Deep Neural Decision Forest for Acoustic Scene Classification

Jianyuan Sun[1,3,*], Xubo Liu[1,*], Xinhao Mei[1], Jinzheng Zhao[1], Mark D. Plumbley[1], Volkan Kılıç[2], Wenwu Wang[1]

[1]Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK
[2]Department of Electrical and Electronics Engineering, Izmir Katip Celebi University, Turkey
[3]College of Computer Science and Technology, Qingdao University, China

***Abstract*—Acoustic scene classification (ASC) aims to classify an audio clip based on the characteristic of the recording environment. In this regard, deep learning based approaches have emerged as a useful tool for ASC problems. Conventional approaches to improving the classification accuracy include integrating auxiliary methods such as attention mechanism, pre-trained models and ensemble multiple sub-networks. However, due to the complexity of audio clips captured from different environments, it is difficult to distinguish their categories without using any auxiliary methods for existing deep learning models using only a single classifier. In this paper, we propose a novel approach for ASC using deep neural decision forest (DNDF). DNDF combines a fixed number of convolutional layers and a decision forest as the final classifier. The decision forest consists of a fixed number of decision tree classifiers, which have been shown to offer better classification performance than a single classifier in some datasets. In particular, the decision forest differs substantially from traditional random forests as it is stochastic, differentiable, and capable of using the back-propagation to update and learn feature representations in neural network. Experimental results on the DCASE2019 and ESC-50 datasets demonstrate that our proposed DNDF method improves the ASC performance in terms of classification accuracy and shows competitive performance as compared with state-of-the-art baselines.**

***Index Terms*—acoustic scene classification, random forest, convolution neural networks, deep learning**

I. Introduction

Acoustic scene classification (ASC) has attracted much attention in the fields of Audio and Acoustic Signal Processing (AASP) [1], [2], as shown in the Detection and Classification of Acoustic Scenes and Events (DCASE) challenges held in recent years, where several benchmark datasets have been introduced. The ASC task focuses on recognizing the audio clips in terms of the type of acoustic environment where they were captured. They are useful in applications such as health care [3], [4], and security surveillance [2].

In past few years, many methods have been developed for ASC. The classical ASC methods tend to employ hand-crafted features, including the Mel Frequency Cepstral Coefficients (MFCCs) [5], [6], spectrogram and log-mel filter banks [7], and to train the well-known classifier, such as support vector machine (SVM) [8] and decision trees [9]. However, theoretical and algorithmic advances together with the increasing capability in computer processing have led to the emergence of more sophisticated methods in artificial intelligence. A representative method for ASC tasks is deep learning which offers superior performance in handling a large number of features. The audio sequences are first converted to time-frequency representations, including log-mel spectrogram, wavelet transform, and short-time Fourier transform, as an input of a deep learning system. In particular, convolutional neural networks (CNNs) [5], [10] and recurrent neural networks (RNNs) based methods were shown to provide state-of-the-art performance on some ASC datasets [11]. Moreover, the CNN variations such as VGG and ResNet are applied to learn ASC representation [12], [13].

To further improve the classification performance, deep learning based approaches are extended with some auxiliary methods, i.e., attention mechanism, pre-trained models and ensemble sub-networks. Ding et al. proposed an ensemble system of the CNN and Gaussian Mixture Model (GMM) based on learned features to improve the classification performance [14]. Sugahara et al. improved the accuracy by the ensemble of ResNet-based models and data augmentation methods, such as mixup, time-shifting, and SpecAugment [15]. Han et al. introduced the attention mechanism to improve deep CNN performance [16]. Moreover, Huang et al. improved the deep CNN using spatial-temporal attention pooling [17]. Bilot et al. proposed a fusion system that uses multi-layer perceptron (MLP) to get the final results from the initial class label probability predictions [18]. In addition, Wang et al. used a custom-designed CNN for the recognition [19].

Conventional methods focus on learning better feature representations by proposing diverse network structures, which usually use the softmax classifier as the final layer. However, due to the complexity of audio clips collected from different environments, it is difficult for existing deep learning models using only a single classifier, i.e., Softmax, to distinguish audio clip categories. In machine learning, there are various well-known classifiers, such as SVM [8] and random forests [20], [21], which show an outstanding classification or prediction performance. In particular, the traditional random forests algorithm has achieved great success in practical applications, which is a typical ensemble method combining a fixed number of decision trees [20]. However, combining deep learning with the random forests method has received little attention due to the limitations

[*] The first two authors contributed equally to this work.

associated with the local optimal strategy, i.e., calculating node features and split thresholds using the Gini index or information gain rate, which results in the challenge in performing backpropagation for combined models [22]. Kontschieder et al. addressed this limitation by introducing DNDFs approach that unifies random forests using the representation learning with deep CNNs, which enables end-to-end training [22]. It was reported that DNDF shows state-of-the-art performance as compared to its machine learning counterparts [23].

In this paper, we investigate the performance of DNDF in ASC. The existing deep learning methods focus on learning good feature representations using the attention mechanism and ensemble sub-networks. Most methods routinely use the softmax classifier as the final layer which may not be sufficient for classifying audio clips of diverse categories from complex environments. Therefore, in our work, we employ a decision forest classifier instead of the softmax as the final predictor. Our work is the first attempt to apply the DNDF for the ASC task. Experiments show that the DNDF method achieves competitive results with the existing state-of-the-art model that uses a pre-trained model. Moreover, the DNDF method obtains better performance as compared to most deep CNN models with the attention mechanism and ensemble sub-networks.

The remainder of this paper is organized as follows. The next section introduces our proposed method. Section III-D presents experimental setup. Section III-E shows experimental results on the DCASE2019 development ASC subtask A and ESC-50 dataset. Conclusions are given in Section IV.

II. Methods

A. Deep Neural Decision Forests (DNDFs)

DNDF is a type of CNN that replaces the softmax layer with decision forests, consisting of several decision trees. Given a classification dataset with input and (finite) output space X and Y. A decision tree is a binary tree consisting of decision nodes and prediction nodes. Here, the symbol $\mathcal{N}$ is used to denote the decision node index of a decision tree. $\mathcal{L}$ denotes the set of the prediction node indices $\{1, ..., L\}$. Each prediction node $l \in \mathcal{L}$ has a probability distribution π_l over Y. Each decision node $n \in \mathcal{N}$ is assigned a decision function $d_n(\cdot;\theta)$, where the parameter θ from the CNN is used to update the feature representation. Moreover, the embedding function $f_n(\cdot;\theta) : X \rightarrow R$ is defined in CNN, which will determine the action of the decision function $d_n(\cdot;\theta)$ of the decision trees. Fig. 1 shows the structure of DNDF, including how decision nodes can be implemented by using the output of the final layer of CNN. For illustration, we only show the example of building a single decision tree in DNDF by using a fixed number of CNN embedding functions.

In DNDF, the fully-connected and convolutional layers are the same as those in a general CNN. The feature representations learned by the fully-connected layer are used as the tree node of the decision trees in decision forests. Therefore, CNN nodes share the same parameter θ that is used to update the feature representation of CNN as the tree nodes. The decision function of each decision node $d_n(\cdot;\theta)$ is defined as follows

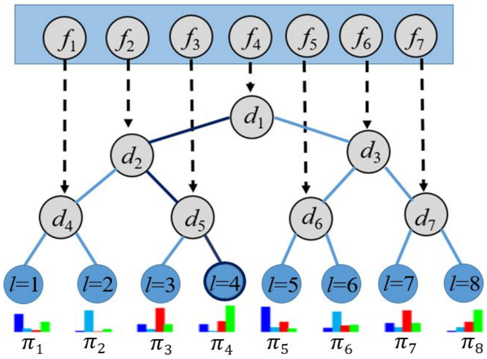

Fig. 1. The architecture of the DNDF model. In this model, seven embedding functions f_n ($n = 1, \ldots, 7$) are provided by the final layer of CNN, which is a real-value function depending on the sample and the parameter θ. Each output of f_n determines the decision function d_n of each node for the decision trees. Each prediction node at the bottom of the decision tree has the probability distribution π_l for each class.

$$d_n(x;\theta) = \sigma(f_n(x;\theta)), \tag{1}$$

where $x \in X$ is the input sample, $\sigma(f_n) = (1 + e^{-f_n})^{-1}$ is the sigmoid activation function, and $f_n(\cdot;\theta)$ is a real-valued function depending on the sample and the parameters θ, which can be regarded as the linear output unit of the neural network node. When a sample $x \in X$ arrives at a node, whether it goes to the left or right subtree of this node is determined by the output of $d_n(x;\theta)$. In DNDF, a sample arrives from a tree node to a leaf node via stochastic routing. The routing function $\mu_l(x|\theta)$ is defined as follows,

$$\mu_l(x|\theta) = \prod_{n \in N} d_n(x;\theta)^{\swarrow} \bar{d}_n(x;\theta)^{\searrow}, \tag{2}$$

where $\bar{d}_n(x;\theta) = 1 - d_n(x;\theta)$, $d_n^{\swarrow}$ represents the route from the current node to the left, and l is the leaf node. As an example in Fig. 1, for the sample reaching the leaf node $l = 4$, we have:

$$\mu_{(l=4)} = d_1(x)\bar{d}_2(x)\bar{d}_5(x). \tag{3}$$

Under the stochastic routing strategy, a sample arrives at a leaf node l, the related tree prediction is determined by the class label distribution π_l. π_{l^y} represents the probability of sample reaching leaf node l to take on class y. The final prediction for a sample that takes on a class y is the average probability of this sample reaching the leaf node, defined as

$$P[y|x,\theta,\pi] = \sum_{l \in \mathcal{L}} \pi_{l^y} \mu_l(x|\theta). \tag{4}$$

For the decision forests, it is an ensemble of several decision trees $\mathbb{F} = T_1, ..., T_k$. The final prediction of decision forests for a sample x is the average output of each tree, that is,

$$P_F[y|x] = \frac{1}{k} \sum_{h=1}^{k} P_{T_h}[y|x]. \tag{5}$$

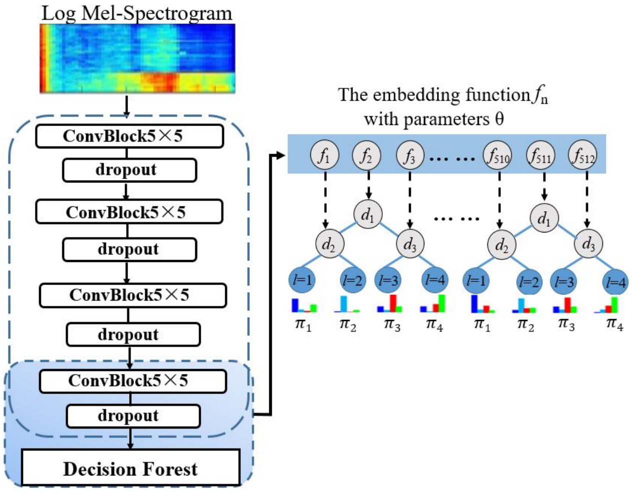

Fig. 2. The architecture of the DNDF model for ASC, where 512 embedding functions f_n ($n = 1, \ldots, 512$) are provided by the final layer of CNN4, and the decision trees are constructed based on these 512 embedding functions. The output of each f_n determines the decision function d_n of each node in the decision trees. Each node at the bottom of the decision tree gives the probability distribution π_l for each class of the acoustic scenes.

B. Application of DNDF in ASC

We proposed a DNDF for the task of ASC. Specifically, DNDF consists of two parts, i.e., CNN and decision forests. In particular, we design a CNN4 with four convolutional blocks. Each convolutional block has one convolutional layer with a kernel size of 5×5. After each convolutional layer, batch normalization and ReLU are used. The channel numbers of each convolutional block are $64, 128, 256, 512$, respectively. Moreover, an average pooling layer with kernel size 2×2 is employed between two neighbouring blocks for down-sampling. The decision forest is used after the fourth convolutional block. Here, DNDF takes log Mel-spectrogram features of the acoustic clips as the inputs. Fig. 2 shows the proposed architecture of DNDF for ASC, where 512 embedding functions $f_n(\cdot;\theta)$ ($n = 1, \ldots, 512$) are provided by the final layer of CNN4 to construct the decision trees, and the parameter θ is used to update the feature representation. In particular, the output of each $f_n(\cdot;\theta)$ determines the decision function $d_n(\cdot;\theta)$ of each node, i.e, Eq. (1). Each node at the bottom of the decision tree gives the probability distribution π_l for each class of the acoustic scenes, i.e., Eq. (4) and Eq. (5).

C. Learning Procedure of DNDF

In DNDF, given a training dataset $\mathcal{T} \subset X \times Y$, we start with random initialization of the common parameter θ of the decision trees and convolutional network. Furthermore, an iterative learning procedure is performed with a predefined number of epochs. Given the value of θ, the estimates of the node parameters π are obtained, setting the initial value in each leaf node as $\pi_{l y}^{(0)} = |Y|^{-1}$. To estimate and update the parameters θ and π, we search for the minimizers of the following empirical risk function, i.e.,

$$R(\theta, \pi; \mathcal{T}) = \frac{1}{|\mathcal{T}|} \sum_{(x,y) \in \mathcal{T}} L(\theta, \pi; x, y), \qquad (6)$$

where $L(\theta, \pi; x, y) = -log(P_T[y|x, \theta, \pi])$ denotes the log-loss term for the training sample $(x, y) \in \mathcal{T}$. Moreover, we divide the training data into mini-batches. Then, we perform stochastic gradient descent (SGD) to update the parameter θ by minimizing the empirical risk function based on each mini-batch. This learning process is given in Algorithm 1.

Algorithm 1: The parameter learning process of DNDF

```
Require: Given training set $\mathcal{T}$, epochs $= K$
Initialization parameter $\theta$
For each Epoch $i \in \{1, ..., K\}$ do
  Use an iterative scheme to compute $\pi$
  Split $\mathcal{T}$ into a fixed number of random mini-batches
    For each mini-batch do
      Use SGD to update $\theta$ by minimizing Eq. (6)
    end for
end for
```

III. Experiments

A. Dataset

To evaluate the performance of DNDF in ASC, the DCASE 2019 development ASC subtask A dataset and the ESC-50 environmental sound classification dataset are used in our experiments. DCASE 2019 is an extension of the DCASE 2018 TUT Urban ASC dataset with 10-second-long clips. There are 10 acoustic scenes in DCASE 2019, including bus, metro, metro_station, park, public_square, street_pedestrian, shopping_mall, tram, street_traffic and airport. The DCASE 2019 ASC dataset contains 9185 audio clips for training and 4185 clips for testing, sampled at $48\,\text{kHz}$. The ESC-50 dataset contains 2000 environmental audio clips each of 5-seconds, sampled $41\,\text{kHz}$, from 50 semantic classes.

B. Audio Processing and Augmentation

The original audio clip is converted to 64-dimensional log Mel-spectrogram by using the short-time Fourier transform with a frame size of 1024 samples, a hop size of 320 samples, and a Hanning window. In addition, SpecAugment is used for data augmentation [24].

C. Baseline

Our proposed method does not use auxiliary techniques such as attention mechanism and pre-trained model. Therefore, we choose the CNN-based variations as well as some CNN methods that use an attention mechanism, pre-trained and ensemble multiple sub-network as the baseline methods.

For the DCASE 2019 ASC dataset, we choose CDNN_CRNN [25], Attention_CNN [16], HPSS_MFCC_CNN [17], MIL_CNN [18] and MFCC_CNN [19] for the comparison. The CDNN_CRNN [25] model is a joint learning model based on a Deep CNN and Convolutional RNN. The Attention_CNN [16] model improves the performance of deep CNN by introducing an attention mechanism. The HPSS_MFCC_CNN

[17] uses deep CNN with spatial-temporal attention pooling. Moreover, the mixup data augmentation technique is employed to improve the classification performance further. Inspired by the multiple instant learning, the MIL_CNN [18] model uses MLP to get the final results from the initial predictions of the class label probability. The MFCC_CNN [19] model extracts the MFCC feature from audio files and uses the CNN with four-layer convolution and two fully connected layers for classification.

For the ESC-50 dataset, we compared existing algorithms including WELACNN [26], ACLNet [27], ENSCNN [28] and ACDNet [29]. The WELACNN model [26] uses the transfer learning technique to learn knowledge from weakly labeled audio data based on a CNN. The ACLNet [27] introduces an efficient CNN architecture by using data augmentation and regularization. The ENSCNN [28] ensembles classifiers by combining six data augmentation techniques and four signal representations to train five pre-trained CNNs. The ACDNET [29] uses a large deep CNN as a pipeline of a network for edge devices with resource constraints. The work [30] is a state-of-the-art attention-based CNN model with transformer encoder, pre-trained on AudioSet [31].

D. Training Procedure

The DNDF model is trained by employing the Adam optimizer with a learning rate of 0.001. Moreover, the batch size is set to 150, the number of epochs is 500, the depth of the decision trees is 10, and the number of decision trees is 100. In particular, for the ESC-50 dataset, to ensure the same settings as the comparison methods, we train the DNDF model with 5-fold cross-validation and report the average classification accuracy. For the comparison methods, we do not perform the training and testing processes. The results of the compared methods are all taken from the public results in their original papers.

E. Results

The same evaluation metric adopted in subtask A of the DCASE 2019 is used, i.e., the classification accuracy. Table I and Table II show the classification results of DNDF and baseline methods on DCASE 2019 ASC subtask A and ESC-50 datasets, respectively.

TABLE I
MEAN CLASSIFICATION ACCURACY OF THE COMPARED METHODS ON THE DCASE2019 DEVELOPMENT ASC SUBTASK A. THE BEST RESULT IS SHOWN IN BOLDFACE.

Model	DCASE2019
CDNN_CRNN [25]	73.70
Attention_CNN [16]	70.70
HPSS_MFCC_CNN [17]	73.90
MIL_CNN [18]	72.30
MFCC_CNN [19]	73.50
DCASE2019_baseline	63.30
DNDF (ours)	**75.90**

TABLE II
MEAN CLASSIFICATION ACCURACY OF THE COMPARED METHODS ON THE ESC-50 DATASET. THE BEST ACCURACY IS SHOWN IN BOLDFACE.

Model	ESC-50
WELACNN [26]	83.50
ACLNet [27]	85.65
ENSCNN [28]	88.65
ACDNet [29]	87.10
SOTA [30]	**95.70**
DNDF (ours)	88.90

TABLE III
THE CLASSIFICATION ACCURACY OF DNDF WITH DIFFERENT NUMBERS OF DECISION TREES ON THE DCASE2019 DEVELOPMENT ASC SUBTASK A AND THE ESC-50 DATASET. THE BEST ACCURACY IS SHOWN IN BOLDFACE.

Number of trees	DCASE2019	ESC-50
5	72.80	84.80
10	73.80	87.50
20	74.40	88.50
50	74.50	88.60
80	75.70	87.70
100	**75.90**	**88.90**
#Tree depth	10	10
#Batch size	150	150

Experimental results show that the DNDF method outperforms the baseline CNN-based methods. In particular, DNDF obtains better accuracy than the CNN-based model with the attention mechanism [16], [17] and ensemble multiple sub-network method [18], [28] as the decision forest can guide the representation learning in lower layers of deep CNN. Moreover, the CNN can gradually learn a good feature representation based on the decision forest prediction results. At the same time, the results also demonstrate that DNDF with decision forest as the final classifier can have satisfactory classification performance. The reason for achieving good prediction results is that DNDF uses a number of decision tree based classifiers, which has a stronger classification ability than a single classifier. However, it is worth noting that DNDF does not outperform the SOTA model. This is because the SOTA method has used the transformer based encoder that has been pre-trained on Audioset [30], while our method does not use pre-training.

In addition, to investigate the effect of the number of decision trees on the performance of DNDF, we evaluate the use of different numbers of the decision trees, and observe the change in the classification accuracy of DNDF. The results are shown in Table III. It can be found that the classification accuracy increases gradually with the number of decision trees. These results illustrate that DNDF is robust with the number of trees.

IV. CONCLUSION

In this paper, we have presented a DNDF model for the ASC by combining the convolution neural network and decision forest. The traditional random forests have a rich and successful history in machine learning and computer vision. However, the traditional random forests and neural networks cannot be learned together in an end-to-end manner due to the non-differentiability of the traditional random forests for the parameters of neural networks. The decision forest in DNDF differs from the conventional random forests because it is stochastic and differentiable. Therefore, the decision forest can learn and optimize the feature representation of the deep neural networks. To the best of our knowledge, our work is the first attempt to use the DNDF for the ASC task. Experiments demonstrated that the DNDF method achieves competitive results with the existing state-of-the-art model which, however, uses a pre-trained model built on large scale training data. Moreover, the DNDF method obtains better performance than existing deep CNN models with the attention mechanism and ensemble multiple sub-networks.

Acknowledgment

This work is partly supported by a Newton Institutional Links Award from the British Council and the Scientific and Technological Research Council of Turkey (TUBITAK), titled "Automated Captioning of Image and Audio for Visually and Hearing Impaired" (Grant numbers 623805725 and 120N995), a grant EP/T019751/1 from the EPSRC of the UK, a PhD scholarship from the University of Surrey, and a Research Scholarship from the China Scholarship Council. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

References

[1] L. Zhang, Z. Shi, and J. Han, "Pyramidal temporal pooling with discriminative mapping for audio classification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 770–784, 2020.

[2] J. Abeßer, "A review of deep learning based methods for acoustic scene classification," *Applied Sciences*, vol. 10, no. 6, 2020.

[3] S. Amiriparian, M. Gerczuk, S. Ottl, N. Cummins, and B. Schuller, "Snore sound classification using image-based deep spectrum features," in *Interspeech*, 2017, pp. 20–24.

[4] X. Liu, Y. Mou, Y. Ma, C. Liu, and Z. Dai, "Speech emotion detection using sliding window feature extraction and ANN," in *IEEE 5th International Conference on Signal and Image Processing*, 2020, pp. 746–750.

[5] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer, "A hybrid approach using binaural i-vectors and deep convolutional neural networks," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events*, vol. 6, pp. 5024–5028, 2016.

[6] X. Liu, T. Iqbal, J. Zhao, Q. Huang, M. D. Plumbley, and W. Wang, "Conditional sound generation using neural discrete time-frequency representation learning," in *IEEE 31st International Workshop on Machine Learning for Signal Processing*, 2021, pp. 1–6.

[7] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, "Acoustic scene classification using convolutional neural networks," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2016.

[8] J. T. Geiger, B. Schuller, and G. Rigoll, "Large-scale audio feature extraction and SVM for acoustic scene classification," in *Applications of Signal Processing to Audio & Acoustics*, 2014, pp. 1–4.

[9] H. Phan, L. Hertel, M. Maass, P. Koch, and A. Mertins, "Label tree embeddings for acoustic scene classification," in *ACM Conference on Multimedia*, 2016, pp. 486–490.

[10] H. Phan, L. Hertel, M. Maass, P. Koch, R. Mazur, and A. Mertins, "Improved audio scene classification based on label-tree embeddings and convolutional neural networks," *IEEE/ACM Transactions Audio Speech Language Process*, vol. 25, no. 6, pp. 1278–1290, 2017.

[11] T. H. Vu and J.-C. Wang, "Acoustic scene and event recognition using recurrent neural networks," *Detection and Classification of Acoustic Scenes and Events*, pp. 1–3, 2016.

[12] O. Mariotti, M. Cord, and O. Schwander, "Exploring deep vision models for acoustic scene classification," in *Detection and Classification of Acoustic Scenes and Events*, 2018, pp. 103–107.

[13] L. Zhang, J. Han, and Z. Shi, "Learning temporal relations from semantic neighbors for acoustic scene classification," *IEEE Signal Processing Letter*, vol. 27, pp. 950–954, 2020.

[14] B. Ding, G. Liu, and J. Liang, "Acoustic scene classification based on ensemble system," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[15] R. Sugahara, M. Osawa, and R. Sato, "Ensemble of simple resnets with various mel-spectrum time-frequency resolutions for acoustic scene classifications," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2021.

[16] H. Liang and Y. Ma, "Acoustic scene classification using attention-based convolutional neural network," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[17] H. Zhenyi and J. Dacan, "Acoustic scene classification based on deep convolutional neural network with spatial-temporal attention pooling," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[18] V. Bilot, N. Duong, and A. Ozerov, "Acoustic scene classification with multiple instance learning and fusion," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[19] Z. Wang, J. Ma, and C. Li, "Acoustic scene classification based on CNN system," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[20] L. B. Statistics and L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] J. Sun, H. Yu, G. Zhong, J. Dong, S. Zhang, and H. Yu, "Random shapley forests: Cooperative game-based random forests with consistency," *IEEE Transaction Cybernetics*, vol. 52, no. 1, pp. 205–214, 2022.

[22] P. Kontschieder, M. Fiterau, A. Criminisi, and S. R. Bulo, "Deep neural decision forests," in *IEEE International Conference on Computer Vision*, 2015, pp. 1467–1475.

[23] S. R. Bulo and P. Kontschieder, "Neural decision forests for semantic image labelling," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 81–88.

[24] 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.

[25] L. Pham, T. Doan, D. T. Ngo, H. Nguyen, and H. H. Kha, "Cdnn-crnn joined model for acoustic scene classification," *Technical report, Detection and Classification of Acoustic Scenes and Events Challenge*, 2019.

[26] A. Kumar, M. Khadkevich, and C. Fügen, "Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 326–330.

[27] J. J. Huang and J. J. A. Leanos, "AclNet: efficient end-to-end audio classification CNN," *arXiv preprint arXiv:1811.06669*, 2018.

[28] L. Nanni, G. Maguolo, S. Brahnam, and M. Paci, "An ensemble of convolutional neural networks for audio classification," *Applied Sciences*, vol. 11, no. 13, pp. 57–96, 2021.

[29] M. Mohaimenuzzaman, C. Bergmeir, I. T. West, and B. Meyer, "Environmental sound classification on the edge: A pipeline for deep acoustic networks on extremely resource-constrained devices," *arXiv preprint arXiv:2103.03483*, 2021.

[30] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio spectrogram transformer," in *Interspeech*, 2021, pp. 571–575.

[31] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 776–780.