dataset. Since UFace computes the reconstruction errors $2k$ (k for the similar and k for the dissimilar images) times for each input face image, it accounts for face variabilities.

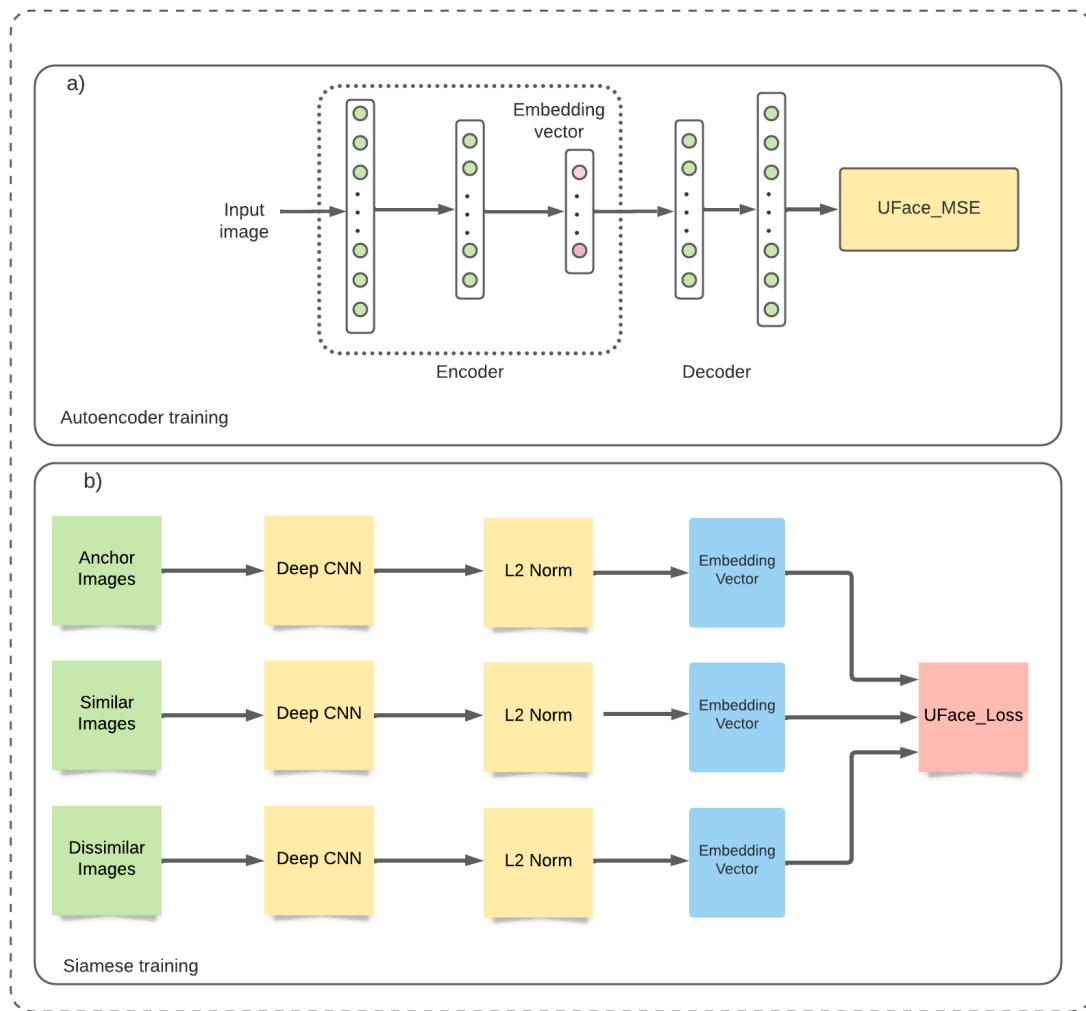


Figure 3.2: The proposed architectures used for training with autoencoder a) and with Siamese network b).

$$UFace_MSE = \frac{1}{\sum_{i=1}^m f(j)} \sum_{i=1}^m \sum_{j=1}^{f(j)} (\hat{x}_i - \tilde{x}_j)^2 \quad (3.4)$$

UFace_MSE is the UFace loss function, where m is the number of training images, $f(j)$ is the function that represents the variable number of k most similar images for the input image x_i , \tilde{x}_j is most similar images for input image x_i and \hat{x}_i is the reconstructed image for the input image x_i . Note that, for the case of dissimilar images, we take the negative of it since it will be maximized.

3.3 The Proposed Siamese Network Training Method

The architectures of the UFace system are shown in Figures 3.3, 3.2 and 3.4, which includes the three modules: preprocessing, training and evaluation, respectively.

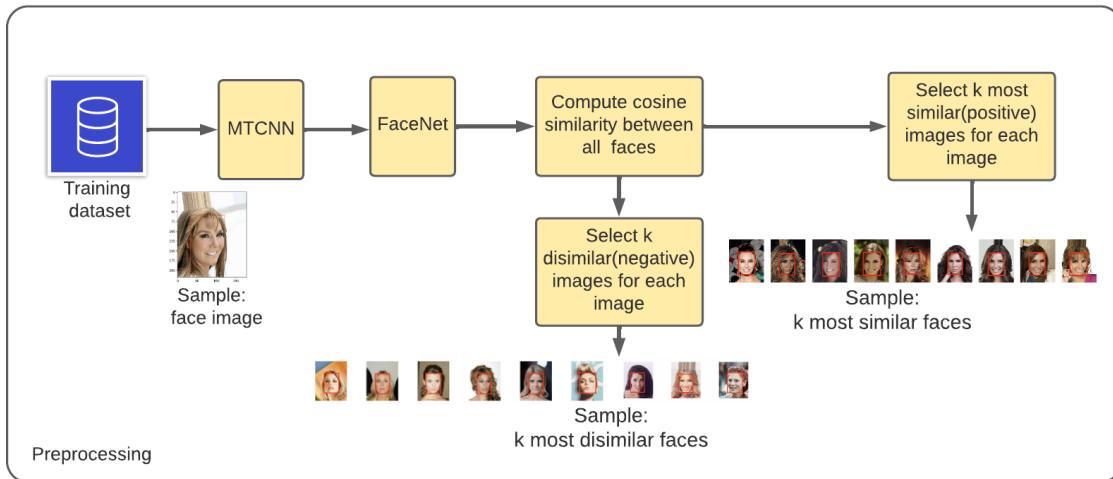


Figure 3.3: UFace preprocessing steps.

The UFace training method on Siamese network using both similar and dissimilar images is shown in Figure 3.2. It has three branches, each of which is the CNN encoder followed by the L2-normalization layer. The branches share the same weights. Branches for training

are fed by an anchor (input image), similar images and dissimilar images. The output of the CNN encoder is known as image embedding. After the L2-normalization layer, the UFace loss function—UFace_Loss (Equation (3.5))—is computed as the error between the embeddings of similar and dissimilar images and the anchor. The loss function reduces deviation between the anchor and similar faces and increases deviation between the anchor and dissimilar faces. While training a model to classify, it optimizes the weights to minimize the loss function, i.e., to reduce the difference between similar faces and increase the difference between dissimilar faces. During the training phase, every input consists of 3 images of faces. Two images are of the same person (one image is considered as anchor and the second is a similar image), and the third is of a different person (dissimilar).

The UFace_Loss (using both k similar and k dissimilar images) loss is computed as

$$\sum_{i=1}^N \sum_{j=1}^{f(j)} (d(f(x_i^a) - f(x_j^s)) - d(f(x_i^a) - f(x_j^d))) + \alpha \quad (3.5)$$

where $f(x)$ takes x as an input and returns an embedding vector, i denotes the i th input, j denotes the j th similar and dissimilar images for the i th input image, a is an anchor image, s is a similar image, d is a dissimilar image, N is the number of training data and $f(j)$ is the function that represents the variable number of k most similar and k most dissimilar images for the input image x_i . The α is a margin that is enforced between positive and negative pairs. It ensures that the model does not make the embeddings equal each other to trivially satisfy the above inequality.

Minimizing the above equation means minimizing the first term (distance between anchor and similar image) and maximizing the second

term (distance between anchor and dissimilar image).

As shown in Figure 3.2b, in UFace Siamese training, the network uses three branches: the anchor, k most similar faces of the anchor and k most dissimilar faces of the anchor. First, the three branches are fed into the CNN network using 112 by 112 pixel images. The CNN encodes the pixel values and provides face embedding vector. Then, the loss between the embedding of the anchor and similar and dissimilar faces is computed. By Equation (3.5), for each anchor image, the loss function is computed 2 times k, where k is the most similar and k dissimilar images with the anchor.

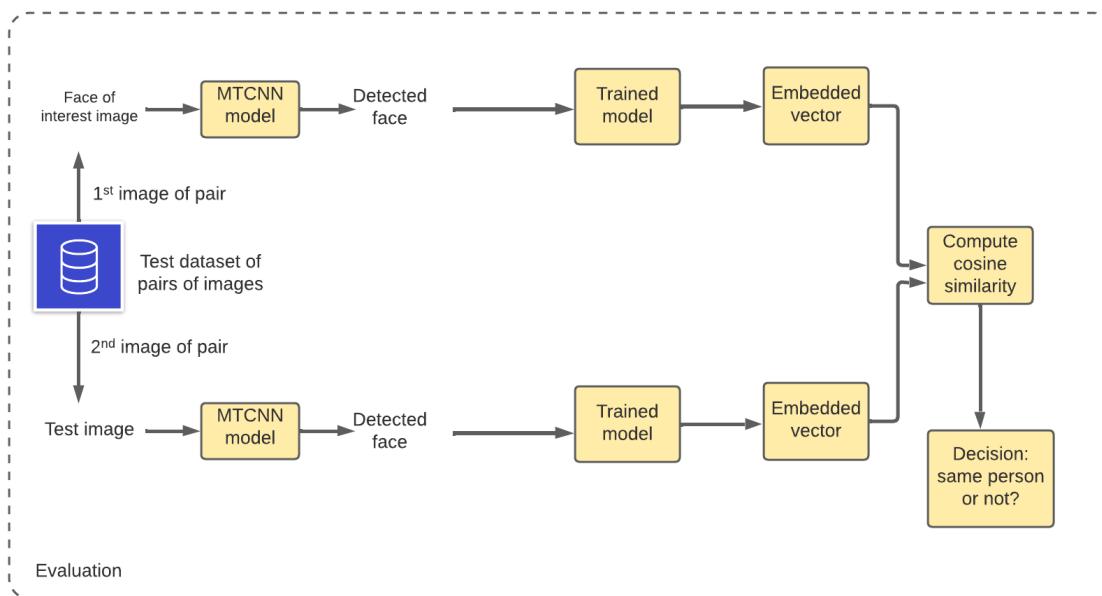


Figure 3.4: Architecture used for the proposed method evaluation.

3.4 Evaluation Method

As shown in Figure 3.4, the goal of face image verification is to decide if two face images belong to the same person or not. Given a pair of input face images, we first use MTCNN to detect faces from the given images. Then, image embeddings are extracted using any

encoder branch of the network for the pairs of test images. Cosine similarity is computed between the two embedding vectors. If the cosine similarity is above the given threshold value, the two images belong to the same person, and not otherwise.

3.5 Datasets

UFace was trained on the CelebA dataset and its performance was tested on four benchmark datasets: LFW, YTF, CALFW and CFP-FP.

CelebA [136] is a dataset that has over 200K images of 10,177 celebrities, which include pose variations and background clutter; it was used for training UFace.

The Labeled Faces in the Wild dataset (LFW) [112] contains 13,233 images of 5,749 people. For testing, the database is randomly (uniformly) split into 10 subsets. Next, 300 matched (of the same person) pairs and 300 mismatched (of different persons) pairs are randomly chosen within each subset. In other words, for testing, 3000 (10×300) matched and 3000 mismatched pairs [112] were used.

The YouTube Faces dataset (YTF) [113] of face videos contains 3425 videos of 1595 people collected from YouTube, with an average of two videos per person. The shortest clip duration is 48 frames and the longest is 6070 frames. The average length of a video clip is 181 frames. For testing, 5K video pairs are randomly chosen and prepared, half of which are pairs of videos of the same person and half are of different people. Thus, for testing, 5K pairs of static images with 2500 of them of the same person and 2500 not of the same person [113] were used.

Cross-age LFW (CALFW) [137] is a newer version of LFW in which 3000 similar face pairs at different ages and 3000 dissimilar face pairs of the same gender are present to reduce the influence of attribute differences between similar/dissimilar pairs. Thus, for testing, 6K pairs of face images were used.

Celebrities in Frontal Profile in the Wild (CFP-FP) [138] is another face verification benchmark dataset with 7000 face images, of which 3500 are same person pairs and 3500 are different person pairs. Thus, for testing, 7K pairs of face images were used.

3.6 Experimental Setup

The Keras deep learning library [139] was used to train the model. It is trained for 100 epochs or until the error is not decreasing, using a batch size of 100 images. It uses backpropagation with stochastic gradient descent (SGD), momentum of 0.91, weight decay of 0.00001 and a logarithmically decaying learning rate from 10^{-2} to 10^{-8} . The dimension of the input images is 112 by 112 pixels.

In order to select the best threshold value, which is used to select the number of similar and dissimilar images for each image, we selected about 10% of the images from the training set and selected the similar and dissimilar images using different threshold values (i.e., from 0.1 to 0.7). We used about 10% of the images as a validation set to tune the threshold value. Thus, all cosine distance scores less than the threshold values were considered as dissimilar images and all cosine distance scores greater than the threshold values were considered as similar images. For example, if we take the threshold values of 0.6 and 0.2, all cosine distances less than 0.2 are considered as dissimilar

and all cosine distance scores above 0.6 are considered as similar images.

The reason for selecting two different threshold values is to choose similar and dissimilar images correctly. The threshold values were optimized experimentally by changing their values from 0.1 to 0.9 and choosing the ones that resulted in the highest accuracy on the validation dataset; the threshold 0.6 was chosen for the similar images and threshold 0.2 for the dissimilar images (to a given image).

After computing the most similar and dissimilar images for each threshold value, we have trained different models (i.e., one model for each threshold value). After training the model, we computed the accuracy of each model on other 1K datasets that were selected from the validation set.

Using threshold values of 0.6 and 0.2 gives us the highest accuracy. Thus, we selected 0.6 and 0.2 as threshold values for similar and dissimilar images, respectively, and selected the most similar/dissimilar images on the remaining 180K training images. Note that we used two threshold values, one to select the similar images and the other to select the dissimilar images; thus, we can reduce the possibility of dissimilar images being selected as similar images and vice versa. A total 10% of the training dataset was used for validation in order to select the best threshold values.

The training was performed using the CelebA [136] dataset. First, the face is detected, including the bounding box around the face. Then, the cosine similarity for each face against the remaining faces in the training dataset is computed. Then, the threshold values are chosen experimentally to select the k most similar and k most dissimilar images for each image.

The autoencoder is a fully connected feed-forward network consisting of 3 hidden layers. As shown in Figure 3.2a, the encoder and decoder are symmetrical. The encoder input and decoder output each have 112 by 112 neurons. The second layer in both the encoder and decoder has 800 neurons. The output of the encoder has 300 neurons, which determines the size of the embedding vector.

The Siamese network has 3 branches, each of which is the CNN encoder followed by the L2-normalization layer. The CNN encoder block is a Resnet100 architecture [129]. It consists of five main layers where each layer contains convolutional and identity blocks. The first layer contains max-pooling, and the last layer contains average pooling. The five layers are followed by two fully connected layers of 800 and 300 neurons, respectively. The CNN encoder encodes the input images (112 by 112) into a 300-dimensional image embedding vector. Note that in addition to convolutional, identity and max-pooling layers, it also uses batch normalization [140] and dropout [84].

3.7 Experimental Results of the Proposed Autoencoder vs Classical Autoencoder

In Table 3.1, * represents the classical autoencoder, ** represents the modified autoencoder with k most similar images, and *** represents the modified autoencoder with both k most similar and k most dissimilar images.

As it is shown in Table 3.1, UFace using autoencoder provides better results than the one based on classical autoencoder training. Note that we use classical autoencoder training as the baseline system.

Table 3.1 shows that the baseline accuracies are 92.76%, 89.97%, 89.22% and 91.88% on LFW, YTF, CALFW and CFP-FP datasets, respectively. It is compared with two UFace models: UFace autoencoder training method using only the k most similar images and UFace autoencoder training using both the k most similar and k most dissimilar images.

From Table 3.1, we see that UFace using autoencoder that uses only the k most similar images results in 95.81%, 93.24%, 92.63% and 95.13% accuracies on the LFW, TYF, CALFW and CFP-FP datasets, respectively. The improvements over the classical autoencoder represent a 3.05%, 3.24%, 3.41% and 3.25% improvement on the LFW, YTF, CALFW and CFP-FP datasets, respectively.

Table 3.1: Accuracy of the classical and the two proposed autoencoder training method.

Model	LFW	YTF	CALFW	CFP-FP
UFace(*)	92.76	89.97	89.22	91.88
UFace(**)	95.81	93.24	92.63	95.13
UFace(***)	96.42	93.92	93.08	95.78

Next, we assess the impact of using also the k most dissimilar images. Table 3.1 shows that using both the k most similar and k most dissimilar images results in 96.42%, 93.92%, 93.08% and 95.78% accuracies on the LFW, YTF, CALFW and CFP-FP datasets, respectively. Thus, using dissimilar images, in addition to the similar images (i.e., 96.42% vs. 95.81% on LFW, 93.92% vs. 93.24% on YTF, 93.08% vs. 92.63% on CALFW and 95.78% vs. 95.13% on CFP-FP). If we compare the UFace autoencoder method that uses both the similar and dissimilar images with the classical autoencoder training method, it provides us 3.66%, 3.95%, 3.86% and 3.9% improvement on LFW, YTF, CALFW and CFP-FP datasets, respectively. Thus, the results reported in Table 3.1 show the advantage of UFace demonstrated on

an autoencoder network that uses both the k most similar and k most dissimilar images.

3.8 Experimental Results of the Proposed Siamese Network

In addition to demonstrating UFace training using the autoencoder network, we also demonstrated UFace training using the Siamese network and compared the performance of the UFace with different state-of-the-art face verification systems.

3.8.1 Labeled Faces in the Wild dataset (LFW) Dataset

Table 3.2 shows a comparison of UFace with the state-of-the-art methods. Note that we compare our best result with the state-of-the-art systems that use both supervised and unsupervised training, whereas the UFace training does not explicitly require labeled data.

Table 3.2: Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on LFW dataset.

Model	Training data size	Labeled/Unlabeled	Accuracy (%)
Fusion	500M	Labeled	98.37 [9]
Facenet	200M	Labeled	99.63 [6]
UniformFace	6.1M	Labeled	99.80 [141]
ArcFace	5.8M	Labeled	99.82 [13]
GroupFace	5.8M	labeled	99.85 [14]
CosFace	5M	Labeled	99.73 [15]
DeepFace-ensemble	4.4M	Labeled	97.35 [7]
Marginal Loss	4M	Labeled	99.48 [10]
CurricularFace	3.8M	Labeled	99.80 [16]
RegularFace	3.1M	Labeled	99.61 [142]
AFRN	3.1M	Labeled	99.85 [143]
VGG Face	2.6M	Labeled	98.95 [8]
Stream Loss	1.5M	Labeled	98.97 [144]
COCO	-	Labeled	99.78 [145]
UFace	200K	Unlabeled	99.40

Although most of the methods such as ArcFace, GroupFace, Marginal Loss and CosFace have slightly better accuracy than UFace, UFace is trained on a much smaller dataset (about 200K images) while most of the state-of-the-art methods use millions of training images.

UFace with Siamese network achieves an accuracy of 99.40%, which is on par both with the state-of-the-art supervised and unsupervised systems. For example, the ArcFace used 5.8M labeled images to achieve 99.82% accuracy, whereas UFace accuracy is 99.40% but required only about 200 K images for training.

3.8.2 YouTube Faces dataset (YTF) Dataset

Similarly, we compare the UFace with Siamese network using similar and dissimilar images with state-of-the-art supervised and unsupervised systems on the YTF dataset. In Table 3.3, VGG Face [8] used 2.6 M labeled training data and achieved slightly over 97% accuracy. In [10], the authors used marginal loss and a labeled 4M training dataset to achieve a comparable result with Facenet [6], which used 200M labeled training data and achieved over 95% accuracy. The drawback of these methods, however, is that they require a huge, labeled dataset for training. On the other hand, UFace uses much less and unlabeled training data to achieve over 96% accuracy. Although, if we compare the UFace Siamese with both the state-of-the-art supervised and unsupervised systems on YTF, its accuracy (i.e., 96.04%) is slightly better than some of the supervised systems, better than the unsupervised systems and almost close to state-of-the-art methods such as ArcFace, GroupFace, CostFace and VGG Face.

Table 3.3: Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on YTF test dataset.

Model	Training data size	Labeled/Unlabeled	Accuracy (%)
Facenet	200M	Labeled	95.12 [6]
UniformFace	6.1M	Labeled	97.70 [141]
ArcFace	5.8M	labeled	98.02 [13]
GroupFace	5.8M	labeled	97.80 [14]
CosFace	5M	labeled	97.60 [15]
DeepFace-single	4.4M	labeled	91.40 [7]
Marginal Loss	4M	labeled	95.98 [10]
RegularFace	3.1M	Labeled	96.70 [142]
AFRN	3.1M	Labeled	97.70 [143]
NAN	3M	labeled	95.70 [146]
VGG Face	2.6M	labeled	97.30 [8]
Stream Loss	1.5M	labeled	96.40 [144]
UFace	200K	unlabeled	96.04

3.8.3 Cross-age LFW (CALFW) and Celebrities in Frontal Profile in the Wild (CFP-FP) Datasets

Table 3.4: Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on CALFW test dataset.

Model	Training data size	Labeled/Unlabeled	Accuracy (%)
ArcFace	5.8M	Labeled	95.45 [13]
GroupFace	5.8M	labeled	96.20 [14]
CurricularFace	3.8M	labeled	96.20 [16]
MegaFace	3.8M	labeled	96.15 [114]
UFace	200K	unlabeled	95.12

In addition to LFW and YTF, the results of UFace have been compared against both state-of-the-art supervised and unsupervised systems on the CALFW and CFP-FP datasets. Table 3.4 and 3.5 show that UFace’s results are close to those of ArcFace. However, the results of the UFace are a bit lower than the GroupFace, CurriculaFace and MegaFace models. If we compare our best results with both supervised and unsupervised ones, Table 3.5 shows that our results are on par with the state-of-the-art unsupervised systems.

The UFace has the following advantages over the state-of-the-art systems.

- Firstly, while the UFace does not explicitly require labeled training data, the state-of-the-art methods do.
- Secondly, the UFace requires only about 200K training data, whereas the state-of-the-art use a minimum of 3.8M and maximum of 5.8M.
- Thirdly, the training time of UFace is much less than that of the state-of-the-art ones because of the amount of training data.
- Lastly, the results of UFace are comparable to the state-of-the-art methods.

Table 3.5: Comparison of the proposed Siamese network (UFace) results with the state-of-the-art methods on CFP-FP dataset.

Model	Training data size	Labeled/Unlabeled	Accuracy (%)
ArcFace	5.8M	Labeled	98.27 [13]
GroupFace	5.8M	labeled	98.63 [14]
CurricularFace	3.8M	labeled	98.37 [16]
Dyn-ArcFace	5.8M	labeled	94.25 [147]
MegaFace	3.8M	labeled	98.46 [114]
CircleLoss	5.8M	labeled	96.02 [148]
UFace	200K	unlabeled	97.89

3.9 Summary

The state-of-the-art deep learning methods for face verification usually require large amounts of labeled data for training. However, it is not always easy to obtain such data. To address this problem, we propose a novel unsupervised deep learning face verification system (UFace) that uses k most similar and k most dissimilar images to a given image that are selected from unlabeled data.

UFace’s performance was evaluated using both the autoencoder approach and Siamese networks approach. As Siamese networks performed much better than the autoencoder, they were used for all the

presented comparisons with state-of-the-art algorithms. Unlike in the classical neural network training, UFace computes its loss function k times with the similar images and k times with the dissimilar images (for a total of $2 \times k$ times) for each input image. UFace is evaluated on four benchmark face verification datasets, namely, Labeled Faces in the Wild (LFW), YouTube Faces (YTF), Cross-age LFW (CALFW) and Celebrities in Frontal Profile in the Wild (CFP-FP). Its performance using the Siamese network achieved accuracies of 99.40%, 96.04%, 95.12% and 97.89%, respectively, which are comparable with the state-of-the-art methods even though UFace uses much less data for training.

Additional advantage of UFace is that it can be used for verification of other types of images in domains where labeled data are not available at all.

Chapter 4

Face Anti-Spoofing System Using Image Quality Features and Deep Learning Approach

Face recognition is one of the most widely used biometric authentication methods but the vulnerability to spoofing attacks limits its usability and confidence [102], [103], [149].

Face recognition is used in a range of applications which require robustness to changes in the environment and resilience to circumvention, which is known as spoofing. Spoofing is defined as an attack where a fraudster tries to gain access to the system by masquerading as a valid user/employee [150]. Its goal is to fool biometric measures by presenting to the sensor (most often a camera) a manufactured artifact, such as a photograph or video to impersonate a valid user. Since such attacks are very frequent, they became a major concern for the designers and users of face recognition systems. As a consequence, spoofing is an active field of research as measured by a multitude of publications [151–155]; dissertations [156–159]; books

[63, 160–162] and standards [163]. There are also international competitions that seek to evaluate performance of the developed countermeasures [164–166]

Most of the early works on face anti-spoof detection methods focus on liveness detection. The authors in [167] introduced a liveness detection method using an eye blinking-based liveness detection method. Whereas the authors in [168] proposed face spoofing detection using mouth localization and motion analysis technique. The authors in [169] proposed a liveness detection method using an optical flow field which is generated by movements of two-dimensional planes and three-dimensional objects. Based on intrinsic biological differences between genuine and spoof traits, different color changes of face videos due to the thermogram or the facial blood flow [170] features are derived for spoofing attack detection.

More recently, Convolutional neural networks (CNN) that are able to automatically find the best features present in the images (for labeled data) were successfully used for detecting spoofing face images, as well as for fingerprint, and iris [171–173]. The authors in [27] were the first to employ CNN for spoofing attack detection. In [174], a semi-supervised learning was used for detecting a spoofing attack using a few labeled data points. Spatiotemporal anti-spoof network (STASN) was proposed in [175] to detect spoofing attacks. In [176] the authors used a bipartite auxiliary supervision network (BASN) for detecting spoofing attacks. In [177] an approach called bi-directional feature pyramid network was proposed for detecting spoofing attacks. In [178] authors proposed a method based on stimulating eye movements using visual stimuli with randomized trajectories. In [179] a head-detection algorithm and deep neural

network were used for detecting a spoofing attack. In [180] a hybrid unsupervised and semi-supervised domain adaptation network for cross-scenario face spoofing attack was used. [91] introduced a CNN based framework with a densely connected network trained using both binary and pixelwise binary supervision (DeepPixBiS) for detecting spoofing attacks. [181] proposed a method for face anti-spoofing that estimates depth information from multiple RGB frames and proposed a supervised method to efficiently encode spatiotemporal information in a spoofing attack. It included two modules: optical flow-guided feature block and convolutional-gated recurrent unit modules, designed to extract short-term and long-term motion to discriminate between living and spoofing faces. [175] proposed a face anti-spoofing model with a spatiotemporal attention mechanism fusing global temporal and local spatial information. [182] proposed Bilateral Convolutional Networks (BCN) that was able to capture intrinsic material-based patterns via aggregating multi-level bilateral macro- and micro- information. [183] proposed a patchwise motion parameterization method, which explores the underlying motion difference between the facial movements re-captured from a planar screen and those from a real face. The authors in [182] were inspired by human material perception to design a novel network for learning intrinsic material-based patterns for attack detection. The authors in [184] used meta-pattern learning, instead of just using manually extracted features, to create a hybrid model to address spoofing attack detection. The authors in [185] proposed an anti-spoofing method based on one-class multiple kernel learning. The authors in [63] described methods to detect 3D facial mask attack. The authors in [186] proposed a face anti-spoofing method

using an ensemble of vision transformer features, where an ensemble of local features are extracted from the intermediate blocks of a vision transformer. The authors in [187] proposed local binary pattern and convolutional neural network based feature fusion model for detecting spoof face attacks. The authors in [188] proposed an anti-spoofing method based on fusing an optical flow and texture features. The authors in [189] explored face spoofing attacks based on light detection and ranging sensors against light variation.

Even when there are identity verification mechanisms in place, fraudsters always find a way to get around them. One such method is face spoofing, in which a fraudster attempts to deceive a facial recognition system by displaying a spoof face to the camera.

The most popular means of spoofing is to put on a valid user's mask and present it to the biometric verification system, which is referred to as a mask attack. Another method is to get hold of and print a photo of a user and present it to the camera, which is known as print attack. Another type of spoofing is a replay-attack, when the system is presented with the screen of a device on which a recorded video of a valid user is played.

One approach to detect a spoofing attack is to analyze the presented spoofing image and identify in it the key features, called image quality (IQ) features, and use them to determine whether the presented image is genuine or spoofed; as there is a voice quality measurements [190]. [69] used 25 such IQ features to distinguish between genuine and spoofed images of a user. In [191] 18 IQ features were used for detecting a spoofing attack and achieved better performance than the one reported in [69]. In [28] an image distortion method (IDA) was used for detecting a spoofing attack, which is based on four face

IQ features, namely, blurriness, color diversity, specular reflection, and chromatic moments.

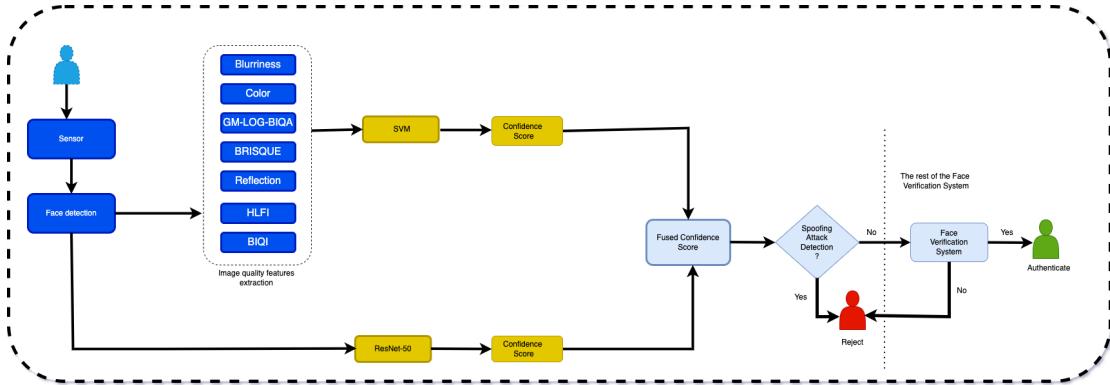


Figure 4.1: The proposed FASS system.

A very different approach to detect spoofing attack is to use deep learning on the presented images instead of manually extracted image quality features and then using some classifier, like those used in [28, 69, 191].

Indeed, recent studies have revealed that the performance of the state-of-the-art face anti-spoofing methods degrades under the real-world variations (e.g., illumination and camera device variations) [63, 192–195], which indicates that more robust face anti-spoofing methods are needed to reach the deployment levels of the face biometric systems.

Face anti-spoofing methods that utilize hand-crafted image features are using standard machine learning classifiers such as Random Forest (RF) or SVM to determine whether the detected facial image is genuine or spoofed. On the other hand, deep learning methods detect the spoofing attack by self-learning the global key features. In this PhD dissertation, we take advantage of the two approaches, and proposed concatenating both at feature and score level.

In this dissertation, the first proposal is called Face Anti-Spoofing System Using Image Quality Features and Deep Learning (FASS), that combines a spoofing detection method based on a small number of image quality features with a spoofing detection method based on deep learning at the confidence score level.

The second proposal is called Hybrid Face Anti-Spoofing Method Concatenating Deep Learning and Hand-Crafted Features (HDLHC), which concatenates deep features which are extracted through several layers (before the classification layer) with the hand-crafted image quality features to improve detection accuracy (whether it is a genuine or spoofed image).

Since deep learning based and manual extracted based features complement each other, their combination will be able to better generalize and improve the spoofing attack detection.

4.1 The Proposed Method

Two approaches have been proposed namely FASS (Figure 4.1) and HDLHC (Figure 4.5). In the first approach (FASS), the scores obtained from the deep learning classifier is fused in a weighted manner with the scores obtained from the classifier based on image quality features. In the second approach (HDLHC), the features obtained using the deep learning method (VGG) is concatenated with the hand-crafted image quality features. Then, the concatenated feature vector fed to the classifier.

4.1.1 The Proposed Image Quality Feature Measurements

There are two main methods for assessing the quality of a presented image features. One uses the so-called full-reference (FR) and the other uses No-Reference (NR). FR method requires access to the genuine, called reference, image of a valid user and also access to the presented, possibly spoofed, image. Thus, it compares the genuine image with the presented (spoofed) image. In contrast, the NR method is based on using only the presented images. In this work, we only focus on selecting NR image quality features as quite often the reference images are not available.

The authors in [69] proposed a binary classification system to detect spoofing attacks for three biometric modalities (Iris, Fingerprint, and Face), using 25 IQ features. Among them only BIQI, NIQE, JQI and HLFI features are the NR features. In [191] the authors used 18 IQ features for face anti-spoofing, with HLFI being the only NR feature. [28] proposed a face spoof detection method using four quality features, all of them were NR features.

Table 4.1: List of the twelve No-Reference (NR) Image Quality (IQ) feature measurements.

NR IQ Features Name	Reference
Blind Image Quality Index (BIQI)	[196]
Naturalness Image Quality Estimator (NIQE)	[197]
High-Low Frequency Index (HLFI)	[198]
Reflection	[72]
Blurriness	[73, 74]
Chromatic Moment	[75]
Color	[75, 199]
Blind/Referenceless Image Spatial QUality Evaluator (BRISQUE)	[200]
Gradient-Magnitude map and Laplacian-of-Gaussian based Blind Image Quality Assessment (GM-LOG-BIQA)	[201]
HDR Image GRADient based Evaluator - 1 (HIGRADE-1)	[202]
Robust BRISQUE index (Robustbrisque)	[197]
Distortion Identification-based Image Verity and INtegrity Evaluation (DIIVINE)	[203]

In Table 4.1, we have selected and listed 12 NR quality features. To check if these 12 features can be further be reduced, we use the min-Redundancy max-Relevance (mRmR) [204] measure, see Equation 4.1. It uses mutual information to define relevance and redundancy of features as it seeks to find a set of features that jointly have the maximal statistical dependency on the classification label and minimum redundancy with respect to the selected features. Equation 4.1 shows how its values are computed for each feature. The best feature is the one with the highest score, second best with the second highest score, etc. The output is a vector of scores for all 12 features. Importantly, we need to take into account that mRmR scoring heavily depends on the data used. Thus, to get a more reliable assessment of the goodness of the features we decided to calculate the mRmR on three datasets, namely, Reply-Attack, CASIA-MFSD and MSU-MFSD to determine the overall importance of features for detecting a spoofing attack.

Next, for the same datasets, in order to determine their ordered combinations (i.e., the first best feature, the first two best together, the first three best together, etc.) we use a measure called ACER, defined in Equation 4.4.

$$score_i(f) = \frac{relevance(f|target)}{redundancy(f|features selected until i - 1)} \quad (4.1)$$

$$APCER = \frac{FP}{(TN + FP)} \quad (4.2)$$

$$BPCER = \frac{FN}{(FN + TP)} \quad (4.3)$$

where FP is false positive, TN is true negative, TP is true positive, and FN is false negative.

$$ACER = \frac{(APCER + BPCER)}{2} \quad (4.4)$$

The results of using mRmR and ACER measures

For the Replay-Attack dataset, the order of best features, according to mRmR, is: Blurriness, Color, GM-LOG-BIQA, BRISQUE, Reflection, HLF1, BIQI, Robustbrisque, Chromatic Moment, DIIVINE, HIGRADE-1, and NIQE, which is shown in Figure 4.2.

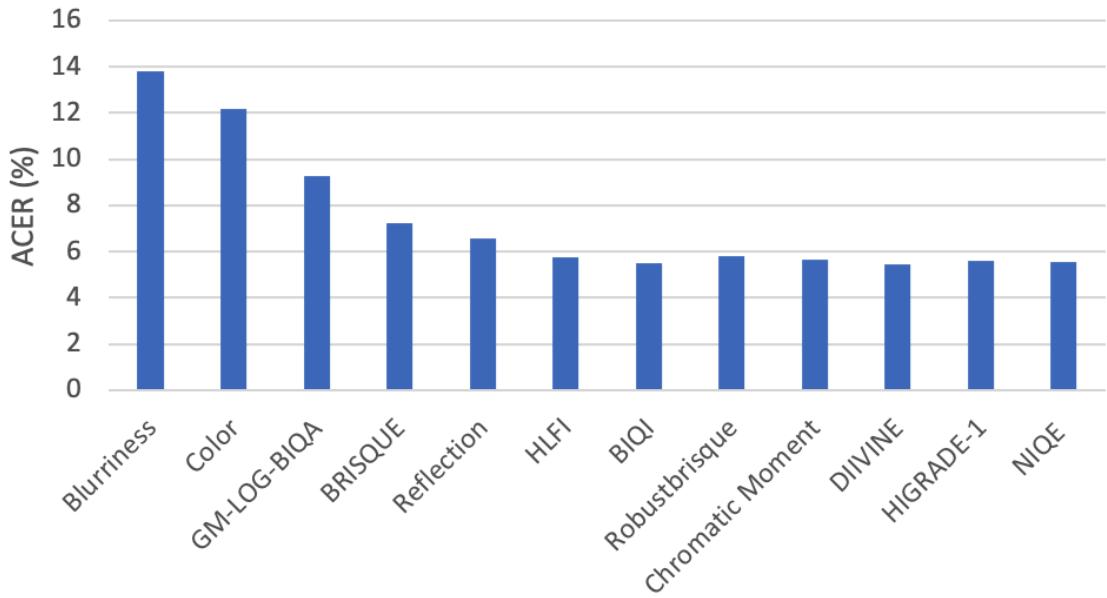


Figure 4.2: ACER value as it changes with adding additional features for the Replay-Attack dataset.

Notice that after the seventh feature, the error rate goes slightly up before slightly going down when 10 features are used. We thus choose the first seven features, namely, Blurriness, Color, GM-LOG-BIQA, BRISQUE, Reflection, HLF1 and BIQI.

For the CASIA-MFSD dataset, the mRmR order of feature is: Blurriness, Color, HLF1, BRISQUE, GM-LOG-BIQA, Reflection, BIQI,

Chromatic Moment, Robustbrisque, HIGRADE-1, NIQE and DIVINE, shown in Figure 4.3.

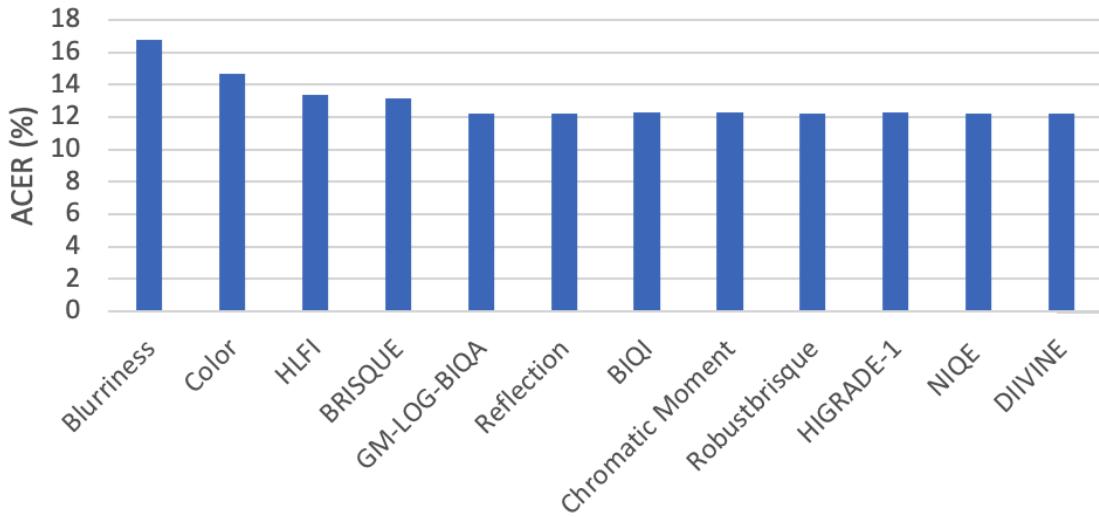


Figure 4.3: ACER value as it changes with adding additional features for the CASIA-MFSD dataset.

We see that after the first 5 features are combined, namely, Blurriness, Color, HLF1, BRISQUE, and GM-LOG-BIQA, ACER value remains the same, thus we chose these five features.

For the MSU-MFSD dataset, the mRmR order of features is: Blurriness, Color, BRISQUE, GM-LOG-BIQA, BIQI, Reflection, HLF1, Chromatic Moment, HIGRADE-1, Robustbrisque, NIQE and DIVINE, shown in Figure 4.4.

We see that the ACER remains about the same after using the first six features: Blurriness, Color, BRISQUE, GM-LOG-BIQA, BIQI and Reflection.

The combined list of best features from the above experiments is Blurriness, Color, GM-LOG-BIQA, BRISQUE, Reflection, HLF1, and BIQI. These seven features are described below for the convenience of the reader.

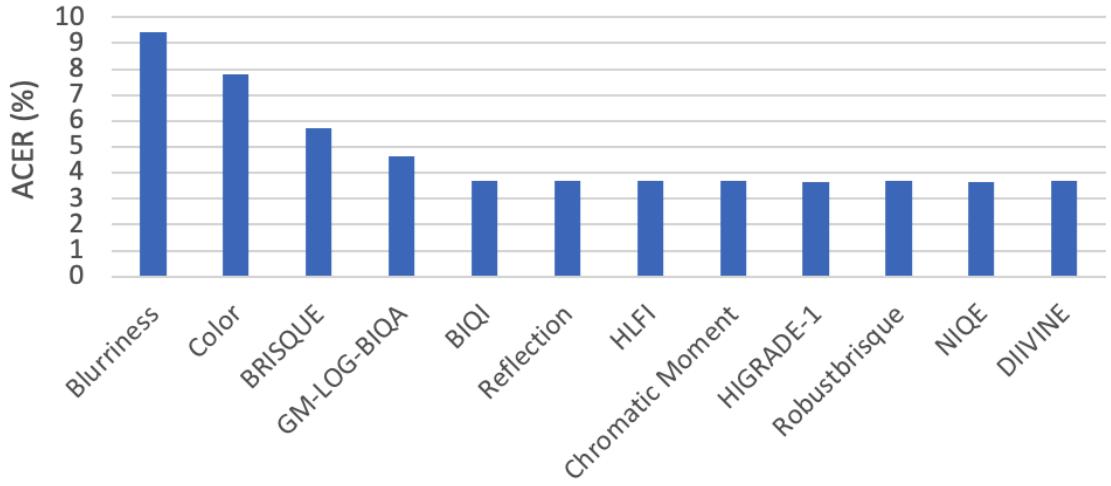


Figure 4.4: ACER value as it changes with adding additional features for the MSU-MFSD dataset.

Blurriness, for short distance spoof attacks, spoof faces are often defocused in mobile phone cameras. The reason is that the spoofing medium (printed paper and screen) usually is of limited size, and the attacker must place them close to the camera to obscure the boundaries of the attack medium. As a result, spoof faces are defocused, and the resulting image blur can be used as indication for anti-spoofing [73, 74].

Color is an important difference between genuine and spoof faces is the color diversity, as genuine faces have richer colors. This diversity fades out in spoof faces due to the color reproduction loss during image/video recapture [75].

GM-LOG-BIQA defines local spatial contrast features that characterize various perceptual image structures related to luminance discontinuities. The Gradient Magnitude (GM) captures the local changes of luminance, while the Laplacian of Gaussian (LOG) is sensitive to local intensity contrast and BIQA is blind image quality assessment which means it doesn't require a reference image to measure the quality of the image [201].

BRISQUE is a Blind/Referenceless Image Spatial Quality Estimator. Its features are derived from the empirical distribution of locally normalized luminance values and their products under a spatial natural scene statistic and they follow a Gaussian-like distribution. These features are then used in support vector regression to map image features to an image quality score [197].

Reflection degrade the quality of face images / videos by obstructing the background scenes. The existence of a reflection component in an image will not only change the color of the object surface but also destroy its edge contour, but the saturated reflection will also lead to the complete loss of image texture information, which provides a good clue for anti-spoofing tasks [205].

HLFI is a High-Low Frequency Index, which uses local gradients as a blind metric to detect blur and noise. It is sensitive to the sharpness of the image, which is done by computing the difference between the power in the lower and upper frequencies of the Fourier Spectrum [198].

BIQI is Blind Image Quality Indices which is a two-step no-reference image quality measurement. Given a distorted image, the first step performs the wavelet transform and extracts features for estimation of the presence of image distortions, and it evaluates the quality of the image across these distortions by applying support vector regression on the wavelet coefficients [196].

4.2 Score Level Fusion

The proposed approach consists of several parts and is depicted in Figure 4.1:

1. Extraction of image quality features from the face images.
2. Using these features as input to an SVM and Random Forest (RF) classifiers to determine if a face image is a genuine or a spoofed face.
3. Using ResNet50 deep neural network to do the similar classification.
4. Merging classification confidence scores of both classifiers to make a final determination whether the presented face is genuine or spoofed.
5. If it is a genuine face, it proceeds into the next part of the face verification system [206].

The FASS system (see Figure 4.1) fuses the results of the SVM and random forest (RF) classifiers (separately) that uses the selected above seven NR quality features with the result of the ResNet50 algorithm that operates directly on raw input images for detecting a face spoofing attack.

The confidence scores of two classifiers are combined in a weighted fashion according to Equation 4.5.

The fused confidence score is calculated as follows:

$$FS = (\alpha * \Theta_x) + ((1 - \alpha) * \Theta_y) \quad (4.5)$$

where FS is the fused confidence score, Θ_x is ResNet-50 confidence score, Θ_y is SVM or Random Forest (RF) confidence score and α is the weight parameter. After checking α values ranging from 0.1 to 0.9, we found that the best results were obtained for $\alpha=0.75$, which is then used in all experiments.

4.3 Feature Level Fusion

As it is shown in Figure 4.5, the proposed method is based on fusing features found by deep learning (not understandable to humans) with hand-crafted features derived from genuine face images. We use 1000 deep features that are extracted by the VGG deep learning network. These are the ones just before they are input to the classification layer and enhance them by adding seven hand-crafted multi-dimensional image quality features which we identified previously in [207]. The seven features form a 453-dimensional vector.

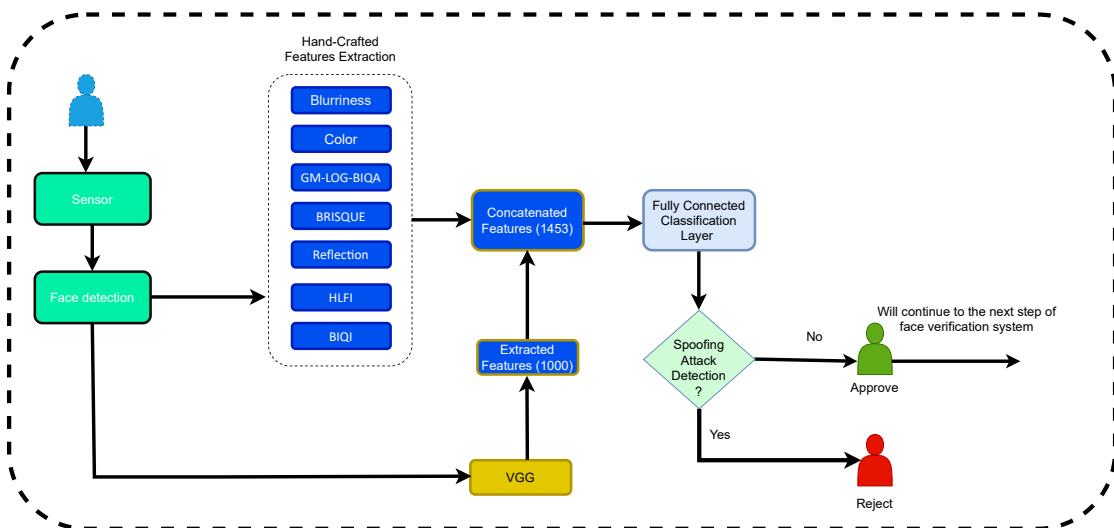


Figure 4.5: The proposed HDLHC system.

Thus, the input to the classification layer of VGG is a 1453-dimensional feature vector. This final feature vector is fed into a fully connected classification layer. During training, binary cross-entropy loss function is minimized to update the network weights. Once the network is trained, it is evaluated on two datasets, namely, Oulu-NPU [208] and SiW [90].

The deep neural network block is inspired by the VGG architecture [76]. It consists of three main blocks, where each block contains two convolutional and one max-pooling layer. The output of the

last max-pooling layer feature vector is concatenated with the seven hand-crafted features, which is followed by the fully connected classification layer.

4.4 Experimental Setup

The Pytorch library [209] was used for implementing the FASS system. All experiments were run for 100 epochs or until the validation error stopped decreasing, whichever was sooner, and using a batch size of 100. Stochastic gradient descent with momentum (0.9), weight decay ($5E - 4$) and a logarithmically decaying learning rate (initialized to 10^{-2} and decaying to 10^{-8}) were used.

Five face spoofing datasets, namely, Replay-Attack [64], CASIA-FASD [62], MSU-MFSD [28], Oulu-NPU [208] and SiW [90] were used to evaluate FASS and compare its results with the state-of-the-art results. Multi-Task Cascaded Convolutional Neural Network [36] was used to detect faces from the video frames. The face images of all five datasets were resized to the size 224×224 pixels for computational efficiency. We used data partitions into the train-validate-test as detailed in the data descriptions below.

4.5 Datasets

Five face anti-spoofing benchmark datasets, namely, Replay-Attack [64], CASIA-FASD [62], MSU-MFSD [28], Oulu-NPU [208] and SiW [90], were used.

Replay-Attack dataset consists of 1200 video clips of photo and video spoof attempts of 50 users, under different lighting conditions.

Training set has 60 genuine and 300 spoof users. Validation set has 60 genuine and 300 spoof images. Test set has 80 genuine and 400 spoof images.

CASIA-MFSD contains 50 users video clips under different resolutions and light conditions. Three spoof face attacks are implemented, which include warped photo attack, cut photo attack and video attack. The dataset contains 600 video clips, in which 120 videos for training, 120 videos for validation and 360 videos for testing are used.

MSU-MFSD dataset has 280 video clips of genuine and spoof faces from 35 users. Two cameras with different resolutions (720×480 and 640×480) were used to record the videos from the 35 users. The 280 videos were divided into training (60 videos), validation (60 videos) and testing (160 videos) datasets, respectively.

Oulu-NPU dataset consists of 4950 video clips and has four testing protocols: Protocol 1 evaluates the effect of the illumination variations; Protocol 2 evaluates the effect of spoofing attack instrument variations; Protocol 3 evaluates the effect of camera device variations; and Protocol 4 is a combination of the three protocols. For all protocols, the 4950 video clips were divided into three disjoint subsets for training, validation and testing, namely, 1800, 1350 and 1800, respectively.

SiW dataset has genuine and spoof videos from 165 users. It has three protocols. The first protocol evaluates the generalization of the face attack detection under different face poses and expressions. The second protocol evaluates the generalization capability on cross-medium of the same spoof type. The third protocol evaluates the performance on an unknown attack. We used 45, 45 and 75 users for training, validation and testing, respectively.

4.6 Evaluation metrics

Performance of biometric verification systems depends on accuracy of acceptance/rejection of the analyzed image [210, 211]. The measures used are false acceptance rate (FAR), Equation 4.6, and false rejection rate (FRR), Equation 4.7. FAR is the ratio of incorrectly accepted spoofing attack faces, whereas FRR is the ratio of incorrectly rejected genuine faces. The commonly used metric in anti-spoofing literature is Half Total Error Rate (HTER), Equation 4.8, while Equal Error Rate (EER) is a value of HTER at which FAR and FRR have the same values.

The other metrics used in ISO standard [212] are Attack Presentation Classification Error Rate (APCER), Equation 4.2, Bona fide Presentation Classification Error Rate (BPCER), Equation 4.3 and Average Classification Error Rate (ACER) Equation 4.4. BPCER (Equation 4.3) and APCER (Equation 4.2) measure genuine and spoof classification error rates, respectively. ACER (Equation 4.4) summarizes the two measures.

$$FAR = \frac{FP}{\text{Spoof Samples}} \quad (4.6)$$

$$FRR = \frac{FN}{\text{Genuine Samples}} \quad (4.7)$$

$$HTER = \frac{(FRR + FAR)}{2} \quad (4.8)$$

where FP is false positive, and FN is false negative.

Table 4.2: Comparison of the proposed Image Quality (IQ) feature measurements with other image quality feature measurement based methods on Replay-Attack, CASIA-MFSD and MSU-MFSD datasets.

Methods	No. of IQ Features	HTER (%) on Replay-Attack	EER (%) on CASIA-MFSD	EER (%) on MSU-MFSD
IDA [28]	4	7.41	13.3	8.58
Galbally [69]	24	15.2	-	-
Costa-Pazo [191]	18	5.28	-	-
FASS with RF	7	4.32	7.02	6.51
FASS with SVM	7	5.02	7.17	6.48

4.7 Experimental Results of the Proposed Method (FASS)

4.7.1 The Proposed Image Quality Feature Measurements on Replay-Attack, CASIA- MFSD and MSU-MFSD Datasets

Table 4.2 compares FASS results with other algorithms that use different numbers of image quality features, namely, 4, 25 and 18 features.

As it is shown in the Table, we compare our two proposed methods (i.e., FASS with Random Forest (RF) and FASS with SVM) with the other three algorithms in order to see the classification accuracy of RF and SVM methods.

We notice in Table 4.2 that FASS performs better results than the other systems on three datasets. FASS with RF and FASS with SVM result in 18.18% and 4.9% HTER relative improvement when compared with [191], respectively. Comparison with the CASA-MFSD dataset, FASS with RF and FASS with SVM provide 47.2% and 46% relative EER improvement over [28], respectively. Compared with [28], FASS with RF and FASS with SVM gave 24.1% and 24.5% relative EER improvement on the MSU-MFSD dataset, respectively.

These results show that the selected seven no-reference image quality features are good for detecting face spoofing attacks. Both RF and SVM classification methods provide more or less similar results. While RF classification has the best results on Replay-Attack and CASIA-MFSD dataset, SVM classification has the best result on MSU-MFSD.

For the Replay-Attack dataset, only HTER results are reported in the literature and, for CASIA-MFSD and MSU-MFSD datasets only EER results are reported, thus we used them in our comparisons.

4.7.2 OULU-NPU Dataset

Table 4.6 shows the results of the FASS and other state-of-the-art systems for anti-spoofing, for four different protocols on the OULU-NPU dataset.

Similar to Table 4.2, our two methods (i.e., FASS with RF and FASS with SVM) are compared with other reported results in terms of accuracy.

From Table 4.6, we see that FASS gave the best APCER (Equation 4.2) value of all anti-spoofing systems on protocol 1. Compared with DeepPixBiS, FASS shows 62.5% relative improvement. However, FASS is not as good using BPCER (Equation 4.3) and ACER (Equation 4.4) measures as DeepPixBiS.

FASS with SVM gives the best ACER (Equation 4.4) value on protocol 2. Compared with FAS-TD, FASS with SVM shows 15.8% relative ACER (Equation 4.4) improvement. It has almost the same APCER (Equation 4.2) value as FAS-TD. However, it lags the STASN on BPCER (Equation 4.3).

Table 4.3: Comparison of the proposed method with the state-of-the-art methods using four protocols on the OULU-NPU dataset.

Protocol	Method	APCER(%)	BPCER(%)	ACER(%)
1	GRADIANT [213]	1.3	12.5	6.9
	DeepPixBiS [91]	0.8	0.0	0.4
	STASN [175]	1.2	2.5	1.9
	Auxiliary [90]	1.6	1.6	1.6
	CPqD [213]	2.9	10.8	6.9
	FaceDs [89]	1.2	1.7	1.5
	MILHP [183]	8.3	0.8	4.6
	BASN [176]	1.5	5.8	3.6
	FAS-TD [181]	2.5	0.0	1.3
	FASS with RF	0.3	0.5	0.6
	FASS with SVM	0.3	1.5	0.9
	DeepPixBiS [91]	11.4	0.6	6.0
2	Auxiliary [90]	2.7	2.7	2.7
	GRADIANT [213]	3.1	1.9	2.5
	STASN [175]	4.2	0.3	2.2
	FAS-TD [181]	1.7	2.0	1.9
	FaceDs [89]	4.2	4.4	4.3
	MILHP [183]	5.6	5.3	5.4
	BASN [176]	2.4	3.1	2.7
	FASS with RF	2.1	0.7	1.7
	FASS with SVM	1.8	1.3	1.6
	DeepPixBiS [91]	11.7±19.6	10.6±14.1	11.1±9.4
	FAS-TD [181]	5.9±1.9	5.9±3.0	5.9±1.0
	GRADIANT[213]	2.6±3.9	5.0±5.3	3.8±2.4
3	FaceDs [89]	4.0±1.8	3.8±1.2	3.6±1.6
	Auxiliary [90]	2.7±1.3	3.1±1.7	2.9±1.5
	MILHP [183]	1.5±1.2	6.4±6.6	4.0±2.9
	BASN [176]	1.8±1.1	3.6±3.5	2.7±1.6
	STASN [175]	4.7±3.9	0.9±1.2	2.8±1.6
	FASS with RF	1.9±1.7	1.2±1.2	1.7±0.3
	FASS with SVM	2.0±1.4	1.8±1.3	1.9±0.6
	DeepPixBiS [91]	36.7±29.7	13.3±14.1	25.0±12.7
	GRADIANT [213]	5.0±4.5	15.0±7.1	10.0±5.0
	Auxiliary [90]	9.3±5.6	10.4±6.0	9.5±6.0
	FAS-TD [181]	14.2±8.7	4.2±3.8	9.2±3.4
	STASN [175]	6.7±10.6	8.3±8.4	7.5±4.7
	MILHP [183]	15.8±12.8	8.3±15.7	12.0±6.2
	FaceDs [89]	5.1±6.3	6.1±5.1	5.6±5.7
4	FASS with RF	4.0±3.6	5.6±3.5	5.2±1.9
	FASS with SVM	4.3±4.5	6.4±5.7	5.4±3.2

Using protocol 3, Table 4.6 shows that the FASS with RF system gives the best ACER (Equation 4.4) value, however, it does not have the best APCER (Equation 4.2) and BPCER (Equation 4.3) values.

On protocol 4, FASS with RF gives the best APCER (Equation 4.2) and ACER (Equation 4.4) values. However, FASS is not as good as the FAS-TD system using BPCER (Equation 4.3) measure.

4.7.3 SiW Dataset

Table 4.4: Comparison of the proposed method with the state-of-the-art methods using three protocols on the SiW dataset.

Protocol	Method	APCER(%)	BPCER(%)	ACER(%)
1	Auxiliary [90]	3.58	3.58	3.58
	STASN [175]	–	–	1.00
	FAS-TD [181]	0.96	0.50	0.73
	BASN [176]	–	–	0.37
	BCN [182]	0.55	0.17	0.36
	FASS with RF	0.46	0.18	0.31
	FASS with SVM	0.49	0.19	0.34
2	Auxiliary [90]	0.57±0.69	0.57±0.69	0.57±0.69
	STASN [175]	–	–	0.28±0.05
	FAS-TD [181]	0.08±0.14	0.21±0.14	0.15±0.14
	BASN [176]	–	–	0.12±0.03
	BCN [182]	0.08±0.17	0.15±0.00	0.11±0.08
	FASS with RF	0.11±0.31	0.14±0.10	0.12±0.02
	FASS with SVM	0.15±0.10	0.13±0.10	0.14±0.03
3	STASN [175]	–	–	12.10±1.50
	Auxiliary [90]	8.31±3.81	8.31±3.80	8.31±3.81
	FAS-TD [181]	3.10±0.81	3.09±0.81	3.10±0.81
	BASN [176]	–	–	6.45±1.80
	BCN [182]	2.55±0.89	2.34±0.47	2.45±0.68
	FASS with RF	2.29±0.24	2.01±0.15	2.03±0.17
	FASS with SVM	2.33±0.17	1.98±0.14	2.15±0.13

Similarly, we compared (Table 4.7) the FASS system on SiW dataset on its 3 different protocols using two classifications (i.e., FASS with RF and FASS with SVM).

We can see, FASS with RF and FASS with SVM had 18.1% 10.9% relative APCER (Equation 4.2) improvement when compared with BCN, respectively. Similarly, FASS with RF had the best ACER (Equation 4.4) result on protocol 1. FASS with SVM provided the second-best ACER (Equation 4.4) value on protocol 1.

On protocol 2, both of FASS's performance using RF and SVM classification methods was not good in terms of APCER (Equation 4.2) when compared with FAS-TD and BCN, however, FASS with SVM was the best performing and FASS with RF is the second best performing in terms of BPCER (Equation 4.3).

As it is shown in Table 4.7, FASS with RF gives us the best APCER value (i.e., 2.29%) and the best ACER value (i.e., 2.03%). Similarly, FASS with SVM had the best result on BPCER (1.98%)

Overall, FASS with RF compared with five state-of-the-art methods on this dataset performed almost the best in terms of ACER (Equation 4.4) on all three protocols (0.31%, 0.12%, and 2.03% respectively). These results show good generalization of FASS for variations of face pose and expression, and for different spoof mediums.

4.7.4 Cross-Dataset Testing between CASIA-MFSD and Replay-Attack Datasets

Table 4.5: Comparison of the proposed method with the state-of-the-art methods using cross-dataset between CASIA-MFSD and Replay-Attack.

Method	Train: CASIA-MFSD Test: Replay-Attack	Train: Replay-Attack Test: CASIA-MFSD
Motion-Mag [214]	50.1	47.0
LBP-TOP [26]	49.7	60.6
STASN [175]	31.5	30.9
Auxiliary [90]	27.6	28.4
FAS-TD [181]	17.5	24.0
LBP [199]	47.0	39.6
Spectral cubes [29]	34.4	50.0
BCN [182]	16.6	36.4
BASN [176]	23.6	29.9
FaceDs [89]	28.5	41.1
FASS with RF	9.1	24.5
FASS with SVM	9.7	25.6

Table 4.5 shows the results of testing using HTER measure for cross-dataset testing (trained on CASIA-MFSD but tested on Replay-Attack dataset), and vice versa. We see that FASS with RF gives us the best result when trained on CASIA-MFSD data and tested on Replay-Attack data. FASS with RF provides us a 45.18% HTER relative improvement compared to BCN system. However, both FASS with RF and FASS with SVM did not give the best results when trained on Replay-Attack but evaluated on CASIA-MFSD dataset. On average, however, they gave better results compared to the other state-of-art system results. The results of Table 4.5 indicate that FASS generalizes well, different from a different distribution.

While several face spoof detection techniques have been proposed, their generalization abilities are still to be improved. We propose an efficient face spoof detection system called FASS which is based on fusing the scores of the two classifiers such as SVM/RF and ResNet50.

4.8 Experimental Results of the Proposed Method (HDLHC)

4.8.1 Oulu-NPU Dataset

Table 4.6 compares the HDLHC method on the Oulu-NPU dataset with several state-of-the-art methods. In protocol 1, HDLHC method gives the best APCER and slightly worse ACER, but DeepPixBiS gives the best BPCER and ACER. In protocol 2, HDLHC method gives the best APCER and ACER whereas STASN has the best BPCER. In protocol 3, HDLHC method gives the best ACER, but worst APCER and BPCER. In protocol 4, HDLHC method gives the

Table 4.6: Comparison of the proposed method (HDLHC) with the state-of-the-art methods on Oulu-NPU dataset.

Protocol	Method	APCER(%)	BPCER(%)	ACER(%)
1	GRADIANT [213]	1.3	12.5	6.9
	DeepPixBiS [91]	0.8	0.0	0.4
	STASN [175]	1.2	2.5	1.9
	Auxiliary [90]	1.6	1.6	1.6
	FaceDs [89]	1.2	1.7	1.5
	MILHP [183]	8.3	0.8	4.6
	BASN [176]	1.5	5.8	3.6
	FAS-TD [181]	2.5	0.0	1.3
	FASS with RF [207]	0.3	0.5	0.6
	FASS with SVM [207]	0.3	1.5	0.9
2	HDLHC	0.2	0.8	0.5
	DeepPixBiS [91]	11.4	0.6	6.0
	Auxiliary [90]	2.7	2.7	2.7
	GRADIANT [213]	3.1	1.9	2.5
	STASN [175]	4.2	0.3	2.2
	FAS-TD [181]	1.7	2.0	1.9
	FaceDs [89]	4.2	4.4	4.3
	MILHP [183]	5.6	5.3	5.4
	BASN [176]	2.4	3.1	2.7
	FASS with RF [207]	2.1	0.7	1.7
3	FASS with SVM [207]	1.8	1.3	1.6
	HDLHC	1.6	0.7	1.2
	DeepPixBiS [91]	11.7±19.6	10.6±14.1	11.1±9.4
	FAS-TD [181]	5.9±1.9	5.9±3.0	5.9±1.0
	GRADIANT[213]	2.6±3.9	5.0±5.3	3.8±2.4
	FaceDs [89]	4.0±1.8	3.8±1.2	3.6±1.6
	Auxiliary [90]	2.7±1.3	3.1±1.7	2.9±1.5
	MILHP [183]	1.5±1.2	6.4±6.6	4.0±2.9
	BASN [176]	1.8±1.1	3.6±3.5	2.7±1.6
	STASN [175]	4.7±3.9	0.9±1.2	2.8±1.6
4	FASS with RF [207]	1.9±1.7	1.2±1.2	1.7±0.3
	FASS with SVM [207]	2.0±1.4	1.8±1.3	1.9±0.6
	HDLHC	1.8±2.3	1.1±3.3	1.5±1.4
	DeepPixBiS [91]	36.7±29.7	13.3±14.1	25.0±12.7
	GRADIANT [213]	5.0±4.5	15.0±7.1	10.0±5.0
	Auxiliary [90]	9.3±5.6	10.4±6.0	9.5±6.0
	FAS-TD [181]	14.2±8.7	4.2±3.8	9.2±3.4
	STASN [175]	6.7±10.6	8.3±8.4	7.5±4.7
	MILHP [183]	15.8±12.8	8.3±15.7	12.0±6.2
	FaceDs [89]	5.1±6.3	6.1±5.1	5.6±5.7

best APCER and ACER. Note that protocol 4 evaluates all the Oulu-NPU dataset variations, which is the most challenging and most similar to real application scenarios. Our method achieves 4.8% ACER, better than the state-of-the-art methods, showing that concatenating deep features with seven hand-crafted image quality features has better performance and generalization ability.

4.8.2 SiW dataset

Table 4.7: Comparison of the proposed method (HDLHC) with the state-of-the-art methods on SiW dataset.

Protocol	Method	APCER(%)	BPCER(%)	ACER(%)
1	Auxiliary [90]	3.58	3.58	3.58
	STASN [175]	—	—	1.00
	FAS-DRL [215]	0.07	0.50	0.28
	FAS-TD [181]	0.96	0.50	0.73
	BASN [176]	-	-	0.37
	BCN [182]	0.55	0.17	0.36
	FASS with RF [207]	0.46	0.18	0.32
	FASS with SVM [207]	0.49	0.19	0.34
2	HDLHC	0.47	0.17	0.32
	Auxiliary [90]	0.57±0.69	0.57±0.69	0.57±0.69
	STASN [175]	—	—	0.28±0.05
	FAS-DRL [215]	0.08±0.17	0.13±0.09	0.10±0.04
	FAS-TD [181]	0.08±0.14	0.21±0.14	0.15±0.14
	BASN [176]	-	-	0.12±0.03
	BCN [182]	0.08±0.17	0.15±0.00	0.11±0.08
	FASS with RF [207]	0.11±0.31	0.14±0.10	0.12±0.02
3	FASS with SVM [207]	0.15±0.10	0.13±0.10	0.14±0.03
	HDLHC	0.09±0.30	0.13±0.20	0.11±0.05
	STASN [175]	—	—	12.10±1.50
	FAS-DRL [215]	9.35±6.14	1.84±2.60	5.59±4.37
	Auxiliary [90]	8.31±3.81	8.31±3.80	8.31±3.81
	FAS-TD [181]	3.10±0.81	3.09±0.81	3.10±0.81
	BASN [176]	-	-	6.45±1.80
	BCN [182]	2.55±0.89	2.34±0.47	2.45±0.68
	FASS with RF [207]	2.29±0.24	2.01±0.15	2.16±0.17
	FASS with SVM [207]	2.33±0.17	1.98±0.14	2.15±0.13
	HDLHC	2.27±0.14	1.99±0.16	2.13±0.18

Table 4.7 shows comparison of HDLHC method on SiW dataset with the state-of-the-art methods. In protocol 1, HDLHC method gives

the highest values for APCER and ACER, whereas BCN has the best value of BPCER. In protocol 2, HDLHC method provides the best BPCER and ACER values, whereas BCN has the best value of APCER. In protocol 3, HDLHC method provides the best value for APCER, BPCER and ACER. Overall, HDLHC method performs most of the time as the best one in terms of ACER on all three protocols. These results indicate good generalization of HDLHC method for variations of spoof mediums.

4.9 Summary

Genuine face image and a spoof face image are very similar although careful visual inspection can find small differences between the two. It is thus reasonable to assume that the image quality features can be identified and used to automatically distinguish between genuine and spoof images.

Following this assumption, we identify and propose seven no-reference face image quality features measurement to be used in spoof detection systems. These features are Blurriness, Color, GM-LOG-BIQA, BRISQUE, Reflection, HLFI, and BIQI.

We then introduce a novel face anti-spoofing system called FASS, that uses these no-reference image quality features as an input to the SVM and RF classifiers. It also uses the original images as input to the deep learning ResNet50 classifier and then combines their results. While deep learning classifiers in general perform better than classifiers that use image quality features extracted from images, the results of FASS show that by fusing the outputs of different classifiers that use different feature inputs improves the overall accuracy.

This paper also introduces another novel face anti-spoofing system called HDLHC that takes the advantage of the traditional hand-crafted image quality features and deep learning. HDLHC method uses the proposed manually extracted seven image quality features in addition to the deep learning features. It leverages VGG network which automatically extracts the deep features from the last layer before the classification layer. It then concatenates the two features and feeds into the classifier. Thus, the hand-crafted image quality features complement deep learning features for better generalization. The experimental results on Oulu-NPU and SiW datasets demonstrate the superiority of the HDLHC method when compared with the state-of-the-art methods.

Chapter 5

Conclusions

This chapter provides a summary of this PhD dissertation. The proposed techniques are reviewed regarding the objectives discussed in Chapter 1. Finally, suggestions for future works are described.

5.1 Conclusions

This PhD dissertation introduces five novel methods: an unsupervised deep learning face verification method using an autoencoder based network using only similar images, an unsupervised deep learning face verification method using an autoencoder based network using both similar and dissimilar images, an unsupervised deep learning face verification method using a Siamese based network using both similar and dissimilar images, face anti-spoofing method using image quality measurements and deep learning at score level and finally, face anti-spoofing method using image quality measurements and deep learning at feature level.

State-of-the art deep learning methods for face verification usually require large amounts of labeled data for training. However, it is not always easy to obtain such data. To address this problem, we

propose novel an unsupervised deep learning face verification system called UFace. It does not require labeled training data as well as it does not require a huge amount of training data. It uses both the most k similar and k dissimilar images to a given image and then, it is demonstrated using an autoencoder and Siamese networks. UFace is evaluated on four benchmark face verification datasets, namely, Labeled Faces in the Wild (LFW), YouTube Faces (YTF), Cross-age LFW (CALFW) and Celebrities in Frontal Profile in the Wild (CFP-FP). Its performance using the Siamese network achieved accuracies of 99.40%, 96.04%, 95.12% and 97.89%, respectively, which are comparable with the state-of-the-art methods. Importantly, UFace uses much less data for training. Additional advantage of UFace is that it can be used for verification of other types of images in domains where labeled data are not easily available.

As state-of-the-art face anti-spoofing systems are still fragile in detecting spoofing attacks, we propose a novel face anti-spoof detection system called FASS and HDLHC. Genuine face image and a spoof face image are very similar although careful visual inspection can find small differences between the two. It was thus, reasonable to assume that the image quality features can be identified and used to automatically distinguish between genuine and spoof images. Following this assumption, we identify and propose a novel set of seven no-reference face image quality measurements and use them in FASS and HDLHC. These image quality feature measurements are Bluriness, Color, GM-LOG-BIQA, BRISQUE, Reflection, HLF1, and BIQI. FASS and HDLHC use these features as an input to the classical classifier such as SVM and Random Forest.

FASS also uses the original images as input to the deep learning ResNet50 network and then it combines the scores of the classical

classifier with the ResNet50 classifier in a weighted fashion. While deep learning classifiers in general perform better than classifiers that use image quality features extracted from images, the results of FASS show that by fusing the outputs of different classifiers that use different feature inputs improves the overall accuracy. FASS is evaluated on the face anti-spoofing benchmark datasets such as Replay-Attack, CASIA-MFSD, MSU-MFSD, OULU-NPU and SiW. FASS perform better than several of the state-of-the-art systems during both intra-datasets and extra-datasets testing scenarios. These results confirm the usefulness of the identified seven no-reference image quality features, which can be used by others in their anti-spoofing research.

HDLHC also use the deep features extracted from VGG network in addition to the seven image quality features. It concatenates both features and feed it to the classifier to distinguish between genuine and spoofed faces. HDLHC is evaluated on two recent face anti-spoofing benchmark datasets: OULU-NPU and SiW. The results show that HDLHC outperforms the state-of-the-art face anti-spoofing methods in many scenarios.

5.2 Future Research Lines

The work performed in this PhD dissertation may be used as a guide for future research lines in biometrics identity verification system and other image recognition tasks where there are a limited number of training data. The possible future research lines that can be continued from our work are outlined as follows:

Firstly, the proposed unsupervised deep learning face verification technique is successfully applied for face verification system. Therefore, it is worth to explore the impact of the proposed technique to verify some objects other than human being. Since face tracking and face verification are close to each other and share some components, the proposed technique can also be applied in face tracking systems.

Secondly, the proposed face anti-spoofing system using image quality measurements and deep learning techniques can also be applied to other biometric spoof detection methods such as fingerprint and iris. Since the proposed image quality measurements are not application specific, thus it is worth to try to apply the same techniques to detect the spoofing attack on fingerprint and iris.

Bibliography

- [1] Wenyi Zhao, Rama Chellappa, P Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458, 2003.
- [2] David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. prentice hall professional technical reference, 2002.
- [3] Anil K Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14(1):4–20, 2004.
- [4] Chris Solomon and Toby Breckon. *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. John Wiley & Sons, 2011.
- [5] Aina Puce, Truett Allison, Maryam Asgari, John C Gore, and Gregory McCarthy. Differential sensitivity of human visual cortex to faces, letterstrings, and textures: a functional magnetic resonance imaging study. *Journal of neuroscience*, 16(16):5205–5215, 1996.
- [6] F. Schroff, D. Kalenichenko, and J. Philbin. unified embedding for face recognition and clustering. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 815–823, 2015.

- [7] Y. Taigman, M. Yang, M. Ranzato, and L. Deepface Wolf. Closing the gap to human-level performance in face verification. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 1701–1708, 2014.
- [8] O. Parkhi, A. Vedaldi, and A. Deep face recognition Zisserman. (british machine vision association. 2015.
- [9] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Web-scale training for face identification. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 2746–2754, 2015.
- [10] J. Deng, Y. Zhou, and S. Zafeiriou. Marginal loss for deep face recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition Workshops*, pages 60–68, 2017.
- [11] Y. Sun, D. Liang, X. Wang, and X. Deepid3 Tang. Technical report, *ArXiv*, title = Face recognition with very deep neural networks, type = Preprint, year = 2015, archivePrefix = arXiv, eprint = 1502.00873.
- [12] Z. Zhu, P. Luo, X. Wang, and X. Tang. Technical report, *ArXiv*, title = Recover canonical-view faces in the wild with deep neural networks, type = Preprint, year = 2014, archivePrefix = arXiv, eprint = 1404.3543.
- [13] J. Deng, J. Guo, N. Xue, and S. Arcface Zafeiriou. Additive angular margin loss for deep face recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 4690–4699, 2019.

- [14] Y. Kim, W. Park, M. Roh, and J. Groupface Shin. Learning latent groups and constructing group-based representations for face recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 5621–5630, 2020.
- [15] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Cosface Liu. Large margin cosine loss for deep face recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 5265–5274, 2018.
- [16] Y. Huang, Y. Wang, Y. Tai, X. Liu, P. Shen, S. Li, J. Li, and F. Curricularface Huang. adaptive curriculum learning loss for deep face recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 5901–5910, 2020.
- [17] X. Wang, S. Zhang, S. Wang, T. Fu, H. Shi, and T. Mei. Misclassified vector guided softmax loss for face recognition. *Proceedings Of The AAAI Conference On Artificial Intelligence*, 34:12241–12248, 2020.
- [18] J. Deng, J. Guo, J. Yang, A. Lattas, and S. Zafeiriou. Variational prototype learning for deep face recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 11906–11915, 2021.
- [19] J. Zhang, X. Yan, Z. Cheng, and X. A Shen. face recognition algorithm based on feature fusion. *Concurr. Comput. Pract. Exp.*, 2022.
- [20] Anil Jain, Lin Hong, and Sharath Pankanti. Biometric identification. *Communications of the ACM*, 43(2):90–98, 2000.

- [21] James L Wayman, Anil K Jain, Davide Maltoni, and Dario Maio. *Biometric systems: Technology, design and performance evaluation*. Springer Science & Business Media, 2005.
- [22] Nesli Erdogmus and Sébastien Marcel. Spoofing in 2d face recognition with 3d masks and anti-spoofing with kinect. In *2013 IEEE sixth international conference on biometrics: theory, applications and systems (BTAS)*, pages 1–6. IEEE, 2013.
- [23] Ivana Chingovska, Nesli Erdogmus, André Anjos, and Sébastien Marcel. Face recognition systems under spoofing attacks. *Face Recognition Across the Imaging Spectrum*, pages 165–194, 2016.
- [24] Ramachandra Raghavendra and Christoph Busch. Novel presentation attack detection algorithm for face recognition system: Application to 3d face mask attack. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 323–327. IEEE, 2014.
- [25] Yan Li, Ke Xu, Qiang Yan, Yingjiu Li, and Robert H Deng. Understanding osn-based facial disclosure against face authentication systems. In *Proceedings of the 9th ACM symposium on Information, computer and communications security*, pages 413–424, 2014.
- [26] Freitas Pereira, Anjos T., and De Martino A. J. & marcel, s. *Can face anti-spoofing countermeasures work in a real world scenario?.*, 2013:1–8, 2013.
- [27] Jianwei Yang, Zhen Lei, and Stan Z Li. Learn convolutional neural network for face anti-spoofing. *arXiv preprint arXiv:1408.5601*, 2014.

- [28] Di Wen, Hu Han, and Anil K Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, 10(4):746–761, 2015.
- [29] A. Pinto, H. Pedrini, W. Schwartz, and A. Rocha. Face spoofing detection through visual codebooks of spectral temporal cubes. *IEEE Transactions On Image Processing*, 24:4726–4740, 2015.
- [30] Keyurkumar Patel, Hu Han, and Anil K Jain. Secure face unlock: Spoof detection on smartphones. *IEEE transactions on information forensics and security*, 11(10):2268–2283, 2016.
- [31] Rafael C Gonzalez. *Digital image processing*. Pearson education india, 2009.
- [32] Manpreet Kaur, Jasdeep Kaur, and Jappreet Kaur. Survey of contrast enhancement techniques based on histogram equalization. *International Journal of Advanced Computer Science and Applications*, 2(7), 2011.
- [33] Shiguang Shan, Wen Gao, Bo Cao, and Debin Zhao. Illumination normalization for robust face recognition against varying lighting conditions. In *2003 IEEE International SOI Conference. Proceedings (Cat. No. 03CH37443)*, pages 157–164. IEEE, 2003.
- [34] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.

- [35] Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid object detection. In *Proceedings. international conference on image processing*, volume 1, pages I–I. IEEE, 2002.
- [36] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23:1499–1503, 2016.
- [37] Fethi Smach, Johel Miteran, Mohamed Atri, Julien Dubois, Mohamed Abid, and Jean-Paul Gauthier. An fpga-based accelerator for fourier descriptors computing for color object recognition using svm. *Journal of Real-Time Image Processing*, 2:249–258, 2007.
- [38] Yassin Kortli, Maher Jridi, Ayman Al Falou, and Mohamed Atri. A novel face detection approach using local binary pattern histogram and support vector machine. In *2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, pages 28–33. IEEE, 2018.
- [39] Thibault Napoléon and Ayman Alfalou. Pose invariant face recognition: 3d model from single photo. *Optics and Lasers in Engineering*, 89:150–161, 2017.
- [40] Mahir Karaaba, Olarik Surinta, Lambert Schomaker, and Marco A Wiering. Robust face recognition by computing distances from multiple histograms of oriented gradients. In *2015 IEEE Symposium Series on Computational Intelligence*, pages 203–209. IEEE, 2015.
- [41] Chunde Huang and Jiaxiang Huang. A fast hog descriptor using lookup table and integral image. *arXiv preprint arXiv:1703.06256*, 2017.

- [42] Amir HajiRassouliha, Thiranja P Babarenda Gamage, Matthew D Parker, Martyn P Nash, Andrew J Taberner, and Poul MF Nielsen. Fpga implementation of 2d cross-correlation for real-time 3d tracking of deformable surfaces. In *2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013)*, pages 352–357. IEEE, 2013.
- [43] Olasimbo Ayodeji Arigbabu, Sharifah Mumtazah Syed Ahmad, Wan Azizun Wan Adnan, Salman Yussof, and Saif Mahmood. Soft biometrics: Gender recognition from unconstrained face images using local feature descriptor. *arXiv preprint arXiv:1702.02537*, 2017.
- [44] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.
- [45] M Annalakshmi, S Mohamed Mansoor Roomi, and A Sheik Naveedh. A hybrid technique for gender classification with slbp and hog features. *Cluster Computing*, 22:11–20, 2019.
- [46] A Annis Fathima, S Ajitha, V Vaidehi, M Hemalatha, R Karthigaiveni, and Ranajit Kumar. Hybrid approach for face recognition combining gabor wavelet and linear discriminant analysis. In *2015 IEEE international conference on computer graphics, vision and information security (CGVIS)*, pages 220–225. IEEE, 2015.
- [47] Ladislav Lenc and Pavel Král. Automatic face recognition system based on the sift features. *Computers & Electrical Engineering*, 46:256–272, 2015.
- [48] Şahin Işık. A comparative evaluation of well-known feature detectors and descriptors. *International Journal of Applied Mathematics Electronics and Computers*, 3(1):1–6, 2014.

- [49] Vytautas Perlibakas. Face recognition using principal component analysis and log-gabor filters. *arXiv preprint cs/0605025*, 2006.
- [50] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.
- [51] Phan Khoi, Lam Huu Thien, and Hoai Viet Vo. Face retrieval based on local binary pattern and its variants: A comprehensive study. *International Journal of Advanced Computer Science and Applications*, 7(6), 2016.
- [52] Meng Xi, Liang Chen, Desanka Polajnar, and Weiyang Tong. Local binary pattern network: A deep learning approach for face recognition. In *2016 IEEE international conference on Image processing (ICIP)*, pages 3224–3228. IEEE, 2016.
- [53] Idelette Laure Kambi Beli and Chunsheng Guo. Enhancing face identification using local binary patterns and k-nearest neighbors. *Journal of Imaging*, 3(3):37, 2017.
- [54] Kathryn Bonnen, Brendan F Klare, and Anil K Jain. Component-based representation in automated face recognition. *IEEE Transactions on Information Forensics and Security*, 8(1):239–253, 2012.
- [55] Jianfeng Ren, Xudong Jiang, and Junsong Yuan. Relaxed local ternary pattern for face recognition. In *2013 IEEE international conference on image processing*, pages 3680–3684. IEEE, 2013.

- [56] Sibt Ul Hussain, Thibault Napoléon, and Frédéric Jurie. Face recognition using local quantized patterns. In *British machine vision conference*, pages 11–pages, 2012.
- [57] 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 14*, pages 499–515. Springer, 2016.
- [58] Weiyang Liu, Yan-Ming Zhang, Xingguo Li, Zhiding Yu, Bo Dai, Tuo Zhao, and Le Song. Deep hyperspherical learning. *Advances in neural information processing systems*, 30, 2017.
- [59] Nalini K Ratha, Jonathan H Connell, and Ruud M Bolle. An analysis of minutiae matching strength. In *Audio-and Video-Based Biometric Person Authentication: Third International Conference, AVBPA 2001 Halmstad, Sweden, June 6–8, 2001 Proceedings 3*, pages 223–228. Springer, 2001.
- [60] IJS Biometrics. Iso/iec 30107-1: 2016. information technology biometric presentation attack detection. *Part 1 Fram. Int. Organ. Stand*, 2016.
- [61] Ileana Buhan and Pieter Hendrik Hartel. *The state of the art in abuse of biometrics*. Centre for Telematics and Information Technology, University of Twente, 2005.
- [62] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. A Li. face antispoofing database with diverse attacks. *2012 5th IAPR International Conference On Biometrics (ICB)*, pages 26–31, 2012.

- [63] S. Liu and P. Yuen. Recent progress on face presentation attack detection of 3d mask attack. *Handbook Of Biometric Anti-Spoofing: Presentation Attack Detection And Vulnerability Assessment*, pages 231–259, 2023.
- [64] I. Chingovska, A. Anjos, and S. Marcel. On the effectiveness of local binary patterns in face anti-spoofing. *2012 BIOSIG-proceedings Of The International Conference Of Biometrics Special Interest Group (BIOSIG)*, pages 1–7, 2012.
- [65] Ioannis Pavlidis and Peter Symosek. The imaging issue in an automatic face/disguise detection system. In *Proceedings IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications (Cat. No. PR00640)*, pages 15–24. IEEE, 2000.
- [66] Zhiwei Zhang, Dong Yi, Zhen Lei, and Stan Z Li. Face liveness detection by learning multispectral reflectance distributions. In *2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, pages 436–441. IEEE, 2011.
- [67] Ramachandra Raghavendra, Kiran B Raja, and Christoph Busch. Presentation attack detection for face recognition using light field camera. *IEEE Transactions on Image Processing*, 24(3):1060–1075, 2015.
- [68] Ethan M Rudd, Manuel Gunther, and Terrance E Boult. Paraph: presentation attack rejection by analyzing polarization hypotheses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 103–110, 2016.
- [69] Javier Galbally, Sébastien Marcel, and Julian Fierrez. Image quality assessment for fake biometric detection: Application to

- iris, fingerprint, and face recognition. *IEEE transactions on image processing*, 23(2):710–724, 2013.
- [70] Javier Galbally and Sébastien Marcel. Face anti-spoofing based on general image quality assessment. In *2014 22nd international conference on pattern recognition*, pages 1173–1178. IEEE, 2014.
- [71] Xiaoyang Tan, Yi Li, Jun Liu, and Lin Jiang. Face liveness detection from a single image with sparse low rank bilinear discriminative model. *ECCV (6)*, 6316:504–517, 2010.
- [72] Xinting Gao, Tian-Tsong Ng, Bo Qiu, and Shih-Fu Chang. Single-view recaptured image detection based on physics-based features. In *2010 IEEE International Conference on Multimedia and Expo*, pages 1469–1474. IEEE, 2010.
- [73] Frederique Crete, Thierry Dolmiere, Patricia Ladret, and Marina Nicolas. The blur effect: perception and estimation with a new no-reference perceptual blur metric. In *Human vision and electronic imaging XII*, volume 6492, pages 196–206. SPIE, 2007.
- [74] Pina Marziliano, Frederic Dufaux, Stefan Winkler, and Touradj Ebrahimi. A no-reference perceptual blur metric. In *Proceedings. International conference on image processing*, volume 3, pages III–III. IEEE, 2002.
- [75] Yuanhao Chen, Zhiwei Li, Mingjing Li, and Wei-Ying Ma. Automatic classification of photographs and graphics. In *2006 IEEE International Conference on Multimedia and Expo*, pages 973–976. IEEE, 2006.

- [76] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [77] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.
- [78] Alexis Conneau, Holger Schwenk, Loïc Barrau, and Yann Le-cun. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781*, 2016.
- [79] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [80] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105(12):2295–2329, 2017.
- [81] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [82] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.
- [83] Li Deng. Three classes of deep learning architectures and their applications: a tutorial survey. *APSIPA transactions on signal and information processing*, 57:58, 2012.
- [84] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Dropout Salakhutdinov. a simple way to prevent neural networks from overfitting. *The Journal Of Machine Learning Research*, 15:1929–1958, 2014.

- [85] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.
- [86] Keyurkumar Patel, Hu Han, and Anil K Jain. Cross-database face antispoofing with robust feature representation. In *Biometric Recognition: 11th Chinese Conference, CCBR 2016, Chengdu, China, October 14-16, 2016, Proceedings 11*, pages 611–619. Springer, 2016.
- [87] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.
- [88] Lei Li, Xiaoyi Feng, Zinelabidine Boulkenafet, Zhaoqiang Xia, Mingming Li, and Abdenour Hadid. An original face anti-spoofing approach using partial convolutional neural network. In *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6. IEEE, 2016.
- [89] Amin Jourabloo, Yaojie Liu, and Xiaoming Liu. Face despoofing: Anti-spoofing via noise modeling. In *Proceedings of the European conference on computer vision (ECCV)*, pages 290–306, 2018.
- [90] Yaojie Liu, Amin Jourabloo, and Xiaoming Liu. Learning deep models for face anti-spoofing: Binary or auxiliary supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 389–398, 2018.

- [91] Anjith George and Sébastien Marcel. Deep pixel-wise binary supervision for face presentation attack detection. In *2019 International Conference on Biometrics (ICB)*, pages 1–8. IEEE, 2019.
- [92] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.
- [93] Zitong Yu, Chenxu Zhao, Zezheng Wang, Yunxiao Qin, Zhuo Su, Xiaobai Li, Feng Zhou, and Guoying Zhao. Searching central difference convolutional networks for face anti-spoofing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5295–5305, 2020.
- [94] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.
- [95] Yuhui Xu, Lingxi Xie, Xiaopeng Zhang, Xin Chen, Guo-Jun Qi, Qi Tian, and Hongkai Xiong. Pc-darts: Partial channel connections for memory-efficient architecture search. *arXiv preprint arXiv:1907.05737*, 2019.
- [96] Xin Li, Wei Wu, Tao Li, Yang Su, and Lilin Yang. Face liveness detection based on parallel cnn. In *Journal of Physics: Conference Series*, volume 1549, page 042069. IOP Publishing, 2020.
- [97] Ketan Kotwal, Sushil Bhattacharjee, Philip Abbet, Zohreh Mostaani, Huang Wei, Xu Wenkang, Zhao Yaxi, and Sébastien Marcel. Domain-specific adaptation of cnn for detecting face

- presentation attacks in nir. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 4(1):135–147, 2022.
- [98] A. Jain and S. Li. Handbook of face recognition. (springer. 2011.
- [99] A. Woubie, L. Koivisto, and T. B”ackstr”om. Voice-quality features for deep neural network based speaker verification systems. *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 176–180, 2021.
- [100] A. Nagrani, J. Chung, W. Xie, and A. Voxceleb Zisserman. Large-scale speaker verification in the wild. *Computer Speech Language*, 60, 2020.
- [101] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 770–778, 2016.
- [102] K. Cios and I. Shin. Image recognition neural network: Irnn. *Neurocomputing*, 7:159–185, 1995.
- [103] J. Shin, D. Smith, W. Swiercz, K. Staley, J. Rickard, J. Montero, L. Kurgan, and K. Cios. Recognition of partially occluded and rotated images with a network of spiking neurons. *IEEE Transactions On Neural Networks*, 21:1697–1709, 2010.
- [104] P. Cachi, S. Ventura, and K. Crba: A Cios. Competitive rate-based algorithm based on competitive spiking neural networks. *Frontiers In Computational Neuroscience*, 15, 2021.

- [105] C. Sanderson, M. Saban, and Y. Gao. On local features for gmm based face verification. *Third International Conference On Information Technology And Applications (ICITA '05)*, 1:650–655, 2005.
- [106] C. McCool and S. Marcel. Parts-based face verification using local frequency bands. *International Conference On Biometrics*, pages 259–268, 2009.
- [107] T. Pereira, M. Angeloni, F. Simões, and J. Silva. Video-based face verification with local binary patterns and svm using gmm supervectors. *International Conference On Computational Science And Its Applications*, pages 240–252, 2012.
- [108] G. Marcialis and F. Roli. Fusion of lda and pca for face verification. *International Workshop On Biometric Authentication*, pages 30–37, 2002.
- [109] S. Marcel and S. Bengio. Improving face verification using skin color information. *Object Recognition Supported By User Interaction For Service Robots*, 2:378–381, 2002.
- [110] Y. Wang and Q. Wu. Research on face recognition technology based on pca and svm. in proceedings of the 2022 7th international conference on big data analytics (icbda), guangzhou, china, 04-06 march 2022;. p, pages 248–252.
- [111] J. Bromley, I. Guyon, Y. LeCun, E. S”ackinger, and R. Shah. Signature verification using a” siamese” time delay neural network. *Advances In Neural Information Processing Systems*, 6:737–744, 1993.

- [112] G. Huang, M. Mattar, T. Berg, and E. Learned-Miller. A database for studying face recognition in unconstrained environments, Labeled faces in the wild, 2008.
- [113] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. *CVPR 2011*, pages 529–534, 2011.
- [114] Q. Meng, S. Zhao, Z. Huang, and F. Magface: A Zhou. universal representation for face recognition and quality assessment. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 14225–14234, 2021.
- [115] X. Huang, X. Zeng, Q. Wu, Y. Lu, X. Huang, and H. Zheng. Face verification based on deep learning for person tracking in hazardous goods factories. *Processes*, 2022.
- [116] H. Elaggoune, M. Belahcene, and S. Hybrid descriptor and Bourennane. and optimized cnn with transfer learning for face recognition. In *Multimed. Tools Appl*, pages 9403–9427. **2022**, 81.
- [117] Ben Fredj, H.; Bouguezzi, S.; Souani, and C. Face. recognition in unconstrained environment with cnn. *Vis. Comput*, 2021:217–226.
- [118] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le. Unsupervised data augmentation for consistency training. *Adv. Neural Inf. Process. Syst*, 2020:6256–6268.
- [119] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. Raffel, E. Cubuk, A. Kurakin, and C. Fixmatch Li. Simplifying semi-supervised learning with consistency and confidence. *Adv. Neural Inf. Process. Syst*, 2020:596–608.

- [120] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. in proceedings of the european conference on computer vision (eccv), munich, germany, 08-14 september 2018;.*p*, pages 132–149.
- [121] S. Gidaris, P. Singh, and N. Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv*, 2018.
- [122] Y. Shu, Y. Yan, S. Chen, J. Xue, C. Shen, and H. Wang. Learning spatial-semantic relationship for facial attribute recognition with limited labeled data. in proceedings of the ieee/cvf conference on computer vision and pattern recognition, nashville, tn, usa, 20-25 june 2021;.*p*, pages 11916–11925.
- [123] M. He, J. Zhang, S. Shan, and X. Chen. Enhancing face recognition with self-supervised 3d reconstruction. in proceedings of the ieee/cvf conference on computer vision and pattern recognition, new orleans, louisiana, usa, 21-24 june 2022;.*p*, pages 4062–4071.
- [124] J. Yin, Y. Xu, N. Wang, Y. Li, and S. Guo. Mask guided unsupervised face frontalization using 3d morphable model from single-view images: A face frontalization framework that can generate identity preserving frontal view image while maintaining the background and color tone from input with only front images for training using 3d morphable model. in proceedings of the 2022 4th asia pacific information technology conference, thailand, 14-16 january 2022;.*p*, pages 23–30.
- [125] M. Khan, S. Jabeen, M. Khan, T. Saba, A. Rehmat, A. Rehman, and U. A Tariq. realistic image generation of face from text description using the fully trained generative adversarial networks. *IEEE Access*, 2020:1250–1260.

- [126] Y. Liu and J. Chen. Unsupervised face frontalization for pose-invariant face recognition. *Image Vis. Comput.*, 2021.
- [127] Y. Hu, X. Wu, B. Yu, R. He, and Z. Sun. Pose-guided photorealistic face rotation. in proceedings of the ieee conference on computer vision and pattern recognition, salt lake city, ut, usa, 18-23 june 2018;. *p*, pages 8398–8406.
- [128] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. in proceedings of the aaai conference on artificial intelligence. 30:25–30, January 2015.
- [129] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Inception-v4 Alemi. inception-resnet and the impact of residual connections on learning. *Proceedings Of The AAAI Conference On Artificial Intelligence*, 31, 2017.
- [130] J. Zabalza, J. Ren, J. Zheng, H. Zhao, C. Qing, Z. Yang, P. Du, and S. Marshall. Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging. *Neurocomputing*, 185:1–10, 2016.
- [131] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Adversarial autoencoders Frey. Arxiv preprint (2015).
- [132] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P. Manzagol, and L. Stacked denoising autoencoders Bottou. Learning useful representations in a deep network with a local denoising criterion. *Journal Of Machine Learning Research*, 11, 2010.
- [133] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. Context encoders Efros. Feature learning by inpainting. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 2536–2544, 2016.

- [134] Z. Ling, S. Kang, H. Zen, A. Senior, M. Schuster, X. Qian, H. Meng, and L. Deng. Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends. *IEEE Signal Processing Magazine*, 32:35–52, 2015.
- [135] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. A Alsaadi. survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.
- [136] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. *Proceedings Of The IEEE International Conference On Computer Vision*, pages 3730–3738, 2015.
- [137] T. Zheng, W. Deng, and J. Cross-age lfw: A Hu. Technical report, *ArXiv*, title = database for studying cross-age face recognition in unconstrained environments, type = Preprint, year = 2017, archivePrefix = arXiv, eprint = 1708.08197.
- [138] S. Sengupta. Frontal to profile face verification in the wild. *IEEE Conference On Applications Of Computer Vision*, 2016:2.
- [139] F. Chollet and Others Keras. keras. io, 2015.
- [140] S. Ioffe and C. Batch normalization Szegedy. Accelerating deep network training by reducing internal covariate shift. *International Conference On Machine Learning*, pages 448–456, 2015.
- [141] Y. Duan, J. Lu, and J. Uniformface Zhou. Learning deep equidistributed representation for face recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 3415–3424, 2019.

- [142] K. Zhao, J. Xu, and M. Regularface Cheng. Deep face recognition via exclusive regularization. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 1136–1144, 2019.
- [143] B. Kang, Y. Kim, B. Jun, and D. Kim. Attentional feature-pair relation networks for accurate face recognition. *Proceedings Of The IEEE/CVF International Conference On Computer Vision*, pages 5472–5481, 2019.
- [144] E. Rashedi, E. Barati, M. Nokleby, and X. Chen. Stream loss. : *ConvNet learning for face verification using unlabeled videos in the wild*, 329:311–319, 2019.
- [145] Y. Liu, H. Li, and X. Wang. Technical report, *ArXiv*, title = Rethinking feature discrimination and polymerization for large-scale recognition, type = Preprint, year = 2017, archivePrefix = arXiv, eprint = 1710.00870.
- [146] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition*, pages 4362–4371, 2017.
- [147] J. Jiao, W. Liu, Y. Mo, J. Jiao, Z. Deng, and X. Dyn-arcFace Chen. dynamic additive angular margin loss for deep face recognition. *Multimedia Tools And Applications*, pages 1–16, 2021.
- [148] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Circle loss: A Wei. unified perspective of pair similarity optimization. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 6398–6407, 2020.

- [149] Anjith George, Zohreh Mostaani, David Geissenbuhler, Olegs Nikisins, André Anjos, and Sébastien Marcel. Biometric face presentation attack detection with multi-channel convolutional neural network. *IEEE Transactions on Information Forensics and Security*, 15:42–55, 2019.
- [150] A. Hadid, N. Evans, S. Marcel, and J. Fierrez. Biometrics systems under spoofing attack: an evaluation methodology and lessons learned. *IEEE Signal Processing Magazine*, 32:20–30, 2015.
- [151] C. Rathgeb, P. Drozdowski, and C. Makeup presentation attacks Busch. Review and detection performance benchmark. *IEEE Access*, 8:24958–22497, 2020.
- [152] F. Abdulkutty, E. Elyan, and P. A Johnston. review of state-of-the-art in face presentation attack detection: From early development to advanced deep learning and multi-modal fusion methods. *Information Fusion*, 75:55–69, 2021.
- [153] M. Fang, N. Damer, F. Kirchbuchner, and A. Kuijper. Real masks and spoof faces: On the masked face presentation attack detection. *Pattern Recognition*, 123, 2022.
- [154] Usman Muhammad, Zitong Yu, and Jukka Komulainen. Self-supervised 2d face presentation attack detection via temporal sequence sampling. *Pattern Recognition Letters*, 156:15–22, 2022.
- [155] A. Woubie and T. B”ackstr”om. Voice quality features for replay attack detection. *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 384–388, 2022.

- [156] A. Multi-modal and Benlamoudi. *and anti-spoofing person identification.* University Of Kasdi Merbah, 2018.
- [157] Z. Li. Cross-domain face presentation attack detection techniques with attention to genuine faces. (nanyang technological university. 2023.
- [158] M. Explainable Nóbrega and Interpretable Face. Presentation Attack Detection Methods, 2021.
- [159] M. Micheletto and Others Fusion. of fingerprint presentation attacks detection and matching: a real approach from the livdet perspective. (università degli studi di cagliari. 2023.
- [160] M. Sebastien, M. Nixon, and S. Li. Handbook of biometric anti-spoofing: trusted biometrics under spoofing attacks. (springer. 2014.
- [161] S. Marcel, M. Nixon, J. Fierrez, and N. Evans. Handbook of biometric anti-spoofing: Presentation attack detection. (springer. 2019.
- [162] S. Marcel, M. Nixon, and S. Li. Handbook of biometric anti-spoofing. (springer. 2014.
- [163] C. Related Standards Busch. Handbook of biometric anti-spoofing: Trusted biometrics under spoofing attacks. *p*, pages 205–215, 2014.
- [164] I. Chingovska, J. Yang, Z. Lei, D. Yi, S. Li, O. Kahm, C. Glaser, N. Damer, A. Kuijper, A. Nouak, and Others The. 2nd competition on counter measures to 2d face spoofing attacks. *2013 International Conference On Biometrics (ICB)*, pages 1–6, 2013.
- [165] L. Ghiani, D. Yambay, V. Mura, S. Tocco, G. Marcialis, F. Roli, and S. Schuckcrs. Livdet 2013 fingerprint liveness detection

- competition 2013. *2013 International Conference On Biometrics (ICB)*, pages 1–6, 2013.
- [166] A. Czajka. Pupil dynamics for iris liveness detection. *IEEE Transactions On Information Forensics And Security*, 10:726–735, 2015.
- [167] L. Sun, G. Pan, Z. Wu, and S. Lao. Blinking-based live face detection using conditional random fields. *International Conference On Biometrics*, pages 252–260, 2007.
- [168] Klaus Kollreider, Hartwig Fronthaler, Maycel Isaac Faraj, and Josef Bigun. Real-time face detection and motion analysis with application in “liveness” assessment. *IEEE Transactions on Information Forensics and Security*, 2(3):548–558, 2007.
- [169] Wei Bao, Hong Li, Nan Li, and Wei Jiang. A liveness detection method for face recognition based on optical flow field. In *2009 International Conference on Image Analysis and Signal Processing*, pages 233–236. IEEE, 2009.
- [170] Si-Qi Liu, Xiangyuan Lan, and Pong C Yuen. Remote photoplethysmography correspondence feature for 3d mask face presentation attack detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 558–573, 2018.
- [171] J. Galbally, S. Marcel, and J. Biometric antispoofing methods: A Fierrez. survey in face recognition. *IEEE Access*, 2:1530–1552, 2014.
- [172] D. Menotti, G. Chiachia, A. Pinto, W. Schwartz, H. Pedrini, A. Falcao, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *IEEE Transactions On Information Forensics And Security*, 10:864–879, 2015.

- [173] K. J. Cios. Deep neural networks—a brief history. *Advances In Data Analysis With Computational Intelligence Methods: Dedicated To Professor Jacek Zurada*, pages 183–200, 2018.
- [174] R. Quan, Y. Wu, X. Yu, and Y. Yang. Progressive transfer learning for face anti-spoofing. *IEEE Transactions On Image Processing*, 30:3946–3955, 2021.
- [175] X. Yang, W. Luo, L. Bao, Y. Gao, D. Gong, S. Zheng, Z. Li, and W. Face anti-spoofing: Model matters Liu. so does data. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 3507–3516, 2019.
- [176] T. Kim, Y. Kim, I. Kim, and D. Basn Kim. Enriching feature representation using bipartite auxiliary supervisions for face anti-spoofing. *Proceedings Of The IEEE/CVF International Conference On Computer Vision Workshops*, 2019.
- [177] K. Roy, M. Hasan, L. Rupty, M. Hossain, S. Sengupta, S. Taus, N. Mohammed, and Others Bi-fpnfas. Bi-directional feature pyramid network for pixel-wise face anti-spoofing by leveraging fourier spectra. *Sensors*, 21:2799, 2021.
- [178] A. Ali, S. Hoque, and F. Deravi. Directed gaze trajectories for biometric presentation attack detection. *Sensors*, 21:1394, 2021.
- [179] M. A Kowalski. study on presentation attack detection in thermal infrared. *Sensors*, 20:3988, 2020.
- [180] Y. Jia, J. Zhang, S. Shan, and X. Chen. Unified unsupervised and semi-supervised domain adaptation network for cross-scenario face anti-spoofing. *Pattern Recognition*, 115, 2021.

- [181] Z. Wang, C. Zhao, Y. Qin, Q. Zhou, G. Qi, J. Wan, and Z. Lei. Technical report, *ArXiv*, title = Exploiting temporal and depth information for multi-frame face anti-spoofing, type = Preprint, year = 2018, archivePrefix = arXiv, eprint = 1811.05118.
- [182] Z. Yu, X. Li, X. Niu, J. Shi, and G. Zhao. Face anti-spoofing with human material perception. *European Conference On Computer Vision*, pages 557–575, 2020.
- [183] C. Lin, Z. Liao, P. Zhou, J. Hu, and B. Ni. Live face verification with multiple instantialized local homographic parameterization. *IJCAI*, pages 814–820, 2018.
- [184] Rizhao Cai, Zhi Li, Renjie Wan, Haoliang Li, Yongjian Hu, and Alex C Kot. Learning meta pattern for face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, 17:1201–1213, 2022.
- [185] Shervin Rahimzadeh Arashloo. Unknown face presentation attack detection via localized learning of multiple kernels. *IEEE Transactions on Information Forensics and Security*, 18:1421–1432, 2023.
- [186] Rouqaiah Al-Refai and Karthik Nandakumar. A unified model for face matching and presentation attack detection using an ensemble of vision transformer features. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 662–671, 2023.
- [187] Ravi Pratap Singh, Ratnakar Dash, and Ramesh Kumar Mohapatra. Lbp and cnn feature fusion for face anti-spoofing. *Pattern Analysis and Applications*, 26(2):773–782, 2023.

- [188] Xin Cheng, Jingmei Zhou, Xiangmo Zhao, Hongfei Wang, and Yuqi Li. A presentation attack detection network based on dynamic convolution and multi-level feature fusion with security and reliability. *Future Generation Computer Systems*, 146:114–121, 2023.
- [189] Yongrae Kim, Hyunmin Gwak, Jaehoon Oh, Minho Kang, Jinkyu Kim, Hyun Kwon, and Sunghwan Kim. Cloudnet: A lidar-based face anti-spoofing model that is robust against light variation. *IEEE Access*, 11:16984–16993, 2023.
- [190] A. Woubie, J. Luque, and J. Hernando. *Using voice-quality measurements with prosodic and spectral features for speaker diarization*. Sixteenth Annual Conference Of The International Speech Communication Association, 2015.
- [191] A. Costa-Pazo, S. Bhattacharjee, E. Vazquez-Fernandez, and S. Marcel. The replay-mobile face presentation-attack database. *2016 International Conference Of The Biometrics Special Interest Group (BIOSIG)*, pages 1–7, 2016.
- [192] Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, and G. Zhao. *Deep learning for face anti-spoofing: A survey*. IEEE Transactions On Pattern Analysis And Machine Intelligence, 2022.
- [193] Z. Wang, Z. Wang, Z. Yu, W. Deng, J. Li, T. Gao, and Z. Wang. Domain generalization via shuffled style assembly for face anti-spoofing. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 4123–4133, 2022.
- [194] C. Wang, Y. Lu, S. Yang, and S. PatchNet: A Lai. simple face anti-spoofing framework via fine-grained patch recognition. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*, pages 20281–20290, 2022.

- [195] C. Wang, B. Yu, and J. A Zhou. Learnable gradient operator for face presentation attack detection. *Pattern Recognition*, 135, 2023.
- [196] A. Moorthy and A. A Bovik. modular framework for constructing blind universal quality indices. *IEEE Signal Processing Letters*, 17, 2009.
- [197] A. Mittal, A. Moorthy, and A. Bovik. Making image quality assessment robust. *2012 Conference Record Of The Forty Sixth Asilomar Conference On Signals, Systems And Computers (ASILOMAR)*, pages 1718–1722, 2012.
- [198] X. Zhu and P. A Milanfar. no-reference sharpness metric sensitive to blur and noise. *2009 International Workshop On Quality Of Multimedia Experience*, pages 64–69, 2009.
- [199] Z. Boulkenafet, J. Komulainen, and A. Hadid. Face anti-spoofing based on color texture analysis. *2015 IEEE International Conference On Image Processing (ICIP)*, pages 2636–2640, 2015.
- [200] A. Mittal, A. Moorthy, and A. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions On Image Processing*, 21:4695–4708, 2012.
- [201] W. Xue, X. Mou, L. Zhang, A. Bovik, and X. Feng. Blind image quality assessment using joint statistics of gradient magnitude and laplacian features. *IEEE Transactions On Image Processing*, 23:4850–4862, 2014.

- [202] D. Kundu, D. Ghadiyaram, A. Bovik, and B. Evans. No-reference image quality assessment for high dynamic range images. *2016 50th Asilomar Conference On Signals, Systems And Computers*, pages 1847–1852, 2016.
- [203] A. Moorthy and A. Bovik. Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE Transactions On Image Processing*, 20:3350–3364, 2011.
- [204] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 27:1226–1238, 2005.
- [205] R. Tan and K. Ikeuchi. Separating reflection components of textured surfaces using a single image. *Digitally Archiving Cultural Objects*, pages 353–384, 2008.
- [206] E. Solomon, A. Woubie, and K. UFace Cios. An unsupervised deep learning face verification system. *Electronics*, 11:3909, 2022.
- [207] Enoch Solomon and Krzysztof J Cios. Fass: Face anti-spoofing system using image quality features and deep learning. *Electronics*, 12(10):2199, 2023.
- [208] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Oulu-npu: A Hadid. mobile face presentation attack database with real-world variations. *2017 12th IEEE International Conference On Automatic Face Gesture Recognition (FG 2017)*, pages 612–618, 2017.
- [209] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, and

- Others Pytorch: An imperative style. high-performance deep learning library. *Advances In Neural Information Processing Systems*, 32, 2019.
- [210] J. Galbally, F. Alonso-Fernandez, J. Fierrez, and J. A Ortega-Garcia. high performance fingerprint liveness detection method based on quality related features. *Future Generation Computer Systems*, 28:311–321, 2012.
- [211] I. Chingovska and Dos Anjos. A. & marcel, s. biometrics evaluation under spoofing attacks. *IEEE Transactions On Information Forensics And Security*, 9:2264–2276, 2014.
- [212] R. Ramachandra and C. Busch. Presentation attack detection methods for face recognition systems: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 50:1–37, 2017.
- [213] Z. Boulkenafet, J. Komulainen, Z. Akhtar, A. Benlamoudi, D. Samai, S. Bekhouche, A. Ouafi, F. Dornaika, A. Taleb-Ahmed, L. Qin, and A. Others. competition on generalized software-based face presentation attack detection in mobile scenarios. *2017 IEEE International Joint Conference On Biometrics (IJCB)*, pages 688–696, 2017.
- [214] S. Bharadwaj, T. Dhamecha, M. Vatsa, and R. Singh. Computationally efficient face spoofing detection with motion magnification. *Proceedings Of The IEEE Conference On Computer Vision And Pattern Recognition Workshops*, pages 105–110, 2013.
- [215] Ke-Yue Zhang, Taiping Yao, Jian Zhang, Ying Tai, Shouhong Ding, Jilin Li, Feiyue Huang, Haichuan Song, and Lizhuang

- Ma. Face anti-spoofing via disentangled representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, pages 641–657. Springer, 2020.