This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

Impact of score fusion on voice biometrics and presentation attack detection in cross-database evaluations

Pavel Korshunov and Sébastien Marcel

***Abstract*—Research in the area of automatic speaker verification (ASV) has advanced enough for the industry to start using ASV systems in practical applications. However, these systems are highly vulnerable to spoofing or presentation attacks, limiting their wide deployment. Therefore, it is important to develop mechanisms that can detect such attacks, and it is equally important for these mechanisms to be seamlessly integrated into existing ASV systems for practical and attack-resistant solutions. To be practical, however, an attack detection should have (i) high accuracy, (ii) be well-generalized for different attacks, and (iii) be simple and efficient. Several audio-based presentation attack detection (PAD) methods have been proposed recently but their evaluation was usually done on a single, often obscure, database with limited number of attacks. Therefore, in this paper, we conduct an extensive study of eight state of the art PAD methods and evaluate their ability to detect known and unknown attacks (e.g., in a cross-database scenario) using two major publicly available speaker databases with spoofing attacks: AVspoof and ASVspoof. We investigate whether combining several PAD systems via score fusion can improve attack detection accuracy. We also study the impact of fusing PAD systems (via parallel and cascading schemes) with two *i-vector* and inter-session variability (ISV)-based ASV systems on the overall performance in both *bona fide* (no attacks) and *spoof* scenarios. The evaluation results question the efficiency and practicality of the existing PAD systems, especially when comparing results for individual databases and cross-database data. Fusing several PAD systems can lead to a slightly improved performance, however, how to select which systems to fuse remains an open question. Joint ASV-PAD systems show a significantly increased resistance to the attacks at the expense of slightly degraded performance for bona fide scenarios.**

I. Introduction

Recent years have shown an increase in both the accuracy of biometric systems and their practical use. The application of biometrics is becoming widespread with fingerprint sensors in smartphones, automatic face recognition in social networks and video-based applications, and speaker recognition in phone banking and other phone-based services. The popularization of the biometric systems, however, exposed their major flaw — high vulnerability to spoofing attacks [1]. A fingerprint sensor can be easily tricked with a simple glue-made mold, a face recognition system can be accessed using a printed photo, and a speaker recognition system can be spoofed with a replay of pre-recorded voice. The ease with which a biometric system can be spoofed demonstrates the importance of developing efficient anti-spoofing systems that can detect both known (conceivable now) and unknown (possible in the future) spoofing attacks.

P. Korshunov and S. Marcel are in Biometrics group at Idiap Research Institute, Martigny, Switzerland; emails: {pavel.korshunov,sebastien.marcel}@idiap.ch

Manuscript received .....

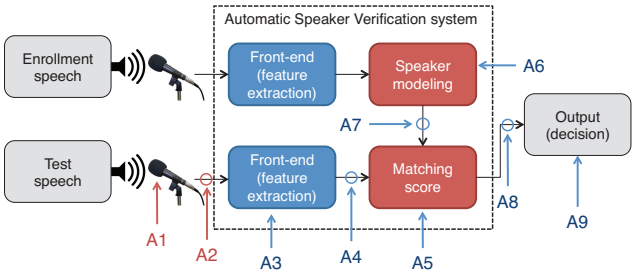


Fig. 1: Possible attack places in a typical ASV system.

In this paper, we focus on the spoofing attack detection or presentation attack detection (PAD) systems in the context of voice biometrics and their integration with automatic speaker verification (ASV) systems. Ideally, the final joint ASV-PAD system should be resistant to presentation attacks just like its PAD component and, in the same time, have the same verification accuracy as its ASV component, while generalizing well across different types of genuine and attack data.

Given the complexity of a practical ASV system, several different modules of the system are prone to attacks, as it is identified in ISO/IEC 30107-1 standard [2] and illustrated by Figure 1. Depending on the usage scenario, two of the most vulnerable places in an ASV system are marked by 'A1' (aka 'physical access' as defined in [3] or presentation attacks) and 'A2' (aka 'logical access' attacks as defined in [3]) in the figure. In this paper, we consider both logical access and presentation attacks but we focus on presentation attacks, because they are often easier to perform, e.g., *replay attacks* require no special knowledge of voice conversion or speech synthesis algorithms, and they are considered to be a serious threat by the industry, as reflected in the standard [2]. Presentation attacks assume that either a stolen set of user's samples or an automatically generated set of samples is replayed to a microphone of the ASV system under attack with an attempt to mimic the genuine registered user.

According to a comprehensive recent survey by Wu *et al.* [4], most of the available work on anti-spoofing focuses on synthetic attacks, such as voice conversion and speech synthesis. Researchers typically resort to taking databases designed for verification and identification tasks and adding synthetically generated spoofing attacks, such as voice conver-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

sion or speech synthesis, since they do not require additional lengthy recording sessions. This approach led to the lack of easily available databases with replay attacks and shaped the anti-spoofing research in the direction that focused mostly on the synthetic attacks detection.

That is why the most commonly used public database for evaluation of PAD systems, ASVspoof[1] [3], created as part of the 2015 Interspeech anti-spoofing challenge, contains only synthetically generated spoofing attacks. These attacks are assumed to be fed into a verification system directly bypassing its microphone, and are also coined as logical access attacks [3].

Recently, a database, called AVspoof[2] [5], with several replay-based attacks (logical access attacks are also provided) became publicly available. It contains a comprehensive set of presentation attacks, including, (i) the direct replay attacks when a genuine data is played back using a laptop and two phones (Samsung Galaxy S4 and iPhone 3G), (ii) synthesized speech replayed with a laptop, and (iii) an attack data, generated using a voice conversion algorithm, replayed with a laptop. The release of this database gives an opportunity to investigate the performance of current PAD methods on replay-based attacks and to evaluate how well these methods generalize across different types of data.

Therefore, taking the recent work of Sahidullah *et al.* [6], which benchmarked several anti-spoofing systems, as a starting point, we have performed a preliminary study of several PAD methods in cross-database scenario [7], as well as, fusion of PAD with ASV systems [8]. This paper extends the preliminary work by investigating the impact of several score fusion methods on PAD and joint ASV-PAD performances. The raw scores are also calibrated, so that they can be interpreted as practically useful likelihood ratios.

For the evaluations, we have selected eight well-performing methods and developed their open source implementation based on a well-known Bob framework [9][3]. Hence, we have implemented: GMM-based classifier using cepstral-based features with rectangular (RFCC), mel-scale triangular (MFCC) [10], inverted mel-scale triangular (IMFCC), and linear triangular (LFCC) filters [11], spectral flux-based features (SSFC) [12], subband centroid frequency (SCFC) [13], and subband centroid magnitude (SCMC) [13] features. We also included recently proposed constant Q cepstral coefficients (CQCCs) [14], which were shown good performance on ASVspoof database[4]. Figure 5 illustrates the processing flow of the implemented PAD systems.

We first evaluate these selected PAD systems on ASVspoof and AVspoof databases, and then, to understand how well the systems can generalize across different types of data and attacks, we conducted an extensive cross-database evaluation by training the systems on data from one database and testing them on data from another database. The aim of these evaluations is to demonstrate the importance of presentation attacks and to understand how efficient and practical the currently available PAD systems are.

In addition to evaluating individual PAD methods, we also evaluated joint PAD systems obtained by fusing several systems via score fusion approach, using mean, logistic regression, and polynomial logistic regression fusion methods. The correct performance of the fusion is ensured by using scores pre-calibrated with logistic regression.

However, having presentation attack detection methods is not enough for practical use. Such PAD systems should be seamlessly and effectively integrated with existing ASV systems. In this paper, we integrate speaker verification and presentation attack detection systems (individual and fused PAD systems) also by using score fusion, but in addition to parallel scheme (see Figure 3), we also consider cascading fusion (see Figure 2). The score fusion-based integration allows to separate *bona fide* data of the valid users, who are trying to be verified by the system, from both presentation attacks and genuine data of the non-valid users or so-called *zero-impostors*. For ASV system, we adopt verification approaches based on inter-session variability (ISV) modeling [15] and *i-vectors* [16], as the state of the art systems for speaker verification.

To allow researchers to verify, reproduce, and improve our work, we provide all implementations of PAD and ASV systems, as well as fusion, considered in this paper as an open source package available to public[5].

The main contributions of this paper are:

- An extensive evaluation of PAD systems and their fusion-based derivatives on AVspoof and ASVspoof databases, including a cross-database scenario;
- Integration of different PAD and ASV systems into one joint ASV-PAD system based on parallel and cascading score fusion techniques;
- Open source implementation of seven state of the art PAD systems, fusion tools, and evaluation framework;

II. Background and related work

The research on presentation attack detection is far from being matured, especially, if compared to the significant advances in speech analysis and speaker verification. In this section, we provide an overview of the most typical features and classifiers used in PAD systems and discuss score fusion technique of joining several systems together.

A. Features

A survey by Wu *et al.* [4] provides a comprehensive overview of the spoofing attacks and the currently available attack detection methods. These methods use features mostly based on the audio spectrogram, such as spectral- and cepstral-based features [17], phase-based features [18], the combination of amplitude and phase features [19], and audio quality based features [20]. Features directly extracted from a spectrogram can also be used, as per the recent work that relies on local maxima of spectrogram [21], which showed an impressive

[1]http://datashare.is.ed.ac.uk/handle/10283/853

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

[3]http://idiap.github.io/bob/

[4]Precomputed CQCC features were provided by the authors.

[5]https://pypi.python.org/pypi/bob.paper.jstsp_2017

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

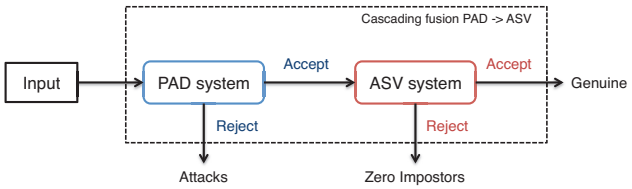


Fig. 2: A joint PAD-ASV system based on cascading score fusion (reversed order of the systems leads to the same results).

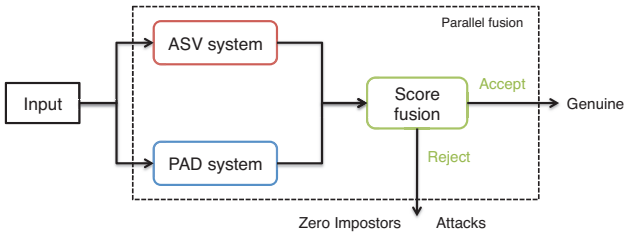


Fig. 3: A joint ASV-PAD system based on parallel score fusion.

performance, albeit for the evaluation database that was based on a set of speech recordings collected with VoIP phones, which provided little challenge for an anti-spoofing system. Constant Q cepstral coefficients (CQCCs) [14] features were proposed recently and they have shown a superior performance in detecting both know and unknown attacks in ASVspoof database. Also, a higher computational layer can be added, for instance, Alegre *et al.* [22] proposed to use histograms of Local Binary Patterns (LBP), which can be computed directly from a set of pre-selected spectral, phase-based, or other features.

B. Classifiers

Besides determining 'good features for detecting presentation attacks', it is also important to correctly classify the computed feature vectors as belonging to bona fide or spoofed data. Choosing a reliable classifier is especially important given a possibly unpredictable nature of attacks in a practical system, since it is not known in advance what kind of attack the perpetrator may use. The most common approach to classification is to use one of the well-known classifiers, which is usually pre-trained on the examples of both real and spoofed data. To simulate realistic environments, the classifier can be trained on a subset of the attacks, termed *known attacks*, and tested on the larger set of attacks that include both known and *unknown attacks*.

Different methods use different classifiers but the most common choices include logistic regression, support vector machine (SVM), and Gaussian mixture model (GMM) classifiers. The benchmarking study on logical access attacks [6] finds GMMs to be more successful compared to two-class SVM (combined with an LBP-based feature extraction from [22]) in detecting synthetic spoofing attacks. Deep learning networks are also showing promising performance in simultaneous feature selection and classification [23].

The research on automatic speaker verification is more established with regular competitions conducted by National Institute of Standards and Technology (NIST) since 1996[6]. Many techniques have been proposed with the most notable systems based on GMM, inter-session variability (ISV) modeling [15], joint factor analysis (JFA) [24], and *i-vectors* [16].

In this paper, we consider ASV systems based on ISV and *i-vectors*, as they represent the state of the art systems for speech verification.

C. Score fusion

In this paper, we focus on a score level fusion as a means to integrate different PAD systems or ASV and PAD systems into one joint system. Due to relative simplicity of such fusion and the evidence that it leads to a better performing combined systems, this operation has become popular among researchers. However, the danger is to rely on score fusion blindly without studying how it can affect different systems in different scenarios.

One way to fuse ASV and PAD or several PAD systems at the score level is to use a parallel scheme, as it is illustrated in Figure 3. In this case, the scores from each of N system are combined into a new feature vector of length N that needs to be classified. The classification task can be performed using different approaches, and, in this paper, we consider three different algorithms: (i) a logistic regression classifier, denoted as 'LR', which leads to a straight line separation, (ii) a polynomial logistic regression, denoted as 'PLR', which results in a polynomial separation line, and (iii) a simple mean function, denoted as 'Mean', which is taken on scores of the fused systems. For 'LR' and 'PLR' fusion, the classifier is pre-trained on the score-feature vectors from a training set.

Another common way, especially to combine PAD and ASV systems is a cascading scheme, in which one system is used first and only the samples that are accepted by this system (based on its own threshold) are then passed to the second system, which will further filter the samples, using its own independently determined threshold. Effectively, cascading scheme can be viewed as a *logical and* of two independent systems (see the separation of scores in cascading fusion illustrated by Figure 7a). Strictly speaking, when considering one PAD and one ASV systems, there are two variants of cascading scheme: (i) when ASV is used first, followed by PAD, and (ii) when PAD is used first, followed by ASV (see Figure 2). Although these schemes are equivalent, i.e., *and* operation is commutative, and they both lead to the same filtering results (the same error rates), we consider variant (ii), since it is defined in ISO/IEC 30107-1 standard [2].

When using a score level fusion for integrating ASV with PAD, it is important to perform a thorough evaluation of the combined/fused system to understand how incorporating PAD affects verification accuracy for both real and spoofed data. In this paper, we adopt an evaluation methodology specifically designed for performance assessment of fusion system proposed in [25].

[6]http://www.nist.gov/itl/iad/mig/sre.cfm

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

Fig. 4: AVspoof database recording setup.

III. Spoofing databases

Appropriate databases are necessary for testing different presentation attack detection approaches. These databases need to contain a set of practically feasible presentation attacks and also data for speaker verification task, so that a verification system can be tested for both issues: the accuracy of speaker verification and the resistance to the attacks.

Currently, two comprehensive publicly available databases exist that can be used for vulnerability analysis of ASV systems, the evaluation of PAD methods, and evaluation of joint ASV-PAD systems: ASVspoof and AVspoof. Both databases contain logical access attacks (LAs), while AVspoof also contains presentation attacks (PAs). For the ease of comparison with ASVspoof, the set of attacks in AVspoof is split into LA and PA subsets (see Table I).

TABLE I: Number of utterances in different subsets of AVspoof and ASVspoof databases.

Database	Type of data	Train	Dev	Eval
AVspoof	enroll data	780	780	868
	impostors	54509	54925	70620
	real data	4973	4995	5576
	LA attacks	17890	17890	20060
	PA attacks	38580	38580	43320
ASVspoof	enroll data	-	175	230
	impostors	-	9975	18400
	real data	3750	3497	9404
	known attacks	12625	49875	92000
	unknown attacks	-	-	92000

A. ASVspoof database

ASVspoof[1] database was created for a 2015 Interspeech anti-spoofing challenge [3]. It contains genuine speech data from 106 speakers (45 male and 61 female), while spoofed speech was generated using speech synthesis and voice conversion algorithms. In total, database has 10 spoofing attacks, five of which are considered 'unknown', since they appear in the evaluation set only and hence PAD systems are not trained on them.

B. AVspoof database

To our knowledge, the largest publicly available database containing speech presentation attacks is AVspoof[2] [5]. It contains of genuine speech samples from 44 participants (31 males and 13 females) recorded over the period of two months in four sessions, each scheduled several days apart in different setups and environmental conditions. The recording devices, including microphone AT2020USB+, Samsung Galaxy S4 phone, and iPhone 3GS, and the environments are shown in Figure 4.

From the recorded genuine data, two major types of attacks were created for AVspoof database: 'logical access' attacks (LA for short), similar to those in ASVspoof database [3] but generated using (i) a statistical parametric-based speech synthesis algorithm [26] and (ii) a voice conversion algorithm from Festvox[7], and the presentation attacks (PA for short).

When generating presentation attacks, the assumption is that a verification system is installed on a laptop (with an internal built-in microphone) and an attacker is trying to gain access to this system by playing back to it a pre-recorded genuine data or an automatically generated synthetic data using some playback device. In AVspoof database, presentation attacks consist of (i) direct replay attacks when a genuine data is played back using a laptop with internal speakers, a laptop with external high quality speakers, Samsung Galaxy S4 phone, and iPhone 3G, (ii) synthesized speech replayed with a laptop, and (iii) converted voice attacks replayed with a laptop.

C. Evaluation protocol

In a single database evaluation, the training (*Train*) subset of a given database is used for training a PAD or an ASV system. The development (*Dev*) set is used for determining hyper-parameters of the system and evaluation (*Eval*) set is used to test the system. In a cross-database evaluation, the training and development sets are taken from one database, while evaluation set is taken from another database. For PAD systems, a cross-attack evaluation is also possible, when the training and development sets contain one type of attack, e.g., logical access attacks only, while evaluation set contains another type, e.g., presentation or replay attacks only.

For evaluation of PAD systems, the following metrics are recommended [27]: attack presentation classification error rate

[7] http://festvox.org/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

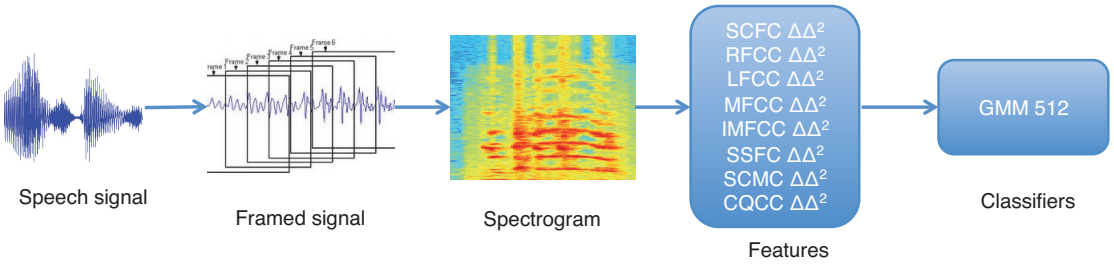


Fig. 5: Processing flow of considered individual PAD systems (as per systems in [6]).

(APCER) and bona fide presentation classification error rate (BPCER). APCER is the number of attacks misclassified as bona fide samples divided by the total number of attacks, and is defined as follows:

$$\text{APCER} = \frac{1}{N}\sum_{i=1}^{N}(1 - Res_i), \tag{1}$$

where N represents the number of attack presentations. Res_i takes value 1 if the i-th presentation is classified as an attack presentation, and value 0 if classified as a bona fide presentation. Thus, APCER can we considered as the equivalent to FAR for PAD systems, as it reflects the observed ratio of falsely accepted attack samples in relation to the total number of presented attacks.

By definition, BPCER is the number of incorrectly classified bona fide samples divided by the total number of bona fide samples:

$$\text{BPCER} = \frac{\sum_{i=1}^{N_{BF}} Res_i}{N_{BF}}, \tag{2}$$

where N_{BF} represents the number of bona fide presentations, and Res_i is defined similar to APCER. Thus, BPCER can we considered as the equivalent to FRR for PAD systems, as it reflects the observed ratio of falsely rejected genuine samples in relation to the total number of bona fide (genuine) samples. We compute equal error rate (EER) as the rate when APCER and BPCER are equal.

ASV and joint ASV-PAD systems are evaluated under two operational scenarios: *bona fide* scenario with no attacks and the goal to separate genuine samples from zero-effort impostors and *spoof* scenario with the goal to separate genuine samples from attacks. For bona fide scenario, we report false match rate (FMR), which is similar to FAR, and false non-match rate (FNMR), which is similar to FRR, while for spoof scenario, we report impostor attack presentation match rate (IAPMR), which is the proportion of attacks that incorrectly accepted as genuine samples by the joint ASV-PAD system (for details, see recommendations in ISO/IEC 30107-3 [27]).

When analyzing, comparing, and especially fusing PAD and ASV systems, it is important that the scores are calibrated in a form of likelihood ratio. Raw scores can be mapped to log-likelihood ratio scores with logistic regression classifier and an associated cost of calibration C_{llr} can be used as an application-independent performance measure of calibrated PAD or ASV system. For more details on the score calibration, please refer to [28].

Therefore, in this paper, we report EER rates (on *Eval* set) when testing the considered PAD systems on each database, for the sake of consistency with the previous literature, notably [6], and BPCER and APCER of PAD systems (using the EER threshold computed on *Dev* set) when testing PADs in cross-database scenario. EER has been commonly used within the speech community to measure the performance of ASV and PAD systems, while BPCER and APCER are the newly standardized metrics and we advocate for the use of the open evaluation standards in the literature. We also report calibration cost C_{llr} and the discrimination loss C_{llr}^{min} metrics for the individual PAD systems. FMR, FNMR, and IAPMR are reported for ASV and joint ASV-PAD systems on evaluation set (using EER threshold computed on the development set).

IV. PAD SYSTEMS

The processing flow of considered presentation attack detection systems is illustrated in Figure 5. Data from training, development, and evaluation sets is first processed to extract corresponding features. The systems mainly differ by the features used for the classification, while the classification is done using Gaussian mixture model (GMM)-based classifier (as the best in [6]), which is trained separately for bona fide and spoof data of the training set. The trained models are then used to compute scores for the features from development and evaluation sets as the difference between likelihoods to the two GMM models [8]. Each GMM model is trained using 10 expectation-maximization (EM) iterations and has 512 Gaussians components.

A. Individual PAD systems

We have selected several state of the art methods for attacks detection in speech (please see Figure 5 for the schematic overview), which were recently evaluated by Sahidullah *et al.* [6] on ASVspoof database with an addition of CQCC features based method [14].

We selected four cepstral-based features with rectangular (RFCC), mel-scale triangular (MFCC) [10], inverted mel-scale triangular (IMFCC), and linear triangular (LFCC) filters [11]. These features are computed from a power spectrum (power of magnitude of 512-sized FFT) by applying one of the above filters of a given size (we use size 20 as per [6]). We also implemented spectral flux-based features (SSFC) [12], which are Euclidean distances between power spectrums (normalized by the maximum value) of two consecutive frames, subband

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

TABLE II: Performance of PAD systems and their fused derivatives in terms of average EER (%), C_{llr}, and C_{llr}^{min} of calibrated scores for evaluation sets of ASVspoof [3] and AVspoof [5] databases.

PAD systems	Fusion	ASVspoof (Eval)							AVspoof (Eval)					
		Known			S10	Unknown			LA			PA		
		EER	C_{llr}	C_{llr}^{min}	EER	EER	C_{llr}	C_{llr}^{min}	EER	C_{llr}	C_{llr}^{min}	EER	C_{llr}	C_{llr}^{min}
SCFC	-	0.11	0.732	0.006	23.92	5.17	0.951	0.625	0.00	0.730	0.000	5.34	0.761	0.160
RFCC	-	0.14	0.731	0.009	6.34	1.32	0.825	0.230	0.04	0.729	0.001	3.27	0.785	0.117
LFCC	-	0.13	0.730	0.005	5.56	1.20	0.818	0.211	0.00	0.728	0.000	4.73	0.811	0.153
MFCC	-	**0.47**	**0.737**	**0.023**	**14.03**	**2.93**	**0.877**	**0.435**	**0.00**	**0.727**	**0.000**	**5.43**	**0.812**	**0.165**
IMFCC	-	0.20	0.730	0.007	5.11	1.57	0.804	0.192	0.00	0.728	0.000	4.09	0.797	0.137
SSFC	-	0.27	0.733	0.016	7.15	1.60	0.819	0.251	0.70	0.734	0.027	4.70	0.800	0.160
SCMC	-	0.19	0.731	0.009	6.32	1.37	0.812	0.229	0.01	0.728	0.000	3.95	0.805	0.141
CQCC	-	0.10	0.732	0.008	1.59	0.58	0.756	0.061	0.66	0.733	0.028	3.84	0.796	0.128
8-fused-PADs	**Mean**	**0.04**	**0.732**	**0.003**	**1.74**	**0.37**	**0.828**	**0.077**	**0.00**	**0.729**	**0.000**	**3.10**	**0.793**	**0.111**
LFCC-MFCC-SCFC	Mean	0.07	0.733	0.004	7.24	1.48	0.877	0.256	0.00	0.728	0.000	4.82	0.791	0.150
CQCC-MFCC-SCFC	Mean	0.03	0.734	0.003	2.14	0.46	0.854	0.088	0.00	0.730	0.000	3.96	0.786	0.129
CQCC-MFCC	Mean	0.08	0.734	0.006	2.18	0.47	0.811	0.085	0.02	0.730	0.001	4.14	0.802	0.132
LFCC-MFCC	Mean	0.13	0.733	0.005	7.08	1.46	0.845	0.249	0.00	0.728	0.000	5.08	0.811	0.153
IMFCC-MFCC	Mean	0.15	0.734	0.006	6.26	1.29	0.838	0.219	0.00	0.728	0.000	4.09	0.803	0.133
SCFC-SCMC	Mean	0.08	0.732	0.004	7.00	1.47	0.876	0.249	0.00	0.729	0.000	3.84	0.780	0.144
SCFC-CQCC	Mean	0.03	0.732	0.002	1.82	0.50	0.844	0.071	0.05	0.732	0.002	3.72	0.775	0.129
8-fused-PADs	LR	0.06	0.011	0.004	1.43	0.43	0.093	0.064	0.00	0.002	0.000	3.57	0.563	0.123
LFCC-MFCC-SCFC	**LR**	**0.07**	**0.014**	**0.003**	**6.43**	**1.34**	**0.517**	**0.234**	**0.00**	**0.000**	**0.000**	**4.43**	**0.513**	**0.151**
CQCC-MFCC-SCFC	LR	0.05	0.010	0.004	1.31	0.38	0.082	0.054	0.00	0.000	0.000	3.62	0.493	0.125
CQCC-MFCC	LR	0.07	0.013	0.006	1.38	0.37	0.081	0.058	0.00	0.001	0.000	3.73	0.184	0.122
LFCC-MFCC	LR	0.12	0.017	0.004	6.32	1.30	0.403	0.228	0.00	0.000	0.000	5.01	0.241	0.152
IMFCC-MFCC	LR	0.13	0.018	0.006	5.63	1.19	0.327	0.200	0.00	0.001	0.000	3.84	0.326	0.130
SCFC-SCMC	LR	0.07	0.016	0.004	7.34	1.54	0.506	0.257	0.00	0.001	0.000	3.82	0.482	0.149
SCFC-CQCC	LR	0.05	0.010	0.003	1.46	0.47	0.086	0.057	0.02	0.005	0.000	3.70	0.493	0.129
8-fused-PADs	PLR	50.10	5.534	0.096	61.71	53.55	9.811	0.732	0.00	0.000	0.000	3.39	1.175	0.115
LFCC-MFCC-SCFC	PLR	0.14	0.030	0.011	6.44	1.43	0.372	0.240	0.00	0.000	0.000	4.41	0.993	0.148
CQCC-MFCC-SCFC	PLR	50.09	4.879	0.056	57.62	52.25	6.338	0.592	0.00	0.000	0.000	3.70	0.974	0.124
CQCC-MFCC	PLR	50.06	4.699	0.048	51.47	50.96	4.965	0.249	0.00	0.001	0.000	22.20	3.732	0.675
LFCC-MFCC	PLR	0.16	0.036	0.010	5.43	1.17	0.286	0.207	0.00	0.005	0.000	4.43	0.191	0.147
IMFCC-MFCC	PLR	0.19	0.035	0.012	5.03	1.13	0.256	0.190	0.00	0.002	0.000	3.83	0.238	0.124
SCFC-SCMC	PLR	0.13	0.035	0.012	7.14	1.54	0.380	0.255	0.00	0.000	0.000	3.84	0.817	0.144
SCFC-CQCC	PLR	50.08	4.671	0.049	52.23	51.20	5.037	0.342	0.00	0.005	0.000	3.75	0.855	0.128

centroid frequency (SCFC) [13], and subband centroid magnitude (SCMC) [13] features. A discrete cosine transform (DCT-II) is applied to all above features, except for SCFC, and first 20 coefficients are taken.

Before computing selected features, a given audio sample is first split into overlapping 20ms-long speech frames with 10ms overlap. The frames are pre-emphasized with 0.97 coefficient and pre-processed by applying Hamming window. Then, for all features, we compute deltas and double-deltas [29] and keep only these features (40 in total) for the classifier. We kept only deltas and delta-deltas, because [6] reported that the static features degraded performance of PAD systems.

In addition to the above features, we also consider recently proposed CQCC [14], which are computed using constant Q transform instead of FFT. To be consistent with the other features and fair in the systems comparison, we used also only delta and delta-deltas (40 features in total) derived from 19 plus C_0 coefficients.

B. Fused PAD systems

In addition to eight individual PAD systems, we also used parallel score fusion with logistic regression (LR), polynomial logistic regression (PLR), and mean-based score classification (Mean), algorithms to produce different combinations of joint PAD systems. We limited all possible combinations of systems to include first 8 PADs fused into one system, as well as, various combinations with MFCC-based and CQCC-based PADs as the most popular and the most recent features for PAD, respectively. To avoid prior to the evaluations, the raw scores from each individual PAD system are pre-calibrated with logistic regression based on Platts sigmoid method [30] by modeling scores of the training set and applying it on the scores from development and evaluation sets.

V. Evaluation of PAD systems

The selected PAD systems (see Section IV) and their fused derivatives are evaluated on each ASVspoof and AVspoof database and in cross-database scenario. To keep results comparable with previous work [6], [31], we computed average EER (*Eval* set) for single database evaluations and APCER with BPCER for cross-database evaluations. APCER with BPCER are computed for *Eval* set of a given dataset using the EER threshold obtained from the *Dev* set from another dataset (see Table III). The calibration cost C_{llr} and the discrimination loss C_{llr}^{min} of the resulted calibrated scores are provided.

A. ASVspoof vs. AVspoof

The results of evaluating the selected methods on different types of attacks from ASVspoof and AVspoof databases are presented in Table II. These results are obtained by training each PAD system on *Train* set of a given database, then tuned on *Dev* set and tested on *Eval* set of the same database.

In Table II, the results for *known* and *unknown* attacks of *Eval* set of ASVspoof are presented separately to demonstrate the differences between these two types of attacks provided in ASVspoof database. The main contribution to the higher EER

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

TABLE III: Performance of PAD systems and their fused derivatives in terms of average APCER (%), BPCER (%), and C_{llr} of calibrated scores in cross-database testing on ASVspoof [3] and AVspoof [5] databases.

PAD system	Fusion	ASVspoof (Train/Dev)						AVspoof-LA (Train/Dev)					
		AVspoof-LA (Eval)			AVspoof-PA (Eval)			AVspoof-PA (Eval)			ASVspoof (Eval)		
		APCER	BPCER	C_{llr}	APCER	BPCER	C_{llr}	APCER	BPCER	C_{llr}	APCER	BPCER	C_{llr}
SCFC	-	0.10	2.76	0.751	10.20	2.76	0.809	15.12	0.00	0.887	39.62	0.35	0.970
RFCC	-	0.29	69.57	0.887	7.51	69.57	0.927	26.39	0.00	0.902	48.32	2.86	0.988
LFCC	-	1.30	0.13	0.740	21.03	0.13	0.868	17.70	0.00	0.930	37.49	0.02	0.958
MFCC	-	**1.20**	**2.55**	**0.764**	**17.09**	**2.55**	**0.838**	**10.60**	**0.00**	**0.819**	**19.72**	**1.22**	**0.870**
IMFCC	-	4.57	0.00	0.761	92.98	0.00	1.122	99.14	0.00	1.164	43.00	0.60	0.966
SSFC	-	4.81	64.47	0.899	18.89	64.47	0.973	71.84	0.68	1.047	63.45	23.54	1.070
SCMC	-	0.75	1.70	0.750	22.61	1.70	0.866	15.94	0.00	0.861	45.97	0.01	0.978
CQCC	-	13.99	57.05	0.968	66.29	57.05	1.191	44.65	0.61	1.009	0.86	100.00	1.009
8-fused-PADs	**Mean**	**0.41**	**12.73**	**0.804**	**12.46**	**12.73**	**0.930**	**19.71**	**0.00**	**0.944**	**26.97**	**5.25**	**0.959**
LFCC-MFCC-SCFC	Mean	0.21	0.39	0.750	11.08	0.39	0.835	11.48	0.00	0.876	23.23	0.21	0.929
CQCC-MFCC-SCFC	Mean	0.20	36.78	0.817	11.54	36.78	0.929	18.38	0.00	0.900	3.67	35.77	0.924
CQCC-MFCC	Mean	0.93	49.71	0.855	21.81	49.71	0.997	20.77	0.00	0.908	1.22	99.74	0.914
LFCC-MFCC	Mean	0.88	0.52	0.751	16.78	0.52	0.851	10.92	0.00	0.872	21.33	0.55	0.911
IMFCC-MFCC	Mean	1.36	0.34	0.761	25.91	0.34	0.967	13.29	0.00	0.978	21.82	0.81	0.914
SCFC-SCMC	Mean	0.13	0.82	0.750	11.51	0.82	0.835	17.59	0.00	0.873	41.39	0.03	0.971
SCFC-CQCC	Mean	0.17	49.86	0.848	12.94	49.86	0.980	28.70	0.02	0.945	1.46	99.91	0.960
8-fused-PADs	LR	11.68	56.08	9.207	60.57	56.08	9.532	20.34	0.00	1.694	15.09	36.36	1.487
LFCC-MFCC-SCFC	**LR**	**0.17**	**0.27**	**0.041**	**11.53**	**0.27**	**0.292**	**10.79**	**0.00**	**0.655**	**20.90**	**0.45**	**1.179**
CQCC-MFCC-SCFC	LR	9.41	55.76	8.272	57.63	55.76	8.579	14.10	0.00	0.878	14.90	1.98	0.596
CQCC-MFCC	LR	11.09	56.33	9.086	61.11	56.33	9.626	13.67	0.00	0.824	1.51	94.73	0.427
LFCC-MFCC	LR	0.87	0.38	0.046	16.18	0.38	0.374	10.82	0.00	0.618	20.93	0.64	1.129
IMFCC-MFCC	LR	1.55	0.04	0.049	35.24	0.04	0.553	12.92	0.00	0.650	21.48	0.81	0.973
SCFC-SCMC	LR	0.11	0.77	0.098	11.38	0.77	0.324	17.34	0.00	0.826	41.21	0.03	1.933
SCFC-CQCC	LR	9.51	55.49	8.494	57.70	55.49	8.760	26.14	0.00	1.636	2.82	99.45	0.761
8-fused-PADs	PLR	38.74	0.38	5.159	93.11	0.38	8.552	12.23	0.04	1.130	8.95	29.64	1.345
LFCC-MFCC-SCFC	PLR	0.55	0.20	0.162	13.10	0.20	0.368	10.54	0.00	0.743	19.96	0.65	1.226
CQCC-MFCC-SCFC	PLR	31.04	1.95	3.662	88.15	1.95	6.532	11.79	0.00	0.856	14.39	2.30	0.568
CQCC-MFCC	PLR	28.98	45.62	8.961	85.07	45.62	10.210	11.91	0.00	0.764	11.50	8.57	0.397
LFCC-MFCC	PLR	1.01	0.16	0.062	17.97	0.16	0.338	10.70	0.00	0.691	20.47	0.71	1.173
IMFCC-MFCC	PLR	2.70	0.00	0.062	74.61	0.00	0.442	12.17	0.00	0.669	20.67	0.91	0.944
SCFC-SCMC	PLR	0.07	1.69	0.499	10.36	1.69	0.660	15.86	0.00	0.918	39.87	0.04	2.249
SCFC-CQCC	PLR	28.97	38.77	6.311	85.53	38.77	9.109	21.92	0.00	1.745	23.20	7.19	1.166

of unknown is given by a more challenging attack 'S10' of the *Eval* set (see column 'S10' of Table II for the results for this attack).

Since AVspoof contains both logical access (LA for short) and presentation attacks (PA), the results for these two types of attacks are also presented separately. Hence, it allows to compare the performance on ASVspoof database (it has logical access attacks only) with AVspoof-LA attacks.

From the results in Table II, we can note that (i) LA set of AVspoof is less challenging compared to ASVspoof for almost all methods, (ii) unknown attacks and, especially, 'S10' attack, for which PADs are not trained, are more challenging, and (iii) presentation attacks are also more challenging compared to LA attacks.

It can be also noted that PAD systems fused using a simple Mean fusion are *on par* or sometimes performing even better than systems fused with LR (though, LR generally leads to lower C_{llr} compared to Mean). A probable reason for this is the performed pre-calibration of the scores using logistic regression. Calibration insures that the scores are well distributed within $[0, 1]$ range, leading to similar EER-based thresholds among individual PAD systems. Hence, Mean, which can be considered as a special case of LR, leads to 'good enough' fusion results.

B. Cross-database evaluation

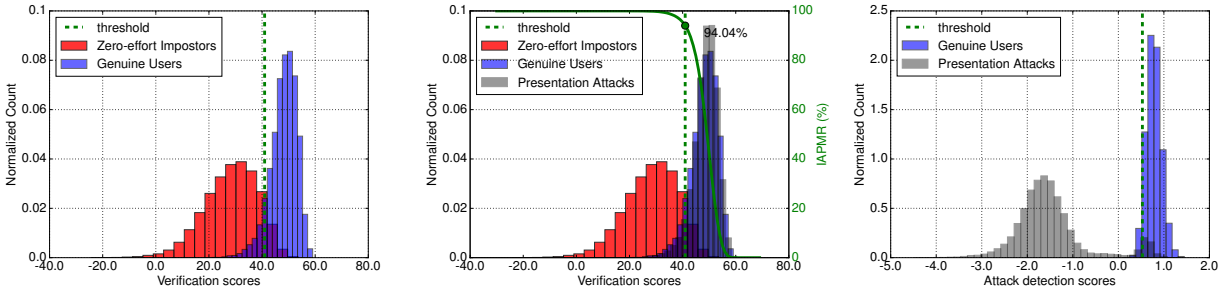
Table III presents the cross-database results when a given PAD system is trained and tuned using *Train* and *Dev* sets from one database but is tested using *Eval* set from another database. For instance, results in the second column of the table are obtained by using training and development sets from ASVspoof database but evaluation set from AVspoof-LA. Also, we evaluated the effect of using one type of attacks (e.g., logical access from AVspoof-LA) for training and another type (e.g., presentation attacks of AVspoof-PA) for testing (the results are in the last column of the table).

From Table III, we can note that all methods generalize poorly across different datasets with BPCER reaching 100%, for example, especially, CQCC-based PAD showing poor performance for all cross-database evaluations. It is also interesting to note that even similar methods, for instance, RFCC and LFCC-based, have very different accuracy in cross-database testing, even though they showed less drastic difference in single-database evaluations (see Table II).

VI. ASV SYSTEMS

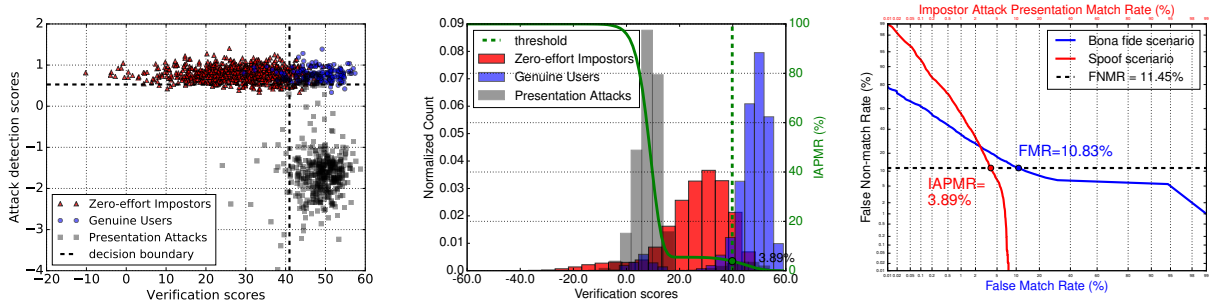
We consider two ASV systems based on inter-session variability (ISV) modeling [15] and *i-vectors* [16], which are the state of the art speaker verification systems able to effectively deal with intra-class and inter-class variability. In these systems, voice activity detection is based on the modulation of the energy around 4Hz, the features include 20 MFCCs and energy, with their first and second derivatives, and modeling was performed with 256 Gaussian components using 25 EM iterations. In *i-vectors* based system, the dimension of *i-vectors* is 100. Universal background model (UBM) was

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing



(a) ASV system, bona fide scenario (b) ASV system, spoof scenario (c) PAD system

Fig. 6: Histogram distributions of uncalibrated scores of *i-vector* ASV and *MFCC*-based PAD systems.



(a) Scatter plot, Eval set (b) Histogram distribution, Eval set (c) DET curves, Eval set

Fig. 7: A scatter plot and histogram distributions of uncalibrated scores and DET curves of calibrated scores of joint *i-vector* ASV with *MFCC*-based PAD system via cascading fusion (see the row in *italic* for *i-vector* system in Table IV).

trained using training set of MOBIO database[8], while the clients were enrolled using corresponding genuine enrollment data of development and evaluation sets of AVspoof database.

Since our main focus is on presentation attacks, we evaluate ASV and ASV-PAD joint systems on presentation attacks of AVspoof database, i.e., considering AVspoof-PA subset only.

A. Vulnerability of ASV to presentation attacks

Table IV demonstrates how *i-vectors* ('no fusion' row) and ISV-based ('no fusion' row) ASV systems perform in two different scenarios: (i) when there are no attacks present (zero-impostors only), referred as *bona fide scenario* (defined by ISO/IEC [27]), and (ii) when the system is being spoofed with presentation attacks, referred as *spoof scenario*. Histograms of score distribution in Figure 6b also illustrate the effect of attacks on *i-vectors* based ASV system in spoof scenario, compared to bona fide scenario in Figure 6a.

From Table IV, it can be noted that both ASV systems perform relatively well under bona fide scenario with ISV-based system showing lower FMR of 4.46%. However, when a spoofed data is introduced, without a PAD system in place, the IAPMR significantly increases reaching 92.41% for ISV-based and 94.04% for *i-vectors* based systems. It means that a typical verification system is not able to correctly distinguish presentation attacks from genuine data.

[8]https://www.idiap.ch/dataset/mobio

B. Integration of PAD with ASV

As described in Section IV, multiple presentation attack detection systems have been considered to detect whether a given speech sample is genuine or spoofed. However, the purpose of a PAD system is to work in tandem with a verification system, so that the joint system can effectively separate the genuine data from both zero-effort impostors (genuine data but incorrect identity) and presentation attacks (spoofed data for the correct identity).

In this section, we evaluate cascading and parallel score fusion of integrating ASV with different PAD systems. It is important to understand the main differences between these two schemes (see Figure 2 and Figure 3 for illustration). In cascading approach, each system, PAD and ASV, are trained and tuned independently on their respective training and development sets, including determining relevant thresholds on the development set. These system parameters (e.g., thresholds) are used to filter samples from the evaluation set by first applying PAD threshold to reject mostly spoofed data (hence, accepting most of the genuine and zero-effort impostor data), and then applying ASV threshold to reject mostly zero-effort impostors (see scatter plot in Figure 7a for illustration). In the end, only the desired genuine samples are accepted. It means that once these parameters are chosen for each system, they cannot be later changed for the joint ASV-PAD system, even if the requirements to FAR or FRR may change. By contrast, parallel scheme considers both ASV and PAD systems jointly

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSTSP.2017.2692389, IEEE Journal of Selected Topics in Signal Processing

TABLE IV: Fusing *i-vector* and ISV-based verification systems with the selected PAD systems (highlighted in bold in Table III) on evaluation set of AVspoof database.

ASV system	Fused with PAD	Type of fusion	Zero-impostors only		PAs only
			FMR (%)	FNMR (%)	IAPMR (%)
ISV-based	no fusion	-	4.46	9.90	92.41
	8-fused-PAD (Mean)	**Cascade**	**4.99**	**9.62**	**1.79**
	8-fused-PAD (Mean)	Mean	7.31	13.40	1.74
	8-fused-PAD (Mean)	LR	8.15	13.70	1.55
	8-fused-PAD (Mean)	PLR	4.96	9.09	2.40
	LFCC-MFCC-SCFC (LR)	Cascade	5.80	10.73	3.24
	LFCC-MFCC-SCFC (LR)	Mean	4.46	9.94	5.40
	LFCC-MFCC-SCFC (LR)	LR	4.49	10.00	5.11
	LFCC-MFCC-SCFC (LR)	PLR	4.77	9.98	98.04
	MFCC	Cascade	6.57	12.00	4.19
	MFCC	Mean	23.05	22.73	28.98
	MFCC	LR	25.40	24.72	2.68
	MFCC	PLR	4.97	10.75	5.17
i-vectors based	no fusion	-	8.85	8.31	94.04
	8-fused-PAD (Mean)	**Cascade**	**9.45**	**8.05**	**1.52**
	8-fused-PAD (Mean)	Mean	12.15	11.77	1.46
	8-fused-PAD (Mean)	LR	8.72	8.20	95.03
	8-fused-PAD (Mean)	PLR	17.47	23.30	98.71
	MFCC	*Cascade*	*10.83*	*11.45*	*3.89*
	LFCC-MFCC-SCFC (LR)	Cascade	10.14	9.77	2.84
	LFCC-MFCC-SCFC (LR)	Mean	8.84	8.37	4.92
	LFCC-MFCC-SCFC (LR)	LR	8.78	8.45	97.50
	LFCC-MFCC-SCFC (LR)	PLR	8.95	8.41	97.62
	MFCC	Mean	26.33	19.44	19.47
	MFCC	LR	8.77	8.33	94.28
	MFCC	PLR	9.60	10.47	95.76

and allows to tune all parameters of the system at the time of the fusion, including the fusion algorithm that separates genuine subset from the rest of the data in the joint scores and the threshold selected on the joint scores. It means that in parallel scheme, depending on the requirements to the final system, FMR and FMNR values can be fine-tuned for each genuine, zero-effort impostor, or attack data.

Based on the results in individual and cross-database evaluations, we have selected 3 PAD systems that performed the most well consistently across all databases and attacks: *8-fused PADs* fused via Mean score-fusion, LFCC-MFCC-SCFC fused via LR, and a simple MFCC-based PAD. These systems are highlighted in bold in Table II and Table III.

As results presented in Table IV demonstrate, integration with PAD system can effectively reduce IAPMR from above 90% of the ASV (both ISV-based and *i-vector*) to IAPMR down to 1.52%, which is the best performing joint system of *i-vector* ASV fused with *8-fused PAD* via cascade fusion. Such drastic improvement in the attack detection comes with an increase in FMR (from 4.46% to 4.99% when ASV is ISV and from 8.85% to 9.45% when ASV is *i-vector*).

From the Table IV, it is clear that *8-fused PAD* fused with both ASV systems via cascading scheme leads to more superior overall performance. However, a simple MFCC-based PAD system also performs reasonably well when fused with ASV via cascade fusion (see Figure 7c for DET plots in different scenarios), although. An important practical advantage of using MFCC-based PAD is that MFCC are the most commonly used fast to compute features in speech processing. We highlighted the results for MFCC-based PAD in Table IV in *italic* and illustrated its performance with scatter plot in Figure 7a, histogram distributions in Figure 7b, and DET curves in Figure 7c.

The Table IV shows that cascading fusion mostly leads to a better overall performance compared to parallel scheme. However, compared to a cascading scheme, where each fused system is independent and has to be tuned separately for disjoint set of parameter requirements, parallel scheme is more flexible, because it allows to tune several parameters of the fusion, as if it was one single system consisting of interdependent components. Such flexibility can be valuable in practical systems. See [25] for a detailed comparison of the different fusion schemes and their discussion.

VII. Conclusion

In this paper, we conducted a cross-database evaluation of several state of the art speech presentation attack detection methods and their different score fusion-based derivatives implemented as open source. We used two recent comprehensive databases with speech spoofing attacks: ASVspoof ('logical access' attacks only) and AVspoof ('logical access' and presentation attacks). The results demonstrated that the evaluated PAD systems generalize poorly across different databases and data.

We also considered score-based integration of several PAD and ASV systems following both cascading and parallel schemes. Evaluation results show a significantly increased resistance of joined ASV-PAD systems to presentation attacks from AVspoof database. Cascading fusion leads to a better overall performance compared to parallel scheme.

In the future, we will focus on the development of novel presentation attacks, especially targeting mobile environment, since so far the attack devices that run an ASV system are assumed to be laptops. We will also explore multimodal systems, when both ASV and PAD systems of different modalities, e.g., speech and video, are integrated to improve the performance in both bona fide and spoof scenarios. And we will investigate whether deep learning approaches for presentation attack

detection can lead to higher detection accuracies and whether they can generalize better in cross-database scenario compared to more traditional approaches.

ACKNOWLEDGMENTS

This work was partially funded by Google Inc. and Norwegian SWAN project. The authors would also like to thank Md Sahidullah for providing implementation details of the features from [6] paper and to Héctor Delgado Flores for providing precomputed CQCC features from [14].

REFERENCES

[1] S. Marcel, M. S. Nixon, and S. Z. Li, *Handbook of Biometric Anti-Spoofing: Trusted Biometrics Under Spoofing Attacks*. Springer-Verlag London, 2014.

[2] ISO/IEC JTC 1/SC 37 Biometrics, "DIS 30107-1, information technology – biometrics presentation attack detection," American National Standards Institute, Jan. 2016.

[3] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2037–2041.

[4] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *Speech Communication*, vol. 66, pp. 130–153, Feb. 2015.

[5] S. Kucur Ergunay, E. Khoury, A. Lazaridis, and S. Marcel, "On the vulnerability of speaker verification to realistic voice spoofing," in *IEEE International Conference on Biometrics: Theory, Applications and Systems*, Sep. 2015.

[6] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2087–2091.

[7] P. Korshunov and S. Marcel, "Cross-database evaluation of audio-based spoofing detection systems," in *INTERSPEECH*, Sep. 2016.

[8] ——, "Joint operation of voice biometrics and presentation attack detection," in *IEEE International Conference on Biometrics: Theory, Applications and Systems*, Niagara Falls, NY, USA, Sep. 2016.

[9] A. Anjos, L. E. Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel, "Bob: a free signal processing and machine learning toolbox for researchers," in *ACM international conference on Multimedia (ACMMM)*, Oct. 2012.

[10] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[11] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 2, pp. 254–272, Apr. 1981.

[12] E. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, Apr. 1997, pp. 1331–1334.

[13] P. N. Le, E. Ambikairajah, J. Epps, V. Sethu, and E. Choi, "Investigation of spectral centroid features for cognitive load classification," *Speech Commun.*, vol. 53, no. 4, pp. 540–551, Apr. 2011.

[14] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," in *Odyssey*, Bilbao, Spain, Jun. 2016, pp. 283–290.

[15] R. Vogt and S. Sridharan, "Explicit modelling of session variability for speaker verification," *Comput. Speech Lang.*, vol. 22, no. 1, pp. 17–38, Jan. 2008.

[16] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Trans. Audio, Speech and Lang. Proc.*, vol. 19, no. 4, pp. 788–798, May 2011.

[17] S. Shiota, F. Villavicencio, J. Yamagishi, N. Ono, I. Echizen, and T. Matsui, "Voice liveness detection algorithms based on pop noise caused by human breath for automatic speaker verification," in *INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 239–243.

[18] 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 Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2280–2290, Oct. 2012.

[19] T. B. Patel and H. A. Patil, "Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech," in *INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2062–2066.

[20] A. Janicki, "Spoofing countermeasure based on analysis of linear prediction error," in *INTERSPEECH*, Dresden, Germany, Sep. 2015, pp. 2077–2081.

[21] J. Gaka, M. Grzywacz, and R. Samborski, "Playback attack detection for text-dependent speaker verification over telephone channels," *Speech Communication*, vol. 67, pp. 143 – 153, 2015.

[22] F. Alegre, A. Amehraye, and N. Evans, "A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns," in *IEEE International Conference on Biometrics: Theory, Applications and Systems*, Arlington, VA, Sep. 2013.

[23] P. Korshunov, S. Marcel, H. Muckenhirn, A. R. Gonalves, A. G. S. Mello, R. P. V. Violato, F. O. Simoes, M. U. Neto, M. de Assis Angeloni, J. A. Stuchi, H. Dinkel, N. Chen, Y. Qian, D. Paul, G. Saha, and M. Sahidullah, "Overview of BTAS 2016 speaker anti-spoofing competition," in *IEEE International Conference on Biometrics: Theory, Applications and Systems*, Niagara Falls, NY, USA, Sep. 2016, pp. 1–6.

[24] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, May 2007.

[25] I. Chingovska, A. Anjos, and S. Marcel, "Biometrics evaluation under spoofing attacks," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 12, pp. 2264–2276, Dec. 2014.

[26] H. Zen, K. Tokuda, and A. W. Black, "Review: Statistical parametric speech synthesis," *Speech Commun.*, vol. 51, no. 11, pp. 1039–1064, Nov. 2009.

[27] ISO/IEC JTC 1/SC 37 Biometrics, "DIS 30107-3:2016, information technology – biometrics presentation attack detection — part 3: Testing and reporting," American National Standards Institute, Oct. 2016.

[28] M. I. Mandasari, M. Gnther, R. Wallace, R. Saeidi, S. Marcel, and D. A. van Leeuwen, "Score calibration in face recognition," *IET Biometrics*, vol. 3, no. 4, pp. 246–256, 2014.

[29] F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 6, pp. 871–879, Jun. 1988.

[30] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in large margin classifiers*. MIT Press, 1999, pp. 61–74.

[31] U. Scherhag, A. Nautsch, C. Rathgeb, and C. Busch, "Unit-selection attack detection based on unfiltered frequency-domain features," in *INTERSPEECH*, San Francisco, USA, 09 2016, p. 2209.

Pavel Korshunov is a research associate in Biometrics group at the Idiap Research Institute (CH), working on speaker presentation attack detection (anti-spoofing) and detection of inconsistencies between audio and video. He is also a contributor to the signal processing and machine learning open source toolbox "Bob". His research interests include computer vision, crowdsourcing, high dynamic range imaging, privacy protection, speaker anti-spoofing, and machine learning.

Sébastien Marcel is a senior researcher at the Idiap Research Institute (CH), where he heads Biometrics group and conducts research on face recognition, speaker recognition, vein recognition, and presentation attack detection (anti-spoofing) in different modalities. He is a lecturer at EPFL (CH) and an Associate Editor of IEEE Signal Processing Letters. His interests are in pattern recognition and machine learning with a focus on biometrics security.