ACOUSTIC SCENE CLASSIFICATION BASED ON A LARGE-MARGIN FACTORIZED CNN

Janghoon Cho, Sungrack Yun, Hyoungwoo Park, Jungyun Eum, Kyuwoong Hwang

Qualcomm AI Research*, Qualcomm Korea YH
343, Hakdong-ro, Gangnam-gu, Seoul, Korea
{janghoon, sungrack, c_hyoupa, c_jeum, kyuwoong}@qti.qualcomm.com

ABSTRACT

In this paper, we present an acoustic scene classification framework based on a large-margin factorized convolutional neural network (CNN). We adopt the factorized CNN to learn the patterns in the time-frequency domain by factorizing the 2D kernel into two separate 1D kernels. The factorized kernel leads to learn the main component of two patterns: the long-term ambient and short-term event sounds which are the key patterns of the audio scene classification. In training our model, we consider the loss function based on the triplet sampling such that the same audio scene samples from different environments are minimized, and simultaneously the different audio scene samples are maximized. With this loss function, the samples from the same audio scene are clustered independently of the environment, and thus we can get the classifier with better generalization ability in an unseen environment. We evaluated our audio scene classification framework using the dataset of the DCASE challenge 2019 task1A. Experimental results show that the proposed algorithm improves the performance of the baseline network and reduces the number of parameters to one third. Furthermore, the performance gain is higher on unseen data, and it shows that the proposed algorithm has better generalization ability.

Index Terms— Acoustic scene classification, Factorized convolutional neural network, Triplet sampling

1. INTRODUCTION

The interest of acoustic scene classification (ASC) has been continuously increasing in the last few years and is becoming an important research in the fields of acoustic signal processing. The ASC aims to identify different environments given the sounds they produce [1] and has various applications in context-awareness and surveillance [2, 3, 4]: e.g. the device which recognizes the environmental sound by analyzing the surrounded audio information. With the release of large scale datasets and challenge tasks by Detection and Classification of Acoustic Scenes and Events (DCASE) [5, 6], the ASC has become a very popular research topic in audio signal processing. We have an increasing number of research centers, companies, and universities participating in the DCASE challenge and workshop every year.

In the past decade, deep learning has accomplished many achievements in audio, image, and natural language processing. Especially, the algorithms based on the convolutional neural network (CNN) are dominant in ASC [7, 1, 8, 9] tasks. In [2], a simple CNN with 2 layers was adopted, and many attempts were tried to solve the overfitting problem in improving the ASC performance by increasing the model complexity: e.g. number of layers in CNN. In [7], SubSpectralNet method was introduced to capture more enhanced features with a convolutional layer by splitting the time-frequency features into sub-spectrograms. In [8], a simple preprocessing method was adopted to emphasize the different aspects of the acoustic scene. In [10, 11, 12], an ensemble of various acoustic features such as MFCC, HPSS, i-vector and the technique that independently learns the classifiers of each feature was proposed. However, the method is heuristic and requires a lot of computation in the front-end step before CNN inference step.

In order to solve the problems of the above-mentioned conventional methods, we propose an algorithm that uses only one feature and does not increase the model complexity of CNN. The pattern of each acoustic scene in the time-frequency domain can be represented as a low-rank matrix, thus we consider designing a 2D convolution layer as two consecutive convolution layers with 1D kernels. Also, we consider a loss function such that the samples from the same audio scene are clustered independently of the environment to be robust to unseen environment. In short, our proposed ASC framework is based on the Sub-Spectral Net [7], and the kernel factorization and loss function are applied to reduce the computational complexity and increase the generalization ability on unseen environment. We evaluated our ASC framework using the dataset of DCASE task1A, and all experimental results show that the proposed algorithm with the data augmentation techniques [13, 14] significantly improves the accuracy.

The rest of the paper is organized as follows. Section 2 formulates the problem of ASC. Section 3 describes the proposed algorithm including our factorized CNN structure and novel loss function. Section 4 presents the experimental results and analysis. Finally, Section 5 concludes.

2. ACOUSTIC SCENE CLASSIFICATION

Audio scene classification, the task1A of the DCASE 2019 challenge [6], is a process of predicting a label y^* given an input audio clip $\mathbf{x}$ as:

$$y^* = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{f}_\mathbf{x}; \theta) \quad (1)$$

where $p(y|\mathbf{f}_\mathbf{x}; \theta)$ is the audio scene posterior given the feature map $\mathbf{f}_\mathbf{x}$ with the network parameter θ, and $\mathcal{Y}$ is the entire set of scene labels. The input audio clip $\mathbf{x}$ contains only one audio scene, and the feature map of $\mathbf{x}$, $\mathbf{f}_\mathbf{x}$, can be obtained using various algorithms such as deep audio embeddings [15, 16], log-mel [17, 18, 10, 12], amplitude modulation filter bank [19, 20], and perceptual weighted power spectrogram [11]. In this paper, we use the 40 log-mel, $\mathbf{f}_\mathbf{x} \in \mathrm{R}^a$, since recently many approaches [10, 12] adopt the log-mel feature and show good performances in audio scene classification task. The

*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

https://doi.org/10.33682/8xh4-jm46

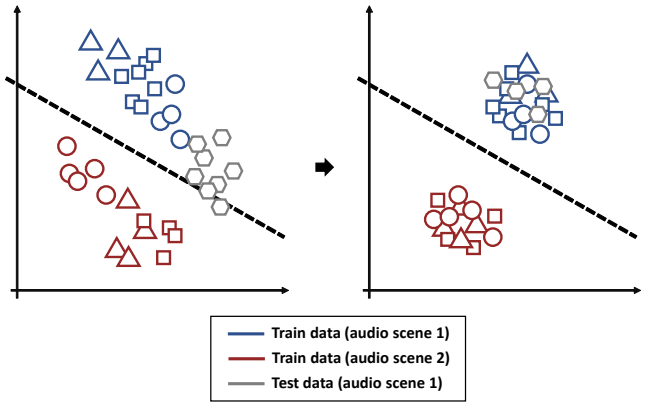

Figure 1: Examples of two audio scene samples (red, blue) from different cities (Triangle, rectangle, circle, hexagon). Given an audio scene, the audio scene from the same city are more clustered than from different cities. For the better generalization ability on unseen city, we apply a loss such that the samples are clustered independently of the cities.

posterior $p(y|\mathbf{f_x};\theta)$ is the probabilistic score of the audio scene label y using the softmax:

$$p(y|\mathbf{f_x};\theta) = \frac{\exp(M_y(\mathbf{f_x}))}{\sum_{v\in\mathcal{Y}}\exp(M_v(\mathbf{f_x}))} \quad (2)$$

where $M_{v\in\mathcal{Y}}(\mathbf{f_x})$ is the output of $\mathbf{f_x}$ obtained from the final layer of the audio scene classification network M.

The dataset of task1A consists of the audio scene samples recorded in a number of cities: i.e. samples of an audio scene (e.g. airport) were recorded in a number of locations (e.g. Amsterdam, London, Helsinki). In real applications, the audio scene classifier can be trained to classify N audio scenes using the dataset recorded in a limited number of cities, and the classifier may be deployed to the test environments which are unseen cities in the training dataset. In such cases, the test samples can be misclassified since the classification boundary may not accurately separate the audio scene samples from unseen cities. This is illustrated in Fig. 1. In this example, there are two audio scene feature points (red, blue) from three different cities (triangle, circle, rectangle). And, given an audio scene, it's highly probable that the samples from the same city are more clustered than from different cities. The black dashed line is the classification boundary trained with the audio scene data from three different cities. Here, we may have the test samples from unseen city (gray hexagon) which can be presented in the other region near the classification boundary. In this case, some of the test samples will be misclassified. In contrast to the left-side of the Fig. 1, if the audio scene features are clustered independently of the cities, and the within-class distances are minimized, then we can have the classifier with more generalization ability especially when there are audio scene samples from unseen cities as shown in the right-side of the Fig. 1.

3. PROPOSED ALGORITHM

The block diagram of our proposed algorithm is illustrated in Fig. 2. The overall structure is based on Sub-Spectral Net [7] which assumes that the key patterns of each class are concentrated on

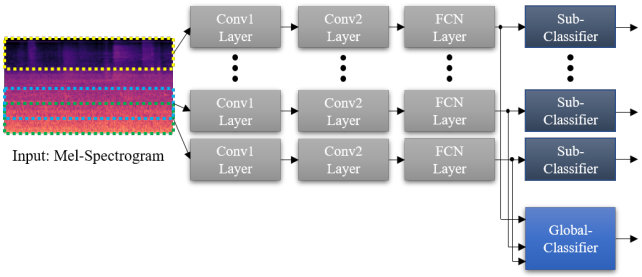

Figure 2: Block diagram of proposed acoustic scene classification algorithm.

the specific frequency bins, and those key patterns are effectively learned by dividing the input mel-spectrogram into the multiple sub-bands and using the CNN classifier for each sub-band independently. By dividing the input mel-spectrogram into the multiple sub-bands and learning the CNN classifier of each sub-band independently, the key patterns of specific frequency bins are effectively learned. The CNN classifiers consist of two convolution layers and one fully connected network as in the baseline network of DCASE 2019. The outputs of all FCN layers are concatenated, and they are used as the input of the global classifier.

3.1. Factorized CNN

Recently, the CNN-based algorithms have shown great performance and the state-of-the-art result in various areas such as image classification, audio/speech processing, speech recognition, and speaker verification [21, 22]. As CNN-based algorithms are also a popular trend in the ASC task, most of the algorithms submitted to the DCASE 2018 are based on CNN [10, 11, 12]. The mel-spectrogram feature of audio data can be regarded as an image, and the CNN-based algorithm can be used to recognize the audio characteristics such as the phoneme and human voices.

When we train a CNN-based model for acoustic scene classification, the over-fitting problem can be occurred due to the limitation of the data even when we use simple 2-layer CNN structure is used, while learned convolution filters had a noisy pattern that was difficult to analyze. The CNN model tends to memorize all training audio scenes including the noise components which do not help classification, and it may cause the poor generalization performance.

To resolve the above-mentioned problem, we propose the factorized CNN based on low-rank matrix factorization. Low-rank matrix factorization is also widely used technique in audio signal separation [23, 24], and it is based on that the mel-spectrogram of audio signal can be represented as the summation of a small number of rank 1 matrices. This leads to classify the acoustic scene with a small number of parameters for the ASC network.

In the classification of acoustic scenes, we can consider the following two audio elements: one is the ambient signal over a long period of time, and the other is an event signal with short period such as bird sound in a park and car horn in the road. As shown in Fig. 3, the ambient signals for each acoustic scene show faint stripes of horizontal lines in mel-spectrogram due to the statistical stationarity over time. Also, many audio examples for this kind of event signals show the pattern with the horizontal stripe, and this can be represented as a rank-1 matrix.

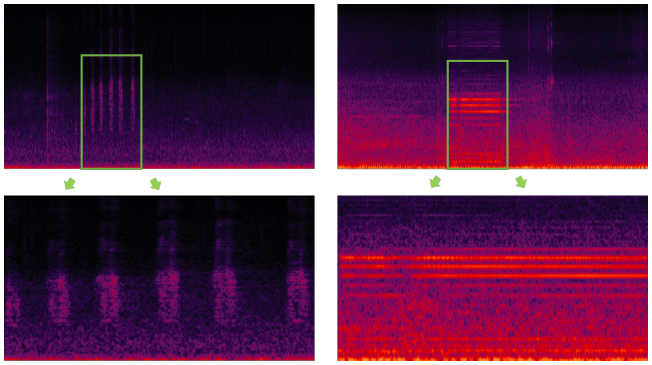

Figure 3: The mel-spectrograms of the sound of a bird in a park and the horn of a car. (Top left) Mel-spectrogram of "park-stockholm-102-2895-a.wav" in the development dataset (Bottom left) Zoom-up of top left (Top right) Mel-spectrogram of "street_traffic-london-271-8255-a.wav" (Bottom right) Zoom-up of top right

The detail specification of each layer is described in Figure 4. Rather than using a single conv2D layer which has rectangular (k, k) kernel, we use two consecutive conv2D layers. These two conv2D layers have 1 dimensional kernels $(k, 1)$ and $(1, k)$. When the rectangular (k, k) kernel is a rank-1 matrix and can be factorized into $(k, 1)$ and $(1, k)$ vectors, conv2D with (k, k) kernel becomes equivalent to conv2D with $(k, 1)$ and conv2D with $(1, k)$. So, this network is equivalent to conv2D layer with a rank-1 matrix kernel. With the convolution kernel of rank-1 matrix, we can reduce the over-fitting of learning noisy patterns from training data. Also, the number of model parameters can be reduced since two 1-dim kernels need $2k$ while the square kernel needs k^2 parameters.the square kernel needs k^2 parameters.

The factorized CNN originally proposed in this paper is different from the network in [25] where it factorized the 3D convolution layer as a single intra-channel convolution and a linear channel projection.

3.2. Large-margin loss function

As the acoustic scene classification is the multi-class classification problem, which maximizes the probability of the correct label and minimizes that of all the others, most of the algorithms are using the cross-entropy loss function. The cross entropy between the true label y and recognized output $\hat{y}$ is given as

$$L_{\text{CE}} = -\sum_{y} y \log(\hat{y}). \tag{3}$$

However, the cross-entropy loss only focuses on fitting or classifying the training data accurately; it does not explicitly encourage a large decision margin for classification [26]. Even when training the simple CNN classifier which has only 2 layers with the cross entropy loss, we can observe that the training loss converges to zero, while the test loss does not converge.

In this context, we consider a loss function similar to the triplet loss but slightly different. Triplet loss function is widely used loss function in many machine learning tasks such as person re-identification, face clustering, and speaker embedding [27, 22]. It enforces the positive pairs to be closer, and the negative pairs to be further and can be expressed as

Layer	Composition
Conv1	$\text{Conv2D}\left(C^1_{in}, C^1_{out}, \text{kernel size} = (k,1), \text{stride} = 1, \text{padding} = ((k-1)/2, 0)\right)$
	$\text{Conv2D}\left(C^1_{out}, C^1_{out}, \text{kernel size} = (1,k), \text{stride} = 1, \text{padding} = (0,(k-1)/2), \text{groups} = C^1_{out}\right)$
	$\text{BatchNorm}(C^1_{out})$
	$\text{Activation}('relu')$
	$\text{MaxPool2D}(Y/10, 5)$
	$\text{DropOut}(0.3)$
Conv2	$\text{Conv2D}\left(C^2_{in}, C^2_{out}, \text{kernel size} = (k,1), \text{stride} = 1, \text{padding} = ((k-1)/2, 0)\right)$
	$\text{Conv2D}\left(C^2_{out}, C^2_{out}, \text{kernel size} = (1,k), \text{stride} = 1, \text{padding} = (0,(k-1)/2), \text{groups} = C^2_{out}\right)$
	$\text{BatchNorm}(C^2_{out})$
	$\text{Activation}('relu')$
	$\text{MaxPool2D}(4,100)$
	$\text{DropOut}(0.3)$
FCN	$\text{Dense}(2 \times C^2_{out}, 32, \text{activation} = 'relu')$
Sub-Classifier	$\text{DropOut}(0.3)$
	$\text{Dense}(32, 10, \text{activation} = 'softmax')$
Global-Classifier	Multi Layer Perceptron

Figure 4: The detail specifications of each block.

$$L_{\text{triplet}} = \max\left(\|\mathbf{x}_a - \mathbf{x}_p\|^2 - \|\mathbf{x}_a - \mathbf{x}_n\|^2 + \alpha, 0\right), \tag{4}$$

where $\mathbf{x}$ indicates the embedding vector which is the output of the final convolution layer before entering the fully connected layer, and $\mathbf{x}_a$, $\mathbf{x}_p$, and $\mathbf{x}_n$ are anchor, positive, and negative embedding vectors, respectively. α indicates the triplet loss margin parameter. The anchor and positive pair should come from the same class, and the anchor and negative pair should come from the different classes. In our case, we modified the sampling: we choose the positive sample from the same class but different environment (city) to cluster the samples independently of the environment as shown in Fig. 1. Finally, we combine the cross-entropy and triplet losses as

$$L_{\text{ASC}} = L_{\text{CE}} + \gamma L_{\text{triplet}} \tag{5}$$

where γ is the hyper-parameter which should be tuned.

To reduce the triplet loss effectively, we should choose a distant anchor-positive pair and a close anchor-negative pair, however, it is very inefficient to figure out the distance of all pairs for choosing efficient pairs. Fortunately, the additional label which describes the city of each acoustic scene is given in DCASE 2019 dataset. The sound signals of the same acoustic scene and city is more similar than that of the same acoustic scene and different city, therefore we choose all anchor-positive pairs from the same acoustic scene and different city and all anchor-negative pairs randomly.

3.3. Data augmentation

Since the number of training data is limited in the development dataset, it is necessary to perform data augmentation to increase the performance of unknown data. Most of the algorithms participating in the DCASE challenge are using a deep neural network-based algorithm with high model complexity, so they are using data augmentation and it improves the performance.

We used mix-up [13] and spec-augment [14] for the data augmentation. Mix-up is one of the most popular method in past DCASE 2018 challenge. It creates a new training sample by mixing a pair of two randomly chosen training samples. Spec-augment is an effective approach which shows significant performance improvement in acoustic speech recognition recently. It replaces values by zeros in randomly chosen time-frequency bands. It is also effective in acoustic scene classification task, and applied in most of the algorithms submitted in DCASE 2019 challenge.

Validation dataset	Overall		Unseen city	
Feature (Logmel number)	40	200	40	200
SubSpectralNet [7]	68.93	73.44	58.29	68.71
FCNN	71.15	75.97	59.89	70.32
FCNN-mixup	71.57	76.25	62.19	68.70
FCNN-spec	71.85	76.44	63.9	71.74
FCNN-mixup-spec	72.76	75.97	62.62	70.40
FCNN-triplet	72.67	76.61	63.07	70.24
FCNN-triplet-spec	**73.14**	**77.19**	**64.23**	**72.38**
DCASE 2019 Rank 1 [28]	85.10		-	

Table 1: Classification accuracies (%) of the proposed algorithm with different settings and the baseline CNN algorithm.

Input feature	Network	# of param
Logmel 40	DCASE baseline	117K
	SubSpectralNet [7]	331K
	Proposed FCNN	113K
Logmel 200	SubSpectralNet [7]	2,541K
	Proposed FCNN	871K
DCASE 2019 Rank 1 [28]		48M

Table 2: The number of parameters for the DCASE baseline, SubSpectralNet, and our proposed FCNN

4. EXPERIMENT

4.1. Dataset

The dataset for this task is the TAU Urban Acoustic Scenes 2019 dataset, consisting of recordings from various acoustic scenes in ten large European cities. For each recording location, there are 5-6 minutes of audio. The original recordings were split into segments with a length of 10 seconds that are provided in individual files. The dataset includes 10 scenes such as 'airport' and 'shopping mall'. TAU Urban Acoustic Scenes 2019 development dataset contains 40 hours of data with total of 14400 segments. Here, we used 9185 segments as a training dataset and 4185 segments as an evaluation dataset, and this split is given in the first fold of the validation set. For evaluation of unseen city, we used 1440 segments which is recorded in Milan and not appeared in the training dataset.

4.2. Setup

Our source code is implemented as python script using Torch library and our experiment is conducted on a GeForce GTX TITAN X GPU having 12Gb RAM. As the stereo audio segments whose length is 10 seconds are sampled as 48kHz in DCASE 2019 development dataset, we used 40 and 200 logmel features of the stereo channel without down-sampling as the input of CNN. Also, we set the sub-spectrogram size as 20 and overlap as 10 as the same setting in [7]. The input/output channel sizes of CNN structure is assumed to be $C^1_{in} = 2, C^1_{out} = 64, C^2_{in} = 64, C^2_{out} = 64$ for 40 logmel case and $C^1_{in} = 2, C^1_{out} = 32, C^2_{in} = 32, C^2_{out} = 64$ for 200 logmel case. The kernel size parameter is set to be $k = 7$ for all logmel cases. All of the convolution filters and weight matrices in dense layers are initialized by kaiming normal and xavier normal functions in pyTorch, respectively. The learning rate is set to 0.001 with Adam optimizer. Here, the hyper-parameters of triplet loss margin, and the balance coefficient between the cross-entropy and triplet losses are respectively given as $\alpha = 0.2$, and $\gamma = 10$.

4.3. Result

The DCASE 2019 development dataset includes recordings from ten cities, and the training subset contains only 9 cities excepting for Milan as unseen city. By checking the performance of unseen city, we can measure the generalization ability of the learned model. Therefore, we measured not only the accuracy of overall validation dataset but also the accuracy of the dataset only containing unseen city.

Experimental results are enumerated in Table 1. Since the classification accuracy of the evaluation set is saturated before the 100-th epoch, we trained all models for 200 epochs and averaged the accuracies of last 10 epochs. For each evaluation of the algorithms, the model training is conducted twice and the accuracies are also averaged.

For the overall dataset, the factorized CNN (FCNN) improves the performance of the baseline network by 2.22% and 2.53% for 40 and 200 logmel cases, respectively. Further improvement is confirmed when using the mix-up and spec-augment data augmentation. By adopting the proposed loss function combined the triplet loss also improves the performance of the FCNN by 1.52% and 0.64%. When the spec-augment is applied to the FCNN with the proposed loss function, the best performance which improves the baseline network by 4.21% and 3.75% is obtained. Here, mix-up cannot be used with triplet loss since the labels of augmented data using mix-up are not discrete. For the only evaluation of unseen city, the performance trends are similar to the overall dataset, and the quantity of the performance enhancement is increased. The total increase in performance of unseen city is 5.94% and 3.67%. For 40 logmel case, the performance increase is relatively 41% more than the increase of the overall dataset, and it supports that the proposed algorithm is robust against unseen data. Note that the performance comparison with DCASE 2019 Challenge Rank 1 system [28] is unfair since the system characteristics such as the number of input features, model size, usage of ensemble and so on are different from ours, but we enumerated it as a reference.

As the number of parameters in CNN is one of the main criteria, we briefly compared the number of model parameters. As enumerated in Table 2, the number of proposed FCNN parameters is about one-third of SubSpectralNet, and it is almost the same as DCASE baseline network.

5. CONCLUSION

In this paper, an acoustic scene classification algorithm based on a large-margin factorized CNN is proposed. The motivation of a factorized CNN is that the most key patterns in the mel-spectrogram including the long-term ambient and short-term event sounds are low-rank, and the factorized CNN can effectively learn these key patterns with reducing the number of model parameters. To increase the generalization performance of the learned model, the triplet loss is combined with the cross-entropy loss function. Experimental results show that our proposed algorithm outperforms a conventional simple CNN-based algorithm with decreasing the model complexity. Further improvement on unseen data is also shown and it supports that the proposed algorithm has better generalization performance.

6. REFERENCES

[1] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[2] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, "DCASE 2016 acoustic scene classification using convolutional neural networks," in *Proc. Workshop Detection Classif. Acoust. Scenes Events*, 2016, pp. 95–99.

[3] 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.* IEEE, 2005, pp. 158–161.

[4] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition with time–frequency audio features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

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

[6] "DCASE2019 task1a description," http://dcase.community/challenge2019/task-acoustic-scene-classification#subtask-a.

[7] S. S. R. Phaye, E. Benetos, and Y. Wang, "Subspectralnet–using sub-spectrogram based convolutional neural networks for acoustic scene classification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 825–829.

[8] Y. Han, J. Park, and K. Lee, "Convolutional neural networks with binaural representations and background subtraction for acoustic scene classification," *the Detection and Classification of Acoustic Scenes and Events (DCASE)*, pp. 1–5, 2017.

[9] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.

[10] 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.

[11] M. Dorfer, B. Lehner, H. Eghbal-zadeh, H. Christop, P. Fabian, and W. Gerhard, "Acoustic scene classification with fully convolutional neural networks and i-vectors," *Tech. Rep., DCASE2018 Challenge*, 2018.

[12] H. Zeinali, L. Burget, and J. Cernocky, "Convolutional neural networks and x-vector embedding for DCASE2018 acoustic scene classification challenge," *arXiv preprint arXiv:1810.04273*, 2018.

[13] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[14] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[15] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3852–3856.

[16] V. Arora, M. Sun, and C. Wang, "Deep embeddings for rare audio event detection with imbalanced data," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3297–3301.

[17] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[18] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4624–4628.

[19] T. Dau, B. Kollmeier, and A. Kohlrausch, "Modeling auditory processing of amplitude modulation. i. detection and masking with narrow-band carriers," *The Journal of the Acoustical Society of America*, vol. 102, no. 5, pp. 2892–2905, 1997.

[20] N. Moritz, J. Anemüller, and B. Kollmeier, "An auditory inspired amplitude modulation filter bank for robust feature extraction in automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 11, pp. 1926–1937, 2015.

[21] A. Torfi, J. Dawson, and N. M. Nasrabadi, "Text-independent speaker verification using 3d convolutional neural networks," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[22] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[23] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Real-time online singing voice separation from monaural recordings using robust low-rank modeling." in *ISMIR*. Citeseer, 2012, pp. 67–72.

[24] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2009.

[25] M. Wang, B. Liu, and H. Foroosh, "Factorized convolutional neural networks," in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.

[26] X. Li, D. Chang, T. Tian, and J. Cao, "Large-margin regularized softmax cross-entropy loss," *IEEE Access*, vol. 7, pp. 19 572–19 578, 2019.

[27] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[28] H. Chen, Z. Liu, Z. Liu, P. Zhang, and Y. Yan, "Integrating the data augmentation scheme with various classifiers for acoustic scene modeling," DCASE2019 Challenge, Tech. Rep., June 2019.