The DKU-OPPO System Description for the First SASV Challenge: Unknown Attack Samples Detection Using Embedding Augmentation and One-class Confusion Loss

Xingming Wang[1,3], *Xiaoyi Qin*[1,3], *Yikang Wang*[1], *Yunfei Xu*[2], *Ming Li*[1,3]

[1]Data Science Research Center, Duke Kunshan University, Kunshan, China
[2]Guangdong OPPO Mobile Telecommunications Corp., Ltd., Guangzhou, China
[3] School of Computer Science, Wuhan University, Wuhan, China

xingming.wang@dukekunshan.edu.cn, ming.li369@dukekunshan.edu.cn

Abstract

This paper describes our DKU-OPPO submission to the first Spoofing-Aware Speaker Verification (SASV) Challenge. The SASV challenge aims to build an ensemble system that simultaneously detects zero-effort impostor access attempts and spoofing attacks from target speaker audios. We first split the task into speaker verification and anti-poof tasks optimized separately. We introduce ResNet34, SE-ResNet34, SimAM-ResNet34 and ECAPA-TDNN models for speaker verification systems, achieving state-of-the-art (SOTA) performance in this field. For countermeasures systems, based on the AASIST-SAP model, we propose two methods to improve the generalization for unseen attack methods 1) Embedding Random Sampling Augmentation(ERSA). 2) One-Class Confusion Loss(OCCL). The AASIST-ASP mounted with OCCL model achieves 0% and 0.36% SPF-EER on the development and evaluation set. Finally, we compare and propose different fusion strategies, e.g., score-sum ensemble and cascade ensemble. The final submitted cascade system obtains the 0.209% SASV-EER on the evaluation set.

Index Terms: Anti-spoofing, Speaker verification, One-class

1. Introduction

Even though the performance of Automatic Speaker Verification(ASV) has improved dramatically during the past few years, the lack of consideration for the reliability of the ASV system makes it difficult to be applied in real-world scenarios. The spoofing countermeasure (CM) system is normally used to detect spoofing audios, which could ensure the security of ASV systems[1]. The ASVspoof challenge [2] has greatly contributed to the improvement of CM system performance. Although CM is framed as a subtask derived from ASV, most of the research on these two tasks has been carried out independently in previous works. This dichotomy may lead to CM systems not being well suited to some ASV scenarios due to overfitting and domain mismatch. To address this gap, the organizers of the ASVspoof Challenge proposed the tandem detection cost function (t-DCF) metric [3], which is highly correlated to both the ASV system and the CM system, to replace the equal error rate (EER) metric, which relied only on the CM system itself. However, the ASVspoof Challenge still focuses on designing and optimizing a stand-alone CM system to calculate the min t-DCF metric in combination with a given official black-box ASV system. This prevents participants from improving the overall performance by enhancing the ASV system or leveraging joint optimization. Therefore, the first spoofing-aware speaker verification (SASV) challenge [4], which aims to promote the development of integrated systems that can perform both ASV and CM, has been held this year. The key goal of this challenge is to build a combined system that can detect both zero-effort impostor access attempts and spoofing attacks from target speaker audios simultaneously.

The first SASV challenge focuses on logical access spoofing attacks (LA), such as text-to-speech (TTS) and voice conversion (VC) rather than physical access spoofing attacks (PA). Due to the lack of corpora and uncertainty of application prospects, few studies involving joint ASV and CM optimization have been conducted in the past. As mentioned in the official evaluation plan [4], jointly optimized solutions can generally be classified into two categories. The first is ensemble systems based on a fusion of separate ASV and CM systems. Gomez et al. [5] used an embedding concatenation strategy to construct an ensemble classification system. Another approach is to build a single integrated system. Li et al. [6] proposed a single model using multi-task learning and contrastive loss.

Similar to the score combination approach with SASV baseline 1, we investigated and implemented a dual-system score cascade model, which effectively combines the classification scores of ASV and CM systems and decreases SASV-EER to 0.21%. Despite all the room for innovation, the cascade model has beaten all the innovative solutions we have tried, which means that the cascade system is still feasible without considering the computational overhead and real-time requirements. Also, we investigate several innovative schemes, including random embedding sampling augmentation and one-class confusion loss function, both of them operate effectively to improve the CM single-system performance.

The rest of this paper is organized as follows. In section 2, our submitted system for the SASV challenge is represented, which mainly focuses on network structure and score fusion strategies. Detailed implementation information of the use of datasets and hyperparameters of models is provided in Section 3. Section 4 describes and discusses the results based on our submissions in the progress phase. Conclusions are provided in Section 5.

2. System Description

This section describes our submitted system for the first SASV challenge. Overall, we first constructed and improved our ASV and CM systems with SOTA models separately and fused scores from two systems in the combined system similar to Baseline 1.

2.1. ASV subsystem

In this part, we will introduce four different structure speaker verification systems, including the ResNet, SE-ResNet, SimAM-ResNet and ECAPA-TDNN models.

2.1.1. ResNet

For the ResNet module, we adopt the same structure as [7]. The network structure contains three main components: a front-end pattern extractor, an encoder layer, and a back-end classifier. The ResNet34[8] structure is employed as the front-end pattern extractor, which learns a frame-level representation from the input acoustic feature. The widths (number of channels) of the residual blocks are $\{32, 64, 128, 256\}$. The global statistic pooling (GSP) layer, which computes the mean and standard deviation of the output feature maps, can project the variable length input to the fixed-length vector. The output of a fully connected layer with 128 dim followed after the pooling layer is adopted as the speaker embedding layer. The ArcFace[9] (s=32,m=0.2) which could increase intra-speaker distances while ensuring inter-speaker compactness is used as a classifier .

2.1.2. ResNet mounted with attention module

The attention mechanisms achieve great success in the ASV field. Following our previous work[10], we adopted the SE-ResNet and SimAM-ResNet for speaker verification systems. The Squeeze-and-Excitation (SE) module [11] employs the channel-wise attention to capture the task-relevant features. The SimAM is designed based on some well-known neuroscience theories and generates 3D attentions weights for the feature maps. Different from the ResNet system, we increase the widths from $\{32, 64, 128, 256\}$ to $\{64, 128, 256, 512\}$. The encoding layer is based on attentive statistics pooling (ASP) [12]. The speaker embedding is with a dimension of 256. The classifier is the same as with the ResNet system, and the detailed configuration of the neural network is shown in [10].

2.1.3. ECAPA-TDNN

The ECAPA-TDNN network[13] achieved great success in speaker verification in recent years. For this model, 1024 feature channels were used to scale up the network. The dimension of the bottleneck in the SE-Block is set to 256. The front-end feature extractor is followed by an attentive statistics pooling layer[12] that calculates the mean and standard deviations of the final frame-level features. The classifier is also the same as the ResNet system.

2.2. CM subsystem

This subsection describes the basic network structure of our CM subsystems and the One-Class Confusion Loss and the Embedding Random Sampling Augmentation we proposed in this challenge.

2.2.1. Basic network architecture

We use AASIST [14], which is also used in the baseline as the backbone network. AASIST has a RawNet2 based encoder and a graph attention network based graph module. AASIST utilizes raw waveforms as input to learn meaningful high-dimensional spectro-tempora feature maps and then extract graph nodes of feature maps in temporal and frequency domains respectively. With a stack node that learns information from all nodes, the final CM embedding is attained by concatenating various nodes' mean and maximum values. Moreover, as mentioned in the paper [15], Tak et al. provide an improved architecture in which the max pooling layer of the encoded feature maps is replaced by 2D self-attentive pooling [16], named AASIST-SAP in this paper.

2.2.2. One-class confusion loss function

Although the basic models AASIST and AASIST-SAP obtain great results in the development and evaluation set, there is a great performance gap between the development and evaluation set since the attack algorithms are not overlapped. Therefore, it is necessary to reduce the domain gap. Although the domain of the attack algorithm is unpredictable, the space of bonafide audios is unique. Inspired by one-class learning [17, 18], we proposed the one-class confusion loss which is similar to pairwise confusion loss [19].

The binary cross-entropy loss can be defined as follows:

$$\mathcal{L}_{ce} = \sum_{i} -(y_i \log(p_i) + (1 - y_i) \log(1 - p_i))$$

where $y_i \in \{0, 1\}$ is class label and p_i is the probability output of classifier. The anti-spoof model is trained using a combined objective with the cross-entropy loss and the proposed one-class confusion loss, which is defined as:

$$\mathcal{L}_{occ} = \sum_{i} \sum_{j \neq i} ||e_i - e_j||^2$$

where e_i denotes the embedding vector extracted from the bonafide audios. The purpose of this loss function is to make the Euclidean distance of all real samples more compact in the embedding space. Since the attacks audios are unknown, the one-class confusion loss is only applied on bonafide audios during the training process. Therefore, the final combination loss function is defined as follows:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{occ}$$

where lambda is a constant hyperparameter. It is worth mentioning that not all our experiments use combination loss.

2.2.3. Embedding random sampling augmentation

Considering that the evaluation set contains many unseen logical attacks [20], we propose a fine-tuning embedding data augmentation strategy that aims to improve the robustness of the model for unknown scenarios inspired by [21]. The key idea of this method is to randomly sample from the boundary spoof Gaussian distribution, which may be closer to bonafide audios. Firstly, we initialize the embedding centers of bonafide audio and each type of spoof audio in the development set separately based on the pre-trained model. The boundary embedding centers of each type of spoof audio are defined as the average of the bonafide embedding center and spoof embedding centers. During fine-tuning, the boundary centers are dynamically updated based on the spoof embedding center of current iteration samples. Then we randomly generated samples From $N(\hat{\mu}, \Sigma)$ where N is a Gaussian distribution, $\hat{\mu}$ is the boundary spoof embedding center and Σ is the covariance matrix of spoof embeddings calculated in advance. After each 5 epoch training, the mean and covariance matrix of embedding centers will be updated during validation. A more detailed description can be found in 1,

Algorithm 1 Framework of embedding random sampling augmentation algorithmic.

Input:
The training, development data and labels, X_t Y_t X_d Y_d;
The pre-trained embedding extractor M;
The pre-trained classifier C;
Output: Embedding extractor M and classifier C;
$embd_{dev} \leftarrow M(X_d)$
$\mu_b \leftarrow \mathbb{E}[embd_{dev}|Y_d = bonafide]$
$\mu_{s[A01:A06]} \leftarrow \mathbb{E}[embd_{dev}|Y_d = spoof_{[A01:A06]}]$
$\hat{\mu}_{s[A01:A06]} \leftarrow (\mu_b + \mu_{s[A01:A06]})/2$
$\Sigma_s \leftarrow \text{cov}(embd_{dev}|Y_d = spoof)$
for $i \in [1, TotalEpoch]$ **do**
  if i % 5 **then**
    Update $\hat{\mu}_{s[A01:A06]}$, σ_s
  end if
  for x, y in Dataset(X_t, Y_t) **do**
    $e \leftarrow M(x)$
    $\mu_b \leftarrow \mathbb{E}[e|y = spoof]$
    $\mu_{bs} \leftarrow \hat{\mu}_{s[A01:A06]} * \alpha + \mu_b * (1 - \alpha)$
    $e_g \sim N(\mu_{bs}, \Sigma_s)$,$y_g \leftarrow spoof$
    loss $\leftarrow \mathcal{L}_{ce}$(C($e_g$),$y_g$) + $\mathcal{L}_{ce}$ (C(e),y)
  end for
end for

2.3. Combined system

2.3.1. Score-fusion system

Baseline 1 generates the final SASV score by simply score sum.

$$S_{fus} = S_{cm} + S_{sv}$$

where S_{cm} denotes the CM system score and S_{asv} denotes the ASV system score. There is great numerical variation between the scores of the baseline ASV system and the CM system. Thus, we explore normalized score multiplication in order to generate the SASV score with a more rational distribution [22].

$$S_{fus} = \sigma(S_{cm}) \times \sigma(S_{sv})$$

where σ denotes $sigmoid$ normalization. Moreover, we also adopt Bosaris[23] for score calibration.

$$S_{fus} = W_{cm} * S_{cm} + W_{sv} * S_{sv}$$

2.3.2. Cascade systems

Since both current ASV modules and CM modules we trained performing well and work independently, we explore building a cascade ensemble system. As is shown in Figure 1, the cascade system consists of two tandem modules: a) the first module is hard decisions based thresholds, the thresholds are determined by the equal error rate (EER) on the development set; b) the second module is soft decisions that the first module result turns the score. For example, when the score of the first module after hard decisions is negative, the corresponding value of the second score is reset to the minimum score of the development set. That is means that once the test audio is determined to be negative by one module, the outcome of another module is meaningless.

For cascade system, two options are considered in this paper: the first is ASV module followed by CM module, named Cascade-ASV-CM; and the other is CM module followed by ASV module, named Cascade-CM-ASV. The thresholds of the first system were determined by the equal error rate (EER) on the development set.

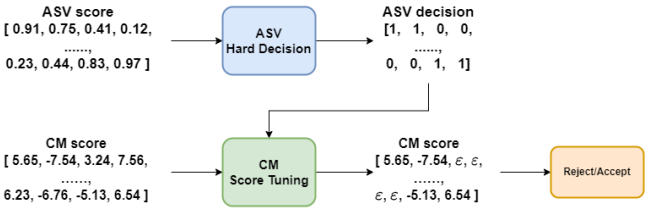

Figure 1: *The illustration of the ASV followed by CM cascade system. ε represents the minimum CM score in the development set. The CM followed by ASV cascade system is also achieved by exchanging ASV and CM system positions*

3. Experimental setup

3.1. Data usage and evaluation metrics

All datasets we used for training and validation are ASVspoof2019 [20] LA train partition, ASVspoof2019 LA development partition, and VoxCeleb 2 [24] as requested by the organizers. The ASVspoof2019 LA database consists of bonafide and spoof audios. Though the database has both speaker and spoofing labels, it was generally only used for voice anti-spoofing due to the low number of total speakers. The VoxCeleb 2 database contains 1128246 audios from 6112 speakers and was widely used for ASV training. The official SASV evaluation trial consists of audios from ASVspoof2019 LA evaluation partition, with unseen logical access spoofing attacks compared with audios in the train and development partitions.

The SASV-EER, which represents the equal error rate between target samples and both nontarget and spoof samples, is set as the primary metric. The SPF-EER and SV-EER are adopted as secondary metrics. Those two metrics only consider spoof negative samples and nontarget negative samples respectively.

3.2. Domain mismatch between ASVspoof2019 LA and VoxCeleb 2

Although the baseline AASIST-based CM system has excellent performance on the ASVspoof2019 evaluation set, we found that it performs poorly on VoxCeleb 2. Most bonafide audios in VoxCeleb 2 will be classified as spoof audios. We summarize two reasons that may lead to this phenomenon

1. Most of the audio in the VoxCeleb contains various noises and has been coded and transcribed.

2. The CM model trained based on the ASVspoof2019 LA dataset may learn the priori information of silent segments[25].

There are great domain mismatches between ASVspoof2019 LA and VoxCeleb 2 datasets, which makes it difficult to improve the performance of the CM system using VoxCeleb 2 dataset. Hence, by pre-trained model self-adaptive filtering, 20000+ domain-matched bonafide audios have been selected from VoxCeleb 2 dataset.

3.3. Model setup

3.3.1. ASV subsystem

For feature extraction, logarithmical Mel-spectrogram is extracted by applying 80 Mel filters on the spectrogram computed over Hamming windows of 20ms shifted by 10ms. We adopt the SOTA ASV models, namely ResNet34, SimAM-ResNet34 and ECAPA-TDNN, as the ASV system. The on-the-fly data augmentation [26] is employed to add additive background noise or convolutional reverberation noise for the time-domain waveform. The MUSAN [27] and RIR Noise [28] datasets are used as noise sources and room impulse response functions, respectively. To further diversify training samples, we apply amplification or playback speed change (pitch remains untouched) to audio signals. Also, we apply speaker augmentation with speed perturbation [29, 30, 31]. We adopt the Reduceonplateau learning rate (LR) scheduler with 0.1 initial LR. The SGD optimizer is adopted to update the model parameters.

3.3.2. CM subsystem

In contrast to the baseline training strategy, our trained AASIST-ASP network receives random length audio between 3-5 seconds as input. The initial learning rate is 0.001 with a Reduceonplateau learning ratescheduler. Adam optimizer is used to update the weights in models. The embedding random sample augmentation is only used during fine-tuning with 2 generated embeddings per center. Since there are 6 boundary spoof embedding centers, there will be 12 generated embeddings per batch. The batch size is set as 64 in this phase. And for the one-class confusion loss, λ is set as 1 during training.

4. Results and discussion

4.1. Results on ASV system

Table 1 reports the results of different speaker verification models. Our employed models achieve SOTA result on the VoxCeleb1 original test set. In addition, our models outperform the baseline system on the SV task. The ResNet with statistic pooling achieves the best single model performance. Since the attention mechanism relies on the large-scale data drive and easy overfits on the target domain data, the generalization of ResNet with statistic pooling is better than other models.

Table 1: *The performances of different speaker verification systems on the VoxCeleb1 original test set and SASV set.*

Model	Vox-O EER[%]	SV-EER[%]	
		Dev	Eval
ECAPA (Baseline)	-	1.86	1.64
ResNet GSP	0.851	**0.135**	**0.192**
SE-ResNet34 ASP	0.776	0.404	0.410
SimAM-ResNet34 ASP	0.643	0.404	0.252
ECAPA-TDNN	0.734	0.225	0.228

4.2. Results on CM system

Table 2 shows the anti-spoofing performance of different CM single systems. It can be seen from the table that the AASIST based model achieves a great performance improvement by replacing the max pooling layer with ASP. In addition, the model achieves a further generalizability improvement in the evaluation set by fine-tuning with the embedding random sampling augmentation strategy despite a little performance degradation in the development set. It is worth mentioning that although we extracted the VoxCeleb subset using adaptive filtering of the pre-trained model, simply adding these bonafide samples for the training set did not seem to work. The addition of the one-class confusion loss effectively solves this phenomenon, and the model's overall performance is further enhanced. This improvement may be attributed to the fact that this loss function makes the Euclidean distances between embeddings of bonafide audios in VoxCeleb 2 and ASVSpoof2019 LA closer, and thus the bonafide embedding space is more compact.

Table 2: *Comparison of different single CM systems based on SPF-EER used in SASV challenge. The $ERSA$ represents embedding random sample augmentation while the $OCCL$ denotes training with one-class confusion loss respectively. The Vox-sub represents sub-bonafide audios selected from VoxCeleb 2 as mentioned in Sec 3.2*

Model	Data	SPF-EER[%]	
		Dev	Eval
AASIST(Baseline)	19LA	0.07	0.67
AASIST-SAP[15] V1	19LA	0.067	0.570
AASIST-SAP+$ERSA$	19LA	0.067	0.510
AASIST-SAP[15] V2	19LA+ Vox-sub	0.049	1.564
AASIST-SAP+$OCCL$	19LA + Vox-sub	**0.000**	**0.360**

4.3. Results on ensemble system

The results of our experiments are summarized in Table 3, which include the baseline systems, the SV sub-systems, the CM sub-systems, the score fusion systems and the cascade systems.

4.3.1. Score fusion system performance

As can be seen from the score fusion section of the table, the simple summation method performs poorly due to the differences among the score distributions of the different systems. This problem can be effectively mitigated by normalizing the scores through the $sigmoid$ function and multiplying them together [22]. The optimal result in this part is also obtained by this method.

4.3.2. Cascade system performace

The cascade systems section of the table shows only partial results from our cascade combinations. We have noted that the AASIST CM system in the baseline is highly complementary to the systems trained by ourselves. Furthermore, we observe that while the Cascade-CM-ASV approach performed better on the development set, the Cascade-ASV-CM approach generally performed better on the evaluation set, possibly due to the fact that the development set has appeared in the training data of CM systems. In the other words, for unknown scenarios, the SV model EER will be lower and more suitable as the hard decision module in a tandem system. Therefore, we ultimately chose to submit the results of the Cascade-ASV-CM method.

Table 3: *Performance of different systems evaluated in the SASV Challenge. Due to a large number of combinations, only selected combinations are listed. The σ denotes $sigmoid$ normalization and $\times$ denotes multiplication.*

ID	Model	Fusion	SV-EER[%]		SPF-EER[%]		SASV-EER[%]	
			Dev	Eval	Dev	Eval	Dev	Eval
1	AASIST(CM, Baseline)	-	46.01	49.24	0.07	0.67	15.86	24.38
2	ECAPA-TDNN (SV, Baseline)	-	1.86	1.64	20.28	30.75	17.31	23.84
	Baseline 1 (official)	Sum	32.89	35.33	0.07	0.67	13.06	19.31
	Baseline 2 (official)	Back-end ensemble	7.94	9.29	0.07	0.80	3.10	5.23
CM System								
3	AASIST-SAP V1	-	48.543	48.464	0.067	0.570	16.298	25.344
4	AASIST-SAP+$ERSA$	-	47.304	47.188	0.067	**0.510**	15.963	24.655
5	AASIST-SAP V2	-	46.968	50.575	**0.049**	1.564	16.212	25.943
6	AASIST-SAP+$OCCL$	-	50.644	55.161	**0.000**	**0.360**	16.328	26.872
ASV System								
7	ResNet34 GSP	-	**0.135**	**0.192**	14.084	23.069	11.616	17.449
8	SE-ResNet34 ASP	-	0.404	0.410	11.540	22.402	9.745	16.888
9	SimAM-ResNet34 ASP	-	0.404	0.252	12.011	22.500	10.512	16.994
10	ECAPA-TDNN	-	**0.225**	**0.228**	14.420	21.899	12.354	16.795
Score-fuse	ASV & CM							
	ID 7 & ID 1	Sum	30.526	33.800	0.054	0.484	11.725	19.423
	ID 7 & ID 1	σ and $\times$	1.011	4.134	0.067	0.512	0.670	3.557
	ID 7 & ID 1	Bosaris	0.520	0.917	0.130	1.226	0.266	1.124
	ID 7 & ID 6	Sum	0.135	0.410	0.000	0.261	0.128	0.410
	ID 7 & ID 6	σ and $\times$	0.135	0.357	0.004	0.332	0.132	0.354
	ID 7 & ID 6	Bosaris	0.135	0.353	0.004	0.344	0.131	0.353
	ID 7+8+10 & ID 6	Sum	0.173	0.298	0.004	0.288	0.100	0.300
	ID 7+8+10 & ID 6	σ and $\times$	0.173	0.291	0.004	0.322	0.100	0.310
	ID 7+8+10 & ID 6	Bosaris	0.173	0.285	0.004	0.451	0.100	0.354
	ID 7 & ID 1+3+4+6	Sum	40.066	42.011	0.009	0.225	14.278	22.432
	ID 7 & ID 1+3+4+6	σ norm and Sum	0.202	0.417	0.000	0.223	0.135	0.354
	ID 7 & ID 1+3+4+6	σ and $\times$	0.202	0.354	0.009	0.223	0.135	0.335
	ID 7 & ID 1+3+4+6	Bosaris	47.237	47.318	0.000	0.242	15.682	24.009
	ID 7+8+9+10 & ID 1+3+4+6	σ and $\times$	0.269	0.345	0.009	0.223	0.107	0.282
	ID 7+8+10 & ID 1+6	σ and $\times$	0.202	0.317	0.000	0.186	0.103	0.261
Cascade	ASV & CM							
	ID 7 & ID 6	Cascade-ASV-CM	**0.135**	0.205	0.135	0.410	0.135	0.391
		Cascade-CM-ASV	**0.135**	0.410	**0.000**	0.298	0.128	0.410
	ID 7+8+10 & ID 1+3+4+6	Cascade-ASV-CM	0.202	0.462	0.202	0.219	0.202	**0.223**
		Cascade-CM-ASV	0.173	0.242	**0.004**	0.308	**0.100**	0.261
	ID 7+8+10 & ID 1+4+6	Cascade-ASV-CM **(best)**	0.202	0.462	0.202	0.186	0.202	**0.209**
		Cascade-CM-ASV	0.173	0.242	0.000	0.230	**0.096**	0.242

5. Conclusion

In this paper, we describe our DKU-OPPO submitted systems for the first SASV challenge. Two ensemble solutions are discussed in the paper, the first being a score fusion strategy where we obtained 0.27% SASV-EER on the evaluation set using $sigmoid$ normalization followed by multiplication. The second solution is building a cascade-based system, which ultimately achieved 0.21% EER on the evaluation set. Moreover, we propose an embedding random sampling fine-tuning strategy and the one-class confusion loss, both of which improve the performance of the CM subsystem. Taking the submitted system as the pivot, we will further explore the differences between the various combinatorial strategies and a single system structure that outperforms cascaded systems.

6. References

[1] 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.

[2] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, and H. Delgado, "ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection," in *Proc. ASVspoof2021 workshop*, 2021, pp. 47–54.

[3] 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 ," in *Proc. Speaker Odyssey*, 2018, pp. 312–319.

[4] 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.

[5] 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 Transactions on Information Forensics and Security*, vol. 16, pp. 1579–1593, 2020.

[6] 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.

[7] W. Cai, J. Chen, and M. Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," in *Proc. Speaker Odyssey*, 2018, pp. 74–81.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[9] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *Proc. CVPR*, 2019, pp. 4685–4694.

[10] X. Qin, N. Li, C. Weng, D. Su, and M. Li, "Simple attention module based speaker verification with iterative noisy label detection," in *Proc. ICASSP*, 2022.

[11] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *Proc. CVPR*, 2018.

[12] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive Statistics Pooling for Deep Speaker Embedding," *Proc. Interspeech*, 2018.

[13] D. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.

[14] 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.

[15] H. Tak, M. Todisco, X. Wang, J.-w. Jung, J. Yamagishi, and N. Evans, "Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation," *arXiv preprint arXiv:2202.12233*, 2022.

[16] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification." in *Interspeech*, vol. 2018, 2018, pp. 3573–3577.

[17] Y. Zhang, F. Jiang, and Z. Duan, "One-class learning towards synthetic voice spoofing detection," *IEEE Signal Processing Letters*, vol. 28, pp. 937–941, 2021.

[18] X. Wang, X. Qin, T. Zhu, C. Wang, S. Zhang, and M. Li, "The dku-cmri system for the asvspoof 2021 challenge: Vocoder based replay channel response estimation," *Proc. ASVspoof2021 workshop*, pp. 16–21, 2021.

[19] A. Dubey, O. Gupta, P. Guo, R. Raskar, R. Farrell, and N. Naik, "Pairwise confusion for fine-grained visual classification," in *Proc. ECCV*, 2018, pp. 70–86.

[20] 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.

[21] Y. Baweja, P. Oza, P. Perera, and V. M. Patel, "Anomaly detection-based unknown face presentation attack detection," in *Proc. IJCB*. IEEE, 2020, pp. 1–9.

[22] Y. Zhang, G. Zhu, and Z. Duan, "A new fusion strategy for spoofing aware speaker verification," *arXiv preprint arXiv:2202.05253*, 2022.

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

[24] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[25] N. Müller, F. Dieckmann, P. Czempin, R. Canals, K. Böttinger, and J. Williams, "Speech is Silver, Silence is Golden: What do ASVspoof-trained Models Really Learn?" in *Proc. ASVspoof2021 workshop*, 2021, pp. 55–60.

[26] W. Cai, J. Chen, J. Zhang, and M. Li, "On-the-Fly Data Loader and Utterance-Level Aggregation for Speaker and Language Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1038–1051, 2020.

[27] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.08484*.

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

[29] H. Yamamoto, L. K.A., K. Okabe, and T. Koshinaka, "Speaker Augmentation and Bandwidth Extension for Deep Speaker Embedding," in *Proc. Interspeech*, 2019, pp. 406–410.

[30] W. Wang, D. Cai, X. Qin, and M. Li, "The DKU-DukeECE Systems for VoxCeleb Speaker Recognition Challenge 2020," *arXiv.2010.12731*.

[31] X. Qin, C. Wang, Y. Ma, M. Liu, S. Zhang, and M. Li, "Our Learned Lessons from Cross-Lingual Speaker Verification: The CRMI-DKU System Description for the Short-Duration Speaker Verification Challenge 2021," in *Proc. Interspeech*, 2021, pp. 2317–2321.