Interspeech 2018
2-6 September 2018, Hyderabad

A Deep Identity Representation for Noise Robust Spoofing Detection

Alejandro Gomez-Alanis[1], *Antonio M. Peinado*[1], *Jose A. Gonzalez*[2], *and Angel M. Gomez*[1]

[1]University of Granada, Granada, Spain
[2]University of Malaga, Malaga, Spain
{agomezalanis,amp,amgg}@ugr.es, jgonzalez@lcc.uma.es

Abstract

The issue of the spoofing attacks which may affect automatic speaker verification systems (ASVs) has recently received an increased attention, so that a number of countermeasures have been developed for detecting high technology attacks such as speech synthesis and voice conversion. However, the performance of anti-spoofing systems degrades significantly in noisy conditions. To address this issue, we propose a deep learning framework to extract spoofing identity vectors, as well as the use of soft missing-data masks. The proposed feature extraction employs a convolutional neural network (CNN) plus a recurrent neural network (RNN) in order to provide a single deep feature vector per utterance. Thus, the CNN is treated as a convolutional feature extractor that operates at the frame level. On top of the CNN outputs, the RNN is employed to obtain a single spoofing identity representation of the whole utterance. Experimental evaluation is carried out on both a clean and a noisy version of the ASVSpoof2015 corpus. The experimental results show that our proposals clearly outperforms other methods recently proposed such as the popular CQCC+GMM system or other similar deep feature systems for both seen and unseen noisy conditions.

Index Terms: Spoofing detection, noise robustness, speaker verification, deep learning, missing-data masks.

1. Introduction

In recent years, automatic speaker verification (ASV) [1, 2, 3] technology has gained an increased interest due to its commercial applications. As the importance of this technology grows, so does the concerns about its security. In ASV, an impostor could gain unauthorized access to a system by using spoofing attacks [4]. For ASV, four types of spoofing attacks have been identified [5]: (i) replay (i.e. using pre-recorded voice of the target user), (ii) impersonation (i.e. mimicking the voice of the target voice), and also either (iii) text-to-speech synthesis (TTS) or (iv) voice conversion (VC) systems to generate artificial speech resembling the voice of a legitimate user. In this work, we are interested in providing anti-spoofing measures against spoofing attacks based on either VC or TTS.

As shown in [5], state-of-the-art ASV systems are highly vulnerable to TTS/VC based spoofing attacks. Thus, the development of anti-spoofing techniques is a subject that has recently attracted the attention of a number of researchers [4, 5]. Broadly speaking, these techniques attempt to identify synthetic speech by detecting the artifacts produced by the speech vocoders used in TTS/VC systems. For instance, a popular approach attempts to detect the phase artifacts introduced by minimum-phase vocoders [8]. Although these countermeasures have been successfully applied in clean conditions, they are known to fail when the attacks are deployed in noisy scenarios. As shown in [10], the performance of the spoofing countermeasures trained on clean conditions is significantly degraded in noisy scenarios and this deterioration increases as the signal-to-noise ratio (SNR) decreases. Thus, providing robust anti-spoofing techniques against noisy conditions is also becoming a key issue.

The literature about ASV anti-spoofing in noisy conditions is scarce due to the novelty of this area. One of the first studies was carried out in [11], where the robustness of various front-end features were evaluated under different noisy conditions. In [10], a neural network was trained as an anti-spoofing detection system, and several front-end features were tested under five additive noises and reverberant conditions. Also, the use of frame-level deep features were proposed and evaluated in [12], being justified as a mean to extract useful information for spoofing detection from the noise corrupted spectral features. These deep features were extracted using several neural network architectures.

In this work, we propose a CNN+RNN system to get a single spoofing identity representation per utterance, which is robust to noisy and reverberant conditions. Although a CNN+RNN model was firstly proposed in [20] for anti-spoofing, our proposed system presents four important differences: (1) it uses context windows to avoid applying a padding or cropping method to the input of the system, (2) the CNN and RNN are not optimized simultaneously, (3) it introduces noise features for an increased noise robustness, and (4) the CNN+RNN framework is not used as the final classifier. In contrast to the DNN and CNN systems proposed in [12], our spoofing identity representation is not obtained by averaging the frame-level deep features of an utterance. Instead of that, we propose a recurrent layer which is fed with the outputs of the CNN in order to learn long-term dependencies. Furthermore, we propose a novel methodology for noise awareness based on the use of missing-data masks [6, 7], which define the reliability of the spectro-temporal regions in the noisy spectrum.

This paper is organized as follows. Section 2 describes the proposed (CNN+RNN) deep feature extractor and the LDA back-end employed along the work. Then, in Section 3, we outline the speech corpora, the network training, and the performance evaluation details. Section 4 discusses the results of our system under clean and noisy scenarios, and shows a comparison with other relevant anti-spoofing systems. Finally, we present the conclusions derived from this research in Section 5.

2. System description

This section is devoted to the description of the proposed anti-spoofing feature extraction procedure. First, Section 2.1 describes the different front-end stages: input spectral feature extraction, frame-level CNN deep feature extraction, and RNN (utterance-level) identity feature extraction. The linear discriminant analysis (LDA) classifier employed as back-end is detailed in Section 2.2. A block diagram of the proposed feature extrac-

10.21437/Interspeech.2018-1909

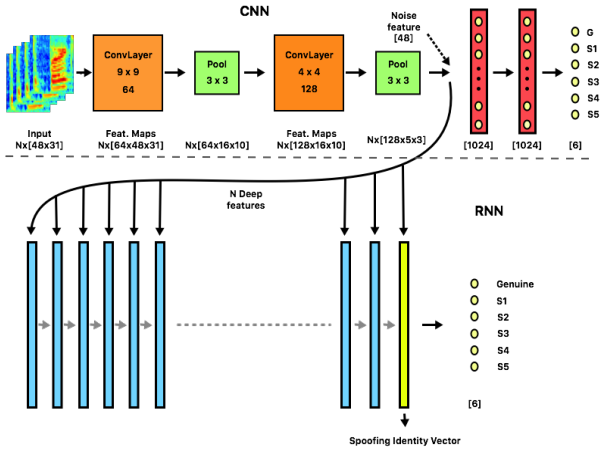

Figure 1: *Deep learning framework to extract a spoofing identity representation per utterance (N represents the number of context windows per utterance).*

tion system is shown in Fig. 1. All deep models are implemented with the Tensorflow toolkit [14].

2.1. Front-end

The proposed CNN+RNN front-end system provides a single spoofing identity representation of the whole utterance as shown in Fig. 1. The frame window size is 25 ms with 10 ms of frame shift. A context window of 31 frames (centered at the frame being processed) is used to obtain the input signal spectral features which are fed into the system. Then, the CNN provides a deep feature vector per window, and all deep features vectors of the considered utterance are processed by the RNN in order to obtain the spoofing identity vector of the utterance.

As demonstrated in [12], traditional log MEL filterbank features (FBANK) are more robust to noise than the recently proposed constant Q cepstral coefficients (CQCCs) [16]. Thus, we have also adopted FBANK features. In contrast to [12] and [15], we use a 48-dim static FBANK without delta and acceleration coefficients, as we have realized that the context window of 31 frames is already exploiting the correlations between successive frames. Therefore, a higher spectral resolution is achieved while the size of the spectral feature vector is smaller than in [12]. The result of this processing block is a feature matrix of size $[48 \times 31]$ per frame. The FBANK features are obtained using the HTK toolkit [18]. Mean and variance normalization is applied to the resulting FBANK parameters.

In our architecture, the CNN plays the role of a frame-level deep feature extractor providing one feature vector for each context window of 31 frames. In order to do this, the CNN acts as a classifier whose task consists of determining whether the input features are either genuine or belong to one of the 5 spoofing attacks (S1, S2, S3, S4 or S5) present in the training set. Our CNN uses 2 convolutional and pooling layers as feature extractors, followed by 2 fully connected layers of 1024 sigmoid neurons with a softmax layer of 6 neurons as classification layer. To prevent the problem of overfitting, 50 % and 40 % dropout is applied to the 2 fully connected layers, respectively.

The first convolutional layer obtains 64 feature maps using 9×9 filters. This results in a volume of size $[64 \times 48 \times 31]$. Then, the pooling layer performs a downsampling operation along the spatial dimensions (width=31, height=48) using 3×3 filters, resulting in a smaller volume of $[64 \times 6 \times 10]$. The second convolutional layer obtains 128 features using 4×4 filters, which results in a volume of $[128 \times 16 \times 10]$. After that, a second pooling layer with 3×3 filters reduces the final volume to $[128 \times 5 \times 3]$. In the two pooling layers, we use a stride of 3 and a VALID padding. Finally, the 128 features of size $[5 \times 3]$ are concatenated to make up a deep feature vector of 1920 components.

As shown in Fig. 1, the deep features obtained from CNN are fed into an RNN, which computes the anti-spoofing identity vector for the utterance. The advantage of using an RNN is its ability for learning the long-term dependencies of the subsequent deep feature vectors. The activation function of the RNN is a gated recurrent unit (GRU) [19]. Finally, a fully connected layer containing 6 neurons (one per class: genuine, S1, S2, S3, S4 and S5) is connected to the output of the last time step, followed by a softmax layer. The state of the last time step represents the single deep identity spoofing vector of the whole utterance.

In addition to multi-condition training, in this work we evaluate two types of noise features aimed at improving the robustness against noise of our anti-spoofing detection methods. First, noise-aware training (NAT) is implemented by using a noise code per utterance, which is computed by averaging the 48 spectral features of the first 10 frames of the utterance. Second, in order to have a more finer grain detail about the reliability of each spectro-temporal region of the noisy utterance, we propose the use of soft masks [6, 7]. Each mask defines, for each spectral feature, the probability that this feature is contaminated by noise. To compute the mask, noise is firstly estimated for each frame by linearly interpolating two independent noise estimates computed by averaging the first and last $N = 10$ frames of each utterance. Next, the SNR is estimated from the original noisy features and the noise estimates. A sigmoid function is finally applied to the SNR values to compress them between the $[0, 1]$ range in order to obtain the missing-data masks. In both cases (NAT and missing-data masks), the noise features are appended to the output of the convolutional layers of the CNN, which results in deep features of 1968 components as shown in Fig. 1. Finally, this augmented deep feature vector is fed into the RNN.

2.2. Back-end

As shown in [15], a linear discriminant analysis (LDA) back-end achieves the best performance for anti-spoofing in comparison with other state-of-the-art techniques. In general, LDA classification has shown a high performance for a variety of tasks [21, 22]. Thus, we will employ an LDA back-end to assign a genuine speech confidence score to each utterance. Our LDA classifier uses 6 classes which represent genuine speech and the five known spoofing attacks considered in the training set. The genuine class score is the only used for decision.

3. Experimental framework

In order to evaluate the performance of our proposed techniques, the ASVspoof 2015 corpus [9], a well-known database containing data from different spoofing attacks under clean conditions, was employed. Also, a noisy version of this corpus [10] was also considered to evaluate the robustness of the different proposals against noise. Details about the methodology followed for training and testing are given in this section.

3.1. Speech corpus

The clean ASVspoof 2015 corpus [9] defines three datasets (training, development and evaluation), each one containing a mix of genuine and spoofed speech. Spoofing attacks were generated either by TTS or VC. A total of 10 types of spoofing attacks (S1 to S10) are defined: three of them are implemented using speech synthesis (S3, S4 and S10), and the remaining seven ones (S1, S2, S5, S6, S7, S8 and S9) using different voice conversion systems. Attacks S1 to S5 are referred to as *known attacks*, since the training and development sets contain data for these types of attacks, while attacks S6 to S10 are referred to as *unknown attacks*, because they only appear in the evaluation set. More details about this corpus can be found in [9].

In order to evaluate the robustness of our proposals against noise, the noisy version of the ASVspoof 2015 corpus (described in [10]) was also employed. This version was generated by artificially distorting the signals in the original, clean corpus with different noise types at various signal-to-noise ratio (SNR) levels. In particular, 5 additive noise types (white noise, babble, volvo, street and café) were added to the clean signals at three SNR levels (20, 10 and 0 dB). Three reverberant scenarios were also considered by convolving the clean signals with three room impulse responses (RIR) with different T60 values (0.3, 0.6 and 0.9s). Thus, in total, 18 different noisy conditions (15 additive noises and 3 reverberant conditions) were finally considered. As suggested in [12], data in the noisy corpus was divided into *seen* and *unseen* conditions for further realism. The *seen condition* consists of white, babble and street noises, and the 3 reverberant conditions, which are present in the training, development and evaluation datasets. On the other hand, the *unseen condition* contains café and volvo noises, which are only present in the development sets. More details about this corpus are given in [10, 12].

3.2. Training

As mentioned in the previous section (front-end description), FBANK spectral features, extracted from a 48-filter Mel-filterbank, are used to represent the speech signal. These features were normalized in mean and variance.

In the clean scenario, the original ASVspoof 2015 corpus [9] is used for training and evaluation. Here, our anti-spoofing system does not append any noise features at the output of the convolutional layers.

In the noisy scenario, the noisy version of the ASVspoof 2015 corpus [10] is used. Here, multi-condition training is applied in order to get higher level features that are more robust against noise. Furthermore, noise features are appended to the output of the convolutional layers in order to increase this robustness. We have tested two types of noise features: (1) noise aware training (NAT), and (2) soft noise masks (MASK). This augmented feature vector is fed into both the upper layers of the CNN and the RNN.

In both scenarios, the separately CNN and RNN are trained using Adam optimizer [23]. Also, early stopping is applied in order to stop the training process when no improvement is obtained after ten iterations.

3.3. Performance evaluation

The equal error rate (EER) is used to evaluate the system performance. As described in the ASVspoof 2015 challenge evaluation plan [9], the EER was computed independently for each spoofing algorithm and then the average EER across all attacks was used. To compute the average EER, we used the Bosaris toolkit [13].

4. Results

The results by our proposal and other techniques from the literature are presented below.

4.1. Clean scenario

Table 1 shows a comparison of the performance of different anti-spoofing systems in the clean version of ASVspoof 2015 database. The FBANK+CNN+LDA system has already been proposed in [12], but as its performance is not provided in this reference for the clean scenario, we have evaluated instead our proposed system removing the RNN and averaging the deep features for getting the identity spoofing vector of the utterance as in [12]. The CQCC+GMM system achieves the best average performance, although our proposed system (FBANK+CNN+RNN+LDA) achieves the best results for the known attacks. Compared to the rest of deep learning systems (Spectro+CNN+RNN [20], Best DNN [15], Best RNN [15] and FBANK+CNN+LDA), our proposal outperforms all of them in the known and unknown attacks. Particularly noteworthy is the result of our system in the S10 attack. The CFCC-IF system [17] achieves a lower EER in the S10 attack than our proposed system, but our proposal performs 0.21 % better on average when considering all the attacks.

4.2. Noisy scenario

Table 2 compares the performance of five different systems on the noisy version of the ASVspoof 2015 database. Multi-condition training was used in all cases. In the table, the performance of NAT and MASK techniques for noise awareness is also evaluated. The last four systems use FBANK as input features and an LDA as final classifier. For the sake of clarity, these two acronyms have been removed.

As shown in Table 2, when multi-condition training is used, our CNN+MASK+RNN system achieves the best overall performance in the clean condition, even outperforming CQCC+GMM, which was the best system in Table 1. Furthermore, the use of the RNN decreases the total average EER from 1.09 % and 0.93 % to 0.59 % and 0.47 % when using NAT and MASK techniques, respectively. This result shows the importance of getting the identity spoofing representation of an utterance using a recurrent layer to learn the long-term dependencies, instead of averaging the deep features as in [12].

When evaluated under noisy conditions, the CQCC+GMM system performs very poorly even for the seen noises (those used for multi-condition training). On the contrary, our CNN+MASK+RNN system achieves the best results with an overall relative improvement of 26.6 % compared to CQCC+GMM. Moreover, the use of the proposed MASK noise features provides the best robustness against noise outperforming NAT in both models (CNN and CNN+RNN). Specifically, it reduces a 0.6% and 0.3% the total average EER, respectively.

The CQCC+GMM performs again very poorly in the unseen noise conditions when compared to our proposals. As in the seen noisy conditions, the MASK noise feature obtains significantly better results than NAT and so does the hybrid CNN+RNN model in comparison with the CNN model.

To sum up, the results under noisy conditions show that our two proposals (RNN for utterance-level identity representation and MASK noise awareness) significantly improve the perfor-

Table 1: *Comparison on evaluation clean dataset for each spoofing attack in terms of (%) EER*

System	Known Attacks						Unknown Attacks						Total
	S1	S2	S3	S4	S5	Avg.	S6	S7	S8	S9	S10	Avg.	Avg.
CQCC + GMM [16]	**0.00**	0.10	**0.00**	**0.00**	0.13	0.05	**0.10**	0.06	1.03	0.05	**1.07**	**0.46**	**0.26**
Spectro + CNN + RNN [20]	0.16	0.50	0.03	0.03	1.38	0.40	0.85	0.91	0.03	0.59	14.27	3.33	1.86
Best DNN [15]	**0.00**	0.10	**0.00**	**0.00**	0.20	0.10	0.20	**0.00**	**0.00**	**0.00**	25.5	5.10	2.60
Best RNN [15]	**0.00**	0.90	**0.00**	**0.00**	0.30	0.20	0.80	0.50	**0.00**	0.70	10.70	2.50	1.40
CFCC-IF [17]	0.10	0.86	**0.00**	**0.00**	1.08	0.41	0.85	0.24	0.14	0.35	8.49	2.01	1.21
FBANK + CNN + LDA	0.02	1.07	**0.00**	**0.00**	0.51	0.32	1.03	0.44	0.05	0.51	20.57	4.52	2.42
FBANK + CNN + RNN + LDA	**0.00**	**0.08**	0.00	0.00	**0.07**	**0.03**	0.22	0.10	0.08	0.13	9.34	1.97	1.00

Table 2: *Comparison on evaluation noisy dataset in terms of average (%) EER using multi-condition training*

Eval. Condition	CQCC + GMM [12]			CNN + NAT			CNN + MASK			CNN + NAT + RNN			CNN + MASK + RNN		
	Kn.	Un.	Avg.	Kn.	Un.	Avg.	Kn.	Un.	Avg.	Kn.	Un.	Avg.	Kn.	Un.	Avg.
clean	0.10	**0.90**	0.50	0.14	2.03	1.09	0.12	1.74	0.93	0.04	1.13	0.59	**0.03**	**0.90**	**0.47**
white_snr_20	46.8	44.6	45.7	1.7	4.3	3.0	1.4	3.9	2.7	1.1	2.9	2.0	**0.8**	**2.5**	**1.7**
white_snr_10	48.9	48.1	48.5	3.2	5.1	4.4	2.7	4.6	3.7	**2.1**	3.7	**2.9**	2.3	**3.4**	**2.9**
white_snr_0	49.3	48.9	49.1	7.9	10.0	9.0	7.2	9.3	8.3	6.9	9.1	8.0	**5.9**	**8.6**	**7.3**
babble_snr_20	18.2	18.3	18.3	3.1	4.6	3.9	2.9	4.1	3.5	2.5	**3.7**	**3.1**	**2.3**	3.9	**3.1**
babble_snr_10	33.9	33.6	33.8	5.7	6.7	6.2	5.2	5.9	5.6	4.1	4.8	4.5	**3.7**	**4.5**	**4.1**
babble_snr_0	44.6	44.0	44.3	12.9	14.7	13.8	12.1	13.6	12.9	10.1	11.7	10.9	**9.5**	**10.6**	**10.1**
street_snr_20	22.7	22.3	22.5	3.9	5.1	4.5	2.7	4.2	3.5	2.1	3.5	2.8	**1.9**	**3.1**	**2.5**
street_snr_10	37.5	36.3	36.9	6.1	7.5	6.8	5.1	6.7	5.9	4.6	5.7	5.2	**4.1**	**5.4**	**4.8**
street_snr_0	46.1	45.4	45.8	11.1	13.7	12.4	10.1	12.4	11.3	9.1	10.8	10.0	**8.7**	**9.9**	**9.3**
reverberation_0.3	8.4	9.3	8.9	1.3	2.1	1.7	1.5	2.2	1.9	1.2	**1.8**	1.5	**1.1**	1.9	**1.5**
reverberation_0.6	10.6	7.8	9.2	1.6	2.0	1.8	1.5	2.1	1.8	**1.4**	1.7	**1.6**	1.6	**1.5**	**1.6**
reverberation_0.9	7.6	6.9	7.3	1.5	1.9	1.7	1.4	1.7	1.6	1.2	**1.5**	**1.4**	**1.1**	1.6	**1.4**
Avg. Seen Noise	31.2	30.5	30.8	5.0	6.5	5.8	4.5	5.9	5.2	3.9	5.1	4.5	**3.6**	**4.7**	**4.2**
cafe_snr_20	30.7	30.1	30.4	2.9	5.3	4.1	2.7	5.4	4.1	1.9	4.7	3.3	**1.8**	**4.5**	**3.2**
cafe_snr_10	42.1	41.3	41.7	5.6	8.1	6.9	5.3	7.8	6.6	4.7	6.1	5.4	**4.5**	**5.7**	**5.1**
cafe_snr_0	49.8	47.1	47.3	13.5	20.0	16.8	12.4	18.7	15.6	10.7	15.4	13.1	**10.1**	**14.3**	**12.2**
volvo_snr_20	0.9	**2.7**	**1.8**	1.0	3.7	2.4	0.9	3.4	2.2	**0.7**	3.1	1.9	0.8	3.0	1.9
volvo_snr_10	4.3	5.6	4.9	2.4	4.9	3.7	2.1	4.5	3.3	1.7	3.6	2.7	**1.5**	**3.4**	**2.5**
volvo_snr_0	13.0	13.0	13.0	3.7	5.0	4.4	3.4	4.7	4.1	3.1	3.7	3.4	**2.7**	**3.5**	**3.1**
Avg. Unseen Noise	23.1	23.3	23.2	4.9	7.8	6.4	4.5	7.4	6.0	3.8	6.1	5.0	**3.6**	**5.7**	**4.7**

mance in both seen and unseen noisy conditions with respect the two reference techniques (CQCC+GMM, CNN+NAT). It must be taken into account that although CNN+NAT is the best isolated deep feature extraction proposed in [12], this reference also proposes a combination of DNN, CNN, RNN and NAT for frame-level feature extraction that outperforms CNN+NAT. However, this combination is not directly comparable with our CNN+MASK+RNN since it is a fusion of techniques unlike our proposal. Despite this, it is worth mentioning that this combination can only outperform our best proposal in the case of seen noises but not in the case of the unseen ones. Finally, it is worth noticing that, although the CQCC+GMM system has been proved to get the best state-of-the-art results using the clean ASVspoof 2015 database, our FBANK+CNN+MASK+RNN+LDA system gets a better performance in the clean evaluation dataset when using multi-condition training.

5. Conclusions

This paper has proposed a novel technique for the extraction of deep identity features for an efficient detection of spoofing attacks in clean and noisy environments. In our system, a CNN+RNN hybrid architecture is employed to embed the utterances as a single vector, providing information about whether the utterance is genuine or spoofed. Furthermore, to increase the noise robustness of our anti-spoofing detector, a soft missing-data mask technique has been proposed.

Our system has been evaluated on the ASVspoof 2015 clean corpus and on a distorted version of the same corpus, including both additive noise and reverberation. The experimental results have shown that our best proposal outperforms the CQCC+GMM system (baseline of the ASVspoof 2017 challenge [24]) and the best isolated deep feature extractor proposed in [12] (CNN+NAT) for both seen and unseen distorted conditions, respectively.

In the future, we plan to integrate other noise mask estimation techniques in the deep feature extraction procedure in order to obtain further improvements in noisy conditions. Also, we will investigate the incorporation of phase-based features that could complete the signal information lost by the FBANK features.

6. Acknowledgements

This work has been supported by the Spanish MINECO/FEDER Project TEC2016-80141-P and the Spanish Ministry of Education through the National Program FPU (grant reference FPU16/05490). We also acknowledge the support of NVIDIA Corporation with the donation of a Titan X GPU. Moreover, we would like to thank Mr. Xiaohai Tian from Nanyang Technological University, Singapore for sharing the noisy version of ASVspoof 2015 database.

7. References

[1] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, no. 1-2, pp. 91–108, 1995.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 130–153, 2011.

[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, no. 4, pp. 788-798, 2015.

[5] Z. Wu et al., "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 768–783, 2016.

[6] M. P. Cooke, P. Green, L. Josifovski, and A. Vizinho, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, vol. 34, pp. 267–285, 2001.

[7] J. P. Barker, L. Josifovski, M. P. Cooke, and P. D. Green, "Soft decisions in missing data techniques for robust automatic speech recognition," in *Proc. ICSLP*, 2000, pp. 373–376.

[8] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, and D. Erro, "A cross-vocoder study of speaker independent synthetic speech detection using phase information," in *Proc. InterSpeech*, 2014, pp. 1663–1667.

[9] 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 *Proc. InterSpeech*, 2015, pp. 2037–2041.

[10] X. Tian, Z.Wu, X. Xiao, E. S. Chng, and H. Li, "An investigation of spoofing speech detection under additive noise and reverberant condition," in *Proc. InterSpeech*, 2016, pp. 1715–1719.

[11] C. Hanilci, T. Kinnunen, M. Sahidullah, and A. Sizov, "Spoofing detection goes noisy: An analysis of synthetic speech detection in the presence of additive noise," in *Speech Communication*, vol. 85, pp. 83–97, 2016.

[12] Y. Qian, N. Chen, H. Dinkel, and Z. Wu, "Deep Feature Engineering for Noise Robust Spoofing Detection," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 10, pp. 1942–1955, 2017.

[13] N. Brümmer and E. deVilliers, "The BOSARIS toolkit: Theory, algorithms and code for surviving the new DCF," in *NIST SRE11 Speaker Recognition Workshop*, Atlanta, Georgia, USA, Dec. 2011, pp. 1–23.

[14] Martín Abadi, Ashish Agarwal, Paul Barham, et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[15] Y. Qian, N. Chen, and K. Yu, "Deep Features for automatic spoofing detection," *Speech Communication*, vol. 85, pp. 43–52, 2016.

[16] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," *Proc. Odyssey*, 2016, pp. 249–252.

[17] 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," *Proc. Interspeech*, 2015, pp. 2062–2066.

[18] Young, S., et al. The HTK Book, Version 3.4. Cambridge University Engineering Department (2006).

[19] Kyunghyun Cho, et al., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *Proc. Empirical Methods in Natural Language Processing*, 2014, pp. 1724–1734.

[20] Chunlei Zhang, Chengzhu Yu, and John H. L. Hansen, "An investigation of Deep-Learning Frameworks for Speaker Verification Antispoofing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, pp. 684–694, 2017.

[21] Q. Jin and A. Waibel, "Application of LDA to speaker recognition," *Proc. Interspeech*, 2010, pp. 250–253.

[22] M. McLaren and D. Van Leeuwen, "Source-normalised-and-weighted LDA for robust speaker recognition using i-vectors," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 5456–5459.

[23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6890*, 2014.

[24] Tommi Kinnunen, Md Sahidullah, Héctor Delgado, et al. "The ASVspoof 2017 Challenge: Asessing the Limits of Replay Spoofing Attack Detection," *Proc. Interspeech*, 2017, pp. 2–6.