INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Anti-Spoofing Speaker Verification System with Multi-Feature Integration and Multi-Task Learning

Rongjin Li[1], *Miao Zhao*[2], *Zheng Li*[1], *Lin Li*[1], *Qingyang Hong*[2]

[1]School of Electronic Science and Engineering, Xiamen University, China
[2]School of Information Science and Engineering, Xiamen University, China

lilin@xmu.edu.cn, qyhong@xmu.edu.cn

Abstract

Speaker anti-spoofing is crucial to prevent security breaches when the speaker verification systems encounter the spoofed attacks from the advanced speech synthesis algorithms and high fidelity replay devices. In this paper, we propose a framework based on multiple features integration and multi-task learning (MFMT) for improving anti-spoofing performance. It is important to integrate the complementary information of multiple spectral features within the network, such as MFCC, C-QCC, Fbank, etc., as often a single kind of feature is not enough to grasp the global spoofing cues and it generalizes poorly. Furthermore, we propose a helpful butterfly unit (BU) for multi-task learning to propagate the shared representations between the binary decision task and the other auxiliary task. The BU can obtain task representations of other branch during forward propagation and prevent the gradient from assimilating the branch during back propagation. Our proposed system yielded an EER of 9.01% on ASVspoof 2017, while the best single system and the average scores fusion obtained the evaluation EER of 2.39% and 0.96% on ASVspoof 2019 PA, respectively.

Index Terms: multi-feature integration, multi-task learning, stitching layer, butterfly unit, anti-spoofing, speaker verification

1. Introduction

Speaker verification is one of the most important biometric authentication systems since attackers present various spoofed strategies in order to pose as the genuine users. These spoofed strategies include voice conversion (VC) [1], speech synthesis (SS) [2], and audio replay, etc. The replay attack poses the greatest threat to automatic speaker verification (ASV) systems since it can copy authenticated users' voices with high fidelity devices [3]. To counteract these spoofed attacks, countermeasures (CM) have been developed to detect spoofed attacks before speaker verification.

To prevent ASV spoofing, high time-frequency resolution features have become popular solutions, as they identify crucial voiceprint cues using binary classifiers [4, 5, 6]. Cochlear filter cepstral coefficients (CFCC) and the change in instantaneous frequency (CFCCIF) were proposed for training two simple Gaussian mixture model (GMM) classifiers to detect genuine and spoofed speech, respectively [5]. Constant Q cepstral coefficients (CQCCs) [7], which use the constant Q transform (CQT) instead of the short-time Fourier transform (STFT) to process speech signals, perform better than common Mel-frequency cepstral coefficients (MFCCs). If we increase the number of frames and the number of bins per frame simultaneously [8], the Fourier-based features also have great time-frequency analysis potential. However, we realize that utilizing only one kind of acoustic feature is insufficient for capturing global spoofed cues when facing unknown spoofed speech. If one classifier could integrate the regular STFT, the geometrical CQT, or other scale features, such a classifier would mine their complementary effects and learn discriminative information from various feature engineering.

Later in the deep learning era, convolutional neural networks (CNN) have performed much better than using GMM directly [9, 10, 11, 12]. For example, Light CNN (LCNN) with a max-feature-map (MFM) activation function [13] extracts quite high-level embeddings from the log power spectrogram, which is obtained via CQT or STFT [9, 12]. When binary classes are well-separable in a high-level feature space, it is suitable to use simple two-class GMMs to obtain log-likelihood ratios (LLR). However, when encountering different algorithms or devices, a network with binary supervision only generates a binary decision without a reasonable attack-specific explanation. Different speech synthesis algorithms may sound unnatural in different ways and to different degrees. When compared with genuine audio, replayed audio may contain two extra noise sources, including replay devices and a second recording environment [14]. These noises are not identical to the internal noises of genuine speech. Hence, multi-task learning can be introduced to learn this auxiliary information from different spoofed attacks [15, 16]. However, the crucial problem is to design a multi-task learning system that can detect shared representations between tasks, while avoiding deviating from the primary task [17].

To address these issues, we propose a novel framework based on multiple feature integration and multi-task learning (MFMT) to distinguish genuine speech from spoofed speech. For the input, multiple features are fed into a neural network simultaneously and concatenated in a stitching layer, which integrates different feature engineering methods within the network. The stitching layer is able to learn a great deal of complementary representations. To improve the robustness of multi-task learning, we propose a new butterfly unit (BU) to propagate other task-specific representations during forward propagation and to prevent the gradient from adjusting to other tasks during backward propagation. This BU is similar to the butterfly operation in the fast Fourier transform, which requires forward operations without backward "paths". Considering that our primary task is a binary decision, the BU produces auxiliary representations of the main task and clips the gradient to avoid assimilation.

The remainder of this paper is organized as follows. Section 2 describes details of our framework for multi-feature and multi-task learning with a butterfly unit (BU-MFMT). Section 3 introduces the datasets in ASVspoof 2017 and 2019 along with the common baseline systems, and our experimental setup and results are presented in section 4. Finally, the conclusion is given in section 5.

http://dx.doi.org/10.21437/Interspeech.2019-1698

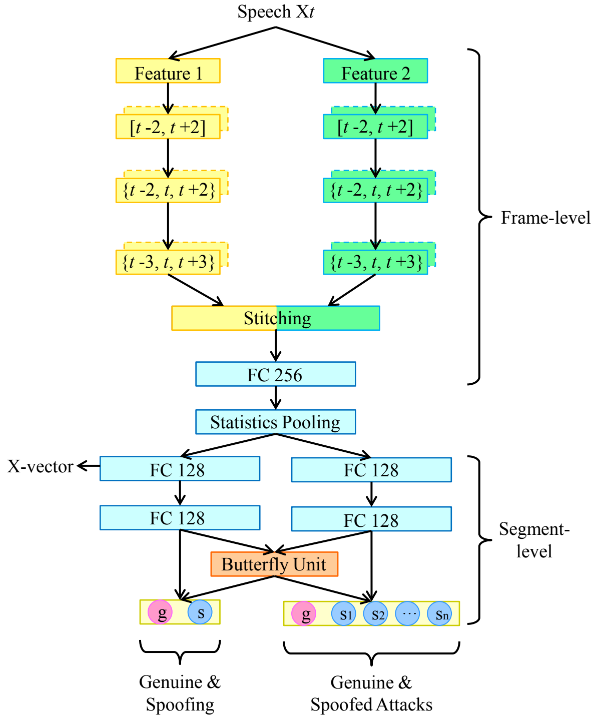

Figure 1: *Butterfly Unit Multi-Feature Multi-Task Framework.*

2. Proposed Methods for Anti-Spoofing

In this section, we present our BU-MFMT framework for anti-spoofing speaker verification in Figure 1. Our basic method is to extract the x-vector [18, 19] from the time delay neural network (TDNN) [20], and then we classify the x-vectors further and make the final decision. Since the main task is a binary decision, the network is modified to be lighter. The other layers are 256-dimensional, except for the first three TDNN layers and the two segment-level fully-connected layers (FCs), which are both 128-dimensional.

2.1. Stitching Layer for Multi-Feature Integration

In feature engineering, different scale features contain different information. In Figure 2, replayed speech has different spectral expressions on the STFT and CQT, respectively. The silent regions and the edges of voiced regions show regular spectral bands in the STFT, whereas the lower frequencies in the C-QT have higher frequency resolution, and the high frequencies have higher temporal resolution [7]. STFT and CQT demonstrate spoofed cues with different scales; consequently, the goal of our network is to integrate this complementary information simultaneously in order to improve the upper boundary.

In Figure 1, the stitching layer integrates complementary information based on multiple features from different scales. Multiple features from the same period of the same speech sample are simultaneously fed into corresponding input layers in the network, and then their representations x are stitched together in the stitching layer:

$$x \leftarrow Append(x_{feat1}, x_{feat2}) \tag{1}$$

The fully-connected stitching layer is a high-level frame-level representation. The multi-feature network demonstrates great robustness and performs even better than the average score fusion of multiple independent single-feature systems.

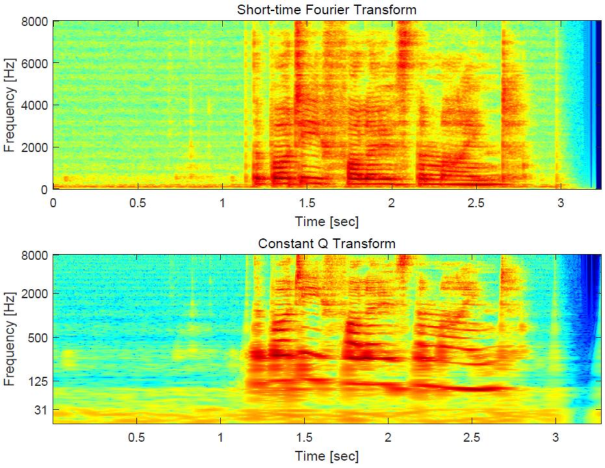

Figure 2: *Spectrograms computed with the STFT (top) and the CQT (bottom). The spoofed speech sample includes a female voice saying the words, "the body follows the mind," from ASVspoof 2019 physical access training.*

2.2. Multi-Task Learning of Spoofed Attacks

Spoofed cues are critical for anti-spoofing speaker verification, especially for replayed spoofed attacks. A genuine audio sample includes the noise from the ASV recording devices $n(t)$ and the first recording environment $e(t)$, namely internal noises [14, 15]. Although internal noise is inherent in every speech sample, spoofed speech contains extra noise signals, including noises from the microphone device $mic(t)$, the replay devices $R(t)$ and the second recording environment $E(t)$. Extra noise signals are impulse responses, and they are convolved in genuine speech [14]. Genuine and spoofed samples can be expressed by the following equations:

$$y_{genuine}(t) = x(t) * e(t) * n(t) \tag{2}$$

$$y_{spoof}(t) = x(t) * e(t) * mic(t) * R(t) * E(t) * n(t), \tag{3}$$

where $x(t)$ is the voice of the real speaker. $y_{genuine}(t)$ and $y_{spoof}(t)$ represent the genuine signal and the spoofed signal, respectively; these values flow into the speaker verification system.

If a hacker uses a high fidelity recording device to record a real voice at a very close range (10-50cm), the impacts of the microphone device and the first recording environment are neglected and Equation 3 is simplified as:

$$y_{spoof}(t) = x(t) * R(t) * E(t) * n(t) \tag{4}$$

Compared with Equation 2, the $R(t)$ and $E(t)$ noise signals are the most critical spoofed cues and loopholes. Hence, we propose training the network for binary decision making and replayed noise source recognition simultaneously. Since the two tasks have a synergistic relationship, the network is capable of learning extra attack-specific information in order to optimize the binary decision. Furthermore, we suggest adopting two branches to learn the two tasks in Figure 1. If the shared part includes the segment-level fully-connected layers, the x-vectors will be affected by different tasks in each iteration and become less robust. The above architecture is also suitable for VC and SS spoofed attacks.

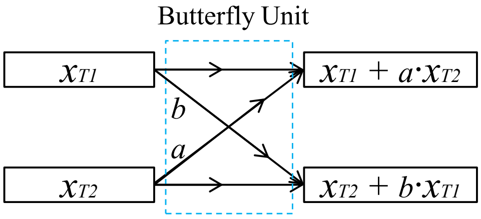

Figure 3: *The Butterfly Unit Operations.*

2.3. Butterfly Unit for Multi-Task Learning

In multi-task learning, if the binary loss function learns information from the shared representations, the x-vector in this branch will have task-specific representations. Furthermore, the two tasks have the same genuine node in Figure 1, and each spoofed attack is an affiliation of the spoofed node. So, we propose a novel butterfly unit at the segment-level in Figure 3, which propagates the representation from another task to the current task. Considering the robustness of this x-vector, we clip the gradient from current task at the BU during backward propagation. So, the softmax loss L_{T_1} for the binary decision, the loss L_{T_2} for spoofing recognition, and their partial derivative are computed as:

$$\begin{cases} \hat{x}_{T_1} = x_{T_1} + a \cdot x_{T_2} \\ \hat{x}_{T_2} = x_{T_2} + b \cdot x_{T_1} \end{cases} \tag{5}$$

$$\begin{cases} \frac{\partial L_{T_1}}{\partial \hat{x}_{T_1}} = \frac{\partial L_{T_1}}{\partial x_{T_1}} + \beta \cdot a \cdot \frac{\partial L_{T_1}}{\partial x_{T_2}} \\ \\ \frac{\partial L_{T_2}}{\partial \hat{x}_{T_2}} = \frac{\partial L_{T_2}}{\partial x_{T_2}} + \beta \cdot b \cdot \frac{\partial L_{T_2}}{\partial x_{T_1}} \end{cases} \tag{6}$$

where β is a scale for the clipped gradient, and a and b are the scale parameters for forward propagation. When the network learns the shared representation between two tasks, the gradient descent bypasses the last and penultimate layers of another branch in each iteration, which does not affect the x-vector's robustness. In our work, we set $\beta = 0$ and $a = b = 0.925$.

3. Experiments

3.1. Datasets

All experiments in this paper were performed on ASVspoof 2017 version 2.0 [21, 22] and ASVspoof 2019 [23] datasets. Detailed descriptions of these datasets are found in Table 1. Both of these datasets are separately partitioned into three subsets: a training set, a development set, and an evaluation set. ASVspoof 2017 only contains replayed attacks. In ASVspoof 2019, there are two data conditions, and the logical access (LA) includes SS and VC, while the physical access (PA) is comprised of replayed attacks. In our experiments, we only used the training set without any data augmentation to compare with our proposed systems on the evaluation set.

3.2. Experimental Setup

Table 1: *The ASVspoof 2017 and ASVspoof 2019 dataset distributions for the training, development, and evaluation sets.*

Dataset	Subset	Speaker	Utterance	
			Bona fide	Spoof
ASVspoof 2017	Train	10	1,507	1,507
	Dev	8	760	950
	Eval	24	1,298	12,008
ASVspoof 2019 Logical access	Train	20	2,580	22,800
	Dev	20	2,548	22,296
ASVspoof 2019 Physical access	Train	20	5,400	48,600
	Dev	20	5,400	24,300

All experiments were conducted with the Kaldi toolkit [24]. We used four kinds of features, including MFCC, CQCC, Filterbanks (Fbank), and STFT spectrogram. As for MFCC and F-bank, a window with a length of 50ms and a frame shift of 4ms was applied. These acoustic features were extracted based on 160 filters before conducting the discrete cosine transform, and the dimension of MFCC is 40. The CQCC consists of the first 40 static coefficients, including the 0-th order cepstral coefficient. The STFT spectrograms were extracted with the following parameters: a window with a length of 25ms and a frame shift of 4ms, and 256 points were obtained for each frame. No voice activity detection (VAD) is performed on the data since silent segments and the edge of voiced regions are more likely to contain spoofed cues. We applied cepstral mean normalization (CMN) to all features and length normalization to all x-vectors.

In terms of performance assessment, all scores are not processed by score normalization. The equal error rate (EER) is computed for both ASVspoof 2017 and 2019 based on the Bosaris toolkit [25]. In addition, for the purpose of improving the co-operation system in spoofed CM and ASV, the minimum normalized tandem detection cost function (min t-DCF) is adopted as the primary metric in ASVspoof 2019 [26].

As for the baseline, we compared the proposed systems with the standard baseline system, i.e. the CQCC-GMM. With 30-dimensional CQCC and its delta and delta-delta derivatives, two 512-component GMM models are trained for genuine and spoofed speech, respectively. The log-likelihood ratio of each tested speech sample from the genuine model and spoofed model is taken as the final score during evaluation [22, 23].

3.3. Back-End Optimizations

For different data sets, we adopt two back-end strategies, including logistic regression (LR) and maximum mutual information Gaussian mixture models (MMI-GMM) [27] to optimize the x-vectors. On ASVspoof 2017 and ASVspoof 2019 PA, we chose the LR to classify the x-vectors after length normalization and linear discriminant analysis (LDA). The x-vector dimension here is not reduced by LDA. Whereas for the LA, we respectively trained two 64-component MMI-GMMs with genuine speech and spoofed speech to compute the LLR.

4. Results and Discussion

Since BU-MFMT is able to derive many sub-variants, such as multi-feature single-task (MFST) and multi-feature multi-task without the butterfly unit (MFMT), we present the results of them separately. The results of our proposed MFST on ASVspoof 2017 and 2019 are shown in Table 2. In order to verify the effects of multiple features, we present the results of the average scores fusion. Since the dimensions of the spectrogram are too large, we did not consider it as a candidate for the multi-feature single task. As for the labels of spoofed attacks in multi-task learning, we chose the second recording environment $E(t)$ from ASVspoof 2017 for a total of four classes. Meanwhile, we considered both $E(t)$ and $R(t)$ from ASVspoof 2019 PA for a total of nine classes, and we selected six SS and VC classes

Table 2: *The Results of Baseline, TDNN, Average Score Fusion, and Multi-Feature Single-Task from ASVspoof 2017 and 2019 datasets.*

System	Description	ASVspoof 2017	ASVspoof 2019 LA		ASVspoof 2019 PA	
		Evaluation	Development		Development	
		EER (%)	EER (%)	t-DCF	EER (%)	t-DCF
Baseline	S0.CQCC	30.79	0.43	0.0123	9.87	0.1953
TDNN	S1.CQCC	16.02	0.86	0.0360	6.43	0.1802
	S2.MFCC	13.94	0.31	0.0092	2.26	0.0558
	S3.Fbank	**12.63**	0.01	0.0002	**1.47**	**0.0344**
	S4.Spectrogram	15.72	**0.00**	**0.0000**	3.74	0.1002
Score Fusion	S1 + S2	11.94	0.43	0.0132	2.28	0.0578
	S1 + S3	11.40	0.04	0.0014	1.81	0.0466
	S2 + S3	9.94	0.04	0.0011	1.28	0.0293
MFST	S5.CQCC+MFCC	10.25	0.16	0.0043	1.61	0.0433
	S6.CQCC+Fbank	10.79	0.04	0.0007	1.50	0.0383
	S7.MFCC+Fbank	**9.78**	**0.00**	**0.0001**	**1.28**	**0.0330**

Table 3: *The Results of Multi-Feature Multi-Task, Butterfly Unit, and Average Score Fusion from ASVspoof 2017 and 2019 datasets.*

System	Description	ASVspoof 2017	ASVspoof 2019 LA		ASVspoof2019 PA	
		Evaluation	Development		Development	
		EER (%)	EER (%)	t-DCF	EER (%)	t-DCF
S8.MFMT	MFCC+Fbank	9.40	0.00	0.0000	2.07	0.0571
S9.BU-MFMT		9.01	0.00	0.0000	1.55	0.0455
Score Fusion	S1+S4+S5+S6+S7+S8+S9	**7.94**	**0.00**	**0.0000**	**0.67**	**0.0148**

from ASVspoof 2019 LA.

4.1. The Results of Multi-Feature Single-Task

As seen in Table 2, due to limited data and the somewhat uncontrolled recording conditions, the effects of feature engineering are not as ideal in the ASVspoof 2017. Although the improvement from feature engineering is slight, our proposed architectures perform better. The MFSTs perform better than all single-feature systems, and the average score fusions of those systems, especially the integration of MFCC and Fbank. Our system obtains a 1.6% absolute improvement over the average score fusion.

In contrast, feature engineering generates significant effects on the ASVspoof 2019 LA dataset. Compared to other features in the TDNN, the spectrogram features exhibit excellent results, i.e., EER = 0.00% and min t-DCF = 0.0000. It is likely due to the detailed spectral information contained in the STFT spectrograms, which is significant for discriminating between genuine and SS/VC speech. Furthermore, we find that the presence of the Fbank feature in all datasets is highly discriminative and contains more knowledge of the spoofed cues to help the binary decision. Compared to average score fusion, all features are highly complementary and benefit from the MFSTs, especially the integration of CQT and STFT. Finally, since the two related robust single-feature systems, MFCC and Fbank, and their MFSTs achieve satisfactory results, the following MFMT and BU-MFMT will be tested on the MFCC and Fbank.

4.2. Results of Butterfly Unit Multi-Feature Multi-Task

In Table 3, for ASVspoof 2017, the BU-MFMT significantly outperforms the MFMT and MFST, which are based on the integration of MFCC and Fbank. Whereas for ASVspoof 2019, the MFMTs are somewhat worse, but the BU-MFMTs propagate shared representations well, which verifies that the BU improve multi-task learning robustness. For ASVspoof 2019 PA, the BU-MFMT yields a 25.12% relative improvement over the MFMT. This performance improvement in the evaluation set also verifies the effectiveness of the BU-MFMT for spoofed attack recognition.

Table 4: *The Results of ASVspoof 2019 Evaluation Set.*

System	ASVspoof 2019 LA		ASVspoof 2019 PA	
	EER (%)	t-DCF	EER (%)	t-DCF
Primary	7.63	0.2129	0.96	0.0266

4.3. Results of ASVspoof 2019 Evaluation Set

We obtain the best fusion results for the development set based on only seven subsystems, as shown in Table 3. We took the average score fusions as the primary system and the results of the evaluation are in Table 4. However, the primary for LA presents degradation, which may be due to the over-fitting problem.

5. Conclusions

In this paper, we propose a novel multi-feature multi-task architecture for anti-spoofing speaker verification systems. The two key concepts include integrating feature engineering within the network and better propagating shared representations in multi-task learning. The methods we propose achieve satisfactory results and provide inspiration to this community. In future work, we will study how to improve feature engineering and stitching layers for anti-spoofing.

6. Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No.61876160).

7. References

[1] Z. Wu and H. Li, "Voice Conversion versus Speaker Verification: an Overview," *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014.

[2] 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, 2012.

[3] 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, 2015.

[4] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A Comparison of Features for Synthetic Speech Detection," *ISCA (the International Speech Communication Association)*, 2015.

[5] 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 *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[6] G. Suthokumar, V. Sethu, C. Wijenayake, and E. Ambikairajah, "Modulation Dynamic Features for the Detection of Replay Attacks," in *Proc. Interspeech*, 2018, pp. 691–695.

[7] M. Todisco, H. Delgado, and N. Evans, "A New Feature for Automatic Speaker Verification Anti-Spoofing: Constant Q Cepstral Coefficients," in *Speaker Odyssey Workshop, Bilbao, Spain*, vol. 25, 2016, pp. 249–252.

[8] Z. Chen, W. Zhang, Z. Xie, X. Xu, and D. Chen, "Recurrent Neural Networks for Automatic Replay Spoofing Attack Detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2052–2056.

[9] G. Lavrentyeva, S. Novoselov, E. Malykh, A. Kozlov, O. Kudashev, and V. Shchemelinin, "Audio Replay Attack Detection with Deep Learning Frameworks," *Interspeech*, pp. 82–86, 2017.

[10] Z. Chen, Z. Xie, W. Zhang, and X. Xu, "ResNet and Model Fusion for Automatic Spoofing Detection," *Interspeech*, pp. 102–106, 2017.

[11] F. Tom, M. Jain, and P. Dey, "End-To-End Audio Replay Attack Detection Using Deep Convolutional Networks with Attention," *Interspeech*, pp. 681–685, 2018.

[12] K. Sriskandaraja, V. Sethu, and E. Ambikairajah, "Deep Siamese Architecture Based Replay Detection for Secure Voice Biometric," *Interspeech*, pp. 671–675, 2018.

[13] X. Wu, R. He, Z. Sun, and T. Tan, "A Light CNN for Deep Face Representation with Noisy Labels," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.

[14] F. Alegre, A. Janicki, and N. Evans, "Re-Assessing the Threat of Replay Spoofing Attacks against Automatic Speaker Verification," in *2014 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2014, pp. 1–6.

[15] H.-J. Shim, J.-W. Jung, H.-S. Heo, S.-H. Yoon, and H.-J. Yu, "Replay Spoofing Detection System for Automatic Speaker Verification using Multi-task Learning of Noise Classes," in *2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*. IEEE, 2018, pp. 172–176.

[16] Y. Liu, L. He, J. Liu, and M. T. Johnson, "Speaker Embedding Extraction with Phonetic Information," *arXiv preprint arXiv:1804.04862*, 2018.

[17] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-Stitch Networks for Multi-Task Learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3994–4003.

[18] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep Neural Network Embeddings for Text-Independent Speaker Verification," in *Interspeech*, 2017, pp. 999–1003.

[19] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN Embeddings for Speaker Recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[20] V. Peddinti, D. Povey, and S. Khudanpur, "A Time Delay Neural Network Architecture for Efficient Modeling of Long Temporal Contexts," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[21] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. A. Lee, "The ASVspoof 2017 Challenge: Assessing the Limits of Replay Spoofing Attack Detection," in *ISCA (the International Speech Communication Association)*, 2017.

[22] H. Delgado, M. Todisco, M. Sahidullah, N. Evans, T. Kinnunen, K. Lee, and J. Yamagishi, "ASVspoof 2017 Version 2.0: Meta-Data Analysis and Baseline Enhancements," in *Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018.

[23] A. consortium, "ASVspoof 2019: Automatic Speaker Verification Spoofing and Countermeasures Challenge Evaluation Plan," http://www.asvspoof.org/asvspoof2019/asvspoof2019_evaluation_plan.pdf, accessed January 15, 2019.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi Speech Recognition Toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[25] N. Brümmer and E. De Villiers, "The Bosaris Toolkit: Theory, Algorithms and Code for Surviving the New DCF," *arXiv preprint arXiv:1304.2865*, 2013.

[26] T. Kinnunen, K. A. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "T-DCF: a Detection Cost Function for the Tandem Assessment of Spoofing Countermeasures and Automatic Speaker Verification," *arXiv preprint arXiv:1804.09618*, 2018.

[27] A. McCree, "Multiclass Discriminative Training of I-vector Language Recognition," in *Proc. of Speaker Odyssey*, 2014, pp. 166–171.