Democratic and Popular Republic of Algeria

Ministry of Higher Education and Scientific Research

University of Kasdi Merbah, Ouargla

Faculty of New Information and Communication Technologies

Department of Electronic and Telecommunication

Multi-Modal and Anti-Spoofing Person Identification

Dissertation Submitted to the Department of Electronic and Telecommunication in Candidacy for the Degree of "Doctor" 3rd Cycle LMD in Communication and signal processing.

Presented by:

Azeddine Benlamoudi

Jury Members

President	Fatima Zohra LAALLAM	MCA- Ouargla University
Supervisor	Kamal Eddine AIADI	Prof- Ouargla University
Examiner	Mohamed Assaad HAMIDA	MCA- Ouargla University
Examiner	Fouad CHEBBARA	MCA- Ouargla University
Examiner	Abdelmalik TALEB-AHMED	Prof- Valenciennes University
Examiner	Athmane ZITOUNI	MCA- Biskra University
Guest	Djamel SAMAI	MCB- Ouargla University

Academic Year: 2017 – 2018

Dedication

I dedicate this modest work to:

My beloved parents: **Djamel** and **Bournissa Fatma**, for your prayers and your loves, for their encouragement, affection, advice and sacrifice.
I hope you will find in this work my deep appreciation and respect for you.

My beloved wife: **Hemidi Chahinez**, for your prayers and your loves, for their precious assistant and encouragement, when I needed moral support.

My brothers and sisters: **Tahar**, **Sadek**, **Hanane**, **abdelkrim**, **Hayatte**.

My supervisors: **Djamel Samai**, **Ouafi abdelkrim**, **Abdelmalik Taleb-Ahmed** and **Abdenour Hadid**. May **Allah** grant your health, happiness and long life and make sure that I never disappoint you.

All my friends and colleagues.

All my teachers of the Department of Electrical Engineering of Ouargla and Biskra.

Finally, I dedicate this modest work to all those I love and appreciate.

Acknowledgments

*I wish to thank first and foremost, **ALLAH** the all-powerful who gave me, during all my years of education, health, courage and patience to come to this day.*

*I would like to thank my parents **Djamel & Bournissa Fatma** for their friendship, encouragement and caring over all these years, for always being there for me through thick and thin and without whom this project would not be possible. I would also like to thank my dead grandmother **Boulifa Fatima & Athmani Mouni** my **ALLAH** bless her soul, and my aunts, uncles and cousins for their understanding and support throughout all these years.*

*I would also like to acknowledge my dissertation supervisors **Dr. Samai Djamel, Prof. Ouafi abdekrime, Prof. Abdelmalik TALEB-AHMED and Prof. Abdenour Hadid** for their insight, support and sharing of knowledge that has made this Thesis possible. My sincere thanks also go to **Prof. Kamal Eddine AIADI** who provided me the opportunity to join the PhD study.*

*I would like to thank my wife **Hemidi Chahinaz** for her love and constant support, and for keeping me sane over the past few months. But most of all, thank you for being my best friend. I owe you everything.*

*My special thanks are extended to the staff of **IVS** company (bosses **Franck SELVE** and **Hervé SERGEANT**, my ex-teammates **Vincent, Lou, Quentin, Siegfried, Sophie, JB, Kevin, François, and Clément**) for their assistance to pursue my graduate school studies.*

*Last but not least, to my friends **Rostom, Seif and Salah** and all colleagues that helped me grow as a person and were always there for me during the good and bad times in my life. Thank you.*

To each and every one of you – Thank you.

Abstract

Because of its natural and non-intrusive interaction, identity verification and recognition using facial information is among the most active areas in computer vision research. Unfortunately, it has been shown that conventional 2D face recognition techniques are vulnerable to spoof attack, where a person tries to masquerade as another one by falsifying his biometric data and there by gaining an illegitimate advantage.

This thesis explores different directions for software-based face anti-spoofing. In this context, we proposed a new approach which can be applied in both static and dynamic face anti-spoofing. The proposed approach consists of the following three main stages: 1) face alignment and preprocessing; 2) feature extraction and selection; 3) classification. The purpose of face alignment is to localize faces in images, rectify the 2D pose of each face and then crop the region of interest. The preprocessing stage is important since the subsequent stages depend on it and since it can affect the final performance of the system. Feature extraction and selection stage extracts the facial features. These features are extracted either by a holistic method or by a local method. The extracted features are then selected using a supervised feature selection method in order to omit possible irrelevant features. In the last stage, the classification is used to differentiate between real and fake faces.

The proposed methods are applied to several case studies for the face mode. At face alignment, the results show the advantage of using the rotation of face. At face representation stage, the use of Frame-Difference improves the performance of the system. Also, a comparison between Multi-Block and Multi-Level on face representation is presented. The case studies are furthermore used to demonstrate the framework and its potential in the evaluation of biometric under spoofing attacks. Overall, the experimental results prove the importance of our method for creating trustworthy face anti-spoofing systems.

Keywords

Computer Vision, Biometrics, Face Recognition, Spoofing-attacks, Counter-Measures, Anti-

Spoofing, Liveness Detection, Presentation Attack, Face Representation, Frame-Difference, Fisher-Score, Face Alignment.

Résumé

En raison de son interaction naturelle et non intrusive, la vérification de l'identité et la reconnaissance à l'aide d'informations faciales figurent parmi les domaines les plus actifs de la recherche en vision par ordinateur. Malheureusement, il a été démontré que les techniques classiques de reconnaissance faciale en 2D sont vulnérables aux attaques par usurpation, où une personne tente de se faire passer pour une autre en falsifier ses données biométriques et en y gagnant un avantage illégitime.

Cette thèse explore différentes directions pour l'anti-usurpation de visage basée sur le logiciel. Dans ce contexte, nous avons proposé une nouvelle approche qui peut être appliquée dans le cas statique et dynamique sur l'anti-usurpation de visage. L'approche proposée comprend les trois étapes principales suivantes: 1) Alignement de visage et prétraitement; 2) L'extraction et la sélection de caractéristiques; 3) Classification. Le but de l'alignement du visage est de localiser les visages dans les images, de rectifier la pose 2D de chaque visage et ensuit de recadrer la région d'intérêt. L' étape de prétraitement est importantes car les étapes ultérieures en dépendent et peuvent affecter les performances finales du système. L' étape d'extraction et de sélection de caractéristiques permet d'extraire les traits du visage. Ces caractéristiques sont extraites soit par une méthode globale ou par une méthode local. Les caractéristiques extraites sont ensuite sélectionnées en utilisant un procédé de sélection de caractéristique supervisée afin d'éliminer d'éventuelles fonctions non pertinentes. Dans la dernière étape, la classification est utilisée pour la différentier entre les visages réels et faux.

Les méthodes proposées sont appliquées à plusieurs études de cas pour le mode visage. À l'alignement du visage, les résultats montrent l'avantage d'utiliser la rotation du visage. À l'étape de la représentation du visage, l'utilisation de la différence de frame améliore les performances du système. De plus, une comparaison entre la représentation multi-blocs et multi-niveaux est présentée. Les études de cas sont en outre utilisées pour démontrer le système et son potentiel pour l'évaluation des attaques biométriques, sous usurpation d'identité. Dans l'ensemble, les résultats expérimentaux prouvent l'importance de notre méthode pour la création

de systèmes fiables pour anti-usurpation de visage.

Mote Clé

Vision par Ordinateur, Biométrie, Reconnaissance de Visage, Attaques par Usurpation d'identité, Contre-Mesures, Anti-Usurpation, Détection de Vivacité, Attaque de Présentation, Représentation de Visage, Différence de Frame, Score de Fisher, Alignement de Visage.

الملخص:

بسبب تفاعلها الطبيعي وغير التدخلي ، التحقق من الهوية باستخدام معلومات الوجه هو من بين المجالات الأكثر نشاطا في بحوث الرؤية الحاسوبية. لسوء الحظ، فقد تبين أن تقنيات التعرف على الوجه التقليدية عرضة للهجوم بانتحال الشخصية، حيث يحاول شخص التنكر بصفة شخص آخرعن طريق تزوير البيانات البيومترية لهذا الشخص وذلك من أجل الحصول على ميزة غير مشروعة.

هذه الأطروحة تستكشف اتجاهات مختلفة للوجه القائم على البرمجيات لمكافحة الانتحال. وفي هذا السياق، اقترحنا نهجا جديدا يمكن تطبيقه على الوجهين الساكن والديناميكي على السواء. ويتكون النهج المقترح من المراحل الرئيسية الثلاث التالية: ١) محاذاة الوجه و المعالجة المسبقة؛ ٢) استخراج البيانات؛ ٣) التصنيف. الغرض من محاذاة الوجه هو تحديد الوجوه في الصور، وتصحيح شكل الوجه و قص المنطقة المهتم بها. وتعتبر مرحلة المعالجة المسبقة مهمة لأن المراحل اللاحقة تعتمد عليها، و يمكن أن تؤثر على الأداء النهائي للنظام. في مرحلة استخراج البيانات يتم من خلالها اضهار ملامح الوجه. يتم استخراج هذه الملامح إما بطريقة شمولية أو بطريقة محلية. ثم يتم اختيار الميزات المستخرجة باستخدام طريقة اختيار المراقبة من أجل حذف الميزات التي ليس لها صلة . في المرحلة الأخيرة، التصنيف يستخدم في التفريق بين الوجوه الحقيقية والوهمية.

طبقنا الطرق المقترحة على العديد من دراسات الحالة لوضع الوجه. في محاذاة الوجه، تظهر النتائج ميزة استخدام دوران الوجه. في مرحلة تمثيل الوجه، واستخدام الفرق بين الصوريحسن أداء النظام. أيضا، يتم عرض مقارنة بين متعددة التقسيمات ومتعددة المستويات. كما تستخدم دراسات الحالة لإثبات قوة النظام المقترح في حالة القياسات الحيوية تحت هجمات الانتحال. وعموما، فإن النتائج التجريبية تثبت اهمية الطريقة المقترحة لخلق نظام جدير بالثقة من اجل مكافحة انتحال الوجه.

الكلمات الدلالية:

رؤية حاسوبية، القياسات الحيوية، التعرف على الوجه، هجمات انتحال، تدابير مكافحة الانتحال، كشف الحياة، هجوم العرض، تمثيل الوجه، الفرق بين الصور، نقاط فيشر، محاذاة الوجه.

Scientific Productions

Publications in journals

- **A. Benlamoudi**, KE. Aiadi, A. Ouafi, D. Samai, and M. Oussalah, "Face Anti-Spoofing Based-on Frame Difference and Multi-Level Representation," J. Electron. Imaging. 26(4), 043007 (Jul 21, 2017). http://dx.doi.org/10.1117/1.JEI.26.4.043007.

Publications in international conferences

- SE. Bekhouche, A. Ouafi, A. Taleb-Ahmed, A. Hadid, and **A. Benlamoudi**, "Facial age estimation using bsif and lbp," First International Conference on Electrical Engineering ICEEB'14, December 2014.

- **A. Benlamoudi**, A. Samai, A. Ouafi, A. Taleb-Ahmed, SE. Bekhouche, and A. Hadid, "Face spoofing detection from single images using active shape models with Stasm and LBP," Troisième Conférence internationale sur la Vision Artificielle CVA' 2015, April 2015.

- SE. Bekhouche, A. Ouafi, **A. Benlamoudi**, A. Taleb-Ahmed, and A. Hadid, "Facial age estimation and gender classification using multi level local phase quantization," 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT), May 2015.

- **A. Benlamoudi**, D. Samai, A. Ouafi, SE. Bekhouche, A. Taleb-Ahmed, and A. Hadid, "Face spoofing detection using local binary patterns and Fisher Score," 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT), May 2015.

- SE. Bekhouche, A. Ouafi, **A. Benlamoudi**, A. Taleb-Ahmed, and A. Hadid, "Automatic age estimation and gender classification in the wild," International Conference on Automatic control, Telecommunications and Signals (ICATS15), November 2015.

- **A. Benlamoudi**, D. Samai, A. Ouafi, SE. Bekhouche, A. Taleb-Ahmed, and A. Hadid, "Face spoofing detection using multi-level local phase quantization (ML-LPQ)," International Conference on Automatic control, Telecommunications and Signals (ICATS15), November 2015.

- Boulkenafet Z, Komulainen J, Akhtar Z, **Benlamoudi A**, Samai D, Bekhouche SE, Ouafi A, Dornaika F, Taleb-Ahmed A, Qin L, Peng F, Zhang L B, Long M, Bhilare S, Kanhangad V, Costa-Pazo A, Vazquez-Fernandez E, Pérez-Cabo D, Moreira-Pésrez J J, Gonzélez-Jiménez D, Mohammadi A, Bhattacharjee S, Marcel S, Volkova S, Tang Y, Abe N, Li L, Feng X, Xia Z, Jiang X, Liu S, Shao E, Yuen P C, Almeida W R, Andalé F, Padilha E, Bertocco G, Dias W, Wainer J, Torres E, Rocha A, Angeloni M A, Folego G, Godoy A and Hadid A, "A Competition on Generalized Software-based Face Presentation Attack Detection in Mobile Scenarios," International Joint Conference on Biometrics (IJCB), 2017.

- Soraya Zehani, A. Toumi, **A. Benlamoudi**, A. Taleb-Ahmed and M. Mimi, " Features Extraction using Different Histograms for Texture Classification", Fifth International Conference on Image and Signal Processing and their Applications (ISPA), December 2017.

Publications in national conferences

- ME. Zighem, A. Ouafi, SE. Bekhouche, **A. Benlamoudi**, and A. Taleb-Ahmed, "Age Estimation Based on Color Facial Texture," 10 ème Conférence sur le Génie Electrique, April 2017.

- **A. Benlamoudi**, F. Bougourzi, ME. Zighem, SE. Bekhouche, A. Ouafi, and A. Taleb-Ahmed, "Face Anti-Spoofing Combining MLLBP and MLBSIF," 10 ème Conférence sur le Génie Electrique, April 2017.

- F. Bougourzi, SE. Bekhouche, ME. Zighem, **A. Benlamoudi**, A. Ouafi, and A. Taleb-Ahmed, "A Comparative Study On Textures Descriptors In Facial Gender Classification," 10 ème Conférence sur le Génie Electrique, April 2017.

Contents

List of Figures

List of Tables

Acronyms

APCER Attack Presentation Classification Error Rate

BPCER BonaFide Presentation Classification Error Rate

BSIF Binarized Statistical Image Features

CNN Convolutional Neural Network

DET Detection Error Trade-Off

DMD Dynamic Mode Decomposition

DSIFT Dense Scale Invariant Feature Transform

SIFT Scale Invariant Feature Transform

SURF Speeded-Up Robust Features

DoG Difference of Gaussian

EER Equal Error Rates

ELBP Extended Local Binary Pattern

ELLR Extended LikeLihood ratio

FAR False Acceptance Rate

FD+ML+LPQ+FS Frame Difference+Multi Level+Local Phase Quantization+Fisher-Score

FD+ML+LPQ Frame Difference+Multi Level+Local Phase Quantization

FD+ML Frame Difference+Multi Level

FD Frame Difference

FRR False Rejection Rate

FS Fisher-Score

HOOF Histogram of Oriented Optical Flow

HSV Hue, Saturation, and Value

HTER Half Total Error Rate

IDA Image Distortion Analysis

IJCB International Joint Conference on Biometrics

IQA Image Quality Assessment

LBP-TOP Local Binary Patterns on Three Orthogonal Planes

LBPV Local Binary Patterns Variance

LBP Local Binary Patterns

LDA Linear Discriminant Analysis

LLR LikeLihood ratio

LOCO Leave One Camera Out

LPQ Local Phase Quantization

MB-LPQ Multi Block Local Phase Quantization

MBSIF-TOP Multiscale Binarized Statistical Image Features on Three Orthogonal Planes

MB Multi Block

ML-LPQ Multi Level Local Phase Quantization

MLPQ-TOP Multiscale Local Phase Quantization on Three Orthogonal Planes

ML Multi Level

PAI Presentation Attack Instruments

PRIP Patterns Recognition and Image Processing

PSF Point Spread Function

PS Pictorial Structure

RBF Kernel Radial Basis Function Kernel

RGB Red Green Bleu

ROC Receiver Operating Characteristic

ROI Region Of Interest

STFT Short-Time Fourier Transform

SVM Support Vector Machine

VJ Viola & Jones

YCbCr Luminance; Chroma Blue; Chroma Red

1

Introduction

Contents

1.1 Introduction

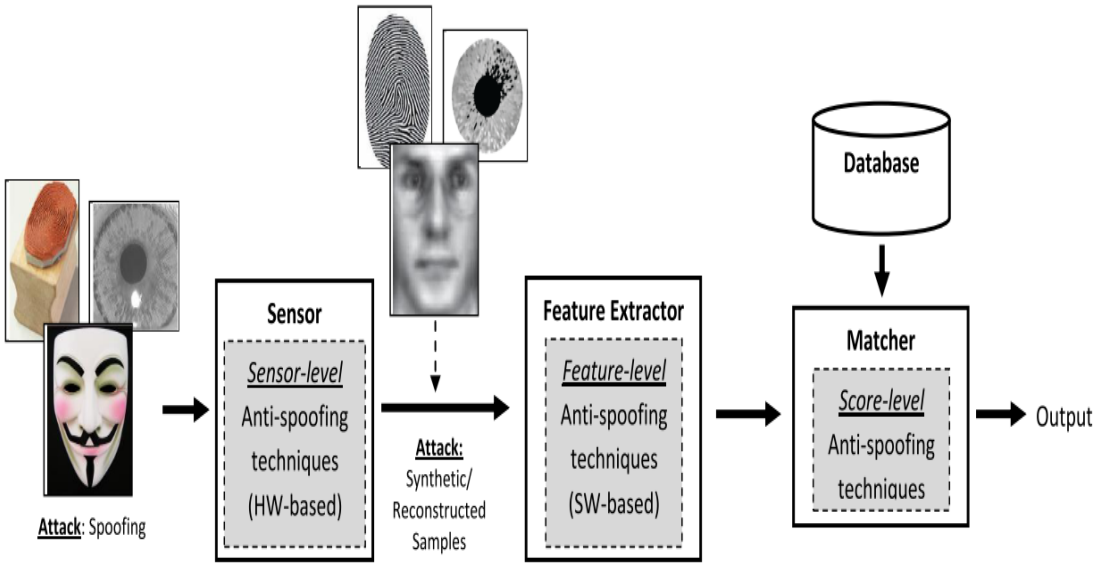


Figure 1.1: General diagram of a biometric system specifying the modules where the three types of anti-spoofing techniques may be integrated (sensor-level, feature-level and score-level). Also displayed are the two different type of attacks for which anti-spoofing techniques may offer protection: spoofing and attacks carried out with synthetic or reconstructed samples [1].

There are numerous types of identity documents: national identity card, passports, social security card, health insurance card, employer's card, banker's card, driving license, etc. Most of the identity documents, with the exception of the national identity card, are function-specific and context-dependent, even though in practice they may be accepted in other contexts. Identity management for persons using biometrics has indeed become a reality not only because of the biometric passport (e-passport) but also because of the presence of more and more biometric-enabled applications for personal computers and mobile phones [2]. Furthermore, a growing number of developing countries are using biometric technologies to create national identification programs.

Biometric technology has much application in our life. They include access control, border control, civil registry, entertainment, finance, forensic, health care, law enforcement, social media, social networking, surveillance, robotics, human-computer interaction, games, transportation, etc. This technology is needed and dominated by security-related applications in a

lot of markets due to increasing security threats in recent times [3]. Also, increase in unauthorized immigration, visa fraud, credit card fraud, border intrusion, and so on leads to a growing need for high security. Such see the importance of biometric technology and have been shown to be promising candidates for either replacing or augmenting conventional security technologies. For those applications of biometric more than for others, reliable recognition is of great importance.

The biometric technology such we can see the importance of uses as security in a lot of application also is easy to attack (spoof). Attacks and vulnerabilities of biometric systems that are being reported to the public from hacking groups [1] attempting to get recognition, from real criminal cases, or even from live demonstrations at biometric and security specific conferences.

As a consequence, in recent years, there has been an increasing interest in the evaluation of biometric systems security, which has led to the creation of numerous and very diverse initiatives focused on this field of research: publication of many research works disclosing and evaluating different biometric vulnerabilities, proposal of new protection methods, related books and book chapters, Ph.D. and MSc theses which propose and analyze different biometric spoofing and anti-spoofing techniques, publication of several standards in the area and of different supporting documents and protection profiles in the framework of the security evaluation standard common criteria for the objective assessment of commercial systems, certification of different commercial products in the framework of the common criteria, patented anti-spoofing mechanisms for biometric systems, specific tracks, sessions and workshops in biometric specific and general signal processing conferences, organization of competitions focused on vulnerability assessment, acquisition of specific datasets, creation of groups and laboratories specialized in the evaluation of biometric security, European projects with the biometric security topic as their main research interest.

In the last years, the spoofing biometric security context described above has promoted, and the significant amount of research which has resulted in publications in journals, conferences, and media, describing new anti-spoofing algorithms and systems that intend to make this technology safer. The most deployed, popular and mature modalities such as a face, fingerprints an iris in biometric technology are also been shown to be the most exposed to spoofing. At the

moment, the amount of new contributions and initiatives in the area of anti-spoofing requires a significant condensation effort to keep track of all new information in order to form a clear picture of the state-of-the-art as of today. As an example, a series of chronological milestones related to the evolution of biometric spoofing are shown in Figure. 1.1.

1.2 Background and Motivation

Biometric experts agree that it is impractical to prevent a collection of biometric data from an individual [4]. The ANSI standard committee formulated what is called a security axiom for biometrics: "The security of a biometric system cannot rely on keeping biometric data secret" [5]. Rather, they recommend building preventive measurements to defend against fabricated replicas of biometric samples. O'Gorman [6] rightfully declares that it is not the secrecy what makes a good authenticator, but the difficulty to counterfeit the original. He argues that copy-resistance goes along with uniqueness as a fundamental principle a good biometrics should stand upon [7]. This gives the essence of the motivation to develop counter-measures to spoofing attacks in order to foster even wider adoption of biometrics as an authentication method.

Due to the increasing need and investments for security applications, authentication by biometric verification is becoming increasingly common in corporate and public security systems. The reason is that biometrics enable reliable and efficient identity management systems by using physical and behavioral characteristics of the subjects that are permanent, universal and easy to access [8]. Each biometric trait has their own advantages and disadvantages. For example, the fingerprint is the most wide-spread biometric from a commercial point of view [9], however, it requires strong user collaboration. Similarly, iris recognition is very accurate, however, it highly depends on the image quality and also requires the active participation of the subjects. Face recognition is advantageous in terms of both accessibility and reliability. It allows identification at relatively high distances for unaware subjects that do not have to cooperate.

It is important to note that the spoofing attacks arise as an issue from the practical usage of biometrics, rather than as a problem inspired by a scientific curiosity. Ever since Matsumoto *et al.* [10] demonstrated the vulnerability of several commercial fingerprint recognition devices to

spoofing attacks with gummy fingers, every new commercial biometric authentication system is being put to similar tests by security enthusiasts. For example, the authors in [11] successfully deceived the face authentication systems of several laptops with fake facial images at the Black Hat Security Conference. The first commercial fingerprint authentication on smartphones has been spoofed with artificial fingers too [12]. While the goal of the above-mentioned examples is to draw attention to the vulnerability of biometric recognition systems, criminal acts involving spoofing attacks on deployed biometric systems have been recorded as well. The case of an illegal immigrant trying to deceive the airport fingerprint scanner in Japan with a tape with someone else's fingerprint is one of the examples [13]. Another one [14] concerns a doctor who falsely registers her colleagues as present at work by spoofing the fingerprint scanner tracking the employee attendance.

In this thesis, two challenges in face recognition are analyzed, which are spoofing and disguise variations. Although these challenges affect the performances of 2D face recognition systems significantly, the studies on these topics are limited. In a spoofing attempt, a person tries to masquerade as another person and thereby, tries to gain access to a recognition system. Since face data can be acquired easily in a contact less manner, spoofing is a real threat for face recognition systems. Due to the limited number of studies on this topic, today spoofing (including anti-spoofing) is a very popular topic for researchers in face recognition domain. In this dissertation, the main motivation is to develop countermeasure techniques in order to protect face recognition systems against spoofing attacks. For this purpose, we investigated 2D face anti-spoofing see Figure. 1.2.

1.3 Objectives and Contributions

The most common spoofing attacks to face recognition systems are achieved by using photographs and videos due to their convenience and low cost. It has been shown that face recognition systems are vulnerable to photograph and video attacks. The aim is to develop non-intrusive countermeasures without extra devices and human involvement which can be integrated into existing face recognition systems to protect them against spoofing attacks.

Figure 1.2: Example of face spoofing.

This thesis focuses on exploring software-based approaches for improving the robustness of 2D face authentication systems to spoofing attacks. All proposed methods are based on analyzing single image or short video sequences captured with conventional cameras, i.e. the sensor embedded in the face verification system that acquires the samples for the actual recognition.

1.4 Thesis Outline

The rest of this thesis is as follows:

Chapter 2 gives an overview on face anti-spoofing, including an introduction to the vulnerabilities of face authentication systems, a literature review of the state-of-the-art techniques and description of the publicly available databases used in the experiments of this thesis.

Chapter 3 describes the key ideas behind texture based face anti-spoofing, then summarizes the main findings in print and video attack detection.

Chapter 4 illustrates the methods proposed in the previous chapters on case studies in the face mode. The case studies are based on several state-of-the-art face anti-spoofing systems and include extensive experiments to assess their performance on databases in face spoofing detection.

Chapter 5 conclude the thesis with a summary of its contributions and achievements and we give an outline of possible directions for future work.

2

Overview on biometric anti-spoofing

Contents

2.1 Introduction

For good and fast security, many applications face recognition has become an important topic. The biometric system is a security identification and authentication system. Such system uses automated methods of verifying or recognizing the identity of a living person based on his physiological or behavioral characteristics. Which is permanent, universal and easy to access. These characteristics include fingerprints, facial, Iris, and voice. This is why the topic of biometrics attracts higher attention today.

In this section, we give first a general introduction to the vulnerabilities of biometric systems and anti-spoofing. Then, the scope is narrowed down to 2D face modality and various aspects of face anti-spoofing, including different attack scenarios and the state of the art in face spoof detection with a particular focus on software-based countermeasures. Finally, face spoofing related benchmark datasets are introduced which have been the basis of the work in this thesis.

2.2 Biometric anti-spoofing

To understand biometric spoofing and come with practically usable anti-spoofing systems, one needs to go through several stages. Studying biometrics in general is the first stage. the second stage is face recognition. In the last stage, we studied spoofing attacks on face recognition and how they are created and performed.

2.2.1 Biometrics in General

Biometric systems are usually consist of the following components (See Figure. 2.1):

- Sensor module: this module acquires biometric data (e.g. face image).
- Feature extraction module: This module is used to extract features of a biometric trait (e.g. fingerprint minutiae).
- Matching module: The matching module compares the acquired biometric features with the stored biometric templates and then match (similarity) scores are generated.

- Decision-making module: The user's identity is is accepted or rejected based on the scores.

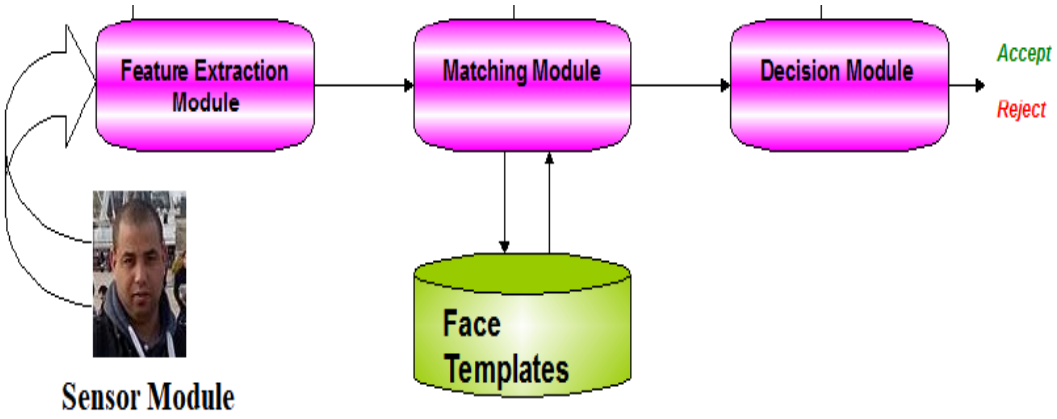


Figure 2.1: biometric system architecture.

There are two modes in biometric systems, which are enrollment mode and authentication mode. Furthermore, authentication is achieved either in verification mode or identification mode [15].

- Enrollment mode: Subjects present one or more biometric data samples. The biometric templates are generated from these samples. These templates constitute the gallery set. Enrollment is generally performed in a well-controlled environment.

- Authentication mode: Biometric data of user is acquired and used by the system either for verification or identification purposes. The biometric data captured for recognition is a probe sample.

  In verification mode, the probe sample is matched with the claimed template for validation, and it either accepts or rejects the identity claimed. Verification is one-to-one matching.

  On the other hand, in identification mode, all biometric references in the gallery are examined and the one with the best match-score denotes the class of the input.

In verification mode, if the match score is above some threshold, the identity claimed is accepted. Otherwise, it is rejected. There are four outcomes of this setting which are:

- True accept: the person is a genuine and the demand is verified.
- True reject: the person is an impostor and the demand is not verified.
- False accept: the person is an impostor and the demand is verified.
- False reject: the person is a genuine and the demand is not verified.

In order to show the verification performances of the recognition systems, generally ROC curve is utilized. It represents the probability of true acceptance versus probability of false acceptance. In this thesis, we report the verification performances not only using ROC curves. In some cases, we also show verification performances using DET curve, which is a variant of ROC curve. The primary difference is that y-axis is the false rejection rate instead of true acceptance rate in DET curve. Finally, in this study, we also use the term EER to show verification performances. EER is the value where False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal (See Figure. 2.2).

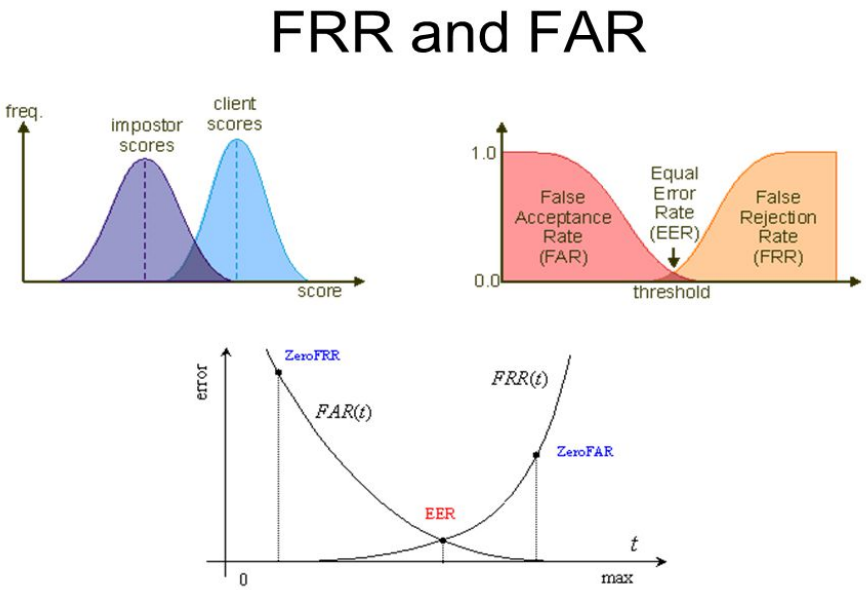


Figure 2.2: False rejection rate and false acceptance rate of a biometric verification system.

2.2.2 Face Recognition

The human face plays an important role in our social interaction by conveying people's identity. Using the human face as a key to security, biometric face recognition (See Figure 2.3) technology has received significant attention in the last several years due to its potential for a wide variety of applications in both law enforcement and non-law enforcement.

As compared with other biometrics systems using fingerprint, palm-print and iris, face recognition has distinct advantages because of its non-contact process. Face images can be captured from a distance without touching the person being identified, and the identification does not require interacting with the person. In addition, face recognition serves the crime deterrent purpose because face images that have been recorded and archived can later help identify a person.

Figure 2.3: Examples of face recognition system.

2.2.3 Spoofing attacks in face recognition

The main reason spoofing attacks in face recognition is that a biometric sample is a face represented in a digital image, which is intrinsically highly reproducible by several means like printed photos and electronic portable devices (laptops and even cellular phones) capable of showing images and videos [16–19]. In the context of face biometrics, an impostor tries to access the system as a valid user with three approaches (See Figure 2.4) [16, 20–22]:

- Showing a photography of a valid user (See Figure 2.4(a))
- Showing a video of a valid user (See Figure 2.4(c))
- Showing a 3D facial model of a valid user (See Figure 2.4(b))

(a) photography **(b)** 3D facial model

(c) video

Figure 2.4: Examples of face spoofing.

Besides, a face recognition system can be built with very low cost hardware, and it is particularly suitable for low risk application. Therefore, in these cases, adding of specific hardware or interaction to ensure reliability is not necessarily an affordable solution. This implies that a simple photo spoofing attack can represent a security problem for a face recognition system. In fact, most of papers in the literature refer to the problem as a task of photo attack detection as it represents a cheap and effective way to perform an attack. Commonly cited papers refer to the problem of photo attack detection in two major complementary directions [17–19] :

- Static analysis, based on the fundamental idea that during the manufacturing process of a photo attack, a certain loss of information occurs and also peculiar noise is introduced [18].

- Video analysis, that tries to detect, as humans do, facial physiological clues like blinks, mouth movements and changes in facial expression [18].

For anti-spoofing methods there are four major categories [22, 23]: data-driven characterization, user behavior modeling, user interaction need, and the presence of additional devices.

To counteract the threat of spoofing, various liveness detection methods have been investigated, particularly for the three main physical biometric methodologies: fingerprint [24, 25] face [26, 27] and iris [20] recognition. However, all liveness detection techniques can be classified into one of the following three categories [28, 29]:

- Intrinsic properties of a living body: A living human body exhibits a number of measurable properties such as: density and elasticity (physical), capacity, resistance, and permittivity (electrical), reflectance and absorbance (spectral), color and opacity (visual).

- Involuntary signals of a living body: Pulse, blood pressure, fluctuation of papillary size, perspiration, blood flow brain wave signals, and electrical heart signals.

- Bodily responses to external stimuli: Those are challenge-response methods that either look for voluntary or reflexive behavioral responses from the user. Asking user to smile or blink is an example of a voluntary behavioral response. Pupil dilation is an example of involuntary response.

2.3 State of the art in face anti-spoofing

There are many different terms to differentiate between biometric trait originates from a living legitimate subject or from some other source as as anti-spoofing, spoof detection, and presentation attack detection. Also, there is another term referred called liveness detection and is a synonym for spoof detection in some fields but, in general, it can be used to refer to a more

limited problem of sensing vitality signs, like eye blinking or heartbeat. In this thesis, this term is treated as a subcategory for anti-spoofing methods.

Spoofing attack detection can be performed before capturing the actual biometric data or when processing the acquired sample using three different approaches:

- 1) use only the information acquired for identification purposes.
- 2) further process the data or acquire additional information over time to find clues of a possible spoofing attack.
- 3) use additional sensors and software to find out a representation that is more suitable for capturing inherent differences between genuine subjects and fake ones than the original biometric data.

The anti-spoofing research has mainly concentrated on further processing and collecting the biometric data, or using additional instruments because based on the acquired biometric sample it is really hard to tell if the presented biometric trait is valid or not [2].

Like in the case of attack terminology, there exists no unified taxonomy for the different spoof detection approaches. The aforementioned techniques can be categorized in several ways, e.g. based on the working mechanisms into methods utilizing the intrinsic properties of the biometric samples, liveness cues or contextual information, or based on the biometric system module in which they are integrated into sensor-level (hardware-based), feature-level (software-based) or score-level techniques.

In this thesis, however, the score-level techniques are not considered as a separate technique but as a part of the whole biometric system design which is a research topic of its own that is not related only to anti-spoofing and has not been explored much yet. Thus, a three-part categorization is followed dividing the individual spoof detection schemes into hardware-based, software-based and multi-modal techniques.

There are many ways to detect spoof attacks. In this work, we will study two part of face anti-spoofing methods which are Hardware-based and Software-based techniques. In the following section, we present all previous work in face anti-spoofing techniques (See Table 2.1).

2.3.1 Hardware based techniques

The Hardware-based techniques advocate incorporating extra hardware devices in order to differentiate between the real and the fake faces.

We give in this section the recent work in this domain Ng *et al.* [30] used randomized temporal effective cues in the form of facial expressions to verify the liveness of users. Pavlidis *et al.* [31] showed that band of the near-infrared ($1.4um - 2.4um$) is particularly advantageous for disguise detection purposes. Chetty *et al.* [32] combined acoustic and visual feature vectors to distinguish live synchronous audio-video recordings from Replay-Attacks that use audio with a still photo. Erdogmus *et al.* [33] used the depth information to discriminate between the real and the 2D spoofing attacks. Smith *et al.* [34] proposed an approach for face recognition systems that can counter the attacks by using the color reflected from the user face which is displayed on the mobile devices. These reflections are used to determine whether the images were captured in real-time. Wang *et al.* [35] proposed a novel face liveness detection approach to counter spoofing attacks by recovering sparse 3D facial structure. Other methods [36–38] used different visual spectrum (complementary infrared, near infrared...) to distinguish between the genuine faces and the spoof attacks.

2.3.2 Software based techniques

The software-based techniques use the simple Red Green Bleu (RGB) images to detect the spoof attacks. These methods can be divided into static based and dynamic based techniques. The static based techniques are applied on a single image, while the dynamic based techniques are applied on video sequences. In below we give the recent work on the both static and dynamic techniques respectively.

The most used methods to differentiate between the real faces and the fake ones are based on texture analysis. Texture analysis counter-measures take advantage of texture patterns that may look unnatural when exploring the input image data. Examples of detectable texture patterns are printing failures or overall image blur. In [27], they described a method for print-attack detection by exploiting the differences in the 2-D Fourier spectra by comparing the hard-copies of client faces and the real accesses. The method work well for down-sampled photos of the attacked

identity, but is likely to fail for higher-quality samples. Li *et al.* [27] detected print-attacks by exploiting differences in the 2-D Fourier spectra of hard-copies of faces and real accesses. The method works well for down-sampled photo attacks, but is likely to fail for higher-quality samples. Bai *et al.* [39] analyzed the micro-textures using a linear SVM classifier to detect spoof attacks.

In [40, 41], the authors used the LBP as a descriptor to detect the spoof attack. Also in [42], the authors used other variant of the LBP descriptor which is Local Binary Patterns Variance (LBPV) that was used to differentiate between the real and the fake faces. Yang *et al.* [43] introduced a face recognition based on pooling the features extracted from the different face components using the Fisher criterion. Wen *et al.* [44] proposed a method based on Image Distortion Analysis (IDA). They used four different features: specular reflection, blurriness, chromatic moments and color diversity were used to represent the face images. These features can capture the differences between the real and the fake images without capturing the details information related to the user-identity. Patel *et al.* [45] studied the effect of the different channel of the RGB color spaces (R,G, B, and Gray Scale) and the different face regions on the performance of the LBP and Dense Scale Invariant Feature Transform (DSIFT) based methods. Their experiments show that extracting the texture from the red channel gives the best results. Boulkenafet *et al.* [46] proposed a method of face anti-spoofing based on color texture analysis. After representing the RGB images in two color spaces: Hue, Saturation, and Value (HSV) and Luminance; Chroma Blue; Chroma Red (YCbCr), they used the LBP descriptor to extract the texture features from each channel then they concatenated these features to differentiate between real and fake faces.

Galbally *et al.* [47] proposed an Image Quality Assessment (IQA) using 14 quality measures to distinguish between the real and the fake faces. In [48] the same authors evaluated 25 different quality measures, which were also used for fingerprint and iris anti-spoofing.

Recently, some methods such as [49, 50] used the user specific information to enhance the performance of the texture based face anti-spoofing methods. Biggio *et al.* [51] addressed the problem of spoof attacks on biometrics by using two modals which are face and fingerprint. They tested different score-fusion rules such as Sum, Product, Weighted sum by Linear Dis-

criminant Analysis (LDA), LikeLihood ratio (LLR) and Extended LikeLihood ratio (ELLR).

Motion analysis one is interested in detecting clues generated when two dimensional counterfeits are presented to the input camera system [40, 52, 53], for example photos or video clips. Kollreider *et al.* [26] evaluated the trajectories of selected part of the face from short sequence of images using a simplified optical flow analysis followed by a heuristic classifier. The same authors introduced a method [54] to fuse these scores with liveness properties such as eye-blinks or mouth movements. Bao *et al.* [55] proposed a method to detect attacks produced with planar media using optical flow based motion estimation.

Arashloo *et al.* [56] used kernel discriminant analysis fusion to combine two spatial temporal descriptors Multiscale Binarized Statistical Image Features on Three Orthogonal Planes (MBSIF-TOP) and Multiscale Local Phase Quantization on Three Orthogonal Planes (MLPQ-TOP). Pereira *et al.* [57] also worked with the dynamic texture based on Local Binary Patterns on Three Orthogonal Planes (LBP-TOP) to differentiate between real and fake person. This last method showed better performances compared to the simple LBP methods proposed in [40–42]. The reason behind the good results of LBP-TOP is that temporal information plays an important role in face anti-spoofing. Pinto *et al.* [58] proposed a method based on temporal and spectral information, which use the time-spectral features as low-level descriptors and use the visual codebook concept to find mid-level features descriptors. Tirunagari *et al.* [59] proposed an algorithm called Dynamic Mode Decomposition (DMD) to capture the visual dynamics while LBP is used to capture the dynamic patterns. Bharadwaj *et al.* [60] used the Eulerian motion magnification to enhance the motion cues. It was found that extracting Histogram of Oriented Optical Flow (HOOF) from the enhanced video yields an enhanced result with respect to the state-of-the-art results on the Replay-Attack database. Komulainen *et al.* [61] also used a fusion between the motion and the texture features to enhance the classification performances. Kollreider *et al.* [62] proposed novel strategies to avert advanced spoofing attempts such as replayed videos by analyzing the motion of the lips only. Liveness detection tries to capture signs of life from the user images by analyzing spontaneous movements that cannot be detected in photographs, such as eye blinks which are supposed to occur once every 2-4 seconds in humans [17, 63]. Pan *et al.* [64] exploited the observation that humans blink once every (2-4 s)

and proposed an eye blink-based anti-spoofing method.

Garcia *et al.* [65] proposed face spoofing detection by searching for Moiré patterns due to the overlap of the digital grids. Their detection is based on peak detection in the frequency domain. They used SVM with Radial Basis Function Kernel (RBF Kernel) for the classification. They conducted their experiments on Replay Attack Corpus and Moiré databases.

Other techniques in face anti-spoofing are based on textures using 3D modal such as [66, 67]. In 3D modal, the attacker uses a mask to spoof the system because of that the use of wrinkles would be a great assistant to detect the attack. In [66], they presented a study which addresses the spoofing issue by analyzing the feasibility to perform low-cost attacks with self-manufactured three-dimensional (3D) printed models to 2.5D and 3D face recognition systems. Erdogmus and Marcel [67] inspected the spoofing potential of subject-specific 3D facial masks for different recognition systems and address the detection problem of this more complex attack type. Also, the authors performed experiments on two different databases.

Recently, deep learning approaches have been used in face anti-spoofing, especially using Convolutional Neural Network (CNN). For instance, authors in [68] focused on two general-purpose approaches to build image-based anti-spoofing systems using convolutional networks. Their systems deal with several attack types in three biometric modalities, iris, face, and fingerprint. The first approach consists of learning suitable convolutional network architectures for each domain, while the second approach focuses on learning the weights of the network via back-propagation.

2.4 Face Spoofing Databases

The majority of face-spoofing databases provide only real access and spoofing attack samples for the clients. The directions for face anti-spoofing explored in this thesis are largely based on five publicly available databases, the NUAA Photo Imposter Database, the Replay-Attack Database, the CASIA Face Anti-Spoofing Database, the MSU Mobile Face Spoofing Database and OULU-NPU face PAD Database which are introduced in this section.

Table 2.1: A summary of published methods on face spoof detection

Athors	Methods	Data-bases	years
Kollreider *et al.* [62]	Motion	MIT–CMU YALE Recaptured	2007
Biggio *et al.* [51]	Multimodal	LivDet11 Photo Attack Personal Photo Attack Print Attack	2011
Chingovska *et al.* [41]	Texture	REPLAY ATTACK CASIA FAS NUAA photograph imposter	2012
Määttä *et al.* [40]	Texture	Yale Recaptured PRINT ATTACK	2012
Kose & Dugelay [42]	Texture	NUAA photograph imposter	2012
Erdogmus & and Marcel [67]	3D	Morpho 3D Mask Attack	2013
Yang *et al.* [43]	Texture	NUAA photograph imposter CASIA FAS PRINT ATTACK	2013
Komulainen *et al.* [61]	Motion	REPLAY ATTACK	2013
Galbally & Marcel [47]	Image Quality Assessment	CASIA FAS REPLAY ATTACK	2014
Galbally *et al.* [48]	Multimodal	Iris spoof, Iris-Synthetic LivDet REPLAY-ATTACK	2014
Bharadwaj *et al.* [60]	Motion	PRINT ATTACK REPLAY ATTACK CASIA FAS	2014
Pereira *et al.* [57]	Motion	REPLAY ATTACK CASIA FAS	2014
Menotti *et al.* [68]	Deep Learning	Warsaw, Biosec & MobBIOfake Replay-Attack & 3DMAD Biometrika, CrossMatch, Italdata & Swipe	2015
Garcia & Queiroz [65]	Moiré-Pattern	Replay Attack Moiré	2015
Yang *et al.* [49]	Person-Specific	CASIA FAS REPLAY ATTACK	2015
Chingovska & Anjos [50]	Person-Specific	REPLAY ATTACK	2015
Wen *et al.* [44]	Motion	REPLAY ATTACK CASIA FAS MSU MFS	2015
Pinto *et al.* [58]	Motion	CASIA FAS REPLAY ATTACK UVAD 3DMAD	2015
Tirunagari *et al.* [59]	Motion	PRINT ATTACK REPLAY ATTACK CASIA FAS	2015
Arashloo & Kittler [56]	Texture	REPLAY ATTACK CASIA FAS NUAA photograph imposter	2015
Boulkenafet *et al.* [46]	Colour Texture	CASIA FAS REPLAY ATTACK	2015
Patel *et al.* [45]	Colour Texture	REPLAY ATTACK CASIA FAS MSU MFS	2015
Galbally and Satta [66]	3D	3DFS-DB EURECOM MASK-ATTACK DB IDIAP MASK-ATTACK DB	2016

2.4.1 NUAA Photo Imposter Database

The NUAA Photograph Imposter Database [1] (PID) is proposed by Tan *et al.* [69] can be considered as the first publicly available spoofing database because the evaluation is based on binary classification task of differentiating genuine faces from fake ones with a predefined protocol. The dataset contains images of both real client accesses and photo attacks using both photo-quality and laser-quality prints that were collected in three sessions at intervals of about two weeks. During each session, the environmental and illumination conditions were different. Examples of cropped facial images from the database can be seen in Figure. 2.5. The client accesses and spoofing attacks were recorded using a generic webcam with resolution of 640×480 pixels and altogether there are about 500 images (20fps) for each subject's recording. When capturing the data, the main idea was to make the live subjects look like a static as much as possible by minimizing the movements and the eye-blinking, i.e. resembling a photograph. In contrast, five different photo-attacks were simulated using 2D facial prints with varying motions. The high-quality photos of the targeted person were printed on photographic paper of two sizes $6.8cm \times 10.2cm$ (small) and $8.9cm \times 12.7cm$ (big) using a traditional development method, or on a 70g white A4 paper using a conventional Hewlet-Packard color printer. Unfortunately, the printing option is not included in the metadata of the database.

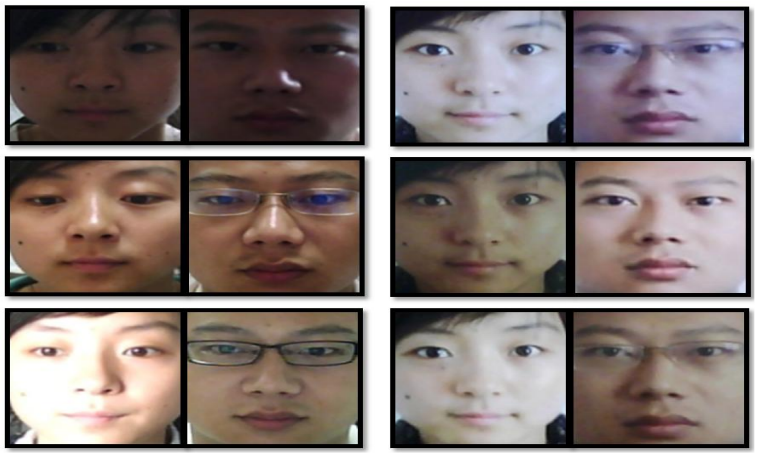


Figure 2.5: Illustration of the samples from the database. In each column (from top to bottom) samples are respectively from session 1, session 2 and session 3. In each row, the left pair are from a live human and the right from a photo.

[1] http://parnec.nuaa.edu.cn/xtan/data/nuaaimposterdb.html

The dataset is composed of images of fifteen subjects (in three sessions for most of the subjects) that are decomposed into two separate sets for training and testing purposes. The training set consists of images from the first two sessions only. The test set consists of the images from the remaining third session. The training set contains altogether 1,743 face images of nine real clients (889 and 854 from the first and the second sessions, respectively) and 1,748 imposter images of the same nine clients (855 and 893 images from the first and the second sessions, respectively). The test set is constructed from 3,362 client samples and 5,761 imposter images taken during the third session. Only three clients who took part in the first two sessions attended the third session.

Furthermore, six new clients and their photographs are introduced in the test set to further increase the level of difficulty. There is no specific development set provided in the database, thus cross-validation or fixed validation set has to be used for tuning the algorithms. The database contains also the data needed for face normalization and the geometrically normalized face images of 64×64 pixels which were used in the experiments by Tan *et al.* [69], thus making it easier to compare the results between different spoof detection techniques.

2.4.2 Replay-Attack Database

The Replay-Attack Database [2] is proposed by Chingovska *et al.* [41] and its subsets (the Print-Attack Database [3] [52] and the Photo-Attack Database [70]) consist of short video recordings (roughly ten seconds) of both real accesses and corresponding attack attempts. The studied attack scenarios in the dataset can be categorized based on display media, A4 sized hard copy (print), iPhone 3GS (mobile) and iPad with a resolution of 1024×768 (high def) and two attack types, photo and video. Also two illumination conditions are introduced: controlled with uniform background scene and fluorescent lamp illumination and adverse with non-uniform background scene and day-light illumination. The videos clips were captured using an Apple 13-inch MacBook laptop and its embedded webcam with a relatively low-quality resolution of 320×240 pixels (QVGA) at 25 fps.

[2] https://www.idiap.ch/dataset/replayattack
[3] https://www.idiap.ch/dataset/printattack

When recording the real client accesses of fifteen seconds, the subjects were asked to look at the laptop camera as during normal authentication process. Unlike in traditional laptop authentication scenario, the laptop was placed on top of a small stand in order to capture frontal-pose faces. For creating the attacks, two photographs and two video clips were taken of each person in each of the two illumination and background settings used for recording the real accesses. The first photograph/video clip was recorded using iPhone 3GS (3.1 megapixel camera) and the second using a high-resolution 12.1 megapixel Canon Power-Shot SX200 IS camera. To maximize the attack quality, the subjects were asked to look up-front like in the case real access attempts. Furthermore, each spoofing attack video clip of ten seconds was recorded with two different support modes, hand-held and fixed-support. Figure 2.6 shows examples of the genuine and attack samples in the different conditions explored by the Replay Attack Database.

Figure 2.6: Examples from the Replay-Attack Database. The first row presents images taken from the controlled scenario, while the second row corresponds to the images from the adverse scenario. From the left to the right: real faces and hand video, hand photo, fixed video and fixed photo.

In total, the Replay-Attack Database contains 50 different identities and 1,300 video clips of which 300 correspond to real-accesses (three trials in two different conditions for each of client. The first trial for each subject is dedicated solely for evaluating face verification systems, i.e. not used for evaluating anti-spoofing performance. The remaining 200 real-accesses and 1,000 attack video clips are divided into training, development and test sets (360, 360 and 480 videos,

respectively) for evaluating the binary spoof detection classifiers. The subject-disjoint subsets were randomly selected, i.e. identities that are on one of the subsets do not appear in any other set. Thus, the anti-spoofing models are not trained for detecting person-specific appearance or facial dynamics. In order to enable system-level evaluation, i.e. the joint operation of recognition and anti-spoofing algorithm, the identities between the verification protocol and anti-spoofing protocols match. The dataset provides eighteen protocols for evaluating the effectiveness of the anti-spoofing methods under different conditions, including support, fake face type and quality.

The training set is used for training the countermeasure, whereas the development set operates as a separate validation set for estimating a threshold value to be used on the test set. The database protocol defines the EER as a decision threshold. The actual test set is used only to report results. As a performance measure, the protocol suggests reporting the HTER on the test data. The dataset provides also automatically annotated face bounding boxes for convenience.

2.4.3 CASIA Face Anti-Spoofing Database

The CASIA Face Anti-Spoofing Database [4] (FASD) is proposed by Zhang *et al.* [71] introduces some significant improvements to previous databases because it provides more variations in the collected data. The authors indicated that imaging quality of different cameras is an important factor that may influence the robustness of anti-spoofing techniques, especially methods analyzing the facial texture. Thus, the database contains data from 50 real clients and the corresponding forged samples collected using three different devices with varying quality, old webcam (low quality) with resolution of 480×640, new webcam (normal quality) both with resolution of 640×480 and a Sony NEX-5 digital system camera with a resolution of 1920×1080 (high-quality). However, in order to save memory and computational burden, the original 1920×1080 resolution videos have been cropped into patches of 1280×720 pixels which contain only the face region, thus maximizing the appearance quality of the target faces. Example images of a genuine face at the different imaging qualities can be seen in Figure 2.7.

Both real client accesses and the corresponding attack attempts are captured in natural office

[4] http://www.cbsr.ia.ac.cn/english/FaceAntiSpoofDatabases.asp

Figure 2.7: Samples showing the different imaging qualities (low, normal and high, respectively) extracted from the CASIA FASD.

scenes. The subjects are required to exhibit eye blinking during data capture as the authors argue that motion is crucial cue for face spoof detection, thus it is important to provide motion like in challenge-response based systems. The attack scenarios in the dataset are based on three types of fake faces which include warped photo, cut photo (photographic mask) and video attacks. The high-quality samples of the targeted faces were generated from the videos captured with the Sony NEX-5. The facial prints were printed on copper paper in order to achieve better quality spoofs compared to conventional A4 printing paper and to avoid printing artefacts that are very obvious in the Replay-Attack Database [41]. The warped photo attacks were performed like in the work by [17, 54, 69]. The eye regions were cut off in order to create photographic masks and eye blinking was simulated either by the attacker or by sliding another piece of paper behind the resulting cut photo (See Figure 2.8). The video attacks were executed using iPad with a screen resolution of 1024×768, thus the original high resolution of 1280×720 is downsized compared to the photo attacks (See Figure 2.8).

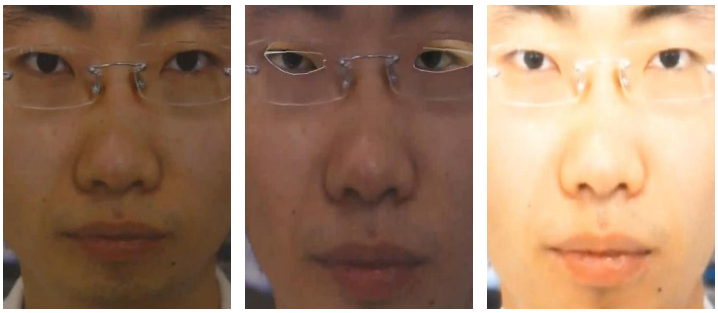


Figure 2.8: Samples showing the different media attacks (warped, cut and video, respectively) extracted from the CASIA FASD.

Altogether the database consists of 600 video clips and the identities are divided into subject-

disjoint subsets for training and testing (240 and 360, respectively). Since the main purpose of the database is to investigate the possible effects of different fake face types and imaging qualities, the test protocol consists of seven scenarios in which particular train and test samples are to be used. The quality test considers the three imaging qualities separately, low (1), normal (2) and high quality (3), and evaluates the overall spoof detection performance under a variety of attacks at the given imaging quality. Similarly, the fake face test assesses how robust the anti-spoofing measure is to specific fake face attacks, warped photo (4), cut photo (5) and video attacks (6), regardless of the imaging quality. In the overall test (7), all data is used to give a more general evaluation. Examples of the different scenarios in the database can be seen in Figure 2.9. The results of each scenario are reported as DET curves and EER. Results of a baseline system are also provided along the database. Inspired by the work by [69], the baseline system considers the high frequency information in the facial region using multiple Difference of Gaussian (DoG) features and SVM classifier.

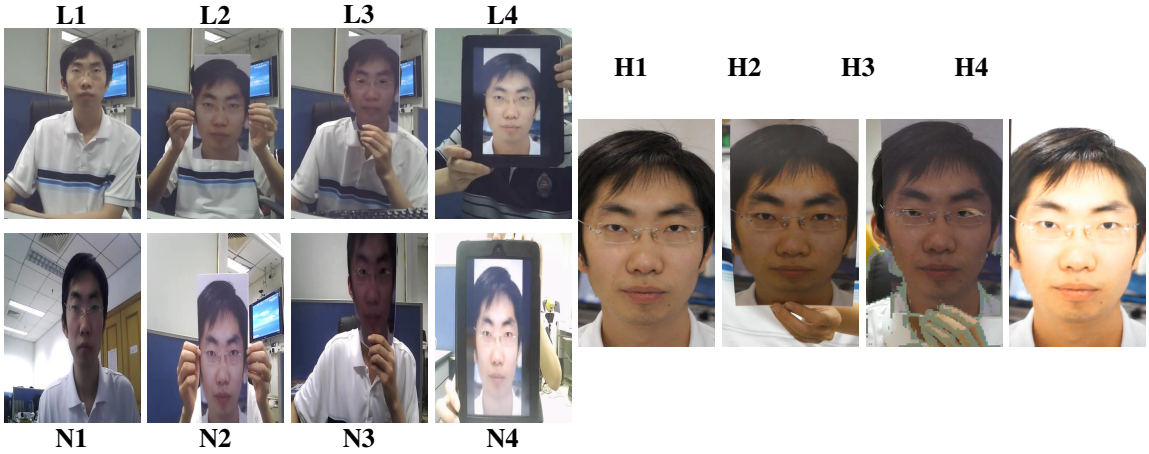


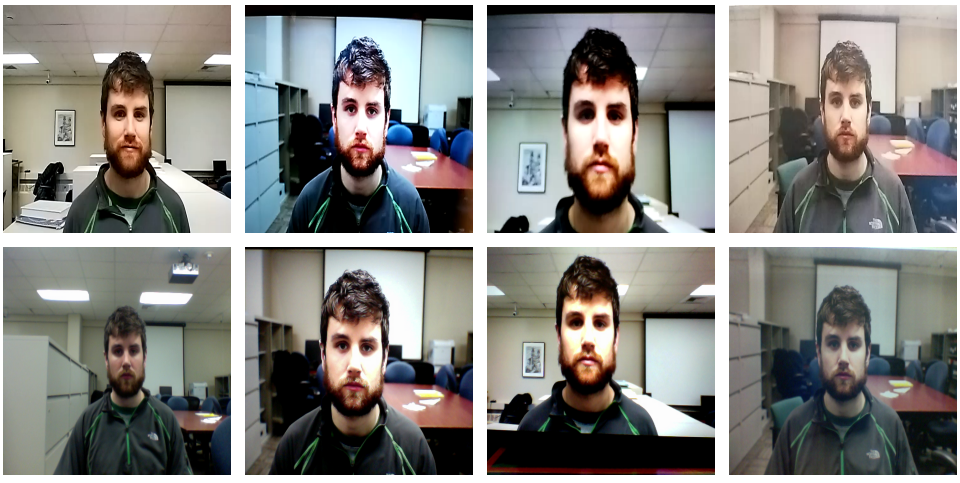
Figure 2.9: Samples from the CASIA face anti-spoofing database. L, N and H for Low, Normal and High quality, respectively. 1, 2, 3 and 4 for real face, warped photo, cut photo and video attacks, respectively.

2.4.4 MSU Mobile Face Spoofing Database

The publicly available MSU Mobile Face Spoof Database [5] (MFSD) is proposed by Wen *et al.* [44] for face spoof attacks was produced at the Michigan State University by the Patterns

[5] http://biometrics.cse.msu.edu/Publications/Databases/MSUMobileFaceSpoofing/index.htm

Recognition and Image Processing (PRIP) group. The database consists of 280 video clips of photo and video attack attempts to 35 clients. It was made by mobile phone to capture both genuine face and spoof attacks. They used two types of cameras: 1) built-in camera in MacBook Air 13 inch (640×480) and 2) front-facing camera of the Google Nexus 5 Android phone (720×480). Each subject had two videos recordings, the first one is captured by Laptop camera and the second one is captured using Android camera (See Figure 2.10). To generate the attacks, high-resolution video was captured for each subject using two devices: 1) Canon Power Shot 550D SLR camera, recording 18.0M pixel photographs and 1080p high-definition video clips, 2) iPhone 5S back-facing camera, recording 1080p video clips. There are three types of spoof attack, the first one 1) high-resolution replay video attacks using an iPad Air screen, with resolution of 2048×1536, the second 2) mobile phone replay video attacks using an iPhone 5S screen, with resolution of 1136×640, and the last 3) printed photo attacks using an A3 paper with fully-occupied printed photo of the client's biometry, with paper size of : $11' \times 17'$ (279mm x 432 mm), printed by a HP Colour Laserjet CP6015xh printer, with printing resolution of 1200×600 dpi. In the last, to evaluate the performance, the 35 subjects of MSU MFSD database were divided into two subsets, 15 subject for training and 20 subject for testing.



(a) **(b)** **(c)** **(d)**

Figure 2.10: Example images of genuine and spoof faces of one of the subjects in the MSU MFSD database captured using Google Nexus 5 smart phone camera (top row) and MacBook Air 13" laptop camera (bottom row). (a) Genuine faces; (b) Spoof faces generated by iPad for video replay attack; (c) Spoof faces generated by iPhone for video replay attack; (d) Spoof faces generated for printed photo attack.

Figure 2.11: Samples of the subjects recorded in the database.

2.4.5 OULU-NPU face PAD Database

The aim of the dataset is particularly at evaluating the generalization of new PAD methods in more realistic mobile authentication scenarios by considering three covariates: unknown environmental conditions (namely illumination and background scene), acquisition devices and Presentation Attack Instruments (PAI), separately and at once. In the following, we describe the new OULU-NPU face PAD database and its evaluation protocols in detail.

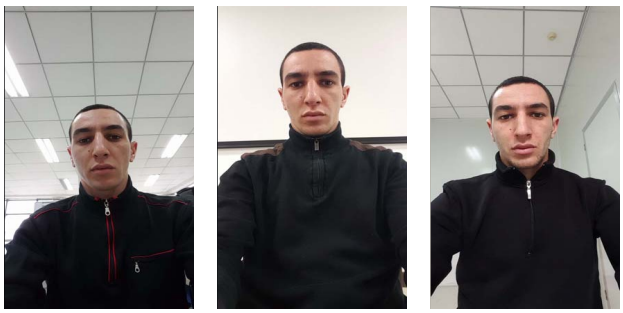
2.4.5.A Collection of real access attempts

The OULU-NPU presentation attack detection database [6] is proposed by Boulkenafet *et al.* [72] includes short video sequences of real access and attack attempts corresponding to 55 subjects (15 female and 40 male). Figure 2.11 shows samples of these subjects. The real access attempts were recorded in three different sessions separated by a time interval of one week. During each session, a different illumination condition and background scene were considered (See Figure2.12):

- Session 1: The recordings were taken in an open-plan office where the electronic light was switched on and the windows blinds were up and the windows were located behind the users.

- Session 2: The recordings were taken in a meeting room where the electronic light was the only source of illumination.

[6] https://sites.google.com/site/oulunpudatabase/



(a) Session 1 **(b)** Session 2 **(c)** Session 3

Figure 2.12: Sample images of a real subject highlighting the illumination conditions across the three different scenarios.

- Session 3: The recordings were taken in a small office where the electronic light was switched on and the windows blinds were up and the windows were located in front of the users.

During each session, the subjects recorded two videos of themselves (one for the enrollment and one for the actual access attempt) using the frontal cameras of the mobile devices. In order to simulate realistic mobile authentication scenarios, the video length was limited to five seconds and the clients were asked to hold the mobile device like they were being authenticated but without deviating too much from their natural posture while normal device usage.

The recent advances in sensor technology have introduced high-resolution cameras also to the mid-range models of the last generation mobile devices capable of capturing good quality images (and videos) in daylight and indoor conditions. Considering that the acquisition quality of the embedded (both front and rear) cameras can be expected to be growing generation by generation, we selected six smartphones with high-quality front-facing cameras in price range from 250€ to 600€ for the data collection:

- Samsung Galaxy S6 edge (Phone 1) with 5 MP frontal camera.
- HTC Desire EYE (Phone 2) with 13 MP frontal camera.
- MEIZU X5 (Phone 3) with 5 MP frontal camera.
- ASUS Zenfone Selfie (Phone 4) with 13 MP frontal camera.

(a) Samsung **(b)** HTC **(c)** MEIZU **(d)** ASUS **(e)** Sony **(f)** OPPO

Figure 2.13: Sample images showing the image quality of the different camera devices.

- Sony XPERIA C5 Ultra Dual (Phone 5) with 13 MP frontal camera.
- OPPO N3 (Phone 6) with 16 MP rotating camera.

The videos were recorded at Full HD resolution, i.e. 1920×1080 using the frontal cameras of the six mobile devices and the same camera software installed on each device. Even though the nominal camera resolution of some mobile devices is the same, like Sony XPERIA C5 Ultra Dual, HTC Desire EYE and ASUS Zenfone Selfie (13 MP), significant differences can be observed in the quality of the resulting videos as demonstrated in Figure 2.13.

2.4.5.B Attack creation

Assuming that the legitimate users are trying to get authenticated in multiple conditions, it is important to collect the data of genuine subjects in multiple lighting conditions from the usability point of view. In contrast, the attackers try to present as high-quality artifact as they can to the input camera in order to maximize the chance of successfully fooling a face biometric system. Therefore, the attacks should be carefully designed and conducted in order to guarantee that they are indeed hard to detect.

During each of the three sessions, a high-resolution photo and video of each user was captured using the back camera of the Samsung Galaxy S6 Edge phone capable of taking 16 MP still images and Full HD videos. These high resolution photos and videos were then used to create the presentation attacks. The attack types considered in this database are print and video-replay attacks:

(a) Print 1 **(b)** Print 2 **(c)** Print 3 **(d)** Print 4

Figure 2.14: Samples of print and replay attacks taken with the front camera of Sony XPERIA C5 Ultra Dual.

- Print attacks: The high resolution photos were printed on A3 glossy paper using two different printers: a Canon imagePRESS C6011 (Printer 1) and a Canon PIXMA iX6550 (Printer 2).

- Video-replay attacks: The high-resolution videos were replayed on two different display devices: a 19" Dell UltraSharp 1905FP display with 1280×1024 resolution (Display 1) and an early 2015 Macbook 13" laptop with Retina display of 2560×1600 resolution (Display 2).

The print and video-replay attacks were then recorded using the frontal cameras of the six mobile phones. While capturing the print attacks, the facial prints were held by the operator and captured with stationary capturing devices in order to maximize the image quality but still introduce some noticeable motion in the print attacks. In contrast, when recording the video-replay attacks both of the capturing devices and PAIs were stationary. Furthermore, we paid special attention that the background scene of the attacks matches the real accesses during each session and that the attack videos do not contain the bezels of the screens or edges of the prints. Figure 2.14 shows samples of the attacks captured using the Sony XPERIA C5 Ultra Dual.

2.4.5.C Evaluation protocols

To evaluate the performances of the face PAD methods on the OULU-NPU database, we designed four protocols.

- Protocol I: The first protocol is designed to evaluate the generalization of the face PAD methods under different environmental conditions, namely illumination and background scene. As the data is recorded in three sessions with different illumination conditions and locations, the train, development and evaluation sets can be constructed using video recordings taken from different sessions, See Table 2.2.

- Protocol II: Since different PAI (i.e. different displays and printers) create different artifacts, it is necessary to develop face PAD methods robust to this kind of variations. The second protocol is designed to evaluate the effect of the PAI variation on the performance of the face PAD methods by introducing previously unseen PAI in the test set as shown in Table 2.2.

- Protocol III: One of the critical issues in face anti-spoofing and image classification in general is the generalization across different acquisition devices. A Leave One Camera Out (LOCO) protocol is designed to study the sensor interoperability of the face PAD methods. In each iteration, the real and the attack videos recorded with five smartphones are used to train and tune the countermeasure model. Then, the generalization of the method is assessed using the videos recorded with the remaining smartphone.

- Protocol IV: In the last and most challenging scenario, the previous three protocols are combined to simulate the real-world operational conditions. To be more specific, the generalization abilities of the face PAD methods are evaluated simultaneously across previously unseen illumination conditions, background scenes, PAIs and input sensors, Table 2.2.

In all these protocols, the 55 subjects were divided into three subject-disjoint subsets for training, development and testing (20, 15 and 20, respectively). Tables 2.2 gives a detailed information about the video recordings used in the train, development and test sets of each protocol.

For the performance evaluation, they selected the recently standardized ISO/IEC 30107-3 metrics [71], Attack Presentation Classification Error Rate (APCER) and BonaFide Presentation

Table 2.2: The detailed information about the video recordings in the train, development and test sets of each protocol.

Protocol	Subset	Session	Phones	Users	Attacks created using	# real videos	# attack videos	# all videos
Protocol I	Train	Session 1,2	6 Phones	1-20	Printer 1,2; Display 1,2	240	960	1200
	Dev	Session 1,2	6 Phones	21-35	Printer 1,2; Display 1,2	180	720	900
	Test	Session 3	6 Phones	36-55	Printer 1,2; Display 1,2	240	960	1200
Protocol II	Train	Session 1,2,3	6 Phones	1-20	Printer 1; Display 1	360	720	1080
	Dev	Session 1,2,3	6 Phones	21-35	Printer 1; Display 1	270	540	810
	Test	Session 1,2,3	6 Phones	36-55	Printer 2; Display 2	360	720	1080
Protocol III	Train	Session 1,2,3	5 Phones	1-20	Printer 1,2; Display 1,2	300	1200	1500
	Dev	Session 1,2,3	5 Phones	21-35	Printer 1,2; Display 1,2	225	900	1125
	Test	Session 1,2,3	1 Phones	36-55	Printer 1,2; Display 1,2	60	240	300
Protocol VI	Train	Session 1,2	5 Phones	1-20	Printer 1; Display 1	200	400	600
	Dev	Session 1,2	5 Phones	21-35	Printer 1; Display 1	150	300	450
	Test	Session 3	1 Phones	36-55	Printer 2; Display 2	20	40	60

Classification Error Rate (BPCER):

$$APCER_{PAI} = \frac{1}{N_{PAI}} \sum_{i=1}^{N_{PAI}} (1 - Res_i) \tag{2.1}$$

$$BPCER = \frac{\sum_{i=1}^{N_{BF}} (Res_i)}{N_{BF}} \tag{2.2}$$

Where, N_{PAI}, is the number of the attack presentations for the given PAI,N_{BF} is the total number of the bona fide presentations. Res_i takes the value 1 if the ith presentation is classified as an attack presentation and 0 if classified as bona fide presentation. These two metrics correspond to the FAR and FRR commonly used in the PAD related literature. However, $APCER_{PAI}$ is computed separately for each PAI (e.g. print or display) and the overall PAD performance corresponds to the attack with the highest APCER, i.e. the "worst case scenario".

2.5 Conclusion

In this chapter, we started with a short introduction and introduction to general biometrics and face recognition in face anti-spoofing which is analyzed in two parts as face spoofing and face anti-spoof. Next, we presented the state-of-the-art. Since in this thesis the variations due to facial alterations are analyzed, we presented an in-depth description for the current state of these variations. Then, to the best of our knowledge, we gave a brief description of 5 pub-

licly available face spoofing databases, differing in the data format, the number of clients and samples, protocol, types of attacks, as well as the quality of the recording devices.

3

Face Anti-Spoofing Methods

Contents

3.1 Introduction

There are several types of spoofing attacks such as photograph, video or mask attacks. In our study, we first analyzed photograph attacks. This preliminary study helped us to gain an insight on the topic of face spoofing and countermeasures. Various approaches have been developed to detect photograph spoofing. The existing techniques mainly concentrate on liveness detection and motion analysis. There are also several countermeasure techniques based on texture analysis, which can be applied on single images. Face anti-spoofing can be divided in four main components are: face preprocessing, feature extraction, face representation, features selection and classification.

3.2 Face preprocessing

Preprocessing of facial image data is very important part of face recognition. Also face is one of the most common parts used by people to recognize each other. But the last one it is easily affected by light condition and facial expression changing and other reasons [73]. So before extracting features we can preprocess face images to improve the face recognition rate. Finally we explain how we use preprocessing step by step which are: face detection, eyes localization and face normalization.

3.2.1 Face detection

Face detection is an important step in many computer vision systems, like face spoofing in our works. So for this reason we use VJ algorithm [74] in our test to detect the ROI which is the face in our case (See Figure 3.1). The characteristics of VJ algorithm which make it a good detection algorithm are [75]:

- Robust: very high detection rate (true-positive rate) and very low false-positive rate always.
- Real time: VJ can be used in real time application.

- Face detection only (not recognition): The goal is to distinguish faces from non-faces (detection is the first step in the recognition process).

The algorithm of VJ has four stages [75]:

- Haar Feature Selection
- Creating an Integral Image
- Adaboost Training
- Cascading Classifiers

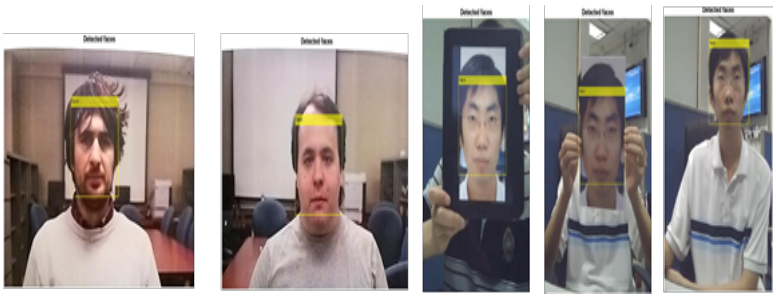


Figure 3.1: Example face detection using VJ

3.2.2 Eyes localization

The importance of detecting and localizing eyes position in face is decisive for the initialization of many application like face recognition, face expression, face anti-spoofing, age estimation, gender classification, etc. The difference between eyes detection and eyes localization is that the last one are given more accurate prediction of the eye positions than the first one. The algorithm which we use in our test is called PS [76] model for precise eye localization that's is good for face images taken under uncontrolled conditions, but can't localize the eyes directly. We must detect the face first and resize it into (100×100). This is the reason why we use VJ algorithm (See Figure 3.2).

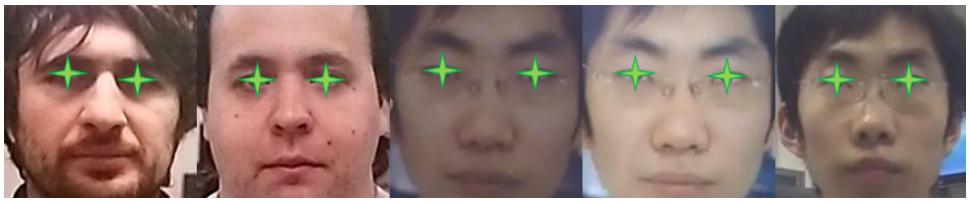
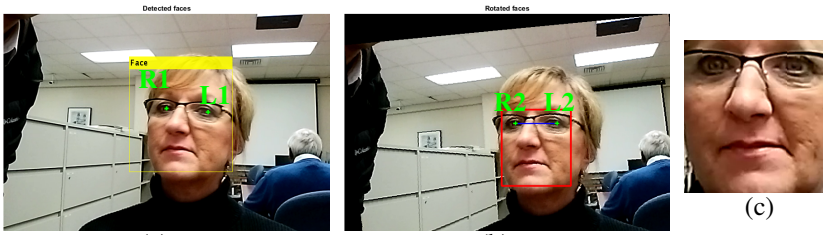


Figure 3.2: Example eye localization by PS algorithm.

3.2.3 Face normalization

After face detection and eye localization, we must normalize the face. In face normalization, we rotate and crop the face depending on the coordinate of eyes (See Figure 3.3) which was obtained by eye localization algorithm [76]. In the figure 3.4 and equations below we try to explain how to rotate and crop the face using the coordinates of eyes. Then after rotate and crop the face we resize the ROI.



(a) (b) (c)

Figure 3.3: Example of face alignment. a) face & eyes detection b) pose correction c) face ROI.

$$
\begin{aligned}
L1_x &= (L_x \times (m/100) + bbox(1), L_y \times (m/100) + bbox(2)) \\
R1_x &= (R_x \times (m/100) + bbox(1), R_y \times (m/100) + bbox(2))
\end{aligned}
\tag{3.1}
$$

$$
\theta = tan^{-1}(\frac{R1_y - L1_y}{R1_x - L1_x}) \tag{3.2}
$$

$$
\begin{aligned}
L2_x &= C_x + (L1_x - C_x).cos(\theta) - (L1_y - C_y).sin(\theta) \\
L2_y &= C_y + (L1_x - C_x).sin(\theta) + (L1_y - C_y).cos(\theta) \\
R2_x &= C_x + (R1_x - C_x).cos(\theta) - (R1_y - C_y).sin(\theta) \\
R2_y &= C_y + (R1_x - C_x).sin(\theta) + (R1_y - C_y).cos(\theta)
\end{aligned}
\tag{3.3}
$$

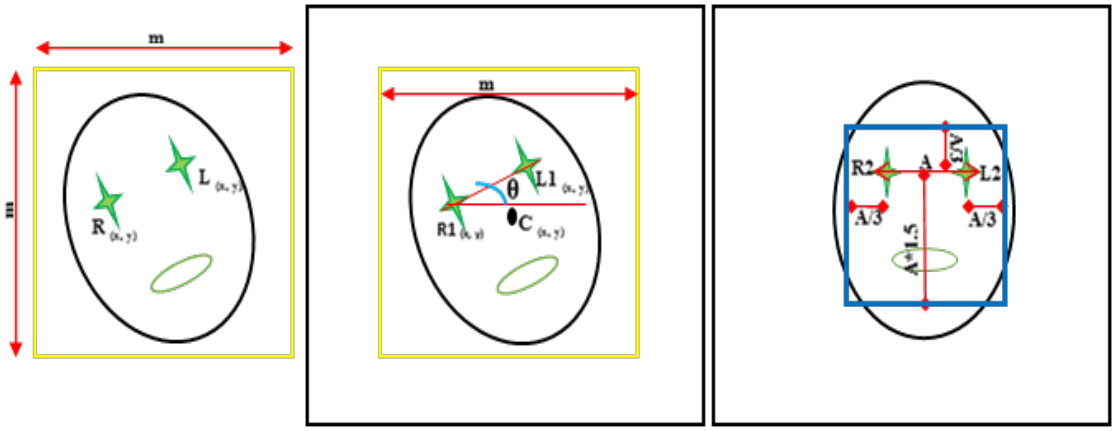


Figure 3.4: Detail of rotate & crop of face

3.3 Features extraction

We studied three famous descriptors in texture that can be applied for extracting the features from face spoofing challenges, motivated by its invariance and respect to monotonic gray scale transformations. Those descriptors are: LBP, LPQ, and BSIF. We explain below those descriptors in the details.

3.3.1 Local Binary Pattern

Local Binary Pattern is a texture descriptor proposed by Ojala *et al.* [77], it is power-full in extracting the features from image to give a one vector in each image. The descriptor is given by calculating the difference between a pixel of an image by threshold a 3×3 neighborhood

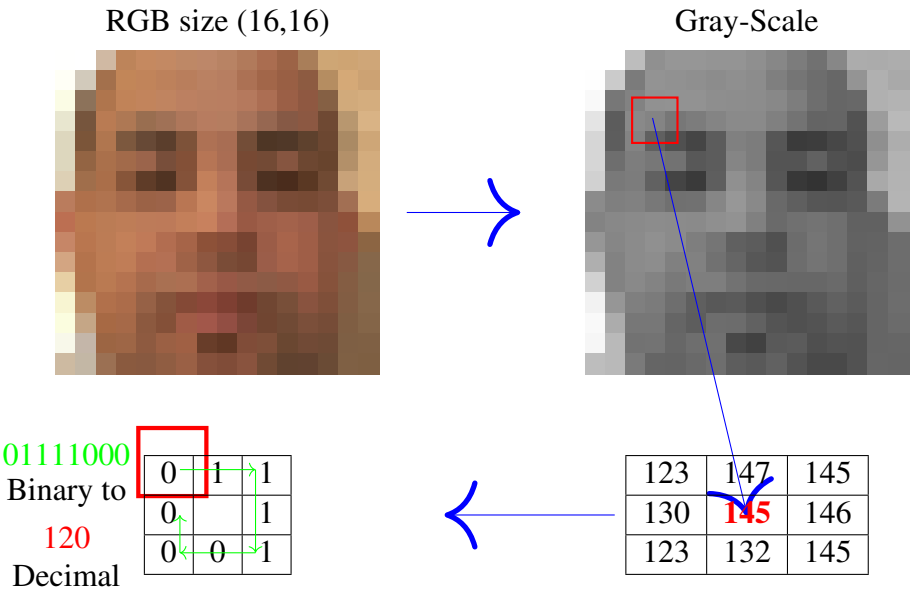


Figure 3.5: An example of Local Binary Pattern

and the center pixel value with the condition to gives 0 or 1. With those binary number are gives as direction clockwise we must convert to decimal number like figure 3.5.

The first LBP descriptor is used with (3×3) neighborhoods like in figure 3.5. In another term, given value of each position of pixel (x_c, y_c). The LBP features is obtained by comparison between center pixel (p_c) and the around pixels (p_n), n(0,1,..7). In below we give the expression with equation 3.4

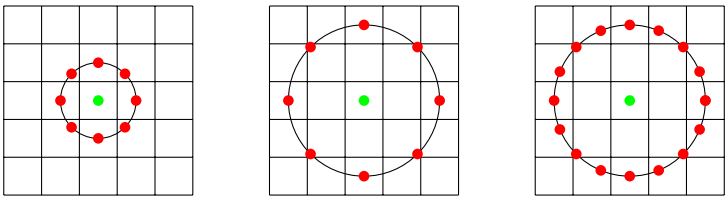
$$LBP(x_c, y_c) = \sum_{n=0}^{7} S(p_n - p_c)2^n \tag{3.4}$$

Where:

- p_c: center pixel which is the value of pixel (x_c, y_c)
- p_n: values of around center pixel p_c
- $S(X)$: is defined as:

$$S(X) = \begin{cases} 1 & \text{if } X \geq 0 \\ 0 & otherwise. \end{cases} \tag{3.5}$$

- X : is $p_n - p_c$



(8,1) (8,2) (16,2)

Figure 3.6: ELBP different sample points (P) and radius (R)

There is another variation of the original LBP called Extended Local Binary Pattern (ELBP) [78]. The ELBP is working with a different size on neighborhoods, using circular neighborhoods and bilinear interpolation of pixel values. The equation 3.6 defined the ELBP:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{p-1} S(g_p - g_c)2^p \tag{3.6}$$

We give some examples in below with $ELBP_{P,R}$ which are: $ELBP_{(8,1)}$, $ELBP_{(8,2)}$ and $ELBP_{(16,2)}$. Where, P is sampling points on a circle and R is the radius.

After described the $ELBP_{P,R}$ method, there is another extension of original LBP called Local Binary Pattern uniform [79] which defined by equation 3.7:

$$LBP^{u2}_{P,R}(x_c, y_c) = \sum_{p=0}^{p-1} S(g_p - g_c)2^p \tag{3.7}$$

The LBP uniform called with this name uniform if only have at least two transition from 0 to 1 or vice versa. We give an example to explain the uniform pattern, like 00000000 have zero transition, 11111111 zero transition, 00111000 two transition and 00001111 one transition.

In another hand, when there is more than two transition is non-uniform like 00110011 have three transitions, 01010101 seven transitions, 00110101 five transitions and 10101111 four transition. In all uniform LBP mapping there is a separate label on each output given by convert binary to decimal however non-uniform LBP have one label is 58 of 8 sampling points and 242 of 16 sampling points. Finally, the number for mapping for pattern of the LBP with P sample point is given by this relation $P(P-1)+3$. In the case of P = 8 it is has 59 bins and for P = 16 it is has 243 bins. Also, each LBP code can be considered as micro texture which are: Spot,

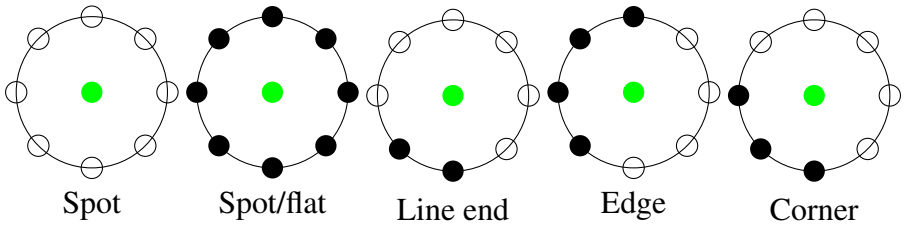


Figure 3.7: Different texture primitives detected by the LBP

Spot/flat, Line end, Edge and in the last Corner. We give an example for those micro-texture in figure 3.7.

3.3.2 Local Phase Quantization

The local phase quantization is a texture descriptor proposed by Ojansivu *et al.* [80]. Also It was robust blur and outperform for texture classification compared others descriptors. In below we explain the importance line of local phase quantization.

The spatial blurring is given by convolution between two matrices which are the image intensity and a Point Spread Function (PSF), like equation 3.8 below:

$$g(x) = (f * h)(x) \tag{3.8}$$

Where:

- g(x) is the blurred image
- f(x) is the true image
- h(x) is point spread function (PSF)
- x is a vector of coordinates $[x, y]^T$

The equation 3.8 transformed in the frequency domain with the discrete Fourier transforms (DFT) as $g(x)$ is $G(u)$, $f(x)$ is $F(u)$ and $h(x)$ is $H(u)$ and u is a vector of coordinates $[u, v]^T$ in the last the equation 3.9 is given by :

$$G(u) = F(u).H(u) \tag{3.9}$$

the equation 3.9 is given also by two term are magnitude and phase

$$\begin{aligned} |G(u)| &= |F(u)| \,.\, |H(u)| \\ \underline{/G}(u) &= \underline{/F}(u) + \underline{/H}(u) \end{aligned} \tag{3.10}$$

Always H is real value and $\underline{/H}$ is equal to (0 or π) if the PSF is centrally symmetric. Furthermore, the shape of H for a regular PSF is close to a Gaussian or a sinc-function ensuring that at least the low frequency values of H are positive. At these frequencies, $\underline{/H} = 0$ causing $\underline{/F}$ to be a blur invariant property. Because LPQ uses finite size 2-D discrete Short-Time Fourier Transform (STFT) computed locally, this invariance is in part disturbed but is still pertinent.

In LPQ, the phase is examined in local neighborhoods N_x at each pixel position $x = [x_1, x_2]^T$ of the image $f(x)$. These local spectra are computed using a discrete STFT defined by:

$$F(u, x) = \sum_y f(y) w_R(y - x) e^{-2j\pi u^T y} \tag{3.11}$$

Where u is the frequency, and $w(x)$ is a window function defining the neighborhood N_x. In the case of regular LPQ, w_R is a $N_R - by - N_R$ rectangle given as $w_R(x) = 1$ if $|x_1|, |x_2| < N/2$ and 0 otherwise.

The local Fourier coefficients are computed at four frequency points $u_1 = [a, 0]^T$, $u_2 = [0, a]^T$, $u_3 = [a, a]^T$, and $u_4 = [a, -a]^T$, where a is a sufficiently small scalar to satisfy $H(u_i) > 0$. For each pixel position this results in a vector

$$F(x) = [F(u_1, x), F(u_2, x), F(u_3, x), F(u_4, x)] \tag{3.12}$$

The phase information in the Fourier coefficients is recorded by observing the signs of the real and imaginary parts of each component in $F(x)$. This is done by using a simple scalar quantization

$$q_i = \begin{cases} 1 & \text{if } g_i \geq 0 \\ 0 & otherwise. \end{cases} \tag{3.13}$$

Where g_j is the $j-th$ component of the vector $G(x) = [ReF(x), ImF(x)]$.

The resulting eight binary coefficients q are represented as integer values between $0-255$ using coding $f_{LPQ}\ (x) = \sum_{j=1}^{8}\ q_j 2^{j-1}$. Finally, a histogram of these values from all positions is composed, and used as a 256-dimensional feature vector in classification.

3.3.3 Binarized Statistical Image Features

Binarized Statistical Image Features is a descriptor of texture proposed by Kannala *et al.* [81]. The BSIF represent by a binary code string for the pixels of a given image. The code value of a pixel is considered as a local descriptor of the image into pixels surroundings.

Given an image patch X of size $(l \times l)$ pixels and a linear filter W_i of the same size, the filter response s_i is obtained by :

$$s_i = \sum_{u,v} W_i(u,v)X(u,v) = W_i^T x \tag{3.14}$$

Where vector notation is introduced in the latter stage. If we have n linear filters W_i, we may stack them to a matrix W and compute all responses at once:

$$S = W_x \tag{3.15}$$

Given a random sample of natural image patches, we determine the filters W_i so that the elements s_i of s are as independent as possible when considered as random variables [81]. The binary code string b, which corresponds to image patch x, is obtained by binarizing each element s_i of s as follows:

$$b_i = \begin{cases} 1 & \text{if } s_i > 0 \\ 0 & otherwise. \end{cases} \tag{3.16}$$

Where b_i is the i^{th} element of b.

In this manner one may compute a $n-bit$ binary code string b for each pixel. As described above, there are two parameters in BSIF descriptor: the filter size l and the length n of the bit string.

3.4 Face representation

Face representation is a technique which divides the face ROI into blocks to get more details of the face. We apply then, any descriptor before using this technique. In below we explain two technique which are: MB and ML.

3.4.1 Multi-Blocks face representation (MB)

MB is a technique which divide the face ROI into $(n \times n)$ sub-blocks. We apply one of our descriptors and divide the face with MB to give us more features of the face ROI. Figure 3.8 shows how we divided a face in multi blocks.

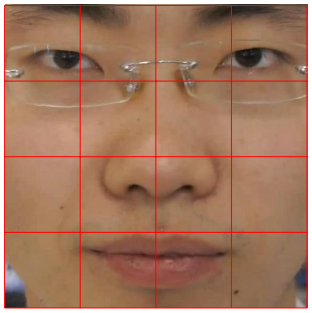


Figure 3.8: Example of Multi-Blocks.

3.4.2 Multi-Levels face representation (ML)

ML algorithm is a technique to divide the face ROI into sub-blocks which combine all features from different MB. In another term, we take the whole face ROI then we divided the ROI in four sub-blocks (two MB) and so until we reach the intended n level (See Figure 3.9). In the last, our results of ML is like this equation: $1^2 + 2^2 + 2^3 + ... + 2^n$.

3.5 Features selection

In this section after we describe the algorithm which are used in our approach and those algorithms have large vector (features). We must use method for selection and reducing the large vector which named FS algorithm. In below we explain FS on detail. Fisher-score [82]

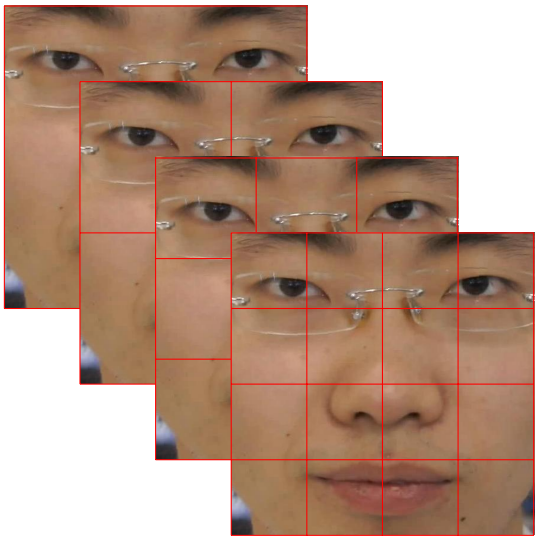


Figure 3.9: Example Multi-Levels.

is one of the most known method for feature selection. The idea in FS is to select each feature independently according to its scores under the Fisher criterion. We used FS in our approach to select the features and reduce the bin histograms and keep the best of histogram bins.

3.6 Classification

In this section, we are going to introduce the SVM machine learning algorithm. A SVM is a supervised machine learning algorithm that can be employed for both classification and regression purposes. SVMs are more commonly used in classification problems and as such, this is what we will focus on in this section.

Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. A schematic example ((Fig. 3.10) is shown in the illustration. In this example, the objects belong either to class GREEN or RED. The separating line defines a boundary on the right side of which all objects are GREEN and to the left of which all objects are RED. Any new object (white circle) falling to the right is labeled, i.e., classified, as GREEN (or classified as RED should it fall to the left of the separating line). SVMs are based on the idea of finding a hyperplane that best divides a dataset into two classes, as shown in the figure (3.10).

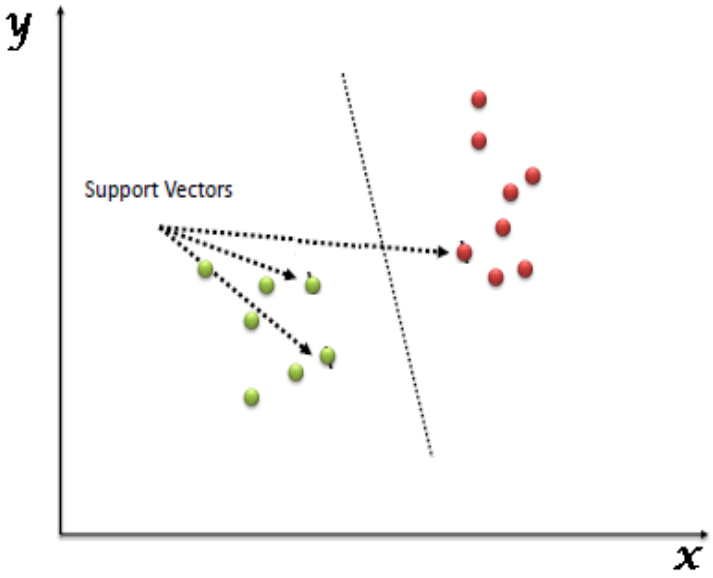


Figure 3.10: Example of SVM

Support vectors are the data points nearest to the hyperplane, the points of a data set that, if removed, would alter the position of the dividing hyperplane. Because of this, they can be considered the critical elements of a data set. As a simple example, for a classification task with only two features (like the figure 3.11), you can think of a hyperplane as a line that linearly separates and classifies a set of data. Intuitively, the further from the hyperplane our data points lie, the more confident we are that they have been correctly classified. We therefore want our data points to be as far away from the hyperplane as possible, while still being on the correct side of it. So when new testing data is added, whatever side of the hyperplane it lands will decide the class that we assign to it.

The distance between the hyperplane and the nearest data point from either set is known as the margin. The goal is to choose a hyperplane with the greatest possible margin between the hyperplane and any point within the training set, giving a greater chance of new data being classified correctly.

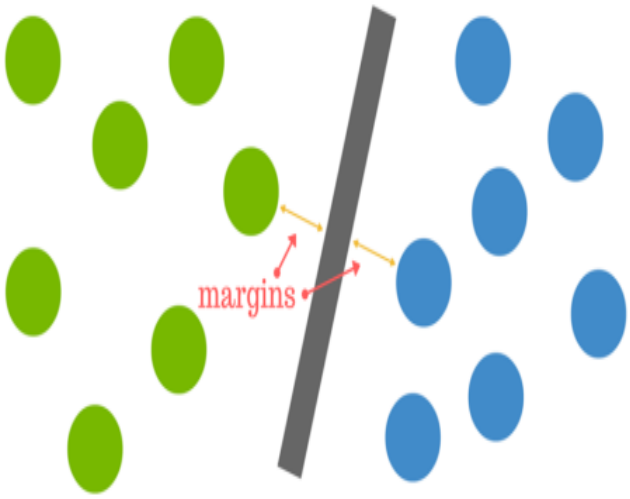


Figure 3.11: Example of hyperplane

3.7 Conclusion

In this chapter, we studied the system of anti-spoofing attack. This system is based on analysis of different contrast and texture characteristics of captured and recaptured images. The face anti-spoofing system has advantages, which is no need for user collaboration. There have been several studies on countermeasure techniques for the detection of spoof attacks. Compared to 2D spoofing attacks such as photograph and video, 3D mask attacks to face recognition systems is a considerably new subject. Even the impact of 3D mask attacks on existing recognition systems had not been analyzed before, in our study. In the next chapter, our proposed system of spoof detection and comparison of existing methods are detailed.

4

Experimental Results and Discussion

Contents

4.1 Introduction

In this chapter, we develop several case studies to employ the concepts introduced throughout this thesis on the face biometrics. The case studies are based on several prominent face anti-spoofing systems and state-of-the-the-art or face anti-spoofing features. The objective is to demonstrate the benefits of the anti-spoofing systems in creating more trustworthy face biometrics. In this research, we study two types of face spoofing: a photograph and a video of a valid user. For the first type of attack, we present an anti-spoofing solution based on a holistic representation of the face region, through a robust set of low-level feature descriptors, able to capture the differences between live and spoof images. For the second attack, we perform an analysis of the noise generated by the recaptured video to distinguish between both classes.

Usually, the system of face anti-spoofing divided on three phase, but in our approach consists of six phases are: face preprocessing, motion extraction by frame difference, features extraction, face representation, features selection, and classification. In the sections below we will give the effectiveness of the first five phases, effectiveness of color texture and finally, present our framework on face anti-spoofing.

4.2 Effectiveness of face alignment

In this section, we will explain the effectiveness of our proposed face alignment [83, 84] (See section 3.2). We evaluated the proposed approach (See Figure 4.1) on the NUAA Photograph Imposter Database [69]. In our experiments, we used Matlab2013b, beginning with VJ algorithm [74] to locate all components of the face images. Using Stasm [85] to detect landmark of the face image, the eyes are localized. The coordinates of the eyes are used to adjust, and then to crop the face as explain in (See section 3.2). All cropped faces are resized to a consistent size 64×64. We also divided the normalized faces in 9 block with overlapping algorithm before applying $LBP^{U2}_{(8,2)}$ to extract the local features in each region of the image. In this step, we computed the histogram of each block to get 59 bin histograms. We concatenated then these histograms in a simple one of a $531bin$. For classification, we used SVM classification. We applied our approach using face detection without Stasm using the same image normaliza-

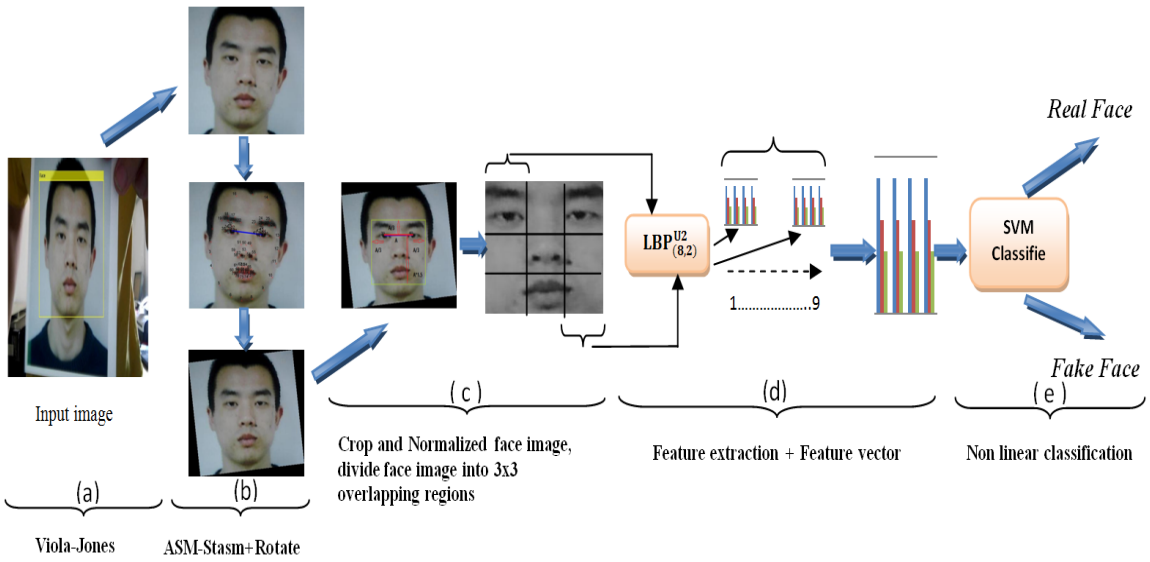


Figure 4.1: The proposed approach :(a) VJ algorithm, (b) Active Shape Models with Stasm, (c) Crop and normalzide the face, (d) Feature extraction using LBP and (e) Non-linear SVM classifier for determining a real face or fake.

tion in NUAA databases in one hand. In other hand, we calculated the results using the VJ algorithm [74] and Active Shape Model with Stasm [85]. Also, for 107 images not detected by Stasm, we have used a manual detection by manual calculation coordinates of eyes. We compared our results with those of the state-of-the-art: LBP+Gabor+HOG [20], LBP overlapping [86], Bad Illumination Conditions [87]. For fair comparison, we used the same protocol with other authors: 1743 live images, 1748 non-live, for train and 3362 live and 5761 non-live samples for test.

The performance of the three detection (without Stasm, with Stasm, and manual correction) associated with texture operator LBP in terms of ROC and DET curve are shown in (Fig.4.2 and Fig. 4.3). From the results, we can notice that when using face alignment are good compared to that without face alignment. The EER, shown in Table 4.1, indicates that the detection with Stasm (EER= 2.4) is better than without Stasm (EER = 3.9). For the manual detection of 107 image not detected by Stasm (EER= 0.6) gives best results. So, we have to develop an algorithm to detect automatically any coordinate of eyes.

Table 4.1: Performance comparison between our proposed approach and the best results on the same database using the same protocol.

Methods	Accuracy %	EER	AUC
Bad Illumination Conditions [87]	93	8.2	-
LBP overlapping [86]	-	2.9	0.99
LBP+Gabor+HOG [20]	98	1.1	0.999
Without_stasm [83]	97.31	3.9	0.9930
With_stasm [83]	98.41	2.4	0.9975
With_stasm (manual correction) [83]	**99.61**	**0.6**	**0.9998**

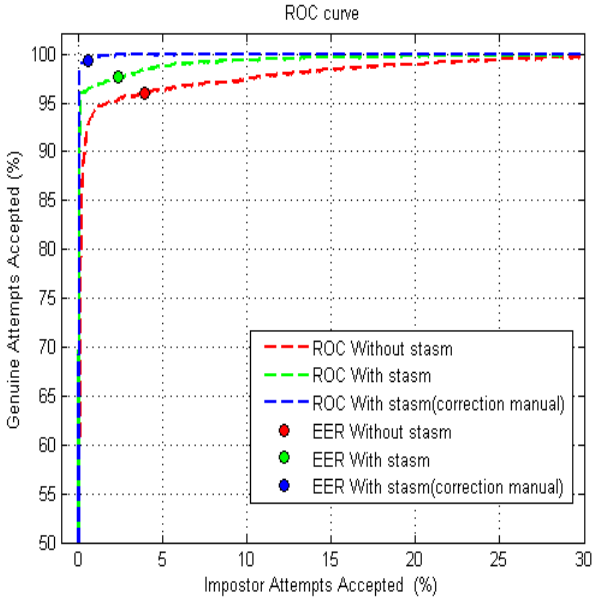


Figure 4.2: Performance (ROC curves) of the proposed approach without Stasm,with Stasm, and manual correction.

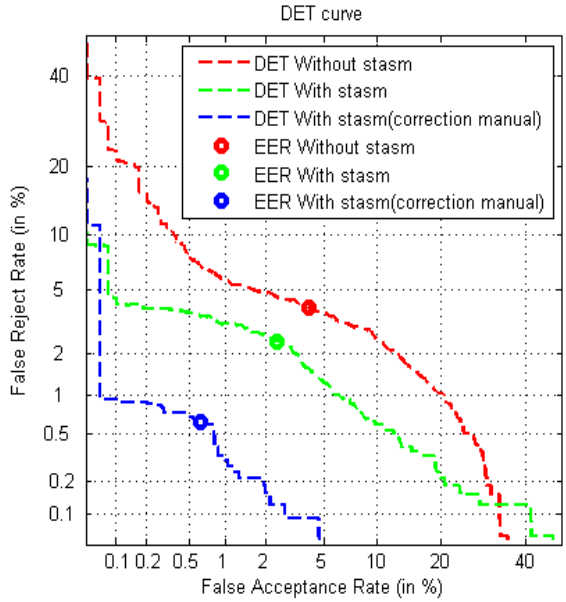


Figure 4.3: Performance (DET curves) of the proposed approach without Stasm, with Stasm, and manual correction.

4.3 Effectiveness of frame difference (our contribution)

In our work, we propose an algorithm for face spoofing detection based on an extended FD algorithm [88]. We also combine this extended frame difference algorithm with a ML representation to take into account both dynamic and static information. This combination gave us multiple parts of the foreground image (See Figure 4.4).

Commonly, the FD technique computes the difference between the current and the previous frame. A threshold is then used to obtain the foreground which is a binary image. The equation

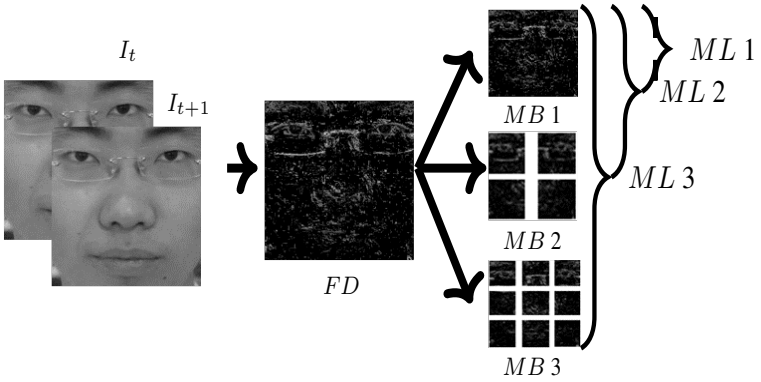


Figure 4.4: Principle of FD and ML.

of the frame difference is given by:

$$F_t = |I_t - I_{t-1}| \tag{4.1}$$

Where :

- F_t: Difference between two frames
- I_t: Current frame
- I_{t-1}: Previous frame

In our case, we used a threshold only to eliminate the unchanged pixels values between the two successive frames. If there is a motion, the foreground pixels takes the value of the current frame. However, if there is no motion, the foreground pixel is set to zero (see Equation 4.2). Figures 4.5, 4.6 and Tables 4.2, 4.3 demonstrate the effectiveness of using our FD approach.

$$F'(i,j)_t = \begin{cases} I_t(i,j) & if\ F(i,j)_t > T \\ 0 & otherwise. \end{cases} \tag{4.2}$$

Where:

- F'_t: Foreground
- T: Threshold = 0

In Table 4.2, we computed the entropy of the real and the fake face of the same person. The entropy describes the quantity of information of the image, the image entropy equation is given by [89]:

$$E = -\sum_{i=0}^{255} P_i.log_2(P_i) \tag{4.3}$$

where E is the entropy of F' and P_i is the probability of the color i. Both real and fake faces have three quality (Low, Normal, and High). We took into account two type of attack, Printed

and Video attack. We can observe that the entropy is greater in the case of real faces compared to fake faces in all scenarios.

Table 4.2: Entropy

Qualities	Real	Printed Attacks		Video attack
Low	4.16	1.76	1.58	1.96
Normal	4.13	2.17	1.57	1.79
High	4.92	3.87	3.35	4.68

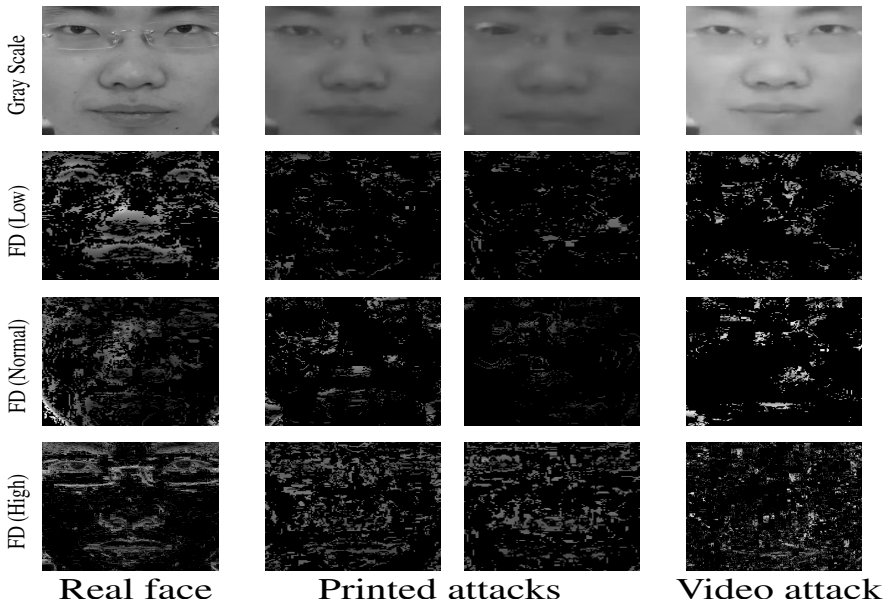


Figure 4.5: Example of a genuine face and corresponding print and video attacks in grey-scale and FD.

Visually, we observe from Figure 4.5 that when computing the foreground of the real faces, the facial features are more visible compared to the case of fake ones in all qualities and all types of attack. From these remarks, we have been motivated to use FD in face anti-spoofing. This frame difference allows us to extract motion in foreground and illustrate the fake faces. Then the ML face representation is applied on the foreground of the frame difference that permits to obtain multiple blocks then followed by texture descriptor. The use of FD (motion) combined with ML [90] (representation) and texture description improve the results. This was proved experimentally (See Figure 4.6 and Table 4.3).

We observe from Table 4.3 and Figure 4.6, when we used Texture descriptors (2) (LBP, LPQ

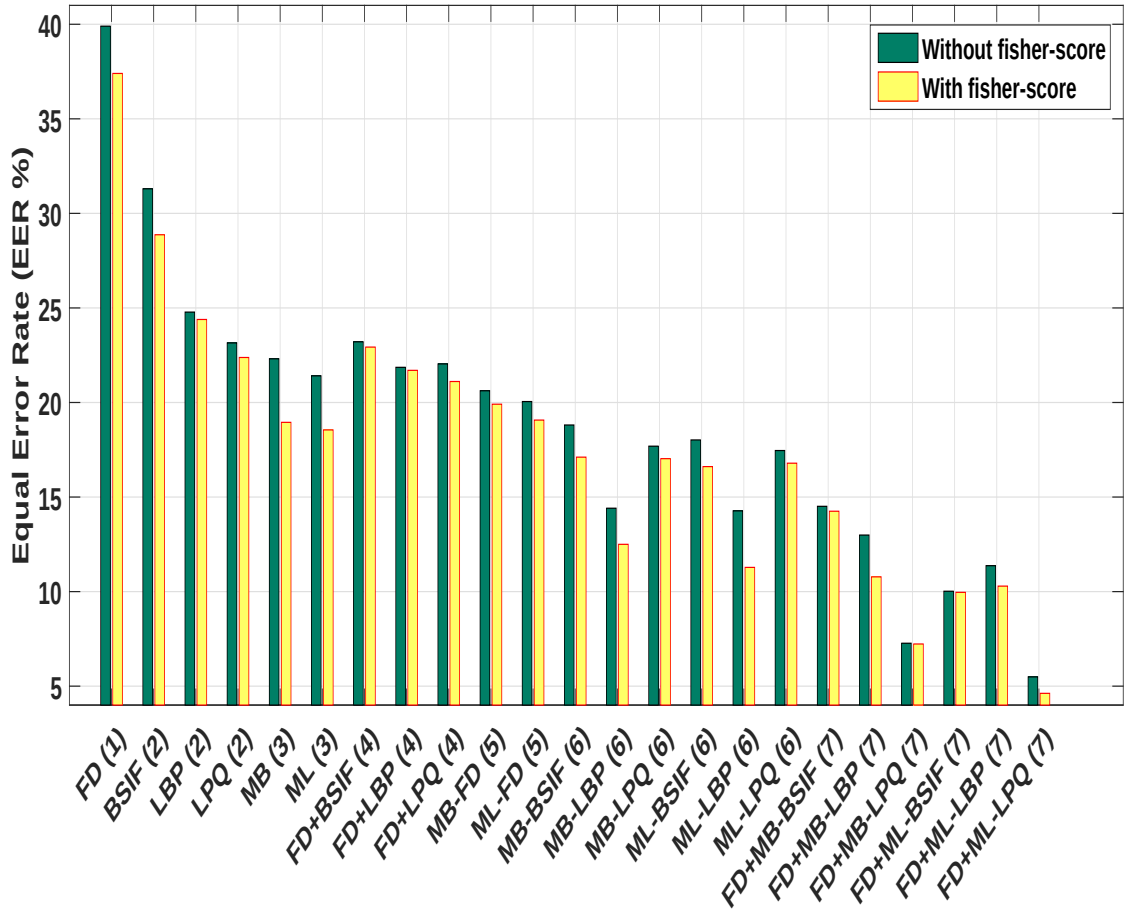


Figure 4.6: Impact of FD (Motion) on Representation and Texture across the CASIA FASD.

and BSIF), we got an improvement in EER compared to using Motion (1) (FD) only by computing the histogram of FD directly. The results are better when we combined Motion and Texture (4). Another aspect is when we using ML and MB Representations (3), this improves the results of both motion (1) and texture (2). We can remark also that, combining Representation with Motion (5) or with Texture (6) improves clearly the results. In the case of combining Motion FD with Texture and Representation (7) the results are better compared to all previous methods. Finally, adding FS to any method improves the EER. This is the reason why we choose to use (**(Motion) FD + (Texture) LPQ + (Representation) ML + FS**) as new approach.

Table 4.3: Results in EER (%) on CASIA for Motion (FD), Representation (ML, MB) and Texture (LBP,LPQ, BSIF).

Types of methods	Methods	Without fisher	With fisher
Motion (1)	FD	39.90	37.40
Texture (2)	BSIF	31.30	28.87
	LBP	24.78	24.39
	LPQ	23.15	22.38
Representation (3)	MB	22.31	18.95
	ML	21.41	18.55
Motion + Texture (4)	FD-BSIF	23.21	22.93
	FD-LBP	21.86	21.70
	FD-LPQ	22.04	21.11
Motion + Representation (5)	FD-MB	20.62	19.91
	FD-ML	20.05	19.07
Representation + Texture (6)	MB-BSIF	18.81	17.11
	MB-LBP	14.41	12.50
	MB-LPQ	17.69	17.03
	ML-BSIF	18.02	16.61
	ML-LBP	14.27	11.28
	ML-LPQ	17.46	16.79
Motion + Representation + Texture (7)	FD-MB-BSIF	14.51	14.25
	FD-MB-LBP	12.99	10.78
	FD-MB-LPQ	07.27	07.23
	FD-ML-BSIF	10.02	09.96
	FD-ML-LBP	11.37	10.29
	FD-ML-LPQ	05.49	04.62

4.4 Effectiveness of different descriptors (features extraction)

Visual information contained in images is usually represented by low-level feature descriptors focusing on different types of information, such as color, texture, and shape. An adequate feature descriptor is able to discriminate between regions with different characteristics and it allows similar regions to be grouped together even when captured under noisy conditions. However, it is usually difficult to have a single feature descriptor adequate for many application

domains [91]; this has motivated researchers to develop a variety of feature extraction methods.

We focus in our work on the development and the analysis of feature extraction methods so that a better representation may be extracted from the visual information contained in images and videos. In below, we present three famous descriptors (LBP, LPQ, and BSIF) with different operators and we will study which one is good of each descriptor for our work.

4.4.1 Different operators of LBP

The size of the histogram in a multi-resolution analysis, in spatial domain, increases linearly with the number of neighborhoods P. The choice of an appropriate LBP representation in the planes is an important issue since it impacts the size of the histograms. Using uniform patterns or rotation invariant extensions, in one or multiple planes, may bring a significant reduction in computational complexity.

In this experiment, the effectiveness of different LBP operators presented in Table 4.4 and Figure 4.7 show the performance, in terms of EER, configuring each plane as $\text{LBP}^{(u2)}$ (uniform patterns), $\text{LBP}^{(ri)}$ (rotation invariant), and $\text{LBP}^{(riu2)}$ (rotation invariant uniform patterns).

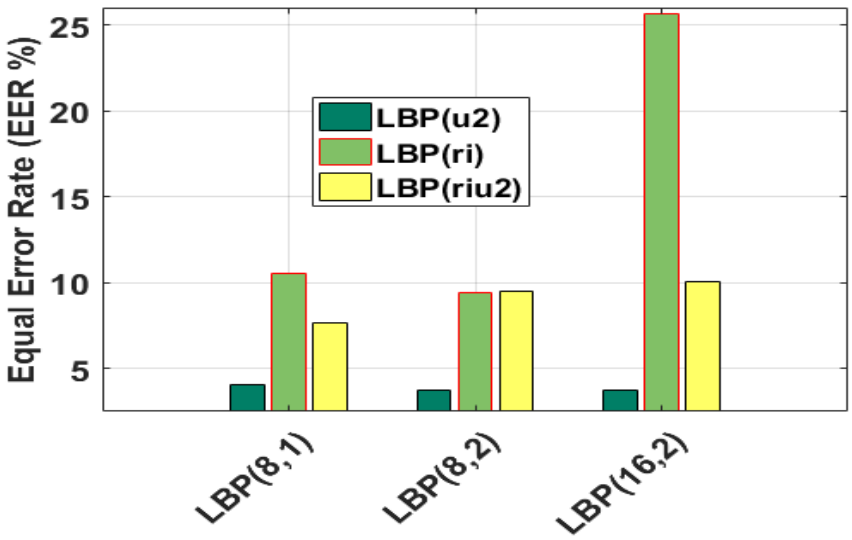


Figure 4.7: Bar graph of EER on different operator of LBP.

Table 4.4: EER on different operator of LBP

Operator of LBP	EER (%)
$LBP^{u2}_{(8,1)}$	04.06
$\mathbf{LBP^{u2}_{(8,2)}}$	**03.71**
$LBP^{u2}_{(16,2)}$	03.71
$LBP^{ri}_{(8,1)}$	10.55
$LBP^{ri}_{(8,2)}$	09.39
$LBP^{ri}_{(16,2)}$	25.66
$LBP^{riu2}_{(8,1)}$	07.62
$LBP^{riu2}_{(8,2)}$	09.51
$LBP^{riu2}_{(16,2)}$	10.06

4.4.2 Different operators of LPQ

The LPQ value is first computed for every pixel of the given image. Next, local histograms with 256 bins are computed within a sliding window. We compute the concatenated histogram descriptor for varying window sizes and with different radius for the neighborhood of each pixel see Table 4.5. In our experiments, we compare LPQ with different radius, windows sizes and the results are present in Figure 4.8.

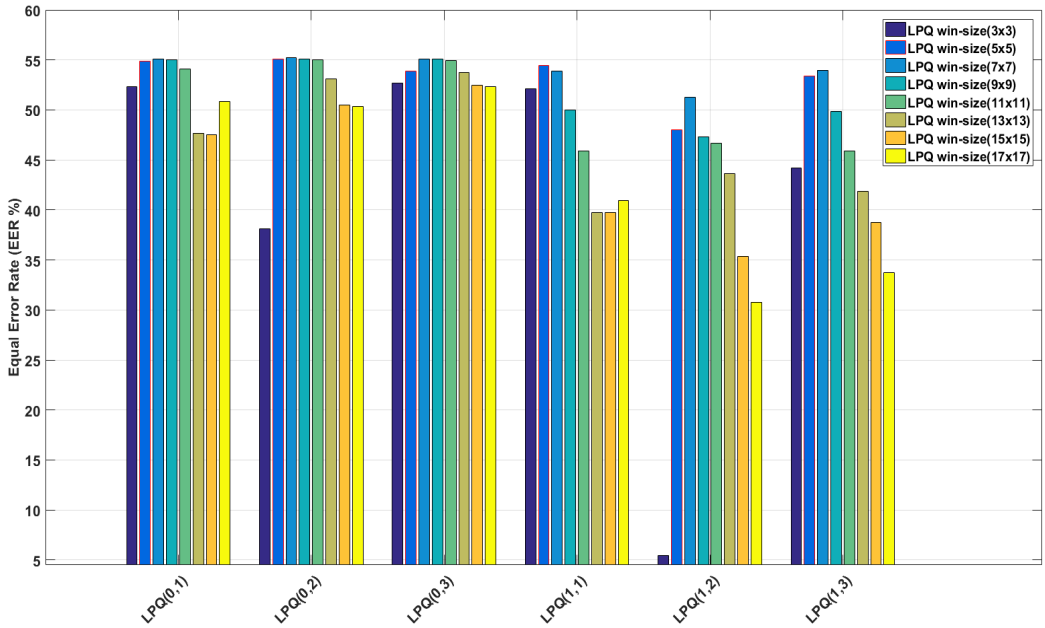


Figure 4.8: Bar graph of EER on different operator of LPQ.

Table 4.5: EER on different operator of LPQ

Operator of LPQ	EER (%)	Operator of LPQ	EER (%)
$LPQ_{(3,0,1)}$	52.32	$LPQ_{(11,0,1)}$	54.07
$LPQ_{(3,0,2)}$	38.10	$LPQ_{(11,0,2)}$	55.02
$LPQ_{(3,0,3)}$	52.70	$LPQ_{(11,0,3)}$	54.93
$LPQ_{(3,1,1)}$	52.11	$LPQ_{(11,1,1)}$	45.89
$\mathbf{LPQ_{(3,1,2)}}$	**05.47**	$LPQ_{(11,1,2)}$	46.67
$LPQ_{(3,1,3)}$	44.17	$LPQ_{(11,1,3)}$	45.92
$LPQ_{(5,0,1)}$	54.90	$LPQ_{(13,0,1)}$	47.65
$LPQ_{(5,0,2)}$	55.11	$LPQ_{(13,0,2)}$	53.12
$LPQ_{(5,0,3)}$	53.86	$LPQ_{(13,0,3)}$	53.77
$LPQ_{(5,1,1)}$	54.43	$LPQ_{(13,1,1)}$	39.73
$LPQ_{(5,1,2)}$	48.00	$LPQ_{(13,1,2)}$	43.63
$LPQ_{(5,1,3)}$	53.42	$LPQ_{(13,1,3)}$	41.86
$LPQ_{(7,0,1)}$	55.11	$LPQ_{(15,0,1)}$	47.50
$LPQ_{(7,0,2)}$	55.20	$LPQ_{(15,0,2)}$	50.47
$LPQ_{(7,0,3)}$	55.11	$LPQ_{(15,0,3)}$	52.46
$LPQ_{(7,1,1)}$	53.89	$LPQ_{(15,1,1)}$	39.76
$LPQ_{(7,1,2)}$	51.27	$LPQ_{(15,1,2)}$	35.36
$LPQ_{(7,1,3)}$	53.92	$LPQ_{(15,1,3)}$	38.72
$LPQ_{(9,0,1)}$	55.05	$LPQ_{(17,0,1)}$	50.84
$LPQ_{(9,0,2)}$	55.08	$LPQ_{(17,0,2)}$	50.35
$LPQ_{(9,0,3)}$	55.11	$LPQ_{(17,0,3)}$	52.32
$LPQ_{(9,1,1)}$	49.99	$LPQ_{(17,1,1)}$	40.93
$LPQ_{(9,1,2)}$	47.31	$LPQ_{(17,1,2)}$	30.75
$LPQ_{(9,1,3)}$	49.88	$LPQ_{(17,1,3)}$	33.72

4.4.3 Different operators of BSIF

In our experiments, we use the standard filters, which represent eight different orientations of edges. As before, we extract a local descriptor for different window sizes, overlap between neighboring windows and different filter sizes (see Table 4.6) and concatenate each local histogram to a global histogram representation. For all experiments, we use 5 until 9-bit code words and the 5×5 until 17×17 filters and the results are presented in Figure 4.9.

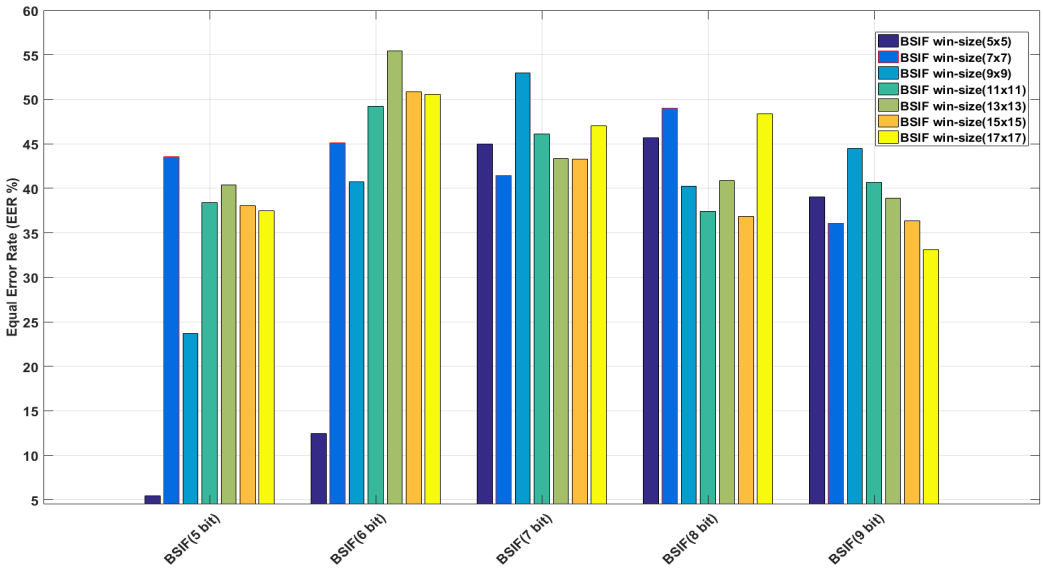


Figure 4.9: Bar graph of EER on different operator of BSIF.

Table 4.6: EER on different operator of BSIF

Operator of BSIF	EER (%)	Operator of BSIF	EER (%)
BSIF$_{(5\times5,5)}$	**05.44**	$BSIF_{(13\times13,5)}$	40.40
$BSIF_{(5\times5,6)}$	12.43	$BSIF_{(13\times13,6)}$	55.41
$BSIF_{(5\times5,7)}$	44.97	$BSIF_{(13\times13,7)}$	43.33
$BSIF_{(5\times5,8)}$	45.65	$BSIF_{(13\times13,8)}$	40.89
$BSIF_{(5\times5,9)}$	39.02	$BSIF_{(13\times13,9)}$	38.93
$BSIF_{(7\times7,5)}$	43.57	$BSIF_{(15\times15,5)}$	38.07
$BSIF_{(7\times7,6)}$	45.11	$BSIF_{(15\times15,6)}$	50.83
$BSIF_{(7\times7,7)}$	41.41	$BSIF_{(15\times15,7)}$	43.27
$BSIF_{(7\times7,8)}$	49.04	$BSIF_{(15\times15,8)}$	36.82
$BSIF_{(7\times7,9)}$	36.10	$BSIF_{(15\times15,9)}$	36.37
$BSIF_{(9\times9,5)}$	23.70	$BSIF_{(17\times17,1)}$	37.47
$BSIF_{(9\times9,6)}$	40.72	$BSIF_{(17\times17,1)}$	50.59
$BSIF_{(9\times9,7)}$	52.94	$BSIF_{(17\times17,1)}$	47.05
$BSIF_{(9\times9,8)}$	40.27	$BSIF_{(17\times17,1)}$	48.39
$BSIF_{(9\times9,9)}$	44.46	$BSIF_{(17\times17,1)}$	33.10
$BSIF_{(11\times11,5)}$	38.41		
$BSIF_{(11\times11,6)}$	49.22		
$BSIF_{(11\times11,7)}$	46.13		
$BSIF_{(11\times11,8)}$	37.40		
$BSIF_{(11\times11,9)}$	40.65		

Table 4.7: Comparison of number of frames in term of (EER)

Number of frame	EER (%)	Number of frame	EER (%)
5	25.86	**125**	15.38
10	22.25	**150**	15.98
15	19.24	**175**	16.06
25	17.90	**200**	14.98
50	20.03	**225**	14.54
75	16.97	**250**	13.98
100	18.93	**275**	14.39

Table 4.8: Comparison between the different MB-LPQ

MB-LPQ divisions	EER (%) without fisher score	EER (%) with fisher score
1 x 1	**13.98**	**13.31**
2 x 2	15.94	15.47
3 x 3	17.72	15.95
4 x 4	21.19	18.59
5 x 5	14.30	13.96

4.5 Effectiveness of face representation

In our experiment, we choose the LPQ descriptor because it gives best results compared to others studied descriptors. For the params of LPQ, we consider a window size of 5×5. We used LPQ on the overall test of CASIA database because it have all qualities and attacks. We take on each step N numbers of frames and calculate the EER for testing which the number of frames gives the best results. After this test, we decided to take 250 frames on each video. The Table 4.7 shows the result.

Now, we will compare our two face representation MB-LPQ and ML-LPQ [90]. The results of MB-LPQ with and without FS are presented in Figure 4.10 and Figure 4.11 respectively. The results of ML-LPQ with and without FS are presented in Figure 4.12 and Figure 4.13 respectively. The Table 4.8 and Table 4.9 show the compared results of MB-LPQ and ML-LPQ respectively.

We compared now the MB-LPQ with ML-LPQ. The Both descriptors are used with and without FS. As we see in Table (4.8) and DET curves (Figures 4.11 and 4.10), MB-LPQ with fisher score, the (EER = 13.31 %) is good compared the MB-LPQ without FS MB-LPQ, the (EER = 13.98 %. After that in Table (4.9) and DET curves (Figures 4.13 and 4.12), ML-LPQ

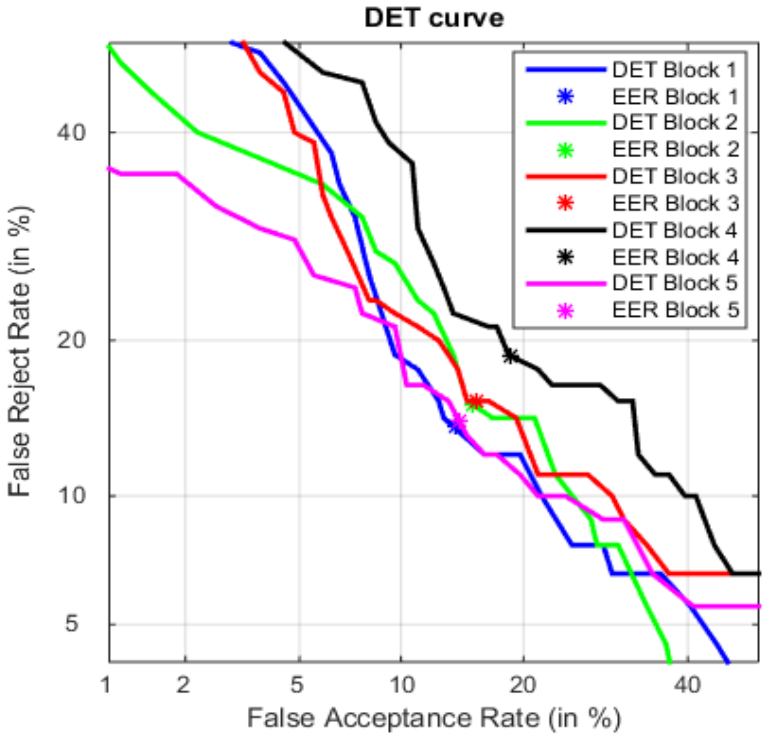


Figure 4.10: DET of MB-LPQ with FS.

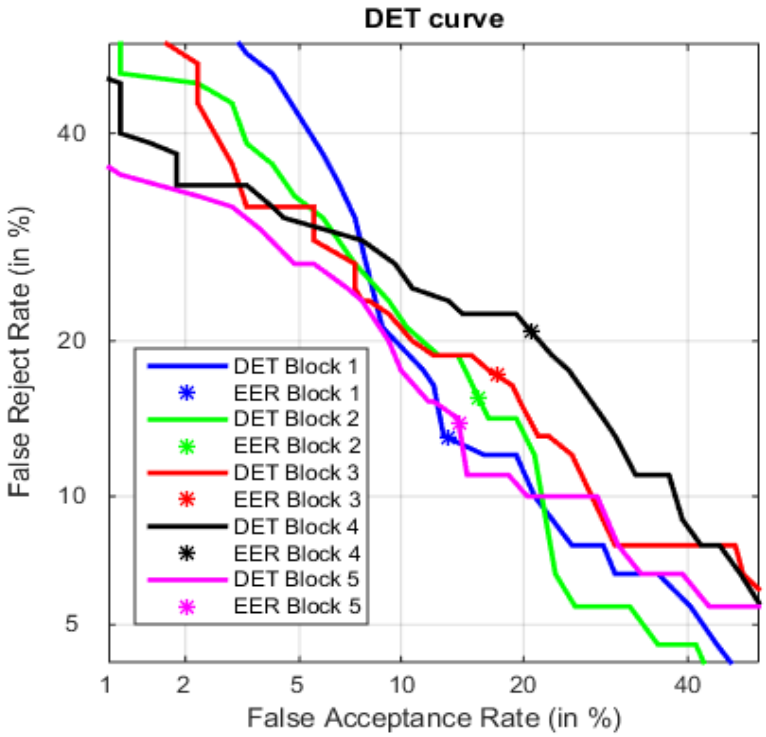


Figure 4.11: DET of MB-LPQ without FS.

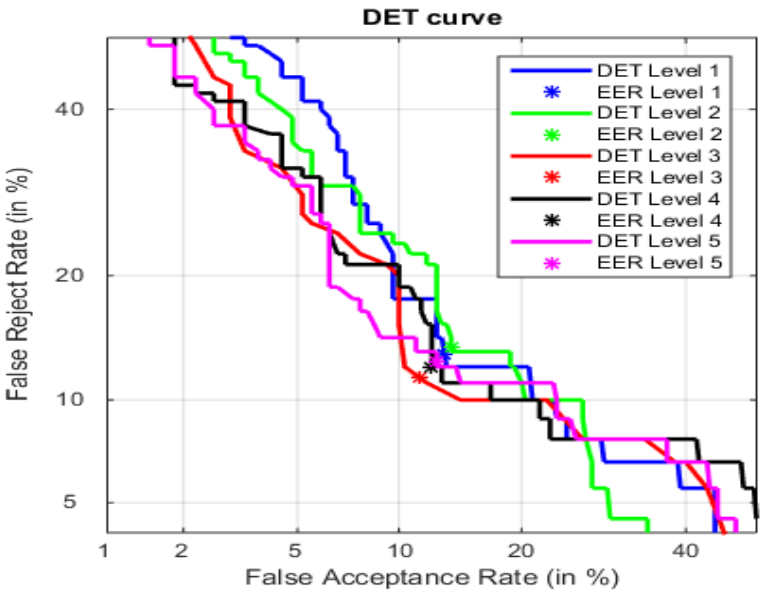


Figure 4.12: DET of ML-LPQ with FS (3 level).

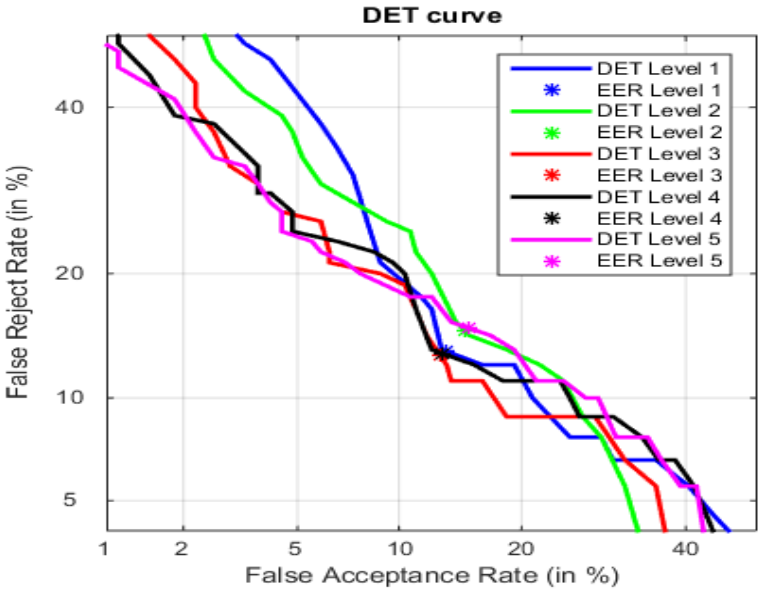


Figure 4.13: DET of ML-LPQ without FS (3 level).

Table 4.9: Comparaison between different levels of ML-LPQ

ML-LPQ Level	EER (%) without FS	EER (%) with FS
1	13.98	13.31
2	14.93	14.34
3	**12.97**	**11.39**
4	13.26	12.47
5	15.85	12.85

Table 4.10: Comparison of the results (in EER %) between our proposed approach and the state-of-the-art on CASIA data base

methodes & scenario	Low (1)	Normal (2)	High (3)	Warped(4)	Cut(5)	Video(6)	Overall(7)
IQA [47]	31.7	22.20	05.60	26.10	18.30	34.40	32.40
DoG baseline [71]	13.00	13.00	26.00	16.00	06.00	24.00	17.00
LBP$^{u2}_{8,1}$ [57]	11.00	17.00	13.00	13.00	16.00	16.00	16.00
LBP overlapping fisher [92]	07.20	08.80	14.40	12.00	10.00	14.70	13.10
Multi-LBP [60]	12.77	16.66	26.66	15.55	25.55	17.77	17.77
Mag-Multi-LBP [60]	07.22	13.33	29.44	14.44	22.22	13.33	15.74
HOOF [60]	16.66	30.00	26.11	15.55	17.77	38.88	21.11
Mag-HOOF [60]	17.22	33.33	22.77	12.22	20.00	36.60	22.22
HOOF + Multi-LBP [60]	09.44	20.55	16.66	10.00	16.66	24.44	15.55
Mag-HOOF + Mag-Multi-LBP [60]	06.11	23.33	13.88	10.00	14.44	20.00	14.44
CDD [93]	01.50	05.00	02.80	06.40	04.70	00.30	11.80
MB-LPQ(our) [90]	16.31	22.36	11.34	14.20	13.65	10.46	13.98
MB-LPQ fisher (our) [90]	13.37	13.12	08.45	12.11	11.43	07.61	13.31
ML-LPQ(our) [90]	14.83	08.95	05.41	15.83	10.01	10.06	12.97
ML-LPQ fisher (our) [90]	12.49	08.96	05.22	13.62	09.66	10.10	**11.39**

with FS, EER = 11.39 % is good compared to ML-LPQ without fisher score, EER = 12.97 %. Finally, we that outcome ML-LPQ with FS is the best.

Now after discussion of our result and outcome that ML-LPQ with FS gives the best result. To test the robustness of our system, we start now comparing our approach with state-of-the-the-art on CASIA face anti-spoofing database, which already have 7 scenario called the low quality (1), normal quality (2) and high quality (3), warped photo attack (4), cut photo attack (5), video attack (6), the last scenario is overall test (7) which have all type of qualities and attacks. The Table 4.10 shows the comparison of results between the state-of-the-art and the $7scenario$ of CASIA databases. Finally for more comparison we show the DET curves of all scenarios using MB-LPQ and ML-LPQ with and without FS (See Figures 4.14, 4.15, 4.16 and 4.17).

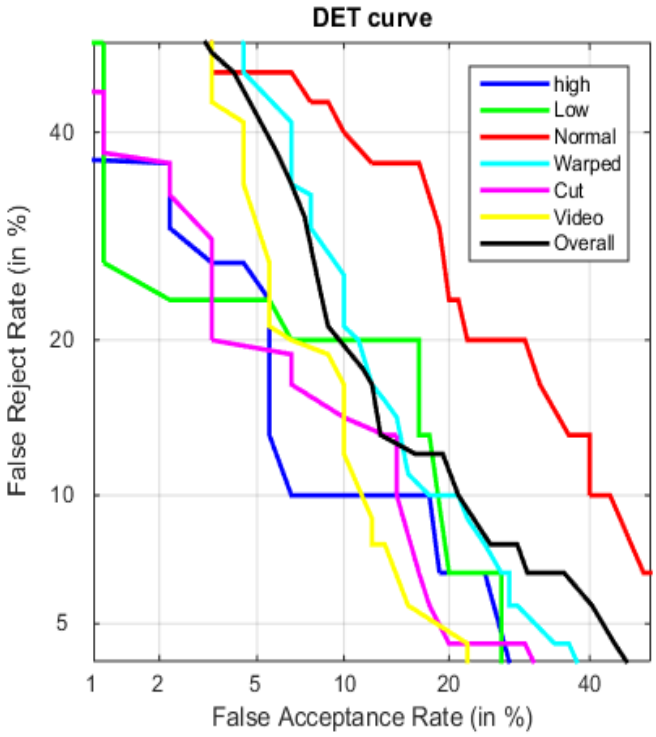


Figure 4.14: DET of MB-LPQ without FS, 7 scenario.

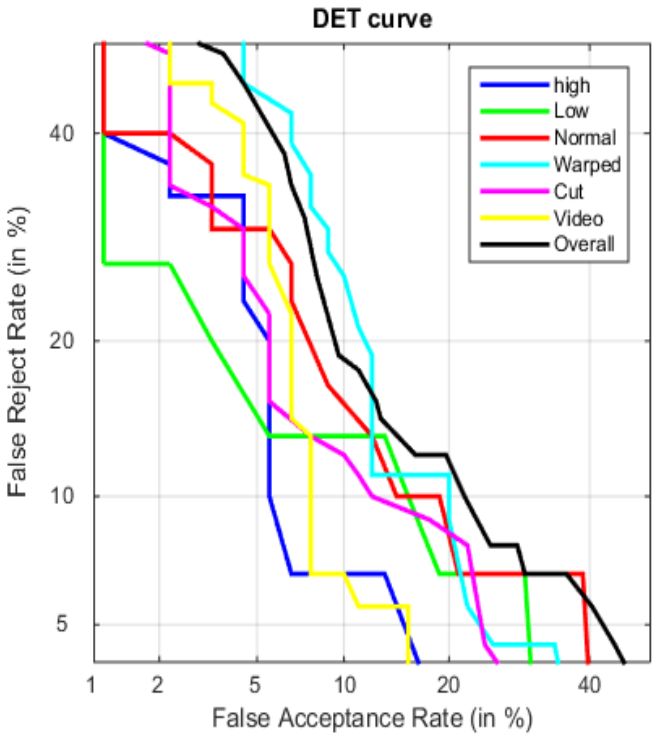


Figure 4.15: DET of MB-LPQ with FS, 7 scenario.

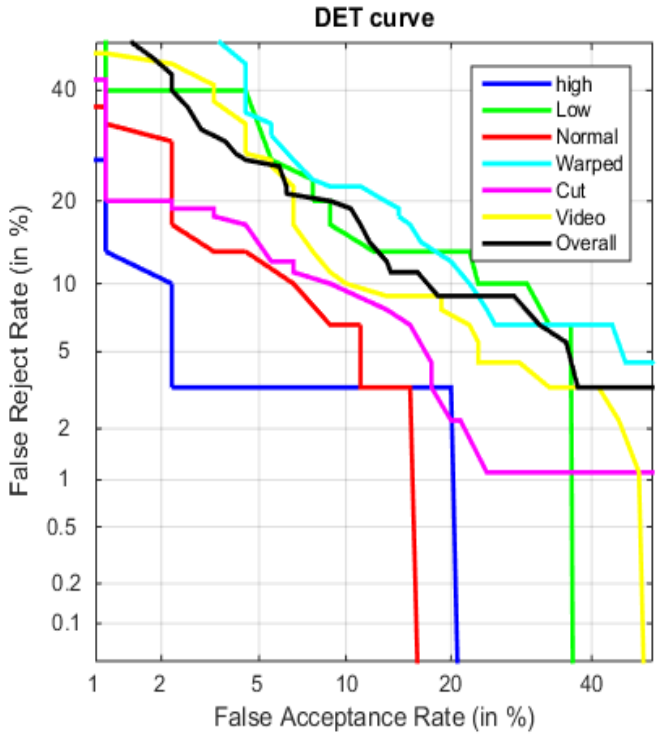


Figure 4.16: DET of ML-LPQ without FS (3 level), 7 scenario.

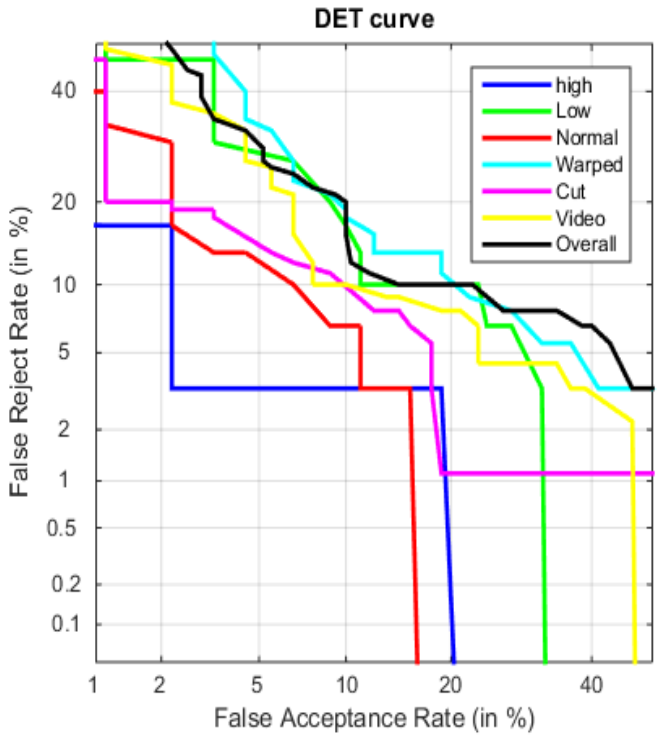


Figure 4.17: DET of ML-LPQ with FS (3 level), 7 scenario.

4.6 Effectiveness of Fisher-Score

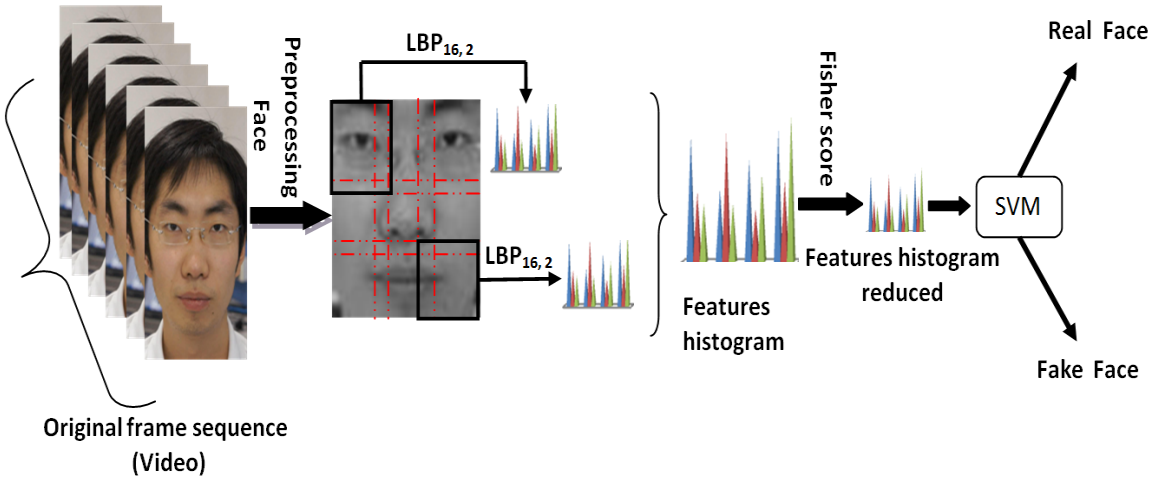


Figure 4.18: The proposed approach using FS

In this section, we give results of our approach using FS as a solution to reduce of big data [92], to approve the results. We give also a comparison of those results with the state-of-the-art. The performance evaluation of the studied anti-spoofing algorithm is measured by the EER using two challenge databases which are: NUAA Photograph Imposter Database and CASIA Face Anti-Spoofing Database.

4.6.1 Results in NUAA Photograph Imposter Database

We compared our results with those of the state-of-the-art: LBP+Gabor+HOG [20], LBP overlapping [86], LPQ [18], Bad Illumination Conditions [87]. For fair comparison, we used the same protocol with other authors: 1743 live images, 1748 non-live, for train and 3362 live and 5761 non-live samples for test. The performance of our approach with texture operators LBP in terms EER, shown in (Table 4.11 and Figures (4.19, 4.20)), indicates that our approach gives best results compared to the state-of-the-art.

Table 4.11: Performance comparison between our proposed approach and the best results on the same database and using the same protocol.

Methods	EER%
Gabor [86]	09.50
Bad Illumination Conditions [87]	08.20
LPQ [18]	04.90
LBP overlapping Blocks [86]	02.90
LBP+Gabor+HOG [20]	01.10
LBP Without_Stasm	03.95
LBP Without_Stasm_Fisher	03.83
8_1_Stasm	03.21
8_1_Stasm_Fisher	03.12
8_2_Stasm	02.32
8_2_Stasm_Fisher	02.25
16_2_Stasm	01.84
16_2_Stasm_Fisher	01.00
16_2_Stasm_M_fisher(correction manual)	00.61

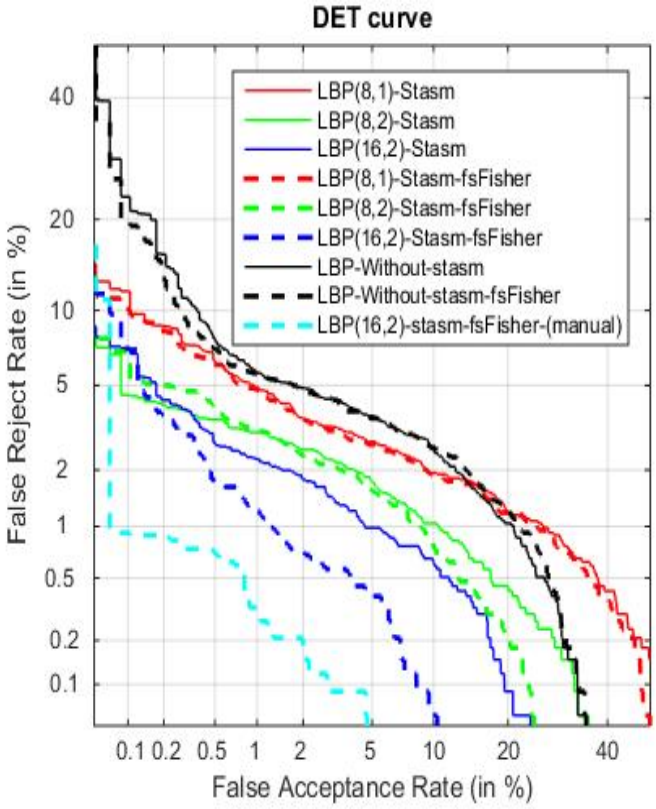


Figure 4.19: Performance (DET curves) of the proposed approach without (Stasm, Fisher), with (Stasm,Fisher).

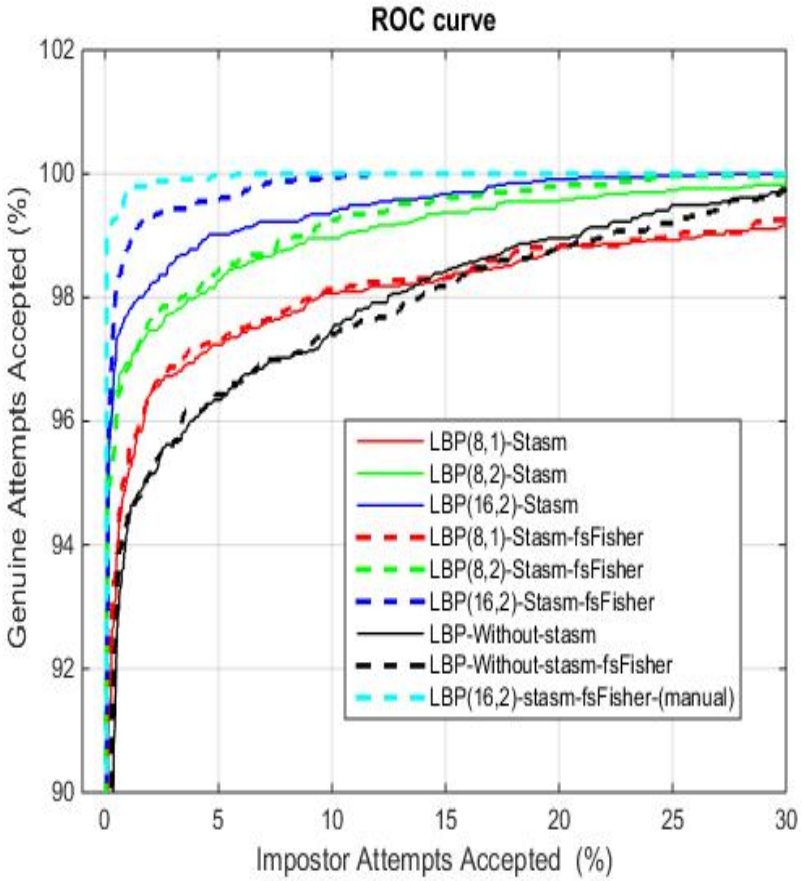


Figure 4.20: Performance (ROC curves) of the proposed approach without (Stasm,Fisher), with (Stasm,Fisher).

4.6.2 Results in CASIA Face Anti-Spoofing Database

This database has only two totally independent datasets train and test. For fair comparison, we used the same protocols reported in [71]. The results presented on term of EER are computed as two test. The first test is per-frame and the second test is per-video. Finally the first test gives score as real image or fake one and the second test gives score as real video or not. The database have seven scenarios in train and test, because the main purpose is to investigate the possible effects of different fake face types and imaging qualities. The scenarios are: low (1), normal (2) and high quality (3), warped photo (4), cut photo (5) and video attacks (6), overall test (7). The results of each scenario are reported as EER in (Table 4.12).

Table 4.12: Comparison of the results (in EER %) between our proposed approach and the state-of-the-art.

Scenario	(1)	(2)	(3)	(4)	(5)	(6)	(7)
IQA [47]	31.7	22.2	5.6	26.1	18.3	34.4	32.4
DoG baseline [71]	13	13	26	16	6	24	17
LBP [57]	11	17	13	13	16	16	16
Our [92]	**7.2**	**8.8**	14.4	**12**	10	**14.7**	**13.1**

4.6.3 Discussion of two databases

When using NUAA database, we have obtained an EER = 1% which represent the best result compared to the other works (Table 4.11). In the case of CASIA database (Table 4.12), we got good results for low and normal quality (EER=7.2% and 8.8%), but in high quality, the results are poor performance compared to others. For warped photo and attack video, our results are good compared to other groups. In addition, when using a cut photo, our results are a little low compared to Zhang *et al.* [71]. In conclusion, the overall test in our work is better than the others in the case of texture algorithm.

4.7 Effectiveness of color texture (challenge on IJCB)

We propose an approach based on MB-LPQ applied on color face images. We participate with this work in a challenge conference IJCB 2017 [94]. The RGB face images of OULU-NPU: A Mobile Face Presentation Attack Database are converted into YCbCr color space and divided into multiple blocks [90, 95]. The LPQ features are extracted from each block and then concatenated into a single feature vector. The LPQ features extracted from each channel are concatenated to form the overall face representation. Each video is represented with a single vector by averaging feature vectors extracted from the first $10 frames$. The score for each video is then computed using a Softmax classifier. The obtained results are illustrated in Table 4.13).

Table 4.13: The performance of the proposed methods under four protocols which are: different illumination and location conditions, novel attacks, input camera variations, and environmental, attack and camera device variations.

Protocol	Methods	Dev	Test				
		EER(%)	Display	Print	Overall		
			APCER(%)	APCER(%)	APCER(%)	BPCER(%)	ACER(%)
Protocol I	MBLPQ	2.2	31.7	44.2	44.2	3.3	23.8
Protocol II	MBLPQ	31.7	31.7	31.7	31.7	31.7	12.9
Protocol III	MBLPQ	2.3 ± 0.6	5.8 ± 5.8	12.9 ± 4.1	12.9 ± 4.1	21.9 ± 22.4	17.4 ± 10.3
Protocol IV	MBLPQ	3.6 ± 0.7	35.0 ± 25.5	45.0 ± 25.9	49.2 ± 27.8	24.2 ± 27.8	36.7 ± 4.7

4.8 Proposed framework

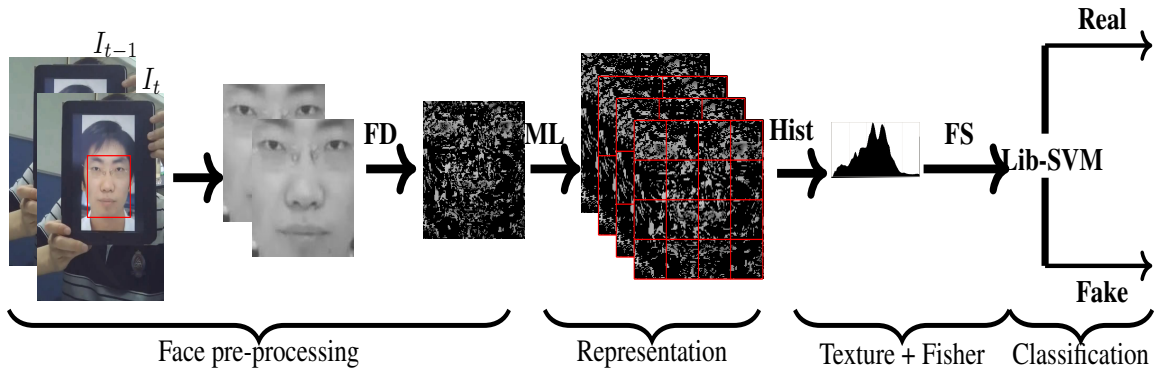


Figure 4.21: Framework of our proposed approach.

The Figure 4.21 illustrates the general structure of our framework [88]. First, we detect the face and localize the eyes center coordinates to normalize the ROI. Second, we extract the motion by using the FD between consecutive faces. Then, we apply ML representation to get multiple blocks to be used in features extraction. Features of all blocks are concatenated to get one feature vector. We used all the previous steps for an input video of 6 seconds (150 frames), then we averaged the feature vectors of all these frames. After that, we ranked the average feature vector by FS. Finally, we used Lib-SVM as a classifier to differentiate between real and fake faces.

Table 4.14: Effect of different time window sizes on CASIA Face Anti-Spoofing Database

Frames	EER %	Frames	EER %
5	38.59	**100**	26.60
10	29.23	**125**	25.08
15	31.04	**150**	**23.15**
25	29.18	**175**	23.37
50	25.91	**200**	23.77
75	25.91	**225**	23.70

4.8.1 Experimental Results

In this subsection we are discussing the effectiveness of choosing the FD associated with ML representation. In this context, first we study the effect of frames number, then we discover the superiority of ML compared to MB representation. Finally, we justify our choice of Frame Difference+Multi Level (FD+ML) by showing the results of CASIA Database.

In Table 4.14, we tested the performance on CASIA-FAS database, with respect to different time window size. Especially, we remarked that the average of 6 second (150 frame) gives better result knowing that the video sequences in the CASIA-FASD can reach 10 second. We observe also, that when using a sufficient number of frame ($upto150$) the noise of the shape of fake faces will be detected easily.

Figure 4.22 shows a comparison between representations (ML and MB), texture descriptors (LBP,LPQ and BSIF) and FD using different levels. The EER is presented as a function using different levels. We observe from the Figure that the performance of ML is better than MB when using the same descriptor. This is because the ML representation gives more detailed information of the image than the MB. Also, we observe in this Figure that, when applying the ML or MB on FD image the performance is improved compared to using them to the image directly. We find that the performance of the combination FD+ML is the best among the other combinations. In our tests, we used three descriptor to compare their performances with the FD and the ML representation. We observe from the same Figure that the LPQ feature extractor in our system gives the best results that is a consequence of the blur-invariant property of LPQ. Also, we observe that there is variation in EER according to the number of level. From level 2 to level 8 the performance is improved progressively because each level takes the information

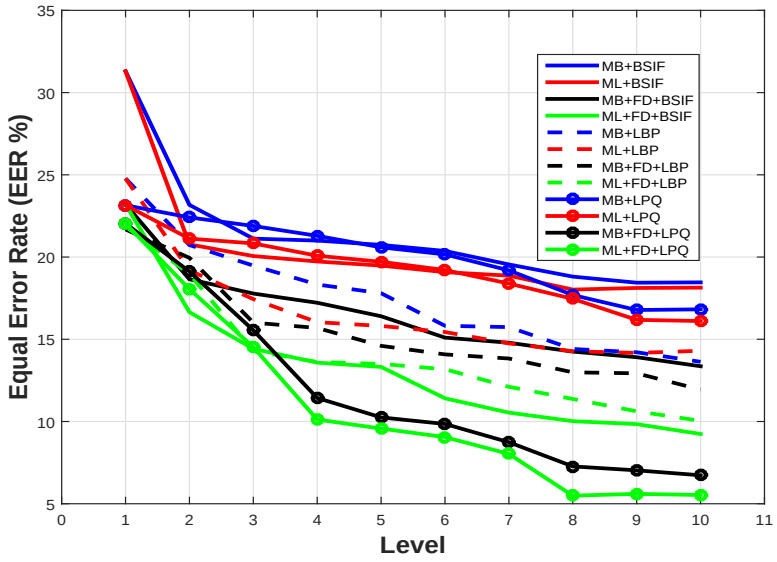


Figure 4.22: Comparison between level number of face representation and FD on CASIA face anti-spoofing.

of the previous levels and the actual level, so it represents more features. Based on the previous analyses, we choose to use FD with ML representation in 8 level. We used also the FS to rank the obtained features and that to improve the results as it is highlighted later on. Next, we will compare our results with the state-of-the-art on CASIA-FAS database. This is summarize in Table 4.15 and Figure 4.23.

To follow the official test protocol of CASIA Face Anti-Spoofing, we computed the EERs for the seven scenarios including different qualities and media. In this section, we will analyze the effects of image qualities and spoofing media on the system performance. Furthermore, we observed that the proposed approach improves the results. The results of our approach using different descriptors are given in Figure 4.23 and Table 4.15.

From the Figure 4.23 (a), we observe that LPQ descriptor combined with FD+ML gives the best results for the different images qualities (low, normal and high), also with spoof media (warped photo, cut photo and video attacks) (See Figure 4.23 (b)). This can be explained by the fact that LPQ works well even in the presence of noise of motion on both real and fake faces compared to LBP and BSIF.

We see from the Table 4.15 that our proposed approach gives better results in all scenario

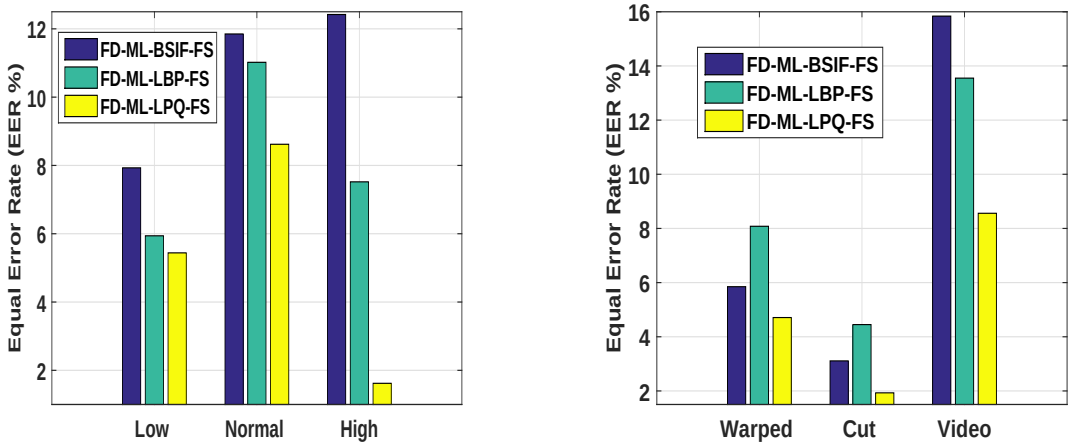


Figure 4.23: Effect of Quality and Spoofing Media on the Performance on the CASIA-FASD. (a) Quality and (b) Spoofing Media

Table 4.15: Comparison between the proposed approach and the state-of-the-the-art methods on different scenario on CASIA Face Anti-Spoofing database

	Scenarios						
Methods	**Low**	**Normal**	**High**	**Warped**	**Cut**	**Video**	**Overall**
IQA [47]	31.70	22.20	05.60	26.10	18.30	34.40	32.40
DoG baseline [71]	13.00	13.00	26.00	16.00	06.00	24.00	17.00
visual codebooks [58]	10.00	17.78	13.33	07.78	22.22	08.89	14.07
LBP-overlapping+fisher [92]	07.20	08.80	14.40	12.00	10.00	14.70	13.10
CDD [43]	**01.50**	**05.00**	02.80	06.40	04.70	**00.30**	11.80
ML-LPQ fisher [90]	12.49	08.96	05.22	13.62	09.66	10.10	11.39
LBP-TOP [57]	10.00	12.00	13.00	06.00	12.00	10.00	10.00
Kernel Fusion [56]	00.70	08.70	13.00	**01.40**	10.10	04.30	07.20
YCbCr+HSV-LBP [46]	07.80	10.10	06.40	07.50	05.40	08.10	06.20
FD-ML-LBP-FS (ours) [88]	05.94	11.02	07.52	08.08	04.45	13.55	10.29
FD-ML-BSIF-FS (ours) [88]	07.93	11.85	12.42	05.85	03.11	15.84	09.96
FD-ML-LPQ-FS (ours) [88]	05.44	08.62	**01.62**	04.71	**01.93**	08.56	**04.62**

compared to CASIA baseline [71] whom create the database. Also, we got the best results in High quality and Cut photos in comparison with the state-of-the-art in the same database. In the case of High quality, our approach can detect effectively the spoof attack because the FD in the case of real faces keeps more information that others qualities (See Figure 4.5). Unlike [57], our approach FD+ML can distinguish easily the cut and real photo because the eye region in the cut photo appears well when using FD. This proves the effectiveness of our approach in face anti-spoofing CASIA database. In the following, we will compare our overall results with the state-of-the-art on three challenge databases: REPLAY-ATTACK, MSU-MFS and CASIA FAS.

We present another experiment about the effectiveness of extracting the texture images using ML representation ($8levels$). We see in Table 4.16 that dividing the whole image on ML improves the robustness of the three descriptors compared to using the whole image. We observe also that the use of ML gives better results on CASIA-FAS, MSU-MFS and Replay-Attack databases. When using ML representation, the EER on CASIA-FASD and MSU-MFSD has been reduced from 23.15% to 17.46% and from 23.22% to 14.90%, respectively. The HTER on the Replay-Attack Database also has been reduced from 15.12% to 12.25%.

We conduct Also an experiment about the effect of the FD on the performance of the ML approach. Table 4.16 shows that applying the ML approaches on the FD improves the performance on the three databases. When using Frame Difference+Multi Level+Local Phase Quantization (FD+ML+LPQ), the performance improvement on CASIA-FAS, MSU-MFS and Replay-Attack databases are 68.55%, 67.78% and 53.06% respectively (see Table 4.16).

Table 4.17 illustrates the effect of features selection on the classification performances. We observe from this Table that using FS method with the FD+ML+LPQ method improves the performance on CASIA-FAS, MSU-MFS and Replay-Attack databases with 15.84 %, 47.91 % and 16.52 % respectively.

4.8.2 Comparison with the state-of-the-art

Tables 4.15 and 4.18 present the comparison of our approach with the state-of-the-art in face anti-spoofing. In Table 4.15, we compared only the results of CASIA-FASD with different

Table 4.16: Effect of the ML on the performance of CASIA, Replay-Attack and MSU databases

Method	CASIA (EER%)	MSU (EER%)	Replay (HTER%)
BSIF	31.30	30.33	23.00
ML-BSIF	18.02	21.85	20.25
FD-ML-BSIF	10.02	08.07	11.66
LBP	24.78	22.12	12.00
ML-LBP	14,27	20.04	09.62
FD-ML-LBP	11,37	07.15	09.70
LPQ	23.15	23.22	15.12
ML-LPQ	17.46	14.90	12.25
FD-ML-LPQ	**05.49**	**04.80**	**05.75**

Table 4.17: Effect of the features selection on the performance of CASIA, Replay-Attack and MSU databases

Method	CASIA (EER%)	MSU (EER%)	Replay (HTER%)
FD+ML-LBP	11,37	07.15	09.70
FD+ML-BSIF	10.02	08.07	11.66
FD+ML-LPQ	05.49	04.80	05.75
FD-ML-LBP-FS	10.29	06.61	08.70
FD-ML-BSIF-FS	09.96	06.14	10.41
FD-ML-LPQ-FS	**04.62**	**02.50**	**04.80**

Table 4.18: Comparison between the proposed countermeasure and the state-of-the-art methods on the three benchmark datasets

Method	CASIA	MSU	Replay-Attack	
	EER%	EER%	EER%	HTER%
IQA [47]	32.40	-	-	15.20
DMD [59]	21.75	-	05.30	03.75
LBP [41]	18.21	-	13.90	13.87
DoG baseline [71]	17.00	-	-	-
Spectral cubes [58]	14.07	-	-	02.75
LBP-overl+fisher [92]	13.10	-	-	-
IDA [44]	12.90	08.58	-	07.41
CDD [43]	11.80	-	-	-
ML-LPQ fisher [90]	11.39	-	-	-
LBP-TOP [57]	10.00	-	07.90	07.60
CNN [96]	07.40	-	06.10	**02.10**
Motion+LBP [61]	-	-	04.50	05.11
Color-LBP [46]	06.20	-	00.40	02.90
Bottleneck Feature Fusion + NN [97]	05.83	-	**00.83**	**00.00**
FD-ML-LPQ-FS (proposed) [88]	**04.62**	**02.50**	05.62	04.80

scenarios, which are: Low, Normal, High, Warped, Cut, Video and Overall test. As we see, our approach gives the best result on the High, Cut and Overall scenarios. In Table 4.18, we observe that our proposed approach Frame Difference+Multi Level+Local Phase Quantization+Fisher-Score (FD+ML+LPQ+FS) gives good results compared with the state-of-the-art on CASIA-FAS and MSU-MFS databases. The EER on CASIA, MSU and REPLAY databases respectively is: 04.62%, 02.50% and 5.62% (See DET curve Figure 4.24). In the case of Replay-Attack database, our method shows interesting results compared to the other methods.

4.8.3 Cross-Database Analysis

To gain insight into the generalization capabilities of our proposed method, we conducted a cross-database evaluation. To be clear, cross-database is a technique when we trained and tuned on one database and test on another database. There are different techniques on analysis, where training and testing occur in distinct databases. In our work, we follow cross-database used in these papers [58, 60, 96, 98]. In these experiments, the countermeasure was trained and tuned with one database each time (CASIA-FAS, MSU MFS or Replay-Attack) and then tested on the

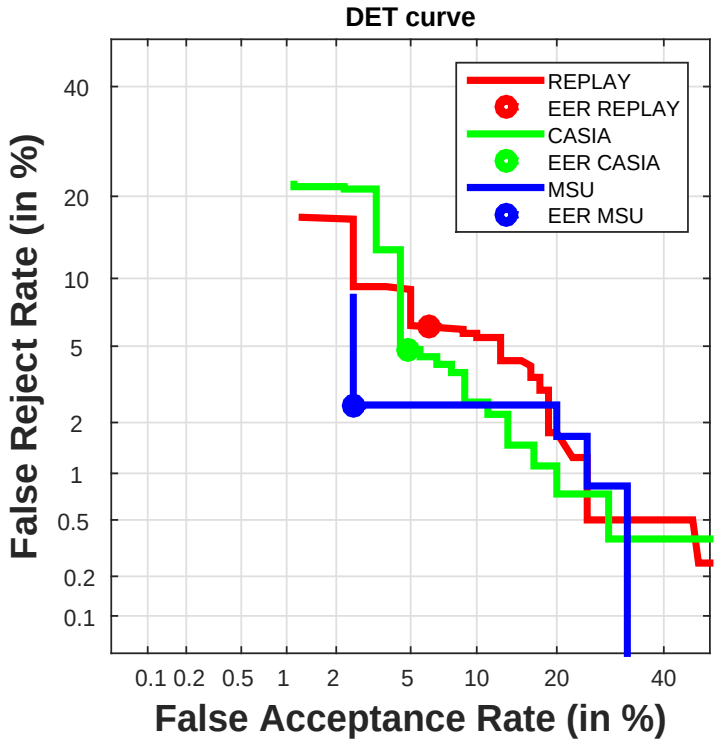


Figure 4.24: DET curve of the proposed approach on REPLAY, CASIA and MSU databases.

other databases. The results are reported in Table 4.19.

We see from Table 4.19, when the model is trained and tuned on CASIA database, then evaluated on the other databases, that the average of performance in terms of HTER for the train, development and test sets on the Replay-Attack database is 51.74% and on MSU database, the average of performance for the train and test sets is 50.76%. When the model is trained and tuned on MSU database and evaluated on the other databases, the average of performance in term of HTER for the train, development and test sets on the Replay-Attack database is 47.33% and on CASIA database, the average performance for train and test sets is 48.64%. Finally, when the model is trained and tuned on Replay-Attack database then evaluated on the other databases, the average of performance in term of HTER for train and test sets are 42.82% and 36.50% on CASIA and on MSU databases respectively. As we observe in Table 4.19, the models trained on Replay-Attack and MSU MFSD Databases give better results than the model trained on CASIA-FASD. The reason why CASIA-FASD is not good as train set compared to other databases on face anti-spoofing because it has different qualities and attacks. In Table 4.20,

Table 4.19: The performance of the cross-database evaluation in terms of HTER(%) on the CASIA-FAS, MSU-MFS and REPLAY-ATTACK

Test on:	CASIA		MSU		Replay-Attack		
Train on:	Train	Test	Train	Test	Train	Dev	Test
CASIA	-	-	51.11	50.41	53.16	51.83	50.25
MSU	47.29	50.00	-	-	46.16	47.83	48.00
Replay-Attack	43.05	42.59	35.00	38.00	-	-	-

Table 4.20: The results of the cross-database experiment on the CASIA-FAS, REPLAY-ATTACK and MSU-MFS database compared with related studies

Method	**Train:**	**Test:**	**HTER %**
Motion [98]	CASIA	Replay	50.20
	Replay	CASIA	47.90
LBP [98]	CASIA	Replay	45.90
	Replay	CASIA	57.60
LBP-TOP [98]	CASIA	Replay	49.70
	Replay	CASIA	60.60
Motion-Mag [60]	CASIA	Replay	50.10
	Replay	CASIA	47.00
Spectral cubes [58]	CASIA	Replay	**34.40**
	Replay	CASIA	50.00
CNN [96]	CASIA	Replay	48.50
	Replay	CASIA	45.50
Proposed [88]	CASIA	Replay	50.25
		MSU	**50.41**
	Replay	CASIA	**42.59**
		MSU	**38.00**
	MSU	CASIA	**50.00**
		Replay	**48.00**

we present the results of our proposed approach compared to the state-of-the-art techniques on cross-database. We observe in Table 4.20, that when we use ML and FD, the performance is affected on face anti-spoofing methods, especially on cross-database compared with the state-of-the-art.

4.9 Conclusion

In this chapter, we presented our approach of face anti-spoofing on 2D image and video. The proposed approach based on face alignment, frame-difference, features extraction, face representation, features selection gives us a good framework on face anti-spoofing. We applied then, our methods on different data modalities. Quantitative comparison (in the context of biometrics spoof) of the proposed framework and the state-of-the-art of NUAA, CASIA, REPLAY ATTACK, MSU and OULU-NPU databases is provided, which can guide the deployment of existing algorithms and the development of new face recognition methods toward more practical systems.

5

Conclusions and perspectives

Contents

5.1 Conclusions

Although face recognition has been investigated extensively, it still suffers from variations due to various factors in real-world scenarios. Each day, sensing technologies advance and the acquisition devices become more accurate and less expensive. 3D face recognition evades illumination and poses problems, however, there are still challenges such as spoofing attacks or disguise variations which affect the performances of both 2D and 3D face recognition.

In this thesis, we have investigated spoofing attacks and disguise variations in face recognition and proposed countermeasure techniques for the protection of face recognition systems against these challenges. In the first part of the thesis, we have explored the topic of spoofing in face recognition. Spoofing is a very new topic for researchers in face biometrics domain. Therefore studies on this topic are limited. It has been shown that face recognition systems are vulnerable to photograph and video attacks. This is why countermeasure techniques are necessary to mitigate the impact of spoofing on face recognition. In Chapter 4, we proposed a countermeasure technique, which is based on texture and contrast analysis, for the detecting of spoof attacks.

The goal of this thesis is to go beyond traditional research in biometric and to investigate a novel approach in face anti-spoofing measures to develop a more robust biometric system. To validate these approach, this thesis will address face spoofing detection with 2D images and videos databases.

It was shown that the static and dynamic characteristic differences between genuine faces and fake ones, such as shading, specular reflections, quality and motion patterns, can be exploited for generic face anti-spoofing using texture feature descriptors, such as LBP, LPQ, and BSIF. The proposed approaches have been successfully applied using three descriptors based on FD and face representation getting best-performing algorithms. Our approach is also based on face alignment applied before features extraction and Fisher-Score used to select and reduce features. These aspects improve clearly the results of our system. The performance of the proposed texture based countermeasures is very encouraging when following the intra-test protocols of the publicly available databases, i.e. when the operating conditions are known.

5.2 Perspectives and future work

The research community has just begun to focus on the problem of spoofing attacks and the current publicly available databases have been a very important kick-off for finding out best practices for spoof detection. The impressive results on the existing benchmark data-sets indicate that more challenging configurations are needed before the research on non-intrusive face anti-spoofing can reach the next level.

In the future, more work should be carried out for designing and collecting new databases with more representative and diverse development set but still with unseen scenarios in the test set simulating the unknown attacks that will be faced in real operational conditions. In addition to variations in the collected data, well-defined test protocols with clear training, development and test sets are needed for an unbiased comparison between various approaches across different databases. Also, we will test our proposed methods using different color image representations instead of the grayscale images. We will use other descriptors, such as Scale Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF), to test the effectiveness of our framework in face anti-spoofing detection. Moreover, we envision the improvement of the face alignment process on 3D face, to investigate on 3D Mask Face Anti-spoofing.

Bibliography

[1] J. Galbally, S. Marcel, and J. Fierrez, "Biometric antispoofing methods: A survey in face recognition," *IEEE Access*, vol. 2, pp. 1530–1552, 2014.

[2] J. Komulainen, "Software-based countermeasures to 2d facial spoofing attacks," PhD dissertation, University of Oulu, faculty of information technology and electrical engineering, department of computer science and engineering, 2015.

[3] F. L. ALEGRE, "La protection des systèmes de reconnaissance de locuteur contre le leurrage," PhD dissertation, Institut Mines-Telecom, ParisTech, University of Paris-Saclay, 2014.

[4] S. M. Matyas and J. Stapleton, "A biometric standard for information management and security," *Computers & Security*, vol. 19, no. 5, pp. 428 – 441, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016740480005029X

[5] ANSI-X9.84-2010, "Biometric information management and security for the financial services industry," *ANSI X9.84-2010*, 2010.

[6] L. O'Gorman, "Comparing passwords, tokens, and biometrics for user authentication," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2019–2020, Dec 2003.

[7] I. Chingovska, "Trustworthy biometric verification under spoofing attacks: Application to the face mode," PhD dissertation, École polytechnique fédérale de Lausanne EPFL, 2015.

[8] N. Kose, "Spoofing and disguise variations in face recognition," PhD dissertation, Institut Mines-Telecom, ParisTech, University of Paris-Saclay, 2014.

[9] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2d and 3d face recognition: A survey," *Pattern Recognition Letters*, vol. 28, no. 14, pp. 1885 – 1906, 2007, image: Information and Control. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865507000189

[10] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of artificial "gummy" fingers on fingerprint systems," vol. 4677, 2002, pp. 4677 – 4677 – 15. [Online]. Available: http://dx.doi.org/10.1117/12.462719

[11] N. M. Duc and B. Q. Minh, "Your face is not your password face authentication bypassing lenovo–asus–toshiba," *Black Hat Briefings*, 2009.

[12] F. Reiger, "Chaos computer club breaks apple touchid," *Blog post*, vol. 21, 2013.

[13] "Million dollar border security machines fooled with ten cent tape," http://findbiometrics.com/million-dollar-border-security-machines-fooled-with-ten-cent-tape/#, accessed: January 8, 2009.

[14] "Doctors used fake fingers to clock in for colleagues at er," https://www.cnet.com/news/doctors-used-fake-fingers-to-clock-in-for-colleagues-at-er/, accessed: March 13, 2013.

[15] W. Xiaomin, X. TaiHua, and Z. Wenfang, "Chaos-based biometrics template protection and secure authentication," in *State of the art in Biometrics*, J. Yang and L. Nanni, Eds. Rijeka: InTech, 2011, ch. 15. [Online]. Available: http://dx.doi.org/10.5772/19599

[16] W. R. Schwartz, A. Rocha, and H. Pedrini, "Face spoofing detection through partial least squares and low-level descriptors," in *2011 International Joint Conference on Biometrics (IJCB)*, Oct 2011, pp. 1–8.

[17] G. Pan, L. Sun, and Z. Wu, *Liveness detection for face recognition*. INTECH Open Access Publisher, 2008.

[18] R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori, S. Ricerche, and F. Roli, "Fusion of multiple clues for photo-attack detection in face recognition systems," in *2011 International Joint Conference on Biometrics (IJCB)*, Oct 2011, pp. 1–6.

[19] T. de Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel, "Lbp – top based countermeasure against face spoofing attacks," in *Computer Vision - ACCV 2012 Workshops*, J.-I. Park and J. Kim, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 121–132.

[20] B. Toth and U. C. von Seelen, "Liveness detection for iris recognition," in *The Presentation Sheet of NIST Workshop, Biometrics and E-Authentication over Open Networks*, 2005.

[21] A. Jain, P. Flynn, and A. A. Ross, *Handbook of biometrics*. Springer Science & Business Media, 2007.

[22] A. d. S. Pinto, H. Pedrini, W. Schwartz, and A. Rocha, "Video-based face spoofing detection through visual rhythm analysis," in *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, Aug 2012, pp. 221–228.

[23] K. A. Nixon, V. Aimale, and R. K. Rowe, "Spoof detection schemes," in *Handbook of biometrics*. Springer, 2008, pp. 403–423.

[24] R. Derakhshani, "Spoof-proofing fingerprint systems using evolutionary time-delay neural networks," in *CIHSPS 2005. Proceedings of the 2005 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety, 2005.*, March 2005, pp. 98–104.

[25] F. Monrose, M. K. Reiter, Q. Li, D. P. Lopresti, and C. Shih, "Toward speech-generated cryptographic keys on resource-constrained devices." in *USENIX Security Symposium*, 2002, pp. 283–296.

[26] K. Kollreider, H. Fronthaler, and J. Bigun, "Evaluating liveness by face images and the structure tensor," in *Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID'05)*, Oct 2005, pp. 75–80.

[27] J. Li, Y. Wang, T. Tan, and A. K. Jain, "Live face detection based on the analysis of fourier spectra," in *Biometric Technology for Human Identification*, vol. 5404. International Society for Optics and Photonics, Aug. 2004, pp. 296–304.

[28] B. Toth, "Biometric liveness detection," *Information Security Bulletin*, vol. 10, no. 8, pp. 291–297, 2005.

[29] R. V. Yampolskiy, "Mimicry attack on strategy-based behavioral biometric," in *Fifth International Conference on Information Technology: New Generations (itng 2008)*, April 2008, pp. 916–921.

[30] E. S. Ng and A. Y. S. Chia, "Face verification using temporal affective cues," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Nov 2012, pp. 1249–1252.

[31] I. Pavlidis and P. 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)*, 2000, pp. 15–24.

[32] G. Chetty and M. Wagner, "Liveness" verification in audio-video authentication," in *Proceedings of the 10th Australian International Conference on Speech Science and Technology (SST'04)*, 2004, pp. 358–363.

[33] N. Erdogmus and S. Marcel, "Spoofing 2d face recognition systems with 3d masks," in *2013 International Conference of the BIOSIG Special Interest Group (BIOSIG)*, no. EPFL-CONF-192407, Sept 2013, pp. 1–8.

[34] D. F. Smith, A. Wiliem, and B. C. Lovell, "Face recognition on consumer devices: Reflections on replay attacks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 736–745, April 2015.

[35] T. Wang, J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection using 3d structure recovered from a single camera," in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–6.

[36] D. Yi, Z. Lei, Z. Zhang, and S. Z. Li, "Face anti-spoofing: Multi-spectral approach," in *Handbook of Biometric Anti-Spoofing: Trusted Biometrics under Spoofing Attacks*, S. Marcel, M. S. Nixon, and S. Z. Li, Eds. London: Springer London, 2014, pp. 83–102. [Online]. Available: https://doi.org/10.1007/978-1-4471-6524-8_5

[37] Y. Kim, J. Na, S. Yoon, and J. Yi, "Masked fake face detection using radiance measurements," *J. Opt. Soc. Am. A*, vol. 26, no. 4, pp. 760–766, Apr 2009. [Online]. Available: http://josaa.osa.org/abstract.cfm?URI=josaa-26-4-760

[38] F. J. Prokoski and R. B. Riedel, "Infrared identification of faces and body parts," in *Biometrics: Personal Identification in Networked Society*, A. K. Jain, R. Bolle, and S. Pankanti, Eds. Boston, MA: Springer US, 1996, pp. 191–212. [Online]. Available: https://doi.org/10.1007/0-306-47044-6_9

[39] J. Bai, T. T. Ng, X. Gao, and Y. Q. Shi, "Is physics-based liveness detection truly possible with a single image?" in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, May 2010, pp. 3425–3428.

[40] J. Maatta, A. Hadid, and M. Pietikainen, "Face spoofing detection from single images using texture and local shape analysis," *IET Biometrics*, vol. 1, no. 1, pp. 3–10, March 2012.

[41] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *2012 BIOSIG - Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG)*, Sept 2012, pp. 1–7.

[42] N. Kose and J. L. Dugelay, "Classification of captured and recaptured images to detect photograph spoofing," in *2012 International Conference on Informatics, Electronics Vision (ICIEV)*, May 2012, pp. 1027–1032.

[43] J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection with component dependent descriptor," in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–6.

[44] D. Wen, H. Han, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 746–761, April 2015.

[45] K. Patel, H. Han, A. K. Jain, and G. Ott, "Live face video vs. spoof face video: Use of moiré; patterns to detect replay video attacks," in *2015 International Conference on Biometrics (ICB)*, May 2015, pp. 98–105.

[46] Z. Boulkenafet, J. Komulainen, and A. Hadid, "Face anti-spoofing based on color texture analysis," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 2636–2640.

[47] J. Galbally and S. Marcel, "Face anti-spoofing based on general image quality assessment," in *2014 22nd International Conference on Pattern Recognition*, Aug 2014, pp. 1173–1178.

[48] J. Galbally, S. Marcel, and J. Fierrez, "Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 710–724, Feb 2014.

[49] J. Yang, Z. Lei, D. Yi, and S. Z. Li, "Person-specific face antispoofing with subject domain adaptation," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 797–809, April 2015.

[50] I. Chingovska and A. R. dos Anjos, "On the use of client identity information for face antispoofing," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 787–796, April 2015.

[51] B. Biggio, Z. Akhtar, G. Fumera, G. L. Marcialis, and F. Roli, "Security evaluation of biometric authentication systems under real spoofing attacks," *IET Biometrics*, vol. 1, no. 1, pp. 11–24, March 2012.

[52] A. Anjos and S. Marcel, "Counter-measures to photo attacks in face recognition: A public database and a baseline," in *2011 International Joint Conference on Biometrics (IJCB)*, Oct 2011, pp. 1–7.

[53] O. Kähm and N. Damer, "2d face liveness detection: An overview," in *2012 BIOSIG - Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG)*, Sept 2012, pp. 1–12.

[54] K. Kollreider, H. Fronthaler, and J. Bigun, "Verifying liveness by multiple experts in face biometrics," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2008, pp. 1–6.

[55] W. Bao, H. Li, N. Li, and W. Jiang, "A liveness detection method for face recognition based on optical flow field," in *2009 International Conference on Image Analysis and Signal Processing*, April 2009, pp. 233–236.

[56] S. R. Arashloo, J. Kittler, and W. Christmas, "Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 11, pp. 2396–2407, Nov 2015.

[57] T. d. Freitas Pereira, J. Komulainen, A. Anjos, J. M. De Martino, A. Hadid, M. Pietikäinen, and S. Marcel, "Face liveness detection using dynamic texture," *EURASIP Journal on Image and Video Processing*, vol. 2014, no. 1, p. 2, Jan 2014. [Online]. Available: https://doi.org/10.1186/1687-5281-2014-2

[58] A. Pinto, H. Pedrini, W. R. Schwartz, and A. Rocha, "Face spoofing detection through visual codebooks of spectral temporal cubes," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4726–4740, Dec 2015.

[59] S. Tirunagari, N. Poh, D. Windridge, A. Iorliam, N. Suki, and A. T. S. Ho, "Detection of face spoofing using visual dynamics," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 762–777, April 2015.

[60] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh, "Face anti-spoofing via motion magnification and multifeature videolet aggregation," 2014.

[61] J. Komulainen, A. Hadid, M. Pietikäinen, A. Anjos, and S. Marcel, "Complementary countermeasures for detecting scenic face spoofing attacks," in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–7.

[62] K. Kollreider, H. Fronthaler, M. I. Faraj, and J. Bigun, "Real-time face detection and motion analysis with application in "liveness" assessment," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 548–558, Sept 2007.

[63] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *2007 IEEE 11th International Conference on Computer Vision*, Oct 2007, pp. 1–8.

[64] K. Kollreider, H. Fronthaler, and J. Bigun, "Non-intrusive liveness detection by face images," *Image and Vision Computing*, vol. 27, no. 3, pp. 233 – 244, 2009, special Issue on Multimodal Biometrics. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885607000893

[65] D. C. Garcia and R. L. de Queiroz, "Face-spoofing 2d-detection based on moiré;-pattern analysis," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 778–786, April 2015.

[66] J. Galbally and R. Satta, "Three-dimensional and two-and-a-half-dimensional face recognition spoofing using three-dimensional printed models," *IET Biometrics*, vol. 5, no. 2, pp. 83–91, 2016.

[67] N. Erdogmus and S. Marcel, "Spoofing face recognition with 3d masks," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 7, pp. 1084–1097, July 2014.

[68] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcão, and A. Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 864–879, April 2015.

[69] X. Tan, Y. Li, J. Liu, and L. Jiang, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 504–517.

[70] A. Anjos, M. M. Chakka, and S. Marcel, "Motion-based counter-measures to photo attacks in face recognition," *IET Biometrics*, vol. 3, no. 3, pp. 147–158, Sept 2014.

[71] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li, "A face antispoofing database with diverse attacks," in *2012 5th IAPR International Conference on Biometrics (ICB)*, March 2012, pp. 26–31.

[72] Z. Boulkenafet, J. Komulainen, L. Li, X. Feng, and A. Hadid, "Oulu-npu: A mobile face presentation attack database with real-world variations," in *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, May 2017, pp. 612–618.

[73] B. Yang and S. Fotios, "Lighting and recognition of emotion conveyed by facial expressions," *Lighting Research & Technology*, vol. 47, no. 8, pp. 964–975, 2015. [Online]. Available: https://doi.org/10.1177/1477153514547753

[74] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004. [Online]. Available: https://doi.org/10.1023/B:VISI.0000013087.49260.fb

[75] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, Jan 1998, pp. 555–562.

[76] X. Tan, F. Song, Z. H. Zhou, and S. Chen, "Enhanced pictorial structures for precise eye localization under incontrolled conditions," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 1621–1628.

[77] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51 – 59, 1996. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0031320395000674

[78] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, Jul 2002.

[79] T. Ojala, K. Valkealahti, E. Oja, and M. Pietikäinen, "Texture discrimination with multidimensional distributions of signed gray-level differences," *Pattern Recognition*, vol. 34, no. 3, pp. 727 – 739, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320300000108

[80] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Image and Signal Processing*, A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 236–243.

[81] J. Kannala and E. Rahtu, "Bsif: Binarized statistical image features," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Nov 2012, pp. 1363–1366.

[82] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. John Wiley & Sons, New York, 2001.

[83] A. Benlamoudi, A. Samai, A. Ouafi, A. Taleb-Ahmed, S. Bekhouche, and A. Hadid, "Face spoofing detection from single images using active shape models with stasm and lbp," in *Troisième Conférence internationale sur la Vision Artificielle CVA' 2015*, Algerie, Tizi Ouzou, Apr 2015.

[84] S. Bekhouche, A. Ouafi, A. Taleb-Ahmed, A. Hadid, and A. Benlamoudi, "Facial age estimation using bsif and lbp," in *Proceeding of the first International Conference on Electrical Engineering ICEEB'14*, Algerie, Biskra, Dec 2014.

[85] S. Milborrow, "Active shape models with stasm," *Stasm Version*, vol. 3, 2009.

[86] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *2011 International Joint Conference on Biometrics (IJCB)*, Oct 2011, pp. 1–7.

[87] B. Peixoto, C. Michelassi, and A. Rocha, "Face liveness detection under bad illumination conditions," in *2011 18th IEEE International Conference on Image Processing*, Sept 2011, pp. 3557–3560.

[88] A. Benlamoudi, K. E. Aiadi, A. Ouafi, D. Samai, and M. Oussalah, "Face antispoofing based on frame difference and multilevel representation," *Journal of Electronic Imaging*, vol. 26, no. 4, p. 043007, 2017. [Online]. Available: http://dx.doi.org/10.1117/1.JEI.26.4.043007

[89] C. E. Shannon, W. Weaver, and A. W. Burks, "The mathematical theory of communication," 1951.

[90] A. Benlamoudi, D. Samai, A. Ouafi, S. E. Bekhouche, A. Taleb-Ahmed, and A. Hadid, "Face spoofing detection using multi-level local phase quantization (ml-lpq)," in *International Conference on Automatic control, Telecommunications and Signals (ICATS15)*, Algerie, Annaba, Nov 2015, pp. 1–5.

[91] A. Benlamoudi, F. Bougourzi, M. Zighem, S. Bekhouche, A. Ouafi, and A. Taleb-Ahmed, "Face anti-spoofing combining mllbp and mlbsif," in *10ème Conférence sur le Génie Electrique*, Algerie, Alger, Apr 2017.

[92] A. Benlamoudi, D. Samai, A. Ouafi, S. E. Bekhouche, A. Taleb-Ahmed, and A. Hadid, "Face spoofing detection using local binary patterns and fisher score," in *2015 3rd International Conference on Control, Engineering Information Technology (CEIT)*, Algerie, Tlemcen, May 2015, pp. 1–5.

[93] J. Yang, Z. Lei, S. Liao, and S. Z. Li, "Face liveness detection with component dependent descriptor," in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–6.

[94] Z. Boulkenafet, J. Komulainen, Z. Akhtar, A. Benlamoudi, D. Samai, S. Bekhouche, A. Ouafi, A. Taleb-Ahmed, L. Qin, F. Peng, L. Zhang, M. Long, S. Bhilare, V. Kanhangad, E. Vazquez-Fernandez, D. Perez-Cabo, J. J. Moreira-Perez, D. G. alez-Jim enez andS. Bhattacharjee, S.Marcel, S. Volkova, Y. Tang, N. Abe, L. Li, X. Feng, Z. Xia, F. Dornaika, A. Costa-Pazo, A. Mohammadi, X. Jiang, R. Shao, P. C. Yuen, W. Almeida, F. Andal, R. Padilha, G. Bertocco, W. Dias, J. Wainer, A. Rocha, M. A. Angeloni, G. Folego, A. Godoy, and A. Hadid, "A competition on generalized software-based face presentation attack detection in mobile scenarios," in *International Joint Conference on Biometrics, 2017 IEEE Sixth International Conference on*. IEEE, 2017, pp. 1–9.

[95] S. E. Bekhouche, A. Ouafi, A. Benlamoudi, A. Taleb-Ahmed, and A. Hadid, "Facial age estimation and gender classification using multi level local phase quantization," in *2015 3rd International Conference on Control, Engineering Information Technology (CEIT)*, Algerie, Tlemcen, May 2015, pp. 1–4.

[96] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," *CoRR*, vol. abs/1408.5601, 2014. [Online]. Available: http://arxiv.org/abs/1408.5601

[97] L. Feng, L.-M. Po, Y. Li, X. Xu, F. Yuan, T. C.-H. Cheung, and K.-W. Cheung, "Integration of image quality and motion cues for face anti-spoofing: A neural network approach," *Journal of Visual Communication and Image Representation*, vol. 38, pp.

451 – 460, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320316300244

[98] T. de Freitas Pereira, A. Anjos, J. M. D. Martino, and S. Marcel, "Can face anti-spoofing countermeasures work in a real world scenario?" in *2013 International Conference on Biometrics (ICB)*, June 2013, pp. 1–8.

A

Code of Project

Listing A.1: faces_alignement.m

```
function [face, IMG_rotate]=faces_alignement(IMG,eyes,cof1,cof2,
    cof3)
% IMG --> input image (face)
% eyes --> cordinate of eyes (x,y)
% cof1,cof2,cof3 --> ktop, kside and kbootom
pos = [eyes(1,1) eyes(2,1); eyes(1,2) eyes(2,2)]; %[Leyex Reyex;
    Leyey Reyey];
[m ,n, ~]=size(IMG);
tg_a=diff(pos(2,:))/diff(pos(1,:));clear pos;
angle=tg_a*(180/pi);
```

```
tg_a = -angle * (pi/180);
IMG_rotate = imrotate(IMG, angle);
[m1 ,n1 ,~]=size(IMG_rotate);
axe_x=[eyes(1,1) eyes(2,1)];
axe_y=[eyes(1,2) eyes(2,2)];
Rx=axe_x(:);Ry=axe_y(:);clear axe_x axe_y
% figure(2),imshow(IMG);
% hold on;
% plot(eyes(1,1), eyes(1,2),'g*','LineWidth',2);
%pause;
Ex = (n1-n)/2;
Ey = (m1-m)/2;
Cx=n/2;Cy=m/2;
New_lan_x= (Cx+(Rx-Cx)*cos(tg_a)-(Ry-Cy)*sin(tg_a))+Ex;
New_lan_y= (Cy+(Rx-Cx)*sin(tg_a)+(Ry-Cy)*cos(tg_a))+Ey;
% figure(12),imshow(IMG_rotate);
% hold on;
% plot(New_lan_x,New_lan_y,'-','LineWidth',2);
% plot(New_lan_x,New_lan_y,'g+','LineWidth',2);
right_eyex = New_lan_x(1,1);left_eyex = New_lan_x(2,1);
right_eyey = New_lan_y(1,1);left_eyey = New_lan_y(2,1);
%pause
%figure(3),imshow(IMG_rotate);
%hold on;
%plot(right_eyex,right_eyey,'r+','LineWidth',2);
%plot(left_eyex,left_eyey,'g+','LineWidth',2);
x=[ right_eyex left_eyex ];y=[right_eyey left_eyey ];
pos = [x(1,2) x(1,1); y(1,2) y(1,1)];
A=abs(diff(pos(1,:)));
xxx=[x(1,1)-(A/cof1) x(1,2)-(A/cof1) x(1,1)+(A/cof1) x(1,2)+(A/
    cof1)];
```

```
39 yyy=[y(1,1)-(A/cof2) y(1,2)-(A/cof2) y(1,1)+(A*cof3) y(1,2)+(A*
       cof3)];
40 maxx = max(xxx); maxy = max(yyy); minx = min(xxx); miny = min(yyy)
       ;
41 face = imcrop(IMG_rotate,[minx miny maxx-minx maxy-miny]);%clear
       IMG_rotate;
```

Listing A.2: `Multi_block.m`

```
1  function HIST = Multi_block(img,num_blk,mapp);
2  % img --> input image
3  % num_blk --> number of blocks
4  % mapp --> number of features
5  [m,n] = size(img);
6  HIST = [];
7  H = floor(m/num_blk);
8  W = floor(n/num_blk);
9  HL = mod(m,H);
10 WL = mod(n,W);
11 for mm = 1:H:m-HL
12     for nn = 1:W:n-WL
13         X = img(mm:mm+H-1,nn:nn+W-1,:);
14         h = hist(X(:),1:mapp);
15         HIST = [HIST reshape(h,1,[])];
16     end
17 end
```

Listing A.3: `Multi_level.m`

```
1  clear;close all;fclose all ;clc;
2  %% Toolbox
```

```
3  addpath(genpath('toolbox\'));
4  %%
5  load NUAA_DB; % information of the database
6  data_NUM = length(data);
7  DIR = 'F:\\NUAA\\NUAA_DB_Stasm\\'; % path of the database
8  IMG_Type = 'jpg'; % type of the images
9  discriptors={'LBP','LPQ','BSIF'};
10 shape={'block','level'}; % type of descriptors
11 tic
12 for sh = 1:2
13     for num_blocks =1:8
14         for F =1:size(discriptors,2)
15             filename=sprintf('%s_params',discriptors{F});
16             load(filename)
17             for jj = 1:size(params,2)
18                 tic
19                 type = sprintf('%s_%s_%d_%d_%d_%d_u2',discriptors{
                       F},shape{sh},num_blocks,params(jj).a,params(jj)
                       .b,params(jj).c);
20                 name = sprintf('Features\\%s\\%s',discriptors{F},
                       type);
21                 opFolder = fullfile(cd, name);
22                 if ~exist(opFolder, 'dir')
23                     mkdir(opFolder);
24                 end
25                 for i=1:data_NUM
26                     disp(i)
27                     imgname = sprintf(strcat(DIR, '\\',num2str(
                           data(i).name)...
28                          ,'.',IMG_Type));
29                     img=imread(imgname);
```

```
30                  Features=[];
31                  if(strcmp(discriptors{F},'LBP'))
32                      mapp = params(jj).b * (params(jj).b -1) +
                            3;
33                      filename=sprintf('mapping_u2_%d',params(jj
                            ).b);
34                      load(filename)
35                      I=lbp(img,params(jj).c,params(jj).b,
                            mapping,'i');
36                  elseif(strcmp(discriptors{F},'LPQ'))
37                      mapp = 256;
38                      I = lpq(img,params(jj).a,params(jj).b,
                            params(jj).c,'im');
39                  elseif(strcmp(discriptors{F},'BSIF'))
40                      filename=sprintf('texturefilters/
                            ICAtextureFilters_%dx%d_%dbit',params(
                            jj).a,params(jj).b,params(jj).c);
41                      load(filename, 'ICAtextureFilters');
42                      numScl=size(ICAtextureFilters,3);
43                      mapp = 2^numScl;
44                      I= bsif(img,ICAtextureFilters,'im');
45                  end
46                  if(strcmp(shape{sh},'block'))
47                      Features = [Features Multi_block(I,
                            num_blocks,mapp)];
48                  else
49                      Features = [Features Multi_level(I,
                            num_blocks,mapp)];
50                  end
51                  name_image = sprintf('%s/%s',name,data(i).name
                    );
```

```
52                      save(sprintf('%s.mat',name_image),'Features','
                           -v7.3');clear Features;
53                      clear Features
54                  end
55                  fprintf('%s done in %.2f second \n',name, toc);
56              end
57          end
58      end
59  end
```

Listing A.4: `Multi_level.m`

```
1  clear;close all;fclose all ;clc;
2  %% Toolbox
3  addpath(genpath('toolbox\'));
4  %%
5  load NUAA_DB; % information of the database
6  data_NUM = length(data);
7  DIR = 'F:\\NUAA\\NUAA_DB_Stasm\\'; % path of the database
8  IMG_Type = 'jpg'; % type of the images
9  discriptors={'LBP','LPQ','BSIF'};
10 shape={'block','level'}; % type of descriptors
11 tic
12 for sh = 1:2
13     for num_blocks =1:8
14         for F =1:size(discriptors,2)
15             filename=sprintf('%s_params',discriptors{F});
16             load(filename)
17             for jj = 1:size(params,2)
18                 tic
19                 type = sprintf('%s_%s_%d_%d_%d_%d_u2',discriptors{
```

```
F},shape{sh},num_blocks,params(jj).a,params(jj)
.b,params(jj).c);
20 name = sprintf('Features\\%s\\%s',discriptors{F},
type);
21 opFolder = fullfile(cd, name);
22 if ~exist(opFolder, 'dir')
23 mkdir(opFolder);
24 end
25 for i=1:data_NUM
26 disp(i)
27 imgname = sprintf(strcat(DIR, '\\',num2str(
data(i).name)...
28 ,'.',IMG_Type));
29 img=imread(imgname);
30 Features=[];
31 if(strcmp(discriptors{F},'LBP'))
32 mapp = params(jj).b * (params(jj).b -1) +
3;
33 filename=sprintf('mapping_u2_%d',params(jj
).b);
34 load(filename)
35 I=lbp(img,params(jj).c,params(jj).b,
mapping,'i');
36 elseif(strcmp(discriptors{F},'LPQ'))
37 mapp = 256;
38 I = lpq(img,params(jj).a,params(jj).b,
params(jj).c,'im');
39 elseif(strcmp(discriptors{F},'BSIF'))
40 filename=sprintf('texturefilters/
ICAtextureFilters_%dx%d_%dbit',params(
jj).a,params(jj).b,params(jj).c);
```

```
41                          load(filename, 'ICAtextureFilters');
42                          numScl=size(ICAtextureFilters,3);
43                          mapp = 2^numScl;
44                          I= bsif(img,ICAtextureFilters,'im');
45                      end
46                      if(strcmp(shape{sh},'block'))
47                          Features = [Features Multi_block(I,
                                num_blocks,mapp)];
48                      else
49                          Features = [Features Multi_level(I,
                                num_blocks,mapp)];
50                      end
51                      name_image = sprintf('%s/%s',name,data(i).name
                            );
52                      save(sprintf('%s.mat',name_image),'Features','
                            -v7.3');clear Features;
53                      clear Features
54                  end
55                  fprintf('%s done in %.2f second \n',name, toc);
56              end
57          end
58      end
59  end
```