Joint Speaker Verification and Anti-Spoofing in the i-Vector Space

Aleksandr Sizov, Elie Khoury, *Member, IEEE,* Tomi Kinnunen, Zhizheng Wu, and Sébastien Marcel, *Member, IEEE*

***Abstract*—Any biometric recognizer is vulnerable to spoofing attacks and hence voice biometric, also called automatic speaker verification (ASV), is no exception; replay, synthesis and conversion attacks all provoke false acceptances unless countermeasures are used. We focus on voice conversion (VC) attacks considered as one of the most challenging for modern recognition systems. To detect spoofing, most existing countermeasures assume explicit or implicit knowledge of a particular VC system and focus on designing discriminative features. In this work, we explore back-end generative models for more generalized countermeasures. Specifically, we model synthesis-channel subspace to perform speaker verification and anti-spoofing jointly in the *i*-vector space, which is a well-established technique for speaker modeling. It enables us to integrate speaker verification and anti-spoofing tasks into one system without any fusion techniques. To validate the proposed approach, we study vocoder-matched and vocoder-mismatched ASV and VC spoofing detection on the NIST 2006 speaker recognition evaluation dataset. Promising results are obtained for standalone countermeasures as well as their combination with ASV systems using score fusion and joint approach.**

***Index Terms*—speaker recognition, spoofing, voice conversion attack, i-vector, joint verification and anti-spoofing.**

I. Introduction

Biometric person authentication [1] plays an increasingly important role in border control, crime prevention and personal data security. While the main biometric techniques (e.g. face, voice, fingerprints) can already handle noisy and mismatched sample comparisons robustly, recognizer vulnerability under malicious *spoofing attacks* remains a serious concern. Indeed, any biometric system has several weak links [2], the most accessible ones being sensor- and transmission-level attacks. Our application is text-independent speaker verification [3] that is used, for instance, to verify customer's identity in telephone banking and in protecting personal data in smartphones.

Speaker verification systems can be spoofed by four major types of attacks: replay, impersonation, speaker-adapted speech synthesis and voice conversion [4]. Because of its flexibility in direct transformation of speaker characteristics, we focus on voice conversion (VC) attacks. VC involves conversion of one speaker's (attacker) utterances towards the target speaker (client) having access to prior training utterance pairs from both [5]. By now, it is well-known that VC attacks pose a serious threat to any speaker verification system. Early studies [6], [7], [8], [9] showed this to be the case regarding traditional Gaussian mixture model (GMM) recognizers. Recent studies involving both text-independent [10], [11] and text-dependent [12] recognizers highlight that the problem persists even with modern recognizers, including *i*-vectors [13]. Interestingly, the quality of the converted voice does not have to be particularly high; even artificial signal attacks [14], [15] involving unintelligible speech can spoof a recognizer. Even if the modern recognizers might provide increased protection [10], [15], their false acceptance typically increases by considerable amount. This is easy to understand, remembering that speaker verification and VC methods use *matched* front- and back-end models, namely, Mel-frequency cepstral features and GMMs.

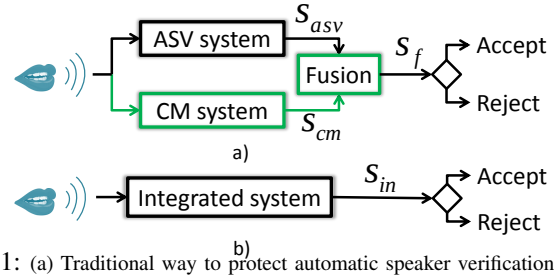


Fig. 1: (a) Traditional way to protect automatic speaker verification (ASV) from spoofing attacks is to independently develop ASV and countermeasure (CM) subsystems that are post-combined with score-level fusion. (b) Our core contribution is a *joint* approach that uses same *i*-vectors for both speaker verification and voice conversion attack detection. $s_{\mathbf{x}}$ means a score produced by system $\mathbf{x}$.

While the prior studies confirm the destructive nature of VC spoofing, much less work exists in designing *countermeasures* to safeguard recognizers from attacks. In principle, this involves two different subproblems. Firstly, spoofing attacks should be *detected*; while an ideal speaker verification system measures the strengths of *target* (same speaker) and *non-target* (different speaker) hypotheses, an ideal spoofing attack detector would assess the strengths of *human* and *non-human* hypotheses, where the latter refers to any tampering/transformation of real human speech or generation of synthetic speech. The second subproblem is to *integrate*

Aleksandr Sizov and Tomi Kinnunen are with School of Computing, University of Eastern Finland, Finland. (email: sizov@cs.uef.fi, tkinnu@cs.joensuu.fi). This work was partially supported from Academy of Finland (proj. no. 253120 and 283256).

Elie Khoury and Sébastien Marcel are with Idiap Research Institute, Switzerland. (email: elie.khoury@idiap.ch; sebastien.marcel@idiap.ch). Their work was supported by the Swiss National Science Foundation (SNSF) under the LOBI project.

Zhizheng Wu is with the Centre for Speech Technology Research, University of Edinburgh, United Kingdom. (email: zhizheng.wu@ed.ac.uk).

Manuscript received June xx, 2014; revised xx xx, 20xx.

the speaker verification and countermeasure opinions. In the general literature on biometric anti-spoofing (including speaker verification), this integration is usually achieved by cascading the countermeasure and the biometric detector or by merging their outputs by late (score) fusion. While this traditional score fusion of speaker verification and spoofing detectors is included in our experiments as a baseline, the core of our contribution (Fig. I) is in integration at the model level.

Regarding detection of VC spoofing attacks, the current solutions (Table I) are all based on feature design based on prior knowledge about the synthesis artefact traces a VC attack leaves to speech signals. For instance, [11], [19], [16] uses phase information known to be absent in the used voice coder technique while [20], [21] uses knowledge that dynamic variation in synthetic speech is reduced in comparison to natural speech. Such hand-crafted low-level features are necessarily designed to detect a particular attack which, however, can never be exactly known in advance. *Generalized* countermeasures, a recent direction in biometric anti-spoofing research, aim at detecting various types of attacks (e.g. synthesis, replay or VC attacks), for instance by modeling only in-class data or using enhanced features such cepstrogram texture [18]. Other biometric modalities, in fact, use also tailored front-end features such as [22], [23] and [24] to detect spoofing attacks.

In this work, we continue the quest for generalized countermeasures with an important distinction from all the prior work: rather than hand-crafting discriminative features to distinguish synthetic and natural utterances, we use the *same* acoustic front-end designed for automatic speaker verification, and focus on generative modeling of spoofing attacks. To this end, we assume that any speech utterance — irrespectively whether a natural or synthetic one — can be presented as a single feature vector ϕ (here, an *i*-vector [13]). We can think a synthetic utterance as a distorted version of a natural human speech passed through a very specific *synthesis channel*. Just like conventional speaker verification systems need to cope with speaker comparisons across varying channels (landline, cellular, close-talking mic), the synthesis-channel adds up another source of variation originating from spoofing attacks that utilize varying analysis-synthesis vocoders or conversion techniques. Even if all possible attacks cannot be known in advance, it is reasonable to assume that they share *some* common speaker-independent properties such as buzziness or discontinuity of F0. Such speaker-independent properties will be reflected in the cepstral features and the *i*-vectors derived from them, enabling detection.

Following the successful path in subspace modeling of inter-speaker and session/channel variations, it is natural to assume the synthesis-channel variations to reside in a low-dimensional subspace, too. Having a development pool of *i*-vectors (different from target or test speakers) derived from both natural and synthetic utterances, we can train independent subspace models for the corresponding natural and synthesis-channel variations. Specifically, inspired by the success of *probabilistic linear discriminant analysis* (PLDA) [25] in state-of-the-art speaker and face recognition, we adopt PLDA for joint modeling of speaker and synthesis channel variations. To the best of our knowledge, this is the first study to adopt PLDA for this novel use[1].

The proposed approach has several advantages. First, sharing the same front-end, integrating speaker verification and countermeasures is straightforward. Second, joint modeling approach retains low computational complexity since separate speaker verification and countermeasure systems are not required. Third, as a generative model, we expect good generalization to detect attacks not presented in training data — we provide a preliminary proof-of-concept utilizing a cross-vocoder evaluation protocol. Fourth, back-end modeling is not tied to a particular front-end or biometric modality; our hope is that researchers working on other biometric modalities find the general framework worth exploring in their application.

II. Database and Protocols

A. Dataset

In this work, we employ the spoofing attack dataset designed in [10], [11]. It is based on the core task "1conv4w-1conv4w" of the Speaker Recognition Evaluation 2006 (SRE06) corpus, which is a widely used standard benchmark database for text-independent speaker verification research. In the spoofing dataset, there are 9,440 gender-matched trials for evaluation, consisting of 3,946 genuine trials, 2,747 impostor trials, and 2,747 impostor trials after VC. We consider two different voice conversion methods: the popular joint-density Gaussian mixture model (JD-GMM) based method [27], and a simplified frame selection (FS) method as detailed in [11]. More details of the dataset design process can be found in [10], [11]. In the JD-GMM conversion, we consider two feature representations, namely Mel-cepstral analysis based features (MCEP) and linear predictive coding based features (LPC), while in the FS conversion, only MCEP features are considered. The difference between JD-GMM and FS conversion is that JD-GMM modifies source features to match that of a target speaker, while FS uses the target speaker features directly to generate converted speech. The repartition of trials for female and male speakers are presented in Table II.

TABLE II: Statistics of the spoofing dataset used in this work. MCEP and LPC refer to Mel cepstral based VC and linear predictive coding based VC.

	Male	***Female***	***Total***
Target speakers	241	342	583
Genuine trials	1,614	2,332	3,946
Impostor trials	1,132	1,615	2,747
MCEP impostor trials	1,132	1,615	2,747
LPC impostor trials	1,132	1,615	2,747
FS impostor trials	1,132	1,615	2,747

B. Conditions

To study the generalization ability of a countermeasure, we define "*matched*" and "*mismatched*" spoof conditions as follow:

[1]Preliminary results are presented in [26]. The current study extends it by developing several new integrated PLDA variants, including a two-stage strategy to train PLDA subspace parameters. The experiments include additional VC technique, ASV systems and score fusion techniques and extended analyses of the results. The theory part and literature review are also expanded considerably to make a self-contained description.

TABLE I: Summary of anti-spoofing approaches to voice conversion and speech synthesis spoofing in the literature. The results are not comparable, as they are using different benchmarking dataset and different protocol. Most of the prior work uses matched conditions whereas this paper considers mismatch conditions.

Countermeasure	**Feature / Model based**	**ASV system**	**FAR(%)**		
			Before spoofing	After spoofing	With CMs
Relative phase shift [16]	Feature	GMM-UBM	0.28	86	2.5
Relative phase shift [16]	Feature	SVM	0.00	81	2.5
Modified group delay [11]	Feature	JFA	3.24	41.25	1.71
Modified group delay [11]	Feature	PLDA	2.99	32.54	1.64
Local binary pattern [17]	Feature	FA	5.60	n/a	1.60
Local binary pattern [18]	Feature	PLDA	3.03	n/a	4.10
Proposed i-vector joint PLDA	Model	PLDA	0.55	14.17	2.97

- **Matched spoof condition:** This is the most studied case in the literature. It assumes that the user has prior knowledge about the vocoding technique of the VC attacks. For example, if the test set contains trials with MCEP-coded VC, the user may use MCEP-coded synthetic speech to design the countermeasure.
- **Mismatched spoof condition:** This was usually neglected in previous work [28]. It assumes that the system designer is prepared to a specific type of spoofing, but the attacks are from a different type.

Table IV presents the spoofing detection error on all possible matched and mismatched conditions for both genders.

In practice, we use the SPTK toolkit[2] to perform MCEP and LPC analysis and synthesis. Similar to [19] and [21], a copy-synthesis approach is employed to generate the MCEP- and LPC-coded speech for training the spoofing detector without undergoing any specific VC technique. That is, we first decompose a speech signal into its Mel-cepstral (or LPC) and fundamental frequency (F0) parameters and then reconstruct an approximated signal directly from these parameters. The reconstructed replica is the special version of the orignal signal passing through synthesis-channel, and in general will be close to the original signal but not exactly the same due to the lossy analysis-synthesis model; perceptually, a buzzy or muffled voice quality can be observed. Such copy-synthesis is a straightforward way to generate training samples for spoofing detection without, however, involving the computationally demanding stochastic VC part, which additionally requires selection of source-target speaker pairs and parallel training set. The copy-synthesis speech of SRE04, SRE05 and SRE06 is generated for both MCEP and LPC. We name the generated corpus as "synthesis" training set, in contrast to the original "natural" training set.

C. Evaluation metrics

We use the notion of *false rejection rate* (FRR) and *false acceptance rate* (FAR) to build all our metrics. False rejection happens when a target speaker is erroneously classified as an impostor. False acceptance is the opposite case when an impostor is misclassified as a target. The evaluation of the ASV system is done in terms of both LICIT and SPOOF protocols [29]. The LICIT protocol, involving *zero-effort* impostors, is the typical evaluation protocol used in verification scenarios, whereas the SPOOF protocol contains *only* spoofed impostors and is used to evaluate system performance when spoofing attacks are present.

To evaluate the ability of the systems to resist attacks, we put more emphasis on FAR. We use *zero-effort FAR* (ZFAR) as a metric for the LICIT protocol and *spoofing FAR* (SFAR) [29] for the SPOOF protocol. The only difference between them is that the former considers zero-effort impostors while the latter considers their spoofed versions. To make ZFARs and SFARs comparable across different systems, we compute them at the threshold when FRR of the system in question equals 1%. We also compute *equal error rate* (EER) — when FRR is equal to FAR — on all test trials pooled together as a single summary.

To independently evaluate the countermeasure performance we use the *spoofing detection error* (SDE):

$$\mathrm{SDE} = \frac{\mathrm{FP}+\mathrm{FN}}{\mathrm{P}+\mathrm{N}}, \tag{1}$$

where FP is the number of samples erroneously classified as positive (i.e. natural speech), FN the number of samples erroneously classified as negative (i.e. spoofing attacks), P is the total number of positive samples and N is the total number of negative samples.

III. Standalone Speaker Verification Systems

During standalone automatic speaker verification (ASV) for each test utterance $\mathcal{O}$ we compute two hypotheses: either $\mathcal{O}$ is produced by the target speaker $\mathcal{X}$ — $H_{\mathcal{X}}$, or it is not — $H_{\overline{\mathcal{X}}}$. Usually the score of the system is the log-likelihood ratio between the probabilities of both hypotheses:

$$s_{\text{asv}} = \log p(\mathcal{O}|H_{\mathcal{X}}) - \log p(\mathcal{O}|H_{\overline{\mathcal{X}}}). \tag{2}$$

Examples of this scoring rule would be given for the GMM-UBM and PLDA systems.

In this paper we consider three systems for the ASV task: GMM-UBM system based on the MFCC vectors [30], cosine scoring [13] and Probabilistic Linear Discriminant Analysis (PLDA) model [25] based on *i*-vectors.

A. GMM-UBM framework

The key component of the GMM-UBM system is the *Univeral background model* (UBM) that is a large GMM (Gaussian Mixture Model) trained on a diverse dataset to be speaker- and text-independent. This UBM represents the universal impostor. For each speaker in the enrolment set we concatenate all the UBM mean vectors into a supervector

[2] http://sp-tk.sourceforge.net/

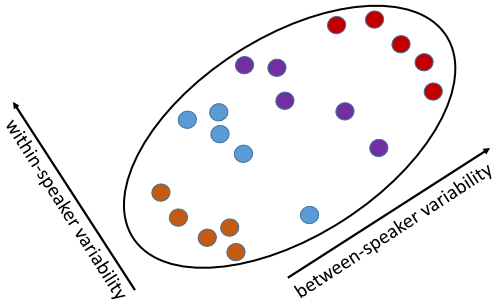


Fig. 2: Distribution of the different speech utterances in a PLDA model. Each utterance is represented by a circle, each color represents a particular speaker. The ellipse is a contour of the Gaussain distribution $\mathcal{N}(\mathbf{0}, \mathbf{V}\mathbf{V}^{\mathsf{T}}+\mathbf{U}\mathbf{U}^{\mathsf{T}}+\mathbf{\Sigma})$ from Eq. (10).

and adapt it to produce a target speaker model. During the verification stage, we compute the score in the form of the log-likelihood ratio (LLR) between hypothesized speaker model and UBM model:

$$s_{\text{ubm}}(\mathbf{X}_{\text{test}}) = \log p(\mathbf{X}_{\text{test}}|\boldsymbol{\theta}_{\text{spk}}) - \log p(\mathbf{X}_{\text{test}}|\boldsymbol{\theta}_{\text{ubm}})\,, \quad (3)$$

where $\mathbf{X}_{\text{test}} = \{\mathbf{x}^1_{\text{test}}, \ldots, \mathbf{x}^N_{\text{test}}\}$ is a sequence of MFCC features for the test utterance $\mathcal{O}$, $\boldsymbol{\theta}_{\text{spk}}$ and $\boldsymbol{\theta}_{\text{ubm}}$ are the GMM parameters for the corresponding model. In this work, we use linear approximation of Eq. (3) proposed in [31].

B. Total variability framework

The total variability paradigm is built upon GMM framework and its aim is to extract a low-dimensional vectors, so-called *i-vectors*, that are a compact version of the GMM mean supervectors:

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{w}\,, \quad (4)$$

where $\mathbf{m}$ is a UBM mean supervector, $\mathbf{s}$ is a target GMM mean supervector, $\mathbf{T}$ is a total variability matrix that encompass both speaker- and channel-variability and $\mathbf{w}$ is a latent variable with the standard normal distribution. For each speaker utterance $\mathcal{O}$, an *i*-vector $\boldsymbol{\phi}$ is produced as a MAP (*maximum-a-posteriori*) estimate of the variable $\mathbf{w}$. Section VI presents the specific parameter values used for the *i*-vector extraction.

To achieve a higher recognition accuracy we map i-vectors into a more discriminative subspace with the following pre-processing algorithms: (1) *radial Gaussianization* [32], which consists of whitening and length-normalization, to reduce non-Gaussian effects [33] as well as mismatch between training and testing subsets, (2) *linear discriminant analysis* (LDA) to learn a linear projection that maximizes between-class variations while minimizing within-class variations.

The first *i*-vector based scoring system that we consider is the **cosine scoring**. It is a very simple and fast method that uses cosine kernel to compute the score between target and test *i*-vectors:

$$s(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}) = \frac{\langle \boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}} \rangle}{\|\boldsymbol{\phi}_{\text{target}}\| \cdot \|\boldsymbol{\phi}_{\text{test}}\|}\,, \quad (5)$$

where $\langle\cdot\rangle$ is a dot product and $\|\cdot\|$ stands for the Euclidean norm. The cosine scoring does not require any training data and does not have any inter-speaker or intra-speaker variability models. The next system that we consider — PLDA model — is a more advanced generative parametric approach that needs training with same versus different speaker labels.

PLDA is a probabilistic framework that models both between- and within-speaker variability. In this study we use several different PLDA variants due to their appealing properties: they allow us to perform session compensation, to regulate model complexity given the amount of training data, and to naturally generate LLR scores.

The **standard** PLDA was introduced in [25]. It assumes that the j-th i-vector $\boldsymbol{\phi}_{i,j}$ of a client i is generated as

$$\boldsymbol{\phi}_{ij} = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_{ij} + \boldsymbol{\varepsilon}_{ij}\,, \quad (6)$$

$$\mathbf{y}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\,, \quad (7)$$

$$\mathbf{x}_{ij} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})\,, \quad (8)$$

$$\boldsymbol{\varepsilon}_{ij} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})\,, \quad (9)$$

where $\boldsymbol{\phi}_{ij} \in \mathbb{R}^{D\times 1}$, $\boldsymbol{\mu}$ is a mean vector, columns of the matrices $\mathbf{V} \in \mathbb{R}^{D\times P}$ and $\mathbf{U} \in \mathbb{R}^{D\times M}$ span the between- and within-speaker subspaces, $\mathbf{y}_i$ and $\mathbf{x}_{ij}$ are their corresponding latent variables, $\boldsymbol{\varepsilon}_{ij}$ is a latent residual noise variable distributed with a diagonal covariance matrix $\mathbf{\Sigma}$. Let us denote the parameters of the PLDA model collectively as $\boldsymbol{\theta}$.

Because PLDA is a linear-Gaussian model, we can integrate out its latent variables to receive another Gaussian distribution [34, Chapter 12]:

$$\begin{aligned} p(\boldsymbol{\phi}_{ij}|\boldsymbol{\theta}) &= \iint p(\boldsymbol{\phi}_{ij}|\mathbf{y}_i, \mathbf{x}_{i,j}, \boldsymbol{\theta}) p(\mathbf{y}_i) p(\mathbf{x}_{ij})\, d\mathbf{y}_i\, d\mathbf{x}_{i,j} \\ &= \mathcal{N}(\boldsymbol{\phi}_{ij}|\boldsymbol{\mu}, \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{U}\mathbf{U}^{\mathsf{T}} + \mathbf{\Sigma}). \end{aligned} \quad (10)$$

The resultant distribution is depicted in Fig. 2.

Initially, PLDA was proposed for face recognition where the input features are very high-dimensional and the training datasets are not large enough to robustly estimate full within- and between-individual subspaces (without dimensionality reduction). In speaker recognition the usual input features for PLDA are 100 to 600 dimensional *i*-vectors, making it possible to not apply any dimensionality reduction. The most popular form of such PLDA variants is the **simplified PLDA** [33]:

$$\boldsymbol{\phi}_{i,j} = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \boldsymbol{\varepsilon}'_{i,j}\,. \quad (11)$$

The only difference from the standard PLDA is that now the channel subspace $\mathbf{U}$ is absorbed into a *full* covariance residual noise matrix $\mathbf{\Sigma}'$.

If we go even further and set both subspace matrices $\mathbf{V}$ and $\mathbf{U}$ to have the full rank we get so called **two-covariance model** [35]. Comparison of all three PLDA models as well as the EM-algorithms to train them are presented in [36].

After we have trained the parameters of the PLDA model, we perform a speaker verification task as follows: for a pair of target and test *i*-vectors $(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}})$, we compute LLR for the probability that they share the same latent identity variable $\mathbf{y}_i$ and, hence, originate from the same person; and the probability that they have different latent identity variables $\mathbf{y}_t$ and $\mathbf{y}_i$ and, therefore, are from different persons:

$$s_{\text{sv}}(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}) = \log \frac{p(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}|\boldsymbol{\theta})}{p(\boldsymbol{\phi}_{\text{target}}|\boldsymbol{\theta}) p(\boldsymbol{\phi}_{\text{test}}|\boldsymbol{\theta})}\,. \quad (12)$$

IV. Standalone Anti-spoofing Systems

Spoofing detection is a binary classification task that aims at isolating prepared attacks from natural zero-effort (both genuine and impostor) trials. For each test utterance $\mathcal{O}$ we compute two hypotheses: either $\mathcal{O}$ is a natural speech $\mathcal{N}$ — $H_{\mathcal{N}}$, or it is not (i.e. synthetic speech) — $H_{\overline{\mathcal{N}}}$.

When dealing with voice conversion (VC) attacks, one may look at the problem from a low-level signal processing point of view and solve it by using prior knowledge about the VC technique (e.g. absence of the phase modeling) as detailed in the introduction. In this study, however, we perform the VC detection task directly in the *i*-vector space and evaluate four different classification methods: the cosine scoring [13], the simplified PLDA (11), *support vector machines* (SVM) with linear kernel [37] and two-stage PLDA introduced in section V-B.

To train these classification methods, we use the extended "natural" + "synthesis" training set — every natural speech utterance has its corresponding vocoded (synthetic) versions. This way for every human speaker we create one or more additional synthetic speakers who have simulated MCEP- and LPC-vocoded versions of the original utterances. Fig. 3 shows *i*-vectors for one typical speaker together with their vocoded versions. It indicates that the MCEP-vocoded *i*-vectors are much closer to those of natural speech in comparison to LPC-vocoded *i*-vectors. This is reasonable since the MCEP vocoder and the *i*-vector extractor (trained on MFCCs) have closely matched signal processing steps.

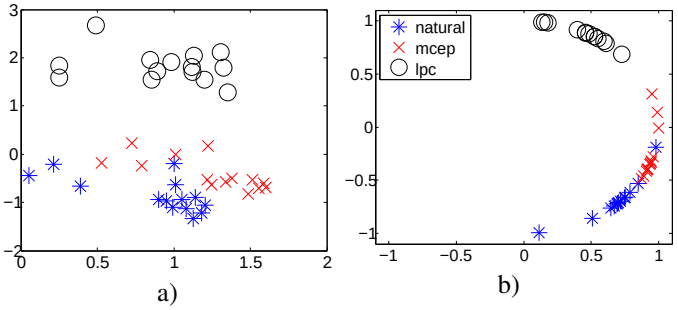


Fig. 3: (a) Natural and vocoded *i*-vectors for one speaker after whitening and LDA projection into two dimensions. All of them are used to train both countermeasures (section IV) and integrated PLDA systems (section V). (b) The same *i*-vectors after length-normalization.

For the cosine-based classifier, first we compute cosine scores (5) between test *i*-vector and every *i*-vector in the training set. Then we select maximum scores among natural and synthetic subsets of the training set. We use these two numbers as the scores for the corresponding hypotheses.

We form natural and synthetic classes from the training set to train both Simplified PLDA and SVM classifiers [37]. During the scoring stage of the Simplified PLDA we take the average of both classes to form two *i*-vectors: one represents natural utterances and the other represents synthetic utterances. Then we take one of them at a time, add test *i*-vector and plug them into the PLDA equation for the log likelihood ratio (12). This way we get the scores for $H_{\mathcal{N}}$ and $H_{\overline{\mathcal{N}}}$ hypotheses.

Two-stage PLDA and its training algorithm are described in the subsection V-B and the Appendix. During the scoring stage, for each test *i*-vector we calculate log likelihood ratio between natural and synthetic hypotheses:

$$s_{\text{cm}}(\boldsymbol{\phi}_{test}) = \log \frac{p(\boldsymbol{\phi}_{test}|H_{\mathcal{N}})}{p(\boldsymbol{\phi}_{test}|H_{\overline{\mathcal{N}}})} = \log \frac{\mathcal{N}(\boldsymbol{\phi}_{test}|\boldsymbol{\mu}_1, \mathbf{V}_1\mathbf{V}_1^{\mathsf{T}} + \boldsymbol{\Sigma})}{\mathcal{N}(\boldsymbol{\phi}_{test}|\boldsymbol{\mu}_2, \mathbf{V}_2\mathbf{V}_2^{\mathsf{T}} + \mathbf{U}_2\mathbf{U}_2^{\mathsf{T}} + \boldsymbol{\Sigma})}\,, \quad (13)$$

where $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are mean vectors of natural and vocoded training data, matrices $\mathbf{V}_1$ and $\boldsymbol{\Sigma}$ are trained on the natural speech and, thus, are used to estimate the probability of the test *i*-vector under "natural" PLDA model, while matrices $\mathbf{V}_2$ and $\mathbf{U}_2$ are trained on the vocoded speech and used to estimate the probability under "synthesis" PLDA model. We use Eq. (10) to compute these probabilities.

V. Joint Speaker Verification and Anti-spoofing

Let us now consider the joint approach, which means that we do speaker verification and anti-spoofing at once. In this case, each test utterance $\mathcal{O}$ has two attributes: indicator of the target speaker — $\mathcal{X}$ and indicator of the natural speech — $\mathcal{N}$. The null hypothesis $H_{(\mathcal{X},\mathcal{N})}$ is that the test utterance is a natural speech utterance from the target speaker. The complementary hypothesis $H_{\overline{(\mathcal{X},\mathcal{N})}}$, in turn, is a union of the other three classes:

$$H_{\overline{(\mathcal{X},\mathcal{N})}} = H_{(\overline{\mathcal{X}},\mathcal{N})} \cup H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})} \cup H_{(\mathcal{X},\overline{\mathcal{N}})}\,, \quad (14)$$

where the first term on the RHS corresponds to the zero-effort impostor trial and the second term to a spoofed (dedicated) impostor attempt. The class $(\mathcal{X},\overline{\mathcal{N}})$, referring to the case when a cooperative genuine user would like to not authenticate him/herself, is meaningless in an authentication context so we do not consider it further.

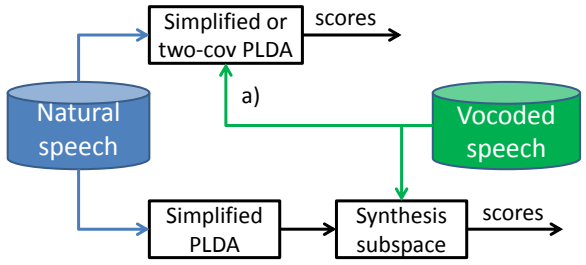


Fig. 4: This scheme shows the allocation of the training data for the integrated PLDA models: (a) one-stage PLDA, (b) two-stage PLDA.

The systems that we evaluate can be broadly classified into two groups: (1) **Score fusion** of two separate blocks: automatic speaker verification (ASV) and countermeasure (CM) systems. (2) **Integrated approach** in which we model both natural and synthetic variability at once. We divide the latter further into one-stage and two-stage models. The difference (see Fig. 4) is that during two-stage approach we have separate steps for natural and vocoded datasets so that we are able to infer *synthesis channel* variability during the second step. This variability corresponds to some speech artifacts induced by the vocoders.

A. Score fusion

Score fusion aims at combining the output scores from both ASV and CM systems. In this approach, ASV system compares positive hypothesis $H_{(\mathcal{X},\mathcal{N})}$ against negative hypothesis $H_{(\overline{\mathcal{X}},\mathcal{N})}$ and CM system compares positive hypothesis $H_{(\mathcal{X},\mathcal{N})} \cup H_{(\overline{\mathcal{X}},\mathcal{N})}$ against negative hypothesis $H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})}$. The positive hypothesis for the joint system is the intersection of the positive hypotheses for both subsystems — $(\mathcal{X},\mathcal{N})$.

To fuse the scores we compute a weighted sum, where the weights are estimated using logistic regression. Logistic regression is one of the most powerful score fusion techniques, which has been successfully employed for combining heterogeneous speaker classifiers [38], [39]. Given a pair of target and test *i*-vectors $(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}})$ and output scores from ASV and CM systems — $s_{\text{asv}}(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}})$ and $s_{\text{cm}}(\boldsymbol{\phi}_{\text{test}})$ — we fuse those scores inside the logistic function:

$$s_f(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}|\,\boldsymbol{\beta}) = g\left(\beta_0 + \beta_1 s_{\text{asv}}(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}) + \beta_2 s_{\text{cm}}(\boldsymbol{\phi}_{\text{test}})\right), \quad (15)$$

where $g(x) = 1/(1+\exp(-x))$ is a logistic sigmoid function and $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2]$ are the regression coefficients, determined by maximum likelihood estimation on a subset of 1000 utterances (+ their synthetic versions) randomly selected from the SRE06 background training set. The optimization is done using the *conjugate-gradient* algorithm [40].

B. Integrated systems

All experiments with the integrated systems are carried out in the *i*-vector space (see subsection III-B).

One-stage PLDA system has the same structure as the baseline simplified PLDA system but it uses the extended "natural" + "synthesis" training set (section IV).

The score of the one-stage integrated system is the standard PLDA log-likelihood ratio (12) but this time we compare $H_{(\mathcal{X},\mathcal{N})}$ hypothesis against $H_{\overline{(\mathcal{X},\mathcal{N})}}$. We are able to check both zero-effort impostor and spoofed impostor hypotheses at once because, after we have added the vocoded data to the training set, the model is able to create more adequate within- and between-speaker variability subspaces. This way we increased the number of impostors that the system can handle. Now the spoofed utterance will be treated as an impostor.

Two-stage PLDA system. In [33], P. Kenny faced the problem how to robustly estimate parameters of the PLDA model when data involves both telephone and microphone speech utterances. For this reason he first estimated PLDA parameters on the telephone dataset and then trained additional channel subspace on the microphone data only. Inspired by this strategy, we propose to train PLDA in two stages as well. Instead of telephone and microphone channels, however, we make a distinction between natural and synthetic speech. At the first stage, we train a simplified PLDA model only on the natural speech:

$$\boldsymbol{\phi}_{i,j} = \boldsymbol{\mu}_1 + \mathbf{V}_1\mathbf{y}_i + \boldsymbol{\varepsilon}_{i,j}\,, \quad (16)$$

then, on the second stage, we estimate new mean vector and add a *synthesis channel* subspace $\mathbf{U}_2$ and train it *only* on the re-synthesized speech (matrix $\mathbf{V}_1$ is also changed to the $\mathbf{V}_2$ during the optimization of the $\mathbf{U}_2$, the details can be found in the Appendix). The final model, therefore, has the same form as a standard PLDA with the difference that the residual covariance matrix $\boldsymbol{\Sigma}$ is now full:

$$\boldsymbol{\phi}_{i,j} = \boldsymbol{\mu}_2 + \mathbf{V}_2\mathbf{y}_i + \mathbf{U}_2\mathbf{x}_{i,j} + \boldsymbol{\varepsilon}_{i,j}\,. \quad (17)$$

Two-stage PLDA allows us to explicitly check both zero-effort impostor hypothesis $H_{(\overline{\mathcal{X}},\mathcal{N})}$ and spoofed impostor hypothesis $H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})}$.

Zero-effort impostor hypothesis $H_{(\overline{\mathcal{X}},\mathcal{N})}$ assumes that the target and the test utterance *i*-vectors originate from different speakers — they have different latent speaker variables — but they both are natural speech utterances. In this case, we should use the parameters computed during the first stage:

$$\begin{bmatrix}\boldsymbol{\phi}_{\text{target}}\\ \boldsymbol{\phi}_{\text{test}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_1\end{bmatrix} + \begin{bmatrix}\mathbf{V}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_1\end{bmatrix}\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{bmatrix} + \begin{bmatrix}\boldsymbol{\varepsilon}_1\\ \boldsymbol{\varepsilon}_2\end{bmatrix}\,. \quad (18)$$

Spoofed impostor hypothesis $H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})}$ also assumes that the target and the test *i*-vectors originate from different speakers[3]. But the difference is that now we consider the test utterance to be a result of a spoofing attack. Thus, we expect them to have a mismatch in the *synthesis channel* subspace $\mathbf{U}_2$:

$$\begin{bmatrix}\boldsymbol{\phi}_{\text{target}}\\ \boldsymbol{\phi}_{\text{test}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}_2\\ \boldsymbol{\mu}_2\end{bmatrix} + \begin{bmatrix}\mathbf{V}_2 & \mathbf{0} & \mathbf{U}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_2 & \mathbf{0} & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\\ \mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix} + \begin{bmatrix}\boldsymbol{\varepsilon}_1\\ \boldsymbol{\varepsilon}_2\end{bmatrix}\,. \quad (19)$$

The positive hypothesis $H_{(\mathcal{X},\mathcal{N})}$ assumes that both utterances belong to the same target speaker:

$$\begin{bmatrix}\boldsymbol{\phi}_{\text{target}}\\ \boldsymbol{\phi}_{\text{test}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_1\end{bmatrix} + \begin{bmatrix}\mathbf{V}_1\\ \mathbf{V}_1\end{bmatrix}\mathbf{y} + \begin{bmatrix}\boldsymbol{\varepsilon}_1\\ \boldsymbol{\varepsilon}_2\end{bmatrix}\,. \quad (20)$$

The overall score of the two-stage PLDA model is similar to the score of the standard PLDA model (12). It is also a log-likelihood ratio between probabilities for positive and negative hypotheses. The difference is that now we have two negative hypotheses: $H_{(\overline{\mathcal{X}},\mathcal{N})}$ and $H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})}$, so we use $\max(\cdot)$ operator to select the most likely one:

$$s(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}) = \quad (21)$$
$$\log \frac{p(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}|H_{(\mathcal{X},\mathcal{N})})}{\max\left(p(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}|H_{(\overline{\mathcal{X}},\mathcal{N})}),\ p(\boldsymbol{\phi}_{\text{target}}, \boldsymbol{\phi}_{\text{test}}|H_{(\overline{\mathcal{X}},\overline{\mathcal{N}})})\right)}\,.$$

To compute the probabilities in Eq. (21), we use composite matrices to rewrite Eq. (18) -(20) in a form of a factor analysis models [41]:

$$\boldsymbol{\phi}' = \boldsymbol{\mu}' + \mathbf{A}\mathbf{z} + \boldsymbol{\varepsilon}'\,, \quad (22)$$

and then apply Eq. (10).

To help the reader re-produce this procedure, we provide an open-source reference implementation (see Conclusions for the pointer).

[3]We also tried the case of a "perfect" spoofing, when it is so good that the latent speaker variables are the same, but it has not worked out for our setup.

VI. Experimental Results

The full experiments were carried out using the open-source speaker recognition toolbox[4] which is a modification of the toolbox *Spear*[5] [42]. Acoustic features are extracted at equally-spaced time instants using a sliding window approach. First, a simple energy-based voice activity detection (VAD) is performed to discard the non-speech parts. Second, 19 MFCC and log energy features together with their first- and second-order derivatives are computed over 20 ms Hamming windowed frames every 10 ms. Finally, utterance-level cepstral mean and variance normalization (CMVN) is applied on the resulting 60-dimensional feature vectors.

After feature extraction, the training of the UBM, the $\mathbf{T}$ subspace and the whitening matrix is done once for all systems, using Fisher, Switchboard, SRE04, SRE05 and SRE06 corpora (from which the enrolment and test data used in our experiments were excluded). The UBM model is composed of 2048 Gaussian components and the rank of the total variability matrix $\mathbf{T}$ is set to 600. It is worth noting that both natural and synthetic speech utterances undergo exactly the same procedure of feature and *i*-vector extraction.

A. Standalone speaker verification results

TABLE III: Performance summary of the standalone speaker verification systems on SRE06 speech conversion database. For evaluation we use the following metrics: *equal error rate* (EER, %) and *zero-effort false acceptance rate* (ZFAR, %) on LICIT protocol. To make ZFARs comparable with the following experiments we compute them at the threshold when *false rejection rate* of the particular system is equal to 1%.

	Female		***Male***	
	EER	*ZFAR*	*EER*	*ZFAR*
GMM	13.38	60.37	12.64	66.61
Cosine	3.42	14.99	4.59	18.82
Simplified PLDA	**0.81**	**0.62**	**0.54**	**0.44**

In this experiment, we evaluate three automatic speaker verification (ASV) techniques: GMM-UBM system and two *i*-vector systems: cosine scoring and simplified PLDA. The PLDA model is trained only on SRE04, SRE05 and SRE06 corpora, without considering the synthetic speech. Table III shows the results on the LICIT protocol, while Fig. 5 illustrates DET plots on both LICIT and SPOOF protocols for both genders. These results clearly show that the simplified PLDA system (*Simp-PLDA*) is superior to the other two speaker verification systems in all cases: EER is at least four times lower compared to the cosine scoring for both genders and there is a considerable gap between DET curves on both protocols. In the remaining experiments, this baseline *Simp-PLDA* system will be used for score fusion and as a reference for system comparison.

B. Standalone anti-spoofing results

In this experiment, we evaluate four back-end countermeasure (CM) techniques: *Cosine*, linear *SVM*, *Simp-PLDA* and *two-stage PLDA*. The training set consists of positive samples (synthetic speech) and negative samples (natural speech).

[4]https://pypi.python.org/pypi/xspear.fast_plda
[5]https://pypi.python.org/pypi/bob.spear

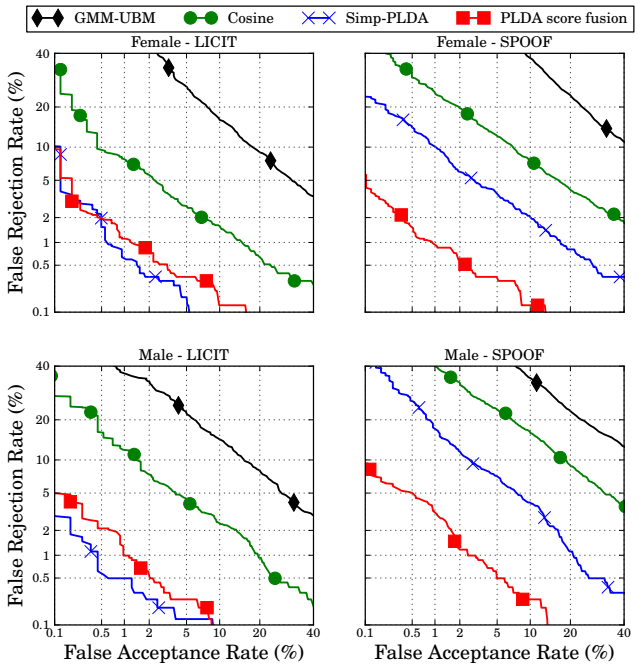


Fig. 5: DET curves of the speaker verification systems. Results are for both female and male trials, and on both LICIT and SPOOF (pooled MCEP- and LPC-vocoded trials).

Table IV reports the spoofing detection error (SDE) on all possible matched and mismatched conditions and for both female and male trials. Clearly, the LPC-coded voice conversion (VC) attacks are easy to detect even in the mismatch case (MCEP vs LPC) where the SDE rates for all systems are less or equal than 1.98% and 3.22% on female and male trials, respectively. In contrast, as illustrated in Fig. 3, MCEP-coded attacks are relatively difficult to detect, especially for mismatch case (LPC vs JD-MCEP), where *cosine scoring* shows the best results for both genders (14.83% and 14.86%). Both *SVM* and *Simp-PLDA* systems failed to generalize for mismatch cases and would not be considered in the following experiments on score fusion.

C. Joint speaker verification and anti-spoofing results

To evaluate our proposed integrated systems, we compare them with the traditional score fusion that combines the scores provided by both ASV and CM systems. Fig. 6 illustrates the scatter plot of the scores of the genuine accesses, zero-effort impostors and spoofing attacks. The three classes are fairly well separated which suggests that score fusion is a good candidate for joint operation of verification and spoofing.

Table V reports the performance of several joint systems for all training conditions and all attack methods. We observe the following:

1) The baseline system *always* reaches the best performance on the LICIT protocol: the corresponding ZFARs for both genders are the lowest ones. This is not surprising since

TABLE IV: Comparison of stand-alone countermeasures. This table shows the spoofing detection error rate SDE (%) on both female and male trials for cosine, SVM, Simplified PLDA (S-PLDA) and two-stage PLDA (2-stg PLDA) classifiers. The second column of the table shows the type of the features we used for the copy-synthesis approach to get vocoded speech for training. They are either Mel-cepstral analysis features (MCEP) or linear predictive coding features (LPC). The third column specifies the attack: we consider two voice conversion techniques, namely, joint-density Gaussian mixture model (JD-GMM) and simplified frame selection (FS). JD-GMM is applied for both MCEP and LPC features. The corresponding methods are called JD-MCEP and JD-LPC. We use only MCEP features for FS conversion.

Conditions	Attacks prepared for	Test attacks	***Female***				***Male***			
			Cosine	SVM	S-PLDA	2-stg PLDA	Cosine	SVM	S-PLDA	2-stg PLDA
Matched conditions	MCEP	JD-MCEP	**3.18**	6.9	8.24	3.52	3.69	**1.80**	2.70	2.13
	MCEP	FS-MCEP	1.81	0.55	**0.29**	1.14	3.28	0.98	**0.61**	1.72
	LPC	JD-LPC	0.64	0.55	**0.35**	0.70	1.11	0.66	**0.37**	1.19
	MCEP+LPC	JD-MCEP	**3.09**	12.76	20.54	**3.00**	4.14	4.18	6.92	**2.83**
	MCEP+LPC	JD-LPC	1.54	**0.55**	0.57	1.19	3.15	1.23	**0.57**	2.21
	MCEP+LPC	FS-MCEP	1.95	0.73	**0.32**	1.25	3.52	1.39	**0.70**	2.17
Mismatched conditions	MCEP	JD-LPC	1.98	1.98	1.19	**1.17**	3.32	0.86	**0.66**	1.76
	LPC	JD-MCEP	**14.83**	46.55	46.55	16.96	**14.86**	39.56	44.96	19.7
	LPC	FS-MCEP	**10.14**	31.84	30.15	6.26	14.62	29.07	32.97	**11.92**

TABLE V: Performance summary of the joint authentication and anti-spoofing systems. In this table we evaluate attacks performed by two voice conversion techniques: joint-density Gaussian mixture model (JD-GMM) and simplified frame selection (FS). For JD-GMM we consider two feature representations: Mel-cepstral analysis features (MCEP) and linear predictive coding features (LPC). The corresponding methods are called JD-MCEP and JD-LPC. We use only MCEP features for FS conversion. Vocoded speech for training is produced by copy-synthesis approach. We use the following metrics for evaluation: *equal error rate* (EER, %) on all test trials pooled together as a generalized estimator of the system, *zero-effort false acceptance rate* (ZFAR, %) on LICIT protocol and *spoofing FAR* (SFAR, %) for each attack method on SPOOF protocol. To make ZFARs and SFARs comparable across different systems we compute them at the threshold when *false rejection rate* of the particular system is equal to 1%. We use two-stage PLDA as the second score fusion method. The baseline systems are the same for all three training cases. We reproduce them for the sake of convenience. The integrated systems are highlighted in gray.

Joint system	***Female***					***Male***				
	EER	*ZFAR*	*SFAR* **JD-MCEP**	**JD-LPC**	**FS-MCEP**	*EER*	*ZFAR*	**JD-MCEP**	*SFAR* **JD-LPC**	**FS-MCEP**
Training on natural and MCEP-vocoded speech										
Baseline	3.72	**0.62**	7.12	10.65	33.93	5.39	**0.44**	7.51	8.66	46.29
Score fusion (cosine)	2.66	1.05	2.85	4.40	20.43	4.22	0.88	3.09	4.06	37.72
Score fusion (PLDA)	**0.94**	1.05	**0.31**	**0.06**	**1.24**	**1.29**	1.33	**0.35**	**0.00**	**4.59**
Simplified PLDA	2.65	0.68	4.03	4.40	17.28	4.54	**0.44**	4.51	4.33	34.28
Two-cov PLDA	2.28	0.87	3.28	3.53	14.37	4.33	0.88	5.48	4.95	35.25
Two-stage PLDA	1.06	2.29	0.5	0.12	1.3	1.91	2.12	1.24	0.44	11.13
Training on natural and LPC-vocoded speech										
Baseline	3.72	**0.62**	7.12	10.65	33.93	5.39	**0.44**	7.51	8.66	46.29
Score fusion (cosine)	2.96	0.93	3.59	5.68	23.34	4.33	0.71	3.27	4.68	38.78
Score fusion (PLDA)	1.73	0.93	**1.86**	**0.00**	9.85	**2.99**	0.71	**2.74**	**0.00**	**19.79**
Simplified PLDA	3.38	0.87	7.25	3.90	29.97	5.31	0.53	9.19	3.98	46.73
Two-cov PLDA	3.00	1.18	6.07	2.72	25.33	4.95	0.71	10.25	3.98	47.26
Two-stage PLDA	**1.51**	2.79	2.66	**0.00**	**7.06**	3.19	2.03	5.92	**0.00**	22.62
Training on natural, MCEP- and LPC-vocoded speech										
Baseline	3.72	**0.62**	7.12	10.65	33.93	5.39	**0.44**	7.51	8.66	46.29
Score fusion (cosine)	2.83	1.05	3.22	5.02	22.10	4.28	0.88	3.18	4.33	38.25
Score fusion (PLDA)	**0.99**	1.18	0.62	**0.00**	2.11	**1.54**	0.97	**0.44**	**0.00**	**6.89**
Simplified PLDA	2.53	0.99	5.20	2.79	19.26	4.40	0.53	4.77	2.83	36.48
Two-cov PLDA	2.23	1.12	3.65	1.73	13.50	4.22	0.71	5.30	2.65	36.40
Two-stage PLDA	1.03	2.35	**0.43**	**0.00**	**1.42**	1.91	2.21	1.24	**0.00**	12.10

the baseline is trained only on the natural data, which makes it tuned to the LICIT protocol. As a downside, its performance dramatically degrades on the SPOOF protocol. Fig. 7 illustrates this further.

2) Comparing the two score fusion variants, cosine scoring and two-stage PLDA, the latter is considerably better in terms of both pooled EERs and SFARs. This might look surprising as they performed comparably with the standalone countermeasures (Table IV). A possible reason is that the log-likelihood ratio scores produced by two-stage PLDA are better calibrated and hence fuse better.
3) Comparing the three integrated systems (highlighted in gray), the one-stage systems are behind the two-stage system on the SPOOF protocol (SFAR) and in terms of EER. Only the two-stage PLDA demonstrates decent generalization abilities in the most challenging conditions, when only LPC-vocoded speech is available for training (the middle part of the Table V). It decreases SFARs of the previously unseen MCEP-based FS attacks by factors of 5 and 2 for female and male trials, respectively. Similar findings are depicted in Fig. 7.
4) Fig. 7 illustrates the best that, for female trials, both the two-stage integrated system and the two-stage fusion system show almost equal performance, while the latter is a clear leader among all the systems for male trials.
5) Comparing the differences between genders, we see that it is much easier for male trials to spoof the systems. In the middle part of the Table V even the best countermeasure cannot reduce SFAR of FS attacks below 19.79%. Similar findings were recently reported in another work [43]. Based

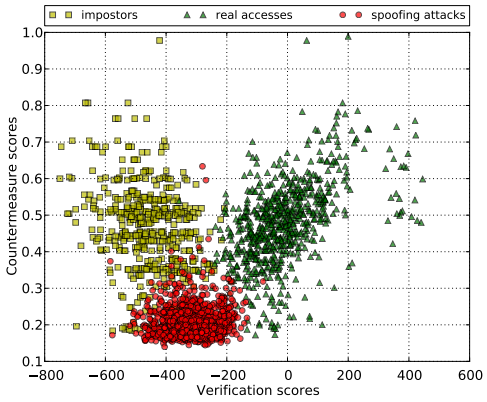


Fig. 6: Scatter plot for male trials. In this example, cosine scoring countermeasure is trained using MCEP-coded speech.

on our experience with voice conversion, we can say that usually male-to-male conversion is much easier than female-to-female conversion. At the same time, vocoders usually work better for males than females. This might explain the observed differences.

Last but not least, Fig. 8 displays the score distributions of both the initial baseline system and the best integrated system (two-stage PLDA) for female trials. The scores of the spoofing attacks are generally shifted to the left by the two-stage PLDA system, leading to increased separation between genuine and impostors trials. The red DET curves in Fig. 5 also confirm this finding: for the SPOOF protocol, the curves are shifted towards the bottom left (i.e. lower error rates). On the LICIT protocol, the curves of both baseline and integrated-system are close to each other although the baseline results are often slightly better.

VII. Conclusions

All the existing literature on voice conversion and synthetic speech detection focuses on designing discriminative features. In this study, we have shown that the problem can be tackled in the space of speaker models, too. Specifically, we suggested using *i*-vectors and PLDA back-end not only for speaker verification but for spoofing detection and joint modeling of speaker and spoof hypotheses. Besides presenting this novel framework, our evaluation protocol involved mismatched vocoder training-test conditions not considered in most earlier studies.

We separately evaluated the accuracy of speaker verification, spoofing detection and the joint systems. Concerning **standalone speaker verification**, the *i*-vector PLDA approach (EER = 0.81% for female and EER = 0.54% for male) outperformed the two other techniques on the LICIT protocol as expected. Under spoofing, however, its overall FAR increased by a factor of 28 for female subset and a factor of 47 for male subset, confirming that *i*-vector PLDA systems without countermeasures are vulnerable to voice conversion attacks. Concerning **standalone spoofing detector**, we found cosine scoring of *i*-vectors and two-stage PLDA systems to be the most stable across different conditions. Regarding the two types of attacks, LPC-coded attacks were easy to detect even

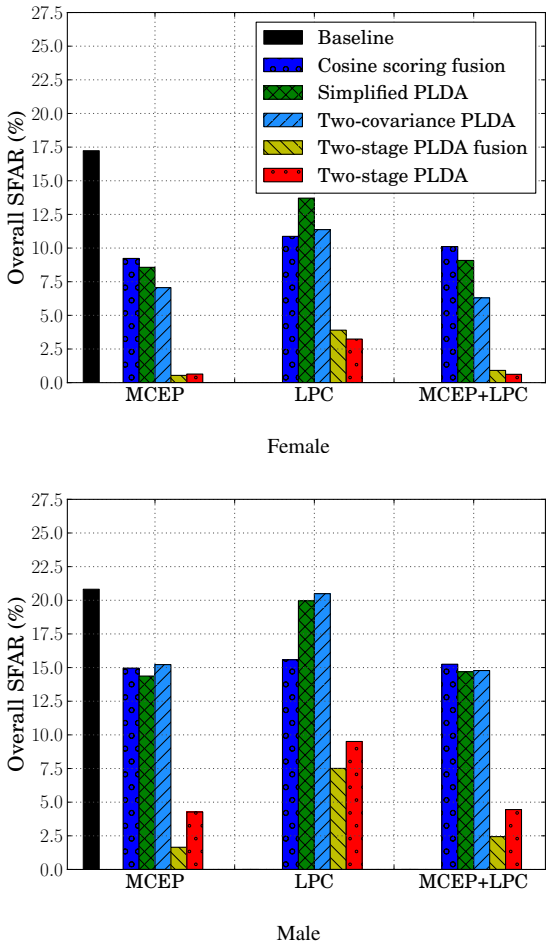


Fig. 7: Overall SFARs on both female and male trials for all joint systems and for all prepared conditions (i.e. MCEP, LPC, MCEP + LPC). The test attacks include both MCEP- and LPC-coded VC speech.

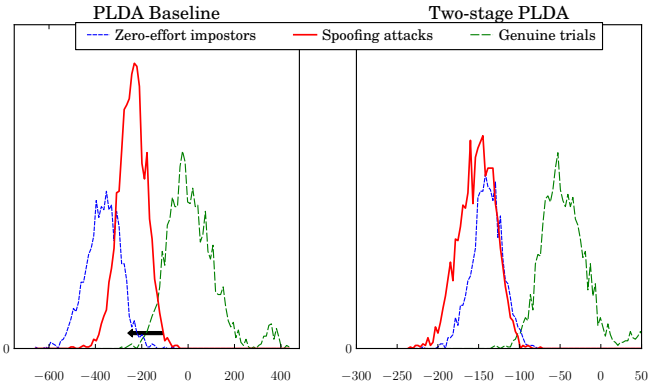


Fig. 8: Score distribution for female trials. This figure shows the score distribution for both baseline and two-stage PLDA system trained on MCEP- and LPC-coded speech.

in the mismatched case while MCEP-coded attacks were more challenging. This result is understandable in the light of *i*-vector distributions graphed above.

Concerning the experiments on **joint speaker verification and anti-spoofing**, the new joint approach of modeling ad-

ditional synthesis-channel subspace outperforms other considered methods by a large margin and shows promising abilities of generalization to the unseen cases. We expect further improvements by combining it with spoofing-specific features to replace MFCCs, for instance, with existing custom features (modified group delay, local binary pattern) or some new features derived through feature learning techniques. Whether such a complete system will generalize well to unseen attacks remains to be seen in future work.

Our study has a few limitations as well. We considered only two different voice conversion techniques: joint-density Gaussian mixture model (JD-GMM) based on MCEP and LPC features and a simplified frame selection (FS) method based on MCEP features. We used the same features to produce vocoded speech for training. These are similar techniques originating from the same software package, SPTK. Thus, further experiments involving more severely mismatched spoofing techniques is required to claim truly generalized countermeasures. Nevertheless, the promising experiments here suggest that the general framework of joint modeling of synthesis channels and natural utterance variations is worth for further exploration. In fact, it would be a possible candidate as a baseline technique for voice anti-spoofing. To help other researchers re-produce our results, we share both the *i*-vectors[6] and the program codes[7] used in this study.

ACKNOWLEDGEMENTS

The authors would like to thank the reviewers for their valuable comments that helped improving both the methodology and representation. Additional thanks go to Dr. Patrick Kenny at CRIM, Canada, for his encouragement and early feedback on our Interspeech paper.

REFERENCES

[1] A. Jain, A. Ross, and S. Pankati, "Biometrics: A tool for information security," *IEEE Trans. on Information Forensics and Security (TIFS)*, vol. 1, no. 2, pp. 125–143, June 2006.

[2] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM Systems Journal*, vol. 40, no. 3, pp. 614–634, 2001.

[3] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: from features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, January 2010.

[4] N. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification," in *Interspeech*, 2013.

[5] Y. Stylianou, "Voice transformation: a survey," in *ICASSP*, 2009.

[6] B. L. Pellom and J. H. Hansen, "An experimental study of speaker verification sensitivity to computer voice-altered imposters," in *ICASSP*, 1999.

[7] P. Perrot, G. Aversano, R. Blouet, M. Charbit, and G. Chollet, "Voice forgery using ALISP: indexation in a client memory," in *ICASSP*, 2005.

[8] D. Matrouf, J.-F. Bonastre, and C. Fredouille, "Effect of speech transformation on impostor acceptance," in *ICASSP*, 2006.

[9] J.-F. Bonastre, D. Matrouf, and C. Fredouille, "Artificial impostor voice transformation effects on false acceptance rates," in *Interspeech*, 2007.

[10] T. Kinnunen, Z.-Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech," in *ICASSP*, 2012.

[11] Z. Wu, T. Kinnunen, E. S. Chng, H. Li, and E. Ambikairajah, "A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case," in *Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2012.

[6] http://www.idiap.ch/resource/biometric/

[7] https://pypi.python.org/pypi/xspear.fast_plda

[12] Z. Kons and H. Aronowitz, "Voice transformation-based spoofing of text-dependent speaker verification systems," in *Interspeech*, 2013.

[13] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.

[14] F. Alegre, R. Vipperla, N. Evans, and B. Fauve, "On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals," in *European Signal Processing Conference (EUSIPCO)*, 2012.

[15] F. Alegre, R. Vipperla, N. Evans *et al.*, "Spoofing countermeasures for the protection of automatic speaker recognition systems against attacks with artificial signals," in *Interspeech*, 2012.

[16] P. L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, "Evaluation of speaker verification security and detection of HMM-based synthetic speech," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2280–2290, 2012.

[17] F. Alegre *et al.*, "A new speaker verification spoofing countermeasure based on local binary patterns," in *Interspeech*, 2013.

[18] F. Alegre, A. Amehraye, and N. Evans, "A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns," in *Proc. Int. Conf. on Biometrics: Theory, Applications and Systems (BTAS)*, 2013.

[19] Z. Wu *et al.*, "Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition," in *Interspeech*, 2012.

[20] F. Alegre, A. Amehraye, and N. Evans, "Spoofing countermeasures to protect automatic speaker verification from voice conversion," in *ICASSP*, 2013.

[21] Z. Wu, X. Xiao, E. S. Chng, and H. Li, "Synthetic speech detection using temporal modulation feature," in *ICASSP*, 2013.

[22] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *Proc. IEEE 11th International Conference on Computer Vision (ICCV)*, 2007.

[23] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *Proc. the International Conference of the Biometrics Special Interest Group (BIOSIG)*, 2012.

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

[25] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE ICCV*, 2007, pp. 1–8.

[26] E. Khoury, T. Kinnunen *et al.*, "Introducing i-vectors for joint anti-spoofing and speaker verification," in *Interspeech*, 2014.

[27] T. Toda, A. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[28] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, and D. Erro, "A cross-vocoder study of speaker independent synthetic speech detection using phase information," in *Interspeech*, 2014.

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

[30] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[31] O. Glembek, L. Burget, N. Dehak, N. Brümmer, and P. Kenny, "Comparison of scoring methods used in speaker recognition with joint factor analysis," in *IEEE ICASSP*, 2009, pp. 4057–4060.

[32] D. Garcia-Romero and C. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," 2011.

[33] P. Kenny, "Bayesian speaker verification with heavy-tailed priors." in *Odyssey*, 2010, p. 14.

[34] C. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., 2006.

[35] N. Brümmer and E. De Villiers, "The speaker partitioning problem," in *Odyssey Speak. and Lan. Recog. Workshop*, 2010.

[36] A. Sizov, K. A. Lee *et al.*, "Unifying probabilistic linear discriminant analysis variants in biometric authentication," in *S+SSPR*, 2014.

[37] V. N. Vapnik, *The Nature of Statistical Learning Theory*, 1995.

[38] S. Pigeon, P. Druyts, and P. Verlinde, "Applying logistic regression to the fusion of the NIST'99 1-speaker submissions," *Digital Signal Processing*, vol. 10, no. 1–3, pp. 237–248, 2000.

[39] N. Brümmer *et al.*, "Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006," *IEEE Trans. on Speech, Audio and Language Processing*, vol. 15, no. 7, pp. 2072–2084, 2007.

[40] T. P. Minka, "Algorithms for maximum-likelihood logistic regression," CMU Statistics Department, Tech. Rep. 758, 2001.

[41] P. Li, Y. Fu, U. Mohammed, J. H. Elder, and S. J. Prince, "Probabilistic models for inference about identity," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 1, pp. 144–157, 2012.
[42] E. Khoury, L. El Shafey, and S. Marcel, "Spear: An open source toolbox for speaker recognition based on Bob," in *ICASSP*, 2014.
[43] Z. Wu, A. Khodabakhsh, C. Demiroglu, J. Yamagishi, D. Saito, T. Toda, and S. King, "Sas: A speaker verification spoofing database containing diverse attacks (to appear)," in *ICASSP*, 2015.
[44] N. Brümmer, "EM for probabilistic LDA," Agnitio Research, Cape Town, Tech. Rep., 2010, sites.google.com/site/nikobrummer/.
[45] ——, "The EM algorithm and minimum divergence," Agnitio Research, Cape Town, Tech. Rep., 2009, sites.google.com/site/nikobrummer/.
[46] J. Luttinen and A. Ilin, "Transformations in variational bayesian factor analysis to speed up learning," *Neurocomputing*, vol. 73, no. 79, pp. 1093 – 1102, 2010.

APPENDIX

Here we present the EM-algorithm to train the synthesis subspace matrix $\mathbf{U}_2$ for the two-stage PLDA model. This algorithm is a modification of the EM-algorithm for the standard PLDA model [44], [36]. First, we subtract synthesis data mean vector $\boldsymbol{\mu}_2$ from the data. The E-step is the same as in the standard PLDA; during it, we compute the following matrices:

$$\mathbf{T}_{\mathbf{x}} = \sum_{ij} \mathbb{E}[\mathbf{x}_{i,j}]\boldsymbol{\phi}_{i,j}^{\mathsf{T}}\,, \tag{23}$$

$$\mathbf{R}_{\mathbf{yx}} = \sum_{ij} \mathbb{E}[\mathbf{y}_i\mathbf{x}_{i,j}]\,, \tag{24}$$

$$\mathbf{R}_{\mathbf{xx}} = \sum_{ij} \mathbb{E}[\mathbf{x}_{i,j}\mathbf{x}_{i,j}^{\mathsf{T}}]\,. \tag{25}$$

At the M-step we update the matrix $\mathbf{U}_2$ as follows:

$$\mathbf{U}_2 = (\mathbf{T}_{\mathbf{x}}^{\mathsf{T}} - \mathbf{V}_2\mathbf{R}_{\mathbf{yx}})\mathbf{R}_{\mathbf{xx}}^{-1}\,, \tag{26}$$

where matrix $\mathbf{V}_2$ is initialized with $\mathbf{V}_1$.

To speed up convergence we apply so-called *minimum-divergence* (MD) step [45], [46]. During this step we do not restrict the latent variables $\mathbf{x}_{i,j}$ to have a standard normal prior, then we maximize w.r.t. prior hyper-parameters and find equivalent representation but with a standard normal prior. This step is efficient in escaping saddle-points. For the MD-step we need a number of auxiliary matrices:

$$\mathbf{G}^{\mathsf{T}} = \mathbf{R}_{\mathbf{yy}}^{-1}\mathbf{R}_{\mathbf{yx}}\,, \tag{27}$$

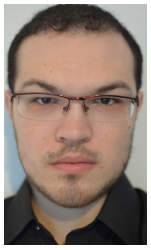
$$\boldsymbol{\mathcal{Z}} = \frac{1}{N}(\mathbf{R}_{\mathbf{xx}} - \mathbf{G}\mathbf{R}_{\mathbf{yx}})\,. \tag{28}$$

After that it is sufficient to apply the following transformations:

$$\mathbf{U}_2 \leftarrow \mathbf{U}_2\mathrm{chol}(\boldsymbol{\mathcal{Z}})\,, \tag{29}$$

$$\mathbf{V}_2 \leftarrow \mathbf{V}_2 + \mathbf{U}_2\mathbf{G}\,, \tag{30}$$

where $\mathrm{chol}(\boldsymbol{\mathcal{Z}})$ is the Cholesky decomposition of the matrix $\boldsymbol{\mathcal{Z}}$. Modification of the matrix $\mathbf{V}_2$ is due to the potential shift of the mean value $(\mathbf{U}_2\mathbf{y}_i)$ that we absorb into the matrix $\mathbf{V}_2$.



Aleksandr Sizov is a PhD student at the Speech and Image Processing Unit at the University of Eastern Finland. He received his Bachelor degree in mathematics from the Saint Petersburg State University, Russia, in 2011, and the M.E degree in computer science from the Saint Petersburg State University of Information Technologies, Mechanics and Optics, Russia, in 2013. His current research interests include speaker verification, anti-spoofing and machine learning.

Dr. Elie Khoury is a postdoctoral researcher at the Biometrics group at the Idiap Research Institute (CH). His research interests include speaker and face recognition, diarization and anti-spoofing. He received his Master degree in 2006 from the University of Toulouse III (FR). He was granted the French ministry fellowship to pursue a Ph.D. in Computer Science at the University of Toulouse (FR) and he successfully defended it in 2010. In 2011, He was a postdoctoral researcher at the University of Maine (FR). He was a visiting researcher at Columbia University (USA), Idiap Research Institute (CH), University of Eastern Finland (FI) and Google in 2008, 2011, 2014 and 2015, respectively.

Docent Tomi Kinnunen received the M.Sc., Ph.Lic. and Ph.D. degrees in computer science from the University of Joensuu (now University of Eastern Finland, UEF), Finland, in 1999, 2004 and 2005, respectively. From 2005 to 2007, he worked as an associate scientist at the Institute for Infocomm Research (I2R), Singapore. Since 2007, he has been with UEF. From 2010 to 2012, he was funded by a post-doc grant from Academy of Finland and he currently holds position of university researcher. He serves as an associate editor in *IEEE/ACM Transactions on Audio, Speech and Language Processing* and *Digital Signal Processing* and was the chair of *Odyssey 2014: the Speaker and Language Recognition Workshop*. His primary research interests include speaker recognition, feature extraction and pattern recognition.

Zhizheng Wu received the B.E. degree in computer science from Hangzhou Dianzi University, Hangzhou, China, in 2006, and the M.E. degree from Nankai University, Tianjin, China, in 2009. He was a Ph.D candidate at the School of Computer Engineering, Nanyang Technological University, Singapore from 2010 to 2014. He is now a research associate at the Centre for Speech Technology Research, University of Edinburgh, United Kingdom. His current research interests include voice conversion, speech synthesis and speaker verification.

Dr. Sébastien Marcel is a Senior Research Scientist at the Idiap where he leads the Biometrics group and conducts research on multi-modal biometrics including face recognition, speaker recognition, vein recognition, as well as spoofing and anti-spoofing. He is also the Director of the Swiss Center for Biometrics Research and Testing. In January 2010, he was appointed Visiting Professor at the University of Cagliari and since 2013 he is an external lecturer in the EPFL Electrical Engineering Doctoral (EDEE) program. Among coordination and participation in European Research projects (FP7 MOBIO, FP7 BBFor2), he was coordinating the EU FP7 ICT TABULA RASA project which aimed to develop spoofing countermeasures for a wide variety of biometrics including mainstream and novel modalities, and he currently coordinates the EU FP7 SEC BEAT project. He is the main organizer of a number of special scientific events and competitive evaluations all involving biometrics, most notably the TABULA RASA Spoofing Challenge that was held in conjunction with the 2013 International Conference on Biometrics. He serves as an Associate Editor for IEEE Transactions on Information Forensics and Security. He is also co-Editor of the upcoming Springer "Handbook on Biometric Anti-Spoofing" and Area Editor for the Encyclopedia of Biometrics (2nd Edition). Finally, he is active in reproducible research in biometrics with the open source signal-processing and machine learning toolbox Bob.