Single domain generalization for audio deepfake detection

Yuankun Xie[1,†], Haonan Cheng[1,†], Yutian Wang[1] and Long Ye[1,*]

[1]State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing, China

Abstract

Audio deepfake detection (ADD) is a prominent problem in artificial intelligence. With diverse spoofing attacks emerging continually, generalization of ADD algorithms in the face of unknown domains and robustness in complex environments become key points for this field. However, when only limited and low-quality learning data is available, as in the case of ADD 2023 Challenge Track 1.2, it is an open issue to achieve good generalization and robustness. In this paper, we propose a **S**huffle **M**ix **A**ggregation and **S**eparation **D**omain **G**eneralization (SM-ASDG) method which enables single-domain generalization. Specifically, we first design a pre-processing module to improve the robustness of the method against low-quality data. Next, we split the single domain into multiple data domains via the proposed data shuffle module. Finally, a well-generalized feature space is constructed through the designed feature extractor and MixStyle domain classifier. The proposed SM-ASDG obtain the weighted equal error rate (WEER) of 23.17% on ADD Challenge Track 1.2, which achieves the Top-5 rank in the challenge.

Keywords

Audio deepfake detection, single domain generalization, self-supervised representation, ADD challenge

1. Introduction

Audio deepfake detection (ADD) is an important yet challenging task, which has raised several concerns due to its high societal impact [1, 2, 3]. This task aims to accurately classify real and fake audio, where one of the main challenges is to identify accurately in the face of unknown spoofing methods or low quality audio.

In recent years, several works [4, 5] achieve promising results on intra-domain datasets. However, the performance of these methods degrades significantly when extending to cross-domain scenarios [6]. This is mainly due to the fact that these methods do not take sufficient account of the unknown domain and the damaged audio quality. Consequently, the issues of generalization and robustness become two key concerns for ADD.

To address generalization and robustness issues, some methods [2, 3] adopt data augmentation schemes to improve model performance by learning diverse audio features over a larger amount of data. Specifically, Piotr et al. [7] utilize a combination of three deepfake and spoofing datasets to increase the training stability. However, larger data sets also lead to higher computational costs. Moreover, as forgery techniques are constantly updated, there are always unknown attack methods outside the domain. Therefore, it is not sufficient to rely on data augmentation and further strategies for improving generalizability are required.

To this end, several methods propose the domain invariant representation learning (DIRL) strategy [8, 9, 10] in order to overcome the issue of generalizing to invisible target domains with limited source data. The DIRL strategy aims to reduce representation differences between multiple different source domains to ensure domain invariance. However, for situations where multiple source domains are not available, as in the case of the ADD 2023 Audio fake game (FG) Challenge [11] where there is only one acceptable training set, the DIRL strategy cannot be applied effectively. In addition, the performance of the ADD method degrades significantly when a large amount of noise, reverberation and other disturbances are mixed into the source domain data. *Therefore, how to construct ADD models with good generalizability and robustness based on single-domain, low-quality data is an open problem that remains to be explored.*

In this paper, we introduce a novel **S**huffle **M**ix **A**ggregation and **S**eparation **D**omain **G**eneralization (SM-ASDG) method for single-domain ADD. The key idea of our approach is assuming that in an ideal classification feature space, the data distribution of real audio can be clustered in a single set, while the data distribution of fake audio should be more scattered. This is because different types of attacks impact more on spoofing audio, although different recording devices or channel also have some impact on real audio. Based on this idea, we propose a modified DIRL strategy that allows the application to a single source domain. To be specific, the proposed SM-ASDG contains a total of four modules, namely pre-processing, data shuffle, feature extractor and MixStyle domain classifier. First, the pre-processing module contains three carefully designed pre-processing strategies

IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis (DADA 2023), August 19, 2023, Macao, S.A.R

*Corresponding author.
† These authors contributed equally.
xieyuankun@cuc.edu.cn (Y. Xie); haonancheng@cuc.edu.cn (H. Cheng); wangyutian@cuc.edu.cn (Y. Wang); yelong@cuc.edu.cn (L. Ye)
https://haonancheng.cn/ (H. Cheng)
0000-0002-8366-9011 (Y. Xie); 0000-0003-3407-4318 (H. Cheng)

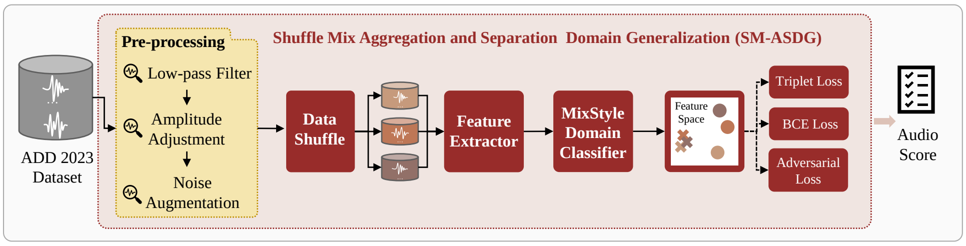

Figure 1: The whole pipeline of our proposed SM-ASDG method. Our proposed framework consists of four parts: the pre-processing module, the data shuffle module, the feature extractor and the MixStyle domain classifier.

to eliminate the effects of noise and other factors on the model and to improve the robustness of the algorithm. Second, the data shuffle module is introduced to approximate a multi-source domain situation by splitting the single domain. Then, we construct a feature extractor based on W2V2-XLS-R [12]. Finally, we propose a MixStyle domain classifier by mixing feature statistics of training samples across source domains. By this means, the model can diversify the style information at the bottom layers of the networks. Our proposed SM-ASDG method achieve outstanding results in the ADD 2023 Audio FG Challenge, demonstrating the effectiveness of our method. In summary, our contributions are as follows:

- We propose SM-ASDG, a high efficient audio deepfake detection method which achieves the top-5 rank in the ADD 2023 challenge track 1.2.
- A modified DIRL strategy is proposed for the situation where only a single source domain is available. The proposed domain generalization strategy can improve performance by 9% to 11% on different models.
- The effects of a series of pre-processing strategies are explored. In addition to common preprocessing methods such as noise addition and reverberation, we also explore the effect of silent frames in forgery identification performance.

2. Proposed Method

2.1. Preprocessing

To address the effect of codec variabilities, we first adopt a low-pass filter [13]. This is because that in complex speech scenarios, focusing on the low-frequency speech components can often make the model more effective. Specifically, we utilize a Chebyshev Type I lowpass filter to preprocess the original 16 kHz signal into a low-pass filtered signal. We set the order of the filter to 8, with maximum ripple and critical frequencies set to 0.05 and 4 kHz, respectively.

We further adjust the amplitude of signals due to the observation that the amplitude of genuine speech is different from that of spoofed speech. In the training set, we observe that the genuine speech has higher amplitude than spoofed speech. This may cause the model tends to classify high-amplitude speech as genuine and low-amplitude speech as spoofed during inference. Thus, we compute the average amplitude of genuine and fake speech and increase the amplitude of each fake speech in the training set to match the average amplitude of genuine speech, thereby equalizing their average amplitudes in training process.

To enhance the robustness of the model in a noisy situation, we introduce a noise enhancement strategy in the pre-processing. We add reverberation and noise obtained from MUSAN [14] and RIR [15] to the original speech, which is a high effective strategy in speech recognition and speaker verification.

2.2. Data Shuffle

To improve the generalization ability of the model, we divide the training data into three different domains randomly. Randomly shuffling the domains enriches the style information of each domain, allowing the domain adversarial loss to aggregate all real speech from various styles. In the experimental section, we further verify that randomly shuffling the domains is more effective than direct grouping the validation set into one domain and the training set into two domains.

2.3. Feature Extractor

We first extract features via a W2V2 based front-end, which is trained using a contrastive method with a masked feature encoder. The front-end feature extractor has a feature extractor with seven CNN layers to process speech signals of different lengths, followed by a Transformer network with 24 layers, 16 attention heads, and an embedding size of 1024 to obtain context representations. Consequently, the last hidden states from the

Table 1
The architecture of Mix Style Domain Classifier

Module	ConvFilter	Output
Input	-	(1,201,1024)
Conv2d/MFM/Pool	5 × 5 / 1 × 1	(32,100,512)
MixStyle	-	(32,100,512)
Conv2d/MFM/BN	1 × 1 / 1 × 1	(32,100,512)
Conv2d/MFM/Pool/BN	3 × 3 / 1 × 1	(48,50,256)
Conv2d/MFM/BN	1 × 1 / 1 × 1	(48,50,256)
Conv2d/MFM/Pool	3 × 3 / 1 × 1	(64,25,128)
Conv2d/MFM/BN	1 × 1 / 1 × 1	(64,25,128)
Conv2d/MFM/BN	3 × 3 / 1 × 1	(32,25,128)
Conv2d/MFM/BN	1 × 1 / 1 × 1	(32,25,128)
Conv2d/MFM/Pool	3 × 3 / 1 × 1	(32,12,64)
Reshape/Transformer	-	(64,384)
Flatten/FC	-	(16,512)

transformer provide reliable contextual information of genuine speech and different from the fake speech.

2.4. MixStyle Domain Classifier

After get the W2V2 feature from feature extractor, we propose a MixStyle Domain Classifier to generate the feature space by optimizing three different loss function. The detailed architecture is described in Table 1, which is modified on the traditional LCNN [16]. In the architecture, MFM means the Max-Feature-Map layer to select the critical channels for ADD task of the feature and BN means Batch Normalization. After MixStyle domain classifier, we get the feature space of the shape (16,512).

Through the mixing of training instance styles, we can implicitly synthesize novel domains, which results in increased domain diversity of the source domains and ultimately improves the generalizability of the trained model. Given an input batch x, we first random choose a reference batch $\tilde{x}$ from x. Then, Mixstyle computes the mixed feature statistics as follow:

$$\begin{aligned} \gamma_{\text{mix}} &= \lambda\sigma(x) + (1-\lambda)\sigma(\tilde{x}), \\ \beta_{mix} &= \lambda\mu(x) + (1-\lambda)\mu(\tilde{x}), \end{aligned} \tag{1}$$

where λ is the weight sample from the Beta distribution $\text{Beta}(\alpha, \alpha)$. We set α to 0.1 in our paper. Then, the style normalized feature x is computed by the mixed feature statistics,

$$\text{MixStyle}(x) = \gamma_{mix}\frac{x-\mu(x)}{\sigma(x)} + \beta_{mix}. \tag{2}$$

2.5. Loss Function

BCE loss. First, our main task is binary classification, which is to determine whether the features obtained are genuine or spoofed. We use several FC layer to down sample the feature from 512 to 1 and compute Binary Cross Entropy (BCE) to classify. It is worth mention that the feature normalization and weight normalization is used for this process, which will balance the numerical values of features and weights from speech signals across different domains, facilitating the convergence of the model.

Triplet loss. Our proposed ASDG strategy is that the real speech from different domain should be aggregated and the spoof one will be separate. The triplet mining method is suitable for the goal, which is defined as follow:

$$\begin{aligned} L_{tri} = \sum_{i}^{N} \left\| f\left(x_i^a\right) - f\left(x_i^r\right) \right\|_2^2 - \\ \left\| f\left(x_i^a\right) - f\left(x_i^f\right) \right\|_2^2 + \alpha, \end{aligned} \tag{3}$$

where x_i^a, x_i^r, x_i^f represent the anchor sample, real sample, and fake sample. By minimizing L_{tri}, the euclidean distance between the anchor and the real sample may get closer while the anchor may get further away from the fake sample. We set α to 0.1 which is a margin value.

Adversarial loss. In the feature space, the distribution of real speech should be aggregated regardless of domain. Thus, we design a single-side domain discriminator with Gradient Reverse Layer (GRL) [17]. Let $p(X_r)$ denotes the distributions of real feature and Y_D denotes the domain of X_r. The adversarial loss function of the domain discriminator is defined as follows:

$$\begin{aligned} &\min_{D}\max_{G} L_{ada}\left(G, D\right) = \\ &\quad - E_{x\sim P(X_r), y\sim Y_D} \sum_{d=1}^{3} p\left(y=d\right) log\left(D\left(G(x)\right)\right), \end{aligned} \tag{4}$$

where d denotes the domain label. The feature generator is trained to learn a robustness feature to spoof the domain discriminator in order to maximize L_{ada}. In the meantime, the discriminator is trained to identify the feature domain by minimizing. To achieve this goal, we use the Gradient Reversal Layer (GRL), which reverses the gradient during back propagation by multiplying negative dynamic coefficients. This makes the discriminator unable to identify the domain of the real feature, which leads to the aggregation of genuine speech in the feature space without being divided by domains.

Total loss. The total loss L_{all} for our system is defined as follow:

$$L_{all} = L_{BCE} + \lambda_1 L_{ada} + \lambda_2 L_{tri}, \tag{5}$$

where λ_1 and λ_2 set to 0.1 to balance the value of three different losses. By utilizing the L_{all} loss, we can construct an optimal classification feature space for ADD task, where genuine speech signals from diverse domains are clustered together while fake speech signals are separated from them.

3. Experiments

3.1. Dataset and metrics

All experiments are conducted on the ADD 2023 Audio FG-D datasets. There are 27,084 audio clips in the training set and 28,324 audio clips in the development set. We divide the dataset as 90%/10% for training and validation, respectively. The audio amplitude in the training set is inconsistent and contains noise, and there are repeated tail segments without valid information. The audio situation in the test set is much more complex, including noise, reverberation, background music, and a large number of silent clips. Therefore, how to improve the generalization and robustness of methods is the core challenge.

Weighted equal error rate (WEER) is used as the evaluation metric, which is defined as

$$WEER = \alpha EER_{R1} + \beta EER_{R2}, \tag{6}$$

where $\alpha = 0.4$ and $\beta = 0.6$ represent the weights for equal error rate (EER) obtained in round 1 (EER_{R1}) and round 2 (EER_{R2}) of ADD Challenge Track 1.2, respectively.

3.2. Implementation details

All training audio files are trimmed or padded to 4s. For baseline AASIST, the input is the raw waveform of about 4s (64000 samples). For baseline Resnet18, we use 80-dimensional LFCCs with a shape of (80,404) as front-end. During training, the parameters of W2V2 front-end are frozen. After front-end, we can get the last hidden states vector with shape of (201, 1024) as input of back-end. For training strategy, the Adam optimizer is adopted with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-9}$ and weight decay is 10^{-4}. The learning rate is initialized as 10^{-5} and halved every 5 epochs.

3.3. Ablation studies on architecture

Impact of backbone models and features. We first investigate the impact of the backbone models and features. As shown in Table 2, we compare with three baseline backbone models: AASIST [5], ResNet18 [18] and LCNN [19]. Furthermore, we compare W2V2 based feature and manual feature connected with the same LCNN back-end. It can be observed that the W2V2 based feature shows better performance than manual feature. This is due to that the W2V2 is trained on a large amount of real utterances from different source domains which can enhance the differential capability in complex scenarios. Moreover, results show that our SM-ASDG model outperforms all backbone models.

Table 2
Performance comparison with the state-of-the-art ADD models on the ADD-FG dataset.

Method	Feature	EER_{R1}	EER_{R2}
AASIST [5]	Raw Audio	49.59	49.21
ResNet18 [18]	LFCC	50.16	49.15
LCNN [19]	Mel	60.05	58.67
LCNN [19]	W2V2	38.57	35.24
SM-ASDG	W2V2	**24.06**	**22.59**

Table 3
Investigation on the MixStyle module.

Model	Feature	EER_{R1}	EER_{R2}
SM-ASDG w/o MIX	W2V2	26.97	27.09
SM-ASDG	W2V2	**24.06**	**22.59**

Impact of MixStyle. We further investigate the impact of the MixStyle units. As shown in Table 3, “SM-ASDG w/o MIX” denotes removing the MixStyle layer from our full model. It can be observed that the performance of the model decreases by 2.91% in round1 and 4.50% in round2 with the removal of MixStyle. This demonstrates the effectiveness of our MixStyle domain classifier module. This is due to the fact that the bottom layer of CNN corresponds to style information and the top layer corresponds to label information. MixStyle enables the diversification of the bottom style information of LCNN. That is, our model can generate diverse new styles of real speech and fake speech to enhance the ability of domain generalization.

Visualization for feature To analyze the effect of the MixStyle and our proposed ASDG backbone, we visualize the distribution of different hidden features using T-SNE [20]. As shown in Figure 2, we randomly select 360 samples for three source data domains. In each domain, we select 60 samples for real utterances and 60 for fake utterances. Figure 2 (a) and (b) demonstrate that the hidden feature distributions become more distinct after applying MixStyle, indicating that MixStyle facilitates the diversification of the bottom style information in LCNN. The feature space depicted in Figure 2 (c) aligns with our conception of an ideal feature space by ASDG, where genuine speech signals are clustered together, while synthetic ones are segregated.

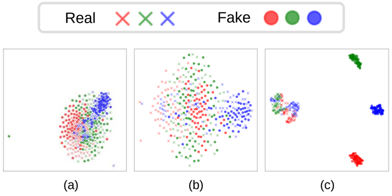

Figure 2: Feature visualization for three different layer. Subfigure (a), (b) and (c) exhibit the visualization result of the layer before applying mixstyle, after mixstyle and final feature space. Different colors indicate features from different domains, which we shuffle to create three new domains. Different shapes represent different category information: point=fake, cross=real.

Table 4
Performance comparison with the same model in different domain segmentation strategies. We propose several variants to investigate the impact of shuffled data and directly divided data.

Domain segmentation strategy	EER_{R1}	EER_{R2}
ASDG (direct)	48.56	47.89
ASDG (shuffle)	37.07	38.67
ASDG+Rawboost (direct)	46.97	44.67
ASDG+Rawboost (shuffle)	34.54	33.27
ASDG+Rawboost+RM (direct)	42.70	41.82
ASDG+Rawboost+RM (shuffle)	32.27	31.44
ASDG+RM (direct)	37.55	38.67
ASDG+RM (shuffle)	**28.05**	**27.96**

3.4. Ablation studies on pre-processing

To improve the robustness and generalization of the model, we explore a series of pre-processing strategies, including data shuffle, noise augmentation, low-pass filtering, amplitude adjustment, and region of interest (ROI) detection.

Does data shuffle help? We first explore the efficacy of data shuffle strategy. As shown in Table 4, we design two domain segmentation schemes, namely data shuffle and direct division. The direct division refers to the directly using the test set and validation set as two separate source domains. The two domain segmentation schemes are used in four variant, namely ASDG model and the ASDG model with different data augmentation strategies. "Rawboost" denotes the raw data boosting and augmentation strategy [3]. We utilize the best performance strategy in ASVspoof2021LA, which combines linear and non-linear convolutive noise with impulsive, signal-dependent noise. "RM" means adding noise and reverberation from RIRs [15] and MUSAN [14] datasets to the audio of training set in a Kaldi [21] like manner. In each pair of comparison data (the red row in Table 4 and its upper row), we can observe that the shuffle strategy can effectively improve the forensic performance. Data shuffle can reduce the order and pattern in the dataset, thus improving the generalization of the model.

Does noise augmentation help? Since the test data contain a large amount of noise and background music that are not available in the source domain dataset, we incorporate a noise augmentation strategy in the preprocessing stage, that is, introducing noise during training to improve the robustness of the model. It can be seen from Table 5 (the fourth row from top) that when the noise augmentation strategy is removed, the model performance decreases by 9.02% in round1 and 9.43% in round2. In addition, the effectiveness of different noise augmentation strategies can also be seen in Table 4 (as shown in red rows), as the RM strategy can maximize the robustness of the model. So we ultimately choose RM strategy as the noise augmentation strategy.

Table 5
Performance comparison with different pre-processing strategies. "MIX" is shorthand for MixStyle, "AMP" is shorthand for amplitude adjustment, "LP" is shorthand for low-pass filtering, "NA" is shorthand for noise augmentation and 'ROI" is shorthand for ROI detection.

Pre-processing strategy	EER_{R1}	EER_{R2}
SM-ASDG w/o MIX	26.97	27.09
SM-ASDG w/o [MIX+AMP]	27.53	27.80
SM-ASDG w/o [MIX+AMP+LP]	28.05	29.24
SM-ASDG w/o [MIX+AMP+LP+NA]	37.07	38.67
SM-ASDG with ROI	26.25	26.45
SM-ASDG	**24.06**	**22.59**

Does low-pass filter help? To against complex speech scenarios, we add low-pass filters to focus on the core frequency of the speech. The result shown in Table 5 (the third row from top) indicates that low-pass filters can help improve the forgery detection performance.

Does amplitude adjustment help? The amplitude level of the data samples varies greatly in the training and testing datasets. However, we find in our experiments that the amplitude of the audio is learned by the model and affects the classification results. Therefore, we adjust the amplitudes to the same interval range uniformly during both testing and training. As shown in Table 5 (the second row from top), audio normalization has obvious effects on the performance.

Does ROI detection help? Due to the large number of silent segments that do not contain speech content in the test data, a straightforward idea to improve performance is to detect only speech segments, that is, to detect ROIs. However, as shown in Table 5 (the red row), when ROI detection is added, the overall performance of the model decreases by 2.19% in round1 and 3.86% in round2. This is mainly due to that silent segments also contain artifact information for distinguishing real and fake audio [22]. Therefore, simply eliminating silent segments does not improve the generalization and robustness of the model.

4. Conclusion

In this paper, we propose SM-ASDG, a novel shuffle mix aggregation and separation domain generalization method for single domain ADD. The proposed method achieves a WEER of 23.17% in ADD 2023 track 1.2 final ranking, which is one of the top-5 performing methods. The outstanding robustness and generalization of the proposed SM-ASDG model is due to our carefully designed preprocessing module, data shuffle and MixStyle domain classification module. In future works, we plan to embed more high-level semantic features of audio, such as sentiment features, into the model to further improve generalization.

References

[1] A. Khan, K. M. Malik, J. Ryan, M. Saravanan, Voice spoofing countermeasures: Taxonomy, state-of-the-art, experimental analysis of generalizability, open challenges, and the way forward, arXiv preprint arXiv:2210.00417 (2022).

[2] A. Cohen, I. Rimon, E. Aflalo, H. Permuter, A study on data augmentation in voice anti-spoofing, Speech Communication 141 (2022) 56–67.

[3] H. Tak, M. Kamble, J. Patino, M. Todisco, N. Evans, Rawboost: A raw data boosting and augmentation method applied to automatic speaker verification anti-spoofing, in: Proceedings of the ICASSP, IEEE, 2022, pp. 6382–6386.

[4] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, A. Larcher, End-to-end anti-spoofing with rawnet2, in: Proceedings of the ICASSP, 2021, pp. 6369–6373.

[5] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, N. Evans, Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks, in: Proceedings of the ICASSP, 2022, pp. 6367–6371.

[6] N. Müller, P. Czempin, F. Diekmann, A. Froghyar, K. Böttinger, Does Audio Deepfake Detection Generalize?, in: Proceedings of the Interspeech, 2022, pp. 2783–2787.

[7] K. Piotr, P. Marcin, S. Piotr, Attack agnostic dataset: Towards generalization and stabilization of audio deepfake detection, in: Proceedings of the Interspeech, 2022, pp. 4023–4027.

[8] T. Matsuura, T. Harada, Domain generalization using a mixture of multiple latent domains, in: Proceedings of the AAAI, volume 34, 2020, pp. 11749–11756.

[9] L. Zhou, J. Luo, X. Gao, W. Li, B. Lei, J. Leng, Selective domain-invariant feature alignment network for face anti-spoofing, IEEE Transactions on Information Forensics and Security 16 (2021) 5352–5365.

[10] Z. Wang, Z. Wang, Z. Yu, W. Deng, J. Li, T. Gao, Z. Wang, Domain generalization via shuffled style assembly for face anti-spoofing, in: Proceedings of the CVPR, 2022, pp. 4123–4133.

[11] Y. Jiangyan, T. Jianhua, F. Ruibo, Y. Xinrui, W. Chenglong, W. Tao, Z. Chuyuan, Z. Xiaohui, Z. Yan, R. Yong, X. Le, Z. Junzuo, G. Hao, W. Zhengqi, L. Shan, L. Zhen, L. Haizhou, Add 2023: the second audio deepfake detection challenge, in: Proceedings of the IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis, 2023.

[12] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, et al., Xls-r: Self-supervised cross-lingual speech representation learning at scale, arXiv preprint arXiv:2111.09296 (2021).

[13] Y. Wang, X. Wang, H. Nishizaki, M. Li, Low pass filtering and bandwidth extension for robust anti-spoofing countermeasure against codec variabilities, arXiv preprint arXiv:2211.06546 (2022).

[14] S. David, C. Guoguo, P. Daniel, Musan: A music, speech, and noise corpus, arXiv:1510.08484 (2015).

[15] K. Tom, P. Vijayaditya, P. Daniel, S. Michael L, K. Sanjeev, A study on data augmentation of reverberant speech for robust speech recognition., in: Proceedings of the ICASSP, 2017, p. 5220–5224.

[16] X. Wang, J. Yamagishi, A comparative study on recent neural spoofing countermeasures for synthetic speech detection, arXiv preprint arXiv:2103.11326 (2021).

[17] Y. Ganin, V. Lempitsky, Unsupervised domain adaptation by backpropagation, in: Proceedings of the ICML, 2015, pp. 1180–1189.

[18] Y. Ma, Z. Ren, S. Xu, RW-Resnet: A Novel Speech Anti-Spoofing Model Using Raw Waveform, in: Proceedings of the Interspeech, 2021, pp. 4144–4148.

[19] G. Lavrentyeva, A. Novoselov, S., M. Volkova, A. Gorlanov, A. Kozlov, Stc antispoofing systems for the asvspoof2019 challenge, arXiv preprint arXiv:1904.05576 (2019).

[20] L. Van der Maaten, G. Hinton, Visualizing data using t-sne., Journal of machine learning research 9 (2008).

[21] R. Mirco, P. Titouan, B. Yoshua, The pytorch-kaldi speech recognition toolkit, in: Proceedings of the ICASSP, 2019, pp. 6465–6469.

[22] Z. Yuxiang, W. Wenchao, Z. Pengyuan, The effect of silence and dual-band fusion in anti-spoofing system, in: Proceedings of the Interspeech, 2021.