Acoustic Scene Classification using Audio Tagging

Jee-weon Jung, Hye-jin Shim, Ju-ho Kim, Seung-bin Kim, and Ha-Jin Yu[†]

School of Computer Science, University of Seoul, Republic of Korea

jeewon.leo.jung@gmail.com, shimhz6.6@gmail.com, wngh1187@naver.com,
kimho1wq@naver.com, hjyu@uos.ac.kr

Abstract

Acoustic scene classification systems using deep neural networks classify given recordings into pre-defined classes. In this study, we propose a novel scheme for acoustic scene classification which adopts an audio tagging system inspired by the human perception mechanism. When humans identify an acoustic scene, the existence of different sound events provides discriminative information which affects the judgement. The proposed framework mimics this mechanism using various approaches. Firstly, we employ three methods to concatenate tag vectors extracted using an audio tagging system with an intermediate hidden layer of an acoustic scene classification system. We also explore the multi-head attention on the feature map of an acoustic scene classification system using tag vectors. Experiments conducted on the detection and classification of acoustic scenes and events 2019 task 1-a dataset demonstrate the effectiveness of the proposed scheme. Concatenation and multi-head attention show a classification accuracy of 75.66 % and 76.58 %, respectively, compared to 73.63 % accuracy of the baseline. The system with the proposed two approaches combined demonstrates an accuracy of 76.75 %.

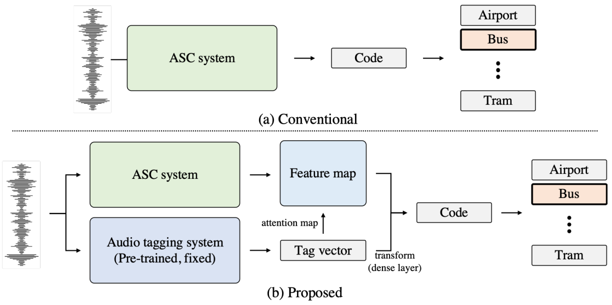
Index Terms: acoustic scene classification, audio tagging, attention

1. Introduction

Acoustic scene classification (ASC) is an emerging task with a wide range of applications. The task is to identify a given audio recording into one of the pre-defined acoustic scenes (i.e. classes). Leveraging the recent advances in deep learning, majority of ASC systems utilize deep neural networks (DNNs) [1–4]. To facilitate the studies on the ASC task, the detection and classification of acoustic scenes and events (DCASE) community is providing a platform with annual competitions and public datasets [5–7].

Most ASC systems perform the task either in an end-to-end manner of classifying the input recording directly or by using a DNN's last hidden layer output as a representation vector and utilize a back-end classifier [8, 9]. Both methodologies utilize a single step approach. On the other hand, humans are known to first recognize the existence of different sound events and then conduct scene classification [10]. This implies that humans conduct ASC task using a two-step approach. For example, knowing that an airplane takeoff sound event exists can be used to infer that the scene is an airport rather than a shopping mall.

Recently there have been few studies to jointly conduct the ASC and the audio tagging task [11–13]. Bear *et. al.* [11] proposed to jointly conduct sound event detection and ASC task where ASC is conducted by a majority voting mechanism based on the result of sound event detection. Imoto *et. al.* [13] studied the improvement of sound event detection performance by using a multi-task learning of sound events and acoustic scenes. It investigated the use of sound event information for the ASC task by proposing a graphical model that sequentially conducts event detection and scene classification. To our knowledge, however, our study is the first to use tag vectors extracted from a separate audio tagging system to improve the ASC system.



(a) Conventional

(b) Proposed

Figure 1: *Conceptual illustration of* ***(a)****: conventional ASC system and* ***(b)****: the proposed scheme using audio tagging.*

In this paper, we propose an ASC system inspired by the human perception mechanism and utilizes tag vectors. We expect that the ASC system can improve by using the information from the tag vectors. A tag vector refers to the output of an audio tagging system that represents the existence of various sound events. Figure 1 illustrates the concept of the proposed scheme compared to conventional DNN-based ASC systems. The proposed framework utilizes a trained audio tagging system that demonstrated the best performance in DCASE 2019 challenge [14]. A given recording is input to both the audio tagging and the ASC system to extract a tag vector and a representation vector (i.e. code). Both the tag vector and the code are used for the ASC task. We propose various methods to utilize the tag vectors for the ASC task.

Specifically, we first propose to concatenate a tag vector with a code of the ASC system. Three methods are addressed in Section 3.1, and are explored to derive a concatenated representation and experimentally validated. We also propose to apply the attention mechanism to the feature map of the ASC task using a tag vector. An attention mechanism exclusively emphasizes a DNN's intermediate representation with or without exploiting external representations [15–19]. Our proposal belongs to the former case where a feature map is attended by an attention map derived from a tag vector. We also explore multi-head attention [16] mechanism which demonstrates competitive performances. Finally, we achieve further improvements by combining the two proposed approaches.

The rest of this paper is organized as follows. Section 2 describes the two baseline systems, an ASC system and an audio tagging system. Section 3 introduces the proposed framework

[†] Corresponding author

Table 1: *DNN architecture of the baseline. Strided-conv refers to a convolutional layer that has a stride size identical to filter length for processing raw waveforms [20]. Numbers inside Conv refer to filter length, stride, and number of filters. (BN: batch normalization [21], FC: fully-connected.)*

Layer	**Input:(479999, 2)**	**Output shape**
Strided -conv	Conv(12,12,128) BN LeakyReLU	(39999, 128)
Res block	{ Conv(3,1,128) BN LeakyReLU Conv(3,1,128) BN LeakyReLU MaxPool(3) } ×7	(18, 128)
Gl avg	Global avg pool()	(128,)
GL max	Global max pool()	(128,)
Concat	-	(256,)
Code	FC(64)	(64,)
Output	FC(10)	(10,)

which utilizes tag vectors for the ASC task. Experimental validation of the proposed approach is presented in Section 4 and the paper is concluded.

2. Baseline systems

2.1. ASC system

The ASC system used for this paper comprises two subsystems: a front-end DNN for feature (code) extraction, and a back-end support vector machine (SVM) for classification. We adopt a front-end DNN that directly inputs raw waveforms [9, 22]. The system first extracts frame-level representations using convolutional layers with residual connections [23, 24]. A global max pooling and a global average pooling aggregates frame-level features, and they are fed to a fully-connected layer. The output of the layer is used as the code. The acoustic scenes are classified in the train phase, but the output layer is removed after training. Table 1 describes the overall architecture.

In this study, an SVM with radial basis function or sigmoid kernel is used for back-end classification. SVM is widely selected as a back-end classifier of the ASC task. Unless otherwise stated, all performance reported in this paper is of the systems using SVM as the back-end classification.

2.2. Audio tagging system

The multi-label audio tagging task determines whether various defined sound events exist in an input audio recording. In this task, the output is a vector of dimension equal to the number of pre-defined sound events. Each dimension has a value within the range of 0 to 1 and describes the posterior probability of a sound event using a sigmoid non-linear function. Throughout this paper, the output vector will be referred to as 'tag vector'. In this study, we use the system proposed by Akiyama *et. al.* [14], the winning system of the DCASE 2019 challenge task 2, which adopts a multi-task learning architecture and soft pseudo-label framework. The system aims to train a multi-label audio tagging system under a scenario where a small portion of human labeled data and abundant auto-generated noisy labels exist. Specifically, we use the system with Mel-spectrogram input and ResNet [23] architecture combination as it is the best performing single system reported. Further details regarding this system are provided in [14].

3. Audio tagging-based ASC

In this section we propose two approaches that use a tag vector for the ASC task. Tag vector in this paper refers to an output of an audio tagging system where each dimension represents the existence of a pre-defined sound event in a real value of 0 to 1. The two proposed methods can also be used concurrently, leading to further improvements. Figure 2 illustrates two methods and the final proposed system using both methods. Figure 2-(a) shows that the first method concatenates a tag vector with an intermediate representation of the ASC system. Figure 2-(b) shows that the second method uses a multi-head attention using tag vectors. The use of the two methods concurrently is depicted in Figure 2-(c).

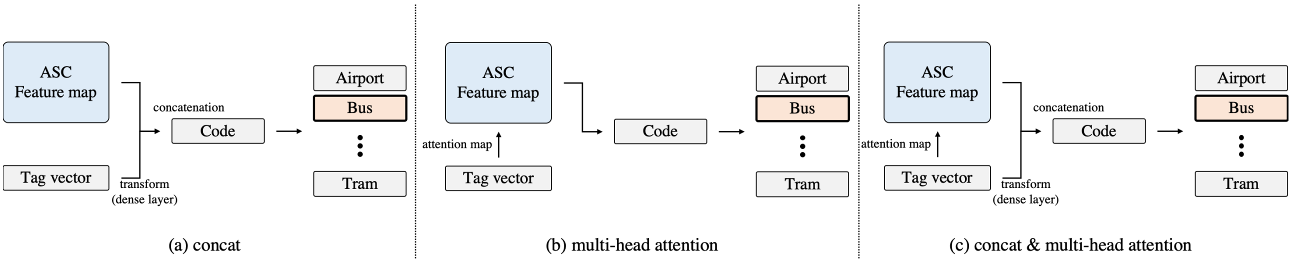
3.1. Concatenation

In this sub-section, three methods are addressed to concatenate a tag vector with the feature map of an ASC system (Figure 2-(a)). The simplest approach is to concatenate the feature map of an ASC system after global pooling ('*Code*' in Table 1) with a tag vector and use it as a code representation. However, this approach increases the dimensionality of the code and the two representations separately exist in each subspace of the concatenated representation. The second method conducts feature transformation using a fully-connected layer after concatenating the code of ASC and a tag vector. In this case, the transformed representation is used as the code.

Additionally, we propose to use a feature transformation to a tag vector before concatenation. A tag vector itself represents the probability of each pre-defined sound events. We hypothesize that the transformations before concatenation via fully-connected layers would lead to more discriminative features. Experiments in Table 2 show that all three approaches show improvements compared to the baseline where the third approach demonstrated the best performance.

3.2. Multi-head attention

Attention is a widely used mechanism that exclusively emphasizes discriminative features [15–19]. It utilizes a vector referred to an attention map where each value of an attention map has a value between 0 and 1, summing up to 1 using a softmax function. Attention is conducted by multiplying a given feature with an attention map. This mechanism has been actively adopted for the ASC task, where self-attention scheme which does not utilize external data was mainly studied. In this study, we propose to use a multi-head attention on the filter dimension of a feature map of the ASC system where a tag vector is used to derive the attention vector. Multi-head attention divides the representation space (filters in this case) into the same number of sub-spaces as the number of heads. It then applies softmax function to each sub-space, whereas conventional attention mechanisms apply softmax function to the entire space [16]. The assumption based on the use of attention with tag vectors is that the information regarding sound events that reside in tag vectors is sufficient to emphasize discriminative filters for the ASC task.



(a) concat

(b) multi-head attention

(c) concat & multi-head attention

Figure 2: *Proposed methods which utilize tag vectors for the ASC task.* ***(a)****: Concatenates the tag vector with the output of ASC feature map. Concatenating with or without feature transformation is explored.* ***(b)****: Conduct a multi-head attention where the tag vector is used to derive an attention map for the ASC feature map.* ***(c)****: Both (a) and (b) are conducted.*

Let M be a feature map of an ASC task, $M \in \mathbb{R}^{f \times t}$ described as:

$$M = \begin{bmatrix} M_{11} & M_{12} & \cdots \\ \vdots & \ddots & \\ M_{h1} & \cdots & M_{ht} \end{bmatrix}, M_{ht} \in \mathbb{R}^{f/h}, \tag{1}$$

where f refers to the number of filters, t refers to the length of the sequence in time dimension, and h refers to the number of heads. Let T be a tag vector, $T \in \mathbb{R}^{c}$ where c refers to the number of pre-defined sounds events in the audio tagging task. We first derive an attention map $am \in \mathbb{R}^{f}$, using a tag vector through feature transformation using fully-connected layers. Then we split the attention map into h which refers to the number of heads and then subsequently apply softmax function and then concatenate again. Derived attention map is denoted as $am = [am_1, am_2, \cdots, am_h]$, $am_h \in \mathbb{R}^{f/h}$. We calculate the attention-applied feature map, M' as:

$$M' = \begin{bmatrix} M_{11} \odot am_1 & M_{12} \odot am_1 & \cdots \\ \vdots & \ddots & \\ M_{h1} \odot am_h & \cdots & M_{ht} \odot am_h \end{bmatrix}, \tag{2}$$

where $\odot$ is the element-wise multiplication. Because we apply a multi-head attention to the filter dimension, identical am_h is multiplied to all elements in time dimension (M_{h1} to M_{ht}).

3.3. Concatenation & multi-head attention

The two approaches addressed in the previous sub-sections either directly concatenate a tag vector or derive an attention map to attend a feature map of the ASC system. We further propose to apply the two methods together, assuming that the two approaches are complementary (Figure 2-(c)). Because both methods convert tag vectors before concatenating or deriving an attention map, we can either perform the same transformation for both methods, or apply transformation individually for each method. The comparison between the two configurations are described in Table 4 and 5.

4. Experiments and results

4.1. Dataset

All experiments regarding the ASC task reported in this paper uses the DCASE 2019 task 1-a dataset [7]. It comprises 40 hours of audio recordings from 12 different European cities. Ten acoustic scenes are defined and each recording is adjusted to a duration of 10 seconds, making it 14400 recordings in total. Each recording has a sampling rate of 48000 and recorded in stereo. We proceed with the official fold-1 configuration train/test subset split and report the overall classification accuracy on the test set[1].

Table 2: *Experimental results of the baseline without the utilization of tag vectors and the three methods using audio tag vectors directly by concatenating to the ASC system. Refer to Figure 2-(a) for illustration.* ***Bold*** *depicts the best performance. (-: not applicable)*

System	Transform after concat	# Layers before transform	Acc
Baseline	-	-	73.63
Codecat	×	-	74.15
Before code	○	×	74.36
	○	1	74.84
	○	2	74.22
	○	**3**	**75.66**
	○	4	75.22
	○	5	74.07

To train the audio tagging system to extract tag vectors, we use the DCASE 2019 task 2 dataset [25]. It includes a curated subset of 4970 audio clips with manual labels. The curated subset has a duration range of 0.3 to 30 seconds and a total of 10.5 hours. 80 sound events are defined. The dataset also includes relatively larger noisy subset of 19815 audio clips with noisy labels. The noisy subset has a duration range of 1 to 15 seconds and a total duration of approximately 80 hours.

4.2. Experimental configurations

The ASC system directly inputs stereo raw waveforms of shape $(479999, 2)$ with pre-emphasis applied. Residual blocks include batch normalization [21] and Leaky ReLU activation functions [26]. The code has a dimensionality of 64. Other details such as convolution layer's filter length are described in Table 1.

The audio tagging system [14] uses Mel-spectrogram of 128 Mel-frequency channels with augmentations including SpecAugment [27] and slicing. Mix-up [28] is applied to both the ASC system and the audio tagging system. ResNet architecture [23] with approximately 44 million trainable parameters is exploited with few modifications. Further details regarding the

[1] Performance using the challenge evaluation set cannot be evaluated because ground truth labels are not released.

Table 3: *Results of conducting multi-head attention using tag vectors. Refer to Figure 2-(b) for illustration.* ***Bold*** *depicts the best performance from each row.*

		# Transform layers for attention map			
		0	1	2	3
# Head	2	75.67	**76.58**	75.93	75.50
	4	75.45	75.22	**76.24**	76.17
	8	74.41	74.36	74.89	**75.17**
	16	73.61	74.47	74.43	**75.96**
	32	75.14	**75.57**	75.24	75.31

Table 4: *Results of applying both concatenation and multi-head attention using tag vectors where tag vector is transformed through separate layers to conduct the two methods. Refer to Figure 2-(c) for illustration.* ***Bold*** *depicts the best performance from each row.*

		# Transform layers for concat/att			
		3/3	3/4	4/3	4/4
# Head	2	74.76	**75.74**	75.17	75.14
	4	75.31	75.84	**76.24**	75.34

audio tagging system are addressed in [14].

4.3. Result analysis

Table 2 demonstrates the performances of the baseline which does not use tag vectors and the three proposed methods that concatenate a tag vector with an ASC code. All performances from Table 2 to 6 are reported using the official fold 1 test set. *Codecat* refers to the first method in Section 3.1 using the concatenation of ASC system's code and tag vector as a new code. *Before code* without and with transformation before concatenation refers to the second and the third method respectively. Results show that the performance improves in all the proposed configurations. The best performance can be achieved by conducting transforms on the tag vector before concatenation and also transforming the concatenated representation to derive the code.

Table 3 describes the result of applying multi-head attention to the ASC system with an attention map derived from the tag vector. We explore the performance difference according to the different numbers of heads and the number of feature transform layers applied to the tag vectors. Results demonstrate that multi-head attention improves the performance more compared to mere concatenation in most configurations. The best configurations are achieved by using one layer for tag vector transformation and 2 heads for attention.

Table 4 and 5 describe the results of applying both concatenation and multi-head attention. Table 4 shows the result using a separate fully-connected layers for transforming a tag vector for concatenation and attention map, whereas, Table 5 shows the result using shared layers for the two methods. For experiments in Table 4, the number of heads of 2 and 4 was explored for multi-head attention and the number of transform layers of 3 and 4 was explored because these configurations demonstrated best performances in Table 2 and 3.

Results show that using separate transform layers (76.24 %) leads to worse performance than using only multi-head attention (76.57 %). Using identical layers for tag vector transformation and conducting both concatenation and attention resulted in further performance improvement, showing an accuracy of 76.75 %. In our analysis, separate transformation layers led to worse performance because of overfitting incurred by too many parameters of fully-connected layers.

Table 5: *Results of applying both concatenation and multi-head attention where tag vector is transformed through separate layers for each method. Refer to Figure 2-(c) for illustration.* ***Bold*** *depicts the best performance from each row.*

		# Transform layers			
		0	1	2	3
# Head	2	**76.12**	75.62	75.34	75.69
	4	75.43	76.00	76.24	**76.75**
	8	75.84	74.71	**75.96**	75.69
	16	**76.00**	75.81	75.91	75.72
	32	75.26	75.45	76.24	**76.27**

Table 6: *Comparison of the proposed system with top three performing system using raw waveform as the input on the DCASE 2019 challenge. (-: not reported)*

System	# Param	Acc (%)
Huang *et. al.* [2]	53452k	76.08
Jung *et. al.* [22]	636k	75.81
Zheng *et. al.* [8]	-	69.23
Ours	676k	**76.75**

Finally, Table 6 shows the comparison between the best performing system of our study and three systems that demonstrated the best performance using raw waveform as input. Huang *et. al.*'s system utilized SincNet architecture [29] which replaces the first convolution layer with the proposed sinc-convolution layers combined with AclNet [30]. Jung *et. al.*'s system utilized teacher-student learning [31] with specialist DNNs [32]. Zheng *et. al.*'s system utilized an end-to-end DNN architecture with random cropping and padding augmentations. Results show that the proposed system shows the best performance among other systems that use raw waveform as input.

5. Conclusion

In this paper, we explored various approaches to exploit audio tagging representations for improving the ASC system which mimics human perception of the ASC task. Three methods were studied to utilize audio tagging representation through a tag vector concatenation. Multi-head attention was adopted to emphasize the feature map of the ASC system using a tag vector. Based on the success of both concatenation and multi-head attention, we achieved further performance improvement by applying the proposed methods concurrently. This system showed an accuracy of 76.75 % on the fold-1 test set configuration of the DCASE task 1-a dataset.

6. Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(2020R1A2C1007081).

7. References

[1] K. Koutini, H. Eghbal-zadeh, and G. Widmer, "Receptive-field-regularized cnn variants for acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, New York University, NY, USA, October 2019, pp. 124–128.

[2] J. Huang, P. Lopez Meyer, H. Lu, H. Cordourier Maruri, and J. Del Hoyo, "Acoustic scene classification using deep learning-based ensemble averaging," DCASE2019 Challenge, Tech. Rep., June 2019.

[3] S. Hyeji and P. Jihwan, "Acoustic scene classification using various pre-processed features and convolutional neural networks," DCASE2019 Challenge, Tech. Rep., June 2019.

[4] H. Phan, O. Y. Chén, L. Pham, P. Koch, M. De Vos, I. McLoughlin, and A. Mertins, "Spatio-temporal attention pooling for audio scene classification," *Proc. Interspeech 2019*, pp. 3845–3849, 2019.

[5] T. Virtanen, A. Mesaros, T. Heittola, A. Diment, E. Vincent, E. Benetos, and B. M. Elizalde, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*. Tampere University of Technology. Laboratory of Signal Processing, 2017.

[6] M. D. Plumbley, C. Kroos, J. P. Bello, G. Richard, D. P. Ellis, and A. Mesaros, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*. Tampere University of Technology. Laboratory of Signal Processing, 2018.

[7] M. Mandel, J. Salamon, and D. P. W. Ellis, *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*. NY, USA: New York University, October 2019.

[8] X. Zheng and J. Yan, "Acoustic scene classification combining log-mel CNN model and end-to-end model," DCASE2019 Challenge, Tech. Rep., June 2019.

[9] J.-w. Jung, H.-s. Heo, H.-j. Shim, and H.-j. Yu, "DNN based multi-level feature ensemble for acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 113–117.

[10] C. Guastavino, "Categorization of environmental sounds." *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale*, vol. 61, no. 1, p. 54, 2007.

[11] H. L. Bear, I. Nolasco, and E. Benetos, "Towards joint sound scene and polyphonic sound event recognition," *arXiv preprint arXiv:1904.10408*, 2019.

[12] K. Imoto, N. Tonami, Y. Koizumi, M. Yasuda, R. Yamanishi, and Y. Yamashita, "Sound event detection by multitask learning of sound events and scenes with soft scene labels," *arXiv preprint arXiv:2002.05848*, 2020.

[13] K. Imoto and S. Shimauchi, "Acoustic scene analysis based on hierarchical generative model of acoustic event sequence," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 10, pp. 2539–2549, 2016.

[14] O. Akiyama and J. Sato, "Multitask learning and semi-supervised learning with noisy data for audio tagging," DCASE2019 Challenge, Tech. Rep., June 2019.

[15] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[17] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[18] P. Safari and J. Hernando, "Self multi-head attention for speaker recognition," *Proc. Interspeech 2019*, pp. 4305–4309, 2019.

[19] Z. Ren, Q. Kong, K. Qian, M. Plumbley, and B. Schuller, "Attention-based convolutional neural networks for acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 39–43.

[20] J.-W. Jung, H.-S. Heo, I.-H. Yang, S.-H. Yoon, H.-J. Shim, and H.-J. Yu, "DNN-based audio scene classification for DCASE2017: Dual input features, balancing cost, and stochastic data duplication," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, November 2017, pp. 59–63.

[21] I. Sergey and S. Christian, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

[22] J.-w. Jung, H.-S. Heo, H.-j. Shim, and H.-J. Yu, "Distilling the knowledge of specialist deep neural networks in acoustic scene classification," in *Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, 2019, p. 114.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] ——, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[25] E. Fonseca, M. Plakal, F. Font, D. P. W. Ellis, and X. Serra, "Audio tagging with noisy labels and minimal supervision," in *Submitted to DCASE2019 Workshop*, NY, USA, 2019.

[26] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1, 2013, p. 3.

[27] 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.

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

[29] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," *arXiv preprint arXiv:1808.00158*, 2018.

[30] J. J. Huang and J. J. A. Leanos, "Aclnet: efficient end-to-end audio classification cnn," *arXiv preprint arXiv:1811.06669*, 2018.

[31] J. Li, R. Zhao, J. Huang, and Y. Gong, "Learning small-size dnn with output-distribution-based criteria," in *Fifteenth annual conference of the international speech communication association*, 2014.

[32] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.