On the use of client identity information for face anti-spoofing

Ivana Chingovska, André Anjos

With biometrics playing the role of a password which can not be replaced if stolen, the necessity of establishing counter-measures to biometric spoofing attacks has been recognized. Regardless of the biometric mode, the typical approach of anti-spoofing systems is to classify biometric evidence based on features discriminating between real accesses and spoofing attacks. For the first time, to the best of our knowledge, this paper studies the amount of client-specific information within these features and how it affects the performance of anti-spoofing systems. We make use of this information to build two client-specific anti-spoofing solutions, one relying on a generative and another one on a discriminative paradigm. The proposed methods, tested on a set of state-of-the-art anti-spoofing features for the face mode, outperform the client-independent approaches with up to 50% relative improvement and exhibit better generalization capabilities on unseen types of spoofing attacks.

***Index Terms*—Spoofing Attack, Counter-Measures, Counter-Spoofing, Biometric Verification, Liveness Detection, Replay**

I. Introduction

The wide deployment of biometric recognition systems in the recent years has been hindered by the discovery of their vulnerability to spoofing attacks: attempts to access the system by presenting a copy of the biometric trait of a user. Depending on the biometric mode, manufacturing such copies may be a matter of a few mouse clicks. For example, for face biometrics, all that is required is downloading and printing a user's photograph from the Internet: task which has become strikingly easy in the era of information globalization and social networks [1]. While more effort and skills may be required to spoof other biometric modes, recent security trials of commercial biometric authentication systems on mobile devices have shown that the vulnerability exists and can be easily exploited [2].

As noted by D. Denning [3], "It's liveness, not secrecy, that counts". The fact that the biometric traits can not be kept secret should not be an obstacle for using biometrics. Such a reasoning has inspired an ever increasing number of liveness detection and anti-spoofing algorithms for many biometric modes. Some of them utilize a specific hardware device ensuring the presence of a living person in front of the system. Others combine multiple modalities, presuming that this increases the difficulty of spoofing the system [4]. Among systems which depend on additional hardware or require user interaction, software-based solutions which use only the evidence taken by the biometric sensor may be the most favorable due to their inexpensiveness and convenience of use [5].

A point which is very often overlooked is that anti-spoofing systems are designated to protect biometric recognition systems and as such need to work jointly with them. One aspect of this cooperation may be sharing information which is available to either the recognition or the anti-spoofing system. One example is the identity of the client who needs to be recognized, which is an essential information for the recognition system. In a verification scenario, a client claims an identity and the system uses this information to match the input sample against a stored model. In an identification scenario, the input sample is matched against several stored models whose identities are known. Nevertheless, the anti-spoofing systems rarely make use of this information.

Usually, the anti-spoofing systems are designed as binary classifiers whose task is to discriminate between real access and spoofing attack samples, with no regards to the client identity. They are based on the assumption that there is no critical difference between the spoofing attacks performed against different clients. Even more, they disregard the variations between the real access samples coming from different clients.

One may argue that the features that anti-spoofing systems use, need to be selected in such a manner that they capture the intrinsic differences between real accesses and spoofing attacks, rendering them unconcerned of the client identities. Indeed, the face anti-spoofing features proposed in the literature take into account several aspects of disparity between real accesses and spoofing attacks, like texture, quality, motion patterns etc., and they show great discrimination capabilities between the two classes of samples. This, however, does not exclude the possibility that the extracted anti-spoofing features are influenced by the characteristics of the individual clients and may retain some client-specific information. This information may be useful to make better discrimination between the real accesses and spoofing attacks of a particular client.

The contribution of this paper is three-fold. Firstly, we investigate to what extent the face anti-spoofing features

Ivana Chingovska is with Idiap Research Institute and Ecole Polytechnique Fédérale de Lausanne, Switzerland, e-mail: ivana.chingovska@idiap.ch

André Anjos is with the Idiap Research Institute, Switzerland, e-mail: andre.anjos@idiap.ch

The authors would like to thank Dr. Sébastien Marcel from Idiap Research Institute for the fruitful discussions and support during the work on this paper. The authors would also like to thank the FP7 European TABULA RASA (257289) and BEAT (284989) projects for their financial support.

encompass client-specific information and whether this influences the performance of the anti-spoofing systems. Secondly, we demonstrate how to exploit this information by describing two new client-specific anti-spoofing approaches. The first one builds generative models for the real accesses of each of the enrolled clients and normalizes the scores with spoofing attack models. The second one incorporates client-specific information into the training of a discriminative binary classifier, resulting in a separate classifier for each client. Focusing on the face mode, we assess the performance of these client-specific approaches and their generalization capabilities to detect spoofing attacks of types not seen during training. As a final contribution, the implementation of the client-specific methods are available as open-source, allowing for testing the proposed methods using different features and biometric modes.

Applying the proposed client-specific methods to three different state-of-the-art face anti-spoofing features, we observe a notable improvement over client-independent approaches which do not use information about the client identity. Moreover, in most cases, the client-specific approaches appear to generalize better in detecting spoofing attacks of a type not seen during training. We would like to emphasize that the main objective of the paper is to establish the prospects of using client-specific information in anti-spoofing, rather than studying an extensive set of anti-spoofing features and their applicability in this context.

In the remainder of this paper, Section II covers the typical features and methods used in face anti-spoofing. The motivation and the description of the proposed client-specific approaches is given in Section III. Section IV describes the experimental findings, while Section V concludes the paper and summarizes future perspectives.

II. Related work

The problem of spoofing detection can be regarded as a binary classification problem, with the real access samples considered the positive class and the spoofing attacks the negative one. The typical anti-spoofing features are selected to have good discrimination capabilities with respect to the two classes, regardless of the identities of the clients that the samples belong to. As such, the features are directly used to train binary classifiers of different kinds.

The anti-spoofing features can be derived from various sources. Some methods use multi-spectral analysis [6], or response to a challenge which is posed to the user [7] to detect spoofing attacks. The mandatory additional hardware required alongside the biometric sensor is the primary drawback of the former approach, while the intrusiveness of the latter. As a result, many authors are in favor of features which are obtained in a completely automatic software-based manner from the sample captured by the biometric sensor [5].

Depending on the cues used to discern spoofing attacks from real accesses, the software-based anti-spoofing features for the face mode fall into one of three categories: texture-based, motion-based and liveness-based. The texture-based features exploit differences in texture between real accesses and attacks. The sources of these differences could be: different optical qualities of the human skin and the spoofing media [8]; blurring due to the limited resolution of the spoofing device [9] or involuntary shaking while performing the attack; artifacts appearing in the spoofing production process [10] or diffuse reflection due to a non-natural shape of the spoofing attack [9]. To exploit the texture discrepancies, several authors analyze the differences between the samples in a specific set of frequency bands [11], [12]. Another popular approach analyzes the reflectance component extracted from the images [13], [14]. Expecting degradation of the quality of the recaptured images in the case of spoofing attacks, [15] derives anti-spoofing features from image quality measurements. Following another trail to the same end, several notable works use low-level texture descriptors extracted from the face region, like Local Binary Patterns (LBP) [16], [17], Gabor wavelets [10] or Histogram of Oriented Gradients (HOG) [9]. Furthermore, [9] demonstrated the benefit of extracting these features from face components, like eyes, nose or mouth, while [18] uses them on faces whose micro and macro motion is enhanced using motion magnification technique. [19] indicates that low-level features can be used to detect the edges of a spoofing media.

Aiming at exploiting the variation of texture patterns in the course of a video segment, [20] proposes a spatio-temporal descriptor of face appearance and dynamics based on the LBP-TOP operator [21]. Another extension of the texture analysis into temporal domain is presented in [22].

The motion-based methods explore movements on the scene which are unusual for a 3D human face, and exploit them as a cue that an attack is being performed. Similar amount of movement in the central and peripheral face parts is one such peculiarity which can be observed in the case of a 2D face attacks [23]. Heuristic of the movement of 2D and 3D surfaces is developed in [24]. The high correlation between the movements of the face region and the background as an indication of a spoofing attack is used in [25] and [26].

The liveness-based methods try to detect evidence of liveness in the scene. Eye-blinking and involuntary lip movements are the most typical signals used by these methods [27].

A recent trend that came into prominence with [28], [26] involves combining several methods at score or feature level. The best results are achieved if the fused methods use complementary features which discern spoofing attacks from different aspects [29]. Such methods achieve the best performances up to date, even when confronted with a multitude of attack types [30].

With regards to the classification step which follows the feature extraction, the systems comply to the binary classification definition. For a sample with feature vector $\mathbf{x}$, the systems determine the value of its class $c \in \{\mathrm{R}, \mathrm{A}\}$, where R stands for the class of Real accesses and A stands for the class of attacks. The majority of anti-spoofing systems use discriminative approaches, like Support Vector Machines (SVM) [13], [17], [10], [19], [9], [20], Linear Discriminant Analysis (LDA) [17], Sparse Logistic Regression [14] and Multi-Layer Perceptron (MLP) [25]. Some systems produce scores on which a threshold can be directly applied [11], [24].

To the best of our knowledge, a generative approach has

not been proposed for anti-spoofing up to the present moment. However, keeping in mind the binary nature of the problem, generative system can be easily developed by using probabilistic theory. Two generative models, which can be based on a Gaussian Mixture Model (GMM), can be built separately for the class of real accesses and for the class of attacks. To infer the class c given the feature vector $\mathbf{x}$, we firstly need to compute the likelihoods $p(\mathbf{x}|c)$. Then, the final decision is taken depending on the value of the log-likelihood ratio (LLR) defined in Equation 1, and a pre-determined decision threshold [31].

$$c_{LLR} = \log \frac{p(\mathbf{x}|c = \mathrm{R})}{p(\mathbf{x}|c = \mathrm{A})} \quad (1)$$

During training of the systems mentioned above, the full set of available real access and spoofing attack samples 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.

III. Client-specific approaches

The client-independent classification methods described in Section II assume that the characteristics extracted from the available samples (both real accesses and spoofing attacks) are invariant across the different clients and do not include any client-specific information. This is a reasonable hypothesis, as the proposed features are specifically designed to capture artifacts of spoofing attacks regardless of the client identities. Yet, we argue that many of the proposed features, even inadvertently, retain information specific to the clients that the samples belong to.

The main objective of this paper is to establish the idea of considering the client identity information when building anti-spoofing systems. It does not intend to compete with the best performing methods like in [30], which rely on complex combinations of anti-spoofing features. Instead, it aims to demonstrate how client-specific classification can be beneficial even when using simple features, upon which many of the best performing methods are built. As a case study, we focus on three types of state-of-the-art face anti-spoofing features: LBP [17], LBP-TOP [20] and MOTION [25], each of which tackles the spoofing attack detection from a different perspective. The source code of our methods is publicly available[1], so that the presented results can be easily reproduced. More importantly, the code can be readily used to analyze the applicability of client-specific approaches for any other anti-spoofing features or combination of features.

The client-specific approaches can use the client identity information in the same way as the biometric recognition system they protect. This information may come from the claim of the client about his identity (in a verification scenario), or from the enrolled model the sample is matched to (in an identification scenario). To exemplify, the operation of a client-specific anti-spoofing system protecting a verification system is illustrated in Fig. 1. The (blue) dashed line feeding the client identity information into the anti-spoofing system, is the main difference of a client-specific approach with respect to a typical client-independent approach.

[1]http://pypi.python.org/pypi/antispoofing.clientspec

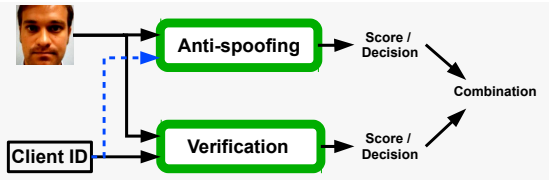


Fig. 1: Flow diagram of the operation of a client-specific anti-spoofing system in a verification scenario

A. Motivation

Besides the characteristics of a live person on the scene, many of the face anti-spoofing features described in Section II may capture intrinsic personal properties of the client the sample belongs to. These properties may be related to the appearance or the behavior of the client. For example, the involuntary movements or eye-blinking patterns, which are an intrinsic client trait, may be manifested into the motion-based or spatio-temporal anti-spoofing features. The physical appearance of the face or the skin tone and surface are likely to have an impact on the texture-based features, like LBP or Gabor wavelets. Indeed, LBP and Gabor wavelets, in a more complex variants, are commonly used as discriminative features in face verification [32]. Although the anti-spoofing features are not likely to be used for face recognition, the above observations give rise to the following questions: (1) do these features carry information about the client; (2) is this information relevant to make a better discrimination between the real access and spoofing samples belonging to that client.

An acclaimed study of the dependence of verification scores on the identity of the clients [33], established the popular Doddington's zoo. In particular, 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 its original form, the Doddington's zoo assumes just two types of inputs to the verification system: genuine users and zero-effort impostors. A recent study on a fingerprint verification system demonstrates the existence of the Doddington's zoo effect when a verification system is confronted with spoofing attacks [34]. The study has shown that the system's vulnerability to spoofing among a population is client-specific as well. We perform a similar analysis, evaluating the scores of state-of-the-art client-independent anti-spoofing systems using the aforementioned LBP, LBP-TOP and MOTION features. The analysis is done on the Replay-Attack face spoofing database [17], details of which are given in Section IV-A. The analyzed client-independent classifier is SVM, which is the baseline classifier used in [17], [20]. Note that, unlike [34], which evaluates the verification scores, we analyze the anti-spoofing scores per client. For the three studied anti-spoofing features, Fig. 2 displays box plots showing the variation of the scores for the real access and spoofing attack samples of each

client in the test set of the database. The decision threshold determined using Equal Error Rate (EER) on the development set is plotted as well. 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.

Fig. 2 demonstrates the existence of client-specific score variations for the client-independent baseline, especially in the case of real access samples. Similar conclusion can be drawn by performing Kruskal-Wallis non-parametric statistical test [35], with null-hypothesis stating that there is no variation in the scores of the samples of the client population. In our analysis, this hypothesis was rejected at a 0.01% significance level.

The high client-specific score variations mean that different clients contribute differently to the system's FAR and FRR. These variations 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 of 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.

According to [36], creating client-specific models is one way to alleviate the issue. In this context, we explore two different directions. The first one presents a generative approach, 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 is a discriminative approach, with separate SVM classifier trained for each client. We show that by considering client-specific aspects, performance can be significantly boosted in both cases.

B. Client-specific generative approach

To build a client-specific generative approach to anti-spoofing, we use Probabilistic Graphical Model framework (PGM) [31]. Let's assume that from each sample we can extract K different feature vectors $\mathbf{x}_k \in \mathbb{R}^{n_k}, k \in 1..K$, each with a dimensionality n_k and related to different types of anti-spoofing cues. Naively, they are assumed to be mutually independent.

While a client-independent generative approach assumes that all the responsibility for the features comes from the sample class $c \in \{\mathrm{R}, \mathrm{A}\}$, a client-specific generative method assumes that the features are not completely independent of other variables, like the identity i of the client the sample belongs to. Such a dependency scheme can be illustrated using a Bayesian network as in Fig. 3.

Having such a model, the conditional probability of a set of features $\mathrm{X} = \{\mathbf{x}_k | k \in 1..K\}$ is given in Eq. 2.

$$p(\mathrm{X}|c, i) = \prod_{k=1}^{K} p(\mathbf{x}_k|c, i) \tag{2}$$

Given the sample features X and the client identity $i = \mathrm{I}$, we need to infer the class $c \in \{\mathrm{R}, \mathrm{A}\}$. For this, we need to compute the likelihoods $p(\mathrm{X}|c, i = \mathrm{I})$ based on a real access and spoofing attack models trained separately for each client. In this study, the likelihood function $p(\mathrm{X}|c, i)$ is modeled with Gaussian Mixture Model (GMM). Afterwards, the decision on the value of c is taken by computing the log-likelihood ratio as in Equation 3 and comparing it with a decision threshold θ.

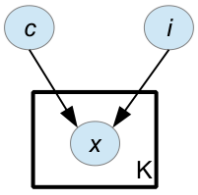


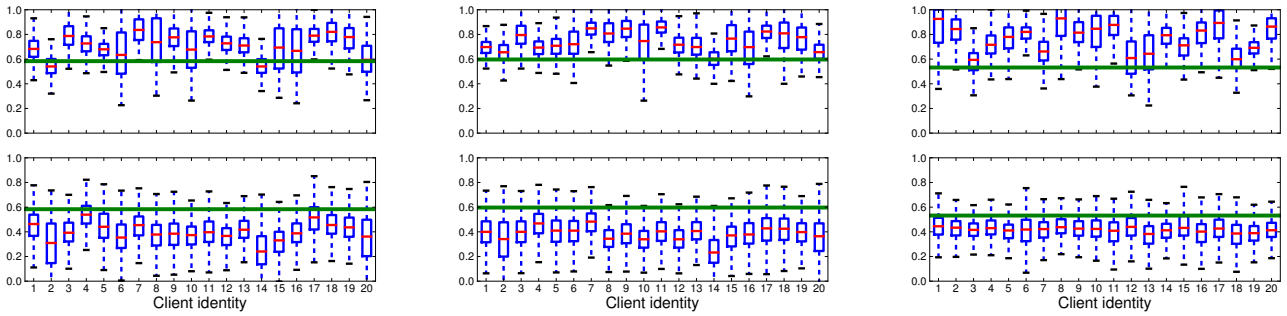
Fig. 3: PGM for client-specific generative approach: features depend on sample class and identity

$$c_{LLR} = \log \frac{p(\mathrm{X}|c = \mathrm{R}, i = \mathrm{I})}{p(\mathrm{X}|c = \mathrm{A}, i = \mathrm{I})} \tag{3}$$

The idea is similar to a well-known method in biometric verification, initially proposed for the task of speaker verification [37]. To distinguish between a genuine client and impostor, the method compares two hypotheses: $\mathcal{H}_0$ stating that the sample comes from the client with identity i, and $\mathcal{H}_1$ as an alternative hypothesis stating that the sample comes from any other client (impostor). The likelihoods for both hypotheses are modeled with GMM. By definition, the model for $\mathcal{H}_0$ should be created using samples from the genuine client. The model for $\mathcal{H}_1$ can be created in different ways. One way is to estimate it as a single model, usually called Universal Background Model (UBM), by using a *background* set of other clients $\mathcal{B}$. Another way is to build a separate model for each of the clients in a predefined set called *cohort* set $\mathcal{C}$ and to estimate the likelihood of $\mathcal{H}_1$ as a function of the likelihoods of the separate models.

In the case of a client-specific spoofing detection, the hypothesis $\mathcal{H}_0$ states that the sample comes from a real access of the client with identity I, and its likelihood can be defined as $p(\mathrm{X}|\mathcal{H}_0) = p(\mathrm{X}|c = \mathrm{R}, i = \mathrm{I})$. The definition of $p(\mathrm{X}|\mathcal{H}_0)$ implies that its model can be created by using the real access samples of the client with identity I. Hence, an essential requirement for the method is the availability of real access samples for each client at training time. To satisfy this requisite, it is enough to use the samples which are used to enroll the client in the biometric recognition system.

The alternative hypothesis $\mathcal{H}_1$ states that the sample comes from an attack of the client with identity I and its likelihood can be defined as $p(\mathrm{X}|\mathcal{H}_1) = p(\mathrm{X}|c = \mathrm{A}, i = \mathrm{I})$. By following this definition, we need to build a model for the attacks for each client. This step requires access to attack samples for each of the clients at training time. 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



(a) LBP features (b) LBP-TOP features (c) MOTION features

Fig. 2: Box plots of the scores obtained with a **client-independent** approach (SVM) for different clients in the test set of Replay-Attack. 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.

very demanding, complex or, if the number of clients is large, too costly. Baseline costs will 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 $\mathcal{H}_1$ as a function of the likelihoods of spoofing attack models for a finite set of cohort clients $\mathcal{C}$. Cohort clients have been extensively used in biometric verification in different setups. In [38] they are used to perform T-normalization of the scores. In [39] they are sorted by similarity with the particular client's model, and only the first N are taken to represent $\mathcal{H}_1$. In [40] $\mathcal{H}_1$ is modeled with the model of the cohort client with highest likelihood. Similarly, [41] considers only the cohort client with the highest likelihood among the cohorts selected after sorting. Although larger cohort set may yield better results [39], selection of a subset of cohorts for each client is sometimes done due to computational limitations.

Having no such constraints, we consider all the clients in the training set as cohort clients. We build attack models for each of the clients in the cohort set $\mathcal{C}$. To estimate the likelihood of the hypothesis $p(\mathrm{X}|\mathcal{H}_1)$, we average the likelihoods of the cohort attack models as in Eq. 4. Finally, the log-likelihood ratio in Eq. 3 can be expressed as in Eq. 5.

$$p(\mathrm{X}|\mathcal{H}_1) = \frac{1}{|\mathcal{C}|}\sum_{\mathrm{J}\in\mathcal{C}} p(\mathrm{X}|c=\mathrm{A}, i=\mathrm{J}) \tag{4}$$

$$c_{LLR} = \log \frac{p(\mathrm{X}|c=\mathrm{R}, i=\mathrm{I})}{\frac{1}{|\mathcal{C}|}\sum_{\mathrm{J}\in\mathcal{C}} p(\mathrm{X}|c=\mathrm{A}, i=\mathrm{J})} \tag{5}$$

The proposed modeling of $\mathcal{H}_1$ has two main advantages. Firstly, it requires availability of spoofing attacks of only a limited number of clients, which can be manufactured and recorded in advance. Secondly, the set of spoofing attacks can be easily augmented in case a new type of spoofing attack appears. Retraining of the system in such a case requires only retraining of the attack models for the cohorts.

Similarly to [37], we use GMM to model the likelihoods for $\mathcal{H}_0$ and $\mathcal{H}_1$. Firstly, we create a UBM for real accesses and attacks using the training set. Then, to create the client-specific real access models for each enrolled client, we perform Maximum A-posteriori (MAP) adaptation of the real access UBM using the client's enrollment samples. Equivalently, to create the attack models, we perform MAP adaptation of the attack UBM using the attack samples of the clients in the cohort set $\mathcal{C}$. The MAP adaptation is performed on the means of the GMM components only, as suggested in [38].

C. Client-specific discriminative approach

As described in Section II, discriminative classifiers have already a well established reputation in face anti-spoofing. One of the most popular among them is SVM, a classifier relying on a margin maximization theory to find a hyperplane that separates positive and negative samples with minimal generalization error. In this context, SVM is typically used in a client-independent way, where all real access samples are considered to belong to the positive class and all spoofing attacks to the negative one.

SVM in a client-specific context first appeared in speaker verification [42]. A single SVM classifier in this setup discriminates between samples coming from a claimed identity and samples coming from an impostor. The full system consists of separate SVM classifiers trained for each enrolled client. 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 we train a separate SVM classifier whose role is to discriminate between real accesses and spoofing attacks for that client. Therefore, ideally, each SVM should be trained using samples of the corresponding client. As in the case of the generative client-specific approaches, we can use the enrollment samples as the positive class. However, as explained in Section III-B, obtaining spoofing attacks for each client may be a costly task.

As for the generative approach, we select a set of cohort clients $\mathcal{C}$ to approximate the spoofing attacks of a particular client and represent the negative class for the client-specific SVMs. Since the samples from the cohort usually outnumber the client samples, the selection of the clients in $\mathcal{C}$ 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 [43]. Instead of cohort clients, [44] 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 [42]. 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.

D. Implementation considerations

In the test phase, client-specific anti-spoofing systems need to compare the query sample with the appropriate client-specific model or discriminative classifier. A question that may arise is how they may obtain the client identity information. It is important to understand that the task of anti-spoofing systems is to protect a biometric recognition system. As a result, they may use any information available to the recognition system, like the client identity. The flow diagram in Fig. 1 exemplifies this concept for the case of a biometric verification system.

Client-specific methods can not be used at enrollment time, when the client models are not yet created. If a protection against spoofing is needed at this point, a client-independent anti-spoofing system can be used at the cost of a lower performance. However, when the biometric system is in operation mode, the client identity is available and, as we show in Section IV, can be used to improve the anti-spoofing performance.

An important requirement for client-specific anti-spoofing systems is the availability of enrollment (gallery) samples for each of the clients. They are used to create client-specific real access models for the generative approach and to train client-specific classifiers in the discriminative ones. In a real world scenario, the anti-spoofing system may use the samples used to enroll clients in the biometric recognition system for this purpose. In the process of enrolling a new client to an existing system, the generative approach builds a real access model of the new client only, based on his enrollment samples and the existing UBM of real accesses. In the discriminative approach, a client-specific SVM for the new client is trained using his enrollment samples and the cohort attack samples. The real access models for all the other clients, as well as the attack models remain unchanged. The same applies for the SVMs of the other clients in the discriminative approach.

IV. Experimental results

A. Database and evaluation methodology

A typical client-independent anti-spoofing system can be evaluated using databases which provide real access and spoofing attack samples for many identities. As explained in Section III-D, client-specific approaches have an additional requirement: enrollment samples for each client. The availability of enrollment samples is important from another aspect as well: it allows for training a biometric recognition system and evaluation of its vulnerability to spoofing attacks [17].

Biometric recognition databases encompass enrollment samples which are used to enroll clients into the biometric recognition system. Unfortunately, the majority of face-spoofing databases provide only real access and spoofing attack samples for the clients and usually lack enrollment data. This is the case, for example, for NUAA Photograph Impostor Database [14] and CASIA-FASD [12], which do not dedicate samples for client enrollment in their protocols. Violating their protocols would make comparison with previous approaches on these databases biased. 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, because this data comes either from a single session (NUAA) or a single video (CASIA-FASD), making them highly correlated.

The Replay-Attack face spoofing database[2] [17] is, to the best of our knowledge, the only publicly available face spoofing database that provides separate enrollment samples. With 50 clients divided into three non-overlapping subsets, it provides a clear protocol that can be used for non-biased training, tuning and testing of algorithms. It encompasses real access samples taken in two conditions and three types of spoofing attacks: printed photographs, digital photographs and video replays.

The goal of our experiments is to show the advantage of client-specific over client-independent approaches for the set of features selected for analysis: LBP, LBP-TOP and MOTION. We report the Half Total Error Rate (HTER) on the development or test set with a threshold obtained on the development set. As suggested by the database protocol, the development set of the database is used to tune specific hyper-parameters of the algorithms.

The Replay-Attack database defines several evaluation protocols related to the type of spoofing attacks they contain. Using these protocols, we perform the experiments in two phases. The first phase covers *intra-protocol* evaluations, where the algorithm is trained and tested using the same protocol. These experiments demonstrate the performance of the algorithm in detecting already seen types of spoofing attacks. The second phase involves *cross-protocol* evaluations, where the algorithm is trained and tuned using one protocol and tested on another one. In this way, similarly to [45], we investigate the capabilities of the algorithm to generalize over an unseen type of spoofing attacks.

B. Features and parameter selection

Different parameterization options have been proposed in the literature for the types of features used in our evaluation, in particular for LBP and LBP-TOP. For example, these features may yield better results in their multi-scale variants [16], [20]. Yet, to study the effects of using client identity information,

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

TABLE I: Comparison of different approaches on grandtest (error rates in %)

		LBP				LBP-TOP				MOTION			
		dev	test			dev	test			dev	test		
		EER	FAR	FRR	**HTER**	**EER**	FAR	FRR	**HTER**	**EER**	FAR	FRR	**HTER**
discriminative	**client-independent**	14.56	9.56	21.29	15.42	8.19	4.45	12.6	8.53	10.73	12.95	10.12	11.53
	client-specific	**10.02**	8.2	11.53	**9.87**	**3.71**	3	4.9	**3.95**	**10.18**	9.29	13.25	**11.27**
generative	**client-independent**	21.33	17.93	25.45	21.69	9.32	8.38	16.92	12.65	12.94	13.91	11.14	12.52
	client-specific	**9.18**	9	11.17	**10.09**	**4.73**	5.36	7.37	**6.36**	**8.91**	9.4	9.92	**9.66**

we focus on the most simple uniform variants $\text{LBP}^{u2}_{8,1}$ and $\text{LBP-TOP}^{u2}_{8,8,8,1,1,1}$.

The described client-specific approaches depend on few hyper-parameters which have been optimized by a grid parameter search on the development set. The parameters for the generative client-specific approach include the number of components in the GMM for the real access and spoofing attack models, as well as the relevance factor for performing MAP adaptation. We found that the LBP and LBP-TOP features require relatively high number of components for the real access GMMs, ranging between 240 - 290 depending on the protocol. The MOTION features, on the other hand, can successfully model the client-specific real access feature space with only 10-50 components. The spoofing attack GMMs consist of smaller number of components: below 100 for each of the features. The number of components is usually the highest for the attack models of the grandtest protocol, possibly due to the spread of the different types of spoofing attacks in the feature space.

The hyper-parameters for the discriminative approach realized with SVM with RBF kernel include the constant C regularizing the decision surface and the kernel parameter γ controlling the support vectors' influence. The best values for these parameters were found to be $C = 1$ and $\gamma = 0$.

Details about the exact parameterization for each of the protocols in our evaluation are given together with the openly available source code, making the reported results fully reproducible.

C. Intra-protocol evaluation

In the first experiment, we evaluate all methods considering the grandtest protocol of Replay-Attack, which consists of all three types of spoofing attacks: printed photographs, digital photographs and video replays. Table I provides a comparison of the performance of the baseline client-independent with client-specific approaches. EER is given for the development set, while FAR, FRR and HTER are given for the test set.

Table I shows that the client-specific approaches consistently perform better than the client-independent ones for all the three types of features, both on the development and test set. The advantage of client-specific approaches is significantly large in almost all the cases: the relative improvement of HTER reaches up to 53.7% for the discriminative techniques and 49.7% for the generative techniques (applied to LBP-TOP features). A single exception where the superiority of the client-specific method is not so prominent happens for the discriminative techniques applied to MOTION features. The reason may be in the way MOTION features are constructed. While LBP and LBP-TOP features are extracted solely from the face region, MOTION features contain elements extracted from the background, possibly introducing a client-independent component. Table I also reveals that the client-independent approaches most often present a large mismatch between the values of FAR and FRR, while the client-specific approaches have no such issue.

To understand the improvement of anti-spoofing performance when client-specific approaches are used, in Fig. 4 we give box plots of the client scores obtained using a client-specific approach. Compared to Fig. 2, the medians of the scores have notably more similar values over the client population, especially for LBP and LBP-TOP features. Furthermore, the variance of the scores for each client notes a significant decrease. Since the client identity is used in modeling the real accesses, the variance is especially small for real access scores (upper row). This shows that client-specific approaches model the feature space in a way that is equally suitable for all the clients, allowing for a single decision threshold which does not favor one client over the other.

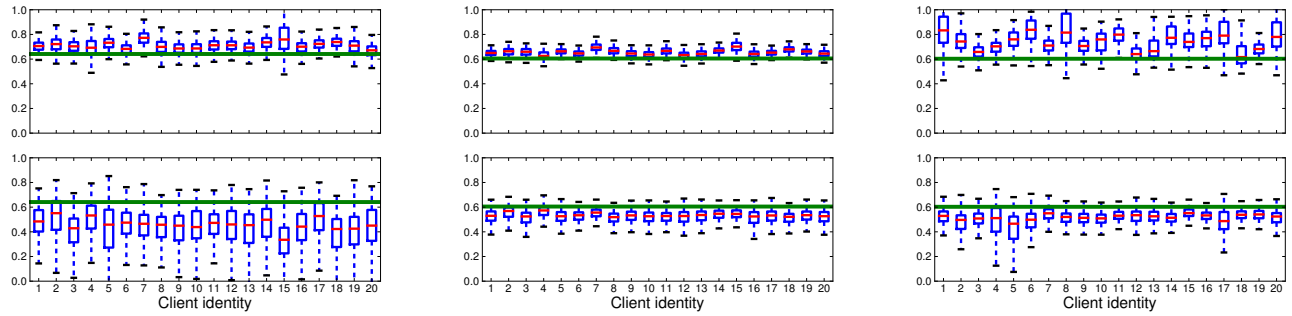
We performed intra-protocol evaluation for the other protocols defined in Replay-Attack, consisting only of a single type of attack. Fig. 5 shows the results for the three types of features. Each group of four bars represents the HTER on the test set for the four compared methods on one of the protocols and the methods belonging to the same category (discriminative and generative) are given next to each other for easy comparison. For LBP and LBP-TOP features, the results are always in favor of the client-specific approaches for all three types of attacks. Absolute HTER drops from 6.86% to 1.15% (LBP features on printed attacks) and from 8.74% to 3.99% (LBP-TOP features on attacks with digital photographs) highlight this advantage.

Once again, MOTION features behave differently and the client-specific approaches perform on similar scale or slightly worse than the client-independent ones. This is especially evident for the video attacks, because from the point of view of these features, there is practically no difference between the motion patterns of a live client and the same client recorded in a video.

Table II compares the best performing configuration of features and methods among the client-independent vs. client-specific approaches for separate protocols of Replay-Attack. It suggests that, regardless of the type of spoofing attacks, a client-specific approach that will outperform any client-independent approach can always be selected.

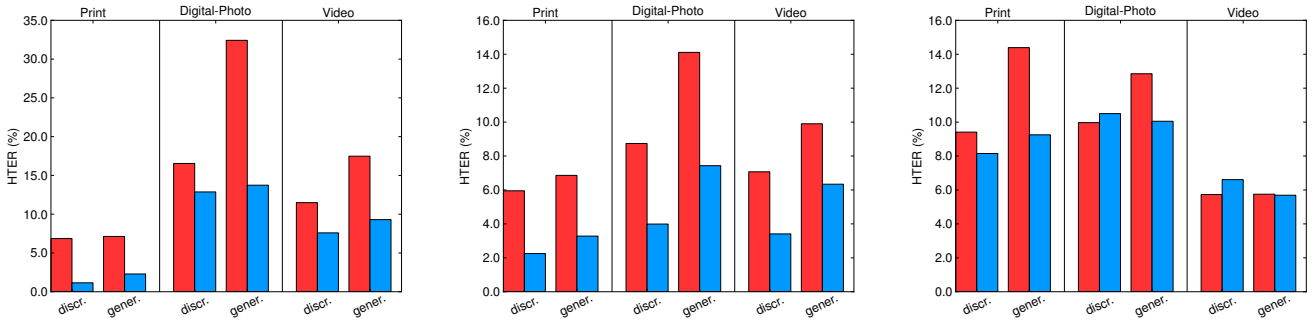
D. Cross-protocol evaluation

The capability to generalize well in detecting spoofing attacks which have not been considered during training is an



(a) LBP features (b) LBP-TOP features (c) MOTION features

Fig. 4: Box plots of the scores obtained with generative **client-specific** approach for different clients in the test set of Replay-Attack. 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) LBP features (b) LBP-TOP features (c) MOTION features

Fig. 5: Intra-protocol evaluation. HTER is computed on the test set. : client-independent approaches; : client-specific approaches. Discriminative approaches are denoted with the shortcut 'discr.', while generative with 'gener.'.

TABLE II: Comparison of best performing client-independent and client-specific approaches in intra-protocol evaluation (HTER in % on the test set)

	Grandtest	**Print**	**Digital-Photo**	**Video**
client-independent	8.53	5.95	8.74	5.73
client-specific	**3.95**	**1.15**	**3.99**	**3.41**

important aspect of any anti-spoofing system [45]. Indeed, having in mind that the possibilities for inventing novel spoofing attacks are unlimited, exhibiting robustness to unseen spoofing attacks may be a major security asset of anti-spoofing systems. In this section, the generalization capabilities of the proposed and the baseline anti-spoofing methods are studied using a cross-protocol evaluation.

We investigate three scenarios. In each one of them, the systems are trained using two out of the three available types of attacks, while the third one serves for testing. The scenarios' descriptions are as follows:

- Scenario 1: train with digital photographs and video replays, test on printed photographs;
- Scenario 2: train with printed photographs and video replays, test on digital photographs;
- Scenario 3: train with printed and digital photographs, test on video replays.

According to this, the three scenarios reveal the generalization capabilities of the algorithms when tested on printed photographs, digital photographs and videos, respectively. The results are given in Fig. 6, where each group of four bars corresponds to one of the described scenarios.

The client-specific approaches outperform the client-independent ones for all the three types of features. The HTER drop is especially significant in the case of LBP and LBP-TOP features. When using these features, the client-independent approaches exhibit very low generalization capabilities for certain types of attacks, like printed or digital photographs. This weakness is easily overcome by their client-specific counterparts. For example, for LBP features, the client-independent discriminative baseline trained in Scenario 1 gives HTER as large as $\sim$45% in detecting printed attacks. On the other hand, the client-specific approaches in the same scenario misclassify only $\sim$7% of the samples. Similar example under Scenario 1 appears in the case of LBP-TOP features: the client-independent baseline gives HTER of more than 38%. The client-specific approaches increase the generalization capabil-

TABLE III: Comparison of best performing client-independent and client-specific approaches in cross-protocol evaluation (HTER in % on the test set)

	Print	Digital-Photo	Video
client-independent	28.27	16.55	9.39
client-specific	**5.00**	**6.02**	**5.09**

ity of these features by a large margin and achieve HTER of only as little as 5%.

While client-specific approaches improve the generalization capabilities of the MOTION features in most of the cases as well, an exception occurs only for the discriminative client-specific method operating under Scenario 3.

Table III compares the best performing configuration of features and method among the client-independent vs. the client-specific approaches in cross-protocol evaluation. It shows that regardless of the types of attacks used for training, a carefully selected client-specific approach will be more robust to new types of attacks than any client-independent approach.

Good cross-protocol generalization of client-specific approaches may be a result of the client-specific models being better suited to separate the real access samples of a particular client from the remainder of the feature space.

V. Conclusions

Anti-spoofing systems are most frequently designated to secure and work in cooperation with biometric recognition systems. This paper is, to the best of our knowledge, the first attempt to make use of the information about the identities of the enrolled clients to improve the performance of anti-spoofing systems. The idea is based on the assumption that the typical anti-spoofing features may, to a certain extent, retain client-specific information. This assumption is supported by an empirical evidence.

We implemented two solutions which use client identity information to detect spoofing attacks: a generative and a discriminative one. Tested with a variety of state-of-the-art features, they notice a significant performance gain on different protocols of Replay-Attack face spoofing database. For the grandtest protocol of Replay-Attack the relative improvement can reach up to $\sim$50%. Furthermore, they generalize significantly better on unseen types of attacks which the system has not been trained with.

Client-specific anti-spoofing systems use the client identity information available to the biometric recognition system. If a spoofing counter-measure is needed at enrollment time, a client-independent anti-spoofing method can be used at the cost of lower performance. However, once the client identity is known, using this information can be of great help in successfully detecting spoofing attacks.

Extensions of this work are possible in several directions. Employing the method with other types of features or a combination of features should be a primary future objective. Furthermore, it is of importance to investigate the applicability of client-specific anti-spoofing systems to other biometric modes. Our framework, whose source code is publicly available for reproducing results, readily enables testing other features or combination of features.

Another direction to be explored is investigating ways to select the cohort set dynamically and in a client-specific way, both for the generative and discriminative approaches.

Finally, the current evaluation of the systems excludes zero-effort impostors which claim a wrong identity. While irrelevant in the client-independent case, this claim will directly influence the decision of the client-specific anti-spoofing systems. Since detecting zero-effort impostors is a task of biometric recognition systems, an evaluation in such circumstances should be done jointly with a recognition system.

References

[1] Y. Li *et al.*, "Understanding OSN-based facial disclosure against face authentication systems," in *9th ACM Symposium on Information, Computer and Communications Security*, 2014, pp. 413–424. 1

[2] C. Arthur, "iPhone 5S fingerprint spoof could lead to ID theft," *The Guardian*, 2013. 1

[3] D. Denning, "Why I love biometrics: It is "liveness," not secrecy, that counts," *Information Security Magazine*, 2001. 1

[4] A. K. Jain, P. Flynn, and A. A. Ross, *Handbook of Biometrics*. Springer-Verlag New York, Inc., 2007. 1

[5] S. Schuckers, *Encyclopedia of Biometrics*. Springer-Verlag, 2009, ch. Liveness Detection: Fingerprint, pp. 924–931. 1, 2

[6] Z. Zhang *et al.*, "Face liveness detection by learning multispectral reflectance distributions," in *Automatic Face Gesture Recognition and Workshops (FG 2011)*, 2011, pp. 436–441. 2

[7] K. Kollreider *et al.*, "Real-time face detection and motion analysis with application in liveness assessment," *Information Forensics and Security, IEEE Transactions on*, vol. 2, no. 3, pp. 548–558, Sept 2007. 2

[8] G. Parziale, J. Dittman, and M. Tistarelli, "Analysis and evaluation of alternatives and advanced solutions for system elements," BioSecure D 9.1.2, 2005. 2

[9] J. Yang *et al.*, "Face liveness detection with component dependent descriptor," in *International Conference on Biometrics (ICB), 2013*, June 2013, pp. 1–6. 2

[10] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using texture and local shape analysis," *IET Biometrics*, vol. 1, pp. 3–10, 2012. 2

[11] J. Li *et al.*, "Live face detection based on the analysis of fourier spectra," *Biometric Tecnology for Human Identification*, 2004. 2

[12] Z. Zhiwei *et al.*, "A face antispoofing database with diverse attacks," in *Proceedings of the 5th IAPR International Conference on Biometrics (ICB'12), New Delhi, India*, 2012. 2, 6

[13] J. Bai *et al.*, "Is physics-based liveness detection truly possible with a single image?" in *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2010. 2

[14] X. Tan *et al.*, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *ECCV (6)*, 2010, pp. 504–517. 2, 6

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

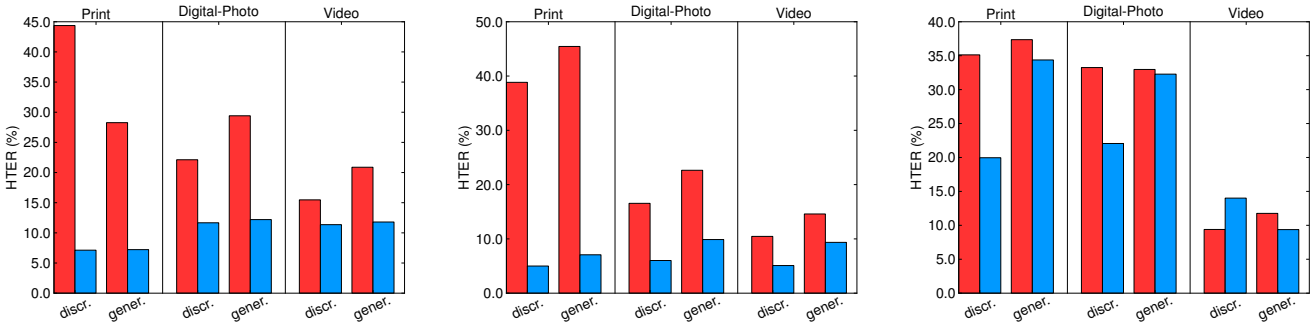
[16] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *International Joint Conference on Biometrics*, 2011, pp. 1–7. 2, 6

[17] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *Biometrics Special Interest Group (BIOSIG), Proceedings of the International Conference of the*, 2012, pp. 1–7. 2, 3, 6

[18] S. Bharadwaj *et al.*, "Computationally efficient face spoofing detection with motion magnification," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013*, June 2013. 2

[19] 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*, Sept 2013, pp. 1–8. 2

[20] T. de Freitas Pereira *et al.*, "Face liveness detection using dynamic texture," *EURASIP Journal on Image and Video Processing*, vol. 2014:2, 2014. 2, 3, 6



(a) LBP features (b) LBP-TOP features (c) MOTION features

Fig. 6: Cross-protocol evaluation. HTER is computed on the test set. : client-independent approaches; : client-specific approaches. Discriminative approaches are denoted with the shortcut 'discr.', while generative with 'gener.'.

[21] 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*, vol. 29, no. 6, pp. 915–928, 2007. 2

[22] A. d. S. Pinto *et al.*, "Video-based face spoofing detection through visual rhythm analysis," in *25th Conference on Graphics, Patterns and Images*, 2012. 2

[23] K. Kollreider, H. Fronthaler, and J. Bigun, "Non-intrusive liveness detection by face images," *Image and Vision Computing*, vol. 27, no. 3, 2009. 2

[24] W. Bao *et al.*, "A liveness detection method for face recognition based on optical flow field," *2009 International Conference on Image Analysis and Signal Processing*, 2009. 2

[25] 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 2011*, 2011. 2, 3

[26] J. Yan *et al.*, "Face liveness detection by exploring multiple scenic clues," in *12th International Conference on Control, Automation, Robotics and Vision*, 2012. 2

[27] G. Pan, Z. Wu, and L. Sun, "Liveness detection for face recognition," *Recent Advances in Face Recognition*, December 2008. 2

[28] R. Tronci *et al.*, "Fusion of multiple clues for photo-attack detection in face recognition systems," in *IJCB*, 2011, pp. 1–6. 2

[29] J. Komulainen *et al.*, "Complementary countermeasures for detecting scenic face spoofing attacks," in *Biometrics (ICB), 2013 International Conference on*, 2013. 2

[30] I. Chingovska *et al.*, "The 2nd competition on counter measures to 2d face spoofing attacks," in *International Conference of Biometrics 2013*, 2013. 2, 3

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. 3, 4

[32] 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. 3

[33] G. Doddington *et al.*, "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. 3

[34] A. Rattani, N. Poh, and A. Ross, "Analysis of user-specific score characteristics for spoof biometric attacks." in *CVPR Workshops*. IEEE, 2012, pp. 124–129. 3

[35] J. D. Gibbons, *Nonparametric Statistical Inference, 2nd. Ed.* 4

[36] N. Poh and J. Kittler, "On the use of log-likelihood ratio based model-specific score normalisation in biometric authentication," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science, S.-W. Lee and S. Li, Eds., 2007, vol. 4642, pp. 614–624. 4

[37] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," in *Digital Signal Processing*, 2000, p. 2000. 4, 5

[38] R. Auckenthaler, M. J. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems." *Digital Signal Processing*, vol. 10, no. 1-3, pp. 42–54, 2000. 5

[39] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," *Speech Communication*, vol. 17, no. 12, pp. 91 – 108, 1995. 5

[40] D. Simon-Zorita *et al.*, "Facing position variability in minutiae-based fingerprint verification through multiple references and score normalization techniques," in *Audio- and Video-Based Biometric Person Authentication*, ser. Lecture Notes in Computer Science, 2003, vol. 2688, pp. 214–223. 5

[41] G. Aggarwal, N. Ratha, and R. Bolle, "Biometric verification: Looking beyond raw similarity scores," in *Computer Vision and Pattern Recognition Workshop*, 2006, pp. 31–31. 5

[42] M. McLaren, "Improving automatic speaker verification using SVM techniques," Ph.D. dissertation, Queensland University of Technology, 2009. 5, 6

[43] S. S. Kajarekar and A. Stolcke, "NAP and WCCN: Comparison of approaches using MLLR-SVM speaker verification system," in *ICASSP (4)*, 2007, pp. 249–252. 6

[44] M. McLaren *et al.*, "Data-driven background dataset selection for SVM-based speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1496 – 1507, 2010. 6

[45] T. de Freitas Pereira *et al.*, "Can face anti-spoofing countermeasures work in a real world scenario?" in *International Conference on Biometrics*, 2013. 6, 8

Ivana Chingovska received a Master degree in Electrical Engineering and Information Technology from Ss. Cyril and Methodius University, Skopje, Republic of Macedonia. Currently, she is a PhD student at École Polytechnique Fédérale de Lausanne (EPFL) and a research assistant at Idiap Research Institute, Switzerland. Her research is focused on biometrics and biometric spoofing detection.

André Anjos holds a Ph. D. in Signal Processing. Since 2010 he has a research position at the Idiap Research Institute, where his interests include biometrics, pattern recognition, computing and reproducible research. He is currently involved in various european and swiss projects with a core focus on Biometrics. He is one of the main authors of Bob, a freely distributed software toolkit for Biometrics. Finally, he also works as a reviewer for major journals in Biometrics, Forensics, Security, Signal and Image Processing and Pattern Recognition.