ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10446444

# ROBUST CROSS-DOMAIN SPEAKER VERIFICATION WITH MULTI-LEVEL DOMAIN ADAPTERS

*Wen Huang[1], Bing Han[1], Shuai Wang[2], Zhengyang Chen[1], Yanmin Qian[1]†*

[1]Auditory Cognition and Computational Acoustics Lab
MoE Key Lab of Artificial Intelligence, AI Institute
Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[2]Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen, China

## ABSTRACT

Speaker verification encounters significant challenges when confronted with diverse domain data, often resulting in performance degradation due to domain mismatch. To enhance performance in cross-domain scenarios, we introduce the Domain Adapter, an adaptable module designed for specific domains. This module learns and integrates domain-specific information with speaker-related data, mitigating domain-related variations and promoting convergence of utterance embeddings from the same speaker across diverse domains. It offers configurability across multiple levels and is adaptable to various backbone architectures. Our proposed module substantially enhances cross-domain performance with minimal parameter increments while effectively generalizing to previously unseen domains. In our experiments, we present results on the 3D-Speaker dataset, which provides acoustically-relevant attributes crucial for domain categorization and the subsequent learning of domain information. The top-performing system integrated with domain adapters achieved 10.8%, 14.8%, and 21.1% EER improvements over the baseline across three 3D-Speaker dataset trials.

***Index Terms***— speaker verification, domain mismatch, cross-domain learning, 3D-Speaker

## 1. INTRODUCTION

Speaker verification, which involves the verification of a speaker's identity based on their vocal characteristics, has advanced considerably with the introduction of deep neural network (DNN)-based speaker embeddings. Past research primarily focused on enhancing architectures for extracting superior speaker embeddings [1, 2, 3, 4] and optimizing loss functions for improved discrimination [5, 6, 7, 8]. These efforts have achieved remarkable success, consistently surpassing previous ones on benchmark datasets such as VoxCeleb [9].

While academic progress in speaker verification on the VoxCeleb dataset has been substantial, real-world industry applications face many challenges due to complex practical scenarios. These scenarios involve diverse environmental conditions, distances, and recording equipment configurations for different individuals. This complexity underscores the issue of multi-domain or cross-domain speaker verification, where "domain" encompasses various factors, including language content, channels, acoustic environment, and more. When speaker models encounter data from diverse domains during training, registration, or testing, it often results in performance degradation due to domain mismatch.

To tackle the domain mismatch problem, various adaptation techniques have been developed, including discrepancy-based alignment and domain adversarial learning. Discrepancy-based alignment aims to minimize the discrepancy between domains and facilitate learning domain-invariant representations [10, 11, 12]. However, it relies on well-defined distance metrics and faces challenges with multiple domains. Domain adversarial learning, on the other hand, implicitly reduces distinctions among diverse domain data through a min-max two-player game [13, 14, 15, 16, 17]. Yet, achieving a balance between the two tasks poses a significant challenge in this method, adding complexity to the training process.

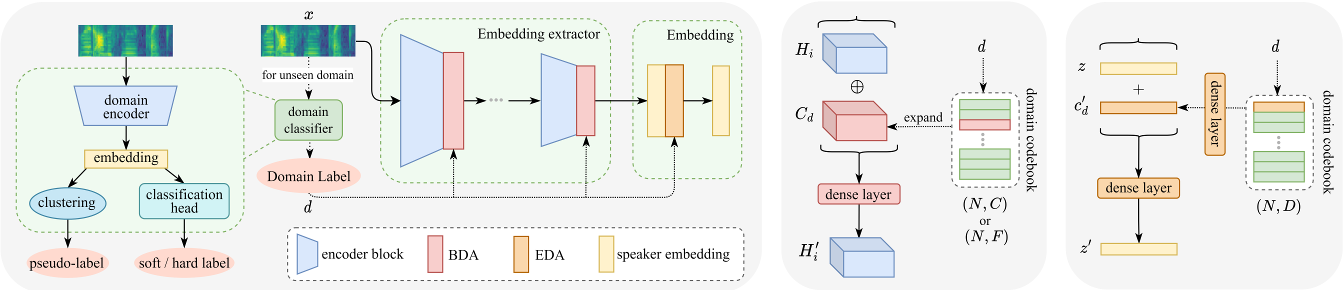
Furthermore, in recent years, academia has acknowledged the significance of this challenge and introduced challenging benchmark datasets to facilitate research on solutions. Notable examples include CNCeleb [18] and 3D-Speaker [19], which encompass data from diverse domains. In this paper, we will present our tailored solution for the 3D-Speaker dataset, addressing the nuances of speaker verification in these complex scenarios. Beyond speaker identities, the dataset of 3D-Speaker also provides labels of *Device*, *Distance*, and *Dialect* attributes. Each utterance within this dataset is recorded using various devices positioned at varying distances. By leveraging these acoustically-relevant attributes, we can effectively partition the data into distinct domains.

This enables us to establish a domain-aware speaker verification system, aiming to utilize domain labels for enhanced performance in cross-domain scenarios. In this paper, we proposed a novel module named "Domain Adapter". This module is domain-specific and highly adaptable, configurable at multiple levels and with different backbones. During the training process, it autonomously incorporates insights from each domain and integrates them with speaker-related data. This adaptive mechanism reduces domain-related variations and promotes the convergence of utterances from the same speaker across diverse domains, consequently improving the overall consistency of speaker information. It is noteworthy that our proposed framework achieves substantial performance improvements with a minimal parameter increment, even when facing a considerable number of domains, while also demonstrating the ability to generalize to previously unseen domains. Experimental results validate the efficacy of the proposed approach, with the best system achieving improvements of 10.8%, 14.8%, and 21.1% in terms of EER over the baseline across three trials in the 3D-Speaker dataset.

## 2. METHODS

To leverage multi-domain data effectively, this paper presents a two-step method: refine domain labels for more accurate domain information and integrate multi-level domain adapters into the model to enhance cross-domain learning performance.

[†] Yanmin Qian is the corresponding author



(a) Model architecture (b) 🟥 BDA (c) 🟧 EDA

**Fig. 1**. The proposed framework with Domain Adapters. Block Domain Adapters (BDA) are inserted between adjacent blocks, while the Embedding Domain Adapter (EDA) is applied to the embeddings. In (b) and (c), we use notations: $N$ for the number of domains, $F$ or $C$ for frequency or channel dimension, and $D$ for code dimension.

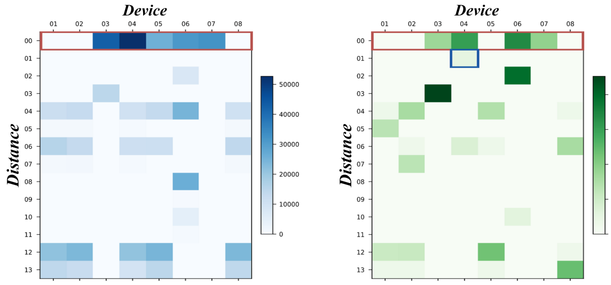
## 2.1. Refining Domain Labels

### 2.1.1. Domain partition strategy

In the context of 3D-Speaker [19], we introduce an innovative strategy to create domain labels. Our method involves partitioning domains based on the combination of two attributes: *Device* and *Distance*. This partitioning aims to create domains that encompass utterances recorded within the same acoustic environment.

There are two key rationales behind this strategy. Firstly, as illustrated in Figure 2, the joint distribution of these two attributes is notably sparse. For instance, certain devices are exclusively associated with a single specific distance. When modeling device and distance independently, the neural network may face challenges in developing a robust representation of device-related information, given that it has only encountered the device in one specific context.

Secondly, device and distance jointly play a pivotal role in shaping the characteristics of the acoustic environment. For example, the same device can yield distinct acoustic environments at varying distances, and conversely, different devices can produce diverse environments even at identical distances.



(a) Train data (b) Test data

**Fig. 2**. Visualization of data distribution in the 3D-Speaker dataset for the attributes *Device* and *Distance*. Color intensity represents the number of utterances. The red border encloses instances from unspecified domains, while the blue border encloses unseen domains.

### 2.1.2. Dealing with unspecified and unseen domains

A notable aspect of the dataset is a special category named *Distance00*, which includes utterances recorded without specified distances. As shown in Table 1, this segment of unspecified data constitutes more than 30% of the train set and over 20% of the test set. It encompasses varied distances and devices, necessitating further partition. Meanwhile, following our partition strategy, a fraction of the test set data does not align with domains seen in training.

**Table 1**. Statistical analysis of the 3D-Speaker dataset. The figures represent the numbers of utterances in each category.

| Set | #Total | #Unspecified | #Unseen |
|---|---|---|---|
| Train | 579,003 | 185,749 | - |
| Test | 18,782 | 5,040 | 258 |

To tackle these issues and enhance the model capacity to handle unseen domains, we developed a **domain classifier** utilizing training data with explicit domain labels. This classifier is designed for two purposes, as illustrated in Figure 1. First, obtain domain embeddings for data with unspecified or unseen domains in both the training and testing sets. These embeddings are utilized to generate pseudo-labels via clustering. In both training and testing stages, we combine the original labels with these pseudo-labels to establish domain labels, which serve as the ground truth. Second, generate soft or hard labels in the testing stage to address situations involving unspecified or previously unseen domains when actual ground truth is not accessible.

Furthermore, to better showcase the performance on unseen domains, we extracted pairs related to unspecified and unseen domains from the three trials of 3D-Speaker. These pairs were combined to form a new trial, named "out-of-domain trial", consisting of a total of 209,495 instances. The resources pertaining to domain partition and trial details can be accessed through the following link: https://github.com/holvan/cross_domain_speaker_verification.

## 2.2. Incorporating Multi-Level Domain Adapters

Once domain labels are acquired, the subsequent challenge lies in efficiently incorporating domain information into the processing pipeline. Common methods often involve extracting domain-related features from the raw input or the intermediate output of the trained model, and then compensating for their influence on the original input. However, this approach has two key challenges, i.e. accurate domain information extraction and effective domain knowledge integration.

Instead of attempting to extract domain information from the input, our approach enables the model to autonomously learn and leverage domain-specific information, while it can dynamically integrate it with speaker-related information to boost the performance. To facilitate this, we propose a novel structure called the **Domain Adapter (DA)**, as illustrated in Figure 1.

To facilitate the model's autonomous learning of domain information, we assume that domain information can be represented by discrete codes stored within a domain codebook denoted as $C$. The domain label $d$ serves as a means to select a specific code $c_d$ from the codebook, enabling the model to adjust its output based on the domain. This selection process can be formulated as:

$$c_d = \Sigma_{i=1}^{N} d_i \cdot C[i] \tag{1}$$

where $N$ represents the codebook size, $d_i$ is either a binary indicator (for one-hot hard labeling) or a weight value (for soft labeling).

To achieve multi-level adaptation, our framework incorporates two distinct types of domain adapters: the Block Domain Adapter (BDA) and the Embedding Domain Adapter (EDA).

**Block Domain Adapter (BDA)** The Block Domain Adapter operates between different blocks of the model's architecture. Considering the outputs of block $i$, denoted as $H_i$, BDA incorporates the current sample's domain label $d$ and discrete domain code $C_d$ through a dense layer $f$. This is represented as:

$$H_i' = f(H_i \oplus C_d) \tag{2}$$

where the $\oplus$ operator offers two interpretation modes: channel-wise addition (BDA-C) and frequency-wise addition (BDA-F), with the code $C_d$ being expanded in accordance with the chosen addition dimension.

**Embedding Domain Adapter (EDA)** The Embedding Domain Adapter functions at the final stage, targeting embeddings. This adapter employs dense layers $f$ and $g$, allowing the embedding output $z$ to be transformed through the domain's discrete code $c_d$:

$$z' = f(z + g(c_d)) \tag{3}$$

By incorporating these multi-level domain adapters, we enable the model to autonomously acquire domain-specific information and integrate it with speaker information using linear transformation. This fusion allows us to mitigate variations from different domains, thus facilitating the extraction of domain-invariant embeddings.

The training pipeline can be summarized as follows: 1) Train a domain classifier using training data with explicit domain labels and generate pseudo-labels for the remaining unspecified data in both the train and test sets. 2) Enhance the pre-trained speaker model by adding domain adapters. Fine-tune the adapters with the encoder block frozen, employing speaker labels for speaker classification loss and ground truth domain labels for adapter control. 3) During testing, extract speaker embeddings from testing data using ground truth or predicted domain labels and conduct scoring.

## 3. EXPERIMENTS

### 3.1. Experimental Settings

**Dataset** In our experiment, we trained all systems using the 3D-Speaker's training dataset [19], which includes 10,000 speakers and 579,003 utterances, totaling 1124 hours of valid speech. Domain labels are generated as described in Section 2.1. This process results in 35 domains, excluding unspecified ones. For unspecified data, their ground truth domain labels are derived using Kmeans clustering, resulting in 10 additional classes.

**Data processing** We use the 80-dimensional fbank features with frame length of 25ms and hop size of 10ms as the input of the model. The data is preprocessed with three types of augmentation, consistent with the official baseline [19]: 1) adding noise with MUSAN [23]; 2) adding reverberation with RIRs [24]; 3) speed perturbation [25].

**Networks** Several baseline models are employed for speaker verification, including ResNet34 [26, 27], CAM++ [21], and ERes2Net [22]. Notably, CAM++ and ERes2Net serve as the official baselines in [19]. For domain classification, we utilize ResNet18 [26]. When dealing with BDA, the code dimension is chosen either as the frequency dimension or the channel dimension, based on the specific block. To ease fine-tuning, BDAs can be initialized as identity functions, preventing distribution incompatibility during training. For EDA, we consistently set the code dimension to 32 and then use a dense layer to transform it into a 512-dimensional embedding.

**Training details** We employ the SGD optimizer for network optimization with a momentum value of 0.9. For speaker verification, we pretrain a baseline model and then finetune it with domain adapters. Both stages employ Additive Angular Margin (AAM-Softmax) loss [5], following the same training configuration in [19]. For domain classifier, we utilize cross-entropy loss for model training, achieving 99% accuracy.

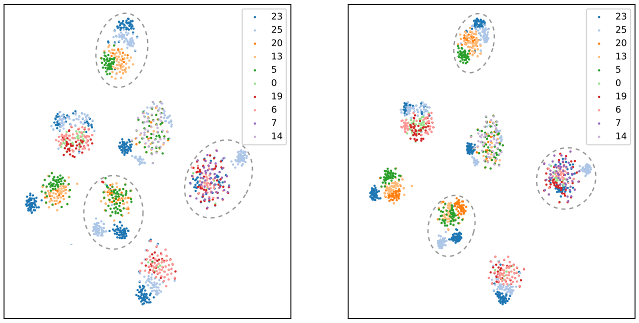
**Evaluation metrics** We use consine similarity for trial scoring. Results are reported in line with three defined trials [19] in terms of Equal Error Rate (EER) and minimum Detection Cost Function (minDCF) with $p_{target} = 0.01, C_{FA} = C_{miss} = 1$.

### 3.2. Performance of Domain Adapters

In this section, the performance comparisons of some baselines and our proposed Domain Adapters are shown in Table 2. We initiate experiments using various adaptation methods with the ResNet34 baseline. Initially, we assess the effectiveness of a straightforward adversarial training approach by introducing a module with a gradient reversal layer for domain classification on speaker embeddings. Although this resulted in some improvements, the impact was moderate rather than substantial.

Next, we test the systems equipped with domain adapters using ground truth domain labels. Among the different adaptations, BDA-F slightly outperforms BDA-C, using fewer parameters. This phenomenon can likely be attributed to the domains we have selected, which are closely associated with factors like device and distance. These factors tend to manifest more prominent variations in the frequency dimension, in contrast to the channel dimension observed in images. In contrast to processing intermediary layer outputs, the approach of directly manipulating speaker embeddings through EDA also yields favorable outcomes.

Moreover, combining both BDA and EDA techniques can further boosts system performance across three trials, improving upon the baseline by 10.8%, 14.8%, and 21.1% in terms of EER. This observation underscores the inherent complementarity between BDA and EDA, highlighting their capacity to accommodate speaker-related information across varying hierarchies. Additionally, we



(a) Baseline

(b) With domain adapters

**Fig. 3**. t-SNE visualization of speaker embeddings, with distinct colors indicating various domains. The legend provides domain labels. Dotted line encloses the embeddings for different utterances from one speaker.

**Table 2**. Performance evaluation of different speaker verification systems in terms of EER and minDCF. The model with GRL describes the adversarial learning that uses GRL upon the speaker verification to perform domain classification. "Frozen" refers to freezing encoder blocks during fine-tuning.

| System | #Params | Frozen | Cross-Device | | Cross-Distance | | Cross-Dialect | |
|---|---|---|---|---|---|---|---|---|
| | | | EER(%) | minDCF | EER(%) | minDCF | EER(%) | minDCF |
| ResNet34 | 7.95M | N | 7.09 | 0.674 | 9.71 | 0.759 | 12.75 | 0.888 |
| + GRL [20] | 7.95M | N | 6.95 | 0.667 | 9.45 | 0.755 | 12.44 | 0.890 |
| + BDA-C | + 0.11M | Y | 6.56 | 0.653 | 8.66 | 0.740 | 10.65 | 0.843 |
| + BDA-F | + 0.02M | Y | 6.52 | 0.655 | 8.54 | 0.733 | 10.59 | 0.853 |
| + EDA | + 0.28M | Y | 6.46 | 0.649 | 8.43 | 0.739 | 10.48 | 0.847 |
| + EDA + BDA-C | + 0.39M | Y | 6.37 | 0.650 | 8.35 | 0.737 | 10.19 | **0.837** |
| + EDA + BDA-F | + 0.30M | Y | **6.32** | **0.643** | **8.27** | 0.730 | **10.06** | 0.840 |
| + EDA + BDA-C | + 0.39M | N | 6.52 | 0.662 | 8.62 | **0.726** | 10.39 | 0.852 |
| + EDA + BDA-F | + 0.30M | N | 6.57 | 0.664 | 8.54 | 0.732 | 10.26 | 0.841 |
| CAM++ [19, 21] | 7.26M | N | 7.75 | 0.723 | 11.29 | 0.783 | 13.44 | 0.886 |
| + EDA + BDA-C$^\dagger$ | + 0.93M | Y | 7.61 | 0.734 | 9.66 | 0.757 | 10.60 | 0.844 |
| ERes2Net [19, 22] | 9.91M | N | 7.21 | 0.678 | 10.18 | 0.757 | 12.52 | 0.886 |
| + EDA + BDA-F | + 0.30M | Y | 6.93 | 0.672 | 9.22 | 0.750 | 11.28 | 0.861 |

$^\dagger$: CAM++ is a model based on 1d convolution blocks and cannot support frequency-wise BDA.

assess the training strategy with both frozen and unfrozen encoder blocks. The results indicate that the system performs better with the frozen encoder block, demonstrating support for hot plugging in our proposed module.

Subsequently, we also conduct experiments on alternative baselines, namely CAM++ and ERes2Net. The experiment results show that proposed Domain Adapters also improve the performance on these models. This demonstrates that our method, in addition to being compatible with residual blocks, can be applied effectively to architectures featuring 1D-convolution TDNN blocks or multi-scale Res2Net blocks, which shows its versatility and robustness across various backbone structures.

To do a deeper insight into the influence from Domain Adapter, t-SNE visualizations for different systems are illustrated in Figure 3. Compared to the baseline, the Domain Adapter fosters proximity among utterances from the same speaker while increasing the distance between different speakers. Additionally, embeddings from the same domain exhibit increased compactness. All these observations demonstrate the advantage of the newly proposed domain adapter method.

### 3.3. Test with different domain labels

**Table 3**. Comparison of different domain labels in the testing stage.

| Test Label | EER(%) | | |
|---|---|---|---|
| | Cross-Device | Cross-Distance | Cross-Dialect |
| Ground Truth | 6.32 | 8.27 | 10.06 |
| Predicted (hard) | 6.97 | 8.44 | 11.13 |
| Predicted (soft) | 6.41 | 8.30 | 10.23 |

As discussed in sections 2.1, utilizing domain labels during the testing phase becomes important. In the initial experiment outlined in Table 2, systems were evaluated using ground truth labels to ensure consistency with the training phase. Assuming the absence of ground truth domain labels for the testing set, we generate their soft or hard labels using the pre-trained domain classifier. As shown in Table 3, utilizing predicted soft labels results in a negligible degradation compared to the ground truth labels, and the predicted hard labels will cause obvious performance degradation. This discrepancy can be attributed to the inherent uncertainty in domain classification, which is better accounted for by the soft labels.

**Table 4**. Performance comparison of systems with and without domain adapters on the out-of-domain trial.

| System | Test label | EER(%) | minDCF |
|---|---|---|---|
| ResNet34 | - | 6.19 | 0.627 |
| + GRL | - | 6.05 | 0.611 |
| + DAs | Predicted (soft) | 5.59 | 0.573 |
| CAM++ | - | 6.42 | 0.656 |
| + DAs | Predicted (soft) | 6.36 | 0.628 |
| ERes2Net | - | 6.27 | 0.620 |
| + DAs | Predicted (soft) | 6.03 | 0.598 |

### 3.4. Performance on unseen domains

Table 4 provides a comparison between systems with and without domain adapters on the "out-of-domain trial" as defined in 2.1. In this scenario, ground truth labels are absent, and all test domain labels are soft labels generated by the domain classifier. It is observed that across various backbone architectures, the domain adapters consistently demonstrate improvements in this trial, which underscores the proposed framework's effective capacity for generalization to unknown domains.

## 4. CONCLUSIONS

In this work, we propose the Domain Adapter, a versatile module that is domain-specific, highly adaptable, configurable at various levels and with different backbone architectures. During training, it independently extracts knowledge from each domain and merges it with speaker-related information, effectively mitigating domain-related disparities. Our framework significantly enhances performance with a minimal increase in model size and demonstrates effective generalization to previously unseen domains. The proposed best system achieves substantial EER improvements of 10.8%, 14.8%, and 21.1% over the baseline in three 3D-Speaker dataset trials.

## 5. ACKNOWLEDGEMENTS

This work was supported in part by China NSFC projects under Grants 62122050 and 62071288, in part by Shanghai Municipal Science and Technology Commission Project under Grant 2021SHZDZX0102, and in part by Shanghai Science and Technology Committee under Grant No. 21511101100. Shuai Wang is supported by Internal Project of Shenzhen Research Institute of Big Data under grant No.J00220230014.

## 6. REFERENCES

[1] Yuan Liu, Yanmin Qian, Nanxin Chen, Tianfan Fu, Ya Zhang, and Kai Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[2] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. ISCA Interspeech*, 2017, vol. 2017, pp. 999–1003.

[3] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *Proc. ICASSP*. IEEE, 2018, pp. 5329–5333.

[4] Yingke Zhu, Tom Ko, David Snyder, Brian Mak, and Daniel Povey, "Self-attentive speaker embeddings for text-independent speaker verification," in *Proc. ISCA Interspeech*, 2018, vol. 2018, pp. 3573–3577.

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. CVPR*, 2019, pp. 4690–4699.

[6] Xu Xiang, Shuai Wang, Houjun Huang, Yanmin Qian, and Kai Yu, "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1652–1656.

[7] Jixuan Wang, Kuan-Chieh Wang, Marc T Law, Frank Rudzicz, and Michael Brudno, "Centroid-based deep metric learning for speaker recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3652–3656.

[8] Chunlei Zhang and Kazuhito Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *Proc. ISCA Interspeech*, 2017, pp. 1487–1491.

[9] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, pp. 101027, 2020.

[10] Zhenyu Wang, Wei Xia, and John H.L Hansen, "Cross-domain adaptation with discrepancy minimization for text-independent forensic speaker verification," in *Proc. ISCA Interspeech*. 2020, pp. 2257–2261, ISCA.

[11] Kong Aik Lee, Qiongqiong Wang, and Takafumi Koshinaka, "The coral+ algorithm for unsupervised domain adaptation of plda," in *Proc. ICASSP*. 2019, pp. 5821–5825, IEEE.

[12] Qiongqiong Wang, Koji Okabe, Kong Aik Lee, and Takafumi Koshinaka, "A generalized framework for domain adaptation of plda in speaker recognition," in *Proc. ICASSP*. 2020, pp. 6619–6623, IEEE.

[13] Youzhi Tu, Man-Wai Mak, and Jen-Tzung Chien, "Variational domain adversarial learning for speaker verification," in *Proc. ISCA Interspeech*. 2019, pp. 4315–4319, ISCA.

[14] Gautam Bhattacharya, Joao Monteiro, Jahangir Alam, and Patrick Kenny, "Generative adversarial speaker embedding networks for domain robust end-to-end speaker verification," in *Proc. ICASSP*. IEEE, 2019, pp. 6226–6230.

[15] Zhengyang Chen, Shuai Wang, and Yanmin Qian, "Adversarial domain adaptation for speaker verification using partially shared network," in *Proc. ISCA Interspeech*, 2020, pp. 3017–3021.

[16] Zhengyang Chen, Shuai Wang, Yanmin Qian, and Kai Yu, "Channel invariant speaker embedding learning with joint multi-task and adversarial training," in *Proc. ICASSP*. IEEE, 2020, pp. 6574–6578.

[17] Xiaoyi Qin, Na Li, Chao Weng, Dan Su, and Ming Li, "Cross-age speaker verification: Learning age-invariant speaker embeddings," *arXiv preprint arXiv:2207.05929*, 2022.

[18] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang, "Cn-celeb: a challenging chinese speaker recognition dataset," in *Proc. ICASSP*. IEEE, 2020, pp. 7604–7608.

[19] Siqi Zheng, Luyao Cheng, Yafeng Chen, Hui Wang, and Qian Chen, "3d-speaker: A large-scale multi-device, multi-distance, and multi-dialect corpus for speech representation disentanglement," *arXiv preprint arXiv:2306.15354*, 2023.

[20] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. ICML*. PMLR, 2015, pp. 1180–1189.

[21] Hui Wang, Siqi Zheng, Yafeng Chen, Luyao Cheng, and Qian Chen, "Cam++: A fast and efficient network for speaker verification using context-aware masking," *arXiv preprint arXiv:2303.00332*, 2023.

[22] Yafeng Chen, Siqi Zheng, Hui Wang, Luyao Cheng, Qian Chen, and Jiajun Qi, "An enhanced res2net with local and global feature fusion for speaker verification," *arXiv preprint arXiv:2305.12838*, 2023.

[23] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[24] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP*. IEEE, 2017, pp. 5220–5224.

[25] Jenthe Thienpondt, Brecht Desplanques, and Kris Demuynck, "The idlab voxceleb speaker recognition challenge 2020 system description," *arXiv preprint arXiv:2010.12468*, 2020.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[27] Hossein Zeinali, Shuai Wang, Anna Silnova, Pavel Matějka, and Oldřich Plchot, "But system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.