Norm-constrained Score-level Ensemble for Spoofing Aware Speaker Verification

Peng Zhang, Peng Hu*, Xueliang Zhang*

Elevoc Technology Co., Ltd, Shenzhen, China
{peng.zhang,peng.hu,xuliang.zhang}@elevoc.com

Abstract

In this paper, we present the Elevoc systems submitted to the Spoofing Aware Speaker Verification Challenge (SASVC) 2022. Our submissions focus on bridge the gap between the automatic speaker verification (ASV) and countermeasure (CM) systems. We investigate a general and efficient norm-constrained score-level ensemble method which jointly processes the scores extracted from ASV and CM subsystems, improving robustness to both zero-effect imposters and spoofing attacks. Furthermore, we explore that the ensemble system can provide better performance when both ASV and CM subsystems are optimized. Experimental results show that our primary system yields 0.45% SV-EER, 0.26% SPF-EER and 0.37% SASV-EER, and obtains more than 96.08%, 66.67% and 94.19% relative improvements over the best performing baseline systems on the SASVC 2022 evaluation set. All of our code and pre-trained models weights are publicly available and reproducible [1].

Index Terms: Speaker verification, anti-spoofing, ensemble of speaker verification and anti-spoofing, SASVC 2022.

1. Introduction

Biometric authentication [1] aim to authenticate the identity claimed by a given individual based on the samples measured from biological processes and/or organs, which is becoming popular in scenario of protecting the security of computers, smart devices, and networks, such as fingerprint, voiceprint and face recognition. Even though current automatic speaker verification (ASV) systems have been robust to noisy environments [2, 3, 4, 5], their vulnerability to malicious spoofing attacks remains a serious concern nowadays [6, 7]. Even state-of-the-art ASV systems can be vulnerable to spoofing attacks [8] generated using speech synthesis / text-to-speech (TTS), voice conversion (VC) or replay attacks. Some such attacks can degrade ASV reliability considerably [9]. Therefore, anti-spoofing should be considered carefully before putting ASV into practical usage.

In recent years, countermeasure (CM) systems have hence been developed which classifies given utterances as spoofed or not spoofed where many deep neural network (DNN) based systems have achieved promising results [10, 11, 12, 13]. While ASV and CM systems have been well studied separately so far, the integration of both systems still requires further research. Todisco et al. [14] propose a separate modeling of two Gaussian back-end systems with a unified threshold for both ASV and CM tasks. Two joint ASV and CM systems are studied in the i-vector [15, 16] and x-vector space [17, 18, 19]. Moreover, Shim et al. [20] propose an end-to-end framework that jointly optimize ASV, CM and the SASV task.

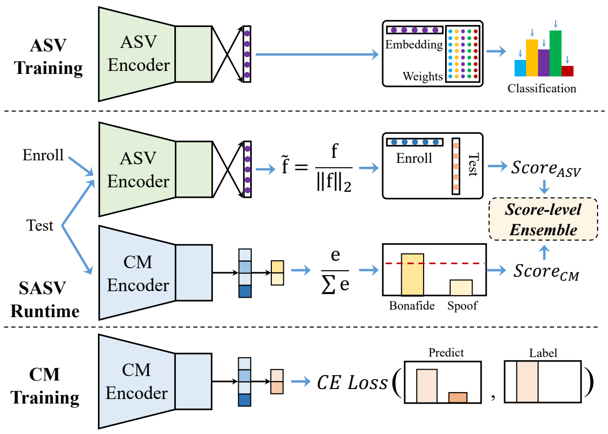

Figure 1: *Overall framework of our proposed score-level ensemble system.*

In this paper, we describe the Elevoc team's submissions developed for the Spoof Aware Speaker Verification Challenge (SASVC) 2022. The main goal of SASVC 2022 is to further improve robustness to both zero-effort impostor access attempts and spoofing attacks by providing a framework to support the optimization of CM and ASV systems. The challenge is to evaluate SASVC using the ASVspoof 2019 LA dataset. Whilst, in the logical access (LA) scenario, the spoof attacks are directly injected into the ASV system, normally generated using TTS and VC technologies.

As illustrated in Figure 1, we propose a spoofing-aware framework for the SASV task. Since the training objectives of the ASV and CM tasks are different, speaker embeddings for the ASV task requires robustness to device and channel different; meanwhile, representation for the CM task uses such information. Based on this, we firstly training the ASV and CM subsystems independently. When in the SASV runtime, the ASV subsystem scores the input enrollment and test utterances, and the CM subsystem distinguishes whether the test utterance is a spoof or a bonafide speech. Finally, ensemble ASV and CM systems based on a score-level ensemble approach to better discriminate between bonafide target speech and zero-effect imposters or spoofing attacks. In our work, we investigate a general and efficient norm-constrained score-level ensemble method that substantially improves the performance of SASV task, which bridge the gap between the ASV and CM systems. Moreover, by exploring different structural feature encoders for ASV and CM subsystems, it is further verified that ensemble SASV system can be delivered better perfor-

*The first two authors contributed equally to this work
[1] https://github.com/WebPrague/SASV2022_DoubleRoc

mance when both ASV and CM subsystems are optimized. Experimental results show that our primary system yields 0.45% SV-EER, 0.26% SPF-EER and 0.37% SASV-EER, and obtains more than 96.08%, 66.67% and 94.19% relative improvements over the best performing baseline systems on the SASVC 2022 evaluation set.

The remainder of this paper is organized as follows: Section 2 introduces the methodology in our submissions. Then, in Section 3, we present the experimental setup. After that, Section 4 evaluates the ensemble SASV systems. Finally, we conclude this paper in Section 5.

2. Methodology

Figure 1 illustrates the overall framework of the Elevoc's submission, which is mainly composed of ASV and CM systems and score-level ensemble modules. In this section, we first inroduce the ASV and CM systems with different topologies explored. Then, a detailed description of different score-level ensemble methods and analyses are provided.

2.1. Standalone Automatic Speaker Verification (ASV) systems

The goal of an ASV system is to determine whether a test utterance is produced by the claimed speaker or not. The conventional ASV framework can be decomposed into a frame-level feature extractor, a pooling layer and utterance-level feature extractor [2]. For our submissions, all of ASV systems are built upon the foundation of our previous work in Short-duration Speaker Verification Challenge (SdSVC) 2021 [21]. By fixing the attentive statistic pooling layer [22] and utterance-level representation layers, we choose three frame-level feature extractors that perform well at SdSVC 2021.

SE-ResNet-34. We use ResNet-34 [23] with Squeeze-Excitation (SE) module [24] for frame-level feature extraction. The SE block can adaptively re-calibrate channel-wise feature responses by explicitly modeling inter-dependencies among channels.

Res2Net-based extractor. We employ employ Res2Net-50 [25] as our feature extractor backbone. Moreover, we integrate Res2Net with cardinality dimension [26], as well as SE block [24], Res2NeXt-50 and SE-Res2Net-50, respectively.

2.2. Standalone Countermeasure (CM) systems

Spoofing detection is a binary classification task which aims at differentiating spoofed speech from bonafide speech. For each test utterance, two hypotheses are computed: either it is bonafide speech, or it is a spoof attack. In our work, CM system is mainly based on the state-of-the-art (SOTA) system on the ASVspoof 2019 LA dataset, which is AASIST [13]. At the same time, the lightweight variant of AASIST (AASIST-L) is also adopted.

AASIST. This is a new end-to-end spoofing detection system based upon graph neural networks, which consists of two modules, high-level feature encoder and graph module. The RawNet2-based [12] encoder used for extracting high-level feature maps from raw input waveforms. Then, a heterogenous stacking graph attention layer (HS-GAL) is used to model spectral and temporal sub-graphs branches which consists of a heterogeneous attention mechanism and a stack node to accumulate heterogeneous information. To enable different branches to learn different groups of spoofing artefacts, each branch includes two HS-GALs and graph pooling layers, followed by a max graph operation (MGO).

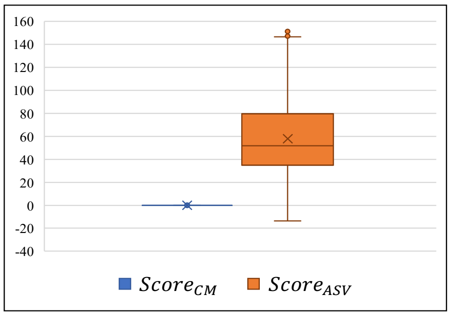

Figure 2: *Score distributions from Automatic Speaker Verification (ASV) and Countermeasure (CM) outputs when speaker embeddings without normalization.*

Efficient feature encoder. Even though the performance of AASIST is well, we find that training directly based on the raw waveforms, the training speed is very slow. Therefore, we explore to improve the feature encoder module which based on the frequency domain input and a lighter architecture. We adopt SE-ResNet-34 and the lightweight version of VGG [27] as high-level feature encoders, which shows that the training efficiency is greatly increased and the performance comparable to AASIST. At the same time, we further verify the effectiveness of the HS-GAL layer and MGO mechanism based on the graph neural networks. The reader is referred to Section 3 for further detail.

2.3. Score-level Ensemble

The ensemble of ASV and CM systems can be achieved ate score-level or at the model/feature level. In our work, we propose a general and efficient score-level ensemble solution capable of achieving a significantly improvement in SASV task.

It's intuitive that we normally use the score-sum ensembles using similarity scores generated from speaker embeddings produced by a pre-trained ASV subsystem and the scores produced by a pre-trained CM subsystem. When the score distribution of the two subsystems as shown in Figure 2 is inconsistent, the overall performance collapse. However, we use the normalized inner-product or cosine similarity as the similarity measurement for speaker embeddings, which significantly boost the performance.

To illustrate this, we perform an experiment which compares the speaker embeddings without normalization, i.e. using the unnormalized inner-product as the similarity measurement. Both ASV and CM subsystems scores are derived from SASV 2022 GitHub repository [2]. We follow the ASV development protocol of ASVspoof2019 LA dataset [9]. The results are listed in Table 1.

As shown in the table, speaker embedding normalization significantly improve the performance of SPF-EER when using the score-sum strategy. Embedding normalization seems to be a crucial step to ensemble independent ASV and CM systems. At the same time, we also compare cosine similarity which shows better performance.

Furthermore, based on the independence of the ASV and CM, we regard them as two independent probability events. The

[2] https://github.com/sasv-challenge/SASVC2022_Baseline

Table 1: *Effect of different score-level ensemble methods on three different EERs (%) of SASVC 2022 development partitions.*

Score Ensemble	Speaker Embedding		SV-EER	SPF-EER	SASV-EER
	Similarity	L2-Norm			
Sum	Cos	N/A	1.82	0.19	1.08
	Inner-Product	W	**1.49**	**0.09**	**0.79**
		W/O	2.84	20.41	17.63
Mul	Cos	N/A	1.82	0.13	1.08
	Inner-Product	W	1.49	0.16	0.81

goal of SASV system is only make the bonafide speech from target speaker has a higher probability score, and vice versa. Therefore, the similarity score of ASV is directly multiplied by the output score of CM system, which is the probability of target speaker and bonafide speech. As shown in Table 1, it also shows good performance when directly ensemble the two system scores by probabilistic multiplication. It's worth noting that when the similarity score of ASV is based on cosine similarity, linear transformation is required to the target interval range [0, 1] before probabilistic multiplication.

3. Experiments

Described in this section are the datasets, implementation details and metrics used for our experiments, together with specific implementation details of ASV and CM systems.

3.1. Datasets

The SASVC 2022 training and evaluation datasets originate from the ASVspoof 2019 LA partition [9] and VoxCeleb 2 [28].

For training the standalone CM system, we employ the ASVspoof 2019 LA partition dataset. The LA contains 17 attacks generated with SOTA TTS and VC technologies, where only six of them are known attacks (six logical attacks for training). In the training partition, which contains 22800 spoof and 2580 bonafide utterances.

For training the standalone SV system, we only employ the VoxCeleb 2 dataset, which contains over 1 million utterances for over 6,000 speakers. Moreover, we employ diverse additive noises and reverberations to make the ASV systems more robust. The additive noises are selected from the MUSAN corpus [29]; The reverberations are generated by using simulated small and medium room impulse responses [30].

The ASVspoof 2019 LA development partition is used for model selection during validation and system combination. We **don't** use any external data or data augmentation technique for training systems.

3.2. Implementation details

All systems are implemented using PyTorch [31], a deep learning toolkit in Python. Mainly implementation details of ASV and CM systems are consistent with the SASVC official baseline implementation described in [32].

The ASV system adopts four different types of feature extractors: (i) ECAPA-TDNN [33], which is consistent with the SASVC baseline ASV subsystem; (ii) SE-ResNet-34; (iii) SE-Res2Net-50; (iv) Res2NeXt-50. All feature extractors are extracted 64 dimensional Mel-filterbanks. Pre-emphasis with a coefficient of 0.97 is applied to the input signal. The spectrograms are extracted with a hamming window of 25 ms width and 10 ms frame shift. Mean and variance normalization is performed by applying instance normalization [34] to the input features. In the meanwhile, feature augmentation [35] is applied during model training to prevent overfitting and to improve generalization with a frequency and temporal masking dimension of 8 and 10, respectively.

During each ASV system training, AAM-Softmax [36] is used as loss function to optimize the networks, which has outstanding performance in ASV task [37]. Given batch size n and N training speakers, the AAM-Softmax loss L_{ASV} is formulated as:

$$L_{ASV} = -\frac{1}{n}\sum_{i=1}^{n} log\frac{e^{s(cos(\theta_{y_i}+m))}}{e^{s(cos(\theta_{y_i}+m))} + \sum_{j=1,j\neq y_i}^{N} e^{s(cos(\theta_j))}}$$

where θ_{y_i} is the angle between the sample embedding x_i with corresponding speaker identity y_i and the speaker prototype $\mathbf{W}_{y_i}$. θ_j is the angle with all other L_2-normalized speaker prototypes stored in a trainable matrix $\mathbf{W} \in \mathbb{R}^{D\times N}$ with D indicating the embedding size. The margin penalty is indicated with m. A scaling factor s is applied to increase the range of the output log-likelihoods. During training, we set $s = 30$, and m set 0.2.

The CM system also adopts four different types of end-to-end systems: (i) AASIST [13]; (ii) AASIST-L (lightweight version of AASIST); (iii) SE-ResNet-34-GPool, which the feature encoder module adopts the same structure as the ASV system; (iv) VGG-C-GPool, which is a compressed version of VGG-16 [27], containing only five convolutional layers. The input of the first two systems are fed to raw waveforms of 64,600 samples ($\approx$ 4 seconds), the last two take 64-dimentional Mel-filterbanks features as input and the graph module is the same as AASIST. All the CM systems is trained to minimise a weighted cross entropy (WCE) loss function, where the ratio of weights assigned to bona-fide and spoofed trials are 9:1 to manage the data imbalance in the ASVspoof 2019 LA training set.

3.3. Fusion and Calibration

We follow the greedy fusion scheme described in [38] to select the best system combination for our primary submission for SASVC 2022. Fusion and calibration are performed with logistic regression with the Bosaris toolkit [39] for multiple classifiers improved overall SASV-EER on the ASVspoof 2019 LA development data.

3.4. Evaluation metrics

System performance is assessed on the evaluation of SASVC 2022. The evaluation set is drawn from ASVspoof 2019 LA evaluation partitions. Performance is reported using three types of equal error rate (EER), including speaker verification (SV)-EER, spoof (SPF)-EER and spoof aware speaker verification (SASV)-EER. SV-EER describes the conventional ASV system performance without considering the presence of presentation attacks. SPF-EER denotes the EER for CM system which only consider whether an input is spoofed. SASV-EER describes overall performance, considering both speaker identity and spoofing, which as the primary metric to evaluate the SASV system performance.

Table 2: *Performance of different systems on three different EERs (%) of SASVC 2022 development and evaluation partitions.* ***Boldface*** *values are the best results.*

System ID	ASV		CM		SV-EER		SPF-EER		SASV-EER	
	Architecture	#Param	Architecture	#Param	Dev	Eval	Dev	Eval	Dev	Eval
Baseline1	ECAPA-TDNN	14M	—	—	1.88	1.63	20.30	30.75	17.38	23.83
Baseline2	Score-level Ensemble				32.88	35.32	0.06	0.67	13.07	19.31
Baseline3	Embedding-level Ensemble				12.87	11.48	0.13	0.78	4.85	6.37
1	ECAPA-TDNN	14M			1.49	1.04	0.09	1.47	0.79	1.26
2	SE-ResNet-34	7M			1.11	0.69	0.11	1.06	0.49	0.89
3	SE-Res2Net-50	10M	AASIST	292K	**0.20**	**0.30**	**0.07**	**0.98**	**0.13**	**0.70**
4	Res2NeXt-50	6M			0.43	0.37	0.07	1.21	0.20	0.86
5	ECAPA-TDNN	14M			1.62	1.23	0.13	1.26	0.84	1.25
6	SE-ResNet-34	7M			1.23	0.86	0.13	0.86	0.54	0.86
7	SE-Res2Net-50	10M	AASIST-L	83K	**0.27**	**0.48**	**0.13**	**0.78**	**0.27**	**0.63**
8	Res2NeXt-50	6M			0.54	0.54	0.13	0.99	0.34	0.80
9	ECAPA-TDNN	14M			1.84	1.71	0.21	1.07	1.01	1.51
10	SE-ResNet-34	7M			1.35	1.47	0.20	0.95	0.77	1.33
11	SE-Res2Net-50	10M	SE-ResNet-34-GPool	816K	**0.61**	1.14	0.20	0.91	0.47	1.02
12	Res2NeXt-50	6M			0.69	**0.97**	**0.20**	**0.88**	**0.47**	**0.91**
13	ECAPA-TDNN	14M			1.68	1.68	0.13	0.90	0.97	1.33
14	SE-ResNet-34	7M			1.28	1.30	0.13	0.88	0.74	1.10
15	SE-Res2Net-50	10M	VGG-C-GPool	165K	**0.54**	0.97	0.13	0.80	**0.40**	0.89
16	Res2NeXt-50	6M			0.67	**0.93**	**0.13**	**0.75**	0.54	**0.86**
Fusion 1	1+5+9+13				1.53	1.17	0.07	0.47	0.81	0.91
Fusion 2	2+6+10+14				1.09	0.91	0.07	**0.24**	0.47	0.63
Fusion 3	3+7+11+15				0.21	0.53	0.07	0.26	0.13	0.43
Fusion 4	4+8+12+16				0.43	0.50	0.07	0.29	0.27	0.45
Fusion 5	Fusion all single systems				**0.19**	0.51	**0.06**	0.32	**0.07**	0.45
Fusion 6	3+7+12+16				0.20	**0.45**	0.07	0.26	0.13	**0.37**

4. Results and Discussion

Table 1 compares effect of different score-level ensemble methods, the similarity score obtains from the normalized speaker embeddings in the ASV system is directly summed to the CM system score to achieve the best performance. Therefore, the comparison of the performance of different systems in Table 2 is based on this score-level integration method for ensemble ASV and CM systems.

Comparison with baseline systems. Table 2 presents a comparison of the performance with the baseline systems. We observe that the ensemble method of ASV and CM systems is crucially important. All our systems greatly exceed the performance of all baseline systems. SE-Res2Net-50 (ASV system) and AASIST-L (CM system) are integrated as the best ensemble system, achieves 0.63% on SASV-EER on the evaluation set, while SV-EER and SPF-EER reach 0.48% and 0.78%, respectively.

Comparison with standalone ASV systems. Table 2 compares the performance differences of standalone ASV systems. Res2Net-based ASV systems show better performance than other extractors under different CM system architectures, and the SE-Res2Net-50 outperforms others in terms of SV-EER and SASV-EER. It further illustrates that the performance of ASV system is important for overall SASV system.

Comparison with standalone CM systems. Comparing the performance of different CM systems, the AASIST-L system with a small amount of parameters has the best performance, while the SE-ResNet-34-GPool system with a large amount of parameters has the worst performance. At the same time, we can see that the frequency domain model VGG-C-GPool and the time domain model AASIST have comparable performance.

Comparison with fusion systems. Figure 2 also shows the performance of multi-systems fusion. Compare systems (Fusion 1-4), it can be seen that the importance of the ASV system performance. We fuse all single ensemble systems (Fusion5) and the optimal single ensemble systems (Fusion6), we can notice that Fusion6 achieves the best performance, reaching 0.37% performance on the eval set. Meanwhile, the performance results of Fusion6 system as our primary submission results.

5. Conclusions

In this paper, we investigate a general and efficient norm-constrained score-level ensemble method which jointly processes the embeddings extracted by ASV and CM systems in order to detect whether the test utterance is bonafide and belongs to the claim speaker. Furthermore, by exploring different structural feature encoders for ASV and CM subsystems, it is further verified that ensemble SASV system can be delivered better performance when both ASV and CM subsystems are optimized. The effectiveness of our systems are verified using official trials of SASVC 2022, where we achieved 0.45% SV-EER, 0.26% SPF-EER and 0.37% SASV-EER. It is worth noting that our methods in this paper are general, which can be highly efficient in practical applications.

6. Acknowledgements

We would like to thank Elevoc R&D colleagues Yongjie Yan, Hua Zhong and Chong Ma for many fruitful discussions. We gratefully acknowledge SASVC 2022 committee for designing and organizing the challenge.

7. References

[1] A. K. Jain, A. Ross, and S. Pankanti, "Biometrics: a tool for information security," *IEEE TIFS*, vol. 1, no. 2, pp. 125–143, 2006.

[2] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *ICASSP*, 2018, pp. 5329–5333.

[3] F. Zhao, H. Li, and X. Zhang, "A robust text-independent speaker verification method based on speech separation and deep speaker," in *ICASSP*, 2019, pp. 6101–6105.

[4] D. Cai, W. Cai, and M. Li, "Within-sample variability-invariant loss for robust speaker recognition under noisy environments," in *ICASSP*, 2020, pp. 6469–6473.

[5] P. Zhang, P. Hu, and X. Zhang, "Deep embedding learning for text-dependent speaker verification." in *Interspeech*, 2020, pp. 3461–3465.

[6] 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.

[7] A. Gomez-Alanis, J. A. Gonzalez-Lopez, and A. M. Peinado, "A kernel density estimation based loss function and its application to asv-spoofing detection," *IEEE Access*, vol. 8, pp. 108 530–108 543, 2020.

[8] Z. Wu, P. L. De Leon, C. Demiroglu, A. Khodabakhsh, S. King, Z.-H. Ling, D. Saito, B. Stewart, T. Toda, M. Wester *et al.*, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE/ACM TASLP*, vol. 24, no. 4, pp. 768–783, 2016.

[9] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee *et al.*, "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, p. 101114, 2020.

[10] J. Yang, R. K. Das, and H. Li, "Significance of subband features for synthetic speech detection," *IEEE TIFS*, vol. 15, pp. 2160–2170, 2019.

[11] J.-w. Jung, H.-j. Shim, H.-S. Heo, and H.-J. Yu, "Replay attack detection with complementary high-resolution information using end-to-end dnn for the asvspoof 2019 challenge," in *Interspeech*, 2019, pp. 1083–1087.

[12] J.-w. Jung, S.-b. Kim, H.-j. Shim, J.-h. Kim, and H.-J. Yu, "Improved rawnet with feature map scaling for text-independent speaker verification using raw waveforms," in *Interspeech*, 2020, pp. 1496–1500.

[13] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, "Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks," *arXiv preprint arXiv:2110.01200*, 2021.

[14] M. Todisco, H. Delgado, K. A. Lee, M. Sahidullah, N. Evans, T. Kinnunen, and J. Yamagishi, "Integrated presentation attack detection and automatic speaker verification: Common features and gaussian back-end fusion," in *Interspeech*, 2018, pp. 77–81.

[15] A. Sizov, E. Khoury, T. Kinnunen, Z. Wu, and S. Marcel, "Joint speaker verification and antispoofing in the i-vector space," *IEEE TIFS*, vol. 10, no. 4, pp. 821–832, 2015.

[16] B. Dhanush, S. Suparna, R. Aarthy, C. Likhita, D. Shashank, H. Harish, and S. Ganapathy, "Factor analysis methods for joint speaker verification and spoof detection," in *ICASSP*, 2017, pp. 5385–5389.

[17] J. Li, M. Sun, and X. Zhang, "Multi-task learning of deep neural networks for joint automatic speaker verification and spoofing detection," in *APSIPA*, 2019, pp. 1517–1522.

[18] J. Li, M. Sun, X. Zhang, and Y. Wang, "Joint decision of anti-spoofing and automatic speaker verification by multi-task learning with contrastive loss," *IEEE Access*, vol. 8, pp. 7907–7915, 2020.

[19] A. Gomez-Alanis, J. A. Gonzalez-Lopez, S. P. Dubagunta, A. M. Peinado, and M. M. Doss, "On joint optimization of automatic speaker verification and anti-spoofing in the embedding space," *IEEE TIFS*, vol. 16, pp. 1579–1593, 2020.

[20] H.-j. Shim, J.-w. Jung, J.-h. Kim, and H.-j. Yu, "Integrated replay spoofing-aware text-independent speaker verification," *Applied Sciences*, vol. 10, no. 18, p. 6292, 2020.

[21] P. Zhang, P. Hu, and X. Zhang, "Investigation of imu&elevoc submission for the short-duration speaker verification challenge 2021," in *Interspeech*, 2021, pp. 2322–2326.

[22] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," in *Interspeech*, 2018, pp. 2252–2256.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018, pp. 7132–7141.

[25] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE TPAMI*, vol. 43, no. 2, pp. 652–662, 2019.

[26] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017, pp. 1492–1500.

[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[28] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Interspeech*, 2018, pp. 1086–1090.

[29] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[30] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*, 2017, pp. 5220–5224.

[31] S. Li, Y. Zhao, R. Varma, O. Salpekar, P. Noordhuis, T. Li, A. Paszke, J. Smith, B. Vaughan, P. Damania *et al.*, "Pytorch distributed: experiences on accelerating data parallel training," *Proceedings of the VLDB Endowment*, vol. 13, no. 12, pp. 3005–3018, 2020.

[32] J.-w. Jung, H. Tak, H.-j. Shim, H.-S. Heo, B.-J. Lee, S.-W. Chung, H.-G. Kang, H.-J. Yu, N. Evans, and T. Kinnunen, "Sasv challenge 2022: A spoofing aware speaker verification challenge evaluation plan," *arXiv preprint arXiv:2201.10283*, 2022.

[33] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Interspeech*, 2020, pp. 3830–3834.

[34] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[35] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, 2019, pp. 2613–2617.

[36] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *CVPR*, 2019, pp. 4690–4699.

[37] J. Thienpondt, B. Desplanques, and K. Demuynck, "The idlab voxsrc-20 submission: Large margin fine-tuning and quality-aware score calibration in dnn based speaker verification," in *ICASSP*, 2021, pp. 5814–5818.

[38] C.-I. Lai, N. Chen, J. Villalba, and N. Dehak, "Assert: Anti-spoofing with squeeze-excitation and residual networks," in *Interspeech*, 2019, pp. 1013–1017.

[39] N. Brummer and E. De Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," *arXiv preprint arXiv:1304.2865*, 2013.