Trustworthy Biometric Verification under Spoofing Attacks: Application to the Face Mode

THÈSE N^{O} 6879 (2016)

PRÉSENTÉE LE 2 MAI 2016

À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE DE L'IDIAP

PROGRAMME DOCTORAL EN GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Ivana CHINGOVSKA

acceptée sur proposition du jury:

Prof. S. Süsstrunk, présidente du jury
Prof. H. Bourlard, Dr S. Marcel, directeurs de thèse
Prof. J. Fierrez, rapporteur
Prof. R. Veldhuis, rapporteur
Prof. J.-Ph. Thiran, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
Suisse
2016

Acknowledgements

Thank you, Sébastien, for giving me the opportunity to be a part of the Biometrics group. Thank you for generously sharing your outstanding expertise in biometrics and taking the discussions straight to the point. Thank you for always being supportive, positive and encouraging. Most of all, thank you for leading a great team down a challenging road, but always making sure that it is a clear and rewarding one for us all.

Thank you, André, for always taking the problems out of the box and inspiring me to do the same. Thank you for selflessly sharing your great knowledge, experience and ideas with the whole team. Thank you for never being satisfied with anything less than perfect, no matter whether it is a paper, code or documentation. I admit, it was not always easy to implement your demands, but I can't express how much I appreciate it now.

Thank you, thesis jury, Prof. Bourlard, Prof. Süsstrunk, Prof. Fierrez, Prof. Veldhuis and Prof. Thiran, for taking the time to read my thesis and give insightful and constructive comments. It was a pleasure to discuss with you, and your comments have surely contributed to the quality of this thesis.

Thank you, Laurent, for all the discussions, suggestions and especially endless debugging sessions in so many occasions. You have been a true inspiration for me not only in programming and research, but also with your genuine enthusiasm to help people. I'll remember and try to exercise that skill every time I have the opportunity! Thank you, everyone in the Biometrics team: Manuel, Elie, Tiago, Nesli, Pedro, Matthias, Chris, Roy, Guillaume, Pavel, Sushil and Hanna, for contributing to an amazing team spirit in 207. I consider myself lucky for having the opportunity to collaborate with you and you have truly made daily obligations an enjoyment.

Thank you, Idiap support team: Nadine, Sylvie, system and engineering groups. I believe that we have been privileged to have you around to make all the administrative and system related issues transparent and to ensure that work at Idiap proceeds flawlessly and without distractions.

Thank you, cool people from the third floor and friends at the coffee machine, for making going to work a pleasant and entertaining experience. Just now I realize that, without noticing, you have really grown to my heart. Should I blame it on the after-lunch discussions, the Friday laughter at Café du Midi (definitely not the beer though), the sun shining on the skiing slopes

all around the valley, the adventures on the hiking paths or the exotic International flavors? I am trying to withhold myself of writing names afraid of missing someone (or shedding a tear), but I just can't: thank you Dayra, Marc, Alexandre, Kenneth, Alexandros, Pierre-Edouard, Branko, Pranay, Raphael, Joan, Gülcan, Tatjana, Phil, Ilja, Cijo, Rui, Wudi, James, Nikos, Petr, Ramya, Afsaneh, Leo, Serena, Rémi, Gwénolé, Paco, Paul and so many other dear folks.

Thank you, Michel, for making me so fond of the French language and the Valaisan lifestyle. French classes, and especially the outings we organized truly enriched the overall PhD experience. Actually, that's not true: for me, they were genuinely indispensable!

Thank you, Bile, Aleksandra, Verce and Tina, for being my remote support team. Thank you for always having ears for my student whims and staying great friends despite the distance.

Thank you, Marco, for enthusiastically standing behind many perspectives of mine and even more, for challenging others. In science, in work, in life. I wish I stay small forever, as growing up with you is so much fun. Merci pour avoir m'apprivoisée, and for hundreds of things beyond words.

Thank you, Alek, for being such a compassionate sister, one to teach and learn from, the only one to play "normal" games with. Thank you, mother and father, for showing me what true care, protection and comfort mean. With your joy and delight, you give 1000 times more value to my efforts and meaning to my achievements. In easy or tough times, it is truly soothing to be able to rely on a merry home with the smell of pumpkin pastry and ajvar with cheese.

Martigny, 18th December 2015 Ivana Chingovska

Abstract

The need for automation of the identity recognition process for a vast number of applications resulted in great advancement of biometric systems in the recent years. Yet, many studies indicate that these systems suffer from vulnerabilities to spoofing (presentation) attacks: a weakness that may compromise their usage in many cases. Face verification systems account for one of the most attractive spoofing targets, due to the easy access to face images of users, as well as the simplicity of the spoofing attack manufacturing process.

Many counter-measures to spoofing have been proposed in the literature. They are based on different cues that are used to distinguish between real accesses and spoofing attacks. The task of detecting spoofing attacks is most often considered as a binary classification problem, with real accesses being the positive class and spoofing attacks being the negative class.

The main objective of this thesis is to put the problem of anti-spoofing in a wider context, with an accent on its cooperation with a biometric verification system. In such a context, it is important to adopt an integrated perspective on biometric verification and anti-spoofing. In this thesis we identify and address three points where integration of the two systems is of interest.

The first integration point is situated at input-level. At this point, we are concerned with providing a unified information that both verification and anti-spoofing systems use. The unified information includes the samples used to enroll clients in the system, as well as the identity claims of the client at query time. We design two anti-spoofing schemes, one with a generative and one with a discriminative approach, which we refer to as client-specific, as opposed to the traditional client-independent ones.

At the second integration point, situated at output-level, we address the issue of combining the output of biometric verification and anti-spoofing systems in order to achieve an optimal combined decision about an input sample. We adopt a multiple expert fusion approach and we investigate several fusion methods, comparing the verification performance and robustness to spoofing of the fused systems.

The third integration point is associated with the evaluation process. The integrated perspective implies three types of inputs for the biometric system: real accesses, zero-effort impostors and spoofing attacks. We propose an evaluation methodology for biometric verification systems under spoofing attacks, called Expected Performance and Spoofability (EPS) framework, which accounts for all the three types of input and the error rates associated with them. Within this framework, we propose the EPS Curve (EPSC), which enables unbiased comparison of systems.

Abstract

The proposed methods are applied on several case studies for the face mode. Overall, the experimental results prove the integration to be beneficial for creating trustworthy face verification systems. At input-level, the results show the advantage of the client-specific approaches over the client-independent ones. At output-level, they present a comparison of the fusion methods. The case studies are furthermore used to demonstrate the EPS framework and its potential in evaluation of biometric verification systems under spoofing attacks.
The source code for the full set of methods is available as free software, as a satellite package to the free signal processing and machine learning toolbox Bob. It can be used to reproduce the results of the face mode case studies presented in this thesis, as well as to perform additional analysis and improve the proposed methods. Furthermore, it can be used to design case studies applying the proposed methods to other biometric modes.

Key words: Spoofing attacks, Counter-measures, Anti-spoofing, Liveness Detection, Presentation Attacks, Presentation Attack Detection, Biometric Verification, Face Verification, Biometric Evaluation

Résumé

Au cours des dernières années, le besoin d'automatisation du processus de reconnaissance d'identité pour un large nombre d'applications a engendré de grands progrès des systèmes biométriques. Cependant, plusieurs études ont montré une vulnérabilité de ces systèmes aux attaques d'usurpation (aussi appelées attaques de présentation ou "spoofing") qui peut compromettre leur utilisation dans de nombreux cas. Les systèmes de vérification du visage sont les plus susceptibles d'être exposés à ces attaques en raison d'un accès aisé aux images des visages des utilisateurs, ainsi qu'à la simplicité de mise en oeuvre du procédé d'usurpation d'identité.

De nombreuses contre-mesures aux attaques de présentation ont été proposées dans la littérature. Elles sont basées sur différents signaux utilisés pour distinguer les vrais accès des attaques spoofées. La tâche de détection des attaques est le plus souvent considérée comme un problème de classification binaire dans lequel les vrais accès constituent les exemples positifs, et les attaques les exemples négatifs.

L'objectif principal de cette thèse est de placer le problème de détection des attaques de présentation au coeur d'un contexte élargi en accentuant sa coopération avec un système de vérification biométrique. Dans un tel contexte, il est important d'adopter une perspective intégrée de la vérification biométrique et de l'détection des attaques de présentation. Dans cette thèse, nous identifions et traitons trois éléments pour lesquels l'intégration des deux systèmes est importante.

Le premier élément d'intégration se situe au niveau des signaux d'entrée. A ce stade, l'objectif est de fournir une information unifiée à la fois aux systèmes de vérification et à la détection des attaques de présentation. L'information unifiée comprend les échantillons utilisés pour enregistrer les clients dans le système, ainsi que l'identité proclamée. Nous concevons deux procédés de détection des attaques de présentation, l'un basé sur une approche générative, l'autre sur une approche discriminante, mais également spécifique au client, contrairement aux procédés traditionnels.

Au deuxième niveau d'intégration, situé en sortie, nous traitons le problème de combinaison entre les sorties des systèmes de vérification biométrique et de détection des attaques de présentation pour réaliser une décision combinée optimale. Nous adoptons une approche de fusion entre plusieurs experts et nous examinons plusieurs méthodes pour celle-ci en comparant la performance de la vérification et la robustesse des systèmes fusionnés aux attaques de présentation.

Le dernier élément d'intégration concerne l'évaluation. La perspective intégrée implique

trois types d'entrée pour le système biométrique: les accès réels, les imposteurs sans effort, et les attaques de présentation. Nous proposons une méthodologie d'évaluation des systèmes de vérification biométrique soumis à des attaques, que nous appelons 'Expected Performance and Spoofability' (EPS), et qui prend en compte les trois types d'entrée et leurs taux d'erreurs associés. Dans ce cadre, nous proposons la courbe 'EPS Curve' (EPSC) qui permet une comparaison non biaisée des systèmes.
Les méthodes proposées sont appliquées dans plusieurs cas d'étude concernant le reconnaissance de visage. Au niveau de l'entrée, les résultats expérimentaux montrent le bénéfice de l'intégration et l'avantage des approches spécifiques aux clients par rapport à celles qui leur sont indépendantes. Au niveau de la sortie, nous présentons une comparaison des méthodes de fusion. Les cas d'études sont par ailleurs utilisés pour démontrer l'usage de la méthode EPS et son potentiel pour évaluer les systèmes de vérification biométrique soumis à des attaques de présentation.
Le code source pour la totalité des méthodes est disponible en tant que logiciel libre comme satellite de la boîte à outils gratuite Bob pour le traitement du signal et l'apprentissage automatique. Il peut être utilisé pour reproduire les résultats des cas d'étude sur le visage présentés dans cette thèse, ainsi que pour procéder à des analyses complémentaires et améliorer les méthodes proposées. De plus, il peut être utilisé pour concevoir des cas d'étude en appliquant les méthodes proposées à d'autres modalités biométriques.

Mots-clés: Attaques de présentation, Contre-mesures, Détection d'attaques de présentation, Spoofing, Anti-spoofing, Détection du caractère vivant, Attaques de présentation, Détection d'attaques de présentation, Vérification biométrique, Vérification du visage, Evaluation biométrique

Contents

List of Figures

List of Tables

Glossary

The most important technical acronyms used in this thesis are listed below in alphabetical order.

AND	logical AND fusion rule
AUE	Area Under EPSC
DET	Detection-Error Trade-off
EER	Equal Error Rate
EM	Expectation-Maximization
EPC	Expected Performance Curve
EPS	Expected Performance and Spoofability framework
EPSC	Expected Performance and Spoofability Curve
FAR	False Acceptance Rate
FRR	False Rejection Rate
GJet	Face verification system based on Gabor Jets
GMM	Gaussian Mixture Model
HOG	Histogram of Oriented Gradients
HTER	Half Total Error Rate
ISV	Inter-Session Variability
LBP	Local Binary Patterns
LBP-TOP	LBP from Three Orthogonal Planes
LDA	Linear Discriminant Analysis
LGBPHS	Local Gabor Binary Pattern Histogram Sequences
LR	Logistic Regression
MAP	Maximum A-Posteriori
ML	Maximum Likelihood
PGM	Probabilistic Graphical Models
PLR	Polynomial Logistic Regression
RBF	Radial Basis Function
SFAR	Spoofing False Acceptance Rate
SUM	SUM fusion rule
SVM	Support Vector Machine
UBM	Universal Background Model
UBMGMM	Face verification system based on UBM using GMMs
WER	Weighted Error Rate

1 Introduction

We live in a digital world: an ever increasing amount of things that we possess or that represent us are stored as sequences of zeros and ones. Our finances, our personal and professional data, the services that we use are all partially or fully in an electronic format. As a result, it has been a long time now since passwords replaced traditional keys to authenticate users and ensure them a secure access not only to the electronic content that belongs to them, but also to physical objects they possess and places they are admitted to.

Passwords rely on a "what I know" paradigm, and are thus a knowledge-based authentication method [O'Gorman, 2003]. They consist of a sequence of characters that a user needs to remember. While widely used and usually considered secure, passwords are infamous for the lack of user convenience [Adams and Sasse, 1999]. This is especially the case when users need to remember multiple passwords, each for a different purpose. Studies like Adams and Sasse [1999]; Armstrong [2003]; Florencio and Herley [2007] show that users are prone to choose easily memorable, crackable passwords, and intentionally or unintentionally disclose them. Advising or forcing the users to use more difficult passwords may increase insecure work practices, like writing them down. Furthermore, passwords may be stolen via phishing or logging with a malicious spyware [Coskun and Herley, 2008].

Another form of authentication, relying on a "what I have" paradigm, are tokens in the form of portable storage devices, like smartcards. As an object-based authentication method, tokens are prone to theft. Thus, it is advised that they are used as a complementary authentication to passwords, rather than as a stand-alone security method [O'Gorman, 2003].

An alternative, ID-based form of authentication, is biometrics and it relies on a "what I am" paradigm. A biometric recognition system establishes the identity of a user by capturing some of his measurable physical and behavioral traits [Jain et al., 2006]. The biometric traits are selected so that they reliably distinguish one person from the other [Matyas Jr. and Stapleton, 2000] and are stable throughout the lifespan of an individual [O'Gorman, 2003]. This puts fingerprint, face, iris and voice among the most popular biometric modes used nowadays. Inconsistent or irreproducible presentation, imperfect signal acquisition and acquisition

condition variabilities [Jain et al., 2006] are biometric challenges which have been steadily addressed in the recent years.

To make a systematic comparison of the above-listed authentication methods, one needs to evaluate them by different criteria. Bonneau et al. [2012] defines simplicity of use and cost of deployment as some of these criteria. In this sense, biometrics has many advantages with respect to passwords and tokens, because it is memory-effortless, scalable for users, and can not be forgotten or lost. Yet, considering the value of the resources guarded by the system, the most important criteria are probably its security assets. Unfortunately, biometrics may be a subject of different security offenses, like targeted impersonation, theft, leaks and phishing.

This thesis is concerned with a particular security risk of biometric systems, *spoofing attacks*. Also referred to as *presentation attacks* [ISO-30107-1, 2014], they are performed when a malicious user claims another user's identity by forging a copy of their biometric trait and presenting it in front of the biometric system [Erdogmus and Marcel, 2014b]. Being recognized and acknowledged in many biometric modes, the risk is, to some extent, inhibiting a large-scale adoption of biometric systems, particularly in cases where human supervision of the authentication process is not possible.

Aiming at making a step forward towards more secure biometric authentication, in this thesis we address several issues regarding the integration of counter-measures to spoofing attacks into biometric systems. Before going into details, we give arguments stating the importance of developing spoofing counter-measures and the integration process. Subsequently, we formalize the objectives of this thesis and summarize its main contributions.

1.1 Background and Motivation

A generic biometric recognition system runs through several stages, like signal acquisition, feature extraction, template creation and matching. According to Ratha et al. [2001] and as illustrated at Fig. 1.1, the security of such a system can be compromised at several attack points. Galbally et al. [2007] group the attacks into two broad categories: indirect and direct. Indirect attacks are performed at a system level and require hacking skills to intrude into the system. They may manipulate, for example, the communication channel, the feature extraction or matching procedure, or tamper the stored templates, as represented by the vulnerability points 2, 3, 4, 5 and 6 on Fig. 1.1. On the other hand, direct attacks are performed before the signal is received by the system. Hence, direct attacks happen at vulnerability point 1 on Fig. 1.1 and they target the sensor itself.

The basic type of sensor-level attack is *zero-effort impostor*, and may arise as a result of a certain degree of similarity between biometric samples from two individuals [Jain et al., 2006]. The danger of such attacks is related to the level of individuality of the biometric trait. Robustness to zero-effort impostors is a fundamental property of a biometric recognition system. Indeed, the biometric community puts fundamental efforts to reduce the successful

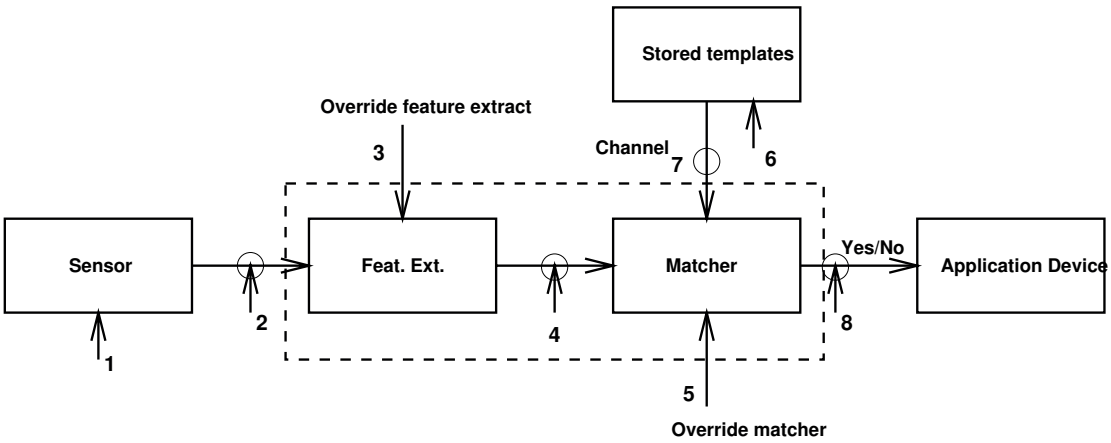


Figure 1.1: Points of vulnerability of generic biometric recognition system [Ratha et al., 2001]

zero-effort impostors by optimizing the recognition algorithms to minimize the similarities between the representations of different clients.

Unlike zero-effort impostors, spoofing attacks are adversary attacks at a sensor level. They involve presenting a copy of the biometric sample of another user in front of the system. Spoofing attacks are conceivable because biometrics, unlike passwords, does not provide security by secrecy, but by uniqueness [O'Gorman, 2003]. Not being a secret is an inherent property of biometrics [Jain et al., 2006], and thus acquiring a copy of a biometric trait of an individual is an attainable task. Unfortunately, information globalization acts in favor for adversary users, making access to biometric data as easy as never before. For example, photos and possibly videos featuring the face of users are available on various Internet websites. Users' voice can be easily recorded and examined at distance. Fingerprint molds can be easily manufactured from latent marks left on cups and door knobs.

Once it is obtained, a copied biometric trait not only allows an attacker to access a biometric system, but may avert the legitimate user to use the compromised biometric trait in the future due to security reasons. This is a consequence of what O'Gorman [2003] calls "the paradox of biometrics": the stability of the biometric trait, which is otherwise a desirable property, leaves no option for compromise recovery, since biometric traits can not be changed or replaced.

Biometric experts agree that it is impractical to prevent collection of biometric data from an individual [Matyas Jr. and Stapleton, 2000]. 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" [ANSI-X9.84-2010, 2010]. Rather, they recommend building preventive measurements to defend against fabricated replicas of biometric samples. O'Gorman [2003] 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. 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.

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. [2002] 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, D. and M. [2009] 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 [ChaosComputerClub, 2013; Swanner, 2014]. 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 [Flink, 2009]. Another one concerns a doctor who falsely registers her colleagues as present at work by spoofing the fingerprint scanner tracking the employee attendance [Matyszczyk, 2013].

Spoofing attacks are less likely to happen in a scenario where the biometric system is attended by a human supervisor. Yet, this depends on the biometric mode: as shown in an example above, it is possible to deceive the human control with a fingerprint spoofing attack. Furthermore, the rise of the use of biometric systems as an integral part of portable personal devices or for other unsupervised applications poses an urgent necessity to address the problem.

1.2 Scope of the Thesis and Terminology

Biometric recognition is used to refer to two different tasks: *verification* and *identification* [Mansfield et al., 2002]. In biometric verification, as a synonym to biometric authentication, a user claims a particular identity and the system needs to verify whether this claim is true based on the biometric trait. In biometric identification, the system needs to identify a biometric sample to belong to one out of many identities.

In biometric verification the user makes a positive claim of an identity and the system makes a one-to-one comparison of the input sample with the stored model of the claimed identity. In biometric identification the user usually makes no claim and the system needs to make a one-to-many comparison of the input sample with the stored models. Identification as a mode of operation of a biometric system is often used in applications where duplicate enrollment needs to be avoided, as well as in negative recognition, where an individual needs to be identified to belong in a certain group (watch list) of identities [Mansfield et al., 2002]. Because of the different purposes of verification and identification systems, adversary attacks at sensor level usually differ too. In biometric verification, the goal of an attack is presenting oneself as some other person. In biometric identification used for one of the purposes stated above, attack means disguising in order to hide one's true identity. In such a case, masquerading

as a particular person is not necessary, because disguise can be achieved by using generic face mask or other artificial materials, wearing glasses or make-up, occluding the face with a scarf, performing extreme facial expressions etc. For the iris mode, it can be achieved simply by using fashion colored contact lenses. For fingerprint mode, users that want to disguise themselves may intentionally make cuts or burns on their fingertips. Typically, the counter-measures to disguise and spoofing attacks as a special form of disguise are different. Furthermore, the different means of operation of verification and identification systems may require different approaches for integration of the counter-measures. The scope of this thesis is thus limited exclusively to spoofing attacks to biometric verification systems, and excludes attacks to biometric identification systems.

It is interesting to note that among the biometric modes, the face mode is presumably one of the most attractive to spoof. The reasons are two-fold. Firstly, face images of users are widely available on the Internet and obtaining them can be a matter of just a few clicks [Li et al., 2014]. Secondly, producing spoofing attacks for the face mode can be as easy as printing a photograph on a paper, or displaying a photograph on an electronic device. Unlike spoofing attacks for fingerprint or iris mode, no special skills or expensive materials are required to produce simple face spoofing attacks. This thesis is focused on the face mode and the value of the proposed solutions is demonstrated on case studies with face data. However, the aspects of spoofing detection treated in this thesis are generally valid for other biometric modes and the proposed solutions can be readily applied to other spoofing attacks.

The terms *spoofing detection*, *counter-measure* and *anti-spoofing method* are interchangeably used in this thesis to refer to systems or techniques for detecting spoofing attacks. The term *liveness detection* is often used in the literature, denoting that spoofing attacks do not demonstrate signs of liveness on the scene, while the other samples do. Indeed, many counter-measures explicitly measure the amount of liveness of the presented sample, like the perspiration of the skin [Schuckers, 2002] for the fingerprint mode, or eye blinking for the face mode [Pan et al., 2007]. However, not all spoofing attacks exhibit absence of liveness, esspecially for the face mode. For example, in an attack where a video of the user's face is played on the screen of a device, the liveness signs can be identical as in a real access. Furthermore, many face spoofing counter-measures operate using cues other than liveness. For these reasons, we avoid the use of the term *liveness detection* in this thesis.

Finally, in this thesis we will interchangeably use the terms *user* and *client* to refer to any person that is enrolled or uses the verification system. We will also interchangeably use *real access* and *genuine access* to denote samples coming from a valid user who is claiming the correct identity in front of the verification system.

1.3 Objectives and Contributions

Thanks to the growing interest of the biometric community towards the problem of spoofing, the number of published works in the domain notes a significant growth in the recent years.

The majority of articles focuses on developing spoofing detection methods and relevant features that can discriminate between real accesses and spoofing attacks. Thus, regardless of the biometric mode, most of the anti-spoofing systems are designed as binary classification systems with real accesses as the positive and spoofing attacks as the negative class.

There are several aspects of biometric spoofing which have received notably less attention. Among these, an essential issue arises due to the fact that anti-spoofing systems are not designated to work in isolation, but in cooperation with a biometric verification system that needs to be protected. This observation indicates the need to put anti-spoofing in a wider context encompassing a verification system. This includes the framework and circumstances within which the anti-spoofing system needs to be situated.

The main objective of this thesis is to highlight the need for an integrated perspective on biometric verification and anti-spoofing. To this end, we identify three points in the biometric verification and anti-spoofing pipeline where integration may play an important role. They are illustrated in Fig. 1.2. The first integration point is at *input-level* and is concerned with the information that both biometric verification and anti-spoofing systems use or may have access to. The second integration point is at *output-level* and focuses on how to take a unified decision about an input sample, based on the output of both biometric verification and anti-spoofing systems. The third point addresses *evaluation* of biometric verification systems when a threat of spoofing attacks is anticipated.

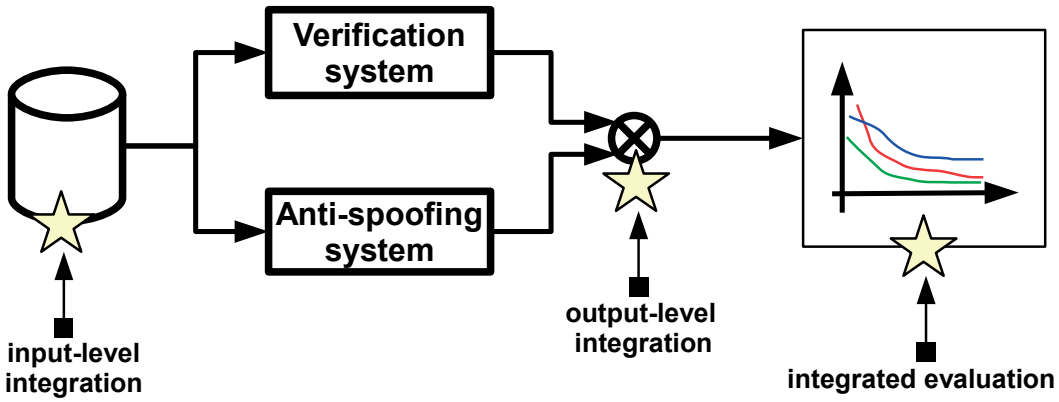


Figure 1.2: Important points of integration for biometric verification and anti-spoofing systems

With respect to the identified integration points, we argue that overlooking the cooperation of biometric verification and anti-spoofing systems has three major consequences:

1. The anti-spoofing system is deprived of particular information that biometric verification system uses imperatively, and which may be useful for spoofing detection;

2. The optimization of the fusion of the two systems towards a unified decision is neglected or completely prevented;

3. Correct evaluation and unbiased comparison of the performance and spoofing vulnerability of biometric verification systems is unattainable.

An integrated perspective at each of the three points stands behind a strong motivation which is thoroughly described throughout this thesis. Integration is, nevertheless, inspired by observations of the operation of biometric verification and anti-spoofing systems in practical applications.

At input-level, we start from the observation that a typical biometric verification system needs two inputs: a biometric sample and a claim about the identity of the client. Based on this claim, the system compares the input sample with the stored models for the client. The system also needs enrollment samples to create these models. On the other hand, a typical anti-spoofing system makes use of neither the client identity, nor the enrollment samples to take its decision. We argue that by integrating and unifying the information that both verification and anti-spoofing systems have access to, we can give an additional advantage to the anti-spoofing system.

At output-level, the issue of integration emerges once we want to practically deploy an anti-spoofing system to protect a biometric verification system. At this point, the decision whether a biometric sample is accepted or rejected depends on both of them. The way to fuse their outputs has a direct impact on both the verification performance of the system and its robustness to spoofing. If we consider that biometric verification and anti-spoofing systems are expert systems, we could optimize the performance of the fused system by exploring different multiple expert fusion approaches [Kittler et al., 1998].

At evaluation level, we first acknowledge that, in addition to genuine samples and zero-effort impostors, we have spoofing attacks as an input to the system. This implies a separate measurement of the errors associated with incorrectly accepted spoofing attacks, in addition to False Acceptance Rate (FAR) and False Rejection Rate (FRR) as error rates associated with incorrectly accepted zero-effort impostors and incorrectly rejected genuine users, respectively [Jain et al., 2008]. Considering that error rates in biometric classification systems are computed with respect to a decision threshold [Mansfield et al., 2002], we need to define an effective way to compute it. In this process, it is important to account on the three types of errors, as well as the three types of inputs. Based on these considerations, we also need to define performance curves that will enable unbiased comparison of biometric verification systems.

In this thesis we propose and compare practical methods to address the integration at all three levels. With respect to this, the major contributions of this thesis are the following:

1. **Input-level integration: design of client-specific anti-spoofing methods.** We examine the option to unify the information that anti-spoofing and verification systems have access to and whether anti-spoofing systems can benefit from it. This information

includes enrollment data which is used to enroll clients in the verification system, as well as information about the claimed client identity which is available at verification time. Abandoning the classical treatment of anti-spoofing systems as binary classification systems with models for real accesses as the positive and spoofing attacks as the negative class, we exploit this information to build separate spoofing models for each client. We refer to this approach as *client-specific*, as opposed to the traditional *client-independent* anti-spoofing approaches which disregard the client identity. We develop two client-specific methods: one based on a generative and one on a discriminative approach.

Publication related to this contribution: [Chingovska and Anjos, 2015].

2. **Output-level integration: comparison of multiple experts fusion methods for biometric verification and anti-spoofing systems.** We emphasize the need to explore several ways to fuse the output of biometric verification and anti-spoofing systems, in order to optimize the systems in terms of verification performance and robustness to spoofing attacks. We analyze several fusion schemes operating at decision-level and score-level.

   Publication related to this contribution: [Chingovska et al., 2013a].

3. **Integrated evaluation: design of evaluation methodology for biometric verification systems under spoofing attacks.** We present the drawbacks of existing methodologies for evaluation of verification systems under spoofing attacks and we demonstrate the need for improvement. We propose a new evaluation methodology, called *Expected Performance and Spoofability* (*EPS*) framework, which accounts for a variable weight of the systems error rates and varying expected probability of the inputs. The accompanying EPS Curve (EPSC) enables unbiased comparison between verification systems.

   Publication related to this contribution: [Chingovska et al., 2014a].

4. **Case studies on the face mode**. The integration concepts and methods elaborated in this thesis are illustrated by case studies on several different state-of-the-art face verification and anti-spoofing methods. The case studies are performed using face spoofing data and include an extensive empirical analysis which serves to compare the proposed methods.

   The results concerned with the input-level integration show that client-specific methods have a significant advantage over their client-independent counter-parts. The advantage is present both when tested with the same or different type of attacks than the ones used for training. With respect to the output-level integration, the results show the benefit of the fusion and a trade-off between systems' verification performance and robustness to spoofing. A large part of the analysis is performed using the EPS framework, demonstrating the usage of EPSC and the value of the methodology in evaluating biometric verification systems under spoofing attacks.

5. **Provision of fully reproducible experiments and reusable code.** The source code of all the methods described in this thesis is available as free software in the software package

`bob.thesis.ichingo2015`[1], which is a satellite package to the free signal processing and machine learning toolbox Bob[2] [Anjos et al., 2012]. The experiments for the case studies can be fully reproduced. Furthermore, the code can be used to create additional case studies based on other verification and anti-spoofing systems or different biometric modes.

Other publications related to this thesis: [Chingovska et al., 2012], [Chingovska et al., 2013b], [Chingovska et al., 2014c], [Chingovska et al., 2014b], [Anjos et al., 2014].

1.4 Thesis Outline

The thesis is composed of seven chapters. A brief summary of each of them is given below.

In **Chapter 2** we cover the existing literature in biometric spoofing and anti-spoofing, particularly focusing on the face mode. First, we give examples of face spoofing attacks and we cover the existing face spoofing databases. Then, we give an overview of the research efforts for face anti-spoofing, systematically categorizing them into several categories based on the cues they use to distinguish between real accesses and spoofing attacks. The overview is primarily focused on features for automatic anti-spoofing methods.

In **Chapter 3** we address input-level integration. We propose client-specific face anti-spoofing methods as a way to unify the information that both verification and anti-spoofing systems use. First, we empirically demonstrate the motivation for client-specific methods. Subsequently, we present the theoretical background of the proposed generative and discriminative client-specific methods.

In **Chapter 4** we focus on output-level integration. We state the need of taking a multiple expert approach and we study several fusion strategies at decision-level and score-level.

In **Chapter 5** we focus on integrated evaluation. We point out the weaknesses of the current evaluation methodologies for biometric verification systems under spoofing attacks and their inability to perform unbiased comparison between systems. Then, we present the Expected Performance and Spoofability (EPS) framework and the corresponding EPS Curve (EPSC) for an unbiased evaluation, which takes into account the three types of inputs and errors of biometric verification systems.

In **Chapter 6** we illustrate 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 verification and anti-spoofing systems and include extensive experiments to assess their performance. In the first part, we show the value of the client-specific approaches proposed for input-level integration in Chapter 3. In the second part, we make comparative analysis of the fusion

[1] https://pypi.python.org/pypi/bob.thesis.ichingo2015
[2] https://www.idiap.ch/software/bob

methods described in Chapter 4. Throughout the analysis we demonstrate the use of the EPS framework presented in Chapter 5 to evaluate biometric verification systems under spoofing attacks.

In **Chapter 7** we conclude the thesis with a summary of its contributions and achievements and we give an outline of possible directions for future work.

2 Literature Review

To understand biometric spoofing and come with practically viable anti-spoofing systems, one needs to go through several stages. Defining spoofing attacks and studying how they are created and performed is the first stage. Based on this knowledge, one can proceed with the second stage, which is developing suitable counter-measures. If then there is a need to set the problem into a context that considers the cooperation with biometric verification system, one needs to create integration mechanisms.

The majority of the work in anti-spoofing is focused on the first two stages. Topics of integration of verification and anti-spoofing systems, as well as of evaluation of biometric verification systems under spoofing attacks are relatively sparsely covered in the literature. Therefore, they are omitted in this literature review and will be covered in the corresponding chapters dedicated to these topics. Instead, we review spoofing and anti-spoofing for the face mode as isolated problems. This is important, as the case studies used to illustrate the integration concepts in this thesis are extensively making use of different face anti-spoofing methods and features. Understanding the advantages of the proposed methods as demonstrated in the experimental evaluation of the case studies, requires a comprehension of the anti-spoofing methods they are based on.

We commence this chapter by covering practical aspects related to fabrication of spoofing attacks for the face mode in Section 2.1. Then, in Section 2.2, we describe the efforts for organized collection of face spoofing data, which resulted in face spoofing databases used to evaluate anti-spoofing systems. We proceed with a systematic overview of face spoofing counter-measures in Section 2.3.

2.1 Face Spoofing Attacks

The quality of the spoofing attacks for the face mode is influenced by several factors. Firstly, there is the quality of the original sample used to produce the attack. For example, the original sample can be a mugshot image taken with user's cooperation, or an image in adversary

conditions taken from distance or downloaded from the Internet. The quality of the recorded input may also vary and may depend on the circumstances under which the spoofing attack is performed, like the illumination conditions or the presence of supervision at the biometric system capturing device. Other factors, categorized by Common Criteria as important for attacks to any kind of information systems are technical expertise, knowledge about the capturing device, window of opportunity etc. [ISO/IEC 15408].

The attacker usually has direct influence neither on the quality of the original sample, which may likely be obtained in an opportunistic manner, nor on the conditions at the side of the biometric system. However, he is fully responsible for the process of fabricating the attack, which includes the choice of the spoofing media, material, devices and tools needed to perform the attacks. These choices determine the *type* of the spoofing attack, as a broad description of its properties. The type of the attack is the basic source of differences between the spoofing attacks, which often serve as cues to detect them.

One of the properties of the spoofing attacks that is conditioned on their type is their dynamics. Based on this, they can be categorized as *static* or *dynamic*. The static spoofing attacks retain the face appearance, but present only a face with no signs of vitality. The dynamic spoofing attacks retain both the face appearance and vitality by exhibiting certain movements which are typical for a human face. Another property is their dimensionality: the face spoofing attacks can be in 2D or 3D.

Up to date, several prominent types of face spoofing attacks have been mentioned in the literature. They have appeared as part of a face spoofing database, or have been demonstrated to successfully spoof an existing face verification system. They are given in Table 2.1. Examples of several different types of attacks are given in Fig. 2.1.

The basic types of attacks can further differ in a number of other aspects, which may or may not depend on the attacker's will. An example is the environment where the original sample is recorded, and it can be *controlled* or *adversary*. A *fixed support* or a *hand support* can be used for holding the spoofing medium [Anjos and Marcel, 2011]. For the attacks performed with a hand support, the involuntary movements of the attacker's hands may give a level of liveness to the static attacks. Komulainen et al. [2013b] defines the term *scenic* spoofing attack referring to attacks where the background content of the original sample used for the spoofing attack is present alongside the face. On the contrary, on a *close-up* attacks the borders of the spoofing medium are integrally visible. This aspect is primarily influenced by the size of the original sample or the spoofing media used to display the attacks.

The complexity, cost and level of expertise to produce different types of spoofing attacks varies significantly. While producing a digital photo attack may require only an access to Internet and a consumer's mobile device, producing 3D masks may require expensive equipment, like a camera or 3D scanner and a 3D printer [Erdogmus and Marcel, 2013a].

As will be discussed in Section 2.3, the type of the attack, as well as the properties related to its

Attack dynamics	Dimensionality	Type	Description
Static	**2D**	print	face image printed on a paper
		digitalphoto	digital face image displayed on a screen of a device
		sketch	face sketched on a paper
	3D	warped print	print attack, warped to a 3D head shape
Dynamic	**2D**	perforated print	print attack with perforated eye regions
		moving print	print attack being moved to mimic face movements
		video	video recording of a face displayed on a screen of a device
	3D	mask	3D mask of a face with perforated eye regions
		make-up	make-up mask to resemble another user

Table 2.1: Types of face spoofing attacks

dynamics, dimensionality or other factors, have an important impact on the choice of features used by the spoofing counter-measures.

2.2 Face Spoofing Databases

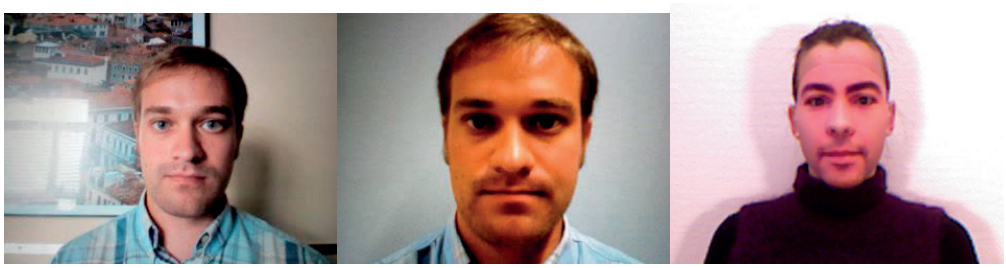
Depending on the attack types, the process of producing spoofing attacks may be time-consuming, sometimes requiring a lot of resources and certain manufacturing skills. Therefore, it is not difficult to imagine that collecting attack data for many clients may be very demanding, and, for certain type of attacks, too expensive [Erdogmus and Marcel, 2013b]. Perhaps this is one of the main reasons why the number of publicly available face spoofing databases is limited.

To the best of our knowledge, there are 5 publicly available face spoofing databases, differing in the data format, number of clients and samples, protocol, types of attacks, as well as the quality of the recording devices. In the following, we give a brief description of all of these databases.

2.2.1 NUAA Photo Impostor Database

The work of Tan et al. [2010] was the first one to present a face spoofing counter-measure evaluated on a publicly available database, called NUAA Photo Impostor Database. It provides 12,614 samples (5,105 real accesses and 7,509 attacks) to 15 clients, in a still image format with resolution of 640x480, recorded in 3 sessions and different illumination conditions. The

(a) Print attack (b) Warped print (c) Perforated print (d) Video attack



(e) Adverse conditions (f) Controlled conditions (g) 3D mask

Figure 2.1: Different types of face spoofing attacks. Fig. (a), (b), (c) and (d) show close-up attacks with visible spoofing media border. Fig. (e) and (f) show scenic attacks. Attacks examples taken from different face spoofing databases [Tan et al., 2010; Zhiwei et al., 2012; Chingovska et al., 2012; Erdogmus and Marcel, 2013b].

database contains only one print and warped print attacks, with face images printed on a normal paper, as well as a photographic paper with two different sizes. There are several attack samples per client, on which the printed paper is presented with different distance and position with respect to the recording device. The database is collected in an adverse environment, with visible borders of the spoofing medium. The protocol of this database divides the samples into train and test sets with no overlapping clients between the two sets.

This database has several constraints, like the limited number of clients and the lack of diversity between the attacks. As will be shown in Section 2.3, the provision of still images instead of videos prevents certain anti-spoofing methods to be evaluated on this database.

Examples of samples from this database are shown in Fig. 2.2.

Figure 2.2: Real access (first column) and spoofing attack samples from NUAA

2.2.2 CASIA Face Anti-spoofing Database

CASIA Face Anti-spoofing Database (CASIA-FASD) [Zhiwei et al., 2012] overcomes some of the main disadvantages of NUAA database. For example, the number of clients is as many as 50, and the samples are in a video format. Furthermore, it provides a larger diversity of spoofing attacks, of which there are as many as 3 types: warped print, perforated print and video attacks. The warped and perforated print attacks are printed on a copper paper, while the video attacks are played on a tablet. The overall diversity of the database is augmented by using 3 different recording devices for the samples: an old web-camera recording low-quality samples with resolution 640x480, a new web-camera recording normal-quality samples with resolution of 640x680 and a high-resolution camera recording high-quality samples with resolution of 1280x720. The database is recorded in adverse conditions, and the majority of attacks are close-up. The total number of samples in the database is 600 (150 real accesses and 450 attacks).

Similarly to NUAA database, the protocol of CASIA-FASD divides the samples into train and test sets with no overlapping clients between the two sets. Depending on the type of attacks and the sample quality, candidate anti-spoofing methods can be evaluated on different sub-protocols.

Fig. 2.3 shows example samples from CASIA-FASD database.

Figure 2.3: Real access (first column) and spoofing attack (warped print, perforated print and video in the last three columns, respectively) samples from CASIA-FASD.

2.2.3 Print-Attack, Photo-Attack and Replay-Attack Databases

Print-Attack [Anjos and Marcel, 2011], Photo-Attack [Anjos et al., 2013] and Replay-Attack [Chingovska et al., 2012] are a family of databases recorded for the same set of 50 clients, but with different types of spoofing attacks. While Print-Attack contains only print attacks, Photo-Attack contains both print and digital photo attacks and Replay-Attack is a super-set of the two of them, and provides additional video attacks. All samples are in a video format, recorded with

a built-in laptop camera and with a resolution of 320x240. An additional level of diversity of the attacks is provided by using two kinds of support for the attacks: hand and fixed. Diversity on a database-level is ensured by recording in two conditions: controlled and adverse. During the recording, special care has been taken that all attacks are scenic and no spoofing media borders are visible. The total number of samples in the database is 1200 (200 real accesses and 1000 attacks).

Besides train and test data, the protocol of Replay-Attack family of databases provides a development set for fine tuning of model parameters. These sets do not have overlapping clients. An exclusive property of Replay-Attack is the provision of 100 additional real access samples recorded in a separate session, which are designated for enrollment purposes. Using this data, one can train a biometric verification system using Replay-Attack and evaluate the effectiveness of its spoofing attacks to deceive such a system. Such an evaluation is not possible for the spoofing attacks in NUAA and CASIA-FASD databases. As will be seen in Chapters 3, 4 and 5, many aspects of anti-spoofing can not be addressed without such data.

Fig. 2.4 shows real and spoofing attack samples from Replay-Attack database.

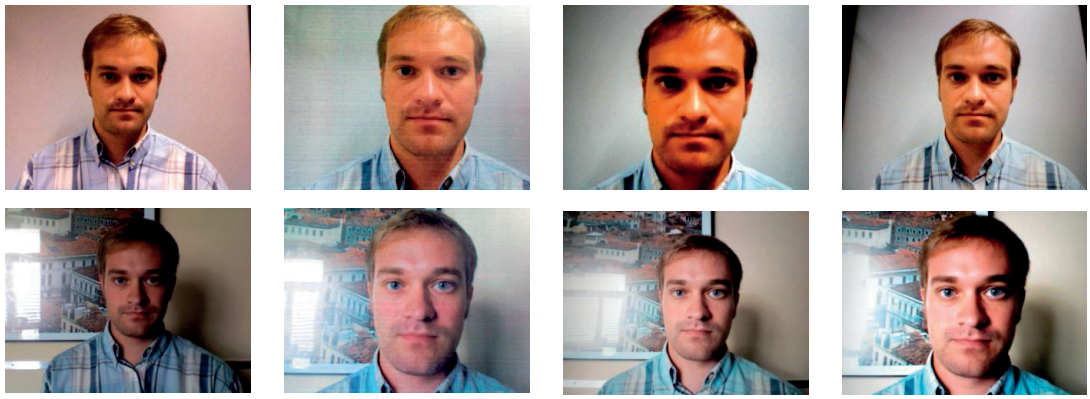


Figure 2.4: Real access (first column) and spoofing attack (print, digital photo, video in the last three columns, respectively) samples from Replay-Attack. Top row: controlled conditions. Bottom row: adverse conditions.

2.2.4 MSU Mobile Face Spoofing Database

The MSU Mobile Face Spoofing Database (MSU-MFSD) appeared recently to provide spoofing attacks targeting exclusively mobile devices [Wen et al., 2015]. It contains 35 clients and 280 video samples (70 real accesses and 210 attacks). The videos are recorded in adverse conditions using two types of cameras: a built-in laptop camera with resolution of 640x480 and a smartphone with resolution of 640x720. There are 3 types of attacks: print attack and video attacks with two different qualities, one replayed on a tablet screen, and one on a smartphone screen. They are all scenic attacks.

MSU-MFSD provides a protocol which contains train and test set only with non-overlapping identities and does not provide separate enrollment data.

2.2.5 3D Mask Attack Database

3D Mask Attack Database (3DMAD) [Erdogmus and Marcel, 2013b] is the first face spoofing database with 3D mask spoofing attacks. The 3D masks for a total of 17 clients are manufactured by a commercial service, using a frontal and a profile image of the face as a source. The database contains samples in two types of data formats: sequences of color data recorded using a camera and depth data recorded using a depth sensor, both with a resolution of 640x480. Using the depth data, unique spoofing counter-measures can be evaluated using this database. The total number of samples is 255 recorded in 3 sessions in a controlled environment.

3DMAD provides a protocol with a train, development and test set, but due to the small number of samples, the authors recommend using cross-validation for the evaluation of anti-spoofing methods. To train a biometric verification system, the database provides an alternative, modified protocol, where the samples from the first session are used for enrollment purposes.

Fig. 2.5 shows cropped and preprocessed faces from both color and depth samples from 3DMAD database.

Figure 2.5: Real access (columns 1 and 2) and mask spoofing attack (columns 3 and 4) samples from 3DMAD. Samples in column 1 and 3 are captured using a camera, samples in columns 2 and 4 are captured using depth sensor.

2.3 Face Anti-Spoofing Methods

Face anti-spoofing methods typically conform to a common flow of operation which is simplified in Fig. 2.6. The process goes through three main stages. The first stage is *preprocessing* and, if present, may consist of grey-scale conversion, face detection, face bounding box extraction and size normalization. The preprocessed sample is then a subject to *feature extraction*, which means mapping the input signal into a suitable feature space. The goal of this step is to isolate the information which is relevant to the task, which in this case is discriminating between real access and spoofing attack samples. In an ideal case, the feature vectors extracted from real

accesses and spoofing attacks will lie in different positions in the feature space. At the final, *classification* stage, the extracted feature vectors are sent to a classifier, which is trained to make the distinction between real accesses and spoofing attacks in the feature space.

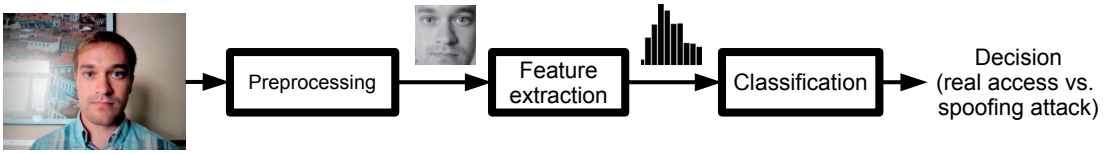


Figure 2.6: Typical flow of operation of a face anti-spoofing method

Different anti-spoofing methods rely on different feature extraction and classification procedures. In the following, we will cover the most prominent techniques in feature extraction and classification for face anti-spoofing.

2.3.1 Face Anti-Spoofing Features

The selection of feature extraction method often depends on the type of attacks that the anti-spoofing method targets. As different types of attacks have different properties, they relevant information that differentiates them from real accesses may be different and may need to be extracted in a different way.

A coarse categorization of anti-spoofing feature extraction methods is proposed by Erdogmus and Marcel [2014b], using the type of the input as a criteria. Three categories are listed as: employing additional hardware, capturing additional data or using the biometric data.

The first category consists of methods that employ additional hardware to capture data different than the one which is used by the biometric verification system. This data, and not the one used for verification, is then used to detect spoofing attacks. Usually, specialized sensors capture cues about the liveness of the subject in front of the system. Examples of additional hardware can be thermal sensors [Prokoski, 1983], near-infrared sensors [Pavlidis and Symosek, 2000; Kim et al., 2009; Zhang et al., 2011], multi-spectral filter [Wang et al., 2013b], spectrograph [Angelopoulou, 2001], or, more recently, light-field camera [Raghavendra et al., 2015]. Methods based on depth information, like the ones presented in [Kose and Dugelay, 2013; Erdogmus and Marcel, 2013b] also fall in this category. 3DMAD database provides samples suitable for developing such methods. This category of methods are often considered inconvenient because of the cost associated with the additional hardware. Furthermore, they may be less convenient from a deployment perspective, because it may be difficult to provide an additional hardware for certain applications, like, for example, mobile devices.

The second category involves methods which capture additional data using the same capturing device used for verification. An example is the method by Wang et al. [2013a], where several additional face images are taken from different viewing angles. The method proposed by Kim

et al. [2013] requires two face images taken with different focus. De Marsico et al. [2012] propose a method that requires images of the user performing various head movements. These methods may be less convenient from users' perspective, as an additional level of cooperation is required from their side in order to capture the additional information.

In the following sections we will focus exclusively on methods from the third category, which encompasses methods that automatically extract anti-spoofing features directly from the biometric data used for verification. Depending on the cues that are used to infer the presence of a live subject in front of the system, these methods can be categorized as based on:

- liveness detection;
- motion analysis;
- visual appearance;
- contextual information;
- feature learning.

Usually, the features extracted for anti-spoofing are hand-crafted based on prior knowledge about the task. The sole exception are the feature learning methods, which extract relevant features in a completely data-driven fashion. In this section we will cover the most notable methods from all these categories. In addition, we will report on methods which fuse several approaches together.

Before proceeding, it is important to notice that several researchers have made attempts to increase the robustness of biometric recognition systems to spoofing attacks by using multiple biometric modes [Ross et al., 2008]. The intuition behind these solutions is that an attacker may need more effort to spoof the system, because there are more modes to spoof. Within such multimodal framework, face has been combined with fingerprint and iris [Johnson et al., 2010; Akhtar et al., 2012; Rodrigues et al., 2009, 2010], or with voice [Chetty and Wagner, 2006b]. However, Johnson et al. [2010]; Akhtar et al. [2012]; Rodrigues et al. [2009, 2010] have proven that one needs to be careful with the choice of combination rules, which may not be helpful if poorly designed. Combination rules designed specifically for the purpose of increased robustness have been proposed in [Rodrigues et al., 2009, 2010].

Liveness Detection

Liveness detection anti-spoofing methods base their decision on the evidence of liveness present on the scene. Usually, eye-blinking, mouth movements and involuntary subtle head movements are considered as evidence of liveness. One of the first attempts to employ eye-blinking for anti-spoofing is performed by Pan et al. [2007] and uses Conditional Random Fields (CRF) to model the state of the eye as open or closed and the correlation between its

state and the observation. With a similar purpose, Wang et al. [2009] use active shape models to detect the eye contours and difference of images to detect the blinking activity. In [Kollreider et al., 2008], eye-blinking detection is combined with analysis of the 3D properties of the subject.

A key, but limiting assumption of the liveness detection methods is that the user will experience the actions that suggest liveness within a given short time frame. For example, Pan et al. [2007] assume that eye blinks happen every 2-4 seconds, which may not be true always and for all the subjects. An attempt to overcome this limitation is done by methods which rely on more frequent, subtle changes in the face region, including color changes due to blood flow. To be able to detect these changes, Bharadwaj et al. [2013] perform Eulerian motion magnification [Wu et al., 2012] as a pre-processing before applying a technique for analyzing the texture or the motion patterns.

The majority of liveness detection methods rely on the involuntary movements of the user. They are mainly targeting static spoofing attacks, while may be easily deceived by spoofing attacks where liveness evidence is present. An alternative for these cases may be the special sub-category of challenge-response methods explicitly asking the user to perform certain action to verify his liveness. Representatives of this type have been already mentioned as methods which require cooperation from the user to capture additional data [Wang et al., 2013a; Kim et al., 2013; De Marsico et al., 2012]. There are various types of challenges that a user can perform: taking a particular head pose [Frischholz and Werner, 2003] or following a moving point with a gaze [Ali et al., 2013] are some of them. Finding the static and dynamic relationship between face and voice information from a speaking face or modeling a speaker in 3D shape is an option for anti-spoofing in a multimodal audio-visual system [Chetty and Wagner, 2006a]. It is important to note that the last approach can successfully detect not only visual, but even audio-visual spoofing attacks, like video playbacks with recorded utterance or 3D synthetic talking heads.

Some of the liveness detection methods, and especially the challenge-response methods are considered to be intrusive, non-friendly and uncomfortable from the aspect of a user experience. In addition, they usually require that the authentication is performed during a prolonged time span. Finally, they are not transparent for the user. In this way, it is possible for a malicious user to guess the liveness cue and try to bypass it.

Motion Analysis

The methods based on motion analysis try to find properties of the motion patterns of a person in front of the system, in order to distinguish them from motion patterns in presence of a spoofing attack. A few of these methods base their approach on the assumption that a person's head, being a 3D object, moves differently than a 2D spoofing attack displayed on a planar media. For example, Kollreider et al. [2009] use optical flow method to track movements on different face parts. The authors assume that, in contrast to a face displayed on a 2D surface,

a 3D face will generate higher amount of motion in central face parts closer to the camera (like the nose) then in the face parts which are further away from the camera (like the ears). Furthermore, a 3D face exhibits motion flows which are in opposite directions for central and peripheral face parts. On the other hand, Bao et al. [2009b] derive a heuristics for the optical flow field for four basic 2D surface motion types: translation, in-plane rotation, panning and swing. On the contrary, a 3D face and facial expressions generate irregular optical flow field. Making no assumptions about the properties of the motion in the case of real accesses and spoofing attacks, Tirunagari et al. [2015] derive features capturing the visual dynamics of the samples using a technique emerging from fluid dynamics field, called Dynamic Mode Decomposition (DMD) [SCHMID, 2010].

Another set of motion-based methods assumes a high correlation between the movements in the face region and the background in the case of a spoofing attack. Such a correlation is unlikely in the case of a real access. Anjos and Marcel [2011] base the computation of the correlation on 10 quantities extracted from the face region and the background. For the same purpose, Anjos et al. [2013] rely on quantization of optical flow motion vectors, while Yan et al. [2012] perform foreground-background consistency analysis.

Similarly to the liveness detection methods, the motion analysis approaches depend on the subtle involuntary movements of the user. In addition, sometimes they capture the motion introduced by an attacker who holds the attack media with his hands. If the presumed motion patterns are absent during the short acquisition process (for example, a very still person who does not blink), the methods may fail. These methods are effective against static spoofing attacks, but may miss dynamic ones. Furthermore, the methods based on motion correlation are particularly targeting scenic spoofing attack.

Visual Appearance

Anti-spoofing methods analyzing the visual appearance stand behind a strong argumentation about the differences in the visual properties of real accesses and spoofing attacks, explained in a number of publications. Firstly, a real face and the human skin have their own optical qualities (absorption, reflection, scattering, refraction), which other materials that can be used as spoofing media (paper, photographic paper or electronic display) do not possess [Parziale et al., 2005]. Similar differences can appear as a result of the diffuse reflection due to a non-natural shape of the spoofing attacks [Yang et al., 2013]. Limited resolution of the device used for spoofing or the involuntary shaking of the spoofing media may cause a blurring in the case of spoofing attacks [Li et al., 2004; Määttä et al., 2012; Yang et al., 2013]. Artifacts appearing in the spoofing production process, like jitter and banding in the case of print attacks [Määttä et al., 2012; Yan et al., 2012] or flickering and Moiré effect in the case of video attacks [da Silva Pinto et al., 2012] are yet another sources of differences between the real accesses and spoofing attacks. Many of these visual properties are indistinguishable for the human eye, but often can be easily extracted using different image processing and computer vision algorithms.

The first approach leveraging on the argument that spoofing attacks are usually of lower resolution and thus contain less high-frequency components is proposed by Li et al. [2004]. The proposed feature vector is based on analysis of the 2D Fourier spectrum of the input image and its energy change over time. Instead of comparing the high-frequency content of the input, Tan et al. [2010] and Zhiwei et al. [2012] base their discrimination on the high-middle band of the Fourier spectrum, which is extracted using Difference of Gaussians (DoG) method.

Some publications assume that the differences between real accesses and attacks are most prominent within the reflectance component of the input image and estimate it in different ways: Tan et al. [2010] use the Lambertian reflectance model [Oren and Nayar, 1995] and Variational Retinex-based method, while Bai et al. [2010] use dichromatic reflection model. Then, Tan et al. [2010] classify the obtained features using Sparse Low-rank bilinear discriminative model, while Bai et al. [2010] compare the gradient histograms of the reflectance images.

A feature set inspired by a physics-based model for recaptured images, which reveals differences in the background contextual information, reflection, surface gradient, color, contrast, chromaticity and blurriness, is created by Gao et al. [2010]. Different sets of visual features related to texture, color, edges and/or gradient are used by Tronci et al. [2011]; Schwartz et al. [2011]. Galbally et al. [2014] generalize the appearance differences into quality differences and uses a feature vector composed of 25 different image quality measures.

Several publications make use of specific computer vision descriptors for texture analysis. Local Binary Pattern (LBP) [Ojala et al., 2002] appears to be the most significantly exploited for the purpose of anti-spoofing, both in its single resolution [Chingovska et al., 2012] and multiresolution variants [Määttä et al., 2011, 2012; Yang et al., 2013]. Histogram of Oriented Gradients (HOG) [Dalal and Triggs, 2005; Schwartz et al., 2011; Määttä et al., 2012; Yang et al., 2013], Grey-level Co-ocurrence Matrix (GLCM) [Schwartz et al., 2011], Haar wavelets [Yan et al., 2012] and Gabor wavelets [Määttä et al., 2012] are some of the other alternatives.

More recently, the analysis of the visual appearance has been enhanced into a temporal domain. In [da Silva Pinto et al., 2012], the authors firstly extract the noise from each video frame, and then summarize the relevant components of its 2D Fourier analysis into so-called Visual Rhythm image. The properties of this image are then captured using GLCM. The method proposed in [Pereira et al., 2014] utilizes LBP-TOP [Zhao and Pietikäinen, 2007], where instead of LBP analysis on a single frame, dynamical LBP analysis on a frame and its neighboring frames is performed.

The methods described before present different rates of success, which can not be easily compared because they are obtained on different types of attacks and usually on databases which are not released publicly. An interesting property of the majority of the visual appearance methods is that they can work even if only a single image is available at input. They are usually applied either on the face bounding box, face parts, or on the full input image. As one of their advantages, they are user-friendly and non-intrusive and do not depend on the behavior of

the user (unlike the liveness detection and motion analysis methods). Furthermore, an attack which can deceive them *a priori* has not been presented up to this moment. They may be expected to successfully detect any of the static or dynamic attacks. Yet, their success may be put into question if the spoofing attacks are printed or displayed on high-resolution media, thus lacking some of the artifacts that these methods rely on. Their generalization properties when applied to different acquisition conditions or types of attacks they are not trained for are also uncertain, since the visual appearance of the samples often depends on the lighting condition, acquisition devices or display media.

Contextual Information

The context of the scene present as a background information in front of the recognition system is used as a cue to detect spoofing attacks. In [Pan et al., 2011], the authors notice that in the case of a spoofing attack, there will be a change in the contextual information of the background when the face appears, which is especially true for scenic attacks. To detect such changes, the authors compare the regions around reference fiducial key points in the region around the face.

The approach presented in [Komulainen et al., 2013a] is targeting close-up attacks and the analyzed contextual information consists of the border of the spoofing medium. The method relies on HOG features to detect upper body and spoofing medium borders.

A drawback of these methods is that they depend on special assumptions about the background of the spoofing attacks and usually target attacks in a concrete context. As different spoofing attacks can present different context, these methods can not be expected to generalize well on spoofing attacks not considered during training.

Feature Learning

Following a recent trend in computer vision, the anti-spoofing community started experimenting with approaches where via deep learning, the anti-spoofing features are automatically learned directly from data. This is in contrast to the previously discussed approaches, where the features are inspired by some particular characteristics that can be observed as common either for real accesses or for some or all types of spoofing attacks. It is argued, however, that the features engineered in this way are suitable only for the type of spoofing attacks they are designed for Yang et al. [2014]; Menotti et al. [2015]. Therefore, Yang et al. [2014] and Menotti et al. [2015] propose to train a Convolutional Neural Network (CNN) [Krizhevsky et al., 2012] to automatically learn features discriminative for anti-spoofing. Experiments with face images in 5 different resolutions are given in [Yang et al., 2014]. On the other hand, Menotti et al. [2015] combine two approaches. With the first, the architecture of the network is optimized by selecting the best one for the problem at hand, out of a family of CNNs with different hyper-parameters. With the second one, the weights of the network are learned via

back-propagation.

2.3.2 Classification Methods

With regards to the classification step which, as shown in Fig. 2.6, follows the feature extraction, the systems usually comply to the binary classification definition. In a binary classification problem, there are two classes: a positive and a negative one. In the case of spoofing detection, real accesses play the role of the positive, while spoofing attacks play the role of the negative class. Let's denote the feature vector of a sample with $\mathbf{x}$. Given $\mathbf{x}$ as an input, the binary system needs to determine the value of its class $c = \mathrm{C}$ where $\mathrm{C} \in \{\mathrm{R}, \mathrm{A}\}$ and R stands for the class of real accesses, while A stands for the class of spoofing attacks.

Bishop [2006] gives three approaches to solve classification problems, including binary classification problems:

- Using a *discriminant function*. A discriminant function $f(\mathbf{x})$ performs a direct map of the feature vector points $\mathbf{x}$ from the input space to a score determining the class label.
- Using a *discriminative model*. In a discriminative model, the posterior probability $p(c = \mathrm{C}|\mathbf{x})$ is directly estimated.
- Using a *generative model*. In a generative model, first the class-conditional probability densities $p(\mathbf{x}|c = \mathrm{C})$, as well as the prior probabilities $p(c = \mathrm{C})$ for each class C are estimated. Then, the posterior probability $p(c = \mathrm{C}|\mathbf{x})$ is determined using the Bayes theorem.

The parameters of the discriminant function, as well as of the discriminative and generative models are determined in the training process using the training data. To obtain the final decision for binary classification problems, a decision threshold needs to be set. All the samples with scores on one side of the threshold are assigned to the same class. The threshold is applied on the score obtained by the discriminant function for the first approach, or the ratio of the log-likelihoods of the data given the two classes for the second and third approach.

The majority of anti-spoofing systems use methods based either on discriminant functions or discriminative models, usually ones which have a well established reputation in the machine learning community. The most popular classifier used for face anti-spoofing is Support Vector Machines (SVM) [Bai et al., 2010; Määttä et al., 2011; Tronci et al., 2011; da Silva Pinto et al., 2012; Chingovska et al., 2012; Komulainen et al., 2013a; Yang et al., 2013; Kose and Dugelay, 2013; Pereira et al., 2014], both with linear or Radial Basis Function (RBF) kernel. For features in the form of histograms, like LBP, homogeneous kernels mapping approximating a χ^2 kernel can be used as well [Määttä et al., 2012]. Other methods use Linear Discriminant Analysis (LDA) [Chingovska et al., 2012; Galbally et al., 2014], Sparse Logistic Regression [Tan et al., 2010], boosting [Pan et al., 2007], random forests [Wang et al., 2009], Partial Least Squares

Regression (PLS) [Schwartz et al., 2011; da Silva Pinto et al., 2012] and Multi-Layer Perceptron (MLP) [Anjos and Marcel, 2011]. Some systems produce scores on which a threshold can be directly applied [Li et al., 2004; Kollreider et al., 2008, 2009; Bao et al., 2009a; Anjos et al., 2013].

During training of any of the systems mentioned above, the full set of real access and spoofing attack samples available in the training set are taken to form the positive and the negative class, respectively. These systems can be considered as client-independent, as the information about the clients that the samples belong to is disregarded.

2.3.3 Fusion of Face Anti-spoofing Methods

As discussed in Section 2.3.1, usually face anti-spoofing features are inspired from some particular properties of either real accesses or one or more types of spoofing attacks. Therefore, they are customized to particular attack type and are often not effective if the attack property assumptions do not hold. Pereira et al. [2013] has made a proof of concept that the anti-spoofing systems may not be able to generalize well on unseen spoofing attacks. A way to mitigate this problem is proposed by the trend of fusing several different anti-spoofing methods to obtain a more general counter-measure effective against a multitude of attack types. Fusion of anti-spoofing method has been done at several levels: feature, score, decision, frame and video level.

The first attempt of fusing has been performed by Kollreider et al. [2008], who detect spoofing attacks based on the decision of two independent methods: one analyzing 3D properties of the head and one the eye-blinking of the user. In [Tronci et al., 2011], the authors develop a fusion scheme at a frame and video-level and apply it to a set of visual appearance cues. In [Schwartz et al., 2011], the fusion of visual appearance cues is done at feature level. Yan et al. [2012] for the first time bring the intuition that the fusion can have a bigger impact if done with complementary counter-measures, i.e. those that address different spoofing attack cues. In the particular case, although subject to some prerequisites of the videos, a method based on motion analysis is fused with a method based on visual appearance.

To measure the level of independence of two anti-spoofing systems, and thus to get an estimation of their complementarity and effectiveness of their fusion, Pereira et al. [2013] propose employing a statistical analysis based on [Kuncheva and Whitaker, 2003]. For the same purpose, Komulainen et al. [2013b] propose to count the common errors. They further show that score-level fusion of several simple anti-spoofing methods which do not involve complex inefficient classifiers may be favorable with respect to a single one which is memory and time requiring.

The trend of fusing multiple complementary anti-spoofing methods continued in the 2nd competition on counter-measures to 2D face spoofing attacks [Chingovska et al., 2013b]. While fusion at score level is the most dominant approach, future efforts should analyze what is the most effective fusion strategy, both in terms of error rates, as well as flexibility of incorporating

a newly developed counter-measure into the fused system.

2.4 Discussion

The increased interest for the problem of anti-spoofing has resulted in a large set of counter-measures, most of which follow a typical flow of operation, as depicted in Fig. 2.6. The major diversity among the counter-measures is provided by the types of features they use, which, on the other hand, are chosen based on some cues differentiating between real accesses and one or more types of spoofing attacks. The counter-measures further differ in other important properties, like their intrusiveness for the user and the type of input they require.

Unfortunately, it is not easy to compare the performance of the proposed spoofing attacks, mostly because they not always target the same types of attacks. Therefore, they are usually tested on different databases, some of which are private. In our opinion, summarizing their properties and grouping them by category is a task more important than performance comparison. Having such a summary, a user can decide which method to use for an application of interest based on the expected type of spoofing attacks, input type that the system provides, as well as ease of implementation and convenience of use.

However, experiments on different databases have shown that face anti-spoofing methods can quite effectively detect spoofing attacks. Best results on Replay-Attack database are as high as 0% HTER when a multitude of different features are fused together [Chingovska et al., 2013b]. Spoofing attacks in 3DMAD database reach nearly perfect accuracy even with simple features based on LBP [Erdogmus and Marcel, 2014a]. The HTER on CASIA-FASD has been reported to be 6.25% using CNN [Yang et al., 2014].

Some of the good results can be attributed to the insufficient complexity and diversity of the publicly available face spoofing databases. The community has recognized the limitations of the currently existing databases, ranging from small number of identities, to small set of spoofing attacks types, to various types of biases. More challenging databases need to be created in future. Considering different materials to produce the spoofing attacks, using better quality equipment, creating more diverse illumination conditions or recording more clients are some of the ways to add to the adversity of the spoofing databases.

While the methods covered in this chapter address the problem of anti-spoofing respecting its binary classification nature, there is very little attention given to the integration of anti-spoofing systems in a framework cooperating with biometric verification systems. In this thesis we will focus on three aspects of this cooperation, as stated in Chapter 1.3. A short summary of the treatment of these aspects in the literature will be given in the corresponding chapters.

3 Input-level Integration: Client-Specific Approaches to Anti-Spoofing

As binary classification systems, anti-spoofing systems rely on a model trained using samples from two classes: a positive class consisting of real access samples and a negative class consisting of spoofing attacks. At training time, a typical anti-spoofing systems uses solely the class label and does not have access to any other information about the samples, including the identity of the client.

At query time, the anti-spoofing system receives a single input, a biometric sample, which needs to be processed and ultimately assigned a label as being accepted (real access) or rejected (spoofing attack). The decision is based upon the feature vectors extracted from the sample. Again, no other information about the sample is used, including the information about the identity of the client the sample comes from. Disregarding this information, anti-spoofing systems of this kind could be referred to as *client-independent*.

Biometric verification systems often operate in another way. Typically, they use one or more enrollment samples for each client to build client models. These samples need to be labeled with the identity of the client. At verification time, the system receives two inputs: the first one is the biometric sample, and the second one is a claim about the client's identity. Based on the claim, the system matches the biometric sample with the client's model and assigns a label as being accepted (genuine user) or rejected (zero-effort impostor).

In the described setting, the biometric verification and anti-spoofing systems rely on different data to build their models. The anti-spoofing system does not use the enrollment samples for the clients. This is despite the fact that it could access them in the same way the verification system does. Similarly, and as shown in Fig. 3.1, the biometric verification and anti-spoofing systems have partially different input at query time. Although the two systems work in parallel and need to take a collaborative decision about whether a sample is accepted or rejected, they do not make use of the same information. The anti-spoofing system neglects the information about the client identity, although it could have an access to it in the same way the verification system does.

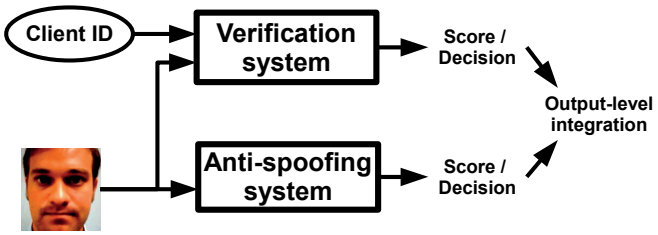


Figure 3.1: Flow diagram of the operation of biometric verification and anti-spoofing systems: no input integration

In this chapter, we focus on the integration of biometric verification and anti-spoofing systems at input-level. The integration refers to unification of the information that the two systems use. We argue that the additional information that the anti-spoofing system can use in the same way biometric verification system does already, may bring an improvement of the spoofing detection. At training time, this information refers to the enrollment samples of the clients. At query time, it refers to the identity information that the client claims to the system. In this context, the input-level integration of the two systems sums up to adding the green dashed line in Fig. 3.2, illustrating the client identity as an input to the anti-spoofing system.

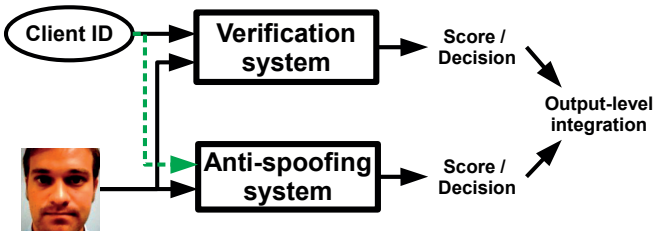


Figure 3.2: Flow diagram of the operation of biometric verification and anti-spoofing systems: input-level integration

Having enrollment client samples at disposal, the anti-spoofing system could build *client-specific* models for each client enrolled in the verification system. At query time, the anti-spoofing system will use the claim about the client identity in order to compare the input sample with the model of the corresponding client.

In the following, we first give a motivation and an empirical justification for using client-specific approaches for anti-spoofing in Section 3.1. Then, we propose two client-specific approaches: one based on a generative paradigm in Section 3.2, and one on a discriminative paradigm in Section 3.3. Throughout this chapter, we will use the following notation:

- $c = \mathrm{C}$ where $\mathrm{C} \in \{\mathrm{R}, \mathrm{A}\}$ is a variable referring to the sample class. R stands for the class of real accesses, while A stands for the class of spoofing attacks.
- $i = \mathrm{I}$ where $\mathrm{I} \in \mathscr{Y}$ and $\mathscr{Y}$ is a set of client identities, is a variable referring to the identity

of the client the sample belongs to.

- $\mathrm{X} = \{\mathbf{x}_k | k \in 1..K\}$ is a variable referring to a single sample and consists of K different feature vectors extracted from that sample. The feature vectors can be derived from different cues. $\mathbf{x}_k \in \mathbb{R}^{d_k} | k \in 1..K$ are d_k-dimensional and we assume that they are mutually independent.

- $\chi = \{(\mathbf{x}_k, y_k) | k = 1..S\}$ is a set of S samples given with their feature vectors $\mathbf{x}_k \in \mathbb{R}^d$ and class labels $y_k \in \{-1, 1\}$. All the feature vectors are derived from a single cue.

3.1 Motivation

The client-independent anti-spoofing approaches are based on the assumption that there is no critical difference between the real access or spoofing attack samples from different clients. This is a reasonable hypothesis as the anti-spoofing features, including the face anti-spoofing features described in Section 2.3.1, are specifically designed to distinguish between real accesses and spoofing attacks, regardless of the client identities. Yet, we argue that many of the anti-spoofing features, even inadvertently, retain information specific for the clients that the samples belong to. These information may be related to the intrinsic personal properties of the clients, like their appearance or behavior. For example, the involuntary movements or eye-blinking patterns, which are an intrinsic client trait, may be manifested into the features extracted by analysis of motion or visual appearance in the temporal domain. The physical properties of the face or the skin tone and surface are likely to have an impact on the features based on visual appearance, like LBP or Gabor wavelets. This may come at no surprise, if we consider that LBP and Gabor wavelets, in more complex variants, are common in face verification [Marcel et al., 2007], where they are used to capture client-specific properties and differentiate between different clients.

Certainly, the anti-spoofing features are not likely to be helpful in face verification. However, the above observations give rise to the following questions: 1. do the anti-spoofing features carry information about the client; and 2. is this information relevant to make a better discrimination between real access and spoofing samples belonging to that client. A positive answer to these questions would justify client-specific anti-spoofing systems. Development of these is possible only if information like enrollment samples for the clients and the client's identity claim is available to the anti-spoofing system. This information can be provided by input-level integration of the biometric verification and anti-spoofing systems.

If there is a meaningful correlation between the anti-spoofing features and the client identity, it may be reflected in the scores of the anti-spoofing system. This, on the other hand, may have a direct impact on the error rates of the anti-spoofing system. This intuition is inspired by an acclaimed study in speaker verification on the dependence of verification scores on the identity of the clients [Doddington et al., 1998]. The study confirms different levels of recognizability for different speakers, leading to inhomogeneities of the performance of the verification

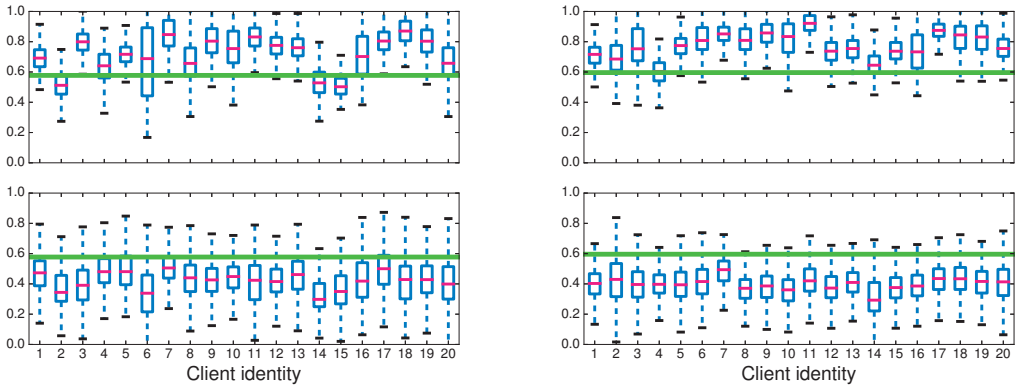
system across the client population. The result of this study is the popular Doddington's zoo, where the clients are categorized in four categories based on their predisposition to influence the error rates of the verification system, like False Acceptance Rate (FAR), False Rejection Rate (FRR) and others. In the Doddington's zoo, typical clients which are verified with an average score are referred to as *sheep*. The clients which are difficult to recognize and tend to increase the FRR are referred to as *goats*, while the clients which are easy to be imitated by others and tend to increase the FAR are referred to as *lambs*. Finally, *wolves* are clients which can easily imitate other clients and also increase the FAR of the system. The study uses statistical tests to examine whether the verification scores across the clients come from the same distribution.

In its original form, the Doddington's zoo assumes just two types of inputs to the verification system: genuine users and zero-effort impostors. Recently, Rattani et al. [2012] performed a similar study on a fingerprint verification system, demonstrating the existence of the Doddington's zoo menagerie when a verification system is confronted with spoofing attacks. In particular, the study shows that the system's vulnerability to spoofing among a population is client-specific as well.

We perform a similar empirical analysis, evaluating the scores of client-independent anti-spoofing systems. The goal is to observe whether the distribution of the scores obtained by anti-spoofing methods is client-specific. For the empirical results, we choose four of the state-of-the-art anti-spoofing features described in Section 2.3.1 and detailed in Section 6.1.1: LBP [Chingovska et al., 2012], LBP-TOP [Pereira et al., 2014], MOTION [Anjos and Marcel, 2011] and HOG [Yang et al., 2013]. They belong to the categories of anti-spoofing features based on analysis of visual appearance, visual appearance in the temporal domain and motion. Using these features, we train a Support Vector Machine (SVM) classifier in a client-independent way, as will be explained in Section 3.3.2. We perform the analysis using Replay-Attack database.

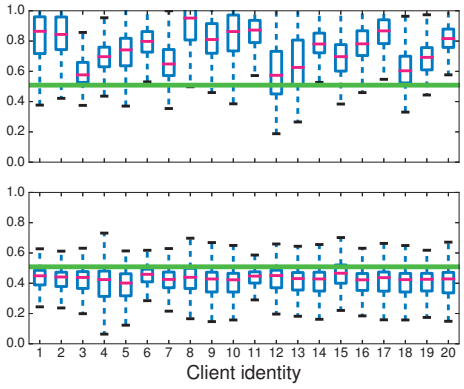
Similarly to [Rattani et al., 2012], we use box plots to show the variation of the scores for the real access and spoofing attack samples of each client in the test set of the database. Note that, unlike [Rattani et al., 2012], which evaluates the verification scores, we analyze the anti-spoofing scores per client. For the four studied anti-spoofing features, the box-plots are shown in Fig. 3.3, with the upper and lower plots representing real access and spoofing scores, respectively. The central bar of each box is the median of the client scores, its upper edge denotes the 75th percentile of the scores, the lower one the 25th percentile, while the whiskers extend to the most extreme non-outlier score values. The decision threshold determined in a client-independent way using Equal Error Rate (EER) on the development set is plotted as well with a green horizontal line.

The high variability of the scores of different clients in Fig. 3.3 demonstrates the existence of client-specific score variations for the client-independent baseline, especially in the case of real access samples. To statistically support this conclusion, we perform Kruskal-Wallis non-parametric statistical test [Kruskal and Wallis, 1952], designated to test whether samples originate from the same distribution. The null-hypothesis of the test states that there is no

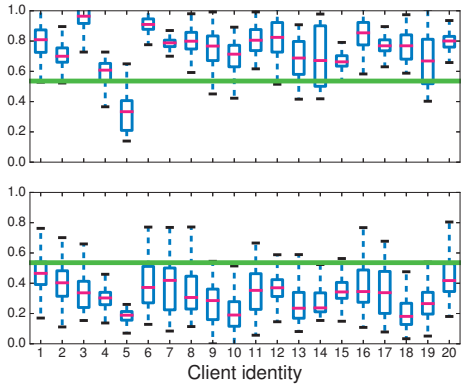


(a) LBP features

(b) LBP-TOP features



(c) MOTION features



(d) HOG features

Figure 3.3: Box plots of the scores obtained with a **client-independent** approach (SVM) for different clients in the test set of Replay-Attack database. Upper plots: scores of real access samples; lower plots: scores of spoofing attacks. The horizontal green line depicts the decision threshold obtained using the development set.

variation in the scores of the samples of the client population. In our analysis, this hypothesis was rejected at a 0.01 significance level for all four types of features and both for the scores of the real access and attack samples.

One of the problems that arise when the scores of a system are client-specific, is that the standard tuning and evaluation of the system is usually sub-optimal. In particular, a single decision threshold of the system will not work equally well for all the clients [Poh and Kittler, 2007], because high client-specific score variations mean that different clients contribute differently to the system's error rates. Client-specific thresholding, where a different decision threshold is used for each client [Jonsson et al., 1999] or client-specific score normalization [Auckenthaler et al., 2000] are some of the ways to alleviate the issue.

The score variations across the client population may mirror the overlap of real access and spoofing attack samples of different clients in the feature space. In such a case, the decision boundary obtained by the client-independent SVM is not equally suitable for all the clients, leading to low performance for certain clients and overall sub-optimal performance of the system. This indicates the presence of client-specific information in the feature space and suggests that a client-independent approach may not be enough to model the features.

We explore two different directions for creating client-specific models. The first one, details of which are given in Section 3.2, is based on a generative paradigm, where we model the real access samples of each client separately and we normalize the scores using generative models for the attacks. The second one, described in Section 3.3, is a discriminative approach, with separate SVM classifier trained for each client. We emphasize once again that input-level integration between biometric verification and anti-spoofing systems is required to create client-specific models and classify input samples.

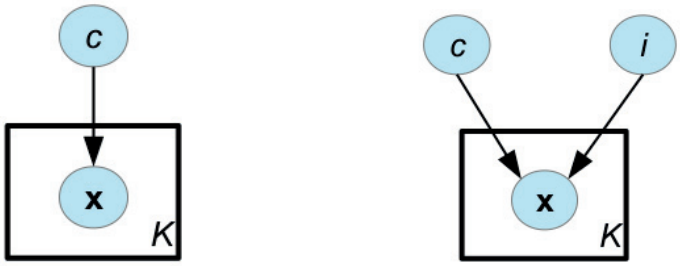
3.2 Generative Client-Specific Anti-Spoofing

In this section, we present Probabilistic Graphical Models (PGM) for development of generative classifiers for anti-spoofing and the Bayesian inference theory adapted for the task. We describe Gaussian Mixture Model (GMM) as a tool to model the likelihoods of the hypotheses of the generative classifiers, both in a client-independent and client-specific scenario. Then, we explain the use of cohort set for the client-specific case. Finally, we discuss several implementation issues regarding generative client-specific anti-spoofing systems.

3.2.1 Probabilistic Graphical Models for Anti-Spoofing

We introduce the generative client-specific anti-spoofing approach through Probabilistic Graphical Models (PGM) [Bishop, 2006], which enable visualization of the dependency relationships between variables, and thus facilitate the mathematical derivations of generative classification models. For a client-independent model, the feature vectors $\mathbf{x}_k$ representing a sample depend only on the sample class c. For a client-specific model, the feature vectors $\mathbf{x}_k$ depend both on the sample class c, as well as the identity i of the client the sample belongs to. PGM dependency schemes representing these two models are given in Fig. 3.4.

In a generative model for binary classification systems, a decision for the sample X is taken by comparing the likelihoods of two hypotheses: $\mathscr{H}_0$ and $\mathscr{H}_1$. Each of these hypotheses is associated with one of the two classes. Then, a score is generated as a ratio between the likelihoods of the two hypotheses. For computational reasons, usually the log of the likelihoods



(a) Client-independent model (b) Client-specific model

Figure 3.4: PGM illustrating the conditional dependence of variables

is computed, as in Eq. 3.1.

$$c_{\text{LLR}} = \log \frac{p(\text{X}|\mathscr{H}_0)}{p(\text{X}|\mathscr{H}_1)} \tag{3.1}$$

The decision about which of the two hypotheses is valid depends on both the log-likelihood ratio score and a threshold τ, as shown in Eq. 3.2. The sample is assigned to the class which is associated with the valid hypothesis.

$$\text{valid hypothesis} = \begin{cases} \mathscr{H}_0, & \text{if } c_{\text{LLR}} > \tau \\ \mathscr{H}_1, & \text{otherwise} \end{cases} \tag{3.2}$$

The difference between the client-independent and the client-specific approach is in the definition of the hypotheses. In a client-independent scenario, $\mathscr{H}_0$ states that X is generated from a real access sample ($\mathscr{H}_0 : c = \text{R}$), while $\mathscr{H}_1$ states that X is generated from an attack ($\mathscr{H}_1 : c = \text{A}$). Having K feature vectors extracted for the sample $\text{X} = \{\mathbf{x}_k | k \in 1..K\}$ and assuming they are mutually independent, the likelihoods of the two hypotheses are given in Eq. 3.3 and Eq. 3.4.

$$p(\text{X}|\mathscr{H}_0) = \prod_{k=1}^{K} p(\mathbf{x}_k|\mathscr{H}_0) = \prod_{k=1}^{K} p(\mathbf{x}_k|c = \text{R}) \tag{3.3}$$

$$p(\text{X}|\mathscr{H}_1) = \prod_{k=1}^{K} p(\mathbf{x}_k|\mathscr{H}_1) = \prod_{k=1}^{K} p(\mathbf{x}_k|c = \text{A}) \tag{3.4}$$

Estimating the likelihood functions in Eq. 3.3 and 3.4 can be done by building generative models for each of the hypotheses. For this, it is required that we have a training data consisting of real access and spoofing attacks samples. It is not necessary that this data is labeled with client identity.

In a client-specific scenario, $\mathscr{H}_0$ states that X is generated from a real access sample from a particular client I ($\mathscr{H}_0 : c = \mathrm{R}, i = \mathrm{I}$). Hence, its likelihood is computed as in Eq. 3.5.

$$p(\mathrm{X}|\mathscr{H}_0) = \prod_{k=1}^{K} p(\mathbf{x}_k|\mathscr{H}_0) = \prod_{k=1}^{K} p(\mathbf{x}_k|c = \mathrm{R}, i = \mathrm{I}) \tag{3.5}$$

Estimating the likelihood function in Eq. 3.5 can be done by building models representing the real access samples for each of the clients separately. This process in turn requires real access samples labeled with client identity. To avoid biased models, it is necessary that this data is separate from the data which is used to evaluate the system. A source for such data could be the enrollment samples which is used to enroll clients into the biometric verification system. This is the point where input-level integration of biometric verification and anti-spoofing system becomes of vital importance.

The alternative hypothesis $\mathscr{H}_1$ states that X is generated from a spoofing attack sample from a particular client I. Estimating the likelihood function of this hypothesis can be done by building models representing the spoofing attack samples for each of the clients separately. Similarly as for $\mathscr{H}_0$, this step requires spoofing attack samples from each client $I \in \mathscr{Y}$. Unfortunately, input-level integration with biometric verification system is of no use in this case, as the enrollment data for biometric verification system usually do not contain spoofing attacks. An option could be collecting spoofing attacks for each client in addition to his enrollment samples. However, the process of producing spoofing attacks may be expensive and time consuming, often requiring a lot of resources and certain manufacturing skills. Therefore, it is not difficult to imagine that collecting attack data for all the clients in a system may be very demanding and complex task. Baseline costs would quickly multiply if the system targets protection against a large number of diverse spoofing attacks.

To overcome this difficulty, we propose to model the alternative hypothesis $\mathscr{H}_1$ as a function of the likelihoods of spoofing attack models for a finite set of *cohort* clients $\mathscr{C}$. Mathematically, this way of modeling for the cohort clients $\mathrm{J} \in \mathscr{C}$ is shown in Eq. 3.6, and the function of cohorts $\mathscr{F}(\cdot)$ could be a maximum or an average [Reynolds et al., 2000].

$$p(\mathrm{X}|\mathscr{H}_1) = \mathscr{F}\left(p(\mathrm{X}|c = \mathrm{A}, i = \mathrm{J}) \mid \mathrm{J} \in \mathscr{C}\right) \tag{3.6}$$

The idea is inspired by the extensive use of cohorts in biometric verification in different setups.

They are used to model the alternative hypothesis $\mathscr{H}_1$, which in biometric verification states that the sample does not come from the client with the claimed identity. In [Reynolds, 1995] the cohorts are sorted by similarity with the particular client's model, and only the first N are taken to represent $\mathscr{H}_1$. In [Auckenthaler et al., 2000] they are used to perform Z-normalization and T-normalization of the scores. In [Simon-Zorita et al., 2003] $\mathscr{H}_1$ is modeled with the model of the cohort client with the highest likelihood. Similarly, Aggarwal et al. [2006] consider only the cohort client with the highest likelihood among the cohorts selected after sorting them by similarity to the particular client's model.

3.2.2 Likelihood Based on Gaussian Mixture Model (GMM)

When it comes to selecting the probability density function to model the likelihoods of the hypotheses, Gaussian Mixture Model (GMM) is the dominant approach in biometric verification, especially in speaker [Reynolds and Rose, 1995; Reynolds et al., 2000] and face [Sanderson and Paliwal, 2003; Lucey and Chen, 2004; Cardinaux et al., 2006; Marcel et al., 2007] verification. By using a weighted linear combination of several Gaussian distributions, GMM have proven to be able to model any complex arbitrary continuous density distribution [Bishop, 2006]. At the same time, it offers a good trade-off between computational requirements, robustness and discrimination capabilities [Reynolds et al., 2000].

A GMM is a weighted sum of M multivariate Gaussian distributions, called components, each parameterized with their means μ_m, variances Σ_m and weights π_m. Given a set χ of data samples $\mathbf{x}_k \in \mathbb{R}^d$, training of GMM model corresponds to finding the parameters $\Theta = \{\pi_m, \mu_m, \Sigma_m\}, m = 1..M$ which maximize the likelihood for the observed data. This Maximum Likelihood (ML) problem can be solved using Expectation-Maximization (EM) algorithm [Dempster et al., 1977]. Mathematical formulation of GMM and its training procedure is given in Appendix A.

In biometric verification, GMM is used to model client distributions for each of the enrolled clients which correspond to the hypothesis $\mathscr{H}_0$ that a sample comes from the claimed identity [Reynolds and Rose, 1995]. It is also used to create Universal Background Model (UBM), which corresponds to the alternative hypothesis $\mathscr{H}_1$ that the sample comes from another identity. The greatest challenge in this approach lies in how to reliably estimate the parameters of the GMM model for a client using a sparse set of enrollment samples of that client. The issue is usually overcome by using the enrollment samples only to adapt the client models starting from an UBM model as a prior. This can be done using Maximum A-Posteriori adaptation (MAP) [Gauvain and Lee, 1994], the details of which can be found in Appendix A.2.

In this thesis, we use GMM to model the likelihoods of the hypotheses for the generative anti-spoofing approaches. In the case of the client-independent models, GMM can be used to model the likelihoods $p(\mathbf{x}_k|c = \mathrm{R})$ and $p(\mathbf{x}_k|c = \mathrm{A})$. In the case of the client-specific models, we base the creation of client-specific anti-spoofing models on the idea coming from biometric verification [Reynolds and Rose, 1995]. We use GMM to $p(\mathbf{x}_k|c = \mathrm{R}, i = \mathrm{I})$ and $p(\mathbf{x}_k|c = \mathrm{A}, i = \mathrm{J})$

for each client I and each cohort J $\in \mathscr{C}$. We obtain the client-specific models by first training a UBM for real accesses and spoofing attacks and then adapting them with MAP adaptation to the client I or the cohort client J, respectively. Several researchers note that MAP where only the means of the UBM are adapted, and not the variances, is effective enough for biometric verification [Reynolds et al., 2000; Lucey and Chen, 2004]. We adopt the same approach of mean-only MAP adaptation for client-specific anti-spoofing models as well. Details about which data was used for each step of the procedure can be found in Sec. 3.2.4.

3.2.3 Cohort Selection

The selection and size of the cohort set $\mathscr{C}$ has been thoroughly studied in biometric verification [Rosenberg and Parthasarathy, 1996; Auckenthaler et al., 2000; Tulyakov et al., 2008]. In some applications, the full set of clients that can be used as cohorts is used [Reynolds, 1995]. However, Reynolds et al. [2000] suggests that client-specific cohort sets usually perform better and have the advantage of lowering the computational costs.

For modeling $\mathscr{H}_1$ in the anti-spoofing scenario, we explored the following alternatives for the cohort selection:

1. **Full cohort set**. The cohort set $\mathscr{C}$ consists of all the clients in the training set of the database.
2. **Static client-specific cohort set**. For each enrolled client I, the cohort models are sorted prior to query time based on a similarity criteria between the cohorts and the client model. Then, a client-specific cohort set $\mathscr{C}_\mathrm{I}$ is created consisting of the first N clients in the sorted cohort list.
3. **Dynamic client-specific cohort set**. For each data sample X of client I, the cohort models are sorted at query time based on a likelihood criteria of the sample given the cohort model. Then, a client-specific cohort set $\mathscr{C}_\mathrm{I}$ is created consisting of the first N clients in the sorted cohort list.

When building a static client-specific cohort set, we sort the cohorts prior to query time and based on the similarity of the cohort's attack model and the client's real access model. Following the proposition of Reynolds [1995], the distance $d_{\mathrm{I,J}}$ between the real access model of client I and the attack model of the cohort J is computed as in Eq. 3.7, where $\mathrm{X_I}$ is a set of real access observations for the client I, while $\mathrm{X_J}$ is a set of attack observations for the cohort J.

$$d_{\mathrm{I,J}} = \log \frac{p(\mathrm{X_I}|c=\mathrm{R}, i=\mathrm{I})}{p(\mathrm{X_I}|c=\mathrm{A}, i=\mathrm{J})} + \log \frac{p(\mathrm{X_J}|c=\mathrm{A}, i=\mathrm{J})}{p(\mathrm{X_J}|c=\mathrm{R}, i=\mathrm{I})} \tag{3.7}$$

The first ratio in Eq. 3.7 measures how well a sample matches with the model of its own client

relative to the model of the cohort. Similar models will result in smaller ratio. Equivalent interpretation can be given for the second ratio in Eq. 3.7. The final distance is a symmetric combination of ratios comparing the two models.

The sorting of the cohorts for a dynamic client-specific cohort set is done at query time, by computing the likelihood of the cohorts' models for the observed data. The higher the likelihood, the higher the rank of the particular cohort model. Such an approach is inspired by the work of Simon-Zorita et al. [2003] and Aggarwal et al. [2006].

Once the cohort set is selected, the likelihood of $\mathscr{H}_1$ is a function of the likelihoods of the cohorts. In particular, we chose an average function, as in Eq. 3.8. To adapt Eq. 3.8 for static or dynamic client-specific cohort set, we need to substitute $\mathscr{C} = \mathscr{C}_\mathrm{I}$.

$$\begin{aligned} p(\mathrm{X}|\mathscr{H}_1) &= \mathscr{F}\left(p(\mathrm{X}|c=\mathrm{A}, i=\mathrm{J}) \mid \mathrm{J} \in \mathscr{C}\right) \\ &= \mathscr{F}\left(\prod_{k=1}^{K} p(\mathbf{x}_k|c=\mathrm{A}, i=\mathrm{J}) \mid \mathrm{J} \in \mathscr{C}\right) \\ &= \frac{1}{|\mathscr{C}|}\sum_{\mathrm{J}\in\mathscr{C}}\prod_{k=1}^{K} p(\mathbf{x}_k|c=\mathrm{A}, i=\mathrm{J}) \end{aligned} \tag{3.8}$$

Having the likelihoods $p(\mathrm{X}|\mathscr{H}_0)$ and $p(\mathrm{X}|\mathscr{H}_1)$ computed as in Eq. 3.5 and 3.8 respectively, Eq. 3.1 can be used to compute the log-likelihood ratio score to compare the two hypotheses for a data sample X.

3.2.4 Implementation Details

For the client-specific anti-spoofing approach, we first create a UBM based on the real accesses of background clients taken from the training set. Using the enrollment samples of each client, we adapt the UBM to client-specific anti-spoofing models. These models are used to compute the likelihood $p(\mathrm{X}|\mathscr{H}_0)$ in Eq. 3.5.

Similarly, we create a UBM based on the attacks of all the clients from the training set. Using the attack samples of each of the cohort clients, we adapt this UBM to client-specific models for the cohort clients. These models are used to compute the likelihood $p(\mathrm{X}|\mathscr{H}_1)$ in Eq. 3.6.

3.3 Discriminative Client-Specific Anti-Spoofing

In this section, we first give a brief overview of Support Vector Machine (SVM) as a popular binary classifier. Then we examine its use for anti-spoofing and we explain how it can be used in a client-specific setting. Although formally, SVM belongs to the category of classification methods that rely on discriminant function, we conform to its more widely accepted

nomenclature and refer to it as a discriminative approach.

3.3.1 Support Vector Machine (SVM)

As described in Section 2.3.2, discriminative models and models based on discriminant function have already a well established reputation in face anti-spoofing. One of the most popular among them is Support Vector Machine (SVM) [Vapnik, 1998; Boser et al., 1992; Cristianini and Shawe-Taylor, 2000]. SVM is a classifier able to discriminatively learn a hyperplane that separates the set $\chi = \{(\mathbf{x}_k, y_k) | k = 1..S\}$ of training samples in $\mathbb{R}^{d \times \{-1,1\}}$, while minimizing its generalization error on unseen samples. The minimization of the generalization error is performed via the maximization of a quantity called *margin* which is related to the minimal distance of the samples to the hyperplane and which needs to be maximized [Fornoni, 2014]. In this context, SVM is also referred to as maximal margin classifier, while the samples from the two classes which lie on the margin are called *support vectors*.

The training of a SVM is realized by estimating the parameters of a real valued linear function $f : \mathbb{R}^d \rightarrow \mathbb{R}$. At test time, for each input sample $\mathbf{x}_k$ a score is computed as $f(\mathbf{x}_k)$. This function can be also represented in a *dual form*, where it is parameterized by the support vectors and a set of Lagrangian multipliers associated with them. In the dual form, the score of an input sample $\mathbf{x}_k$ is computed via its inner products with the support vectors. The mathematical details of these concepts are covered in Appendix B.

To achieve a high separability of the two classes, sometimes the samples need to be projected into a space with higher dimensionality, which can be a computationally expensive operation. The dual form of SVM allows to bypass this complexity through the use of *kernel functions*. Kernel functions enable a direct computation of the inner product between samples in a high-dimensional feature space without explicitly performing the projection. Different kernel functions exist. Among them, the most notable ones are linear, Radial Basis Function (RBF), Polynomial, Histogram Intersection and χ^2 and they define the shape of the boundary between the two classes in the original feature space. Details about the kernels and their parameters can be found in Appendix B.4.

3.3.2 SVM for anti-spoofing

When using SVM for the problem of anti-spoofing, the sample labels are defined as $y_k \in \{\mathrm{R}, \mathrm{A}\}$. SVM has been extensively used for anti-spoofing in a client-independent context, as described in Section 2.3.2. In this case, a single function $f(\mathbf{x})$ is estimated using all training samples $\{(\mathbf{x}_k, y_k) | y_k = \mathrm{R}\}$ as the positive class and $\{(\mathbf{x}_k, y_k) | y_k = \mathrm{A}\}$ as the negative class. The SVM training is performed regardless of the client identity.

A SVM in a client-specific context first appeared in speaker verification [McLaren, 2009]. Several SVM classifiers are trained, one for each of the enrolled clients. A single SVM classifier in this setup discriminates between samples coming from a claimed identity and samples

coming from a zero-effort impostor. When a new query arrives, it is classified by the SVM of the client it is claimed to belong to. Client-specific SVMs in speaker verification are trained using samples for the particular client as a positive class, and samples from a set of other clients as a negative class.

Inspired by this design, we build client-specific SVM for the anti-spoofing task in a similar manner. For each enrolled client $\mathrm{I} \in \mathscr{Y}$, we train a separate SVM classifier defined by $f_{\mathrm{I}} : \mathbb{R}^d \rightarrow \mathbb{R}$, whose role is to discriminate between real accesses and spoofing attacks for that client.

Ideally, each SVM should be trained using samples of the corresponding client: $(\mathbf{x}_{I,k}, \mathrm{R}) \in \chi$ as the positive and $(\mathbf{x}_{I,k}, \mathrm{A}) \in \chi$ as the negative class. As in the case of the generative client-specific approaches, we can use the enrollment samples of each client as the positive class. However, as explained in Section 3.2.1, obtaining spoofing attacks for each client may be a costly task.

As for the generative approach, we select a set of cohort clients $\mathscr{C}$ to approximate the spoofing attacks to represent the negative class for the client-specific SVMs. Therefore, each client-specific SVM is trained using $(\mathbf{x}_{\mathrm{J},k}, \mathrm{A}) | \mathrm{J} \in \mathscr{C}$. Since the samples from the cohort usually outnumber the client samples, the selection of the clients in $\mathscr{C}$ is of great importance and different heuristics to fulfill this task exist in the literature. One possibility is to consider several different cohort sets and to choose the one which gives the best performance on the development set [Kajarekar and Stolcke, 2007]. Instead of cohort clients, McLaren et al. [2010] selects cohort samples out of the samples which are most frequently used as support vectors for the client-specific SVMs on the development set. Using a large cohorts set may be restricted by computation limitations, but may provide better discriminative information [McLaren, 2009]. Therefore, we select all clients in the training set as cohort clients and all their spoofing attacks as negative samples to train the client-specific SVMs.

3.4 Discussion

The observation that biometric verification and anti-spoofing systems need to work in cooperation inspires the idea for their integration at input-level. It refers to unification of the data that is available to any of the two systems. As a result of the input-level integration, the anti-spoofing system can make use of the samples that the biometric verification system uses to enroll clients in order to create client-specific anti-spoofing models. Similarly, it can use the client identity claim to compare a query sample with the corresponding client-specific anti-spoofing model.

We looked at two approaches to build client-specific anti-spoofing models. The first one relies on a generative paradigm and compares GMM-based models of the real access samples of a client and attack models of a set of cohort clients. The second one is built upon a discriminative paradigm and consists of building separate client-specific SVM models.

Client-specific methods can not be used to protect the biometric verification system from

spoofing attacks at enrollment time, when the models are not yet created. If a protection is needed at this point, a client-independent anti-spoofing system should be used at the cost of a lower performance. However, when the verification system is in operation mode, the client-specific anti-spoofing systems can be used as soon as the client claims his identity.

The performance of the client-specific anti-spoofing systems in particular case studies for the face mode is analyzed in Section 6.2 with respect to different parameters and compared to their client-independent counterparts.

4 Output-level Integration: Fusion of Experts

In practice, biometric verification and anti-spoofing systems have a common purpose: to prevent illegitimate access to a certain resource. When considering security aspects of a resource protected using biometrics, an anti-spoofing system provides an additional level of security by being able to cope with attacks of a kind a biometric verification system can not detect. Therefore, the decision whether a sample is accepted or rejected depends on both biometric verification and anti-spoofing systems.

An integration of biometric verification and anti-spoofing systems at output-level is required to obtain a single, unified, decision about a sample. The resulting system can be considered as a biometric verification system with increased robustness to spoofing. However, the verification performance of the integrated system, as well as its robustness to spoofing, are highly dependent on the way the two composing systems are combined. To optimize the performance, it is of great importance to explore different strategies to fuse them.

To tackle the output-level integration of biometric verification and anti-spoofing systems, we take a fusion of multiple experts approach. In biometrics, such approaches have been thoroughly investigated and widely deployed to create multibiometric systems, where combining multiple modes leads to less noise-sensitive and more accurate biometric recognition [Ross et al., 2008]. Yet, fusion of a biometric verification system with an anti-spoofing one is a problem which is addressed much less frequently.

The fusion of biometric systems can happen at several points of the verification pipeline. With respect to this, the fusion techniques can be categorized as: *sensor-level*, *feature-level*, *score-level*, *rank-level* and *decision-level* [Ross et al., 2008]. Without intervening in the internal logic of biometric verification and anti-spoofing systems, their fusion can be realized only at score-level and decision-level. At decision-level, the biometric verification and anti-spoofing systems output decisions d_v and d_s, respectively, based on their own decision threshold, while the fusion module needs to unify the two decisions into one. At score level, the fusion module takes raw scores s_v and s_s produced by the verification and anti-spoofing systems, respectively, combines them into a single score s_f and computes a single decision threshold. The workflow

of the fusion is illustrated in Fig. 4.1.

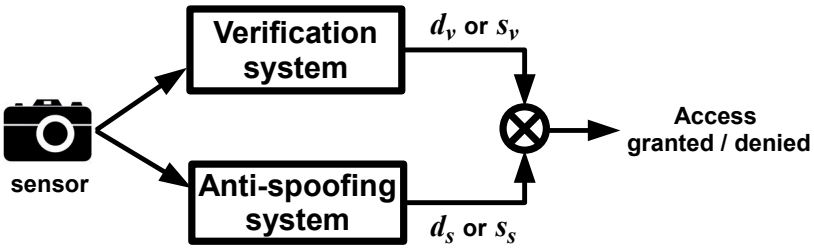


Figure 4.1: Flow diagram of the output-level integration of biometric verification and anti-spoofing systems

An overview of the existing literature on fusion of multiple experts, with an emphasis on the fusion of biometric verification and anti-spoofing systems, is given in Section 4.1. The fusion strategies explored in this work are detailed in Section 4.2.

4.1 Summary of Fusion Methods in Biometrics

We commence this section with an overview of the fusion techniques used for the outputs of biometric verification systems, regardless of the source of the input and the purpose of the fusion. Then, we cover several works where the fusion is targeting biometric verification and anti-spoofing systems.

When performing the output-level integration of biometric verification and anti-spoofing systems, we assume that they provide an output in the form of a decision or a score. Therefore, in this section, we cover decision-level and score-level fusion strategies.

4.1.1 Biometric Verification Systems

Decision-level fusion methods operate on the decisions of the different systems. The simplest strategies are logical AND and OR rules. Logical AND fusion rule results in a positive decision only if all the systems agree on a positive decision. To obtain a positive decision using logical OR fusion rule, it is enough that just one of the systems responds positively. Majority voting and weighted majority voting take the decision based on a linear combination of the decisions of the separate systems. More complex schemes rely on transforming the discrete decisions into continuous probability values or degrees of belief [Ross et al., 2008].

The score-level fusion methods operate on the scores of the separate systems, producing a single score as output of the integrated system. They can be categorized as *fixed*, *density-based* and *learning-based* rules [Roli et al., 2002]. The fixed fusion rules perform a simple mathematical operations on the scores of the separate systems, like sum, product, median,

minimum, maximum etc. These rules require that the scores of the multiple experts are brought into a common domain by performing normalization. The density-based fusion rules use Bayesian theory and compute the likelihood of the set of scores from the separate systems under the hypothesis of the different classes. The class is then determined using likelihood ratio. A theoretical foundation of the density-based rules is given by Kittler et al. [1998] and is used as a framework to derive fixed rules. Finally, the learning-based fusion rules use the scores as an input to a classifier who takes the final decision. Examples include Support Vector Machines (SVM) [Ben-Yacoub et al., 1999; Fierrez-Aguilar et al., 2003], Linear Discriminant Analysis (LDA) [Ben-Yacoub et al., 1999; Ross and Jain, 2003; Wang et al., 2003], decision trees [Ben-Yacoub et al., 1999; Ross and Jain, 2003], multi-layer perceptron [Ben-Yacoub et al., 1999; Wang et al., 2003], Behavior Knowledge Space [Roli et al., 2002; Vatsa et al., 2010] and many more.

As the biometric systems involved in fusion may be of different quality and may give responses with different levels of confidence, their scores are sometimes combined in a weighted manner [Jain and Ross, 2002]. Each of the systems participates in the fusion with a certain weight which is trained or obtained heuristically.

Jain and Ross [2002] were the first to explore fusion tailored to each client. They concluded that the performance of multimodal systems can be improved if client-specific thresholds are employed, or client-specific weights for each of the fused modes are estimated. The trend was followed by Fierrez-Aguilar et al. [2003] who shifted the approach towards the training step by creating client-specific learning-based fusion schemes with SVM. Client-specific training is used also by Kumar and Zhang [2009] who, together with the scores to be fused, supply the client identity to a Feed Forward Neural Network. Toh et al. [2004] introduce the combination of local learning and local decision, both of which refer to incorporating the client identity in the process. Finally, Fierrez-Aguilar et al. [2005] propose adapted score fusion, which combines the scores obtained by separately trained client-independent and client-specific fusion rules.

The aforementioned fusion techniques have been extensively used for fusion of systems based on multiple biometric modes. According to Ross et al. [2008], one of the benefits that such a fusion brings is increased robustness to spoofing attacks of the multimodal system. The intuition behind this reasoning is that in a multimodal system, an attacker faces the difficult task of having to spoof more than one mode. However, in many cases spoofing only one of the modes can be enough, as discussed by Rodrigues et al. [2009]; Johnson et al. [2010]; Akhtar et al. [2012]. This is highly dependent on the used fusion algorithm, which has inspired fusion schemes specifically designed to increase the robustness to spoofing. Most notable are the ones presented in [Rodrigues et al., 2009] which explores fuzzy logic, and [Rodrigues et al., 2010] which extends the likelihood ratio fusion rule by introducing hidden variables denoting the probability of a spoofing attack.

4.1.2 Biometric Verification and Anti-spoofing Systems

Prior work on integration of biometric recognition and anti-spoofing systems has not been as extensive. The first publication treating this topic is by Marasco et al. [2011], who employ anti-spoofing mechanism before the verification stage on a multibiometric system composed of three fingerprint and one face modality. If a spoofing attack is detected for one modality, the corresponding unimodal system does not contribute to the final fusion of the multimodal system, which is performed at score-level.

The first attempt to fuse biometric verification and anti-spoofing systems on a single mode is done by Marasco et al. [2012]. It is important to note that the proposed methods are targeting systems where spoofing is possible both in enrollment and verification stage. Therefore, the anti-spoofing scores of both the input sample, as well as an enrollment sample that is used for comparison, are taken into consideration. The authors analyze four different fusion methods and evaluate them on the fingerprint mode. The first two methods operate at decision-level by sequentially employing a fingerprint verification and anti-spoofing system, or the other way around, which is equivalent to the logical AND fusion rule. The third method targets score-level fusion and is a learning-based method, performing classification on the verification and anti-spoofing scores using several different classifiers. Finally, the fourth one belongs to the density-based fusion rules and models the likelihood of the verification and anti-spoofing scores under certain hypotheses. Interestingly, the developed Bayesian model assumes a dependence of the verification scores on the anti-spoofing scores.

Using a density-based score fusion and under certain hypotheses, Rattani and Poh [2013] model the likelihood of three types of scores: verification, anti-spoofing and image quality score. All of them are considered in the Bayesian model, which additionally assumes that anti-spoofing and image quality scores influence the verification score as well. The densities in the Bayesian model are estimated using GMM, Gaussian Copula and Quadratic Discriminant Analysis (QDA). In an extended version of the approach, the Bayesian model also considers sensor characteristic, which is assumed to influence the verification, anti-spoofing and quality scores [Rattani et al., 2013].

The assumption that spoofing attacks can exist among the enrollment samples, present in the works of Marasco et al. [2012] Rattani and Poh [2013] and Rattani et al. [2013], is reasonable in some cases, but is unsuitable for systems where the verification matching is done with respect to a model instead of a single sample. Furthermore, it poses certain limitations, like the necessity of spoofing samples at enrollment time.

4.2 Fusion Strategies for Biometric Verification and Anti-Spoofing Systems

When fusing biometric verification systems, they share a common reasoning about which samples need to be rejected and which accepted. All of them are trained to reject zero-

effort impostors, and accept genuine accesses. However, to address the fusion of biometric verification and anti-spoofing systems, we have to keep in mind that they are of different nature and are discordant with respect to the scores they assign to samples and what they consider as a positive or negative class. As verification systems are trained to take their decision based on the identity present in the sample, they will give low scores to zero-effort impostors (and hence classify them as negative), but high scores to both genuine accesses and spoofing attacks (and hence classify them as positive). On the other hand, anti-spoofing systems are trained to take their decision based on whether the sample looks genuine or not. Therefore, they will assign low scores to spoofing attacks (and hence classify them as negative), but high scores to both genuine accesses and zero-effort impostors (and hence classify them as positive).

The antagonistic nature of the systems to be fused is illustrated in Table 4.1. The final, fused system, needs to join their decision in a way that will consider only genuine users as positive and both zero-effort impostors and spoofing attacks as a negative class. Based on this observation, we can safely consider both zero-effort impostors and spoofing attacks as a single, enhanced, negative class and use fusion rules that will transform the scores of the two systems accordingly.

Additionally, weighted fusion schemes, where different weights are given to different systems, are not suitable for fusion of biometric verification and anti-spoofing systems. When fusing biometric verification systems, the individual systems operate in a competitive manner, giving responses to a verification problem. In this context, weighting is justified by weaknesses of some of the systems and the different quality of their responses. On the other hand, biometric verification and anti-spoofing systems operate in a collaborative manner where each of them is an expert in a particular domain of the problem and their individual responses are equally important. Certainly, an exception can be made when one can predefine the relative importance of the two systems, depending on the application.

4.2.1 Decision-Level Fusion

Table 4.1 may give hints about the fusion schemes that can be used. In the case of the decision-level fusion approach, the positive class needs to be accepted by both the verification and anti-spoofing system, while for the negative class a rejection from one of the systems is enough. Thus, logical AND fusion rule should be an appropriate choice for decision-level fusion of the two systems.

4.2.2 Score-Level Fusion

Keeping in mind that the biometric verification and anti-spoofing systems give high scores to different classes of samples, then many score-level fusion schemes typically used in biometrics should be applied with care. An example are the density-based score fusion methods [Ross

Table 4.1: Criteria for positive and negative class of a typical verification, anti-spoofing and fused system

	Genuine users	Impostors	Spoofing attacks
Verification system	+	-	+
Anti-spoofing system	+	+	-
Fused system	+	-	-

et al., 2008]. There, the class is determined by computing its *a posteriori* probability given the set of scores from multiple classifiers. If independence is assumed for the scores of the biometric verification and anti-spoofing systems, than the *a posteriori* probability of the class is proportional to the product of the likelihoods of the separate scores. The *a posteriori* probability that a spoofing attack is a positive sample may be high for the verification classifier, but very low for the anti-spoofing system. On the contrary, the *a posteriori* probability that a zero-effort impostor is a positive class may be low for the verification classifier, but high for the anti-spoofing system. As a result, even applying fixed fusion rules as derived by Kittler et al. [1998] may produce fused scores which are not discriminative enough for correct final decisions. In this work, we consider fusion using SUM rule, where the fused score is obtained by summing the scores of the individual systems.

Using one of the anti-spoofing and verification systems for the face mode that will be described in Section 6.1 and applied on Replay-Attack database, a scatter plot of the scores of the two systems is given in Fig. 4.2. The clear clusters of the samples coming from genuine users, zero-effort impostors and spoofing attacks in the 2D space defined by their verification and anti-spoofing scores s_v and s_s respectively, suggests using a learning-based fusion rule. Hence, the score pair is considered as a feature vector $\mathbf{x} = \begin{pmatrix} s_v & s_s \end{pmatrix}^\top$, which is fed to an appropriate classifier.

We select Logistic Regression (LR) [David and Stanley, 2000] as the first learning-based fusion rule, which fits a logistic hypothesis function given in Eq. 4.1 by estimating its parameters Θ. The parameters Θ are in a linear relation with the input variables $\mathbf{x}$.

$$h_\Theta(\mathbf{x}) = \frac{1}{1 + e^{-\Theta^\top \mathbf{x}}} \tag{4.1}$$

Considering the non-linear separation between the clusters in Fig. 4.2, we perform experiments with Polynomial Logistic Regression (PLR) as well. In this case, the parameters Θ are in a polynomial relation with the input variables, i.e. $\mathbf{x} = \begin{pmatrix} s_v & s_s & s_v s_s & s_v^2 & s_s^2 \end{pmatrix}^\top$ in Eq. 4.1.

Finally, it is possible to use a density-based score fusion method over the score pair $\mathbf{x}$. In this approach, class-conditional densities are estimated for the positive and the negative class and the final decision is taken using log-likelihood ratio. The class-conditional densities are

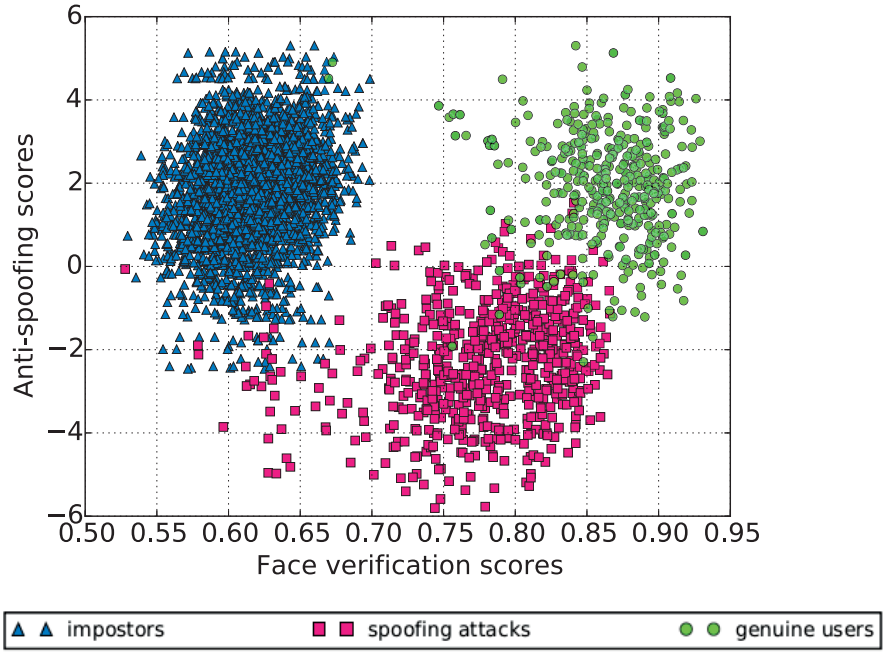


Figure 4.2: Example scatter plot of biometric verification and anti-spoofing system scores on Replay-Attack

modeled using GMM. This approach is similar to [Rattani and Poh, 2013], with the exception that the quality score of the sample is not taken into account.

Some of the presented score-level fusion techniques require that the scores from the systems to be fused are on comparable scales [Ross et al., 2008]. Score normalization techniques are helpful to bring the scores in a common domain. We apply z-normalization to the systems' scores, as in Eq. 4.2, where μ refers to the mean of a set of training scores, while σ refers to their standard deviation.

$$s_{norm} = \frac{s - \mu}{\sigma} \tag{4.2}$$

4.2.3 Implementation Details

A significant implementation difference emerges when fusing biometric verification system with a client-independent and client-specific anti-spoofing system, both of which are described in Chapter 3. In the case of a client-independent anti-spoofing system, the claim of the client identity is irrelevant for the anti-spoofing score. However, this claim plays an essential role in the case of client-specific anti-spoofing systems. This observation is particularly

important when fusing the scores for the zero-effort impostors. In that case, the input sample needs to be scored against an anti-spoofing model of another client, in the same way it is matched with the model of another client by the verification system.

4.3 Discussion

In a biometric verification scenario where spoofing attacks can be expected, the decision to accept or reject an input sample is a task that has to be done jointly by the biometric verification and anti-spoofing systems. The unification of the individual outputs of the two systems is referred to as output-level integration and can involve fusion of the decisions or the scores of the separate systems.

Very few attempts for output-level integration of biometric verification and anti-spoofing systems have been done in the past. On the contrary, there are numerous methods for fusion of biometric verification systems with a theoretically and empirically grounded reputation. Keeping in mind the particularities of the fusion of biometric verification and anti-spoofing systems, we select and adapt a subset of them.

The impact of the fusion, as well as the importance of selection of an optimal fusion method for a particular combination of biometric verification and anti-spoofing systems will be experimentally demonstrated in case studies for the face mode in Section 6.3.

5 Integrated Evaluation of Biometric Verification Systems Under Spoofing Attacks

Similarly to other stages of the operation of biometric verification and anti-spoofing systems, their evaluation is usually performed independently. Typically, the evaluation conventions for binary classification systems are followed for both of them.

An isolated evaluation of anti-spoofing systems gives an important insight about the discriminative abilities of certain anti-spoofing features and classification methods. An isolated evaluation of biometric verification system gives its verification performance. However, each of these assessments gives just a partial understanding of the strength of a system, especially if high security standards are required. The evaluation of such a system needs to report, in a unified way, both the verification performance, as well as its *spoof-ability*, which is equivalent to its vulnerability to spoofing attacks. The existence of three classes instead of two at the input of the biometric verification system, requires a redefinition of the evaluation task.

When devising an evaluation methodology for trustworthy biometric verification system, there are several system design considerations that need to be taken into account. One of them is the fact that the probability of attacks, or the cost of incorrectly accepted spoofing attacks and zero-effort impostors, as well as incorrectly rejected genuine users, depend on the environment where the system will be deployed. For example, spoofing attacks targeting a portable device may be more frequent than attacks at border control systems which are supervised by a human. A suitable evaluation metric needs to provide a mechanism to parameterize with respect to this variable. As an additional requirement, such a metric needs to provide an *a priori* evaluation, where no information from the test set is used during the system design.

In this chapter, we propose a novel evaluation framework for biometric verification systems under spoofing attacks, called Expected Performance and Spoofability (EPS) framework and considering all the criteria imposed by the problem. To do this, we first review the standards for evaluation of biometric systems in their common setup as binary classifiers. Then, we inspect previous efforts to adapt them to the redefined evaluation task, reporting on their drawbacks for deployment in real world conditions. Finally, we describe the EPS Curve (EPSC) and demonstrate its suitability for evaluation of biometric verification systems under spoofing

attacks. The evaluation methods covered in this chapter, including the newly proposed evaluation framework, are illustrated on a hypothetical verification system whose scores are generated artificially.

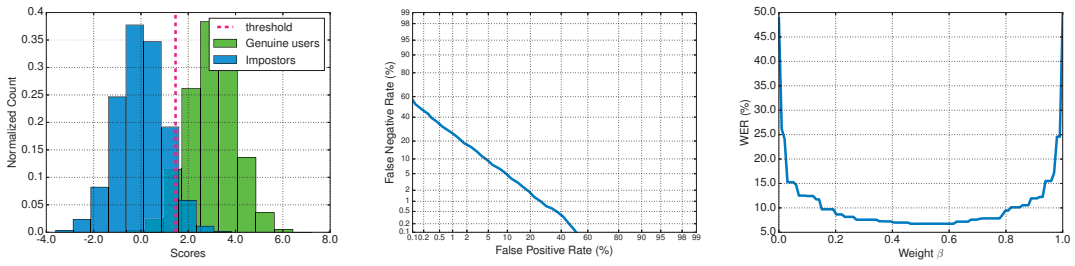
In Section 5.1, we start by surveying the standard evaluation metric for binary classification problems, as a basis for the widely accepted methodology for evaluation of biometric verification and anti-spoofing systems. A definition of the problem of evaluation of biometric verification systems under spoofing attacks, together with the commonly used evaluation methodologies, is given in Section 5.2. Section 5.3 describes the EPS evaluation framework and illustrates its usage and interpretation on a hypothetical biometric verification system. Further examples of its usage in some special cases are given in Section 5.4.

5.1 Summary of Evaluation Metrics and Methodologies in Biometrics

As both biometric verification and anti-spoofing systems by themselves are of binary nature, the overview of the evaluation metrics in biometrics will firstly cover the standard metrics for evaluation of binary classification systems. Then, the adaptations of the general metrics to the specific tasks of biometric verification and anti-spoofing will be given.

5.1.1 Evaluation of Binary Classification Systems

Binary classification systems receive two types of input belonging to two classes, usually referred to as positive and negative class. They are trained to assign scores to the input samples. Then, a decision threshold is calculated to separate the scores of the positive and the negative class and the samples with scores above the threshold are classified as positives, while the ones with scores below the threshold as negatives. Fig. 5.1a shows a plot of the score distributions for the two classes, including the decision threshold.



(a) Score distribution (b) DET curve (c) EPC curve

Figure 5.1: Evaluation plots for a hypothetical binary verification system

Metrics for evaluation of binary classification systems are associated to the types of errors

they commit and how to measure them, as well as to the threshold calculation and evaluation criterion [Poh and Bengio, 2006]. Binary classification systems are subject to two types of errors: False Positive (FP) and False Negative (FN). Typically, the error rates that are reported are False Positive Rate (FPR), which corresponds to the ratio between FP and the total number of negative samples and False Negative Rate (FNR), which corresponds to the ratio between FN and the total number of positive samples.

An objective and unbiased performance evaluation of the binary classification systems requires a database with a specific design and strictly defined protocols. It is recommended that the samples in the database are divided into three subsets: training $\mathscr{D}_{train}$, development (validation) $\mathscr{D}_{dev}$ and test (evaluation) set $\mathscr{D}_{test}$ [Hastie et al., 2001]. Even greater objectivity will be achieved if the identities in separate subsets do not overlap [Lui et al., 2012]. The training set serves to train the system, while its fine tuning is done using the development set. Since in a real world scenario the final system will be used for data which have not been seen before, the performance measure is normally reported on the test set [Hastie et al., 2001; Bailly-Baillire et al., 2003]. An exception from this recommended design may happen if the number of samples in the database is not big enough. In such a case, the samples can be divided only in training and test set, and tuning of the parameters is done with a cross-validation procedure [Hastie et al., 2001].

The decision threshold τ is computed to serve as a boundary between the output scores of the positive and the negative class. By changing this threshold one can balance between FPR and FNR: increasing FPR reduces FNR and vice-versa. However, it is often desired that an optimal threshold τ^* is chosen according to some criterion. One well established criterion is Equal Error Rate (EER) [Poh and Bengio, 2006], which selects the threshold τ^*_{EER} to ensure that the difference between FPR and FNR is as small as possible (Eq. 5.1). The optimal threshold, also referred to as *operating point*, is a tuning parameter, and it is usually determined using the development set [Hastie et al., 2001; Bailly-Baillire et al., 2003].

$$\tau^*_{\text{EER}} = \arg\min_{\tau} |\text{FPR}(\tau, \mathscr{D}_{dev}) - \text{FNR}(\tau, \mathscr{D}_{dev})| \tag{5.1}$$

Once the threshold τ^* is determined, the accuracy of the system can be summarized reporting different metrics. For example, the Detection Cost Function (DCF), given in Eq. 5.2, has been proposed by Martin and Przybocki [2000] and is used in the NIST evaluations [Przybocki et al., 2006]. The DCF accounts for the cost of the error rates (c_{FPR} and c_{FNR}), as well as for the probability of occurrence of positive and negative samples (p_{pos} and p_{neg}).

$$\text{DCF}(\tau^*, \mathscr{D}_{test}) = c_{\text{FPR}} \cdot p_{\text{neg}} \cdot \text{FPR}(\tau^*, \mathscr{D}_{test}) + c_{\text{FNR}} \cdot p_{\text{pos}} \cdot \text{FNR}(\tau^*, \mathscr{D}_{test}) \tag{5.2}$$

By giving equal priors to the occurrence of positive and negative samples and normalizing the cost values, Weighted Error Rate (WER) is proposed by Bailly-Baillire et al. [2003]. In its computation (Eq.5.3), $\beta \in [0,1]$ is the parameter balancing between the cost of FPR and FNR. For the special case of $\beta = 0.5$, the Half Total Error Rate (HTER) is reached.

$$\mathrm{WER}_\beta(\tau^*, \mathscr{D}_{test}) = \beta \cdot \mathrm{FPR}(\tau^*, \mathscr{D}_{test}) + (1-\beta) \cdot \mathrm{FNR}(\tau^*, \mathscr{D}_{test}) \tag{5.3}$$

Important tools in evaluation of classification systems are the different graphical representations of the classification results. For example, to present the trade-off between FPR and FNR depending on the threshold, the performance of the binary classification systems is often visualized using Receiver Operating Characteristic (ROC) curve. Parameterizing over different values for the decision threshold, the ROC curve usually plots FPR versus 1-FNR. Sometimes, when one number is needed to represent the performance of the system in comparison with other systems, the Area Under ROC curve (AUC) may be reported. The higher the AUC the better the system.

A normal deviate transformation of the ROC curve yields the Detection-Error Trade-off (DET) curve [Martin et al., 1997]. Its usage is convenient for comparing systems whose scores follow a Gaussian distribution, since such a transformation guarantees that the curve will become a line. It plots FPR versus FNR. Fig. 5.1b illustrates the DET curve for a hypothetical binary classification system.

Although ROC and DET curves may give an idea about the expected performance of a single system under different thresholds, using them to compare two or more systems can lead to biased conclusions [Bengio et al., 2005]. Usually, when comparing two systems using ROC or DET curves, we select a certain value on the abscissa (most often FPR) as a first step, and then we read the values on the ordinate for the two systems (for example FNR) as a second step. In this way, during the first step, we implicitly choose a threshold *a posteriori*, i.e. on the same data used to read and compare the error rates in the second step. This threshold may not be the optimal one for any of the two systems. However, for an objective comparison, the error rates for the two systems have to be reported at their optimal thresholds, which have to be chosen *a priori*, on a separate data. Unfortunately, by plotting only the error rates on a test set at thresholds not related to the development set, the ROC and DET curves do not give any hint about the optimal thresholds of the two systems. Hence, the conclusions about which one out of two systems is better may be misleading if drawn solely from the ROC or DET curves.

To solve this issue, the so-called Expected Performance Curve (EPC) is proposed by Bengio et al. [2005]. It fills in for two main disadvantages of the ROC and DET curves: firstly, it plots the error rate on the test set depending on a threshold selected *a priori* on the development set; and secondly, it accounts for varying relative cost $\beta \in [0;1]$ of FPR and FNR when calculating the threshold. In the EPC framework, an optimal threshold τ^*_β depending on β is computed

based on a certain criteria on the development set. For example, the threshold can be chosen to minimize WER_β for different values of β, which is the variable parameter plotted on the abscissa. The performance for the calculated values of τ^*_β is then computed on the test set. WER_β or any other measure of importance can be plotted on the ordinate axis. The parameter β can be interpreted as the cost of the error rates, but also as the prior of having a positive or a negative sample as an input. One may observe the error rates and compare systems only in the range of values of β which are of interest for a particular application. The EPC curve is illustrated in Fig. 5.1c for a hypothetical binary classification system.

The performance of a binary system can be summarized in one value by computing the area under the EPC, defined as the expected average of two antagonistic error rates that are being plotted [Bengio et al., 2005].

5.1.2 Evaluation of biometric verification systems

In the domain of biometric verification, the positive and the negative class refer to genuine users and zero-effort impostors, respectively. Accordingly, the number of errors known as FP and FN refer to the number of zero-effort impostors incorrectly classified as genuine users and the number of genuine users incorrectly classified as zero-effort impostors, respectively. Since the positives and the negatives are associated with the action of *acceptance* and *rejection* by the verification system, a common practice is to replace FPR and FNR with False Acceptance Rate (FAR) and False Rejection Rate (FRR), respectively [Jain and Ross, 2008]. Furthermore, due to the process of matching between the samples and the models, FPR and FNR are often reported as False Match Rate (FMR) and False Non-Match Rate (FNMR) [Mansfield et al., 2002]. It is important to note that, in general, the error rates FMR and FNMR are not exactly synonymous with FAR and FRR. However, they can be considered as equivalent in a technology evaluation, which is performed on a pre-collected database [Mansfield et al., 2002; Jain et al., 2008].

5.1.3 Evaluation of anti-spoofing systems

The role of a positive and a negative class in the domain of anti-spoofing is taken by real accesses and spoofing attacks, respectively. The anti-spoofing systems work on the principle of acceptance and rejection as well. Hence, in this scope, FAR and FRR are the most commonly used terms for FPR and FNR too. FAR stands for the ratio of incorrectly accepted spoofing attacks and FRR for the ratio of incorrectly rejected real accesses. These error rates are often substituted with different synonyms by different authors. The most common of them are listed in Table 5.1.

Table 5.1: Typically used error rates for anti-spoofing systems and their synonyms.

Error rate	Acronym	Synonyms
False Positive Rate	FPR	False Acceptance Rate (FAR), False Spoof Acceptance Rate [Marasco et al., 2012], False Living Rate (FLR) [Galbally et al., 2012]
False Negative Rate	FNR	False Rejection Rate (FRR), False Alarm Rate [Pan et al., 2007], False Live Rejection Rate [Marasco et al., 2012], False Fake Rate (FFR) [Galbally et al., 2012]
True Positive Rate	TPR	True Acceptance Rate
True Negative Rate	TNR	True Rejection Rate, detection rate [Pan et al., 2007], [Bao et al., 2009a], Wang et al. [2009], detection accuracy [Zhang et al., 2011]
Half Total Error Rate	HTER	Average Classification Error (ACE) [Galbally et al., 2012]

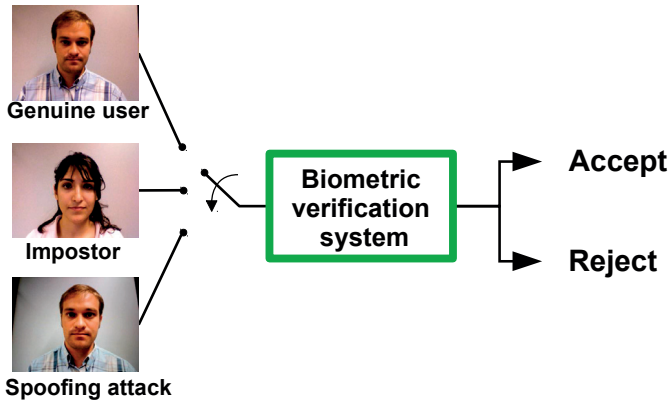


Figure 5.2: Biometric verification system under spoofing attack

5.2 Evaluation of Biometric Verification Systems Under Spoofing Attacks

When treating biometric verification as a binary classification system, the designers are interested in determining the capacity of a given system to discriminate between different identities. Depending on the internal algorithm, these systems may or may not have the competence to discover if the input sample comes from a live person present in front of the system, or a spoofing attack. Nevertheless, the evaluation of biometric verification systems needs to account for the existence of an additional input type, spoofing attacks.

An accurate representation of the operation of a verification system acknowledging the spoofing attack samples as a possible input type, is given in Fig. 5.2. It needs to accept only the samples from the class of genuine users, while both zero-effort impostors and spoofing attacks need to be rejected. Consequently, the system is not necessarily required to be able to discrim-

inate between three classes and the problem does not need to be treated as ternary. Despite the comfort of keeping the binary nature of the verification system, it is still of importance to evaluate how vulnerable the system is to spoofing attacks. In the following, we describe the requirements of a database that can be used for evaluation of biometric verification systems under spoofing attacks, followed by an overview of the commonly used metrics and methodologies.

5.2.1 Database considerations

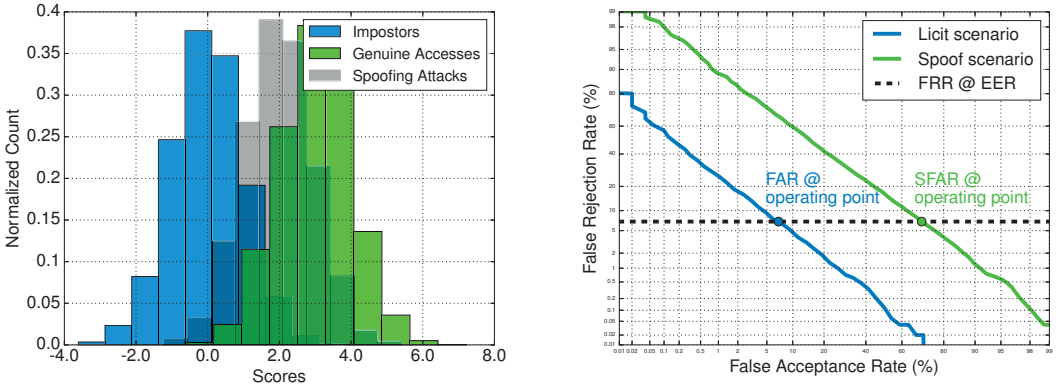
The evaluation the performance of a biometric verification system under spoofing attacks includes training and spoofability assessment of the system. Therefore, it entails a database which satisfies certain criteria. To enable training of a biometric verification system, the database needs to provide enrollment samples to enroll clients in the system. To enable spoofability assessment, it needs to provide spoofing attack samples as well.

To formalize the process of training and evaluating a verification system using a spoofing database, let's represent the identity $i \in \mathcal{Y}$ in the database with the tuple $(\mathrm{X}_i^r, \mathrm{X}_i^s, \mathrm{X}_i^e)$, containing real access X_i^r, spoofing attack X_i^s and enrollment X_i^e samples. Then, the spoofing database, providing data for the identities in $\mathcal{Y}$, can be denoted as $\mathcal{D} = \{(\mathrm{X}_i^r, \mathrm{X}_i^s, \mathrm{X}_i^e) : i \in \mathcal{Y}\}$. The process of training a verification system using the spoofing database means creating a set of models $\mathcal{M} = \{\mathcal{M}_i : i \in \mathcal{Y}\}$, where $\mathcal{M}_i = f(\mathrm{X}_i^e)$ and $f(\cdot)$ is a function that maps samples to a model. Then, the verification system computes the scores for the classes of real accesses, zero-effort impostors and spoofing attacks. The set of scores for the genuine users may be created by comparing the real access samples of one identity to the model of the same identity: $\mathcal{S}_{genuine} = \{g(\mathrm{X}_i^r, \mathcal{M}_i) : i \in \mathcal{Y}\}$, where $g(\cdot,\cdot)$ is a matching function. A logical way to assemble the set of zero-effort impostor scores is by comparing the real access samples of one identity to the models of the other identities in an exhaustive manner (full cross-comparison [Mansfield et al., 2002]), which results in $\mathcal{S}_{impostor} = \{g(\mathrm{X}_i^r, \mathcal{M}_j) : i, j \in \mathcal{Y}, i \neq j\}$. Finally, to assemble the set of spoofing attack scores for the verification system, one needs to compare the spoofing attack samples from one identity to the model of the same identity, which yields $\mathcal{S}_{spoof} = \{g(\mathrm{X}_i^s, \mathcal{M}_i) : i \in \mathcal{Y}\}$.

5.2.2 Summary of Evaluation Metrics and Methodologies for Biometric Verification Systems Under Spoofing Attacks

The evaluation methodologies for biometric verification systems under spoofing attacks need to address two challenges. The first one, observed in Fig. 5.3a, is related to the existence of three score distribution plots corresponding to the three input classes. The challenging question is how to determine the decision threshold to discriminate between the samples to accept and reject. The second challenge refers to the error rates and evaluation metrics for verification systems under spoofing attacks. It emerges because the evaluation metrics presented in Section 5.1.2 are sufficient to describe only the verification performance of a

system. However, the presence of spoofing attacks entails a suitable metric to accompany FAR and FRR and report on the system spoofability.



(a) Score distribution

(b) DET curve

Figure 5.3: Evaluation plots for a hypothetical biometric verification system under spoofing attacks

A widely accepted strategy to address these challenge is to simplify the problem and decompose it into two sub-problems which resemble the original binary classification problem in biometric verification. The sub-problems correspond to two scenarios the system can operate in:

- *Licit* scenario (also called normal operation mode Galbally et al. [2006]): considers genuine users as positive and only zero-effort impostors as negative class,
- *Spoof* scenario: considers genuine users as positive and only spoofing attacks as negative class.

Then, the decision threshold can be determined and evaluation can be performed under the two scenarios in different ways. Unfortunately, while performance metrics for verification systems are well established and widely used, the metrics for verification systems under spoofing attacks are not unified and is ambiguous in different publications. A detailed overview of all the error rates utilized by various authors is given in Table 5.2.

The adopted terminology that we will adhere to is as follows:

- FRR - ratio of incorrectly rejected genuine users,
- FAR - ratio of incorrectly accepted zero-effort impostors,
- SFAR - ratio of incorrectly accepted spoofing attacks [Johnson et al., 2010].

Table 5.2: Typically used error rates for biometric verification systems under spoofing attacks and their synonyms.

Error rate	Acronym	Scenario	Synonyms
False Negative Rate	FNR	Both	False Rejection Rate (FRR), False Non-Match Rate [Galbally et al., 2010], [Marasco et al., 2012], Pmiss [Villalba and Lleida, 2011])
		Both	Global False Rejection Rate (GFRR) [Marasco et al., 2012]
True Positive Rate	TPR	Both	True Acceptance Rate, Genuine Acceptance Rate [Johnson et al., 2012], [Rodrigues et al., 2010]
False Positive Rate	FPR	Licit	False Acceptance Rate (FAR), False Match Rate [Galbally et al., 2010], [Marasco et al., 2012], Pfa [Villalba and Lleida, 2011]
		Spoof	False Acceptance Rate (FAR) [Galbally et al., 2006], Spoof False Acceptance Rate [Johnson et al., 2010], Liveness False Acceptance Rate [Adler and Schuckers, 2009], Success Rate [Ruiz-Albacete et al., 2008], Attack Success Rate [Galbally et al., 2010]
		Both	System False Acceptance Rate (SFAR) [Adler and Schuckers, 2009], Global False Acceptance Rate (GFAR) [Marasco et al., 2012]

With respect to the decision threshold, researchers generally follow two main evaluation methodologies to compute it and to report the error rates it produces, and they are discussed below.

Evaluation Methodology 1. In the first evaluation methodology, two decision threshold calculations are performed separately for the two scenarios [Matsumoto et al., 2002], [Galbally et al., 2006], [Johnson et al., 2010], [Alegre et al., 2012]. Analysis of the system in the licit scenario gives values for FRR and FAR, while analysis in the spoof scenario gives values for FRR and SFAR. Since the analysis produces different threshold in the two scenarios, the two values of FRR are not the same. A major weak point of this type of evaluation is that it outputs two decision thresholds for a single verification system, while naturally a single system can have only one operating point corresponding to one decision threshold. Furthermore, the spoof scenario assumes that all the possible misuses of the system come from spoofing attacks, which in general is not realistic. The threshold calculated in this scenario is not a good discriminating point for a verification system, but rather for an anti-spoofing system and the error rates reported on this way are not a reliable estimate of the system performance under spoofing attacks. The decision threshold and the reported error rates in the spoof scenario

are irrelevant in a real-world scenario. Therefore, this type of evaluation is not compliant to a real-world requirements for operation of a verification system.

Evaluation Methodology 2. The second evaluation methodology is adapted for more realistic performance evaluation. The threshold is calculated using various criteria, for example EER, but almost always using the licit scenario, as it is regarded as a normal operation mode for a verification system. Taking advantage of the fact that the licit and spoof scenario share the same positive class, many publications choose a threshold to achieve a particular desired value of FRR [Galbally et al., 2010; Villalba and Lleida, 2011; Ruiz-Albacete et al., 2008; Rodrigues et al., 2009; Akhtar et al., 2011; Marasco et al., 2011]. Then, using the obtained threshold, FAR for the licit and SFAR in the spoof scenario are reported and compared.

On the hypothetical verification system whose score distribution is plotted in Fig. 5.3a, the threshold is chosen using the EER criteria for the licit scenario. The plotted threshold gives an intuition about how well the system discriminates between genuine users and zero-effort impostors, but also between genuine users and spoofing attacks. Fig. 5.3b draws two DET curves corresponding to the two scenarios. The horizontal line shows the FRR for the chosen threshold. The points where it cuts the DET curves for the two scenarios are the reported error rates.

As an alternative figure delivering similar information as DET for the second evaluation methodology, Rodrigues et al. [2009] suggests to plot FAR vs. SFAR. Thresholds are fixed in order to obtain all the possible values of FAR for the licit scenario and SFAR is computed in the spoof scenario and plotted on the ordinate axis. By plotting the curves for different verification systems, the plot enables to compare which one of them is less prone to spoofing given a particular verification performance.

The issue that the second methodology overlooks is that a system whose decision threshold is optimized for one negative class (usually, the zero-effort impostors), can not be evaluated in a fair manner for another negative class (spoofing attacks). Expectedly, such a threshold will be biased towards the single negative class used for its determination, causing unnecessary larger error rates for the other negative class. If the system is expected to be exposed to two classes of negatives in the test or deployment stage, it would be fair that both of them play a role in the decision of the threshold in the development stage. A novel evaluation methodology to tackle this issue is the subject of Section 5.3.

5.3 Expected Performance and Spoofability (EPS) Framework

Determining the decision threshold for biometric verification systems under spoofing attacks seems to be one of the major issues in the evaluation process. Neither the first, nor the second of the evaluation methodologies explained in Section 5.2.2 offer a method that determines an unbiased threshold applicable in a realistic verification scenario. A fair evaluation of a

system which needs to reject samples of two different classes is possible only if both of them are considered in the development stage. By neglecting the class of spoofing attacks when deciding on the threshold of the verification system, one deliberately exhibits blindness to the danger of spoofing attacks, thus potentially creating a system more vulnerable to spoofing. Moreover, in some cases a necessity may arise to add a cost to the error rates associated with the positive and the negative class, and this cost has to be considered in the process of computing a decision threshold as well.

The most straight-forward way to involve both negative classes (zero-effort impostors and spoofing attacks) in the threshold decision process, is simply to merge them together into a single negative super-class. However, the number of zero-effort impostors and spoofing attacks is highly dependent on the database and follows the database protocol. Hence, the ratio of the two classes into the super-class is different for different databases and can not be controlled. Furthermore, the super-class tends to be biased towards the component with more samples. For example, in a typical biometric verification database with N identities and M samples per identity, the number of zero-effort impostors will be $N \times (N-1) \times M$. On the other hand, if there is a single spoofing attack for any genuine sample in the database, the number of spoofing attacks will be $N \times M$. The above observations lead to the question of what the correct ratio of zero-effort impostors and spoofing attacks into the super-class of negatives is.

As a matter of fact, there may not be a single answer to that. Any ratio of the two negative classes may be valid depending on the deployment conditions. For example, in highly supervised conditions, like airport control gates, spoofing attacks are more difficult to perform, and hence unlikely. On the other hand, unsupervised verification systems of portable devices are much more exposed to spoofing attacks. Thus, tuning the operating point of any system depends on its expected usage scenario.

The message that the metrics DCF, WER_β and EPC convey sounds with the above reasoning for a biometric verification system. EPC obtains a decision threshold based on a parameter β which balances between FAR and FRR and reports the expected performance for a wide range of values for that parameter. The parameter β can be interpreted as the relative cost or importance of FAR and FRR, or the prior of the negative or the positive class. Using EPC, it is possible to compare algorithms depending on the importance of FAR and FRR in a certain usage scenario.

For evaluating biometric verification systems under spoofing attacks, we develop a method inspired by EPC. Being aware that the prior of zero-effort impostors and spoofing attacks can not be known in advance while developing an algorithm, we design an evaluation framework which measures the expected performance of the system for a range of values of a parameter which balances between FAR and SFAR. Moreover, analogously to EPC, we introduce another parameter which considers the cost of the error rates associated with the positive and the negative classes. As it measures both the verification performance and the vulnerability to

spoofing of a system and unifies them into a single value, the adapted evaluation scheme is called Expected Performance and Spoofability (EPS) framework.

The goal of the EPS framework is to analyze and plot error rates regarding the performance and spoofability of a verification system on a test set, with respect to a decision threshold taken on a separate development set. We define two parameters: $\omega \in [0,1]$, which denotes the relative cost of spoofing attacks with respect to zero-effort impostors; and $\beta \in [0,1]$, which denotes the relative cost of the negative classes (zero-effort impostors and spoofing attacks) with respect to the positive class. Using these, we introduce a measurement called FAR_ω, which is a weighted error rate for the two negative classes (zero-effort impostors and spoofing attacks). It is calculated as in Eq. 5.4.

$$\text{FAR}_\omega = \omega \cdot \text{SFAR} + (1-\omega) \cdot \text{FAR} \tag{5.4}$$

The optimal classification threshold $\tau^*_{\omega,\beta}$ depends on both parameters. It is chosen to minimize the weighted difference between FAR_ω and FRR on the development set, as in Eq. 5.5.

$$\tau^*_{\omega,\beta} = \arg\min_\tau |\beta \cdot \text{FAR}_\omega(\tau, \mathscr{D}_{dev}) - (1-\beta) \cdot \text{FRR}(\tau, \mathscr{D}_{dev})| \tag{5.5}$$

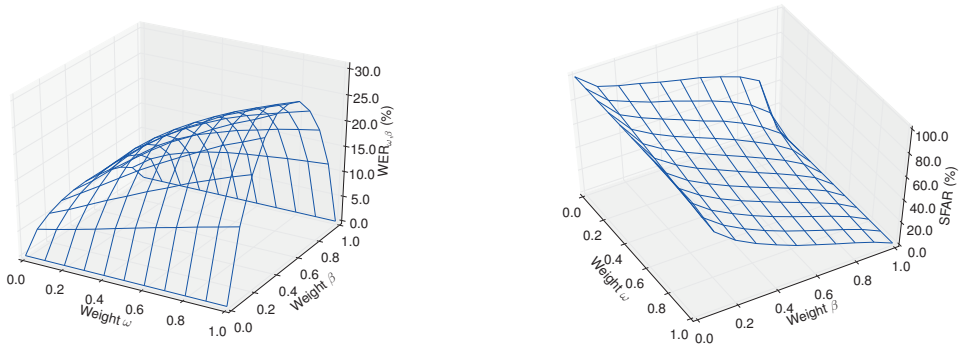
Once an optimal threshold $\tau^*_{\omega,\beta}$ is calculated for certain values of ω and β, different error rates can be computed on the test set. Probably the most important is $\text{WER}_{\omega,\beta}$, which can be accounted as a measurement summarizing both the verification performance and the spoofability of the system and which is calculated as in Eq. 5.6.

$$\text{WER}_{\omega,\beta}(\tau^*_{\omega,\beta}, \mathscr{D}_{test}) = \beta \cdot \text{FAR}_\omega(\tau^*_{\omega,\beta}, \mathscr{D}_{test}) + (1-\beta) \cdot \text{FRR}(\tau^*_{\omega,\beta}, \mathscr{D}_{test}) \tag{5.6}$$

A special case of $\text{WER}_{\omega,\beta}$, obtained by assigning equal cost $\beta = 0.5$ to FAR_w and FRR can be defined as HTER_ω and computed as in Eq. 5.7. In such a case, the criteria for optimal decision threshold is analogous to the EER criteria given in Section 5.1.1.

$$\text{HTER}_\omega(\tau^*_\omega, \mathscr{D}_{test}) = \frac{\text{FAR}_\omega(\tau^*_\omega, \mathscr{D}_{test}) + \text{FRR}(\tau^*_\omega, \mathscr{D}_{test})}{2} \tag{5.7}$$

The parameter ω could be interpreted as relative cost of the error rate related to spoofing attacks. Alternatively, it could be connected to the expected relative number of spoofing attacks among all the negative samples presented to the system. In other words, it could be



(a) WER$_{\omega,\beta}$ (b) SFAR

Figure 5.4: 3D plot of WER$_{\omega,\beta}$ and SFAR of a hypothetical biometric verification system computed using EPS framework

understood as the prior probability of the system being under a spoofing attack when it is misused. If it is expected that there is no danger of spoofing attacks for some particular setup, it can be set to 0. In this case, WER$_{\omega,\beta}$ corresponds to WER$_\beta$ in the traditional evaluation scheme for biometric verification systems. When it is expected that some portion of the illegitimate accesses to the system will be spoofing attacks, ω will reflect their prior and ensure they are not neglected in the process of determining the decision threshold.

As in the computation of WER$_\beta$ in Section 5.1.1, the parameter β could be interpreted as the relative cost of the error rate related to the negative class consisting of both zero-effort impostors and spoofing attacks. This parameter can be controlled according to the needs or to the deployment scenario of the system. For example, if we want to reduce the wrong acceptance of samples to the minimum, while allowing increased number of rejected genuine users, we need to penalize FAR$_\omega$ by setting β as close as possible to 1.

The EPS framework computes error rates for a range of decision thresholds obtained by varying the parameters ω and β. The visualization of the error rates parameterized over two parameters will result in a 3D surface, which, for a hypothetical system is given in Fig. 5.4. Using this plot, we can clearly infer on the expected error rates depending on the parameters' values or range of values which are of interest.

However, a 3D plot may not be convenient for evaluation and analysis when one needs to compare two or more systems. Instead, we suggest plotting the Expected Performance and Spoofability Curve (EPSC), showing WER$_{\omega,\beta}$ with respect to one of the parameters, while the other parameter is fixed to a predefined value. For example, we can fix the parameter $\beta = \beta_0$ and draw a 2D curve which plots WER$_{\omega,\beta}$ on the ordinate with respect to the varying parameter ω on the abscissa. Having in mind that the relative cost given to FAR$_\omega$ and FRR depends mostly

for $\beta \in [0,1]$ **do**
  for $\omega \in [0,1]$ **do**
    define $\text{FAR}_\omega = \omega \cdot \text{SFAR} + (1-\omega) \cdot \text{FAR}$
    $\tau^*_{\omega,\beta} = \arg\min_\tau |\beta \cdot \text{FAR}_\omega(\tau, \mathscr{D}_{dev}) - (1-\beta) \cdot \text{FRR}(\tau, \mathscr{D}_{dev})|$
    compute $\text{WER}_{\omega,\beta}(\tau^*_{\omega,\beta}, \mathscr{D}_{test})$;
    plot $\text{WER}_{\omega,\beta}(\tau^*_{\omega,\beta}, \mathscr{D}_{test})$ **w.r.t.** ω, β
  end for
end for

Figure 5.5: Pseudo code for computing $\text{WER}_{\omega,\beta}$

on the security preferences for the system, it is not difficult to imagine that particular values for β can be selected by an expert. Similarly, if the cost of SFAR and FAR or the prior of spoofing attacks with regards to the zero-effort impostors can be precisely estimated for a particular application, one can set $\omega = \omega_0$ and draw a 2D curve plotting $\text{WER}_{\omega,\beta}$ on the ordinate, with respect to the varying parameter β on the abscissa.

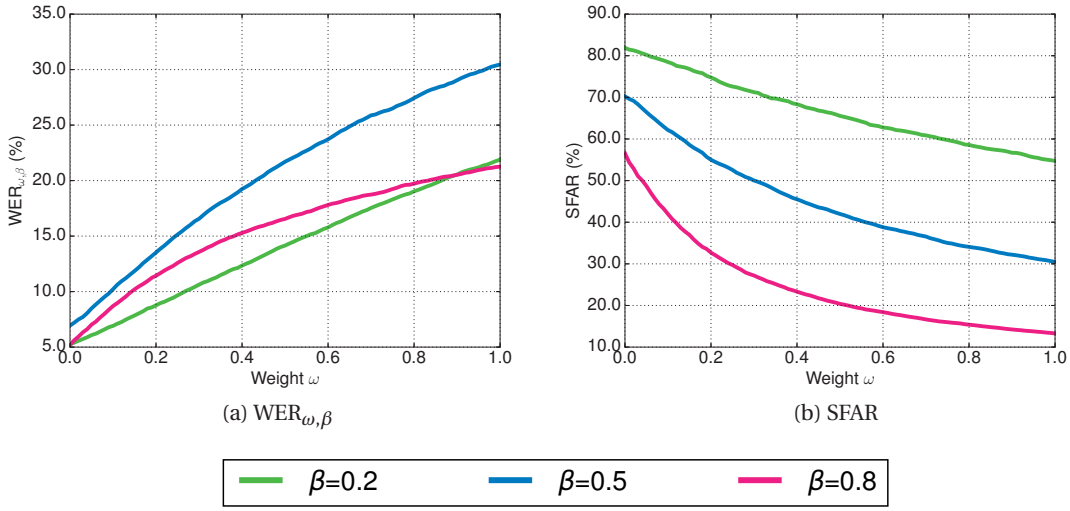
The algorithm on Fig. 5.5 gives the step-by-step procedure to compute and plot $\text{WER}_{\omega,\beta}$ with regards to ω and β for a given verification system. By fixing one of the parameters ω or β, one can plot EPSC for $\text{WER}_{\omega,\beta}$ with regards to the other parameter.

Besides $\text{WER}_{\omega,\beta}$, EPSC can present other error rates which are of interest. For example, plotting SFAR can show how the system's robustness to spoofing changes with regards to ω or β. Alternatively, to report on all the incorrectly accepted samples, FAR_ω can be plotted using EPSC.

Fig. 5.6 and Fig. 5.7 give an illustration of the EPSC plotting the error rates $\text{WER}_{\omega,\beta}$ and SFAR as function of the parameters ω and β, respectively. The plots are generated for the hypothetical verification system whose score distribution is given in Fig. 5.3a.

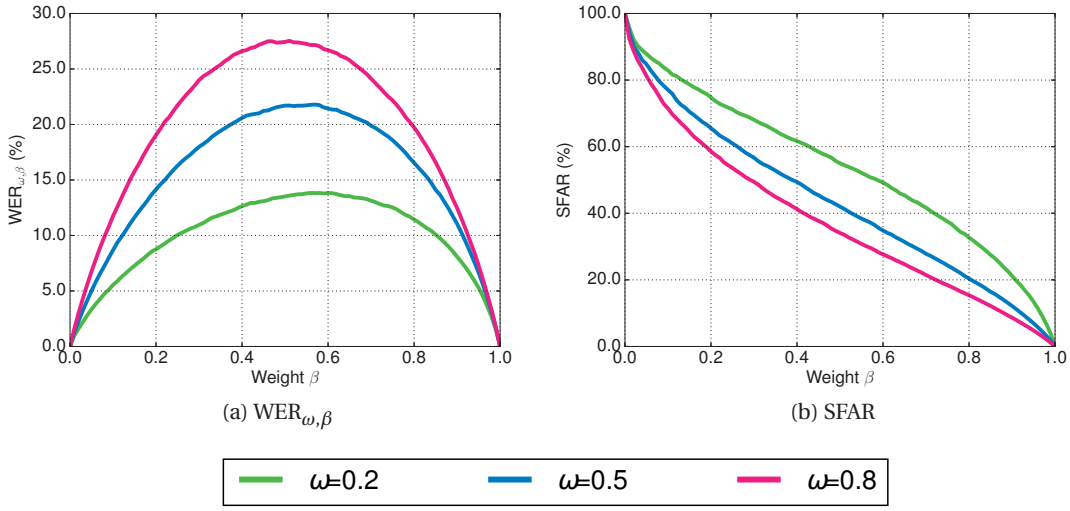
Fig. 5.6a and Fig. 5.6b show $\text{WER}_{\omega,\beta}$ and SFAR with respect to ω for three predefined values of β. The blue curve on Fig. 5.6a, corresponding to $\beta = 0.5$, is equivalent to HTER_ω. The left-most points of the curves correspond to $\omega = 0$, meaning that the decision threshold is obtained disregarding the spoofing attacks as possible input. Hence, the threshold at this point corresponds to the threshold plotted in Fig. 5.3a, calculated for the system when operating in the licit scenario. For the particular hypothetical system and all the three considered values of β, this point corresponds to low $\text{WER}_{\omega,\beta}$, which indicates a system with good verification capabilities, but very high SFAR due to the high overlap of the scores of spoofing attacks and genuine users.

As we increase ω, we give weight to the spoofing attacks so that they have a role in the threshold decision process. In the particular example, this results in a shift of the decision threshold to the right of the score distribution plot in Fig. 5.3a. This decreases the number of spoofing attacks that pass the system, which explains why SFAR decreases with increasing ω. However, the additional caution for the danger of spoofing attacks unavoidably comes with the price



(a) WER$_{\omega,\beta}$ (b) SFAR

Figure 5.6: EPSC of a hypothetical biometric verification system under spoofing attacks, parameterized over ω



(a) WER$_{\omega,\beta}$ (b) SFAR

Figure 5.7: EPSC of a hypothetical biometric verification system under spoofing attacks, parameterized over ω

of more rejected genuine users and thus higher $\text{WER}_{\omega,\beta}$. A system with high robustness to spoofing attacks will show as mild increase of $\text{WER}_{\omega,\beta}$ as possible, with as steep decrease of SFAR as possible.

Fig. 5.7a and Fig. 5.7b show EPSC parameterized over the varying parameter β, for three predefined values of ω. For the extreme cases where $\beta = 0$ and $\beta = 1$, $\text{WER}_{\omega,\beta}$ is 0 because the threshold is determined to minimize the error rate solely associated with the positive or the negative class, respectively. In the case of $\beta = 0$, this results in a successful passing through of all the spoofing attacks.

Considering the spoofing attacks when calculating the decision threshold means taking additional precautions against them. As a result of this, the threshold obtained using EPS framework is better adapted to the input that is expected, contributing to systems with better performance and lower spoofing vulnerability, than systems whose decision threshold has been determined in different way.

The EPSC inherits the advantage of unbiased system comparison from the EPC, because it reports the error rates *a priori*. Since the threshold is always determined using the development set, and the error rates are reported using the test set, one can estimate the expected error rates and spoofability of the system in an unbiased way, on data which has not been seen before. The expected error rates can be reported for a particular value or range of values of the parameters ω and β which are of interest in a particular application. Moreover, EPSC allows for easy and unbiased comparison of verification systems with regards to their performance and robustness to spoofing, simply by comparing the EPSC for the two systems on the same plot. Even more, one can compare verification systems range-wise: which one performs better for a range of values of ω or β.

EPS requires an access to a single score output of the system to be evaluated. Therefore, it is suitable for systems where the verification and anti-spoofing outputs are fused at score level. Systems where the verification and anti-spoofing outputs are fused in a different way can not benefit from EPSC, which displays its greatest limitation.

Finally, if a single number is needed to describe the performance of a system, we define the Area Under EPSC (AUE) metric, which can be computed for a fixed β or ω, as in Eq. 5.8. For example, for a fixed β, it represents the average expected $\text{WER}_{\omega,\beta}$ for all values of ω and is computed using Eq. 5.8. The formula to compute AUE for fixed ω and varying β follows accordingly. Between two systems, better is the one which achieves smaller AUE.

$$\text{AUE} = \int_{\omega\in[0,1]} \text{WER}_{\omega,\beta}(\tau^*_{\omega,\beta}, \mathscr{D}_{test})\,\mathrm{d}\omega \tag{5.8}$$

The AUE can be computed in between certain bounds $a, b \in [0,1]; a < b$, enabling to compare

two systems depending on the required range of the varying parameter.

5.4 EPSC Showcase

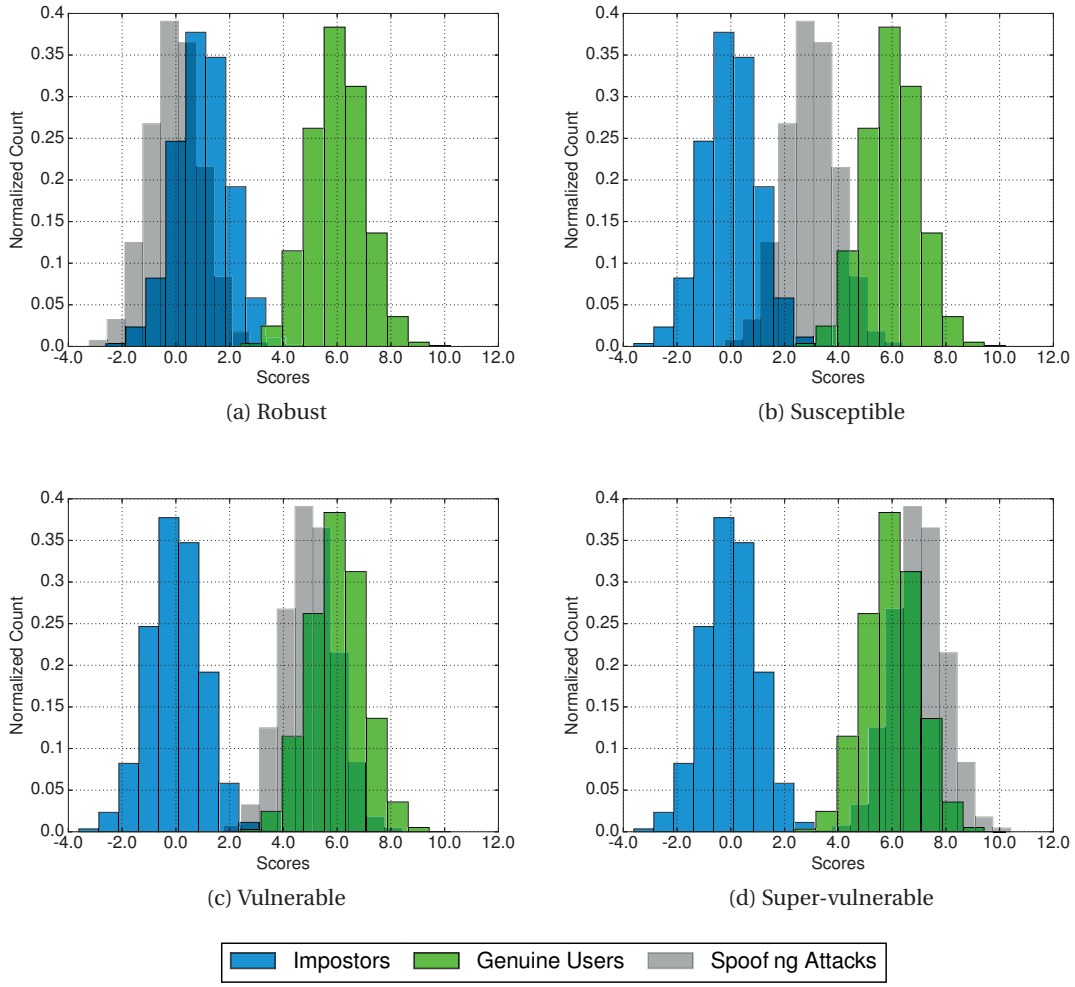
In this section, we illustrate the usage, appearance and interpretation of EPSC for several typical cases of biometric verification systems which behave differently with respect to the spoofing attacks. For this purpose, we categorize the biometric verification systems into four categories, which can be best described by their score distribution plots. As the goal is to illustrate EPSC when the system is under spoofing attacks, for this showcase we assume biometric verification system with good separability between real access and zero-effort impostors, which means good verification performance. The categories, for which score distributions for hypothetical data are given in Fig. 5.8, are as follows:

- Robust (Fig. 5.8a). The score distribution of the spoofing attacks is to the left and not overlapping with the score distribution of the real accesses.
- Susceptible (Fig. 5.8b). The score distribution of the spoofing attacks is to the left and somewhat overlapping with the score distribution of the real accesses.
- Vulnerable (Fig. 5.8c). The score distribution of the spoofing attacks is overlapping with the score distribution of the real accesses.
- Super-vulnerable (Fig. 5.8d). The score distribution of the spoofing attacks is to the right and somewhat overlapping and not overlapping with the score distribution of the real accesses.

The system shown in Fig. 5.8a shows nearly ideal situation, with a great separability between the samples that need to be accepted (real accesses) and the samples that need to be rejected (zero-effort impostors and spoofing attacks). Such systems rarely exist in practice. The system shown in Fig. 5.8b corresponds to a more realistic situation and may be achieved by fusing biometric verification and anti-spoofing system. The system shown in Fig. 5.8c can represent a typical biometric verification system with no explicit anti-spoofing mechanism. Finally, a super-vulnerable system as in Fig. 5.8d is less realistic, as it requires producing spoofing attacks with inconceivable quality.

Fig. 5.9 illustrates the appearance of EPSC for the four categories of biometric verification systems with hypothetical data. The parameter $\beta = 0.5$ is fixed, while the parameter ω varies. It is important to note that for $\omega = 0$, the points on the curves represent the error rates that will be obtained using Evaluation Methodology 2 described in Section 5.2.2.

The general trend for the cases of susceptible, vulnerable and super vulnerable systems is increasing HTER_ω and decreasing SFAR as ω increases. This demonstrates the presence of a trade-off between the robustness to spoofing and overall performance. Exception are highly



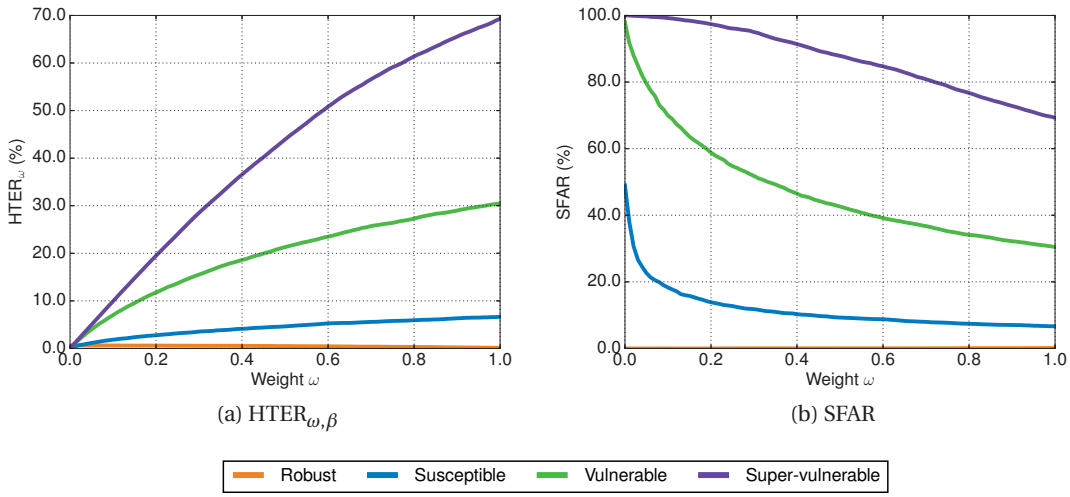
(a) Robust

(b) Susceptible

(c) Vulnerable

(d) Super-vulnerable

Figure 5.8: Score distribution plots for different categories of biometric verification systems



(a) HTER$_{\omega,\beta}$ (b) SFAR

Figure 5.9: EPSC for different categories of hypothetical biometric verification systems

robust systems where using EPSC to determine the decision threshold does not make a big difference with respect to their already very good performance.

In the case of susceptible, vulnerable and super-vulnerable systems, EPSC plays an important role when the probability of spoofing attack, reflected in the parameter ω, is high. If Evaluation Methodology 2 is used as described in Section 5.2.2, then these systems are completely inoperative in cases where spoofing attacks can be expected. However, if EPSC is used, the decision threshold for each ω is optimized for the corresponding probability of spoofing attacks. In this way, the SFAR can be significantly reduced. This comes at the cost of decreased overall performance, in this case represented by HTER$_\omega$. However, this cost depends on the system. For a susceptible system like in Fig. 5.8b, the cost is irrelevant, as the increase of HTER$_\omega$ is mild, while the decrease of SFAR is sharp. For the vulnerable and super-vulnerable systems, the cost is very high, as SFAR is reduced at a slower rate, while HTER$_\omega$ notes a significant increase. This observation reminds that EPSC does not aim at replacing protection with spoofing counter-measures that can bring vulnerable or super-vulnerable systems to the level of susceptible or robust ones. Yet, for any system, it can optimize the decision threshold for the present circumstances.

5.5 Discussion

In the traditional setup, verification and anti-spoofing systems are evaluated independently, using the well-established metrics for binary classification systems. The observation that they need to work in cooperation resulting in a system with a single decision, inevitably brings the necessity for an adjusted evaluation methodology.

We discussed several ways to evaluate verification systems under spoofing attacks. Two of the methodologies that are widely used possess disadvantages that make them either impractical or sub-optimal for deployment.

We proposed a novel evaluation methodology, EPS, which objectively assumes that all the three system inputs, real accesses, zero-effort impostors and spoofing attacks, need to be considered in the threshold decision process. They are taken into account with respect to their prior probability, or the application-dependent cost of the error rates associated with them. In this way, EPS allows for a selection of a decision threshold which is optimal for a given application. Furthermore, EPS is designed to enable unbiased comparison of systems, by setting the decision threshold *a priori*, with no knowledge on the test set in the development phase.

A prerequisite for using EPS framework to optimize the operation of a system is that the system acknowledges three types of inputs: genuine accesses, zero-effort impostors and spoofing attacks. If no such notion is present, the system can still be evaluated using EPS framework, but it can not benefit from it in terms of optimization of the threshold relative to the importance of the different types of inputs. An example of such system is a client-independent anti-spoofing system, which has no notion of zero-effort impostors. Subsequently, decision-level fusion of biometric verification and client-independent anti-spoofing system results in a system with similar problem: the decision of the anti-spoofing system is fixed and can not be optimized using EPS framework, and so is the decision of the fused system. This problem does not exist when fusing biometric verification and client-specific anti-spoofing system at decision-level: the decision threshold of each of the two systems can be optimized separately using EPS framework before the fusion takes place.

The usage of EPS and EPSC was briefly illustrated on hypothetical verification systems. In Chapter 6, it will be used to evaluate and compare realistic verification systems in case studies on the face mode.

It is important to note that a complete API for using EPS framework for evaluation of biometric verification systems under spoofing attacks is available in the open source package `antispoofing.evaluation`[1]. The code assumes no prior information of the biometric mode, internal operation of the system or evaluation database.

[1] https://pypi.python.org/pypi/antispoofing.evaluation

6 Application to Face Verification

Biometric verification systems for the face mode appear to be one of the most attractive targets of spoofing attacks. There are two main factors that contribute to this. The first one has its roots in the wide adoption of online social networks by ever increasing number of users. In recent times, the social networks serve as a main platform where users disclose their personal images, many of which are face images, in large scale. An analysis conducted by Li et al. [2014] shows that the majority of face images shared on the social networks can be readily used to spoof a face verification system simply by displaying them on the screen of a device. This leads us to the second factor which makes spoofing the face mode attractive: the convenience of producing spoofing attacks cheaply and without any expert skills.

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 verification systems and state-of-the-art face anti-spoofing features. The objective is to demonstrate the benefits of the integration of the verification and anti-spoofing systems in creating more trustworthy face biometrics.

With respect to input-level integration, we perform experiments to show the advantages of client-specific over client-independent approaches. For this purpose, we create four case studies based on a subset of the state-of-the-art anti-spoofing features presented in Section 2.3.1. Then, we apply the proposed client-specific modeling approaches and their client-independent counterparts, comparing the results using the evaluation methodology for anti-spoofing systems presented in Section 5.1.

To examine output-level integration, we create case studies where we combine different face verification systems with different anti-spoofing methods. We explore anti-spoofing systems based upon both client-independent and client-specific paradigm. The resulting systems are face verification systems with increased robustness to spoofing attacks. To compare their performance before and after integration, as well as to compare the impact of the different fusion strategies presented in Chapter 4, we use the EPS framework proposed in Section 5.3. Simultaneously, we demonstrate the usage of the EPS framework in practice, indicating its

values for evaluation of biometric verification systems under spoofing attacks.

We emphasize that the code for the methods presented in this thesis is developed and released as free software. The results of the case studies shown in this chapter are fully reproducible using the free software package `bob.thesis.ichingo2015`[1] which accompanies this thesis and which is based on the free signal processing and machine learning toolbox Bob[2] [Anjos et al., 2012].

Detailed description of the anti-spoofing and face verification features and systems, as well as the database used in the case studies is given in Section 6.1. The results regarding the input-level integration are given in Section 6.2, while the results regarding the output-level integration are given in Section 6.3.

6.1 Systems and Database Description

The case studies examined in this chapter are built upon several state-of-the-art face anti-spoofing features and face verification systems. In this section we give a description and justification for their usage. Furthermore, we discuss the choice of a database suitable to apply the case studies on.

6.1.1 Face Anti-spoofing Features

In Section 2.3.1 we gave a review of state-of-the-art features that have been designed to discriminate between real accesses and spoofing attacks. To build the case studies and perform our experiments, we limit ourselves to a subset consisting of four different anti-spoofing features, that we will refer to as LBP, LBP-TOP, MOTION and HOG. The selection of features takes into account several factors, like their discrimination capabilities, implementation details and availability of source code to reproduce their results. Furthermore, the selected features are representatives of different categories, as described in Section 2.3.1.

While the selected face anti-spoofing features have a well-acknowledged value in face anti-spoofing, we are aware that there are methods which achieve better face spoofing detection. Primarily, these methods owe their success to the fusion of several different anti-spoofing methods [Chingovska et al., 2013b] or augmentation of the input [Bharadwaj et al., 2013]. However, the objective of the case studies presented in this chapter is not to outperform superior face anti-spoofing detectors. Instead, they aim at demonstrating the benefits of the concepts proposed in this thesis. In particular, the empirical results show that the advantages of the proposed methods can be achieved using simple features with their basic parameterization. An extensive evaluation of the applicability of other anti-spoofing features in the context of the proposed methods is possible using the free software package `bob.thesis.ichingo2015`.

[1] https://pypi.python.org/pypi/bob.thesis.ichingo2015
[2] http://www.idiap.ch/software/bob

LBP LBP anti-spoofing features belong to the category of features that use visual appearance cues. They are based on the acclaimed Local Binary Pattern (LBP) descriptor, originally proposed for texture classification by Ojala et al. [2002]. It consists of computing a binary pattern over each pixel by comparing its value to the values of pixels in a rectangular or circular neighborhood region with a predefined radius. The obtained binary patterns are then summarized into a histogram over the full image. $\mathrm{LBP}^{type}_{x,y}$ is the usual notation for these features, where x stands for the number of neighboring pixels that are considered, while y is the radius of the circular neighborhood. Depending on the grouping of the binary patterns when creating the histogram, *type* can be regular (r), rotation-invariant (ri) and uniform ($u2$).

As face anti-spoofing feature, LBP was first proposed by Määttä et al. [2011] in a multi-scale version, where several neighborhood regions with different radii are considered. In this work, we use a feature vector created as a histogram of simple $\mathrm{LBP}^{u2}_{8,1}$ [Chingovska et al., 2012]. The features are extracted only from the face region of the input sample, geometrically normalized to 64x64 pixels. The dimension of the final feature vector obtained in this way is 59.

LBP-TOP LBP from Three Orthogonal Planes (LBP-TOP) is a dynamic texture descriptor which extends the computation of LBP features in a temporal dimension [Zhao and Pietikäinen, 2007]. Having video inputs, the binary patterns of a pixel are computed not only in its neighborhoods within its frame, but also within its circular or elliptical neighborhood spanning the preceding and following frames.

LBP-TOP has been used for face spoofing detection by Pereira et al. [2014] and has shown better discrimination capabilities than LBP. In this work, we use $\mathrm{LBP\text{-}TOP}^{u2}_{8,8,8,1,1,1}$, which is extracted over three-dimensional circular neighborhoods with radii 1 and considering 8 neighboring pixels. Similarly to the LBP features, LBP-TOP features are extracted from the face region of the input sample, after a geometric normalization to 64x64 pixels. The obtained feature vector has a dimension of 177.

MOTION The MOTION face anti-spoofing features used in this work are proposed by Anjos and Marcel [2011]. They detect spoofing attacks by estimating the correlation of the movements of the face with regards to the background and rely on the assumption that spoofing attacks may have higher correlation. The algorithm computes a motion coefficient capturing the motion difference between two consecutive frames in a video. The motion coefficient is computed both for the face region, as well as the background. Then, a total of 5 statistical and other measures of the motion coefficient over a window of several consecutive frames are computed: minimum, maximum, mean, standard deviation and the ratio between high and low frequency components. For each frame, the computed measures for the face and background regions are concatenated to create the feature vector, which has a dimension of 10. In this work, we use MOTION features with window size of 20 frames.

HOG Histogram of Oriented Gradients (HOG) has been originally proposed by Dalal and Triggs [2005] as a descriptor for human detection. It requires a computation of the gradient of the pixels in an image. The orientations of the gradients are binned into histograms over overlapping or non-overlapping cells that the image is divided into. Optionally, the cells are grouped into blocks and the cell histograms are normalized withing each block.

HOG has been used for anti-spoofing by Määttä et al. [2012] and Yang et al. [2013]. The HOG features used in this thesis are computed on the face region of the image, geometrically normalized to 64x64. A total of 8 gradient orientations are considered between 0 and 180 degrees. The size of the cells is set to 16 pixels with an overlap of 8 pixels and no block normalization is used. The final feature vector is obtained by concatenating the histograms from the cells and has a dimension of 392.

6.1.2 Face Verification Systems

We base the case studies on four baseline face verification systems which have proven to be state-of-the-art on several face verification databases. To generate the face verification scores, we used their implementation in the free open-source face recognition library Facereclib[3] [Günther et al., 2012].

UBMGMM The first face verification system uses Discrete Cosine Transform (DCT) features extracted from the geometrically normalized faces. The features are extracted over overlapping blocks of 12x12 pixels and after a preprocessing procedure to reduce the impact of illumination variations [Tan and Triggs, 2010]. Face models for the enrolled clients [Cardinaux et al., 2003] are created over the features using a Gaussian Mixture Model (GMM). First, Universal Background Model (UBM) based on a GMM with 512 components is created for a set of background clients. Then, this model is adapted to the face samples of all the enrolled clients using Maximum A Posteriori (MAP) adaptation, creating models for a particular identity. At verification time, an input sample is compared with the model of the identity claimed by the client, as well as the UBM. The final verification score is the log-likelihood ratio between the scores obtained during the two comparisons.

LGBPHS The second face verification system, called Local Gabor Binary Pattern Histogram Sequences (LGBPHS) [Zhang et al., 2005], calculates $\mathrm{LBP}^{u2}_{8,2}$ histograms over input image blocks of size 8x8 and convoluted with 40 Gabor wavelets with 8 orientation and 5 scales [Wiskott et al., 1997]. The concatenated LBP histograms of an input image are compared with the histogram model of the client with the claimed identity using χ^2 measure.

[3] https://pypi.python.org/pypi/facereclib

GJet The third face verification system, referred to as GJet is based on [Wiskott et al., 1997] and extracts Gabor jets from different positions of the image. The Gabor jets are then assembled into a single rectangular grid graph [Günther et al., 2012]. In total, 40 Gabor wavelets with 8 orientation and 5 scales are used. The verification score is generated by comparing the Gabor graphs using the average similarity of the corresponding Gabor jets.

ISV DCT features are used once again in the fourth face verification system, which is based on an Inter-Session Variability Modeling [Wallace et al., 2011]. Similarly to the GMM face verification system, ISV is based on a UBM with 512 components. ISV additionally estimates a 160-dimensional linear subspace of within-class variability. Enrollment of clients is performed by adaptation of the UBM to a specific identity and depending on the within-class variability subspace. The verification scores are obtained as log-likelihood ratio between the scores obtained using the client identity model and the UBM.

6.1.3 Database

The majority of face-spoofing databases provide only real access and spoofing attack samples for the clients and usually lack enrollment data. However, as repeatedly stated in Chapters 3, Chapter 4 and Chapter 5, this is of vital importance at all levels of integration of verification and anti-spoofing systems. To the best of our knowledge, Replay-Attack, together with its subsets Print-Attack and Photo-Attack, is the only face spoofing database that provides enrollment samples. Therefore, all of the case studies were evaluated using Replay-Attack database. The majority of the experiments are performed considering the *Grandtest* protocol of Replay-Attack, which includes all the types of spoofing attacks. However, certain experiments are performed using subprotocols of Replay-Attack, like *Print*, *Digital-Photo* and *Video*, which include only the corresponding types of attacks.

Although the other databases, like NUAA, CASIA-FASD and MSU-MFSD do not provide separate enrollment samples, one can change their protocol and dedicate certain samples for enrollment. However, such a protocol violation will make comparison with previous approaches using the original protocol biased. Furthermore, even if we allow such a violation, for many of the clients it is not possible to select enrollment samples out of the real access data. For NUAA, the samples for the majority of clients come from a single session. Hence, any selection of real access data for enrollment will make the samples for enrollment and evaluation highly correlated. For CASIA-FASD, there are only 3 real access videos per client. Due to their different qualities, selection of any of them for enrollment will bias any method towards a single quality. On the other hand, selecting a part of a video for enrollment will lead to a similar correlation problem as for NUAA database. Containing only 2 real access videos per client with different qualities, MSU-MFSD exhibits similar problems as CASIA-FASD.

6.2 Input-level Integration

In this section, we state our findings with respect to the input-level integration of face verification and anti-spoofing systems. We look at three different case studies, each of which covers one of the face anti-spoofing features described in Section 6.1.1: LBP, LBP-TOP, MOTION and HOG. For each case study, we observe and report the performance of a face anti-spoofing system built with the following approaches: generative client-independent, generative client-specific, discriminative client-independent and discriminative client-specific. The evaluation is done using the evaluation methodologies for anti-spoofing systems, described in Section 5.1.

We would like to emphasize that the evaluation of the input-level integration is to compare client-specific with respect to client-independent approaches only. The comparison has a meaning only if the methods are applied on the same kind of features. Comparison between different features has been done in the studies of Pereira et al. [2014]; Anjos and Marcel [2011]; Yang et al. [2013] and is out of the scope of this work.

The experiments are performed on Replay-Attack database, using its Grandtest, Print, Digital-Photo and Video protocols. For certain experiments, we define two modes of evaluation with respect to the used protocols:

- *Intra-protocol evaluation.* In this evaluation mode, the system is trained and evaluated using the same protocol (or set of protocols).
- *Cross-protocol evaluation.* In this evaluation mode, the system is trained using one protocol (or set of protocols) and is evaluated using another protocol (or set of protocols).

We use intra-protocol evaluation to evaluate the system performance on Grandtest, Print, Digital-Photo and Video protocols, as well as on sets created by pairing the three latter protocols.

Cross-protocol evaluation is important in assessing the capability of an anti-spoofing system to generalize in detecting spoofing attacks which have not been considered during training. As noted by Pereira et al. [2013] and having in mind that the possibilities for inventing novel spoofing attacks are unlimited, robustness to unseen spoofing attacks is a major security asset of anti-spoofing systems. We investigate three cross-protocol evaluation scenarios. In each one of them, the systems are trained using two out of the three considered subprotocols of Replay-Attack, while the third one serves for testing. The scenarios' descriptions are as follows:

- Scenario 1: train with Digital-Photo and Video, test on Print;
- Scenario 2: train with Print and Video, test on Digital-Photo;
- Scenario 3: train with Print and Digital-Photo, test on Video.

According to this, the three scenarios reveal the generalization capabilities of the algorithms when tested on printed photographs, digital photographs and videos, respectively.

We analyze the generative client-specific approach in Section 6.2.1 and the discriminative one in Section 6.2.2. The analysis is performed with respect to different parameters and in comparison with the client-independent approaches, both in intra-protocol and cross-protocol evaluations. Details about the exact parameterization for each of the methods and protocols in our evaluation are given together with the freely available source code in the software package `bob.thesis.ichingo2015`, making the reported results fully reproducible.

6.2.1 Generative Client-Specific Approach

We start the analysis of the generative client-specific approaches by describing the parameter selection. We furthermore examine the effect of the cohort set on the results. Finally, we compare the best setup of client-specific and client-independent approaches on different protocols in intra-protocol and cross-protocol evaluations.

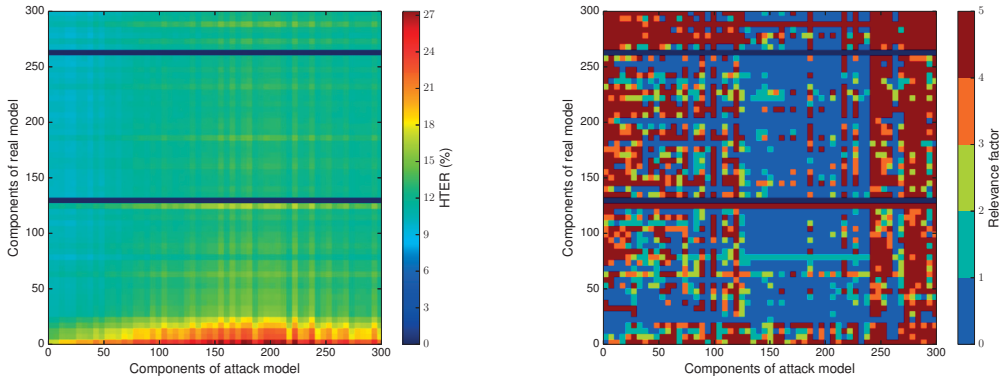
Parameter selection. The described generative client-specific approaches depend on few hyper-parameters, like the number of Gaussian components that comprise the GMM, both for the real access and the spoofing attack model. Another hyper-parameter is the relevance factor which plays a key role in MAP adaptation. All of these parameters have been optimized by a grid parameter search on the development set.

To select the best values for the number of Gaussian components as well as the relevance factor for each of the features, the HTER on the development set was used. The value of HTER for different number of components for the real and attack models is shown in Fig. 6.1a, Fig. 6.2a, Fig. 6.3a and Fig. 6.4a for LBP, LBP-TOP, MOTION and HOG features, respectively. The figures refer to the grandtest protocol only. The dark blue values on the plots correspond to number of components for which grid search results are not available.

We found that LBP, LBP-TOP and HOG features require relatively high number of components for the real access GMMs, ranging between 240 - 290 depending on the protocol. In particular, for the grandtest protocol, the optimal number of Gaussian components for the LBP, LBP-TOP and HOG features is 275, 295 and 295, respectively. The MOTION features, on the other hand, can successfully model the client-specific real accesses with only 10-50 components. For the grandtest protocol, the number of components is 10. The spoofing attack GMMs consist of smaller number of components: below 100 for the majority of protocols for each of the features. For the grandtest protocol, the optimal number of components is 25, 100, 45 and 55 for LBP, LBP-TOP, MOTION and HOG features, respectively.

Fig. 6.1b, Fig. 6.2b, Fig. 6.3b and Fig. 6.4b show the optimal value of the relevance factor for each combinantion of number of components of the real and the attack model. The figures exemplify the grandtest protocol only. For most of the protocols and features best results

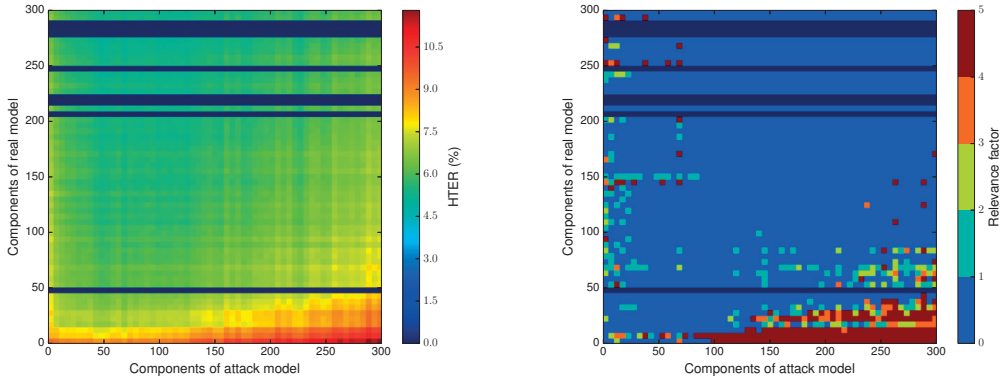
are achieved with low values of the relevance factor ($1 < r < 5$), which means that the MAP adaptation of the GMMs gives more importance to the client-specific data than to the prior. Among those, values close to 1 are more suitable for LBP-TOP features, as well asfor LBP and HOG features when the number of components of the attack model is in the middle of the high range. For the MOTION features, values close to 5 are more suitable. This is an indicator that the LBP-TOP, LBP and HOG features contain higher amount of client-specific information, compared to the MOTION features.



(a) HTER (%)

(b) Relevance factor

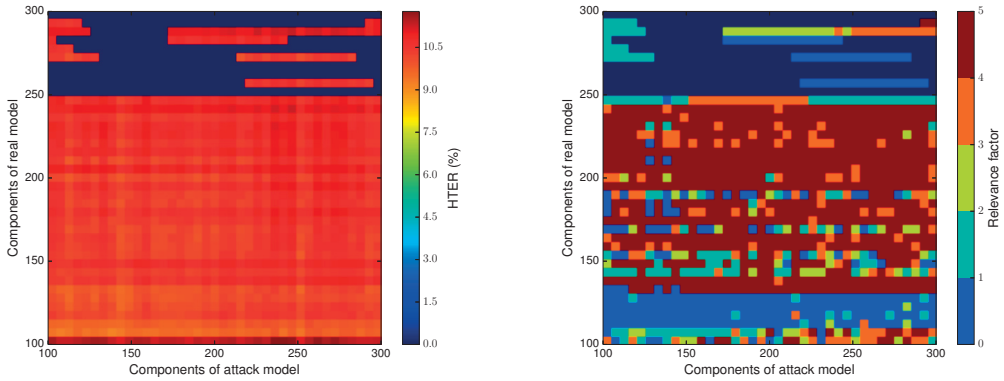
Figure 6.1: **LBP features:** Values of HTER (%) and relevance factor depending on the number of components for the real and attack GMM models



(a) HTER (%)

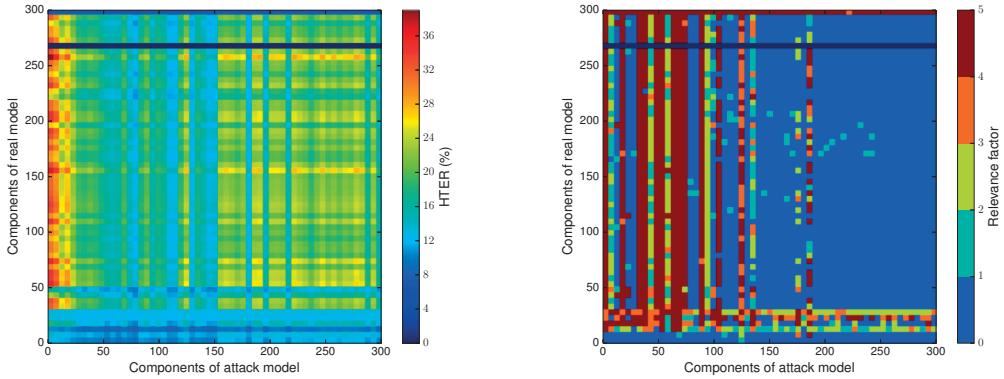
(b) Relevance factor

Figure 6.2: **LBP-TOP features:** Values of HTER (%) and relevance factor depending on the number of components for the real and attack GMM models



(a) HTER (%) (b) Relevance factor

Figure 6.3: **MOTION features:** Values of HTER (%) and relevance factor depending on the number of components for the real and attack GMM models



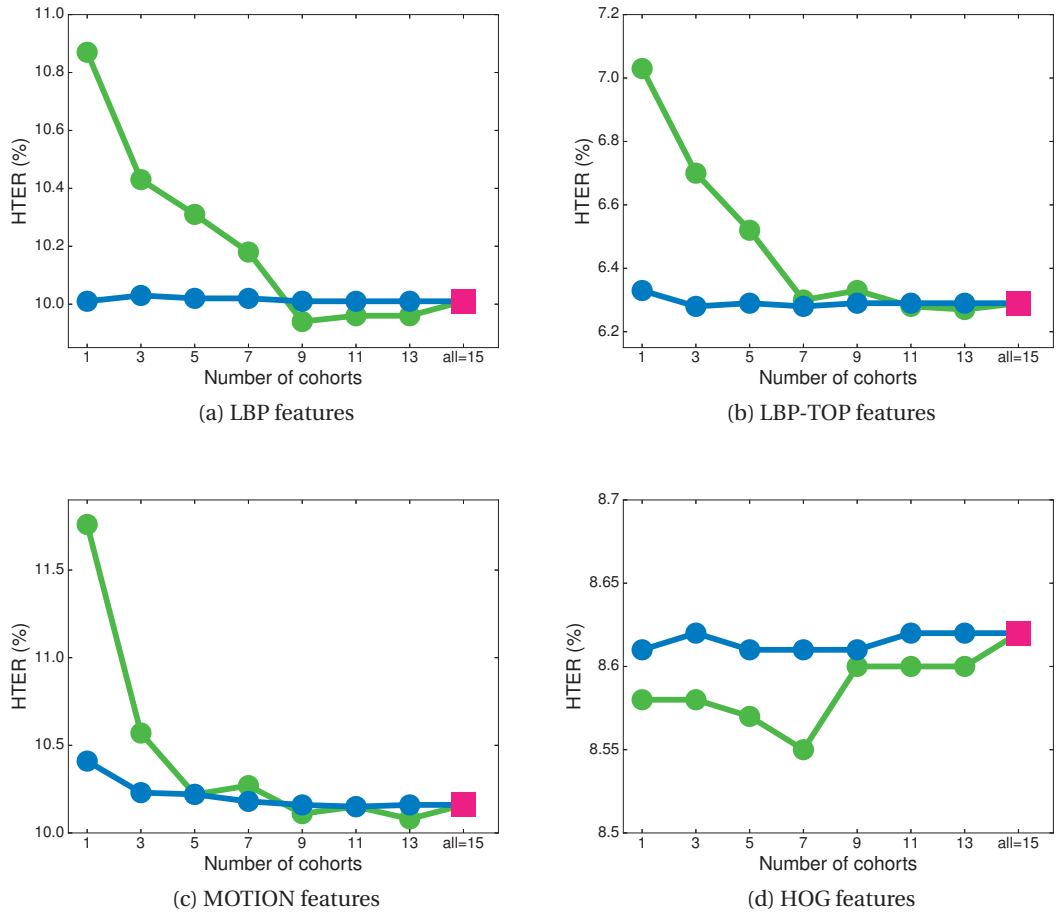
(a) HTER (%) (b) Relevance factor

Figure 6.4: **HOG features:** Values of HTER (%) and relevance factor depending on the number of components for the real and attack GMM models

Effect of the cohort set. The analysis of the cohort set selection is performed on the Grandtest protocol of Replay-Attack. A full cohort set to model the hypothesis $\mathscr{H}_1$ as explained in Section 3.2.3, consists of 15 cohorts, which is the maximum number of clients in the training set of Replay-Attack.

Fig. 6.5 presents the dependence on the performance on the number of cohort models that are considered in the static or dynamic client-specific cohort set. The performance is given in terms of HTER as a mean between the error rates related to misclassified real accesses and spoofing attacks. For the static client-specific cohort set, the general trend suggests that, larger

number of cohorts yields better performance for the majority of the considered features. For example, considering only the closest cohort in the cohort set gives a relativly high HTER for LBP, LBP-TOP and MOTION features. As the number of cohorts increases, HTER decreases. In particular cases, HTER for large enough cohort set (9-13 cohorts) is smaller then HTER considering the full cohort set.



(a) LBP features

(b) LBP-TOP features

(c) MOTION features

(d) HOG features

Figure 6.5: Dependence on the selection of the cohort models. ■: full cohort set, ●: static client-specific cohort set, ●: dynamic client-specific cohort set

On the other hand, the number of cohorts in the dynamic client-specific cohort set appears to make very little difference, as HTER stays relatively stable for all considered features. Furthermore, the dynamic client-specific cohort set yields better performance than the static one for LBP, LBP-TOP and MOTION features and with small number of cohorts . This suggest that the dynamic selection of the cohort sets achieves better sorting of the cohort models in terms of relevance for the client-specific models. HOG features are an exception to this rule, and static client-specific cohort set performs better even than a full cohort set and for smaller number of cohorts.

It is important to note that although HTER varies depending on the cohort set, in absolute terms the variations are minor and within a range of 2%. For example, for LBP features, the minimum HTER is 9.94%, while the maximum is 11%. The HTER when using full cohort set is 10.01%. Therefore, in the subsequent experiments, we will adhere to using the full cohort set.

Intra-protocol evaluation. Using intra-protocol evaluation, we compare generative client-independent and client-specific anti-spoofing methods. Similarly to the client-specific approach, the parameters for the client-independent one have been optimized using grid parameter search.

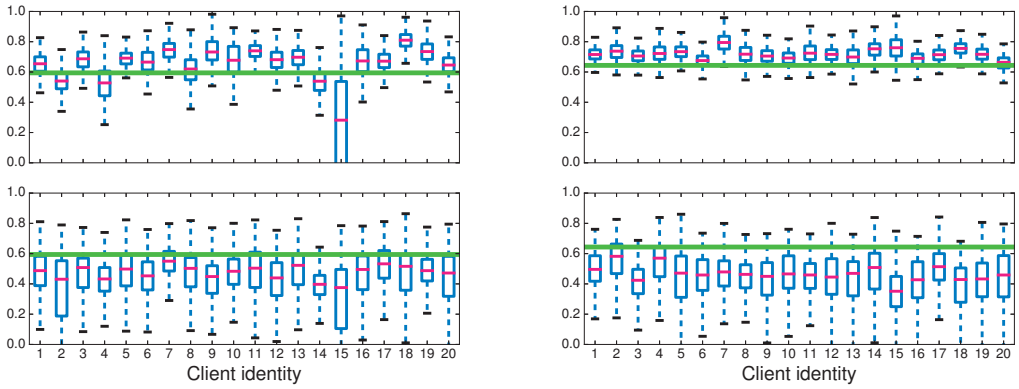
The results for the Grandtest protocol are given in Table 6.1. The generative client-specific approach consistently outperforms the the client-independent one for all the types of features, both on the development and test set. The advantage of the client-specific features is significant in almost all the cases: the relative improvement of HTER is above 50% for all features, except MOTION.

Table 6.1: Performance of **generative** client-independent and client-specific approaches on Grandtest protocol (error rates in %)

	Client-independent				**Client-specific**			
	dev	**test**			**dev**	**test**		
Features	**EER**	FAR	FRR	**HTER**	**EER**	FAR	FRR	**HTER**
LBP	21.33	17.93	25.45	**21.69**	9.97	8.6	11.42	**10.01**
LBP-TOP	9.32	8.38	16.92	**12.65**	5.08	5.07	7.51	**6.29**
MOTION	12.49	13.91	11.14	**12.52**	10.25	10.81	9.51	**10.16**
HOG	16.22	17.79	17.10	**17.44**	6.59	7.04	10.20	**8.62**

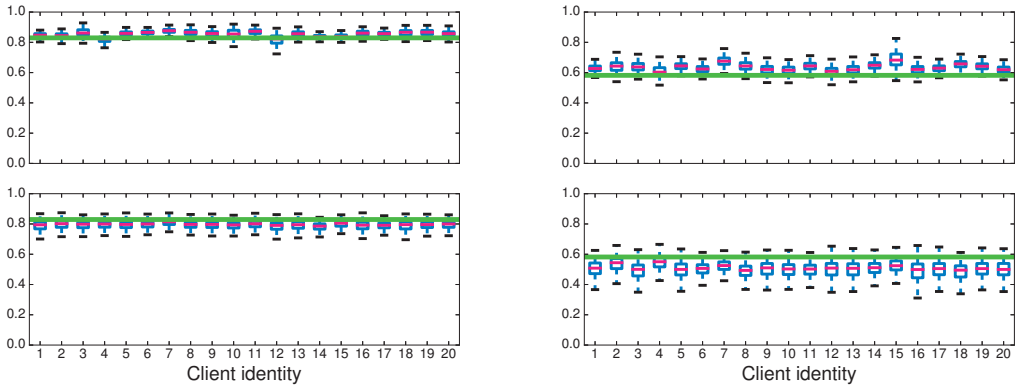
To understand the improvement of anti-spoofing performance when client-specific approach is used, we compare the box plots of the client scores for the client-specific and client-independent approaches. For different features, they are given in Fig. 6.6, Fig. 6.7, Fig. 6.8 and Fig. 6.9, where real access scores are shown in the upper plots and spoofing attacks in the lower ones. As explained in Section 3.1, the central bar of each box is the median of the client scores. Its upper and lower edges represent the 75th and the 25th percentile of the scores, respectively, while the whiskers extend to the most extreme non-outlier score values. The plots also include the decision threshold determined using EER criteria on the development set. On each of the figures, the upper plots show the scores of real access samples while the lower plots show the scores of spoofing attacks.

The box plots for the client-independent approaches demonstrate a high variability of the scores of different clients, as already shown in Section 3.1 and Fig. 3.3. Certain clients exhibit a score distribution which are significantly misaligned compared to the score distributions of the majority of clients, especially for the real access scores for the LBP and HOG features. This suggests the existence of client-specific score variations for the client-independent baseline, which makes a single decision threshold for all clients and samples a suboptimal choice. While



(a) Client-independent (b) Client-specific

Figure 6.6: Box plots of the scores obtained with **generative** anti-spoofing methods: **LBP features**. Upper plots: scores of real access samples; lower plots: scores of spoofing attacks. The horizontal green line depicts the decision threshold on the development set.

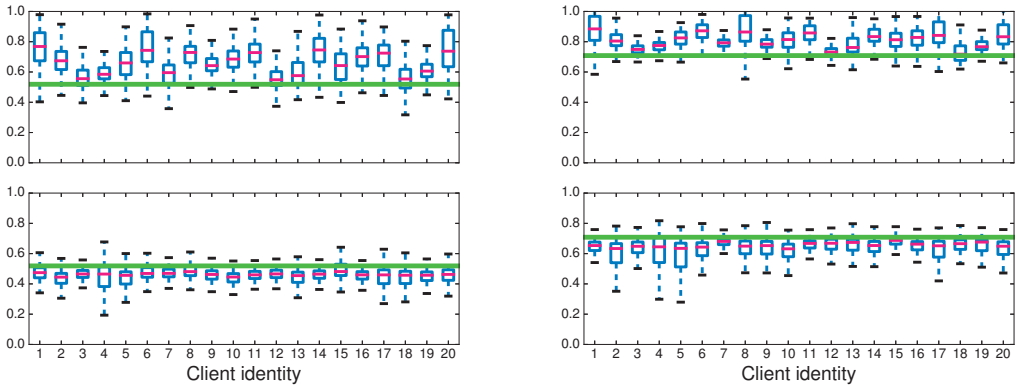


(a) Client-independent (b) Client-specific

Figure 6.7: Box plots of the scores obtained with **generative** anti-spoofing methods: **LBP-TOP features**. Upper plots: scores of real access samples; lower plots: scores of spoofing attacks. The horizontal green line depicts the decision threshold on the development set.

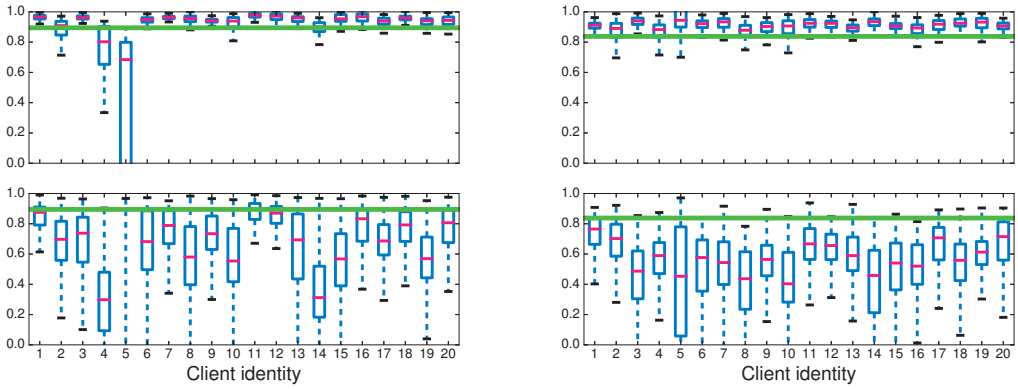
the misalignment of the score distributions are not that prominent for the LBP-TOP features, the separability between the real access and spoofing attack scores is poor.

On the contrary, the client-specific anti-spoofing methods show a much better alignment of the score distributions both for the real access and spoofing attack samples. Besides having similar values across all the clients, the medians of box plots are often further away from the decision threshold, like for example, for the LBP-TOP features. This justifies the employment of a single decision threshold for all clients in client-specific anti-spoofing methods.



(a) Client-independent (b) Client-specific

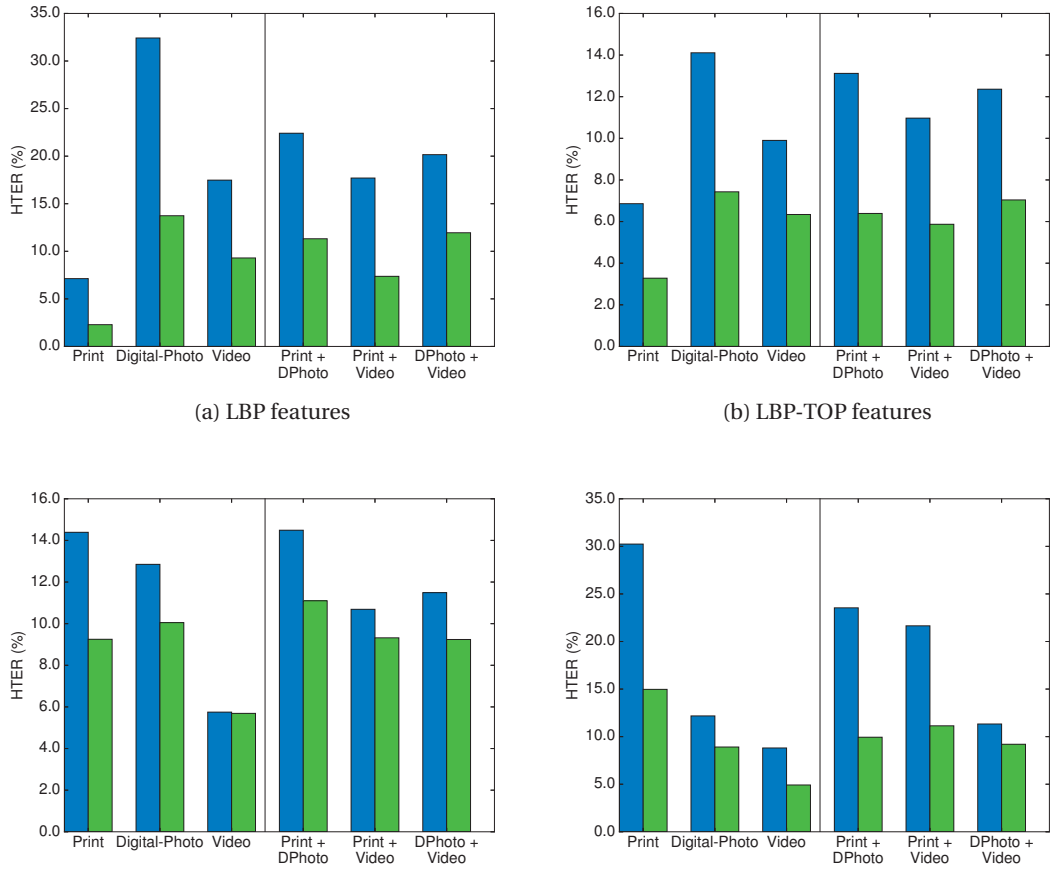
Figure 6.8: Box plots of the scores obtained with **generative** anti-spoofing methods: **MOTION features**. Upper plots: scores of real access samples; lower plots: scores of spoofing attacks. The horizontal green line depicts the decision threshold on the development set.



(a) Client-independent (b) Client-specific

Figure 6.9: Box plots of the scores obtained with **generative** anti-spoofing methods: **HOG features**. Upper plots: scores of real access samples; lower plots: scores of spoofing attacks. The horizontal green line depicts the decision threshold on the development set.

We extend the intra-protocol evaluation by performing experiments using Print, Digital-Photo and Video protocols, as well as pairwise combinations of these protocols. The results of the comparison of the client-independent and client-specific approaches under these protocols are given in terms of HTER on the test set on Fig. 6.10. For all the types of features, they are in favor of the client-specific approach in all the protocols. Furthermore, the client-specific approach achieves a relative advantage of over 50% on several protocols with LBP, LBP-TOP and HOG features.



(a) LBP features

(b) LBP-TOP features

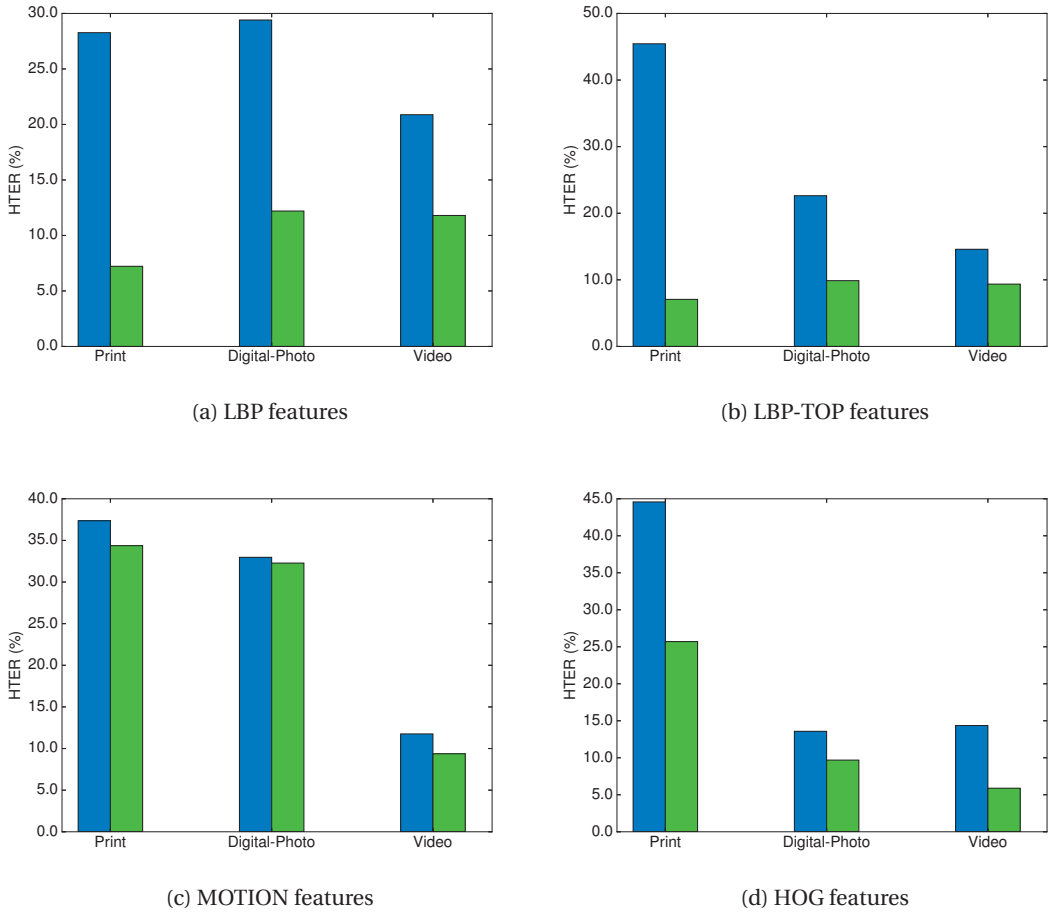
(c) MOTION features

(d) HOG features

Figure 6.10: Intra-protocol evaluation of **generative** anti-spoofing systems. ■: client-independent approach, ■: client-specific approach.

Cross-protocol evaluation. The results of the cross-protocol evaluation of the generative client-independent and client-specific systems are given in Fig. 6.11. They are given in terms of HTER on the test set on the protocol which has not been used during training. The bin labels correspond to the protocol used for evaluation. For example, the first two bins on each plot present the results of a system trained with Digital-Photo and Video protocol and evaluated on the Print protocol.

Similarly to the results of the intra-protocol evaluation, the client-specific approach outperform the client-independent one by a large margin. A relative advantage of ~50% is common for LBP, LBP-TOP and HOG features. However, the MOTION features behave differently and the client-specific approach performs only moderately better than the client-independent one. This may suggest that MOTION features capture less client-specific information than the other features.



(a) LBP features

(b) LBP-TOP features

(c) MOTION features

(d) HOG features

Figure 6.11: Cross-protocol evaluation of **generative** anti-spoofing systems. ■: client-independent approach, ■: client-specific approach.

6.2.2 Discriminative Client-Specific Approach

For the analysis of the discriminative client-specific approaches, we first discuss the parameter selection, as well as the choice of the SVM kernel. Then, we perform a comparison of client-specific and client-independent approaches in their best setup, on different protocols in intra- and cross-protocol evaluations.

Parameter and SVM kernel selection. The key hyper-parameter for any SVM is the constant C regularizing the boundary between the two classes. The results of the performed grid parameter search show that for the client-specific anti-spoofing systems, this parameter does not play a significant role and the results change only marginally with the change of C. Therefore, in all of the experiments, $C = 1$. An exception are the HOG features, where for certain protocols, best results are achieved with larger values of C ($C = 100$).

We examined SVM with different kernels: RBF, polynomial, χ^2 and histogram intersection kernel. For the RBF kernel, the parameter γ has the biggest impact and values which are the inverse of the number of samples give the best results. Yet, other values of γ give similar results. For the polynomial kernel, the most important parameter is the degree d. Best results are achieved with $d = 3$ for LBP, $d = 5$ for LBP-TOP and $d = 4$ for MOTION and HOG. Table 6.2 gives the results of the discriminative client-specific approach using different types of kernels and their optimal parameters.

Table 6.2: Comparison of different SVM kernels for discriminative client-specific approach on Grandtest protocol (HTER in %)

	RBF		**Polynomial**		**Chi-2**		**Intersection**	
Features	**dev**	**test**	**dev**	**test**	**dev**	**test**	**dev**	**test**
LBP	**10.02**	**9.87**	14.22	13.79	16.75	14.96	12.39	11.44
LBP-TOP	3.71	3.95	4.88	5.05	3.61	3.60	**2.83**	**2.85**
MOTION	10.18	11.27	9.29	**9.27**	**9.16**	9.99	10.24	10.35
HOG	**5.99**	**6.83**	14.83	23.19	25.93	30.03	11.4	10.6

Several of the existing implementation of SVM process χ^2 and histogram intersection kernels by prior computing of the full kernel matrix K[4]. Depending on the number of training samples, this operation can require an extensive amount of memory. This limitation can be overcome using kernel approximations, like the Nyström method [Williams and Seeger, 2001]. Instead of computing inner products in high-dimensional space, the kernel approximations learn an explicit feature map to project the features in that space. The projected features can then be trained using linear SVM. The Nyström method enables to compute the feature map using just a random subset of the available training samples. It is important to note that the results in Table 6.2 are obtained using a subset of 1000 training samples for χ^2 and histogram intersection kernel. Better results may be expected if more samples are used.

According to Table 6.2, the best results for LBP and HOG features are obtained using RBF kernel. Although both of these features have a histogram form, the kernels designated for histogram feature spaces, like χ^2 and histogram intersection kernel, exhibit inferior performance. We emphasize once again that this might be due to the random sub-sampling of the training set. On the other hand, the approximated χ^2 and histogram intersection kernels perform better for LBP-TOP features, despite the reduced training set. Among them, histogram intersection kernel performs better. Finally, for MOTION features, polynomial kernel performs the best on the test set, but if we consider the development set to select the best performing method, we should choose χ^2 kernel.

Intra-protocol evaluation. A comparison between discriminative client-independent and client-specific methods is first done in an intra-protocol evaluation. The given results are

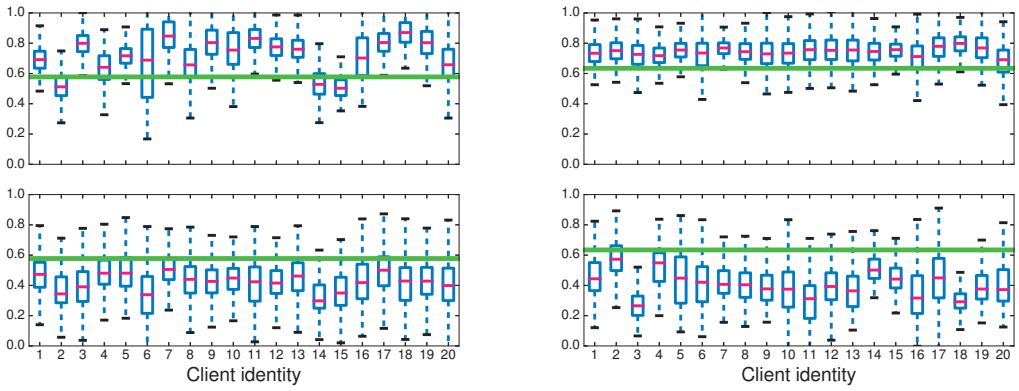
[4] In particular, the software package `scikit-learn` (https://scikit-learn.org) was used for this purpose [Pedregosa et al., 2011]

obtained using RBF kernel for LBP and HOG features, histogram intersection kernel for LBP-TOP features and χ^2 kernel for MOTION features. The optimal parameters for each of the kernels are found with grid parameter search.

Table 6.3 shows the results for the Grandtest protocol. The results confirm that, in the discriminative case, the client-specific method significantly outperforms the client-independent one, both on the development and test set. For LBP, LBP-TOP and HOG features, the relative improvement is particularly high: 36%, 49% and 29%, respectively. Although to a lesser extent, the client-specific method performs better also for MOTION features.

Table 6.3: Performance of **discriminative** client-independent and client-specific approaches on Grandtest protocol (error rates in %)

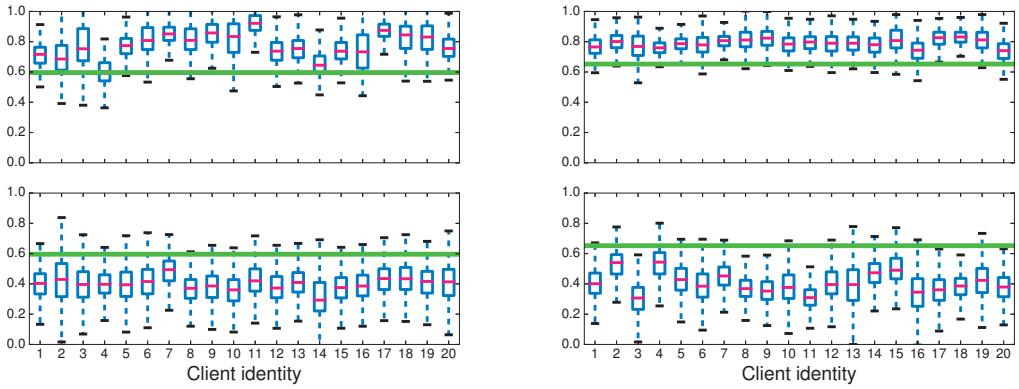
	Client-independent				**Client-specific**			
	dev	**test**			**dev**	**test**		
Features	**EER**	FAR	FRR	**HTER**	**EER**	FAR	FRR	**HTER**
LBP	14.56	9.56	21.29	**15.42**	10.02	8.18	11.53	**9.86**
LBP-TOP	5.55	4.01	7.28	**5.64**	2.83	1.76	3.94	**2.85**
MOTION	9.8	12.08	8.67	**10.38**	9.16	10.11	9.87	**9.99**
HOG	7.69	8.11	11.29	**9.7**	5.99	5.52	8.14	**6.83**



(a) Client-independent (b) Client-specific

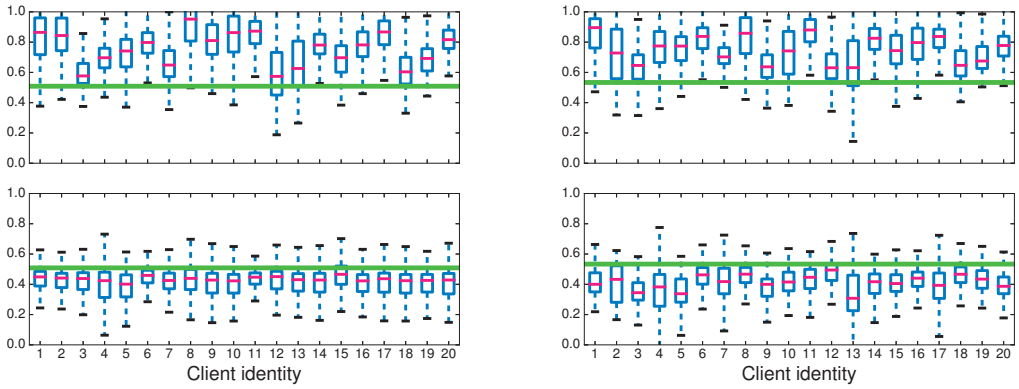
Figure 6.12: Box plots of the scores obtained with **discriminative** anti-spoofing methods: **LBP features**. The horizontal green line depicts the decision threshold on the development set.

The box plots comparing the client scores of the discriminative approaches are given in Fig. 6.12, Fig. 6.13, Fig. 6.14 and Fig. 6.15. Similar values of the medians and uniform box sizes for all clients on Fig. 6.12b, Fig. 6.13b and Fig. 6.15b justify the advantage of the client-specific approach for LBP, LBP-TOP and HOG features. This is especially true for the real access scores. Greater separability of the two classes and less extended whiskers can be observed as well. Fig. 6.14 reveals no significant improvement of the box plot layout for the client-specific approaches for the MOTION features.



(a) Client-independent (b) Client-specific

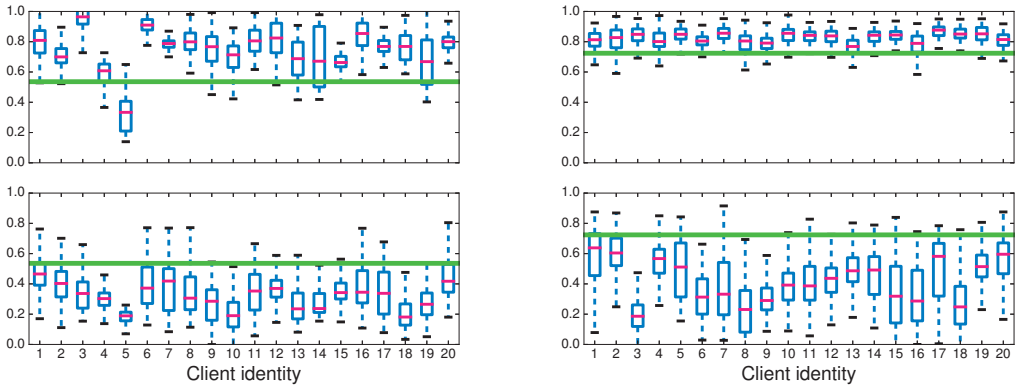
Figure 6.13: Box plots of the scores obtained with **discriminative** anti-spoofing methods: **LBP-TOP features**. The horizontal green line depicts the decision threshold on the development set.



(a) Client-independent (b) Client-specific

Figure 6.14: Box plots of the scores obtained with **discriminative** anti-spoofing methods: **MOTION features**. The horizontal green line depicts the decision threshold on the development set.

The advantage of the client-specific discriminative approach is further indicated on Fig. 6.16, illustrating the results of the intra-protocol experiments on the Print, Digital-Photo and Video protocols, as well as on their pairwise combinations. It is notably significant for LBP, LBP-TOP and HOG features, where the relative improvement is ~ 50% for many of the protocols, and reaches up to 83% and 88% for the Print protocol for LBP and LBP-TOP, respectively. Interestingly, the MOTION features present the only case where the client-independent and client-specific approaches perform on similar scale on few of the protocols, with the client-



(a) Client-independent (b) Client-specific

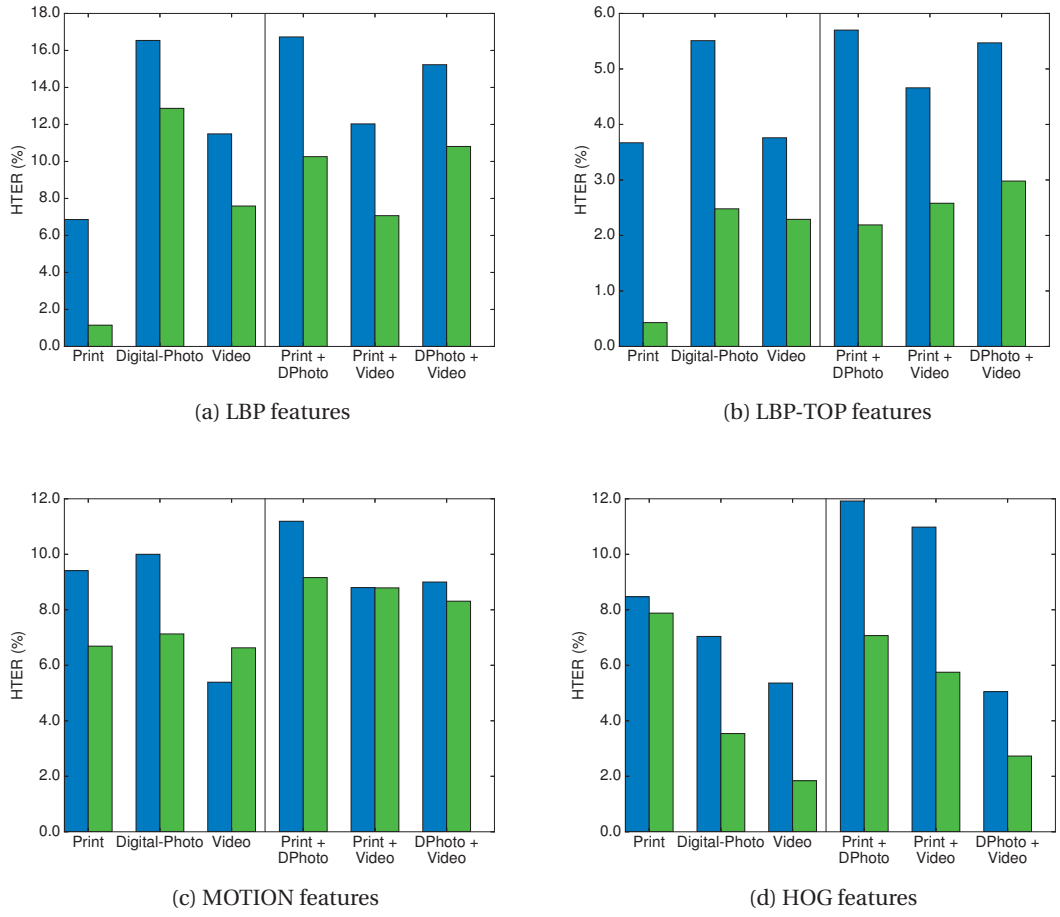
Figure 6.15: Box plots of the scores obtained with **discriminative** anti-spoofing methods: **HOG features**. The horizontal green line depicts the decision threshold on the development set.

independent one being better for the Video protocol. It is important to emphasize that MOTION features behave similarly in the case of the generative approaches as well, indicating less client-specific information retained in these features.

Cross-protocol evaluation. The results of the cross-protocol evaluation of the discriminative anti-spoofing systems is given in Fig. 6.17. Similarly as for the intra-protocol evaluation, RBF kernel is used for LBP and HOG features, while histogram intersection and χ^2 kernels is used for the LBP-TOP and MOTION features, respectively.

Consistently to what is shown by Pereira et al. [2013], for most of the protocols the discriminative client-independent approach exhibits unacceptable spoofing vulnerability when confronted with spoofing attacks that has not been seen during training. For example, when the system is trained with Digital-Photo and Video protocol, it misses the detection of over 35% of the spoofing attacks of the Print protocol for all the features. Although slightly better, the performance is still unsatisfactory for other cross-protocol evaluation scenarios.

The client-specific discriminative anti-spoofing method overcomes these problems. HTER of the attacks that have not been used during training is reduced to 5-10% on LBP, ~ 5% on LBP-TOP features and 5-17% on HOG features. Significant improvement is noted for the MOTION protocol as well, except for the scenario where the system is trained using Print and Digital-Photo protocols and tested using Video protocol. This might be due to the fact that from the point of view of the MOTION features, there is practically no difference between the motion patterns of a live client and a re-captured video of the same client.



(a) LBP features

(b) LBP-TOP features

(c) MOTION features

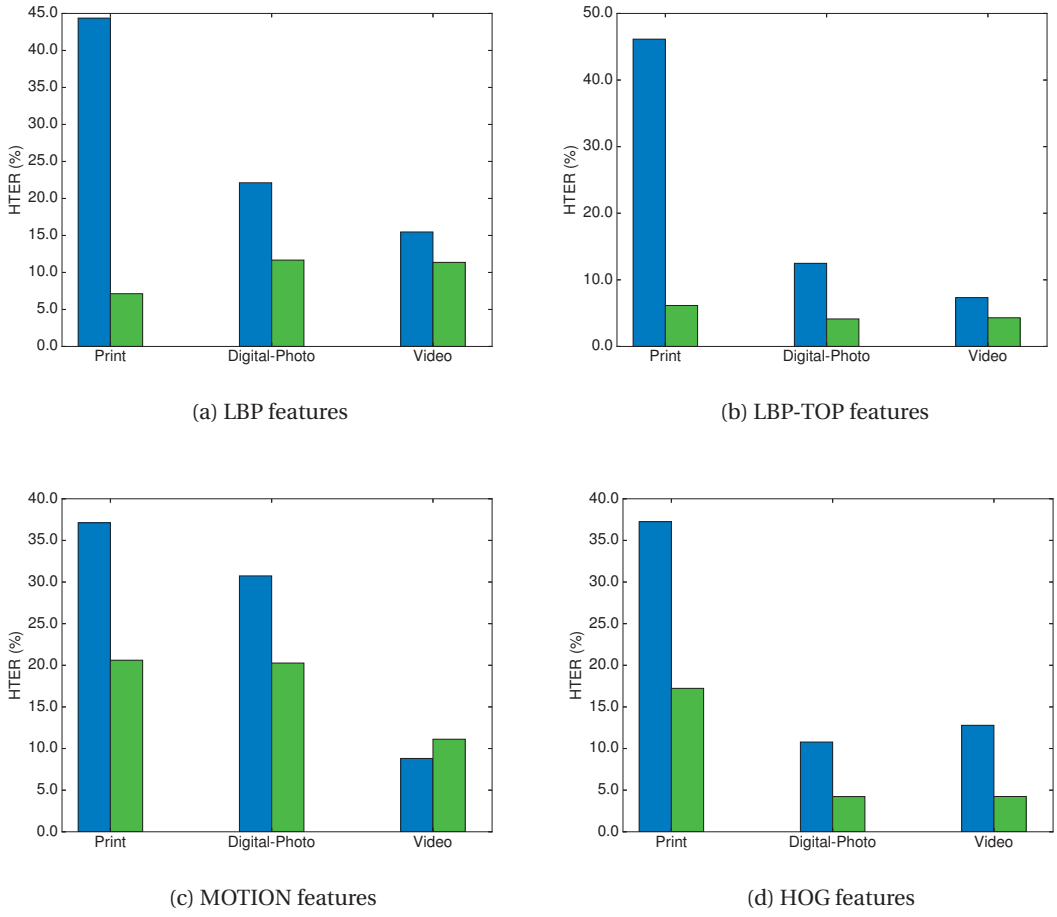
(d) HOG features

Figure 6.16: Intra-protocol evaluation of **discriminative** anti-spoofing systems. ■: client-independent approach, ■: client-specific approach.

6.2.3 Summary

The experimental evaluation shows that the client-specific approaches greatly improve the performance of their client-independent counterparts, both for the generative and discriminative approaches. Depending on the features and the protocol, the relative improvement goes to more than 50%. More importantly, the client-specific approaches significantly outperform the client-independent ones in cross-protocol evaluation too. A explanation may be the fact that, as the box plots show, the client-specific approaches generate real access scores per client which are more aligned and better separated from the spoofing attacks.

The client-specific approaches bring bigger improvement for features which are based on visual appearance, like LBP, LBP-TOP and HOG. The MOTION features appear to be more client-independent.



(a) LBP features

(b) LBP-TOP features

(c) MOTION features

(d) HOG features

Figure 6.17: Cross-protocol evaluation of **discriminative** anti-spoofing systems. ■: client-independent approach, ■: client-specific approach.

To observe whether generative or discriminative client-specific approach performs better, we compare their HTER on the test set of the Grandtest protocol in Table 6.4. For all the studied features, the discriminative approach appears to be more powerful. The advantage is more prominent for LBP-TOP and HOG features.

Table 6.4: Comparison of client-specific generative and discriminative approaches (HTER in %)

Features	Generative	Discriminative
LBP	10.01	**9.86**
LGB-TOP	6.29	**2.85**
MOTION	10.16	**9.99**
HOG	8.62	**6.83**

6.3 Output-level Integration

In this section, we present the results of the output-level integration of face verification and anti-spoofing systems, using the decision-level and score-level fusion methods described in Section 4.2. The case studies used to evaluate the output-level integration include combinations of four face verification systems (UBMGMM, LGBPHS, GJet and ISV) described in Section 6.1.2 and four face anti-spoofing features (LBP, LBP-TOP, MOTION and HOG) described in Section 6.1.1. The case studies are applied on Replay-Attack database, using its Grandtest protocol.

The presented evaluation has two major objectives. The first one is to show how output-level integration with an anti-spoofing system affects the performance of a biometric verification system. The second one is to compare different strategies for fusion of biometric verification and anti-spoofing systems, both in terms of verification performance and vulnerability to spoofing of the final fused system. In addition, the experiments include comparison of fused systems with respect to whether the anti-spoofing system is client-independent or client-specific.

The evaluation is performed using evaluation methodologies described in Section 5.2 and Section 5.3, in particular Evaluation Methodology 2 and EPS framework. Using the two evaluation methodologies side by side, we aim to demonstrate the advantage of EPS and emphasize its fitness for evaluation of biometric verification systems under spoofing attacks. Therefore, throughout some of the experiments, the comparison of the two evaluation methodologies will be stated and intermixed with the evaluation results.

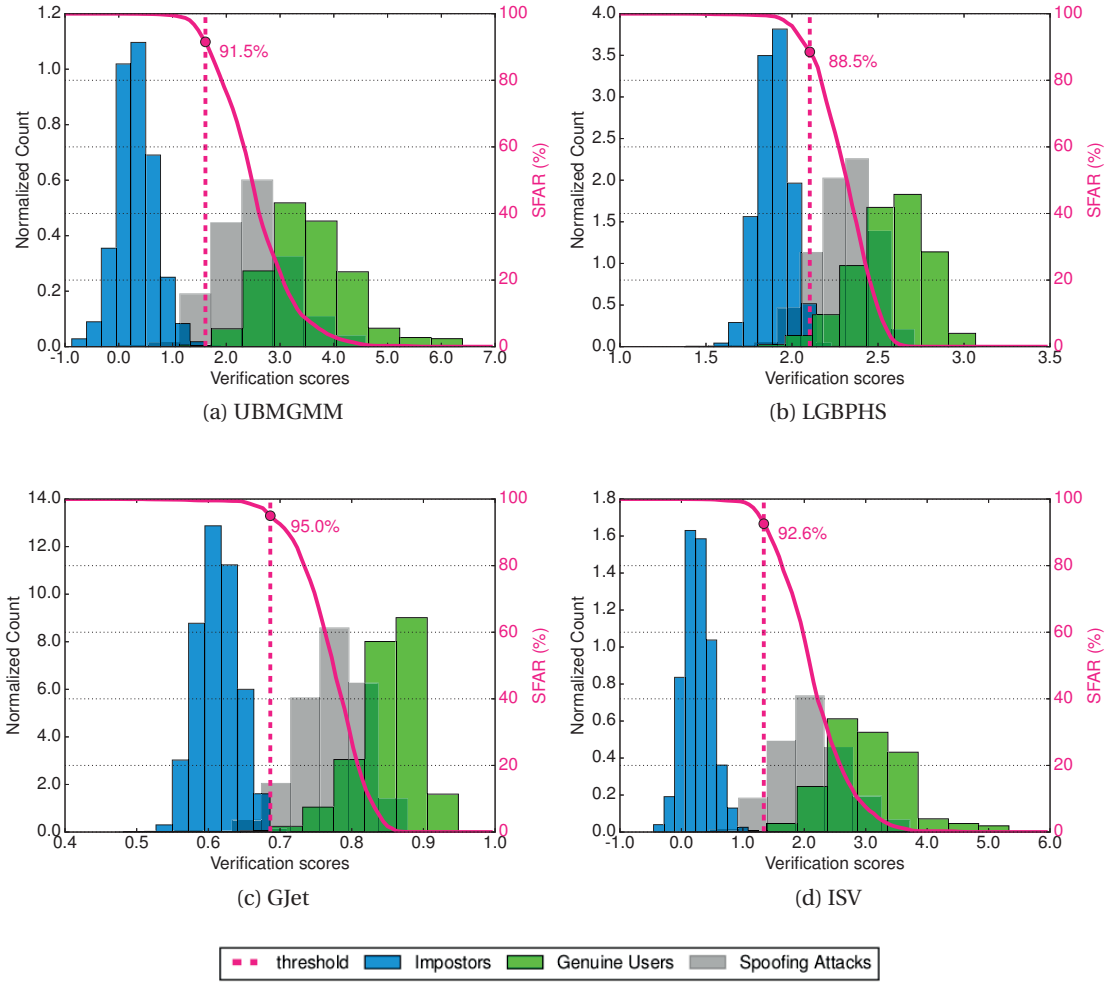
We commence the analysis by stating the performance of the baseline face verification systems in Section 6.3.1 prior to fusion with an anti-spoofing system. Fused systems are compared in 6.3.2. We would like to emphasize that, due to the large number of case studies that are possible to create by combining four face verification systems, four anti-spoofing systems and five fusion methods, only partial analysis covering a subset of the case studies is given. However, by using the freely available software package `bob.thesis.ichingo2015`, one can reproduce all the results given in this section and extend the analysis to other case studies.

6.3.1 Performance of Baseline Face Verification Systems

Before analyzing the effect of fusion of face verification and anti-spoofing, we give the performance of four baseline face verification systems, using both Evaluation Methodology 2 described in Section 5.2 and EPS framework proposed in Section 5.3. As part of the analysis, we compare the conclusions delivered by the two evaluation methodologies.

The score distributions of the four systems are given in Fig. 6.18. The plots given in Fig. 6.19 represent the DET curves for the licit and the spoof scenario for each of the systems. When using the licit scenario, we obtain a DET curve showing the trade-off between FAR and FRR

when no spoofing attacks are present. On the other hand, when using the spoof scenario, we can plot an additional DET curve showing the trade-off between SFAR and FRR and ignoring the existence of zero-effort impostors.



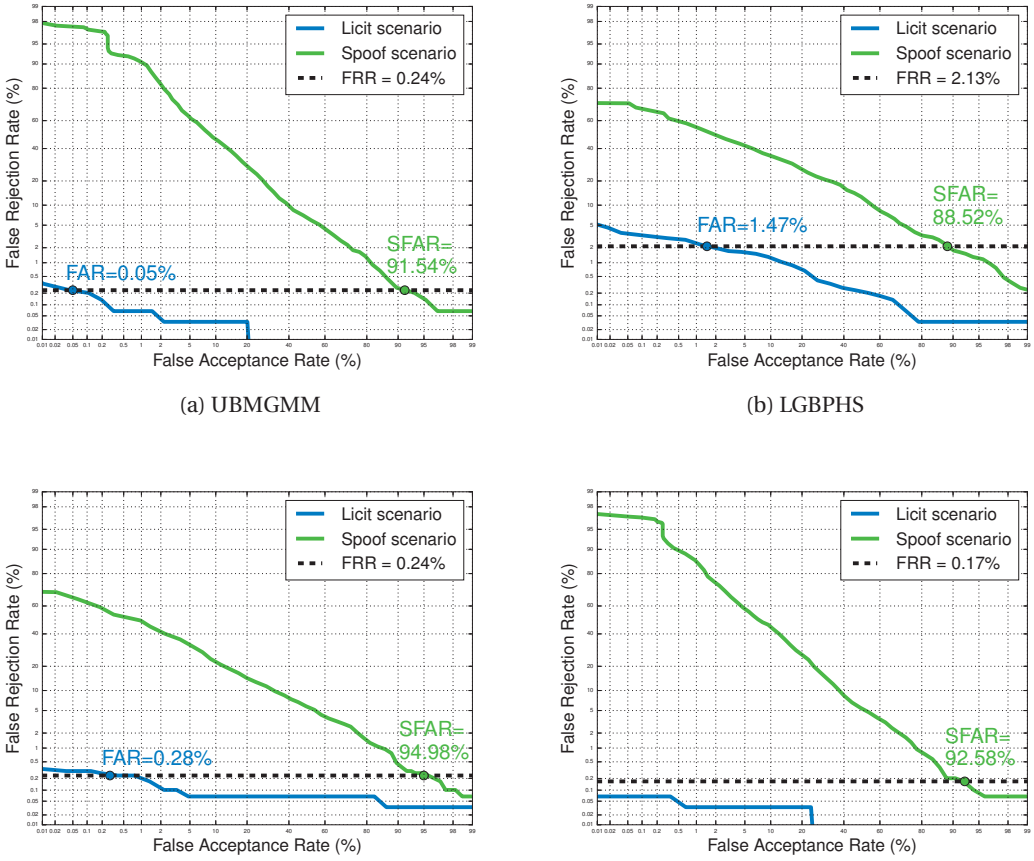
(a) UBMGMM (b) LGBPHS

(c) GJet (d) ISV

Figure 6.18: Score distribution plots for baseline face verification systems using Evaluation Methodology 2.

To assess the verification performance of a system using Evaluation Methodology 2, we determine a decision threshold using the EER criteria and considering only the licit scenario. The vertical lines in Fig. 6.18 correspond to the thresholds for the various systems. Using this decision threshold, we can compute and report FRR, FAR and SFAR. These values for the four baseline systems are given in Table 6.5.

The results show that all the four systems perform well in the verification task. Fig. 6.18 justifies the results: the score distributions for the genuine users and impostors are almost perfectly separated. However, if we keep the decision threshold selected at EER on the development



(a) UBMGMM

(b) LGBPHS

(c) GJet

(d) ISV

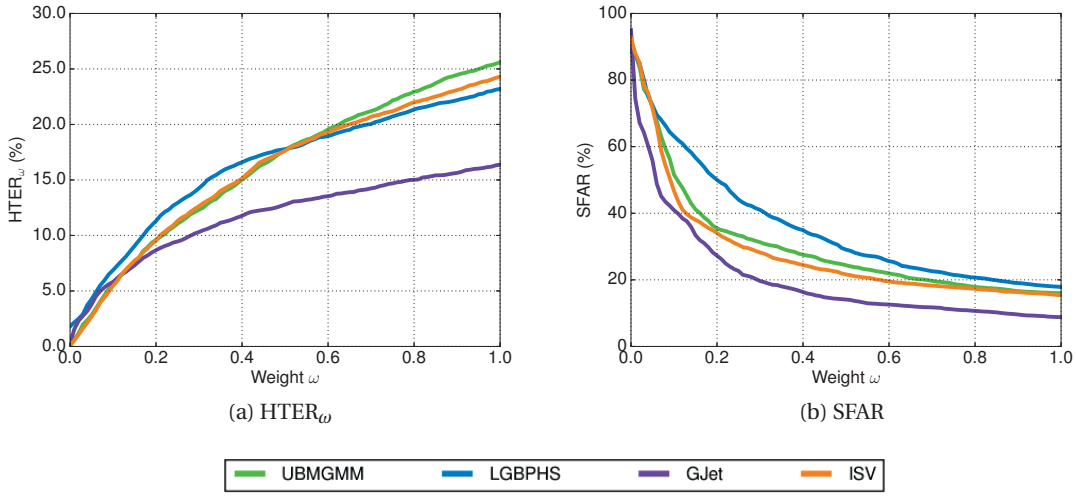
Figure 6.19: Performance of baseline face verification systems using DET curves and Evaluation Methodology 2

Table 6.5: Performance of baseline face verification systems using Evaluation Methodology 2 (in %)

system	FAR	FRR	HTER	SFAR
UBMGMM	0.05	0.24	0.14	91.54
LGBPHS	1.47	2.13	1.8	88.52
GJet	0.28	0.24	0.26	94.98
ISV	0.00	0.17	0.08	92.58

set for the licit protocol, the systems exhibit a great vulnerability to spoofing of around 90%. The results come with no surprise: as suggested by Fig. 6.18, the baseline face verification systems appear to belong to the categories susceptible or vulnerable to spoofing attacks. Using Evaluation Methodology 2, ISV, with 0.08% of HTER seems to perform the best in the verification task. At the same time, GJet, with 94.98% of SFAR, appears to be the most

vulnerable to spoofing among all the systems. These values are obtained only for a threshold which does not assume any spoofing attacks to be possible.



(a) HTER$_\omega$ (b) SFAR

Figure 6.20: EPSC to compare baseline face verification systems

We now proceed with EPS evaluation of the systems. The EPSC given in Fig. 6.20, report HTER$_\omega$ and SFAR for a threshold which considers the relative probability of spoofing attacks, encoded in the parameter ω. Analyzing the EPSC for the four baseline systems, we come to different conclusions. Comparing the HTER$_\omega$ values in Fig. 6.20a, we observe that ISV is best performing only as long as the spoofing attacks appear with a very small probability. After a certain value of ω, GJet shows the best performance. The same applies to the vulnerability to spoofing (Fig. 6.20b): while being the most vulnerable when $\omega \approx 0$, GJet displays the smallest values of SFAR for larger values of ω.

Keeping in mind the conclusions from above, we can discuss two advantages of EPSC over Evaluation Methodology 2. Firstly, it overcomes the exclusiveness in analyzing only zero-effort impostors or spoofing attacks at a time, which is a feature of Evaluation Methodology 2. The HTER$_\omega$ summarizes all the three error rates (FRR, FAR and SFAR) into a single value, combining them based on their cost or the prior of the input classes. Secondly, it rectifies the bias that Evaluation Methodology 2 demonstrates by neglecting the spoofing attacks that may appear. Although this may increase the value of HTER$_\omega$ (EPSC is usually ascending for HTER$_\omega$), it is going to greatly improve the systems vulnerability to spoofing (EPSC is descending for SFAR), especially in conditions where spoofing attacks are highly probable. Finally, by selecting an *a priori* threshold, EPSC allows to objectively compare several systems on the same figure.

The superiority of GJet system which can be visually observed on Fig. 6.20a, can be further confirmed by the value AUE = 0.117. In comparison, AUE = 0.1581 for ISV, which appears to be the second best system.

6.3.2 Performance of Fused Systems

We continue the analysis by comparing the performance of fused face verification and anti-spoofing systems. The comparison is done with respect to the fusion method and whether client-independent or client-specific anti-spoofing system was used. Due to space constraints, for many of the experiments we give the figures only for specific case studies involving a preselected face verification algorithm, which is GJet, as the best performing system from the experiment in Section 6.3.1. From this point onward, all the results will be evaluated using only EPS framework.

Comparison of Fusion Methods

To compare the fusion methods, we rely on case studies with the following specifications:

- GJet for face verification;
- client-specific discriminative system for anti-spoofing;
- LBP, LBP-TOP, MOTION and HOG anti-spoofing features.

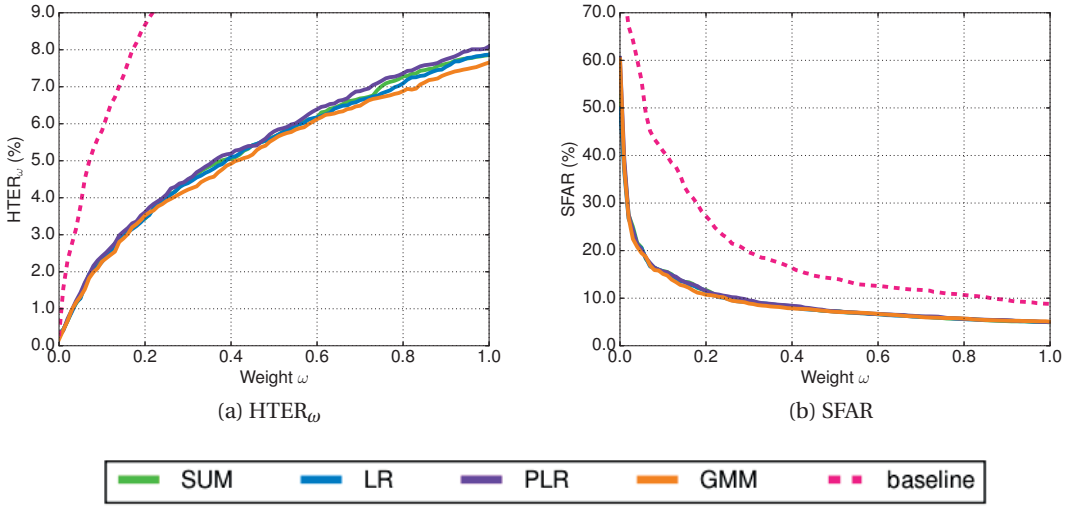
Table 6.6 gives the AUE values for the fusion methods. Comparing these values with AUE value of the baseline GJet system, which is 0.117, we can notice that all the score-level fusion methods yield a significant improvement. For example, the AUE value dropped to 0.0191 for the system fused with LBP-TOP counter-measure and GMM fusion rule. The decision-level fusion method, with AUE value similar to the one of the baseline, is a notable exception, Therefore, we will treat the case of decision-level fusion method separately.

The improvement of the performance after score-level fusion is further reflected on the EPSC for the fused systems, shown in Fig. 6.21, Fig. 6.22, Fig. 6.23 and Fig. 6.24 for different features. It is important to note the considerable drop of HTER_ω and SFAR with respect to the baseline. As ω increases, meaning that spoofing attacks are given greater role in determining the decision threshold, HTER_ω increases, but not as steep as in the case of the baseline system.

To compare systems using their vulnerability to spoofing as a criteria, we look at the SFAR plots shown in Fig. 6.21b, Fig. 6.22b, Fig. 6.23b, and Fig. 6.24b. We notice that, with respect to this criteria, ω has a big impact on the score-level fusion strategies. The vulnerability to spoofing for all systems fused at score-level is very high when $\omega \approx 0$, as the spoofing attacks have a very small contribution in determining the decision threshold. As ω increases, mirroring the consideration of SFAR in the computation of the decision threshold, the vulnerability to spoofing of the systems decreases very quickly. For large values of ω, the score-level fusion methods have very low vulnerability to spoofing. In fact, when comparing SFAR using EPSC, and considering that ω may be associated with the probability of spoofing attacks, is important that SFAR is small for large values of ω. Due to the same reason, large vulnerability to spoofing attacks for small values of ω can be considered as less harmful.

Table 6.6: AUE values for fused systems: **GJet** face verification and discriminative client-specific anti-spoofing method (in %). AUE = 0.117 for the baseline GJet system

	SUM	LR	PLR	GMM	AND
LBP	0.0531	0.0526	0.0542	**0.0513**	0.1024
LBP-TOP	0.0215	0.0223	0.022	**0.0191**	0.1143
MOTION	0.0569	0.0511	0.0498	**0.0411**	0.1786
HOG	0.0564	0.0518	0.1259	**0.0501**	0.0957



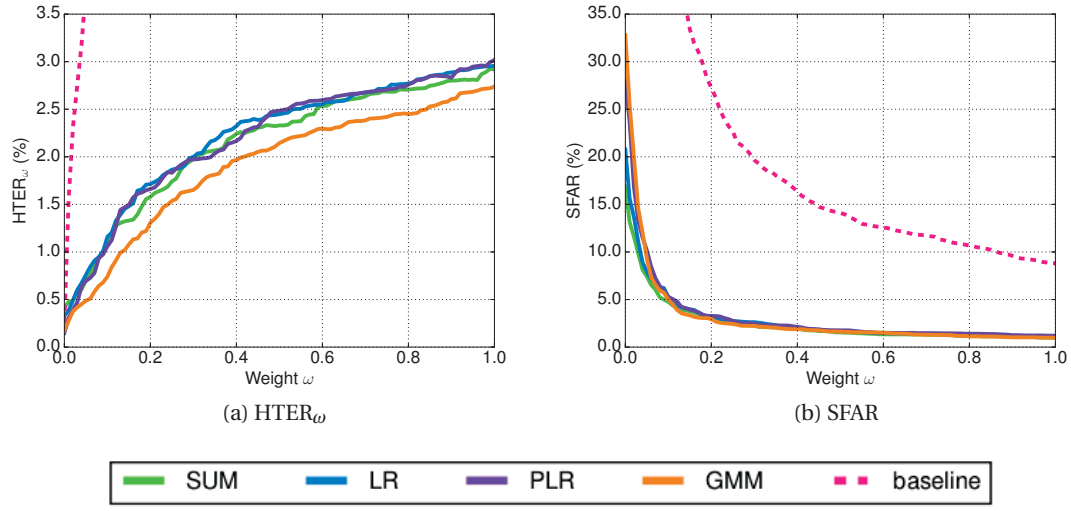
(a) HTER$_\omega$ (b) SFAR

Figure 6.21: Comparison of score-level fusion methods using EPSC: **LBP features**

In general, the GMM fusion rule yields the smallest AUE value for any of the anti-spoofing features. This can be also observed in Fig. 6.21a, Fig. 6.22a, Fig. 6.23a and Fig. 6.24a, comparing the EPSC for HTER$_\omega$. GMM fusion rule yields superior results over the full range of ω. However, EPSC for SFAR shows that in particular cases other fusion methods may yield fused systems with better robustness to spoofing attacks. This is the case, for example, for LBP-TOP features, where better robustness to spoofing can be achieved by using SUM and LR fusion methods for small values of ω. Similar is the case of MOTION features, where SUM fusion method performs the best in terms of SFAR when $\omega < 0.4$.

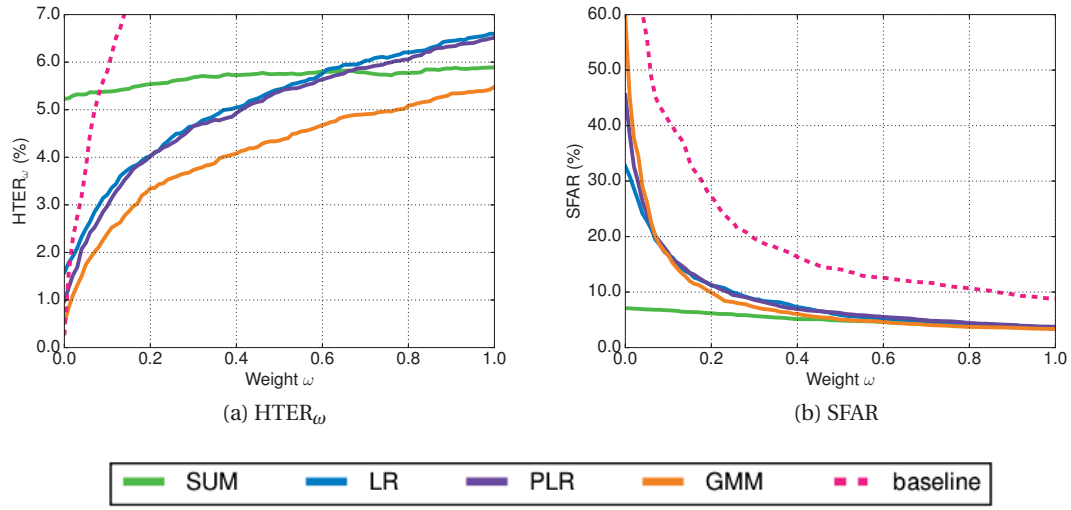
The presented analysis reveals that the conclusions emerging from EPSC depend on different factors. First, they depend on the the selected criteria to be analyzed, like HTER$_\omega$ or SFAR. Furthermore, the error rates vary over the range of ω, and different methods may prevail for different values of ω. Thanks to this property, EPSC enables one to select the method which performs the best for the preferred value of ω.

We continue the analysis by comparing the best performing score-level fusion method in terms of AUE, GMM, with the decision-level fusion method based on logical AND. Due to



(a) HTER$_\omega$ (b) SFAR

Figure 6.22: Comparison of score-level fusion methods using EPSC: **LBP-TOP features**

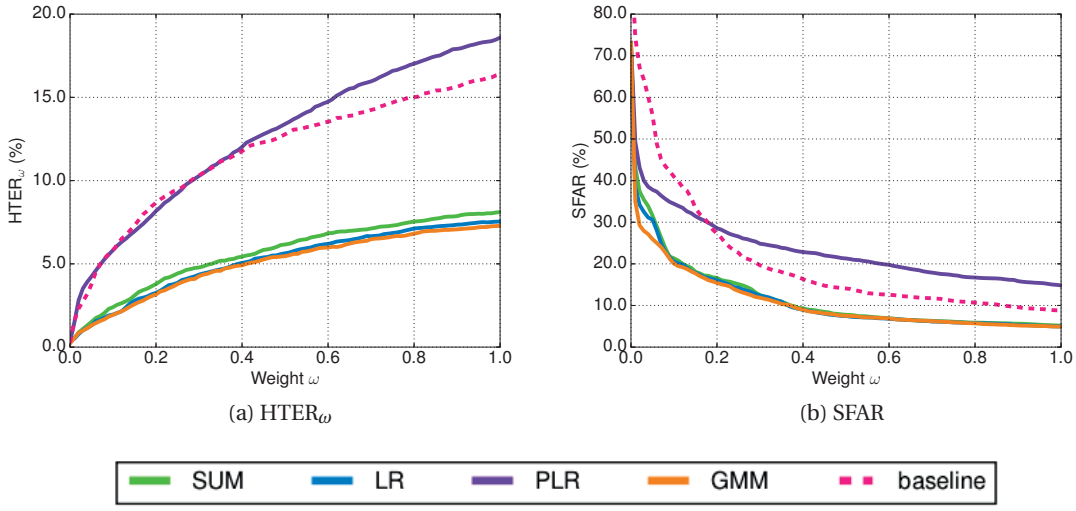


(a) HTER$_\omega$ (b) SFAR

Figure 6.23: Comparison of score-level fusion methods using EPSC: **MOTION features**

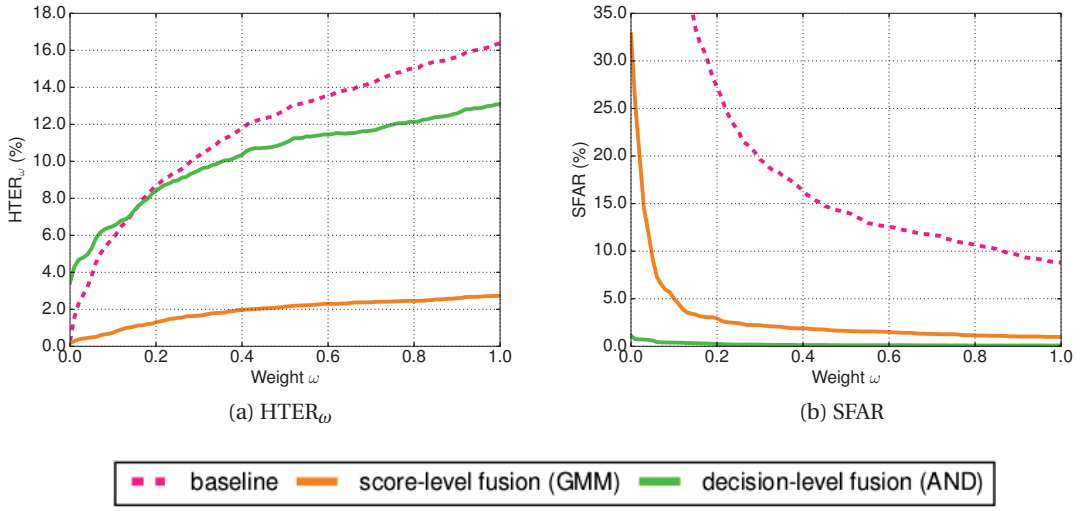
space constraints, we adhere just to LBP-TOP anti-spoofing features. The EPSC comparing the fused systems is shown in Fig. 6.25.

Fig. 6.25a showing the EPSC for HTER$_\omega$ confirms the conclusion derived from the comparison of AUE values in Table 6.6: AND fusion rule performs on similar scale as the baseline system. However, Fig. 6.25b reveals that the system fused with AND fusion rule exhibits impressively low vulnerability to spoofing, compared to both the baseline system, as well as the system fused



(a) HTER$_\omega$ (b) SFAR

Figure 6.24: Comparison of score-level fusion methods using EPSC: **HOG features**

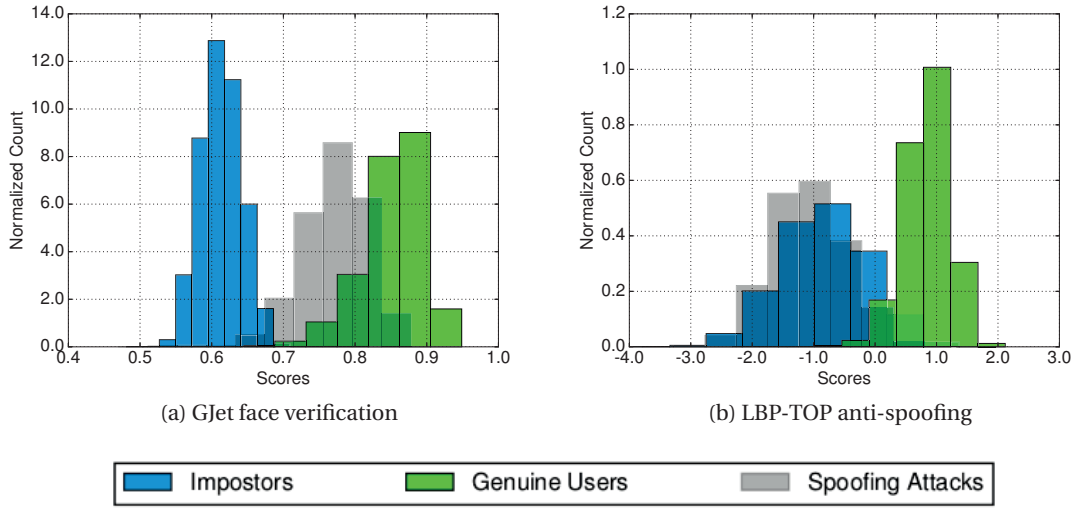


(a) HTER$_\omega$ (b) SFAR

Figure 6.25: Comparison of score-level and decision-level fusion methods using EPSC: **LBP-TOP features**

with GMM fusion method. The reason becomes evident if we analyze the score distribution plots of the face verification and anti-spoofing systems, given in Fig. 6.26a and Fig. 6.26b, respectively. The figures show what was clarified in Section 4.2 and Table 4.1: the verification system is trained to reject zero-effort impostors, while the anti-spoofing system is trained to reject spoofing attacks. Hence the large overlap of genuine users and spoofing attacks for the verification system in Fig. 6.26a, and of the zero-effort impostors and spoofing attacks for the

anti-spoofing system in Fig. 6.26b. As ω increases, the increased importance of the spoofing attacks for the face verification system causes a threshold that results in increased FRR. At the same time, the increase of ω does not affect the FRR for the anti-spoofing system, but contributes to low SFAR. In total, after fusion with logical AND, the FRR, and hence HTER_ω remains similar to the one for the baseline face verification system, while SFAR becomes low.



(a) GJet face verification

(b) LBP-TOP anti-spoofing

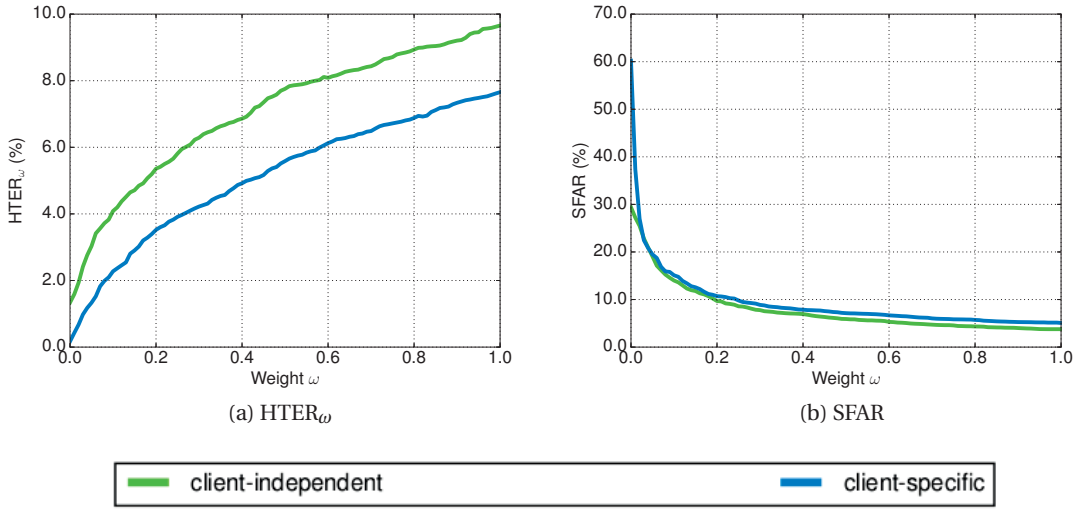
Figure 6.26: Score distributions of face verification and anti-spoofing systems

Fusion with Client-Independent and Client-Specific Anti-spoofing Systems

In Section 6.2, we showed that the client-specific anti-spoofing methods undoubtedly outperform the client-independent ones in most of the cases. In the following experiment, we compare client-independent and client-specific systems in a context of a face verification system they are fused with. The comparison is done on case studies with the following specifications:

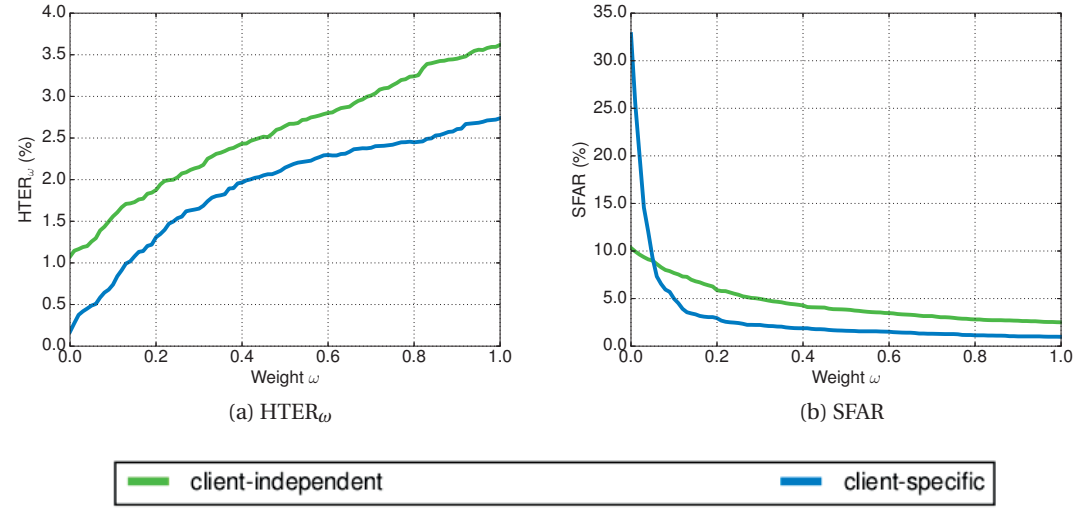
- GJet for face verification;
- LBP, LBP-TOP, MOTION and HOG anti-spoofing features;
- discriminative anti-spoofing systems;
- best performing fusion method in terms of AUE (LR for client-independent systems based on HOG, GMM for all the rest).

The best performing fusion methods for the systems based on client-independent countermeasure were found in a separate experiment, the detailed results of which are not reported.



(a) HTER$_\omega$ (b) SFAR

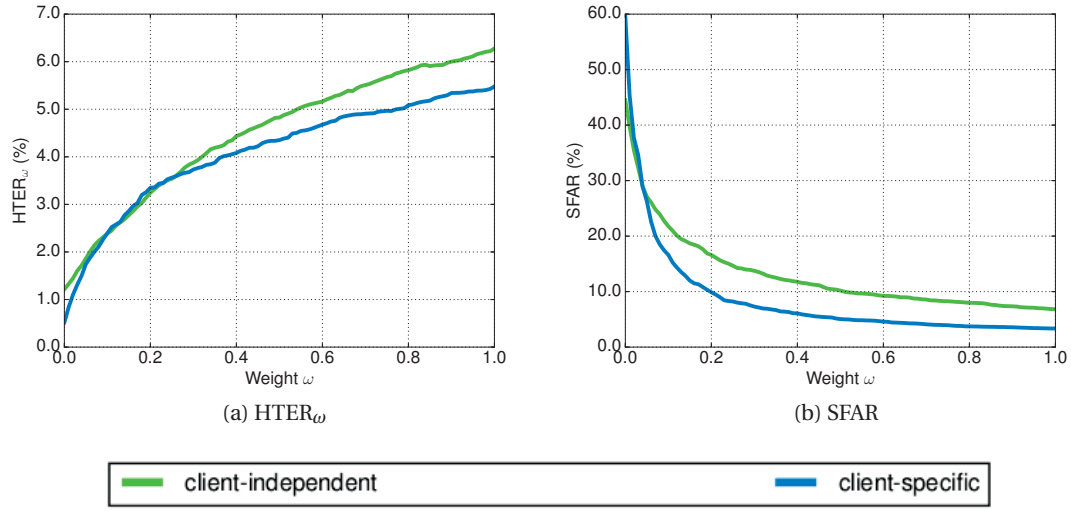
Figure 6.27: Comparison of systems fused with client-independent and client-specific methods using EPSC: **LBP features**



(a) HTER$_\omega$ (b) SFAR

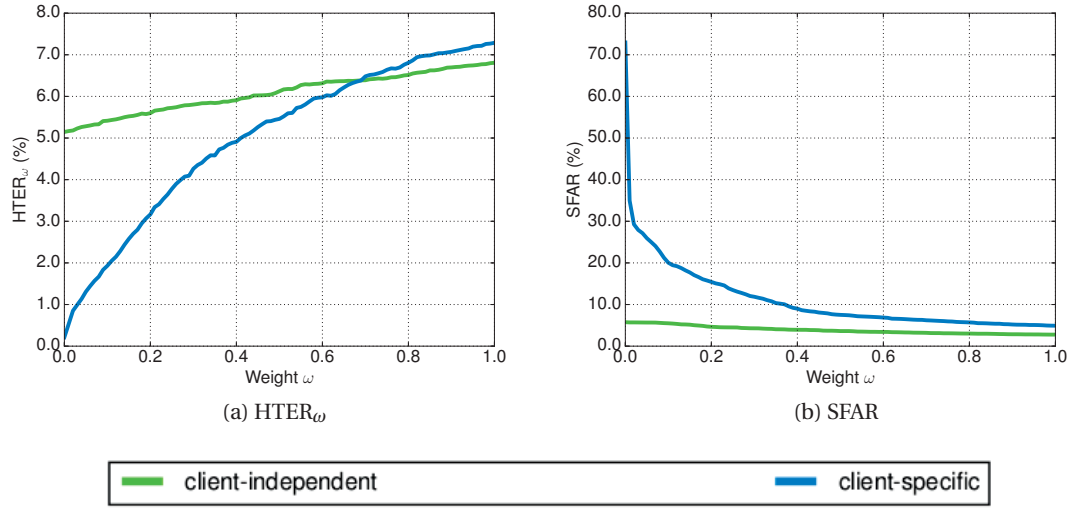
Figure 6.28: Comparison of systems fused with client-independent and client-specific methods using EPSC: **LBP-TOP features**

Fig. 6.27, Fig. 6.28, Fig. 6.29 and Fig. 6.30 show the EPSC plots for the four anti-spoofing features. The EPSC for HTER$_\omega$ shown in Fig. 6.27a, Fig. 6.28a, Fig. 6.29a and Fig. 6.30a support the superiority of the client-specific anti-spoofing methods. This time, it is confirmed for a wide range of decision thresholds which depend on the parameter ω.



(a) HTER$_\omega$ (b) SFAR

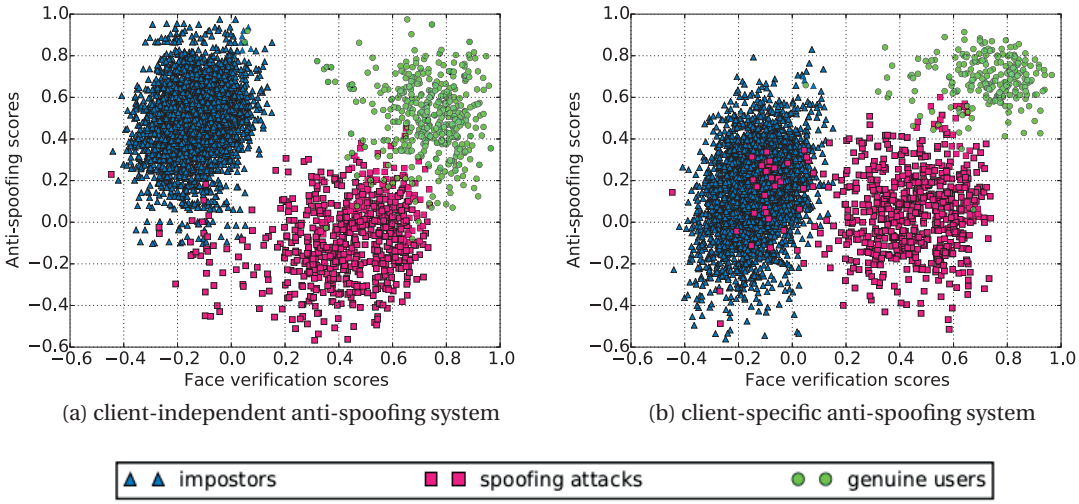
Figure 6.29: Comparison of systems fused with client-independent and client-specific methods using EPSC: **MOTION features**



(a) HTER$_\omega$ (b) SFAR

Figure 6.30: Comparison of systems fused with client-independent and client-specific methods using EPSC: **HOG features**

The advantage of the systems fused with client-specific counter-measure can be explained by the scatter plots of the face verification and anti-spoofing scores, shown in Fig. 6.31. The figure compares the scatter plots when using client-independent and client-specific counter-measure based on LBP-TOP.



(a) client-independent anti-spoofing system

(b) client-specific anti-spoofing system

Figure 6.31: Scatter plots of face verification and anti-spoofing system scores

We draw the attention to zero-effort impostor scores on Fig. 6.31b, which visually appear to be displaced lower with respect to the zero-effort impostor scores on Fig. 6.31a. Although particularly trained to assign low scores to spoofing attacks, the client-specific anti-spoofing system assigns low scores also to the zero-effort impostors. Having a wrong identity claim at input, the client-specific system compares the zero-effort impostor samples to the anti-spoofing model of another client, which results in a low anti-spoofing score. In this sense, the anti-spoofing system partially performs a verification task as well, which appears to be highly beneficial to the overall performance of the fused system.

With respect to the vulnerability to spoofing shown in Fig. 6.27b, Fig. 6.28b, Fig. 6.29b and Fig. 6.30b the systems fused with client-independent counter-measure perform better than the client-specific ones for LBP and HOG features. This applies also to LBP-TOP and MOTION features, but only for $\omega < 0.1$. The client-specific approaches regain their advantage when more importance is given to SFAR. We emphasize that better robustness to spoofing is more important for large values of ω.

Comparison of Fused and Baseline Systems

We finalize the experiments by comparing the performance of the baseline face verification systems before and after they are fused with a spoofing counter-measure. We consider case studies with the following configuration:

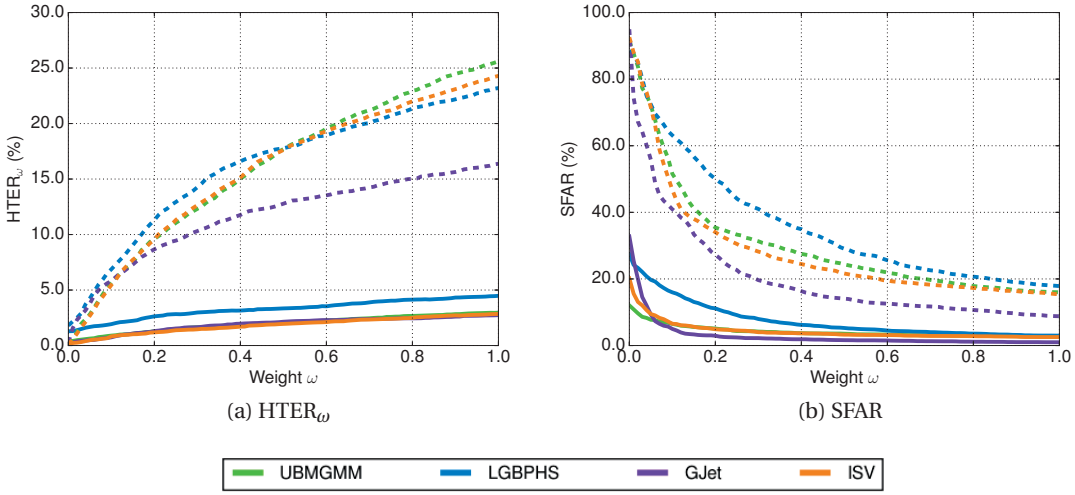
- UBMGMM, LGBPHS, GJet and ISV for face verification;
- LBP-TOP anti-spoofing features;

- client-specific discriminative system for anti-spoofing;
- GMM for fusion.

According to the experiments previously presented in this section, GMM performs the best as a fusion strategy for GJet face verification and LBP-TOP client-specific discriminative anti-spoofing system. In a similar analysis, the results of which are omitted, we found GMM to be the best performing fusion strategy also for UBMGMM, LGBPHS and ISV face verification systems. This is the reason why it is the fusion strategy of choice for this experiment. The AUE values of the baseline and fused methods are given in Table 6.7.

Table 6.7: AUE values for face verification systems before and after fusion with anti-spoofing system

system	AUE of baseline	AUE of GMM fused system
UBMGMM	0.1615	0.0196
LGBPHS	0.1622	0.0328
GJet	0.117	0.0191
ISV	0.1581	0.0183



(a) HTER$_\omega$ (b) SFAR

Figure 6.32: Comparison of baseline (dashed line) and fused (full line) systems

Fig. 6.32 presents the EPSC plots comparing the fused and the baseline systems. The plots convey a similar message as the AUE values: fusion brings a considerable improvement over the baseline systems. As shown in Fig. 6.32a, the baseline and the fused systems behave similarly for small values of ω. It appears that, in some cases, the baseline system even performs better in terms of HTER$_\omega$ when $\omega \approx 0$. As the prior of spoofing attacks is small at this point, fusion with spoofing counter-measure only undesirably increases HTER$_\omega$. However, with the increase of ω, the baseline systems note a drastic rise of HTER$_\omega$. Contrary to the

expectations and as can be concluded from the descending curves in Fig. 6.32b, this is not due to increased vulnerability to spoofing. Rather, it is a result of FRR which increases with ω due to the highly overlapping score distributions for genuine users and spoofing attacks for the baselines, as shown in Fig. 6.18.

Being secured with anti-spoofing methods, the rise of HTER_ω for the fused systems is mild across the full range of ω. This can be explained by their score distribution plots shown in Fig. 6.33. Compared to the plots for the baseline systems given in Fig. 6.18, the fused systems have a spoofing attack score distribution which is shifted towards the zero-effort impostor score distribution, which is a desirable property of trustworthy verification system. Fusion is thus bringing systems which are otherwise categorized as vulnerable or susceptible to spoofing attacks, towards the category of systems robust to spoofing attacks.

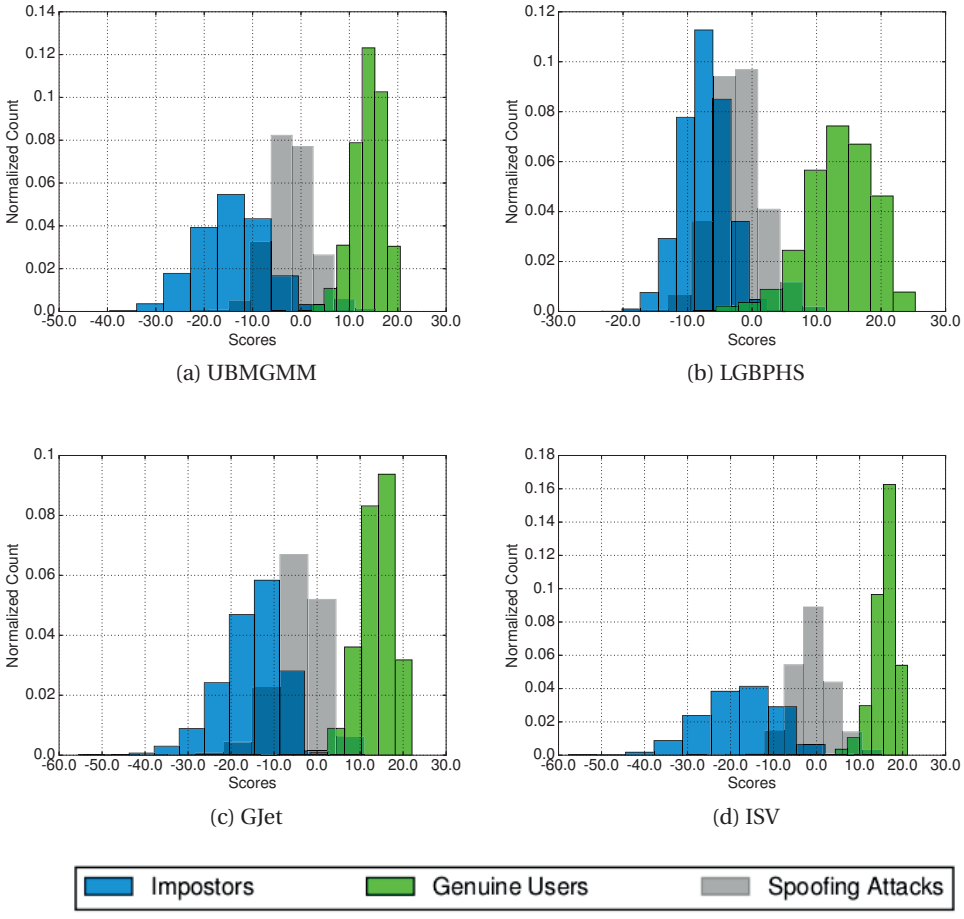
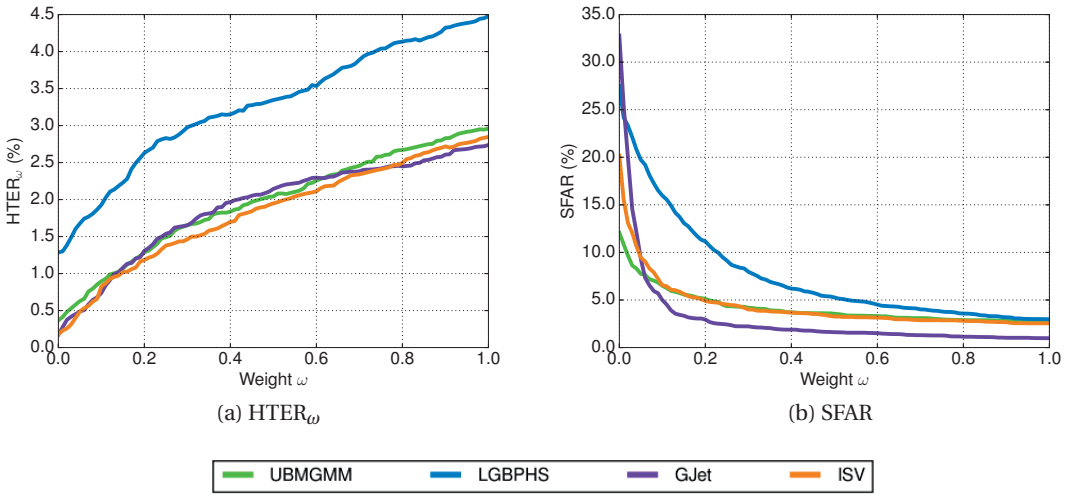


Figure 6.33: Score distribution plots for fused systems

As shown in Fig. 6.32b, the vulnerability to spoofing goes up to unacceptable levels of ≈ 90% for baseline systems. In comparison, the vulnerability to spoofing is up to 10 times lower, even

when ω is small and there is little impact of the spoofing attacks on the decision threshold. More importantly, with the increase of ω, the vulnerability to spoofing attacks drops even further to below 5% for all the systems. As was emphasized several times throughout this thesis, lower SFAR is particularly important for high values of ω.

Finally, Fig. 6.34 gives a closer look at the fused systems only. According to the AUE values given in Table 6.7, the system based on ISV performs the best. However, Fig. 6.34a shows that the systems based on ISV and GJet face verification are interchangeably better for different values of ω. Similarly to other example before, this example illustrates how EPSC enables to select the best system based on the expected prior or cost of spoofing attacks for a given application. In this case, the system based on ISV will be the recommended choice for applications where $\omega < 0.8$, and GJet otherwise. As shown in Fig. 6.34b, the system based on GJet also exhibits higher robustness to spoofing than ISV for $\omega > 0.1$



(a) HTER$_\omega$ (b) SFAR

Figure 6.34: Comparison of fused systems

6.3.3 Summary

The presented evaluations show the vulnerability of several state-of-the-art face verification systems to spoofing attacks. More importantly, they demonstrate how these vulnerabilities can be diminished by fusing face verification and anti-spoofing system at output-level. The performance of the fused systems depends on several factors, like the face verification and anti-spoofing systems which are tested, as well as the fusion mechanism.

Integral and fair comparison of biometric verification systems under spoofing attacks can be achieved using EPS framework. Using this framework, we show the benefits of considering different levels of relative importance of the spoofing attacks, coded into the parameter ω. Two systems may be interchangeably better in terms of performance for two different ranges

of the values of ω. This is the case, for example, for the fused systems based on ISV and GJet. The EPS framework enables comparison of systems based on different criteria, like HTER_ω or SFAR.

Using EPSC, we can verify that the increased robustness to spoofing attacks obtained by fusion does not compromise the system's verification performance. Indeed, for the majority of fused systems, the overall performance of the system represented by HTER_ω is significantly better than that of the baselines. For some systems, exceptions happen for very small values of ω, signifying very low importance of spoofing attacks. In such cases, anti-spoofing systems are not necessary and unprotected baseline verification systems should be used instead.

6.4 Discussion

The integration concepts for biometric verification and anti-spoofing systems at input and output-level were illustrated using case studies for the face mode. They involve state-of-the-art face verification systems and anti-spoofing features. To perform the evaluation, we used widely accepted evaluation methodologies, as well as the novel EPS framework developed specifically for integrated evaluation of verification systems under spoofing attacks.

We would like to emphasize that performing experiments for all the possible case studies that could be built upon the considered face verification, anti-spoofing and fusion methods is not an easily feasible task. Therefore, in this chapter we presented the results only of a subset of the case studies. The presented case studies were selected so that they effectively demonstrate the advantages of input-level integration and the importance of output-level integration. Furthermore, they are designed to appropriately illustrate the convenience of use of EPS framework for evaluation of biometric verification systems under spoofing attacks. More extensive analysis covering a broader set of case studies can be done using the freely available software package `bob.thesis.ichingo2015`.

One of the objectives of the experiments presented in this chapter is to demonstrate the necessity for integrated evaluation of biometric verification and anti-spoofing systems. The results point out that the performance of the fused system depends on all the three involved methods: the biometric verification, anti-spoofing and fusion. It is difficult to anticipate the performance of an integrated system by evaluating them independently. For example, while one face-verification system may perform better than the other when operating independently, the situation might change once they are fused with an anti-spoofing system.

Some of the experiments presented in this chapter are focused on comparing the EPS framework with other evaluation methodologies. They reveal that using other evaluation methodologies may lead to suboptimal decision thresholds and biased conclusions. Evaluation performed with the EPS framework enables to consider different aspects related to the integrated operation of biometric verification and anti-spoofing systems, like its overall performance or vulnerability to spoofing attacks. More importantly, by accounting for the probability or the

cost of different inputs, it enables putting the system into the context where it is going to be deployed and take versatile decisions about the best method to use.

7 Conclusions

At the present degree of maturity, many verification systems are highly vulnerable to spoofing attacks. Effective counter-measures are thus of essential importance for the development of trustworthy biometric verification systems. The topic has been gaining increasing popularity, resulting in a plethora of anti-spoofing methods appointed for different biometric modes and targeting different types of spoofing attacks.

The objective of this thesis is to address the integration of biometric verification and anti-spoofing systems. The first step in the process is to recognize and highlight the necessity of cooperation between the two types of systems as a fundamental issue to the process of deploying them in realistic conditions. We identify three points of integration situated along the biometric verification pipeline. With respect to these points, the integration can be performed at input-level, output-level and at evaluation.

At input-level, the integration is concerned with sharing the information that the biometric verification and anti-spoofing systems use. In particular, we refer to the enrollment samples which biometric verification systems use to create client models, as well as the identity claim which they use at query time. We investigate how making this information accessible by the anti-spoofing system affects its performance.

Output-level integration is important for consolidating the results of the biometric verification and anti-spoofing systems into a single decision. Analyzing several multiple expert fusion approaches, we study what is the impact of the fusion methods on the performance of the final system.

At evaluation level, we investigate whether independent evaluation of biometric verification and anti-spoofing systems is sufficient and we argue that integrated evaluation is of major importance for selecting the most adequate system for certain application. Highlighting the limitations of current evaluation methodologies, we propose a novel framework for evaluating biometric verification systems under spoofing attacks, referred to as EPS.

The achievements of this thesis and the conclusions drawn from the experimental results are

summarized in Section 7.1. The limitations of the proposed methods and the prospects for future work are discussed in Section 7.3.

7.1 Experimental Findings and Achievements

Experimental evaluation of the integration concepts covered in this thesis was performed at input-level and output-level, using the newly proposed framework for integrated evaluation. All the experiments were performed on case studies involving verification and anti-spoofing methods for the face mode, as one of the most vulnerable biometric modes. In this setup, the experimental results and the accompanying analysis confirm that integration of biometric verification and anti-spoofing systems is beneficial at all three levels and is of great importance for creating trustworthy biometric systems.

Input-level integration. By unifying the information that biometric verification and anti-spoofing systems use at input, we are able to create client-specific anti-spoofing systems which take the decision based not only on the input sample, but also on the client identity. We built two types of client-specific systems. The first one is based on a generative paradigm and uses GMM to model real accesses for each client and spoofing attacks for a set of cohort clients. The second one follows a discriminative principle and uses SVM to draw decision boundaries between real accesses of each client and spoofing attacks from a set of cohort clients.

The evaluation of client-specific systems was performed with the objective to assess whether input-level integration can bring improved performance on particular anti-spoofing features. Comparison of anti-spoofing features, as well as attempts to outperform state-of-the-art results which are obtained by combining different anti-spoofing features, is out of the scope of this thesis.

The experimental results on Replay-Attack database lead to the following conclusions:

(a) Client-specific variations in the scores of client-independent anti-spoofing systems exist for all tested anti-spoofing features. Client-specific anti-spoofing systems exhibit less score variations between different clients.

(b) The performance of the client-specific anti-spoofing systems depend on several factors. For the generative systems, most important factors are the number of Gaussian components, as well as the selection of the cohort set. For the discriminative ones based on SVM, the most important factor is the choice of kernel.

(c) Depending on the features and the database protocol, client-specific anti-spoofing systems can relatively improve the performance by more than 50%. In general, features based on visual appearance seem to be better suited for client-specific systems than the ones based on motion.

(d) Client-specific systems exhibit significantly better generalization in detecting types of attacks which have not been seen during training time. Even in this case, the relative improvement can be more than 50% for the visual appearance features. While client-independent systems have unsatisfactory results on unseen types of attacks, the client-specific ones can bring the HTER to less than 10% in some cases.

These conclusions are taken based on the four state-of-the-art features considered in this thesis. More features need to be analyzed to be able to generalize the claims to whole categories of anti-spoofing features.

Output-level integration. Biometric verification and anti-spoofing systems can be considered as experts in different domains: the former for recognizing identities, and the latter for detecting counterfeit input samples. A verification system robust to spoofing attacks can be realized by output-level integration of the two systems and taking a multiple expert fusion approach.

We performed experiments considering several fusion methods working on score-level and decision-level. They lead to the following conclusions:

(a) Fusion generally helps to accomplish systems with greater robustness to spoofing without compromising the verification performance, regardless of the fusion strategy. However, in our experiments, fusion based on GMM appears to be superior over the other fusion strategies.

(b) The systems fused with client-specific counter-measure perform significantly better than the ones fused with client-independent one. Certainly, the most important reason is better spoofing detection capabilities for the client-specific counter-measures. Another reason is the fact that a client-specific counter-measure partially behaves as a verification system, giving low scores not only to spoofing attacks, but to zero-effort impostors as well.

(c) While unsatisfactory for baseline face verification systems, the vulnerability to spoofing attacks drops to below 5% after fusion with a spoofing counter-measure. At the same time, the overall performance of the system, which is reported by accounting all the three types of error rates, is significantly improved as well. The improved results are likely to be a consequence of the distribution of spoofing attack scores, which after fusion is shifted towards the distribution of zero-effort impostors.

Once again, these conclusions emerge from the experiments conducted on the case studies considered in this thesis. Fusion strategies other than GMM may perform better on case studies with different verification and anti-spoofing features.

Integrated evaluation. Independent evaluation of anti-spoofing systems is important to realize their power to discriminate between genuine accesses and spoofing attacks. However, it gives no insight on the reliability of a joint decision with a verification systems in the general goal to accept genuine users and reject both zero-effort impostors and spoofing attacks.

The objective of integrated evaluation is not only to assess the performance of systems under spoofing attacks, but also to account for all the possible inputs to the system: genuine users, zero-effort impostors and spoofing attacks. In the evaluation framework proposed in this thesis, EPS, this is taken in consideration by parameters which can be interpreted as the prior probabilities of the different inputs or the relative costs of the error rates associated with them.

The EPS framework is extensively used to evaluate the results of the experiments in this thesis. Based on the experience of their usage in these practical examples, we came to the following conclusions:

(a) The most important advantage of EPS framework over other evaluation methodologies is that it computes a decision threshold which is optimized given all the three types of inputs. In this way, SFAR of the system is not unnecessarily augmented just because spoofing attacks are not considered.

(b) Using the EPSC, one can report unbiased comparison of systems, as the decision threshold is computed *a priori* on a separate data.

(c) The EPS framework allows for comparison of systems based on different criteria and for different values of the cost parameters. In this way, it enables selection of the best performing method for a given application.

Reproducible results. The code for the experiments performed in this thesis is available as free software in the package `bob.thesis.ichingo2015`[1] and can be used to reproduce all the presented results. Furthermore, it can be used to extend the analysis to additional case studies for the face mode that can be created by combining other verification and anti-spoofing systems. In addition, a complete API for using the EPS framework is provided in the package `antispoofing.evaluation`[2]. It can be used for evaluation of biometric verification systems under spoofing attacks regardless of the biometric mode, internal operation of the system and evaluation database.

7.2 Related Publications

During the course of the work on the individual methods proposed in this thesis, we have published the following publications:

[1] https://pypi.python.org/pypi/bob.thesis.ichingo2015
[2] https://pypi.python.org/pypi/antispoofing.evaluation

Journals

- I. Chingovska, A. Anjos; *On the use of client-specific information for face anti-spoofing*; IEEE Transactions on Information Forensics and Security; Special Issue on Biometric Anti-spoofing; 10(4):787-796, 2015
- I. Chingovska, A. Anjos, S. Marcel; *Biometric evaluation under spoofing attacks*; IEEE Transactions on Information Forensics and Security, 9(12):2264-2276, 2014

Conference proceedings

- I. Chingovska, A. Anjos, S. Marcel; *Anti-spoofing in action: joint operation with verification system*; Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Biometrics; Portland, Oregon, 2013
- I. Chingovska, et al.; *The 2nd competition on counter measures to 2D facial spoofing attacks*; Proceedings of International Conference on Biometrics (ICB); Madrid, Spain, 2013
- I. Chingovska, A. Anjos, S. Marcel; *On the Effectiveness of Local Binary Patterns in Face Anti-spoofing*; Proceedings of the 11th International Conference of the Biometrics Special Interes Group; Darmstadt, Germany 2012

Book chapters

- I. Chingovska, N. Erdogmus, A, Anjos, S. Marcel. *Face Recognition Systems Under Spoofing Attacks*, Face Recognition Across the Electromagnetic Spectrum (to appear)
- I. Chingovska, A. Anjos, S. Marcel; *Evaluation Methodologies*; Handbook of Biometric Anti-spoofing; Springer London, pp.185-2014, 2014
- I. Chingovska, A. Anjos, S. Marcel; *Anti-spoofing: Evaluation Methodologies*; Encyclopedia of Biometrics; Springer US, 2014
- I. Chingovska, A. Anjos, S. Marcel; *Anti-spoofing: Face Databases*; Encyclopedia of Biometrics; Springer US, 2014

During the course of the work on the individual methods proposed in this thesis, we have developed the following free software packages:

- `antispoofing.lbp`[3]

[3] https://pypi.python.org/pypi/antispoofing.lbp

- `antispoofing.utils`[4]
- `antispoofing.fusion_faceverif`[5]
- `antispoofing.clientspec`[6]
- `antispoofing.evaluation`[7]
- `antispoofing.competition_icb2013`[8]

7.3 Perspectives for Future Work

The integration of biometric verification and anti-spoofing systems is a subject which has received little attention from the community. The work presented in this thesis is a step forward towards filling the gap in this field. However, there are several possibilities of how this work can be extended and a numerous directions that can be explored in the future.

Overcoming the limitations of the integration concepts presented in this thesis is one of these directions. One limitation is the fact that client-specific anti-spoofing methods resulting from input-level integration can be applied only at query time, but not at enrollment time. A limitation of the output-level integration methods is that they are classical multiple expert fusion methods and are not optimized for the task of fusion of two systems with discordant criteria about the positive and the negative class. Furthermore, they assume no mutual dependence of the verification and anti-spoofing scores, or dependence of the scores on the identity of the client. If such a dependence is found, it may be helpful to better model the score input space and produce better fused scores. The work on output-level integration can also be extended to include client-specific fusion strategies.

The methods proposed in this thesis assume that the biometric verification and anti-spoofing systems have been developed independently, and their integration is performed at a later stage. Digressing from this approach, the two systems could be developed in a joint manner, either at the feature extraction or the modeling step. These are concepts for integration at the intermediate-level, which is a challenging, but important direction for future work.

An issue that needs to be urgently addressed is the insufficient number of spoofing databases that provide separate samples for enrollment of clients. The lack of such databases not only hinders the development of methods for integration of biometric verification and anti-spoofing systems, but also prevents training a biometric verification system and assessing its vulnerability to spoofing. Unfortunately, Replay-Attack is the sole example of a database with these properties for the face mode. It is essential that databases developed in the future provide

[4] https://pypi.python.org/pypi/antispoofing.utils
[5] https://pypi.python.org/pypi/antispoofing.fusion_faceverif
[6] https://pypi.python.org/pypi/antispoofing.clientspec
[7] https://pypi.python.org/pypi/antispoofing.evaluation
[8] https://pypi.python.org/pypi/antispoofing.competition_icb2013

as many clients and as many types of spoofing attacks as possible. Having such databases will enable, for example, to more reliably measure the impact of the cohort set selection for the client-specific anti-spoofing methods, as well as their generalization capabilities on spoofing attacks not seen during training.

The proposed integration concepts were applied and evaluated only on a subset of case studies for the face mode, which involve a limited set of verification systems and anti-spoofing features. While the presented analysis serves well as a proof of concept, it is important to extend it to other systems and assess the applicability of the methods in a wider context. Such a study, for example, can reveal to what extent we can generalize the assumption of client-specific information within different categories of anti-spoofing features. As part of the future work, such analysis should include case studies in other biometric modes.

A Gaussian Mixture Model (GMM)

Let $\mathcal{N}(\mathbf{x}|\mu,\Sigma)$ be a multivariate Gaussian distribution over the data with mean μ and covariance matrix Σ. If $\mathbf{x}$ is d-dimensional variable, then Σ has a dimensionality of dxd and the Gaussian distribution has the form as in Eq. A.1.

$$\mathcal{N}(\mathbf{x}|\mu,\Sigma) = \frac{1}{(2\pi)^{n/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^{\mathrm{T}}\Sigma^{-1}(\mathbf{x}-\mu)\right\} \tag{A.1}$$

A GMM is a weighted sum of M multivariate Gaussian distributions, called components, each parameterized with $\Theta = \{\pi_m, \mu_m, \Sigma_m\}, m = 1..M$, where π_m is its weight. The weights of the components are parameters of a discrete distribution $p(\pi_1, ..., \pi_m)$ and thus must comply to the constraints: $0 \leq \pi_m \leq 1$ and $\sum_{m=1}^{M} \pi_m = 1$. Mathematically, a GMM is formalized as in Eq. A.2.

$$p(\mathbf{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\mathbf{x}|\mu_m,\Sigma_m) \tag{A.2}$$

GMM models a generative process in which samples can be generated using ancestral sampling [Bishop, 2006]. In this process, first a single component from the GMM is randomly selected from the distribution $p(\pi_1, ..., \pi_m)$. Then, the sample is generated from the Gaussian distribution associated with that component. From a probabilistic perspective, the choice of the component is represented by an M-dimensional random variable $\mathbf{z}$ which satisfies $\forall m = 1..M, z_m \in \{0,1\}$ and $\sum_{m=1}^{M} z_m = 1$. In other words, $\mathbf{z}$ is a vector populated with zeros, with a single element with a value 1 at the index representing the chosen component. The value of $\mathbf{z}$ is never explicitly known. Therefore, it is an unobserved, latent variable, as opposed to the sample, which is an observed variable. The distribution over $\mathbf{z}$ is interpreted through the component weight parameters, so that $\forall m = 1..M, p(z_m = 1) = \pi_m$ [Bishop, 2006].

An important issue for GMM is the choice of the covariance matrix Σ. Most often, a separate covariance matrix is trained for each GMM component [Reynolds and Rose, 1995]. With regards to the properties of the covariance matrix, the most general case supports full covariance matrices. However, this option is computationally expensive, as the GMM training requires repeated inversions of the matrix. An alternative is to use GMM with diagonal covariance matrices, which, subject to the number of components M [Reynolds et al., 2000], can model the distribution equally well as a GMM with full covariance matrices.

A.1 GMM Training

The training of a GMM model is a Maximum Likelihood (ML) problem and consists of finding the parameters $\Theta = \{\pi_m, \mu_m, \Sigma_m\}, m = 1..M$ that maximize the likelihood of all the data $X = \{\mathbf{x}_i | i = 1..N\}$, where N is the number of samples. Due to the presence of latent variables, the ML problem is typically solved using the Expectation-Maximization (EM) algorithm [Dempster et al., 1977]. EM is an iterative algorithm which, starting from an initial choice for the parameters Θ, alternates between two steps, expectation (E) and maximization (M). During the process, it adapts Θ to monotonically increase the likelihood of the given data. Therefore, for each iteration t of the algorithm, it is valid that $p(\mathbf{x}|\Theta^t) < p(\mathbf{x}|\Theta^{t+1})$.

At initialization of Θ, the means of the GMM $\mu_m, m = 1..M$ are usually learned in an unsupervised manner, for example using k-means algorithm [Macqueen, 1967; Lloyd, 1982]. Given the current value of Θ, at each iteration t the two steps proceed as follows:

1. E step: The posterior probability distribution $p(\mathbf{z}|\mathbf{X}, \Theta)$ of latent variables is estimated using Bayes theorem, given the data X and the current value of Θ.
2. M step: Θ is updated in order to maximize the expectation of the data log-likelihood computed under the posterior probability distribution found in the E step, as in Eq. A.3.

$$\Theta_{\text{updated}} \leftarrow \arg\max_{\Theta} \sum_{\mathbf{z}} p(\mathbf{z}|X, \Theta) \log p(X, \mathbf{z}|\Theta) \tag{A.3}$$

The process continues to iterate over E and M steps until convergence criteria is met. N. and Hinton [1993] give a proof that EM algorithm indeed converges to Θ that maximizes the likelihood of the data.

A.2 Maximum A-Posteriori Adaptation (MAP)

A significant amount of data is required to correctly estimate GMM parameters using ML parameter estimators. Unfortunately, the enrollment data which is available for creating client-specific anti-spoofing models is rarely more than a few samples and is not enough to train a GMM model for each client separately.

A similar problem in biometric verification is solved by using a form of Bayesian adaptation, called Maximum A-Posteriori (MAP) [Gauvain and Lee, 1994]. In MAP, first a prior distribution over the parameters that need to be estimated is set. Then, the parameters are updated so that their posterior given their prior and the data is maximized.

First, for each client-specific GMM, a prior distribution $g(\Theta)$ over its parameters is set. A Universal Background Model (UBM), which is a GMM trained via ML on a large set of background identities, is created as well. The parameters of this GMM are denoted as Θ_{UBM}. For each client I with data X_{I}, the likelihood $p(X_{\text{I}}|\Theta_{\text{UBM}})$ is computed. Then, the parameters for each client-specific GMM are updated as given in Eq. A.4.

$$\Theta_{\text{MAP}} \leftarrow \arg\max_{\Theta} p(X_i|\Theta_{\text{UBM}})g(\Theta) \tag{A.4}$$

Depending on the choice of the prior distribution, MAP adaptation for GMM can have a closed form solution and includes a fixed parameter r called a relevance factor. The value of the relevance factor is important because it controls the impact of the client data in the adaptation of the UBM parameters to the client-specific GMM parameters. During MAP, the GMM components which are close to the client-specific data are adapted more then the components which lie further in the feature space.

B Support Vector Machines (SVM)

A Support Vector Machine (SVM) [Vapnik, 1998; Boser et al., 1992; Cristianini and Shawe-Taylor, 2000; Fornoni, 2014] is a classifier which discriminatively learns a hyperplane that separates the set $X = \{(\mathbf{x}_i, y_i) | i = 1..S\}$ of training samples in $\mathbb{R}^{d \times \{-1,1\}}$, while minimizing its generalization error on unseen samples. In its most simple form, it relies on a real valued linear function $f : \mathbb{R}^d \rightarrow \mathbb{R}$. The function $f(\mathbf{x}_i)$ is parameterized by a vector $\mathbf{w} \in \mathbb{R}^d$ and a scalar $b \in \mathbb{R}$, called a bias.

$$f(\mathbf{x}_i) = \mathbf{w} \cdot \mathbf{x}_i + b \tag{B.1}$$

The class label y_i can have two values: $\{-1, 1\}$, corresponding to the positive and the negative class. Geometrically, $\mathbf{w} \cdot \mathbf{x}_i + b = 0$ is the hyperplane which separates the two classes, where $\mathbf{w}$ is its normal and b its distance from the origin. All the points whose projection on $\mathbf{w}$ is greater or equal to $-b$ will be classified positively, while those whose projection on $\mathbf{w}$ is smaller than $-b$ will be classified negatively. In this sense, the value $-b$ can be interpreted as the threshold of the classifier. This decision rule is formalized as in Eq. B.2.

$$\hat{y}_i = \begin{cases} 1, & \text{if } \mathbf{w} \cdot \mathbf{x}_i \geq -b \\ -1, & \text{otherwise} \end{cases} \tag{B.2}$$

B.1 Maximal Margin Classifier

The geometric margin of a sample $\mathbf{x}$ is defined as the Eucledian distance between the sample and the hyperplane [Cristianini and Shawe-Taylor, 2000]. Assuming that X is linearly separable, the minimal geometric margin given in Eq. B.3 refers to the shortest among the Eucledian

distances between all the samples in one class and the hyperplane.

$$g(f)) = \min_{\mathbf{x}_i \in X} \frac{1}{\|\mathbf{w}\|} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \tag{B.3}$$

The goal of a SVM is to maximize the minimal geometric margin by finding optimal values of $\mathbf{w}$ and b. This objective function is defined in Eq. B.4 and it aims at maximizing the generalization capabilities of the classifier. It will ensure positioning of the hyperplane so that it optimally separates the two classes.

$$\mathbf{w}, b \leftarrow \arg\max_{\mathbf{w},b} \min_{\mathbf{x}_i \in X} \frac{1}{\|\mathbf{w}\|} y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \tag{B.4}$$

The objective function in Eq. B.4, is non-linear and non-convex, and thus difficult to optimize. It is therefore solved by transforming it to an equivalent formulation, where $\|\mathbf{w}\|^2$ is minimized subject to a constraint, as shown in Eq. B.5. The classifier obtained in this way is called *hard-margin* classifier, as it requires linear separability of the training data and imposes them to have a margin of 1.

$$\begin{aligned} \mathbf{w}, b &\leftarrow \arg\min_{\mathbf{w},b} \|\mathbf{w}\|^2 \\ \text{subject to } & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \end{aligned} \tag{B.5}$$

The objective function given above is usually optimized using convex optimization theory and Lagrangian multipliers [Cristianini and Shawe-Taylor, 2000]. In this context, a Lagrangian $L(\boldsymbol{\alpha})$ of the SVM objective function is derived subject to Karush-Kuhn-Tucker (KKT) optimality conditions, where $\boldsymbol{\alpha}$ is the vector of Lagrangian multipliers. The training of the SVM then sums up to minimizing $L(\boldsymbol{\alpha})$ which can be proved to have the form of Eq. B.6. The relationship between $\boldsymbol{\alpha}$ and $\mathbf{w}$ is given in Eq. B.7.

$$\begin{aligned} L(\boldsymbol{\alpha}) = \sum_u \alpha_u - \frac{1}{2} \sum_{u,v} \alpha_u \alpha_v y_u y_v \mathbf{x}_u \cdot \mathbf{x}_v \\ \text{subject to } \sum_u \alpha_u y_u = 0 \\ \alpha_u \geq 0, \forall u \end{aligned} \tag{B.6}$$

$$\mathbf{w} = \sum_u \alpha_u y_u \mathbf{x}_u \tag{B.7}$$

The representation of the optimization problem via $L(\boldsymbol{\alpha})$ is called *dual form* and depends only on the Lagrangian multipliers which act as weight coefficients for the training samples. An additional useful property of the dual form, as will be seen later, is that the training samples never appear isolated, but always within an inner product with other training samples.

B.2 Support Vectors and Classification

After the optimization procedure, only a small subset of α_u will be non-zero. In particular, only the samples for which $y_u(\mathbf{w}\cdot\mathbf{x}_u + b) = 1$ will have $\alpha_u \neq 0$. These are the samples which lie exactly on the margin of the classifier and are called *support vectors*. An SVM classifier is completely defined by its support vectors and their Lagrangian coefficients. If Eq. B.7 is substituted in Eq. B.1, a test sample $\mathbf{x}_i$ can be classified using the score obtained as in Eq. B.8.

$$f(\mathbf{x}) = \sum_u \alpha_u y_u \mathbf{x}\cdot\mathbf{x}_u + b \tag{B.8}$$

B.3 Linearly Non-separable Data

When the data from the two classes is not linearly separable, a feasible solution to the optimization problem in Eq. B.6 can not be found. Therefore, a slack variable ξ_u is incorporated into the optimization problem. The slack variable allows for a violation of the constraint $y_u(\mathbf{w}\cdot\mathbf{x}_u + b) \geq 1$ by a little amount and permits that some training samples are on the wrong side of the hyperplane. The constraint in Eq. B.5 is then reformulated to $y_u(\mathbf{w}\cdot\mathbf{x}_u + b) \geq 1 - \xi_u$. At the same time, the objective function incorporates a cost parameter C to regularize the values of ξ_u. Large C increases the penalty on the misclassified samples, forcing a hyperplane with as few misclassifications as possible. Small C restrains the impact of the misclassified samples on the hyperplane. The classifier obtained in this way is called *soft-margin* classifier.

The training of the soft-margin classifier sums up to optimizing $L(\boldsymbol{\alpha})$ as in Eq. B.6 with the additional constrain $\alpha_u \leq C$.

B.4 Kernel Functions

Often, a linear boundary may not be enough to separate the two classes in the original feature space X. A solution is to project the data into a high-dimensional feature space Φ where

they may be linearly separable and then train a SVM. Considering the dual form of $L(\boldsymbol{\alpha})$, this process is made easy by *kernel functions* which have the form given in Eq. B.9 and where $\phi : X \rightarrow \Phi$.

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \tag{B.9}$$

Kernel functions define an inner product of the samples in the high-dimensional feature space Φ. By using kernel functions, the SVM can be trained in the high-dimensional feature space without explicitly projecting the samples. This offers several advantages, like no impact of the dimensionality of Φ on the SVM training complexity and avoidance of the computational costs associated with projecting the features [Cristianini and Shawe-Taylor, 2000]. The processes of projection and SVM training can be merged together by substituting the inner product in Eq. B.6 by the kernel function. The inner product in the classification step can be substituted by a kernel function as well, as in Eq. B.10. It is important to note that in many implementations, a prerequisite for training a SVM using kernels is the computation of the kernel (Gram) matrix $\mathbf{K}_{u,v} \equiv K(\mathbf{x}_u, \mathbf{x}_v)$.

$$f(\mathbf{x}) = \sum_u \alpha_u y_u K(\mathbf{x}_i, \mathbf{x}_u) + b \tag{B.10}$$

Back in the original feature space and depending on the kernel, the boundary between the two classes may be non-linear. In these cases, the parameter C is a trade-off between misclassification of the training samples and smoothness of the boundary. Small values of C will contribute to a smooth boundary.

The most common types of kernels include:

- Linear: $K(\mathbf{x}_u, \mathbf{x}_v) = \mathbf{x}_u \cdot \mathbf{x}_v$
- Radial Basis Function (RBF): $K(\mathbf{x}_u, \mathbf{x}_v) = \exp(-\gamma \|\mathbf{x}_u - \mathbf{x}_v\|^2)$
- Polynomial: $K(\mathbf{x}_u, \mathbf{x}_v) = (\gamma \mathbf{x}_u \cdot \mathbf{x}_v + r)^d$
- Histogram Intersection: $K(\mathbf{x}_u, \mathbf{x}_v) = \sum_i \min\{x_{u,i}, x_{v,i}\}$
- χ^2: $K(\mathbf{x}_u, \mathbf{x}_v) = \exp\left(-\gamma \sum_i \frac{(x_{u,i} - x_{v,i})^2}{x_{u,i} + x_{v,i}}\right)$

The parameter γ in the RBF, polynomial and χ^2 kernel can be interpreted as the inverse of the radius of influence of the support vectors. Small values of γ mean large radius of influence, which leads to smoother boundaries. On the other hand, large values of γ may mean a boundary which is tightly adapted to the training samples.

The most important parameter for the polynomial kernel is its degree d. The larger the degree, the more flexible the decision boundary. The parameter r can be understood as a trade-off between the influence of higher-order versus lower-order terms in the polynomial.

Histogram Intersection kernel [Barla et al., 2003] and χ^2 kernel [Puzicha et al., 1997] have been specifically designed for classification when the feature vectors represent histograms. They are called additive kernels and are based on the corresponding metrics for similarities between histograms [Vedaldi and Zisserman, 2012]. As they assume that the features are histograms, they require that all the feature elements are non-negative and l_1-normalized.

Bibliography

A. Adams and M. A. Sasse. Users are not the enemy. *Communications of the ACM*, 42(12):40–46, December 1999. ISSN 0001-0782.

A. Adler and S. Schuckers. *Encyclopedia of Biometrics*, chapter Security and Liveness, Overview, pages 1146–1152. Springer-Verlag, 2009.

G. Aggarwal, N.K. Ratha, and R.M. Bolle. Biometric verification: Looking beyond raw similarity scores. In *Computer Vision and Pattern Recognition Workshop*, pages 31–31, 2006.

Z. Akhtar, G. Fumera, G-L. Marcialis, and F. Roli. Robustness analysis of likelihood ratio score fusion rule for multi-modal biometric systems under spoof attacks. In *45th IEEE International Carnahan Conference on Security Technology*, pages 237–244, 2011.

Z. Akhtar, G. Fumera, G-L. Marcialis, and F. Roli. Evaluation of serial and parallel multibiometric systems under spoofing attacks. In *5th IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2012.

F. Alegre, R. Vipperla, N. Evans, and B. Fauve. On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 36–40, 2012.

A. Ali, F. Deravi, and S. Hoque. Spoofing attempt detection using gaze colocation. In *Biometrics Special Interest Group (BIOSIG), 2013 International Conference of the*, Sept 2013.

E. Angelopoulou. Understanding the color of human skin. volume 4299, pages 243–251, 2001.

A. Anjos and S. Marcel. Counter-measures to photo attacks in face recognition: a public database and a baseline. In *International Joint Conference on Biometrics (IJCB)*, 2011.

A. Anjos, L. El-Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel. Bob: a free signal processing and machine learning toolbox for researchers. In *20th ACM Conference on Multimedia Systems (ACMMM)*. ACM Press, October 2012.

A. Anjos, M. Mohan Chakka, and S. Marcel. Motion-based counter-measures to photo attacks in face recognition. *Institution of Engineering and Technology Journal on Biometrics*, July 2013.

André Anjos, Ivana Chingovska, and Sébastien Marcel. Anti-spoofing: Face databases. In Stan Z.Li and Anil Jain, editors, *Encyclopedia of Biometrics*. Springer US, second edition edition, 2014. ISBN 978-3-642-27733-7. doi: 10.1007/978-3-642-27733-7_9212-2. URL http://link.springer.com/referenceworkentry/10.1007/978-3-642-27733-7_9067-2.

ANSI-X9.84-2010. Biometric information management and security for the financial services industry. ANSI X9.84-2010, American National Standards Institute (ANSI), 2010.

I. Armstrong. Passwords exposed: Users are the weakest link, 2003. URL http://www.scmagazine.com/passwords-exposed-users-are-the-weakest-link/article/30394/.

R. Auckenthaler, M. J. Carey, and H. Lloyd-Thomas. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10(1-3):42–54, 2000.

J. Bai, T. Ng, X. Gao, and Y. Shi. Is physics-based liveness detection truly possible with a single image? In *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2010.

E. Bailly-Baillire, S. Bengio, F. Bimbot, M. Hamouz, J. Mariéthoz, J. Matas, F. Porée, B. Ruiz, and J.-P. Thiran. The BANCA database and evaluation protocol. In *In Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA03*, 2003.

W. Bao, H. Li, N. Li, and W. Jiang. A liveness detection method for face recognition based on optical flow field. *2009 International Conference on Image Analysis and Signal Processing*, 2009a.

W. Bao, H. Li, N. Li, and W. Jiang. A liveness detection method for face recognition based on optical flow field. *2009 International Conference on Image Analysis and Signal Processing*, pages 223–236, 2009b.

A. Barla, F. Odone, and A. Verri. Histogram intersection kernel for image classification. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3, pages III–513–16 vol.2, Sept 2003.

S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz. Fusion of face and speech data for person identity verification. *IEEE Transactions on Neural Networks*, 10:1065–1075, 1999.

S. Bengio, J. Mariéthoz, and M. Keller. The expected performance curve. In *International Conference on Machine Learning, ICML, Workshop on ROC Analysis in Machine Learning*, 2005.

S. Bharadwaj, T.I. Dhamecha, M. Vatsa, and R. Singh. Computationally efficient face spoofing detection with motion magnification. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 105–110, June 2013.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *Proceedings of the 2012 IEEE Symposium on Security and Privacy*, pages 553–567. IEEE Computer Society, 2012.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152. ACM, 1992. ISBN 0-89791-497-X.

F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of mlp and gmm classifiers for face verification on xm2vts. In *Proceedings of the 4th International Conference on AVBPA*, University of Surrey, Guildford, UK, 2003.

F. Cardinaux, C. Sanderson, and S. Bengio. User authentication via adapted statistical models of face images. *Signal Processing, IEEE Transactions on*, 54(1):361–373, Jan 2006.

ChaosComputerClub. Chaos computer club breaks apple touchid, 2013. URL http://www.ccc.de/updates/2013/ccc-breaks-apple-touchid.

G. Chetty and M. Wagner. Multi-level liveness verification for face-voice biometric authentication. In *In Biometrics symposium 2006*, 2006a.

G. Chetty and M. Wagner. Audio-visual multimodal fusion for biometric person authentication and liveness verification. In *Proceedings of the 2005 NICTA-HCSNet Multimodal User Interaction Workshop - Volume 57*, pages 17–24. Australian Computer Society, Inc., 2006b.

I. Chingovska and A. Anjos. On the use of client identity information for face anti-spoofing. *IEEE Transactions on Information Forensics and Security, Special Issue on Biometric Anti-spoofing*, 2015.

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

I. Chingovska, A. Anjos, and S. Marcel. Anti-spoofing in action: joint operation with a verification system. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Biometrics*, June 2013a.

I. Chingovska, J. Yang, Z. Lei, D. Yi, S. Z. Li, O. Kähm, N. Damer, C. Glaser, A. Kuijper, A. Nouak, J. Komulainen, T. de Freitas Pereira, S. Gupta, S. Bansal, S. Khandelwal, A. Rai, T. Krishna, D. Goyal, M.-A. Waris, H. Zhang, I. Ahmad, S. Kiranyaz, M. Gabbouj, R. Tronci, M. Pili, N. Sirena, F. Roli, J. Galbally, J. Fierrez-Aguilar, A. Pinto, H. Pedrini, W. R. Schwartz, A. Rocha, A. Anjos, and S. Marcel. The 2nd competition on counter measures to 2d face spoofing attacks. In *International Conference of Biometrics (ICB)*, 2013b.

I. Chingovska, A. Anjos, and S. Marcel. Biometrics evaluation under spoofing attacks. *IEEE Transactions on Information Forensics and Security*, 9(12):2264–2276, December 2014a.

Ivana Chingovska, André Anjos, and Sébastien Marcel. Anti-spoofing: Evaluation methodologies. In Stan Z.Li and Anil Jain, editors, *Encyclopedia of Biometrics*. Springer US, 2nd edition edition, 2014b. ISBN 978-3-642-27733-7. doi: 10.1007/978-3-642-27733-7.

Ivana Chingovska, André Anjos, and Sébastien Marcel. Evaluation methodologies. In Sébastien Marcel, Mark Nixon, and Stan Z.Li, editors, *Handbook of Biometric Antispoofing*. Springer, 2014c. ISBN 978-1-4471-6523-1.

Baris Coskun and Cormac Herley. Can "something you know" be saved? In T.-C. Wu, C.-L. Lei, V. Rijmen, and D.-T. Lee, editors, *Information Security*, volume 5222 of *Lecture Notes in Computer Science*, pages 421–440. Springer Berlin Heidelberg, 2008.

N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-78019-5.

Nguyen M. D. and Bui Q. M. Your face is not your password. Black Hat Conference, 2009.

A. da Silva Pinto, H. Pedrini, W. Robson Schwartz, and A. Rocha. Video-based face spoofing detection through visual rhythm analysis. In *25th Conference on Graphics, Patterns and Images*, 2012.

N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, June 2005.

W. H. David and L. Stanley. *Applied Logistic Regression*. Wiley-Interscience Publication, 2000.

M. De Marsico, M. Nappi, D. Riccio, and J. Dugelay. Moving face spoofing detection via 3d projective invariants. In *Biometrics (ICB), 2012 5th IAPR International Conference on*, March 2012.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, 39(1):1–38, 1977.

G. Doddington, W. Ligget, A. Martin, M. Przybocki, and D. Reynolds. SHEEP, GOATS, LAMBS and WOLVES: A statistical analysis of speaker performance in the NIST 1998 speaker recognition evaluation. In *International Conference On Spoken Language Processing*, 1998.

N. Erdogmus and S. Marcel. Spoofing attacks to 2d face recognition systems with 3d masks. In *International Conference of the Biometrics Special Interes Group*, September 2013a.

N. Erdogmus and S. Marcel. Spoofing in 2d face recognition with 3d masks and anti-spoofing with kinect. In *Biometrics: Theory, Applications and Systems*, September 2013b.

N. Erdogmus and S. Marcel. Spoofing face recognition with 3d masks. *IEEE Transactions on Information Forensics and Security*, 9(7):1084–1097, July 2014a.

N. Erdogmus and S. Marcel. Introduction. In *Handbook of Biometric Anti-Spoofing - Trusted Biometrics under Spoofing Attacks*, pages 1–11. 2014b.

J. Fierrez-Aguilar, J. Ortega-Garcia, D. Garcia-Romero, and J. Gonzalez-Rodriguez. A comparative evaluation of fusion strategies for multimodal biometric verification. In *Fourth International Conference on Audio- and Video-based Biometric Person Authentication*, pages 830–837, 2003.

J. Fierrez-Aguilar, D. Garcia-Romero, J. Ortega-Garcia, and J. Gonzalez-Rodriguez. Adapted user-dependent multimodal biometric authentication exploiting general information. *Pattern Recognition Letters*, 26(16):2628–2639, December 2005.

Y. Flink. Million dollar border security machines fooled with ten cent tape, 2009. URL http://findbiometrics.com/million-dollar-border-security-machines-fooled-with-ten-cent-tape/.

D. Florencio and C. Herley. A large-scale study of web password habits. In *Proceedings of the 16th International Conference on World Wide Web*, pages 657–666. ACM, 2007.

M. Fornoni. MathematicaSVM - a hands-on introduction to Support Vector Machines using Mathematica ©. https://github.com/fornoni/MathematicaSVM, 2014.

R.W. Frischholz and A. Werner. Avoiding replay-attacks in a face recognition system using head-pose estimation. In *Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop on*, Oct 2003.

J. Galbally, J. Fierrez-Aguilar, J. D. Rodriguez-Gonzalez, F. Alonso-Fernandez, Javier Ortega-Garcia, and M. Tapiador. On the vulnerability of fingerprint verification systems to fake fingerprints attacks. In *IEEE International Carnahan Conference on Security Technology*, pages 169–179, 2006.

J. Galbally, J. Fierrez-Aguilar, and J. Ortega-Garcia. Vulnerabilities in biometric systems: attacks and recent advances in liveness detection. In *Proc. Spanish Workshop on Biometrics, SWB*, June 2007.

J. Galbally, R. Cappellib, A. Luminib, G. Gonzalez de Rivera, D. Maltoni, J. Fierrez-Aguilar, J. Ortega-Garcia, and D. Maio. An evaluation of direct attacks using fake fingers generated from ISO templates. *Pattern Recognition Letters*, 31(8):725–732, 2010.

J. Galbally, F. Alonso-Fernandez, J. Fierrez-Aguilar, and J. Ortega-Garcia. A high performance fingerprint liveness detection method based on quality related features. *Future Gener. Comput. Syst.*, 28(1):311–321, 2012.

J. Galbally, S. Marcel, and J. Fierrez-Aguilar. Image quality assessment for fake biometric detection: Application to iris, fingerprint and face recognition. *IEEE Trans. on Image Processing*, 23(2):710–724, February 2014.

X. Gao, T.-T. Ng, Q. Bo, and S.-F. Chang. Single-view recaptured image detection based on physics-based features. In *IEEE International Conference on Multimedia & Expo (ICME)*, July 2010.

J. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *Speech and Audio Processing, IEEE Transactions on*, 2(2): 291–298, Apr 1994.

M. Günther, D. Haufe, and R. P. Würtz. Face recognition with disparity corrected Gabor phase differences. In *Artificial Neural Networks and Machine Learning*, volume 7552 of *Lecture Notes in Computer Science*, pages 411–418. Springer Berlin, 2012.

M. Günther, R. Wallace, and S. Marcel. An open source framework for standardized comparisons of face recognition algorithms. In Andrea Fusiello, Vittorio Murino, and Rita Cucchiara, editors, *Computer Vision - ECCV 2012. Workshops and Demonstrations*, volume 7585 of *Lecture Notes in Computer Science*, pages 547–556, October 2012.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer New York Inc., 2001.

ISO-30107-1. Biometric presentation attack detection – part 1: Framework. ISO 30107-1, International Organization for Standardization (ISO), Geneva, Switzerland, 2014.

ISO/IEC 15408. Common Criteria (CC) for Information Technology Security Evaluation - Evaluation Methodology, 2012. URL http://www.commoncriteriaportal.org/cc/.

A. K. Jain and A. Ross. *Handbook of Biometrics*, chapter Introduction to Biometrics. Springer-Verlag, 2008.

A. K. Jain, P. Flynn, and A. Ross, editors. *Handbook of Biometrics*. Springer-Verlag, 2008.

A.K. Jain and A. Ross. Learning user-specific parameters in a multibiometric system. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–57–I–60 vol.1, 2002. doi: 10.1109/ICIP.2002.1037958.

A.K. Jain, A. Ross, and S. Pankanti. Biometrics: a tool for information security. *Information Forensics and Security, IEEE Transactions on*, 1(2):125–143, June 2006.

P. Johnson, R. Lazarick, E. Marasco, E. Newton, A. Ross, and S. Schuckers. Biometric liveness detection: Framework and metrics. In *International Biometric Performance Conference*, 2012.

P. A. Johnson, B. Tan, and S. Schuckers. Multimodal fusion vulnerability to non-zero (spoof) imposters. In *IEEE International Workshop on Information Forensics and Security*, 2010.

K. Jonsson, J. Kittler, Y. P. Li, and J. Matas. Support vector machines for face authentication. In *Image and Vision Computing*, pages 543–553, 1999.

S. S. Kajarekar and A. Stolcke. NAP and WCCN: Comparison of approaches using MLLR-SVM speaker verification system. In *ICASSP (4)*, pages 249–252, 2007.

S. Kim, S. Yu, K. Kim, Y. Ban, and S. Lee. Face liveness detection using variable focusing. In *Biometrics (ICB), 2013 International Conference on*, June 2013.

Y. Kim, J. Na, S. Yoon, and J. Yi. Masked fake face detection using radiance measurements. *Journal of the Optical Society of America A*, 26(4):760–766, Apr 2009.

J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, Mar 1998.

K. Kollreider, H. Fronthaler, and J. Bigun. Verifying liveness by multiple experts in face biometrics. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, pages 1–6, June 2008.

K. Kollreider, H. Fronthaler, and J. Bigun. Non-intrusive liveness detection by face images. *Image and Vision Computing*, 27(3):233–244, 2009.

J. Komulainen, A. Hadid, and M. Pietikainen. Context based face anti-spoofing. In *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, pages 1–8, Sept 2013a.

J. Komulainen, A. Hadid, M. Pietikainen, A. Anjos, and S. Marcel. Complementary countermeasures for detecting scenic face spoofing attacks. In *International Conference on Biometrics*, June 2013b.

N. Kose and J.-L. Dugelay. Countermeasure for the protection of face recognition systems against mask attacks. In *International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–6. IEEE, 2013.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

W. Kruskal and W. A. Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47(260):1–1, 1952.

A. Kumar and D. Zhang. User authentication using fusion of face and palmprint. *International Journal of Image and Graphics*, 09(02):251–270, 2009.

L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, May 2003.

J. Li, Y. Wang, T. Tan, and A. K. Jain. Live face detection based on the analysis of fourier spectra. *Biometric Technology for Human Identification*, 2004.

Yan Li, K. Xu, Q. Yan, Y. Li, and R. H. Deng. Understanding OSN-based facial disclosure against face authentication systems. In *9th ACM Symposium on Information, Computer and Communications Security*, pages 413–424, 2014.

S. Lloyd. Least squares quantization in pcm. *IEEE Trans. Inf. Theor.*, 28(2):129–137, 9 1982. ISSN 0018-9448.

S. Lucey and T. Chen. A gmm parts based face representation for improved verification through relevance adaptation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–855–II–861 Vol.2, June 2004.

Y. M. Lui, D. Bolme, P.J. Phillips, J.R. Beveridge, and B.A. Draper. Preliminary studies on the Good, the Bad, and the Ugly face recognition challenge problem. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012*, pages 9–16, 2012.

J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using micro-texture analysis. In *International Joint Conference on Biometrics*, pages 1–7, 2011.

J. Määttä, A. Hadid, and M. Pietikäinen. Face spoofing detection from single images using texture and local shape analysis. *IET Biometrics*, 1:3–10, 2012.

J. Macqueen. Some methods for classification and analysis of multivariate observations. In *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

A. J. Mansfield, J. L. Wayman, D. Rayner, and J. L. Wayman. Best practices in testing and reporting performance, 2002.

E. Marasco, P. Johnson, C. Sansone, and S. Schuckers. Increase the security of multibiometric systems by incorporating a spoofing detection algorithm in the fusion mechanism. In *Proceedings of the 10th international conference on Multiple classifier systems*, pages 309–318, 2011.

E. Marasco, Y. Ding, and A. Ross. Combining match scores with liveness values in a fingerprint verification system. In *5th IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2012.

S. Marcel, Y. Rodriguez, and G. Heusch. On the recent use of local binary patterns for face authentication. *International Journal on Image and Video Processing, SI on Facial Image Processing*, 2007.

A. Martin and M. Przybocki. The NIST 1999 speaker recognition evaluation - an overview, 2000.

A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki. The DET curve in assessment of detection task performance. In *Eurospeech*, pages 1895–1898, 1997.

T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino. Impact of artificial "gummy" fingers on fingerprint systems. volume 4677. SPIE, 2002.

Stephen M. Matyas Jr. and J. Stapleton. A biometric standard for information management and security. *Computers and Security*, 19(5):428 – 441, 2000.

C. Matyszczyk. Doctors 'used fake fingers' to clock in for colleagues at ER, 2013. URL http://www.cnet.com/news/doctors-used-fake-fingers-to-clock-in-for-colleagues-at-er/.

M. McLaren, R. Vogt, B. Baker, and S. Sridharan. Data-driven background dataset selection for SVM-based speaker verification. *IEEE Transactions on Audio, Speech and Language Processing*, 18(6):1496 – 1507, 2010.

M. L. McLaren. *Improving automatic speaker verification using SVM techniques*. PhD thesis, Queensland University of Technology, 2009.

D. Menotti, G. Chiachia, A. Pinto, W. Schwartz, H. Pedrini, A. Falcao, and A. Rocha. Deep representations for iris, face, and fingerprint spoofing detection. *Information Forensics and Security, IEEE Transactions on*, (99):1–1, 2015.

Radford M. N. and G. E. Hinton. A new view of the em algorithm that justifies incremental and other variants. In *Learning in Graphical Models*, pages 355–368. Kluwer Academic Publishers, 1993.

L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040, Dec 2003.

T. Ojala, M. Pietikäinen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971 –987, July 2002.

M. Oren and S. K. Nayar. Generalization of the lambertian model and implications for machine vision. *International Journal of Computer Vision*, 14(3):227–251, 1995.

G. Pan, L. Sun, Z. Wu, and S. Lao. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *ICCV*. IEEE, 2007.

G. Pan, L. Sun, Z. Wu, and Y. Wang. Monocular camera-based face liveness detection by combining eyeblink and scene context. *Telecommunication Systems*, 47(3-4):215–225, 2011.

G. Parziale, J. Dittman, and M. Tistarelli. Analysis and evaluation of alternatives and advanced solutions for system elements. BioSecure D 9.1.2, 2005.

I. Pavlidis and P. Symosek. The imaging issue in an automatic face/disguise detection system. In *Computer Vision Beyond the Visible Spectrum: Methods and Applications, 2000. Proceedings. IEEE Workshop on*, pages 15–24, 2000.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

T. De Freitas Pereira, A. Anjos, J. M. De Martino, and S. Marcel. Can face anti-spoofing countermeasures work in a real world scenario? In *International Conference on Biometrics*, 2013.

T. De 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*, 2014:2, 2014.

N. Poh and S. Bengio. Database, protocols and tools for evaluating score-level fusion algorithms in biometric authentication. *Pattern Recognition Journal*, 39:223–233, 2006.

Norman Poh and Josef Kittler. On the use of log-likelihood ratio based model-specific score normalisation in biometric authentication. In S.-W. Lee and S. Z. Li, editors, *Advances in Biometrics*, volume 4642 of *Lecture Notes in Computer Science*, pages 614–624. 2007.

F. J. Prokoski. Disguise detection and identification using infrared imagery. volume 0339, pages 27–31, 1983.

M.A. Przybocki, A.F. Martin, and A.N. Le. NIST speaker recognition evaluation chronicles - part 2. In *Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006: The*, June 2006.

J. Puzicha, T. Hofmann, and J.M. Buhmann. Non-parametric similarity measures for unsupervised texture segmentation and image retrieval. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 267–272, Jun 1997.

R. Raghavendra, K.B. Raja, and C. Busch. Presentation attack detection for face recognition using light field camera. *Image Processing, IEEE Transactions on*, (99):1–1, 2015.

NaliniK. Ratha, JonathanH. Connell, and RuudM. Bolle. An analysis of minutiae matching strength. In J. Bigun and F. Smeraldi, editors, *Audio- and Video-Based Biometric Person Authentication*, volume 2091 of *Lecture Notes in Computer Science*, pages 223–228. Springer Berlin Heidelberg, 2001.

A. Rattani and N. Poh. Biometric system design under zero and non-zero effort attacks. In *Biometrics (ICB), 2013 International Conference on*, pages 1–8, June 2013.

A. Rattani, N. Poh, and A. Ross. Analysis of user-specific score characteristics for spoof biometric attacks. In *CVPR Workshops*, pages 124–129. IEEE, 2012.

A. Rattani, N. Poh, and A. Ross. A bayesian approach for modeling sensor influence on quality, liveness and match score values in fingerprint verification. In *Information Forensics and Security (WIFS), 2013 IEEE International Workshop on*, pages 37–42, Nov 2013.

D. A. Reynolds. Speaker identification and verification using gaussian mixture speaker models. *Speech Communication*, 17(1–2):91 – 108, 1995.

D. A. Reynolds and R. C. Rose. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83, 1995.

D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. In *Digital Signal Processing*, page 2000, 2000.

R. N. Rodrigues, L. L. Ling, and V. Govindaraju. Robustness of multimodal biometric fusion methods against spoofing attacks. *Journal of Visual Languages and Computing*, 20(3): 169–179, 2009.

R.N. Rodrigues, N. Kamat, and V. Govindaraju. Evaluation of biometric spoofing in a multimodal system. In *Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on*, 2010.

F. Roli, J. Kittler, G. Fumera, and D. Muntoni. An experimental comparison of classifier fusion rules for multimodal personal identity verification systems. In *Proceedings of the Third International Workshop on Multiple Classifier Systems*, pages 325–336, 2002.

A.E. Rosenberg and S. Parthasarathy. Speaker background models for connected digit password speaker verification. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 1, pages 81–84 vol. 1, May 1996.

A. Ross and A. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24, 2003.

A. Ross, K. Nandakumar, and A. K. Jain. *Handbook of Biometrics*, chapter Introduction to multibiometrics. Springer-Verlag, 2008.

V. Ruiz-Albacete, P. Tome, F. Alonso-Fernandez, J. Galbally, J. Fierrez-Aguilar, and J. Ortega-Garcia. Direct attacks using fake images in iris verification. In *Proc. COST 2101 Workshop on Biometrics and Identity Management, BIOID*, pages 181–190. Springer, May 2008.

C. Sanderson and K. K. Paliwal. Fast features for face authentication under illumination direction changes. *Pattern Recognition Letters*, 24(14), 0 2003. doi: 10.1016/s0167-8655(03)00070-9.

P. J. SCHMID. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 8 2010. ISSN 1469-7645.

Stephanie A. C. Schuckers. Spoofing and anti-spoofing measures. *Information Security Technical Report*, 7:56–62, 2002.

W. R. Schwartz, A. Rocha, and H. Pedrini. Face Spoofing Detection through Partial Least Squares and Low-Level Descriptors. In *International Joint Conference on Biometrics*, 2011.

D. Simon-Zorita, J. Ortega-Garcia, M. Sanchez-Asenjo, and J. Gonzalez-Rodriguez. Facing position variability in minutiae-based fingerprint verification through multiple references and score normalization techniques. In *Audio- and Video-Based Biometric Person Authentication*, volume 2688 of *Lecture Notes in Computer Science*, pages 214–223. 2003.

N. Swanner. Samsung galaxy s5 fingerprint scanner foiled, device hacked, 2014. URL http://androidcommunity.com/samsung-galaxy-s5-fingerprint-scanner-foiled-device-hacked-20140415/.

X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *Image Processing, IEEE Transactions on*, 19(6):1635–1650, June 2010.

X. Tan, Y. Li, J. Liu, and L. Jiang. Face liveness detection from a single image with sparse low rank bilinear discriminative model. In *ECCV (6)*, pages 504–517, 2010.

S. Tirunagari, N. Poh, D. Windridge, A. Iorliam, N. Suki, and A.T.S. Ho. Detection of face spoofing using visual dynamics. *Information Forensics and Security, IEEE Transactions on*, 10(4):762–777, April 2015.

K.-A. Toh, X. Jiang, and W.-Y. Yau. Exploiting global and local decisions for multimodal biometrics verification. *Signal Processing, IEEE Transactions on*, 52(10):3059–3072, Oct 2004.

R. Tronci, D. Muntoni, G. Fadda, M. Pili, N. Sirena, G. Murgia, M. Ristori, and F. Roli. Fusion of multiple clues for photo-attack detection in face recognition systems. In *International Joint Conference of Biometrics (IJCB)*, pages 1–6, 2011.

S. Tulyakov, Z. Zhang, and V. Govindaraju. Comparison of combination methods utilizing t-normalization and second best score model. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, pages 1–5, June 2008.

V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

M. Vatsa, R. Singh, A. Noore, and A. Ross. On the dynamic selection of biometric fusion algorithms. *IEEE Transactions on Information Forensics and Security*, 5(3):470–479, 2010.

A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(3):480–492, March 2012.

J. Villalba and E. Lleida. Preventing replay attacks on speaker verification systems. In *Security Technology (ICCST), 2011 IEEE International Carnahan Conference on*, pages 1–8, 2011.

R. Wallace, M. McLaren, C. McCool, and S. Marcel. Inter-session variability modelling and joint factor analysis for face authentication. In *International Joint Conference on Biometrics*, 2011.

L. Wang, X. Ding, and C. Fang. Face live detection method based on physiological motion analysis. *Tsinghua Science and Technology*, 14(6):685–690, 2009.

T. Wang, J. Yang, Z. Lei, S. Liao, and S. Z. Li. Face liveness detection using 3D structure recovered from a single camera. In *Biometrics (ICB), 2013 International Conference on*, 2013a.

Y. Wang, T. Tan, and A. Jain. Combining face and iris biometrics for identity verification. In *Fourth International Conference on Audio- and Video-based Biometric Person Authentication*, 2003.

Y. Wang, X. Hao, Y. Hou, and C. Guo. A new multispectral method for face liveness detection. In *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*, pages 922–926, Nov 2013b.

D. Wen, H. Han, and A. K. Jain. Face spoof detection with image distortion analysis. *IEEE Transactions on Information Forensics and Security*, PP(99):1–1, 2015.

C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, pages 682–688. MIT Press, 2001.

L. Wiskott, J.-M. Fellous, N. Krüger, and C. Von Der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis And Machine Inteligence*, 19: 775–779, 1997.

H.-Y. Wu, M. Rubinstein, E. Shih, J. Guttag, F. Durand, and W. T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Trans. Graph. (Proceedings SIGGRAPH 2012)*, 31(4), 2012.

J. Yan, Z. Zhang, Z. Lei, D. Yi, and S. Z. Li. Face liveness detection by exploring multiple scenic clues. In *12th International Conference on Control, Automation, robotics and Vision (ICARCV 2012)*, China, December 2012.

J. Yang, Z. Lei, S. Liao, and S.Z. Li. Face liveness detection with component dependent descriptor. In *International Conference on Biometrics (ICB)*, pages 1–6, June 2013.

J. Yang, Z. Lei, and S. Z. Li. Learn convolutional neural network for face anti-spoofing. *CoRR*, abs/1408.5601, 2014.

W. Zhang, S. Shan, and W. Gao ; X. Chen ; H. Zhang. Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 - Volume 01*, ICCV '05, pages 786–791. IEEE Computer Society, 2005.

Z. Zhang, D. Yi, Z. Lei, and S. Z. Li. Face liveness detection by learning multispectral reflectance distributions. pages 436–441, 2011.

G. Zhao and M. Pietikäinen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):915–928, 2007.

Z. Zhiwei, J. Yan, S. Liu, Z. Lei, D. Yi, and S. Z. Li. A face antispoofing database with diverse attacks. In *5th IAPR International Conference on Biometrics (ICB)*, pages 26–31, March 2012.

Ivana Chingovska

Nationality: Macedonian

Pré-de-Foire, 3
1920 – Martigny
Switzerland

http://ivana7c.github.io ivana.cingovska@gmail.com Tel. +41 76 640 6820

- Expert in Biometrics and Biometric Security
- Background in Computer Science and Electrical Engineering
- Experience in Applied Machine Learning and Data Analysis

Education

PhD in Electrical Engineering (ongoing) - Ecole Polytechnique Fédérale de Lausanne (EPFL) – Switzerland
Idiap Research Institute – Martigny, Switzerland

11/2011 - 11/2015 (expected)

Topics: biometrics (face mode), biometric anti-spoofing and security, machine learning, computer vision
Achievements: methods for improving the robustness of face recognition systems to spoofing attacks
method for evaluation of biometric recognition systems under spoofing attacks

MSc. in Electrical Engineering and Information Technologies (10/10) - Faculty of Electrical Engineering and Information Technologies - Skopje, R. Macedonia

2008 - 2011

Topics: content-based search and retrieval, bioinformatics, machine learning, graph analysis
Achievements: methods for protein function prediction based on protein interaction networks
descriptors of 3D protein structure for efficient protein retrieval

BSc. in Electrical Engineering and Information Technologies (10/10) - Faculty of Electrical Engineering and Information Technologies - Skopje, R. Macedonia

2004 – 2008

Topics: programming, software engineering, databases, operating systems, calculus, linear algebra, intelligent systems, computer architectures

Experience

Data Science Fellow – Science 2 Data Science (S2DS) training program – London, UK

08/2015 – 09/2015

Topics: applied machine learning, text processing
Achievements: automatized the data classification process for the Almanac of the National Council for Voluntary Organizations (NCVO)

Junior Teaching Assistant – Faculty of Electrical Engineering and Information Technologies – Skopje, R. Macedonia

03/2009 – 08/2011

Duties: material preparation and lectures on algorithms, data structures, functional programming

Research Intern - School of Computer Science, Cardiff University – Cardiff, Wales, UK

06/2010 - 08/2010

Topics: computer vision, dynamic programming
Achievements: developed video time warping technique for active appearance models

Research Assistant – Digital Image Processing Team, Faculty of Electrical Engineering and Information Technologies – Skopje, R. Macedonia

12/2008 – 06/2010

Topics: computer vision, applied machine learning
Achievements: developed a hierarchical method for image orientation detection

Research Intern – Image Processing Group, Universidad Politécnica de Madrid – Madrid, Spain

07/2009 – 09/2009

Topics: computer vision, applied machine learning
Achievements: developed a method for automatic traffic signs recognition

Research Intern – VisLab, Institute for Systems and Robotics, Instituto Superior Técnico - Lisbon, Portugal

07/2008 – 09/2008

Topics: computer vision
Achievements: developed object tracking method for robotic vision

Software Development Intern - Netcetera doo. - Skopje, R. Macedonia

08/2007 - 10/2007

Topics: web development
Achievements: added and/or optimized several functionalities of company's internal web application

Professional activities

Bob: a free signal processing and machine learning toolbox: contributor (2012 - present)

The 2nd competition on counter measures to 2D facial spoofing attacks: main organizer and participant (2013)

2010 ACM Southeastern Europe Programming Contest – Bucharest, Romania: member of jury

International Association for the Exchange of Students for Technical Experience (IAESTE) – R. Macedonia: member (2007-2011)

Technical skills

Scientific skills: face biometrics, biometric anti-spoofing, computer vision, applied machine learning, statistics, algorithms and data structures, technical writing

Development & Scripting: Python, C/C++, Unix shell scripting, Matlab, SQL, Java, Prolog, LISP

OS: Linux, Windows

Tools: LaTeX, Git, SVN, SUN Grid Engine, Virtualbox

Certificates and awards

Certificate for considerable engagement and achievements in the academic year – Faculty of Electrical Engineering and Information Technologies – Skopje, R. Macedonia (years 2005, 2006, 2007, 2008)

Scholarship awarded to talented undergraduate students of computer science - Government of R. Macedonia (2007)

Personal skills and interests

Languages: Macedonian (mother tongue), English (fluent C1), French (intermediate B1), German (basic A2)

Social skills: analytical, proactive, team-oriented, flexible, communicative, empathetic, supportive, good presentation and oratorical skills, good management skills, leadership skills

Personal interests: nature, health and fitness, society, traveling and new cultures

Publications

In journals:

I. Chingovska, A. Anjos; On the use of client-specific information for face anti-spoofing; IEEE Transactions on Information Forensics and Security; Special Issue on Biometric Anti-spoofing; 10(4):787-796, 2015

I. Chingovska, A. Anjos, S. Marcel; Biometric evaluation under spoofing attacks; IEEE Transactions on Information Forensics and Security, 9(12):2264-2276, 2014

G. Mirceva, **I. Cingovska**, Z. Dimov, D. Davcev; Efficient Approaches for Retrieving Protein Tertiary Structures; Transactions on Computational Biology and Bioinformatics; 9(4):1166-79, 2012

In conference proceedings:

I. Chingovska, A. Anjos, S. Marcel; Anti-spoofing in action: joint operation with verification system; Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Biometrics; Portland, Oregon, 2013

I. Chingovska, et al.; The 2nd competition on counter measures to 2D facial spoofing attacks; Proceedings of International Conference on Biometrics (ICB); Madrid, Spain, 2013

I. Chingovska, A. Anjos, S. Marcel; On the Effectiveness of Local Binary Patterns in Face Anti-spoofing; Proceedings of the 11th International Conference of the Biometrics Special Interes Group; Darmstadt, Germany 2012

I. Cingovska, Z. Ivanovski, F. Martin; Automatic Image Orientation Detection with Prior Hierarchical Content-Based Classification; Proceedings of International Conference on Image Processing (ICIP); Brussels, Belgium, 2011

A. Aubrey, V. Kajic, **I. Cingovska,** P. Rosin, D. Marshall; Mapping and Manipulating Facial Dynamics; Proceedings of International Conference on Automatic Face and Gesture Recognition; Santa Barbara, California, 2011

K. Trivodaliev, **I. Cingovska**, S. Kalajdziski, D. Davcev; Protein Function Prediction Based on Neighborhood Profiles; Proceedings of ICT Innovations; Ohrid, R. Macedonia, 2009

I. Cingovska, G. Mirceva, Z. Dimov, S. Kalajdziski, D. Davcev; Novel Wavelet Based Protein Descriptors; Proceedings of ICT Innovations; Ohrid, R. Macedonia, 2009

Book chapters:

I. Chingovska, A. Anjos, S. Marcel; Anti-spoofing: Evaluation Methodologies; Encyclopedia of Biometrics; Springer US, 2014

I. Chingovska, A. Anjos, S. Marcel; Anti-spoofing: Face Databases; Encyclopedia of Biometrics; Springer, US, 2014

I. Chingovska, A. Anjos, S. Marcel; Evaluation Methodologies; Handbook of Biometric Anti-spoofing; Springer London, pp.185-2014, 2014

K. Trivodaliev, **I.Cingovska**, S. Kalajdziski; Protein Function Prediction by Spectral Clustering of Protein Interaction Networks; Database Theory and Application, Bio-Science and Bio-Technology, Edition: Communications in Computer and Information Science Volume 258, Publisher: Springer Berlin Heidelberg, pp.108-117, 2011