Domain Adaptation Neural Network for Acoustic Scene Classification in Mismatched Conditions

Rui Wang, Mou Wang, Xiao-Lei Zhang and Susanto Rahardja
Center for Intelligent Acoustics and Immersive Communications,
School of Marine Science and Technology,
Northwestern Polytechnical University, Xi'an, China
Research & Development Institute of Northwestern Polytechnical University in Shenzhen, Shenzhen, China
E-mail: {wangrui2018, wangmou21}@mail.nwpu.edu.cn, {xiaolei.zhang, susanto}@nwpu.edu.cn

Abstract—**Acoustic scene classification is a task of predicting the acoustic environment of an audio recording. Because the training and test conditions in most real world acoustic scene classification problems do not match, it is strongly necessary to develop domain adaptation methods to solve the cross-domain problem. In this paper, we propose a domain adaptation neural network (DANN) based acoustic scene classification (ASC) method. Specifically, we first extract an acoustic feature, i.e. log-Mel spectrogram, which has been proven to be effective in previous studies. Then, we train a DANN to project the training and test domains into one common space where the acoustic scenes are categorized jointly. To boost the overall performance of the proposed method, we further train an ensemble of convolutional neural network (CNN) models with different parameter settings respectively. Finally, we fuse the DANN and CNN models by averaging the outputs of the models. We have evaluated the proposed method on the subtask B of task 1 of the DCASE 2019 ASC challenge, which is a closed-set classification problem whose audio recordings were recorded by mismatched devices. Experimental results demonstrate the effectiveness of the proposed method on the acoustic scene classification problem in mismatched conditions.**

I. Introduction

Acoustic scenes carry a large amount of information about surrounding circumstances and physical events [1]. Acoustic scene classification (ASC) [2], which aims to classify audio recordings into predefined acoustic scene classes, is important to many applications, such as robotic navigation [3], context-aware services [4], surveillance [5], etc. It has received much attention in recent years. For example, detection and classification of acoustic scenes and events (DCASE) hosted by IEEE audio and acoustic signal processing is a series of recent challenges, and also one of the first large-scale challenges of ASC research [6]. It is known that the audio recordings collected by different devices usually have some mismatches, since many channel distortions, such as the differences of microphone arrays, sampling rates, and circuit designs between the devices, introduce interruptions to the recordings. Therefore, how to improve the classification performance when training and test data are recorded in mismatched conditions is one of the most challenging problems of ASC.

Domain adaptation is a good choice to reduce the negative effect caused by the domain mismatch. It usually partitions the entire data space into a source domain and a target domain, where the mismatch problem exists between the two domains. Most often, the target domain consists of the test data and sometimes part of the training data. The main purpose of domain adaptation is to transfer the knowledge of the source domain to the target domain, so as to improve the classification accuracy on the target domain. Domain adaptation techniques can be generally categorized into two groups: supervised methods or unsupervised methods, based on whether the target domain has manually labeled data.

Supervised domain adaptation assumes that labeled data are available in the target domain. The labeled data in the target domain are used to modify the classifier trained on the source domain. Traditional classifiers include support vector machine (SVM) [7], [8] and boosted decision tree [9]. Recently, deep neural networks have also been applied successfully [10], [11], which use the labeled data in the target domain to fune-tune the neural networks. Because the labeled data from the target domain are not always available, unsupervised domain adaptation has been largely investigated.

Unsupervised domain adaptation works for the scenarios where no labeled data is available in the target domain. It can be divided into the following four classes. The first class first estimates the labels of the target data by some clustering methods [12]–[14], and then uses the estimated data as part of the source data to train the classifiers. The second class trains compensation models with both the unlabeled target data and the labeled source data to compensate the domain mismatch [15]–[17]. The third class learns a mapping function to project the target data to the source domain, so that the models trained on the source domain can be applied to classify the target data directly without suffering much performance drop [18]. The fourth class first learns a common subspace shared by both the source and target domains, and then conducts model training and test in the subspace [19]. Empirically, the fourth class usually yields good performance with an expense of more complicated algorithm designs than the other approaches.

In this paper, we propose a supervised domain adaptation neural network (DANN) approach for ASC. It first uses DANN to learn a common subspace shared by both the source and target domains with the knowledge of some labeled data

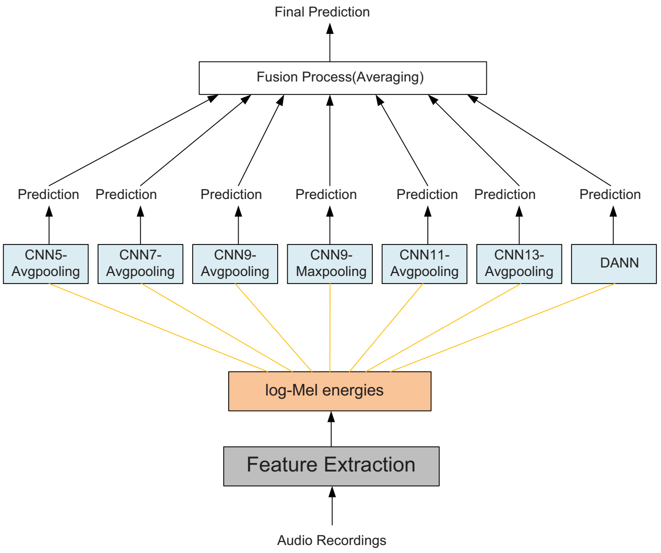

Fig. 1. Diagram of the proposed DANN-based ASC method.

available in the target domain, then it combines the decision scores produced by DANN with the scores produced by an ensemble of convolutional neural network (CNN) models trained on both domains for the final prediction. The approach has two novelties. First, we introduce DANN to learn a common subspace for the ASC problem. Second, although the approach is supervised, it does not fall into existing supervised domain adaption framework of ASC. It is a combination of the advantages of supervised domain adaptation and the fourth class of unsupervised domain adaptation. We evaluated the effectiveness of the DANN-based ASC on the subtask B of task 1 of the DCASE 2019 ASC challenge, where the subtask is a closed-set classification problem with its training and test audio recordings recorded by mismatched devices. Kong *et al.* [20] has proposed a generic CNN-based cross-task baseline system, from which we were inspired to adopt such an ensemble of CNN-based models.

This paper is organized as follows. Section II presents the framework of the proposed method. Section III presents experiments. Section IV concludes this paper.

II. Proposed Method

In this section, we first overview the proposed method in Section II-A, and then present its components in detail in Sections II-B, II-C, and II-D, respectively.

A. System Overview

Figure 1 shows the diagram of the proposed method. Specifically, we first extract 64-dimensional or 128-dimensional log-Mel energies from the original audio recordings. Then, we use the acoustic feature as the input of DANN and 6 CNNs. The 6 CNNs are named CNN5-Avgpooling, CNN7-Avgpooling, CNN9-Avgpooling, CNN9-Maxpooling, CNN11-Avgpooling, CNN13-Avgpooling, where the number after the term "CNN" means the number of the hidden layers of CNN, and "Avgpooling" and "Maxpooling" are two kinds of pooling layers of CNN. Finally, we average the outputs of the DANN and the CNN ensemble for the final prediction.

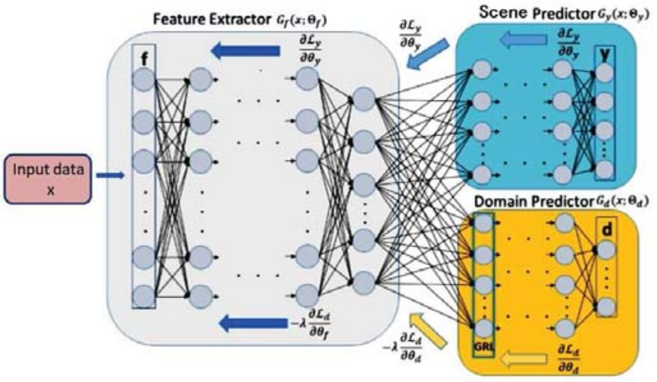
Feature Extractor $G_f(x;\theta_f)$
Scene Predictor $G_y(x;\theta_y)$
Domain Predictor $G_d(x;\theta_d)$
Input data
x
f
y
d
GRL
$\frac{\partial L_y}{\partial \theta_y}$
$\frac{\partial L_y}{\partial \theta_y}$
$\frac{\partial L_y}{\partial \theta_y}$
$-\lambda\frac{\partial L_d}{\partial \theta_f}$
$-\lambda\frac{\partial L_d}{\partial \theta_d}$
$\frac{\partial L_d}{\partial \theta_d}$

Fig. 2. Domain adaptation neural networks.

B. Feature Extraction

The log-Mel energies have been proven to be one of the most suitable acoustic features for ASC [21]. They are extracted as follows. First, we extract a spectrogram from an audio recording by the Hamming window reweighted short-time Fourier transforms, where the frame lengths of the spectrograms are set to 32 or 64 milliseconds respectively with the frame shifts both set to 15 milliseconds. Then, we use 64 or 128 Mel-filter banks respectively to transform the spectrogram to their corresponding Mel-energies. Finally, the log-Mel energy features are obtained by applying the logarithm operator to the Mel-energies.

C. Domain Adaptation Neural Networks

As shown in Fig. 2, DANN projects different domains into one common subsapce for mitigating the domain mismatch problem. It contains three parts: a feature extractor, a scene predictor, and a domain predictor. The feature extractor aims to project acoustic features from different domains into one subspace where the features are scene-discriminative and domain-invariant. The scene predictor aims to classify the segment-level features into predefined scenes. The domain predictor aims to discriminate whether the input audio recording is collected from the source domain or not. Although the structure of DANN, whose output layer contains two separate parts, is similar to that of multitask neural networks, their training processes are fundamentally different. Its principle is to learn a feature representation that is both discriminative and domain-invariant by adversarial training. To achieve this goal, DANN should try to improve the performance of the scene predictor and meanwhile fool the domain predictor. Here we present the training process of DANN in detail as follows.

Suppose a training corpus consists of a source domain and a target domain. A training audio recording from the corpus consists of n frames $[\{\mathbf{x}_i\}_{i=1}^n, \mathbf{y}, \mathbf{d}]$ where $\mathbf{x}_i$ is the input feature of the i-th frame, $\mathbf{y}$ is the ground-truth label of the audio recording, and $\mathbf{d}$ indicates the domain of the audio recording. $\mathbf{y}$ is a one-hot code whose activated position denotes the acoustic scene of the audio recording. If the recording is from the source domain, then $\mathbf{d} = [0, 1]^T$; otherwise, $\mathbf{d} = [1, 0]^T$.

First, the feature extractor learns a mapping function from $\{\mathbf{x}_i\}_{i=1}^n$ to an F-dimensional vector $\mathbf{f}$, which transforms the audio recording to a new representation $[\mathbf{f}, \mathbf{y}, \mathbf{d}]$. Then, the scene predictor learns a mapping function from $\mathbf{f}$ to $\mathbf{y}$. The domain predictor learns a mapping function from $\mathbf{f}$ to $\mathbf{d}$. The feed-forward process are summarized as follows:

$$\mathbf{f} = G_{\theta_\mathbf{f}}(\{\mathbf{x}\}) \tag{1}$$

$$\mathbf{y} = G_{\theta_\mathbf{y}}(\mathbf{f}) \tag{2}$$

$$\mathbf{d} = G_{\theta_\mathbf{d}}(\mathbf{d}) \tag{3}$$

where $\theta_\mathbf{u}$ with $\mathbf{u} \in \{\mathbf{f}, \mathbf{y}, \mathbf{d}\}$ are the parameters of the three components of DANN respectively.

We train the three components jointly and take the output of $G_{\theta_\mathbf{f}}(\cdot)$ as the final output of DANN. The training objective of DANN minimizes the scene classification loss and meanwhile maximizes the domain classification loss. To maximize the domain classification loss, a gradient reversal layer is inserted between the feature extractor and the domain predictor, which passes negative gradients from the domain predictor to the feature extractor. It helps find a saddle point between the two components [19]. If we view the domain predictor as a regularizer, then a common way of balancing the scene predictor and the domain predictor is to introduce a positive hyperparameter $\lambda > 0$ to the domain predictor. Formally, we define the loss function of DANN as:

$$\begin{aligned} E(\theta_\mathbf{f}, \theta_\mathbf{y}, \theta_\mathbf{d}) = L_\mathbf{y}(G_{\theta_\mathbf{y}}(G_{\theta_\mathbf{f}}(\{\mathbf{x}_i\})), \mathbf{y}) - \\ \lambda L_\mathbf{d}(G_{\theta_\mathbf{d}}(G_{\theta_\mathbf{f}}(\{\mathbf{x}_i\})), \mathbf{d}) \end{aligned} \tag{4}$$

where $L_a(\hat{a}, a)$ with $a \in \{\mathbf{y}, \mathbf{d}\}$ evaluates the loss between the ground-truth a and its prediction $\hat{a}$. In this paper, we set $L_\mathbf{y}(\hat{\mathbf{y}}, \mathbf{y})$ to the cross-entropy loss, and set $L_\mathbf{d}(\hat{\mathbf{d}}, \mathbf{d})$ to the mean squared error loss. It is easy to see that minimizing (4) equals to an optimization problem of minimizing $L_\mathbf{y}$ and maximizing $L_\mathbf{d}$ jointly. We use the adaptive moment estimation approach [22] to optimize the network. After training, the feature representation produced by $G_{\theta_\mathbf{f}}(\cdot)$ yields a high scene classification accuracy and meanwhile is domain-insensitive.

D. Convolutional Neural Networks

We follow the parameter setting of [20] to build our CNN models. The parameter setting is illustrated in Table I. Specifically, each CNN consists of multiple convolution blocks and a fully connected softmax output layer except CNN5-Avgpooling. Each convolution block contains two cascaded convolution layers, each of which consists of a 3×3 kernel, a batch normalization operator, and rectified linear units successively. Average pooling or max-pooling with a size of 2×2 or 1×1 is adopted after the convolution operation. Different with other CNN, CNN5-Avgpooling has four convolution layers, each of which consists of a 5×5 kernel, a batch normalization operator, and rectified linear units successively. We train the CNN models on the source data. We use cross-entropy loss as the training criterion.

TABLE I
THE STRUCTURE AND PARAMETER SETTING OF CNN9-AVGPOOLING.

Input	Acoustic features
Convolution block	3x3 Convolution 64-BN-ReLU 3x3 Convolution 64-BN-ReLU 2x2 Average pooling
Convolution block	3x3 Convolution 128-BN-ReLU 3x3 Convolution 128-BN-ReLU 2x2 Average pooling
Convolution block	3x3 Convolution 256-BN-ReLU 3x3 Convolution 256-BN-ReLU 2x2 Average pooling
Convolution block	3x3 Convolution 512-BN-ReLU 3x3 Convolution 512-BN-ReLU 1x1 Average pooling
Output layer	Dense-10-softmax

III. EXPERIMENTS

In this section, we first present the experimental settings in Section III-A, then present the main results in Section III-B, and finally show the effects of the hyperparameters of DANN on performance in Section III-C.

A. Experimental Settings

1) Datasets: We evaluate the effectiveness of our method on the subtask B of task 1 of DCASE 2019. The task adopts the TAU Urban Acoustic Scenes 2019 Mobile dataset [1]. It contains 10 acoustic scenes, including airport, shopping mall, metro station, pedestrian street, public square, street with traffic, tram, bus, metro and urban park. Its audio recordings were recorded by four devices, denoted as devices A, B, C, and D, where the data from device D only appears in the final evaluation and is not public at the time of this paper. The public data consists of a predefined development dataset and an open leaderboard dataset, both of which were recorded by devices A, B, and C. We regard the data from device A as the source data and the data from devices B and C as the target data. The development set, which was collected from 10 European cities, contains 16560 audio segments with a total time of 46 hours, of which 14400 segments with a time of 40 hours were recorded by device A, 1080 segments with a time of 3 hours recorded by device B, and 1080 segments with a time of 3 hours recorded by device C. The leaderboard set were collected from 12 cities.

For the development set, we selected 70% data from device A and 50% data from devices B and C as a training subset, and set the remaining data as a validation subset except 1030 segments unused. We first trained the models of all comparison methods on the training set and picked the best parameter settings of the models on the validation subset. Finally, we retrain all models with the best parameter settings on the entire development set and evaluated the models on the leaderboard dataset.

TABLE II
THE STRUCTURE AND PARAMETER SETTING OF DANN

Feature extractor	3x3 Convolution 64-BN-ReLU 3x3 Convolution 64-BN-ReLU 2x2 Average pooling
	3x3 Convolution 128-BN-ReLU 3x3 Convolution 128-BN-ReLU 2x2 Average pooling
	3x3 Convolution 256-BN-ReLU 3x3 Convolution 256-BN-ReLU 1x1 Average pooling
Scene predictor	Fully connected (dim-256)-BN-ReLU Fully connected (dim-100)-BN-ReLU 10-way softmax
Domain predictor	Fully connected (dim-256)-BN-ReLU 1-way softmax

2) Parameter Settings: As shown in Fig. 1. The proposed method trained 2 DANN and 12 CNN models, of which 1 DANN and 6 CNN models take the 64-dimensional log-Mel energies as their input feature, and the others take the 128-dimensional log-Mel energies as the input. The two groups of models have the same parameter setting. Here we present the parameter setting of one group only as follows. Table II lists the parameter setting of DANN. The feature extractor contains three convolutional blocks. The scene predictor contains fully connected hidden layers with the number of hidden units set to 256 and 100 respectively. The domain predictor contains a fully connected hidden layer with the number of hidden units set to 256. Table I shows the parameter setting of a CNN with four convolutional blocks and average pooling layer. The other 5 CNN models in this group have a similar structure with that in Table I, but different number of convolutional layers and different kinds of pooling layers from the latter which has been described in Fig. 1

We take the official CNN-based baseline as our comparison method [1]. The evaluation criterion is classification accuracy (ACC), which is obtained by averaging the class-wise accuracies of all acoustic scene classes.

B. Main Results

Table III shows the ACC comparison of the comparison methods and their components on the validation and leaderboard sets, where the ACC of the baseline was provided by the DACSE 2019 ASC Challenge [1]. From the table, we see that a single DANN achieves 0.153 and 0.132 absolute ACC improvements over the CNN-based baseline on the validation and leaderboard sets respectively; we also see that the performance of a single DANN is only slightly worse than a CNN ensemble, which proves the effectiveness of DANN in mismatched conditions. We also observe that, if the CNN ensemble and DANN use the same input acoustic feature, then the aggregation of the CNN ensemble and DANN always improve the performance over the CNN ensemble alone on the validation set, which manifests that the CNN ensemble and DANN complement each other.

TABLE III
CLASSIFICATION ACCURACIES (ACC) ON THE VALIDATION AND LEADERBOARD SETS. THE TERMS "64MEL" AND "128MEL" DENOTE THE 64- AND 128-DIMENSIONAL LOG-MEL ENERGIES RESPECTIVELY.

Methods	Validation set	Leaderboard set
Baseline	0.414	0.480
64Mel+CNN ensemble	0.563	0.665
64Mel+DANN	0.544	0.593
64Mel+CNN ensemble+DANN	0.583	0.675
128Mel+CNN ensemble	0.598	0.715
128Mel+DANN	0.567	0.612
128Mel+CNN ensemble+DANN	0.606	0.707

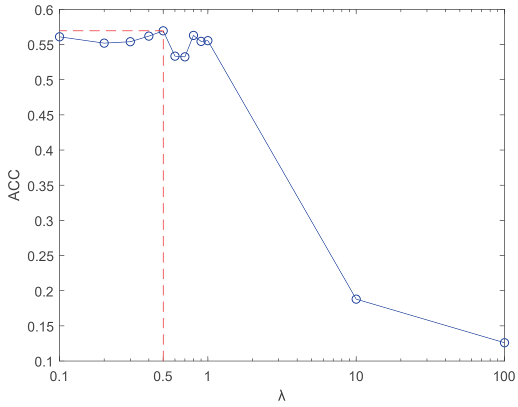

Fig. 3. Classification accuracy of the DANN with different λ on the validation set.

C. Effect of Hyperparameter λ of DANN

Hyperparameter λ balances the training accuracy and domain invariance. To investigate the effect of λ, we took the 128-dimensional log-mel energies as the input feature, and searched λ in grid from 0.1 to 1.0 with an interval of 0.1, as well as two other values—10 and 100. The experimental result on the validation set is shown in Fig. 3. From the figure, we see that the proposed method achieves the highest ACC when $\lambda = 0.5$, and the robust working region of λ is also around 0.5.

IV. CONCLUSION

In this paper, we have proposed the supervised domain adaptation neural network for the acoustic scene classification problem in mismatched conditions. DANN minimizes the classification error and meanwhile reduces the network capacity on discriminating the source data from the target data by adversarial training. The proposed DANN-based ASC contains two components. The first component is DANN. The second component is the CNN ensemble trained without domain adaptation. The two components complement each other. The approach has two novelties. First, DANN is introduced to the ASC problem. Second, the approach brings the idea of the fourth class of the unsupervised domain adaptation techniques to the supervised domain adaptation. We have evaluated the effectiveness of the DANN-based ASC on the subtask B of task 1 of DCASE 2019. Experimental results demonstrate that DANN is able to learn a domain-invariant feature representation. It is not only much powerful than the CNN model without domain adaptation when used alone, but also complementary to the CNN ensemble when used together with the CNN ensemble. The DANN-based ASC achieves over 20% higher ACC than the comparison baseline.

ACKNOWLEDGMENT

This paper was supported in part by the Shenzhen Science and Technology Plan under grant No. JCYJ20170815161820095, in part by the National Natural Science Foundation of China under grant No. 61671381, and in part by the Shaanxi Natural Science Basic Research Program under grant No. 2018JM6035.

REFERENCES

[1] http://dcase.community/challenge2019/.

[2] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," *arXiv preprint arXiv:1807.09840*, 2018.

[3] S. Chu, S. Narayanan, C. . J. Kuo, and M. J. Mataric, "Where am i? scene recognition for mobile robots using audio features," in *2006 IEEE International Conference on Multimedia and Expo*, July 2006, pp. 885–888.

[4] A. J. Eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, 2005.

[5] R. Radhakrishnan, A. Divakaran, and A. Smaragdis, "Audio analysis for surveillance applications," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005.*, Oct 2005, pp. 158–161.

[6] T. Heittola and A. Mesaros, "DCASE 2017 challenge setup: Tasks, datasets and baseline system," DCASE2017 Challenge, Tech. Rep., September 2017.

[7] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer svm for video concept detection," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1375–1381.

[8] A. Bergamo and L. Torresani, "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach," in *Advances in neural information processing systems*, 2010, pp. 181–189.

[9] C. J. Becker, C. M. Christoudias, and P. Fua, "Non-linear domain adaptation with boosting," in *Advances in Neural Information Processing Systems*, 2013, pp. 485–493.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[11] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1717–1724.

[12] D. Garcia-Romero, A. McCree, S. Shum, N. Brummer, and C. Vaquero, "Unsupervised domain adaptation for i-vector speaker recognition," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, 2014.

[13] D. Garcia-Romero, X. Zhang, A. McCree, and D. Povey, "Improving speaker recognition performance in the domain adaptation challenge using deep neural networks," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 378–383.

[14] S. H. Shum, D. A. Reynolds, D. Garcia-Romero, and A. McCree, "Unsupervised clustering approaches for domain adaptation in speaker recognition systems," 2014.

[15] A. Kanagasundaram, D. Dean, and S. Sridharan, "Improving out-domain plda speaker verification using unsupervised inter-dataset variability compensation approach," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4654–4658.

[16] H. Aronowitz, "Inter dataset variability compensation for speaker recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4002–4006.

[17] M. H. Rahman, A. Kanagasundaram, D. Dean, and S. Sridharan, "Dataset-invariant covariance normalization for out-domain plda speaker verification," in *Proceedings of the 16th Annual Conference of the International Speech Communication Association, Interspeech 2015*. International Speech Communication Association, 2015, pp. 1017–1021.

[18] S. Shon, S. Mun, W. Kim, and H. Ko, "Autoencoder based domain adaptation for speaker recognition under insufficient channel information," *arXiv preprint arXiv:1708.01227*, 2017.

[19] Q. Wang, W. Rao, S. Sun, L. Xie, E. S. Chng, and H. Li, "Unsupervised domain adaptation via domain adversarial training for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4889–4893.

[20] Q. Kong, Y. Cao, T. Iqbal, Y. Xu, W. Wang, and M. D. Plumbley, "Cross-task learning for audio tagging, sound event detection and spatial localization: Dcase 2019 baseline systems," *arXiv preprint arXiv:1904.03476*, 2019.

[21] Y. Sakashita and M. Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," *Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge*, 2018.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.