An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks

Artur Janicki, Federico Alegre and Nicholas Evans

***Abstract*—This article analyses the threat of spoofing or presentation attacks in the context of automatic speaker verification (ASV). The implementation of replay attacks requires no specific expertise nor any sophisticated equipment. Replay attacks may thus present a greater risk to ASV than voice conversion or speech synthesis which. As relatively high-technology attacks, the latter are probably beyond the means of the average fraudster. This paper compares the efficacy of each threat using strictly controlled protocols and using seven different ASV systems including a state-of-the-art iVector system with probabilistic linear discriminant analysis. Even if comparatively higher-effort spoofing attacks such as voice conversion and speech synthesis have received greater attention in the recent past, experiments show that low-effort replay attacks provoke greater equal error rates (EERs). Perhaps surprisingly, score normalisation is shown to increase the vulnerability of ASV to replay attacks. The paper also describes and assesses two replay attack countermeasures. A relatively new approach based on the local binary pattern (LBP) analysis of speech spectrograms is shown to outperform a competing approach based on the detection of far-field recordings.**

***Index Terms*—speaker verification, spoofing, replay, countermeasures, local binary patterns.**

I. Introduction

Spoofing refers to the presentation of a falsified or manipulated sample to the sensor of a biometric system in order to provoke a high score and thus illegitimate verification. In recent years, the automatic speaker verification (ASV) community has started to investigate spoofing and countermeasures actively [1], [2]. A growing body of independent work has now demonstrated the vulnerability of ASV systems to spoofing through replayed speech [3], [4], impersonation [5], [6], voice conversion [7], [8], speech synthesis [9], [10] and attacks with non-speech, artificial, tone-like signals [11], [12].

Common to the bulk of previous work is a focus on attacks which require either specific expertise, e.g. impersonation, or high-level technology, e.g. speech synthesis and voice conversion. Only replay attacks can be performed with ease, requiring neither specialist expertise nor equipment. Since they are the most easily implemented and while ignoring potential differences in efficacy, it is reasonable to assume that replay attacks will be the most prolific in practice. Nonetheless, the threat of replay attacks has not been quantified using large, standard datasets and hence never compared to that of the comparatively higher-effort attacks which have received considerably greater attention in the literature [2], [13]. With replay attacks being considerably the easiest to implement and with discreet, high quality audio equipment now available to the masses, replay attacks also merit attention.

Only few studies have addressed replay. The work in [3] assessed the vulnerabilities of an HMM-based, text-dependent ASV system with concatenated digits. While results showed that replay attacks are highly effective, experiments were conducted with data collected from only two speakers. The work in [4] investigated replay using recordings which were collected with close-talk or far-field microphones and then replayed over an analogue or digital telephony channel. The work was conducted with a similarly small corpus with data collected from five speakers and demonstrated the vulnerability of a joint factor analysis (JFA) ASV system; the FAR at the EER threshold increased from 1% to almost 70%. The authors in [14] investigated a text-dependent ASV system exposed to speech replayed using a laptop computer. This first work using the large, standard and publicly available RSR2015 corpus showed that the EER **??what system?? GMM?? iVector??** increased from approximately 4% to more than 20%. **Can you say something further about the thoroughness of the assessments in terms of the number of replay conditions? Do they all, for example, use only a single replay environment? It's about trying to identify the new contribution of this work.**

Missing from the literature, however, is a reliable comparative assessment of replay attacks to voice conversion and speech synthesis using large, standard databases and using a suitably broad range of replay scenarios. Such a study is needed in order to help prioritise future work on developing countermeasures for the greatest threats facing ASV reliability. This paper accordingly aims to assess ASV vulnerabilities to replay attacks using the same ASV systems and base corpora used in previous assessments involving voice conversion and speech synthesis spoofing attacks. In addition, the paper investigates the effectiveness of new countermeasures which aim to distinguish between genuine and replayed speech.

The paper is organised as follows. Section 2 describes speech synthesis and voice conversion spoofing attacks with a comparison to replay attacks. Section 3 presents previous and ongoing work to develop countermeasures against replay attacks, including our own work using the local binary pattern

A. Janicki is with the Institute of Telecomunications, Warsaw University of Technology, Warsaw, Poland, e-mail: A.Janicki@tele.pw.edu.pl.

F. Alegre and N. Evans are with EURECOM, Sophia Antipolis, France, e-mail: {alegre,evans}@eurecom.fr.

A. Janicki was supported by the European Union in the framework of the European Social Fund through the Warsaw University of Technology Development Programme.

F. Alegre and N. Evans were supported by the TABULA RASA project funded under the 7th Framework Programme of the European Union (grant agreement number 257289).

Manuscript received MMMM DD, YYYY; revised MMMM DD, YYYY.

analysis of speech spectrograms. A common experimental framework for the assessment of both vulnerabilities and countermeasures is presented in Section 4. Results are presented in Section 5 and our conclusions and ideas for future works are presented in Section 6.

II. Spoofing speaker verification systems

This section summarises our own approaches to assess the vulnerabilities of ASV systems to different forms of spoofing attack and the . In general, all these methods generate a spoofed speech signal $s(t)$ given the speech signal of target speaker $x(t)$. Whereas the input to a speech synthesis system is a text string, that to voice conversion and replay attacks originates from an independent input speech signal $y(t)$.

A. Speech synthesis

There is a large variety of speech synthesis algorithms, such as formant, diphone, unit-selection and statistical parametric based approaches, in addition to more recent deep-neural network architectures. Whatever the approach, the aim is to generate intelligible, natural speech for a given text string c. In the context of spoofing, a synthetic speech signal is generated according to:

$$s(t) = g_{x(t)}(c), \tag{1}$$

where $g_{x(t)}$ denotes a text-to-speech mapping generated by a synthesis system with speech units or acoustic models extracted or learned from a target speaker $x(t)$. While unit-selection approaches generally require large amounts of speaker-specific data to learn the mapping function $g_{x(t)}$, statistical parametric approaches can synthesize convincing speech signals with the adaptation of well-trained models using relatively small quantities of speaker-specific data.

Our approach to statistical parametric speech synthesis uses hidden Markov models following the approach described in [15]. Our specific implementation uses the HMM-based Speech Synthesis System (HTS)[1] where speech signals are parametrised by STRAIGHT (Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrum) features, Mel-cepstrum coefficients and the logarithm of the fundamental frequency (log F_0) with their delta and acceleration coefficients. Acoustic spectral characteristics and duration probabilities are modelled using multispace distribution hidden semi-Markov models (MSD-HSMM) [16]. Speaker dependent excitation, spectral and duration models are adapted from corresponding independent models according to a speaker adaptation strategy referred to as constrained structural maximum a posteriori linear regression (CSMAPLR) [17]. Finally, time domain signals are synthesised using a vocoder based on Mel-logarithmic spectrum approximation (MLSA) filters. They correspond to STRAIGHT Mel-cepstral coefficients and are driven by a mixed excitation signal and waveforms reconstructed using the pitch synchronous overlap add (PSOLA) method.

[1] http://hts.sp.nitech.ac.jp/

B. Voice conversion

An equation for $s(t)$ is missing - suggest simply applying IFFT to equation 2 with simple explanation regarding OLA

Voice conversion has been used to explore ASV spoofing since the late 90s [8], [7]. One of the most successful approaches is so-called Gaussian-dependent filtering approach in [18]. Here, the spoofing signal $s(t)$ (or $S(f)$ in the spectral domain) is generated by filtering at the frame level the speech signal of a spoofer $y(t)$ in the spectral domain as follows:

$$S(f) = \frac{|H_x(f)|}{|H_y(f)|} Y(f) \tag{2}$$

where $H_x(f)$ and $H_y(f)$ are the vocal tract transfer functions of the targeted speaker and the spoofer respectively. $Y(f)$ is the spoofer's speech signal in the spectral domain whereas $S(f)$ denotes the result after voice conversion. As such, $y(t)$ is mapped or converted towards the target in a spectral-envelope sense, which is sufficient to overcome most ASV systems.

$H_x(f)$ is determined from a set of two Gaussian mixture models (GMMs). The first, denoted as the automatic speaker recognition (asr) model in the original work, is related to ASV feature space and utilised for the calculation of a posteriori probabilities. The second, denoted as the filtering (fil) model, is a tied model of linear predictive cepstral coding (LPCC) coefficients from which $H_x(f)$ is derived. LPCC filter parameters are obtained according to:

$$x_{fil} = \sum_{i=1}^{M} p(g^i_{asr}|y_{asr}) \mu^i_{fil} \tag{3}$$

where $p(g^i_{asr}|y_{asr})$ is the a posteriori probability of Gaussian component g^i_{asr} given the frame y_{asr} and μ^i_{fil} is the mean of component g^i_{fil} which is tied to g^i_{asr}. $H_x(f)$ is estimated from x_{fil} using an LPCC-to-LPC transformation and a time-domain signal is synthesised from converted frames with a standard overlap-add technique. Full details can be found in [18], [19], [20].

C. Replay

Replay attacks are an example of low-effort spoofing; they require simply the replaying of a previously captured speech signal to the ASV microphone. In the absence of suitable countermeasures and considering the widespread availability of consumer devices with high-quality sound systems, replay attacks can typically be realised with ease. Furthermore, used either directly, or through the cutting and pasting of short speech intervals, replayed speech has potential to overcome both text-dependent and text-independent ASV systems. Even though the processes of recording and replaying introduce additive acoustic and convolutive channel and transducer noise, these effects can be attenuated by the noise and other intersession (channel) variability compensation techniques. All of these factors point towards the tangible threat posed by replay attacks.

Ignoring ambient noise in the acoustic environment (which is not specific to the spoofing scenario), replayed speech can be represented as:

Attack	Naïve impostor	Replay	Voice conversion	Speech synthesis
Input	impostor speech	target speech	impostor speech	text
Effort	zero	low	medium-high	high
Effectiveness	low	**(?)**	medium-high	high

TABLE I: Comparison of four different attacks in terms of speech used, required effort and effectiveness.

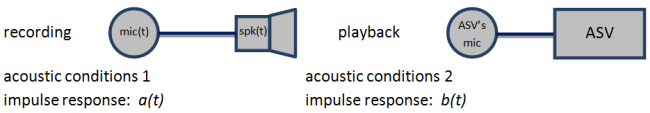

Fig. 1: A schematic diagram of the assumed replay attack configuration. **¡¡¡ Increase font sizes in the figure ¿¿¿**

$$s(t) = x(t) * h(t), \tag{4}$$

where * denotes convolution. The composite replay effects denoted by $h(t)$ include the impulse responses of replay hardware and the replay environment. It is composed by:

$$h(t) = mic(t) * a(t) * spk(t) * b(t) \tag{5}$$

where $mic(t)$ and $spk(t)$ are impulse responses of the microphone and the speaker, respectively, and where $a(t)$ and $b(t)$ are the respective impulse responses of the recording and replay environments. This scenario is illustrated in Fig. 1.

D. Qualitative comparison

Replay, voice conversion and speech synthesis spoofing are forms of *concerted-effort* impostor attacks, as opposed to the naïve or *zero-effort* impostor attacks normally used to assess ASV system performance. A qualitative comparison of all four is is illustrated in Table I, ordered by the level of effort or expertise needed to implement each attack successfully [2].

Compared to naïve impostor attacks, replay attacks require slightly increased effort; they require recording and replaying. Voice conversion and speech synthesis attacks require specialised, often complex algorithms, in addition to nay recording hardware to collect, analyse and parametrise the target and any other auxiliary speech data. They belong to a class of higher-effort spoofing attacks. While voice conversion is based upon the conversion of one speech signal to another, speech synthesis converts a text string to a speech signal, which requires a comparatively higher level of effort or expertise.

One may reasonably suppose that the effectiveness of each attack is correlated with the effort involved in their implementation; the higher the effort, the greater the impact on ASV performance. However, the proof of concept presented in [21] suggests the contrary, showing that replay attacks pose a high level of risk, being effective in overcoming an ASV system while being the easiest of all concerted-effort spoofing attacks to implement. It is the objective of this paper to investigate these contradictory findings and to compare objectively and quantitatively the comparative threat of replay spoofing to those of speech synthesis and voice conversion.

III. Replay countermeasures

Attention now turns to the *detection* of replay spoofing attacks. Given that only little work has investigated ASV vulnerabilities to such attacks, it is hardly surprising that work to develop anti-spoofing countermeasures is similarly limited. This section briefly reviews that past work and then describes two particular replay countermeasures which are explored further in this paper.

A. General approaches

One obvious approach to replay detection involves challenge-response systems which require the speaker to utter a prompted phrase [22]. Challenge-response mechanisms are a form of passive countermeasure. While having potential in preventing some forms of replay attack for some ASV systems, challenge-response countermeasures are not without negative impacts on usability which may render them undesirable for other ASV systems.

Active countermeasures have also been proposed. One such approach involves the storing of previous access attempts and their comparison to new attempts [23]. New access attempts which are deemed too close to previous attempts are rejected. A somewhat similar technique is proposed in [14], where the authors compare spectral bitmaps between access trials and previously stored recordings in a text-dependent ASV scenario.

Other, more generally applicable methods not restricted to any particular ASV scenario are based on the detection of unexpected channel artefacts indicative of recording and replaying. Two such algorithms were reported in [24] for which the EER for a baseline GMM-UBM system was shown to decrease from 40% to 10% with active countermeasures. Channel detection is the basis of the first approach investigated further in this paper.

B. Far-field channel detection

Many scenarios in which user authentication is performed by ASV involve so-called close-talk speech, i.e. situations where speech is collected from an in-situ or closely positioned microphone. Examples include telephone and logical access scenarios or critical infrastructure protection and physical access scenarios. In contrast, since they are likely to be collected surreptitiously or at-distance, replay recordings will exhibit far-field channel effects, effects which can be measured and consequently used to detect replayed speech.

This idea was first investigated in [25]. The work compared close-talk and far-field speech signals parametrised according to 12 channel-sensitive features:

- spectral ratio – sub-band energy ratio from 0-2 kHz and from 2-4 kHz;
- low frequency ratio – sub-band energy ratio from 100-300 Hz and from 300-500 Hz, calculated using speech frames only;
- total modulation index, and
- nine sub-band modulation indices – see [25] for precise sub-band bandwidths.

The spectral ratio reflect reflects the level of spectrum flattening or noise and reverberation introduced by far-field recording. The low frequency ratio reflects the level of high-pass filtering, an artefact typical of speech signals produced by small loudspeakers. The total and sub-band modulation indices reflect the level of additive and, specifically, coloured noise; higher levels of noise present in replay recordings result in lower than average modulation indices. Experiments showed that far-field recordings could be detected with 90% accuracy. **what classifier?**

C. Local binary patterns

The approach to replay detection proposed in [21] is based on the local binary pattern (LBP) analysis of speech spectrograms. Inspired by the original application to image texture analysis [26], the idea was introduced as an ASV spoofing countermeasure in [27]. As illustrated in Fig. 2, LBP analysis is applied to a mel-scaled cepstrogram with appended dynamic features. Modifications made through spoofing are assumed to disturb the natural 'texture' of genuine speech, attributes which are readily detected with LBP.

The standard LBP operator is a non-parametric 3x3 kernel which assigns a binary code to each pixel in an image according to the comparison of its intensity value to that of its eight surrounding pixels [26]. A binary value of 1 is assigned when the intensity of neighbouring pixels (here feature components) is higher, whereas a value of 0 is assigned when neighbouring pixels are of lower or equal intensity. Each pixel is thus assigned one of $2^8 = 256$ binary patterns.

LBPs are determined for each pixel in the mel-scaled cepstrogram thus resulting in a new matrix of reduced dynamic range, here referred to as a *textrogram*. The textrogram captures short-time feature motion beyond that in conventional dynamic parametrisations. Normalised histograms of pixel values constructed for each row of the textrogram are stacked vertically to obtain the anti-spoofing feature vector in the same manner as GMM mean-vectors are stacked to form supervectors. **What classifier - how are the anti-spoofing feature vectors used?** Experimental results presented in [27] showed that the LBP-based textrogram analysis is effective in detecting a range of spoofed speech signals, including artificial signals [11], [12] and speech synthesis (EERs of below 1%) while less effective in detecting voice conversion (EER in the order of 7%).

IV. Experimental setup

The comparison of replay, speech synthesis and voice conversion spoofing attacks requires the sensible choice and adaptation of a common, large and standard corpus and strictly controlled experimental protocols. They are described here.

A. Methodology

We need to say something about the noise - I presume that, since the only noise added is convolutive, there is no additive noise??? We need to be sure that the degradations in performance aren't just because the speech is more noisy.

to do later...

Due to lack of real replay recordings (e.g., similar to the MOBIO corpus collected in real environment [28]) we had to use artificial setup of replay environment, however, using impulse responses calculated using real playback hardware and real acoustic environments.

This work will also verify the effectiveness of replay detection, using two previously described replay countermeasures:

Case for emulated attacks - there is no alternative. Similar to what has been done for all other work.

In the following we describe the ASV systems used in this study, the datasets, protocols and metrics, and then the implementation of replay emulation and implementation of the countermeasures.

B. ASV systems

Since previous work in ASV anti-spoofing has shown different vulnerabilities for different ASV systems, this work was performed with a range of representative technologies, from the standard to the state-of-the-art. They include: (i) a standard Gaussian mixture model with universal background model (GMM-UBM) system with 1024 Gaussian components; (ii) a GMM supervector linear kernel (GSL) system; (iii) a GSL system with nuissance atribute projection (NAP) [30]; (iv) a GSL system with factor analysis (FA) [31]; (v) a GMM-UBM system with factor analysis; (vi) an iVector system [32] (referred to from here on as IV), and (vii) an iVector system with probabilistic linear discriminant analysis (PLDA) [33] and length normalisation [34] (referred to from here on as IV-PLDA). All seven ASV systems were tested with and without score normalisation. Symmetric normalisation (S-norm) [36] was applied to IV and IV-PLDA systems while test normalisation (T-norm) [37] was used for the others.

All ASV systems were implemented with the same LIA-SpkDet toolkit [42] and the ALIZE library [43] and were directly derived from the work in [31]. They use a common UBM with 1024 Gaussian components and a common feature parametrisation: linear frequency cepstral coefficients (LFCCs), their first derivatives and delta energy. The speech activity detector is also common to each system. It fits a 3-component GMM to the log-energy distribution and adjusts the speech/non-speech threshold according to the GMM parameters [38]. This approach has been used successfully in many independent studies, e.g. [39], [40].

C. Datasets, protocols and metrics

Experiments were performed on the male subsets of the 2004, 2005, 2006 and 2008 NIST Speaker Recognition Evaluation datasets, which are from here on referred to as NIST'0x. The NIST'04 and NIST'08 datasets were used for UBM

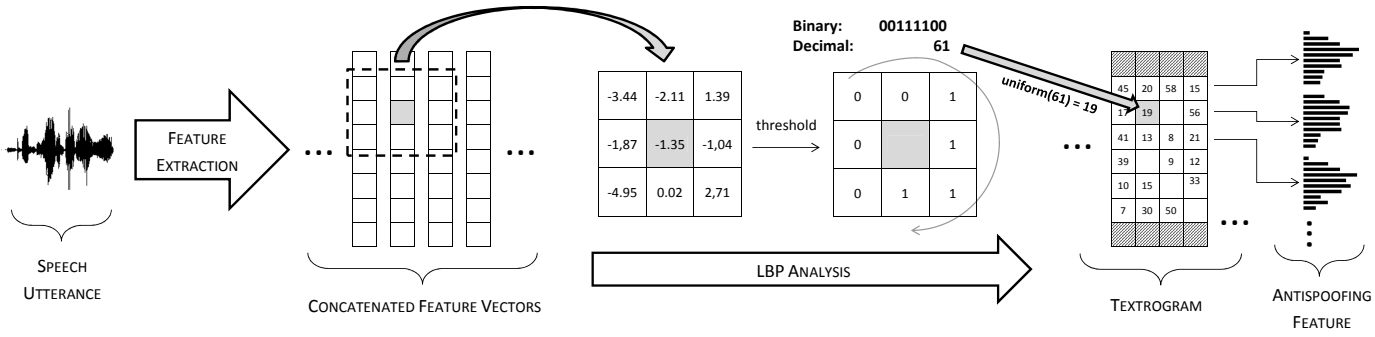

Fig. 2: Schematic diagram of LBP-based feature extraction.**Can you hyphenate antispoofing? I also suggest to reduce the size of the speech utterance and feature extraction arrow in order that you can increase the size of the textrogram, histograms and fontsizes. Possibly, remove the speech utterance and feature extraction arros entirely.**

training. The NIST'05 dataset was used for development, whereas the NIST'06 dataset was used for evaluation; only results for the latter are reported here. Due to the significant amount of data necessary to estimate the total variability matrix T used with the IV-PLDA system, the NIST'06 dataset was added to the pool of background data for development whereas the NIST'05 dataset was used for evaluation. T is thus learned using approximately 11,000 utterances from 900 speakers, while independence between development and evaluation datasets is always respected.

All experiments relate to the 8conv4w-1conv4w condition where one conversation corresponds to an average of 2.5 minutes of speech (one side of a 5 minute conversation). In all cases, however, only one of the eight, randomly selected training conversations was used for enrolment. Results presented in this paper are thus comparable only to those in the literature for the 1conv4w-1conv4w condition. Standard NIST protocols dictate in the order of 1,000 true client tests and 10,000 impostor tests for development and evaluation datasets. To assess spoofing, impostor tests are replaced with spoofed versions of the original utterance, whereas genuine client trials are unchanged. This setup conforms to the general protocols outlined in [2].

Given the consideration of spoofing, and without any standard operating criteria under such a scenario, the equal error rate (EER) metric is preferred to the minimum detection cost function (minDCF). Also reported is the spoofing false acceptance rate (SFAR, [44]) for a false rejection rate (FRR) fixed to the baseline EER.

D. Spoofing emulation

Spoofing attacks are generally assumed to be performed at the microphone level. In the case of voice conversion and speech synthesis, the practical scenario would then involve the recording of suitable data for the training or adaptation of conversion or synthesis systems, the fabrication of a spoofed utterance and then its presentation to the microphone of the ASV system. None of the past work (including the authors') follows this process, preferring instead to emulate spoofing attacks by intervening after the microphone, immediately prior to feature extraction. While it may not reflect the practical scenario, the approach can be justified [] and is that adopted here.

Voice conversion spoofing attacks were emulated with the approach described in Section II-B. A worst-case scenario is considered; conversion is performed with full prior knowledge of the ASV system, i.e. voice conversion is performed with exactly the same front-end processing as that used for ASV. The filtering model (**notation?**) and filter $H_x(f)$ use 19 LPCC and LPC coefficients respectively. Voice conversion is applied to the original impostor utterances which are converted towards the genuine speaker for any given trial.

Speech synthesis attacks are emulated according to the approach described in Section II-A using the voice cloning toolkit[2] with a default configuration and standard speaker-independent models trained on the EMIME corpus [45]. Adaptation data for each target speaker comprises three utterances (with transcriptions). For any given trial, speech synthesis spoofing attacks are generated using arbitrary text, thereby producing a spoofed utterance of duration close to that of the average test utterance.

Replay attacks are emulated according to the approach described in Section II-C. **Needed here is an explanation of why you ignored $mic(t)$ and $a(t)$.** Emulations include a random mix of three different loudspeakers impulse responses $spk(t)$ and three different replay environments $b(t)$. Speaker impulse responses were obtained from [46] and correspond to a low-quality smartphone speaker, a medium-quality tablet speaker and a high-quality standalone speaker. The impulse response and frequency responses of each are illustrated in Fig. 3. There are significant differences in the frequency responses which show in particular the high-pass functions of the lower quality devices.

The first two replay environment impulse responses were obtained from [47] and correspond to an enclosed medium-sized office and an open corridor. The impulse and frequency responses of each are illustrated in Fig. 4. **Not sure what these show?? No significant differences as far as I can see. Consider removing them.** The third impulse response simulates an anechoic chamber with a flat frequency response.

One potential problem here: in the use case relevant to this work, voice conversion speech synthesis and replay

[2]http://homepages.inf.ed.ac.uk/jyamagis/software/page37/page37.html

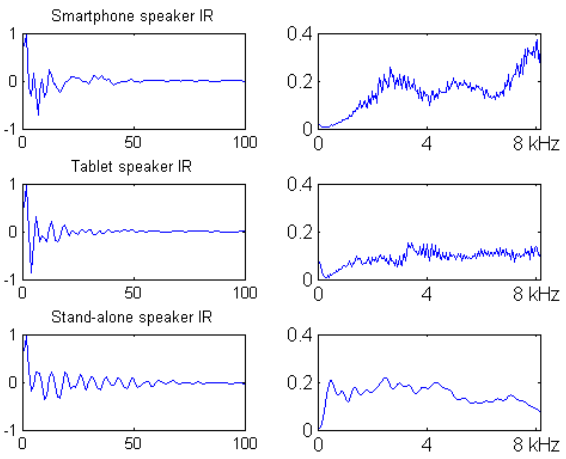

Fig. 3: Impulse left) and frequency (right) responses for three different speakers. **Crop graphs so that they fit the linewidth fully. Increase font size. Label time axes in time, not samples.**

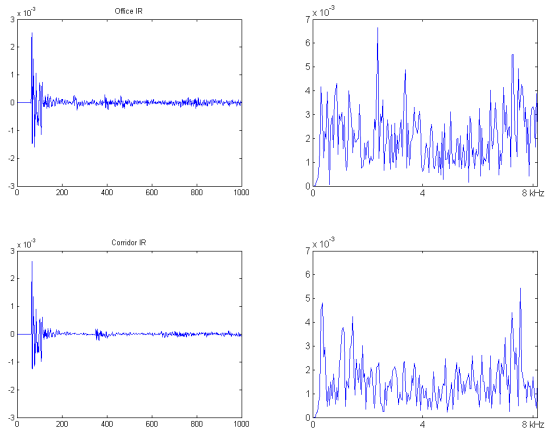

Fig. 4: Impulse (left) and frequency (right) responses for two different replay environments.

attacks will all involve replay, and thus a particular $b(t)$, yet your experiments only consider $b(t)$ for replay. The comparison is therefore not fair. I suggest to remove $b(t)$ in your replay work. In any case there is no difference between the two impulse responses.

E. Countermeasures

The far-field channel detection countermeasure was implemented according to the algorithm originally proposed in [25] and described in Section III-B. The total modulation index was calculated from the speech signal envelope which is approximated by the absolute value of the signal after down-sampling to 60 Hz. The average modulation index was calculated from speech frames whose index is above 0.75. **I struggle to see the correspondence with this description and the one in Section III-B. There, there is no mention of average modulation index, where as here there is no mention of spectral ratio, low frequency ratio and the sub-band modulation indices.**

The LBP countermeasure was implemented using the toolkit provided by The University of Oulu[3]. Normalised acoustic features used for LBP analysis were composed of 51 coefficients: 16 LFCCs and energy plus their corresponding delta and delta-delta coefficients. Analysis is based on speech frames and using only the 58 so-called uniform LBPs[4] as originally described in [26] and for speech processing in [27]. **wasn't there some prior work by Chatlanni et al?** LBP histograms are created for all but the first and last frames, thereby obtaining a $58 \times 49 = 2842$ length feature vector for each vector. **where does the 49 come from? To what does a frame refer? For me, a frame is time-dependent so I don't understand how the 49 is fixed???**

Both countermeasure algorithms were trained using a random subset of 1000 utterances from the NIST'05 dataset which were treated as described in Section **??** in order to generate suitable training data with various acoustic conditions. **So there is some overlap here! Wouldn't it have been better to use different speaker from NIST'04 or NIST'08?** Room and loudspeaker impulse responses (a lecture room, a staircase and a meeting room) different to those used for ASV experiments ensure no countermeasure over-fitting.

A binary support vector machines (SVM) classifier with 3rd order polynomial kernel was learned to differentiate genuine data from spoofed data in the case of the far-field channel detection countermeasure. In contrast, a decision table classifier was used for the LBP countermeasure. These classifiers returned the best results in each case for an area-under-the-ROC metric.

Need to explain how trials detected as spoofs are handled - they are assigned an arbitrarily low score and are thereby automatically rejected. Refer to protocol in ?? and possibly the new Speech comm journal if it is indeed mentioned.

V. Results

Attention now turns to the assessment of ASV vulnerabilities to replay spoofing, which are shown to be comparable to those of voice conversion and speech synthesis. These findings motivate the assessment of replay countermeasures.

A. Replay spoofing

Fig. 5 shows the effect of replay attacks on the score distributions for the IV-PLDA system. While the zero-effort impostor and genuine client distributions are well separated, the latter overlaps significantly with the score distribution for replay attacks collected in an office environment. Increased overlap between these distribution will degrade ASV performance.

Fig. 6 illustrates a detection error trade-off (DET) plot[5] for the same experimental setup. The lower-most, solid profile

[3]http://www.cse.oulu.fi/CMV/Downloads/LBPMatlab

[4]The subset of LBPs which contain at most two bitwise transitions from 0 to 1 or 1 to 0 when the bit pattern is traversed in circular fashion

[5]Produced with the TABULA RASA Scoretoolkit (http://publications.idiap.ch/downolads/reports/2012/Anjos_Idiap-Com-02-2012.pdf)

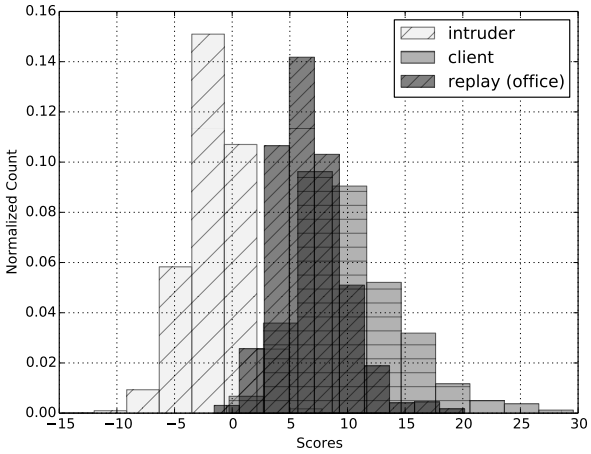

Fig. 5: Score distribution for the IV-PLDA system for replay attacks using a stand-alone speaker and emulation of an office.

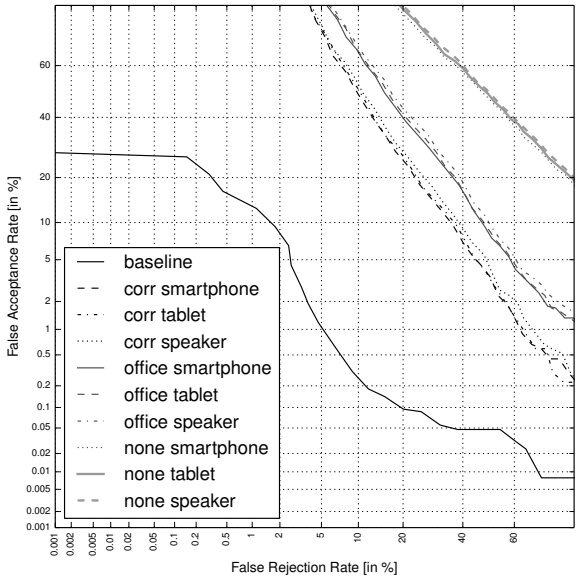

Fig. 6: DET plots for the IV-PLDA system for various replay configurations, compared to the baseline performance. **There is too much information on this plot. I suggest to include only four profiles: (i) the baseline, (ii) spoofing under the same conditions as in Fig. 5 and (iii) and (iv) with FFD and LBP countermeasures - to be referred to later.**

illustrates the performance of the baseline ASV system. The upper-most profile illustrates performance when zero-effort impostors are replaced with replay attacks. Thus, the difference between these two profiles serves as an indication of the system vulnerability to replay spoofing; in this case the degradation in performance is significant.

This trend is repeated across the full set of seven ASV systems and different replay attack configurations; results are illustrated in Table II. Results in the second row show the

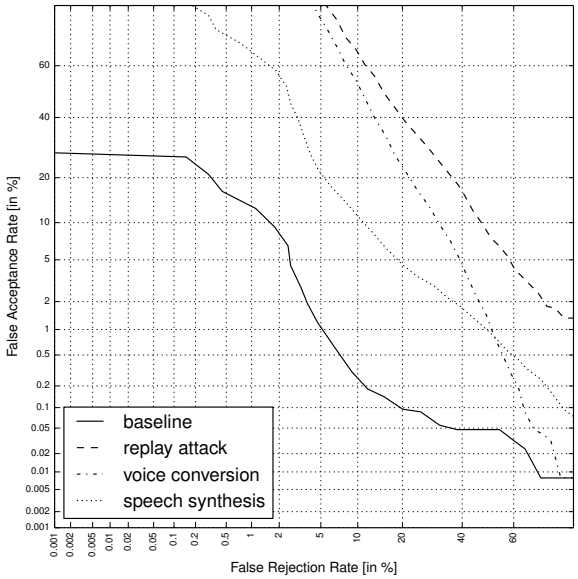

Fig. 7: DET plots for iVector-PLDA system and various attacks. **Axis keys and labels are too small.**

baseline performance for each ASV system and for only zero-effort impostors. As expected, the IV-PLDA system delivers the lowest EER.

Rows 2–5 illustrate the degradation in performance when zero-effort impostors are replaced with replay attacks. EERs are here averaged across the three loudspeaker configurations; other results not reported here showed greater sensitivity to the acoustic environment than to the loudspeaker characteristics. The performance of all seven systems degrades significantly. The EER of the most sensitive GSL system increases from 8% to in the order of 90% for all three acoustic environments. Even the EER of the most resistant SGL-FA system increases to between 22% and 50%. Finally, the EER of the state-of-the-art IV-PLDA system increases to between 25% and 50%.

B. Comparison to voice conversion and speech synthesis

DET profiles showing the vulnerabilities to replay, voice conversion and speech synthesis attacks are illustrated in Fig. 7 for the IV-PLDA system. While the comparison of such profiles is not strictly meaningful[6] it is clear that replay attacks are at least as great a threat to ASV reliability as voice conversion and speech synthesis. This trend is consistent across the full range of ASV systems. Again, while the comparisons are not strictly meaningful, results in rows 6 and 7 of Tab. II show the broad vulnerability of all seven ASV systems to replay attacks. **Can add more here once figures are available.**

[6]The authors are not experts in speech synthesis, nor voice conversion. Different spoofing algorithms may well yield different comparative results. For example, other authors [**?**], [**?**], [**?**], show considerably greater vulnerabilities for speech synthesis

	GMM	SGL	SGL-NAP	SGL-FA	FA	IV	IV-PLDA
Baseline	8.63	8.13	6.31	5.72	5.61	6.72	2.98
Replay: office environment	60.32	92.98	29.92	28.54	30.12	28.89	30.30
Replay: corridor environment	55.91	88.20	23.59	21.62	24.97	23.31	24.53
Replay: anechoic environment	64.40	96.67	49.44	49.31	49.67	49.06	49.46
Voice conversion							
Speech synthesis							

TABLE II: Equal error rates (EERs) for seven different ASV systems and for zero-effort impostors (baseline) and three different replay attack configurations (three different acoustic environments). Results are averaged across the three different loudspeaker configurations.

Environment	EER (%)			SFAR (%)		
	no CM	with FFD	with LBP	no CM	with FFD	with LBP
Office	30.30	13.62	9.56	88.70	63.93	46.29
Corridor	24.53	11.34	7.00	80.91	50.25	30.52
None	49.46	42.14	46.77	97.00	95.17	95.76

TABLE III: EER and SFAR values for various environment of replay attacks, with and without the FFD or LBP countermeasures applied, for IV-PLDA. The SFAR was measured for FRR equal to the baseline EER (2.98%).

C. Replay countermeasure

Fig. 8 illustrates the performance of the far-field and LBP-based countermeasures for replay data collected in an office environment. **Just a simple two-class classifier on its own - not integrated with the ASV systems.**

The middle two profiles in Fig. 6 show the impact of the far-field detection and LBP-based countermeasures when integrated with the IV-PLDA system. While they show reductions in the EER compared to the upper-most profile, they remain far from the original baseline.

Tab. III illustrates the performance of the IV-PLDA system...

we need to speak about this. It's difficult to know why you show results only for the IV-PLDA from here on. Why not the SGL-FA system which seems to be the most robust? I'm also wondering if we can merge tables II and III to save some space. Probably the SFAR does not bring a great deal new to the paper beyond what the EER metrics already show.

The detailed results of experiments with FFD and LBP countermeasures for various replay environment, averaged across the replay devices, are presented in Table III. It shows that the countermeasure performance varies depending on acoustic environment. The relative improvement caused by the countermeasures turned out to be the highest for the office – in this case the EER decreased from 30% down to less than 14% for FFD and less than 10% for LBP. Also for the corridor LBP turned out to be more effective than FFD – 7% EER vs. 11%, respectively, also the SFAR result was much lower (30% vs. 46%). When acoustic conditions were not considered, both countermeasures performed poorly, with the EER results slightly better for FFD. This is also visualised by the shape of the DET plots presented in Fig 8.

Table **??** displays the results of the countermeasure experiments for various replay devices, averaged across different acoustic environments (only office and corridor were taken into account, as they are by far most realistic). Also here the LBP-based countermeasure yields better results than FFD. Both countermeasures helped most for a smartphone and a tablet (the EER values decreased to less than 12% for FFD and to 7.5% for LBP). The results for a stand-alone speaker are only slightly worse (14% and 10%, respectively), most likely due to higher quality of this device (see much better frequency response shown in Fig. 3).

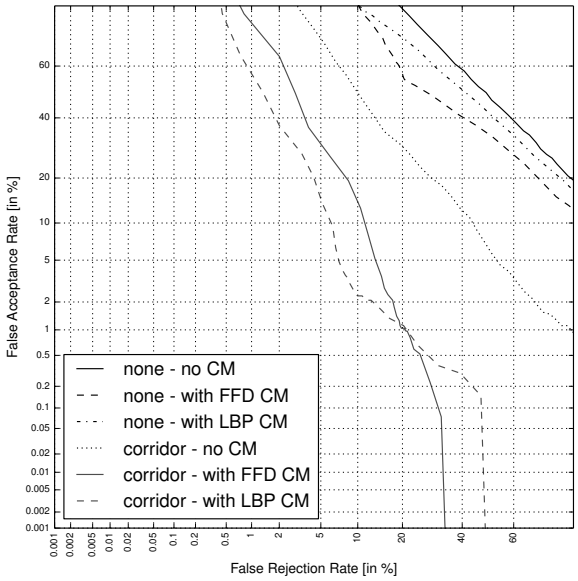

Fig. 8: DET plots for IV-PLDA system for various replay environments, with and without the FFD or LBP countermeasures. **I suggest to replace this DET plot with an ASV-independent assessment - i.e. a pure assessment of the CM on its own.**

D. Interpretation of results

The results presented above corroborate the findings of previous work performed with considerably smaller databases. While reporting the first work with NIST databases and replay attacks emulated in similar fashion to all past work with voice conversion and speech synthesis, there are some aspects of the methodology which must be highlighted in order that results are sensibly interpreted.

First, the results show that replay attacks are of comparable threat to voice conversion and speech synthesis; they are not intended, nor sufficient to show that replay attacks cause the greatest degradations in ASV performance. The relative degradations are naturally a function of the authors' effort and expertise in each approach, whereas they are not expert in voice conversion, nor speech synthesis. Other, perhaps more sophisticated voice conversion and speech synthesis algorithms may well cause greater degradations in ASV performance than those reported here. The work should therefore be taken as evidence only of the importance of ensuring ASV robustness to replay spoofing in the future.

These arguments lead naturally to the comparisons regarding the effort and expertise involved in the implementation of each spoofing attack. Whereas voice conversion and speech synthesis are relatively high-effort, high-technology attacks, replay spoofing is implemented easily, without any specific expertise or equipment. Accordingly, the threat posed by replay attacks should be considered greater in a practical perspective, even in the case that future work shows degradations in ASV performance cased by replay are lower than those caused by voice conversion and speech synthesis; they are in any case a significant threat and the most likely form of spoofing 'in the wild'.

E. Future work

There are also some aspects of the results which require further work to investigate and explain. Of particular note is the vulnerability of the SGL system whose EER increases to 90% when subjected to replay attacks; all other ASV systems have EERs in the order of 25% to 50%. Our initial investigations have shown that score normalisation may be the cause. Other experiments without score normalisation showed EERs of between 30% and 50%. Accordingly, further work is required to study the impact of score normalisation on ASV vulnerabilities to spoofing.

There are also some as yet unanswered questions regarding the impact of channel compensation. The main idea behind the detection of replay attacks essentially involves the detection of unexpected channels. Thus, while channel compensation has unquestionable utility in ensuring the usability of ASV across different devices, for example, it may also work to the fraudster's advantage; it can make the very channel characteristics needed to prevent replay attacks more difficult to detect. There is some evidence of this in the difference between results for the IV and IV-PLDA systems. While the latter gives better baseline performance, it is also marginally more vulnerable to replay attacks. Further work should investigate this link.

Finally, further work should assess the threat with a more application-driven methodology. That chosen here was motivated by the need to compare the threat of replay to that of voice conversion and speech synthesis, simply as a means of prioritising future work, i.e. to determine whether or not replay is a genuine threat. While the the NIST SRE datasets were essential in order to support comparisons to other work, and are almost a requirement as regards publication, they were designed more for surveillance and security scenarios rather than authentication applications more relevant to spoofing. Since replay spoofing is undeniably application specific, future work should thus consider more the application than the dataset. It is stressed, however, that this criticism can be levelled equally to all of the past work.

VI. Conclusions

This paper assesses the threat to automatic speaker verification of replay spoofing attacks. The threat is shown to be of least similar significance to that of voice conversion and speech synthesis. The latter have attracted by far the greatest attention in the literature to date. This paper argues that replay spoofing deserves greater attention; in contrast to voice conversion and speech synthesis, they require no specialist expertise or equipment and are therefore the most likely attack from a practical perspective.

While there is potential to detect replay spoofing attacks, the approaches assessed in this paper rely upon the presence of measurable channel effects. Approaches to channel compensation commonly used in today's state-of-the-art speaker verification systems can reduce channel effects thereby increasing the difficulty in detecting replay attacks. Finally, replay attacks recorded with high-quality sound systems will be almost impossible to detect. Alternative countermeasures will then be needed. Challenge-response mechanisms are a suitable candidate but lack scientific and objective validation.

Lastly, the practical use of these and any other replay countermeasure will be dependent on the application. It is extremely difficult to assess replay spoofing, hence the rather bold assumptions in this work and the use of emulated attacks. While this is also a weakness of much of the past work, it is particularly true in the case of replay. Further work will be needed to assess replay, and spoofing in general, with an application-driven methodology. In all of these, there will be potential for replay attacks. Since any security system is only as strong as its weakest link, replay is a genuine threat.

References

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

[2] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: a survey," *Speech Communications*, vol. 66, pp. 130–153, 10 2014.

[3] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification - a study of technical impostor techniques," in *European Conference on Speech Communication and Technology*, 1999, pp. 1211–1214.

[4] J. Villalba and E. Lleida, "Speaker verification performance degradation against spoofing and tampering attacks," in *FALA workshop*, 2010, pp. 131–134.

[5] M. Blomberg, D. Elenius, and E. Zetterholm, "Speaker verification scores and acoustic analysis of a professional impersonator," in *Proc. FONETIK*, 2004.

[6] M. Farrús, M. Wagner, J. Anguita, and J. Hernando, "How vulnerable are prosodic features to professional imitators?" in *Proc. IEEE ODYSSEY - The Speaker and Language Recognition Workshop*, 2008.

[7] P. Perrot, G. Aversano, R. Blouet, M. Charbit, and G. Chollet, "Voice forgery using ALISP : Indexation in a Client Memory," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, vol. 1, 2005, pp. 17 – 20.

[8] B. Pellom and J. Hansen, "An experimental study of speaker verification sensitivity to computer voice-altered imposters," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, vol. 2, 1999, pp. 837–840.

[9] T. Masuko, T. Hitotsumatsu, K. Tokuda, and T. Kobayashi, "On the security of HMM-based speaker verification systems against imposture using synthetic speech," in *Proc. EUROSPEECH*, 1999.

[10] P. L. D. Leon, V. R. Apsingekar, M. Pucher, and J. Yamagishi, "Revisiting the security of speaker verification systems against imposture using synthetic speech," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, march 2010, pp. 1798 –1801.

[11] F. Alegre, R. Vipperla, N. Evans, and B. Fauve, "On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals," in *Proc. 12th EUSIPCO*, 2012.

[12] F. Alegre, R. Vipperla, and N. Evans, "Spoofing countermeasures for the protection of automatic speaker recognition from attacks with artificial signals," in *Proc. 13th Interspeech*, 2012.

[13] N. Evans, T. Kinnunen, J. Yamagishi, Z. Wu, F. Alegre, and P. L. D. Leon, *Speaker recognition anti-spoofing*, S. Marcel, S. Li, and M. Nixon, Eds. Springer, 2014.

[14] Z. Wu, S. Gao, E. S. Chng, and H. Li, "A study on replay attack and anti-spoofing for text-dependent speaker verification," in *Proc. APSIPA ASC 2014*, 2014.

[15] J. Yamagishi, T. Nose, H. Zen, Z.-H. Ling, T. Toda, K. Tokuda, S. King, and S. Renals, "Robust speaker adaptive HMM based Text-to-Speech Synthesis," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 17, no. 6, pp. 1208–1230, 2009.

[16] M. Russell and R. Moore, "Explicit modelling of state occupancy in hidden markov models for automatic speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 1985, pp. 5–8.

[17] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of Speaker Adaptation Algorithms for HMM-based Speech Synthesis and a Constrained SMAPLR Adaptation Algorithm," *IEEE transactions on Audio, Speech & Language Processing*, vol. 17, no. 1, pp. 66–83, 2009.

[18] D. Matrouf, J.-F. Bonastre, and J.-P. Costa, "Effect of impostor speech transformation on automatic speaker recognition," *Biometrics on the Internet*, p. 37, 2005.

[19] J.-F. Bonastre, D. Matrouf, and C. Fredouille, "Transfer function-based voice transformation for speaker recognition," in *Proc. IEEE ODYSSEY - The Speaker and Language Recognition Workshop*, 2006, pp. 1–6.

[20] ——, "Artificial impostor voice transformation effects on false acceptance rates," in *Proc. Interspeech*, 2007, pp. 2053–2056.

[21] F. Alegre, A. Janicki, and N. Evans, "Re-assessing the threat of replay spoofing attacks against automatic speaker verification," in *Proc. International Conference of the Biometrics Special Interest Group (BIOSIG 2014)*, 2014, (submitted).

[22] D. Petrovska-Delacrétaz and J. Hennebert, "Text-prompted speaker verification experiments with phoneme specific mlps," in *Proc. ICASSP'98*, 1998.

[23] W. Shang and M. Stevenson, "Score normalization in playback attack detection," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, March 2010, pp. 1678–1681.

[24] Z.-F. Wang, G. Wei, and Q.-H. He, "Channel pattern noise based playback attack detection algorithm for speaker recognition," in *Machine Learning and Cybernetics (ICMLC), 2011 International Conference on*, vol. 4, July 2011, pp. 1708–1713.

[25] J. Villalba and E. Lleida, "Preventing replay attacks on speaker verification systems," in *Security Technology (ICCST), 2011 IEEE International Carnahan Conference on*, Oct 2011, pp. 1–8.

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

[27] F. Alegre, R. Vipperla, A. Amehraye, and N. Evans, "A new speaker verification spoofing countermeasure based on local binary patterns," Lyon, France, 2013.

[28] E. Khoury, B. Vesnicer, J. Franco-Pedroso, R. Violato, Z. Boulkcnafet, L. Mazaira Fernandez, M. Diez, J. Kosmala, H. Khemiri, T. Cipr, R. Saeidi, M. Gunther, J. Zganec-Gros, R. Candil, F. Simoes, M. Bengherabi, A. Alvarez Marquina, M. Penagarikano, A. Abad, M. Boulayemen, P. Schwarz, D. Van Leeuwen, J. Gonzalez-Dominguez, M. Neto, E. Boutellaa, P. Gomez Vilda, A. Varona, D. Petrovska-Delacretaz, P. Matejka, J. Gonzalez-Rodriguez, T. Pereira, F. Harizi, L. Rodriguez-Fuentes, L. El Shafey, M. Angeloni, G. Bordel, G. Chollet, and S. Marcel, "The 2013 speaker recognition evaluation in mobile environment," in *Biometrics (ICB), 2013 International Conference on*, June 2013, pp. 1–8.

[29] A. Roy, M. Magimai-Doss, and S. Marcel, "A fast parts-based approach to speaker verification using boosted slice classifiers," *Information Forensics and Security, IEEE Transactions on*, vol. 7, no. 1, pp. 241–254, Feb 2012.

[30] W. M. Campbell, D. Sturim, D. A. Reynolds, and A. Solomonoff, "Svm based speaker verification using a gmm supervector kernel and nap variability compensation," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, vol. 1, may 2006, p. I.

[31] B. Fauve, D. Matrouf, N. Scheffer, J.-F. Bonastre, and J. S. D. Mason, "State-of-the-art performance in text-independent speaker verification through open-source software," *IEEE Transactions on Audio Speech and Language processing*, vol. 15, no. 7, pp. 1960–1968, 2007.

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

[33] P. Li, Y. Fu, U. Mohammed, J. Elder, and S. Prince, "Probabilistic models for inference about identity," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 1, pp. 144–157, 2012.

[34] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *International Conference on Speech Communication and Technology*, 2011, pp. 249–252.

[35] M. McLaren, R. Vogt, B. Baker, and S. Sridharan, "A comparison of session variability compensation approaches for speaker verification," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 4, pp. 802–809, Dec 2010.

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

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

[38] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP J. Appl. Signal Process.*, vol. 2004, pp. 430–451, Jan. 2004. [Online]. Available: http://dx.doi.org/10.1155/S1110865704310024

[39] I. Magrin-Chagnolleau, G. Gravier, and R. Blouet, "Overview of the 2000-2001 elisa consortium research activities," in *2001: A Speaker Odyssey-The Speaker Recognition Workshop*, 2001.

[40] B. Fauve, H. Bredin, W. Karam, F. Verdet, A. Mayoue, G. Chollet, J. Hennebert, R. Lewis, J. Mason, C. Mokbel *et al.*, "Some results from the biosecure talking face evaluation campaign," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 4137–4140.

[41] M. Sahidullah and G. Saha, "Comparison of speech activity detection techniques for speaker recognition," *arXiv e-preprint*, Oct. 2012.

[42] J.-F. Bonastre, N. Scheffer, D. Matrouf, C. Fredouille, A. Larcher, A. Preti, G. Pouchoulin, N. Evans, B. Fauve, and J. Mason, "ALIZE/SpkDet: a state-of-the-art open source software for speaker recognition," in *Proc. IEEE ODYSSEY - The Speaker and Language Recognition Workshop*, vol. 5, 2008, p. 1.

[43] J.-F. Bonastre, N. Scheffer, C. Fredouille, and D. Matrouf, "NIST'04 speaker recognition evaluation campaign: new LIA speaker detection platform based on ALIZE toolkit," in *NIST SRE'04*, 2004.

[44] P. Johnson, B. Tan, and S. Schuckers, "Multimodal fusion vulnerability to non-zero effort (spoof) imposters," in *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, Dec 2010, pp. 1–5.

[45] M. Wester, "The EMIME bilingual database," The University of Edinburgh, Tech. Rep., 2010.

[46] A. Brown, "Aaron brown sound web page," http://www.aaronbrownsound.com/, Apr 2014.

[47] M. Jeub, M. Schäfer, and P. Vary, "A binaural room impulse response database for the evaluation of dereverberation algorithms," in *Proceed-*

ings of the 16th International Conference on Digital Signal Processing, ser. DSP'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 550–554.

Artur Janicki Biography text here.

Federico Alegre Biography text here.

Nicholas Evans Biography text here.