MULTI-SCALE NETWORK BASED ON SPLIT ATTENTION FOR SEMI-SUPERVISED SOUND EVENT DETECTION

Xiujuan Zhu[1,3], *Ying Hu*[1,3*], *Xinghao Sun* [1,3], *Liang He*[1,2],

[1] School of Information Science and Engineering, Xinjiang University, Urumqi, China
{xiujuanzhu841}@gmail.com, {huying}@xju.edu.cn
[2] Department of Electronic Engineering, Tsinghua University, China
[3] Key Laboratory of Signal Detection and Processing in Xinjiang, China

ABSTRACT

Sound scene in real environment is generally composed of different types of sound events meanwhile the time-frequency scales of these events are diverse. Thus, it is important to design a proper mechanism to extract the multi-scale features for sound event detection (SED). In order to improve the discriminative ability of different types of sound events, we propose a multi-scale SED network based on split attention. We design a Multi-scale (MS) module to extract the fine-grained and the coarse-level features in parallel. A Channel Shuffle (CS) operation is introduced to enhance the cross-channel information communication among the features with different scales. Also, a Split Attention (SA) module is designed to learn several sub-features separately and an attention mechanism is followed to generate the corresponding importance coefficients for each sub-features. Experiments on DCASE2021 Task4 dataset demonstrate the effectiveness of our proposed multi-scale network.

Index Terms— sound event detection, multi-scale, channel shuffle, split attention

1. INTRODUCTION

The purpose of sound event detection (SED) is to identify the categories of sound events and detect the onset and offset of the target events in an audio sequence. Unlike audio classification task that it only needs to determine the event categories, detection task also needs to predict the temporal position of occurring events. Thus, SED is a more difficult task. SED has drawn great attention recently in a variety of applications, such as surveillance [1], smart cities and homes [2], [3], as well as multimedia information retrieval [4]. There are three kinds of learning approaches in SED: fully supervised SED, weakly supervised SED and semi-supervised SED. Following the baseline of DCASE2021 Task4 , this paper only focuses on semi-supervised SED based on mean teacher method [5].

Real-life SED is challenging since different sound events exhibit different time-frequency properties. For example, "Dog" and "Dishes" last shorter while "Running water" and "Blender" last longer in the time domain and cover a wider frequency range. If the model performs on a single resolution, it's hard to deal with the different types of sound events. Thus, how to obtain the multi-scale features and integrate the features with inconsistent scales is a key point in SED.

Multi-scale mechanism has drawn great attention in SED task. Zhang et al. [6] proposed Multi-Scale Time-Frequency Attention module to extract the information at multiple resolutions. Ding et al. [7] further proposed an multi-scale detection method based on Hourglass network. The mechanism of Feature pyramid [8] has proved to be useful to obtain multi-resolution features in SED [9] , [10]. Another way to get multi-scale features is to use dilated convolution. Li et al. [11] proposed a dilated convolution recurrent neural network (CRNN) to verify the effectiveness of different dilation rates in convolution layers. Drossos et al. [12] proposed to use dilated convolution instead of GRU to capture long temporal context. Different from the above mentioned methods, in this paper, the multi-scale is only reflected from the convolution kernels of different sizes, it is a relatively simple structure. Su et al. [13] proposed a channel shuffle module to promote cross-channel information communication between the high-level and low-level information. Zhang et al. [14] proposed the ResNeSt based on the split-attention and proved its effectiveness. The group learning mechanism in split-attention ensures the network only to learn sub-features in adjacent channels. Wang et al. [15] also showed that the channel features are mainly related to their adjacent channel features while little related to the remote channel features.

Inspired the above related works, we propose a multi-scale SED network based on split attention. The multi-scale module exploits convolution kernels of different sizes to learn the multi-scale features in parallel, which improves its ability to recognize sound events. Motivated by [13], the channel shuffle operation is adopted to enable the cross-channel information flowing among the features with different scales. Inspired by ResNeSt [14], we adopt split attention module based on group convolution to separately learn sub-features and also generate attention weights to re-weight these sub-features.

This paper is organized as follows. We introduce the proposed SED network in Section 2 , describe the dataset and evaluation metrics in Section 3 , analyze the experimental results in Section 4 , and conclude the paper in Section 5 .

2. PROPOSED METHOD

In this section, we firstly present the overall network structure. Then we separately introduce the proposed multi-scale module, channel shuffle operation and split attention module.

2.1. Network Architecture

As illustrated in Figure 1 , the proposed network adopts CRNN as the backbone architecture. It mainly consists of three parts: multi-scale feature extraction part, bi-directional GRU (Bi-GRU) and lo-

[*] equal contribution with Xiujuan Zhu

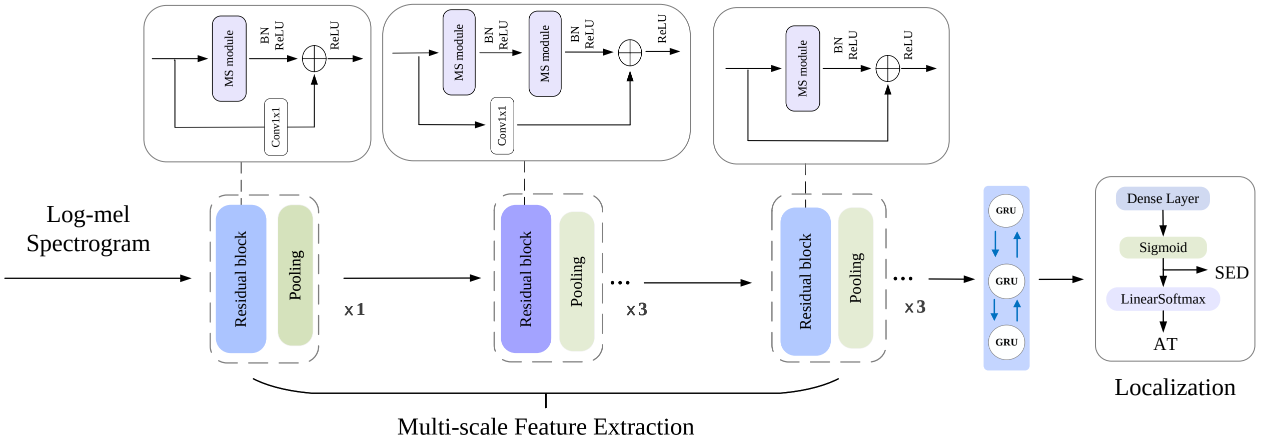

Figure 1: The overall architecture of the Multi-scale network based on Split Attention.

calization parts. The multi-scale feature extraction part is based on 7 residual blocks, each block followed by a pooling layer. In the first one and the last three residual blocks, each of them consists of one multi-scale (MS) module shown in Figure 2, and in the mid three residual blocks, each of them consists of two MS modules. Then a Bi-GRU is used to capture temporal information. The localization part produces frame-level predictions for SED and clip-level predictions for audio tagging (AT). Note that linear softmax [16] is introduced as an aggregation function to produce the clip-level predictions.

2.2. Multi-scale Module

In order to effectively model time-frequency context information, the multi-scale module exploits convolution kernels of different sizes to extract the features of different scales.

As shown in Figure 2 , where a three-branch case is shown, each branch used to learn one single-scale feature map . Thus, multi-scale module can process the input feature at multiple scales in parallel. For a given feature map $X \in \mathbb{R}^{C\times H\times W}$, it firstly undergoes three kinds of scale transformations based on different kernel sizes k_i, thus [X_1, X_2, X_3] are obtained. $X_i \in \mathbb{R}^{C\times H\times W}$represents a specific scale feature can be generated as:

$$X_i = \text{SA}(X, k_i), i = 1, 2, 3 \tag{1}$$

where SA denotes split attention module that is going to be described in details in Section 2.4 . k_i denotes the kernel size used in SA module. Then a pre-processed multi-scale feature $F' \in \mathbb{R}^{3C\times H\times W}$ is obtained by concatenating the multi-scale features X_i:

$$F' = \text{Concat}([X_1, X_2, X_3]) \tag{2}$$

where Concat means the concatenation operation along the channel dimension.

In order to help the network learn a better multi-scale feature, a channel shuffle operation is applied to F' that it improves the information flowing among the features with different scales. A convolution layer with the kernel size of 1×1 is followed to change the channel numbers of output features. Thus, the final output features $F \in \mathbb{R}^{C\times H\times W}$ can be obtained by:

$$F = \text{Conv1} \times 1(\text{CS}(F')) \tag{3}$$

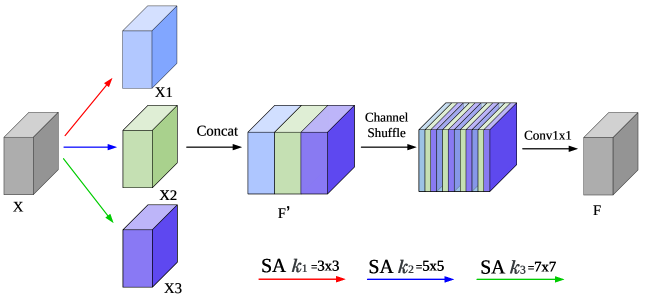

Figure 2: Illustration of our proposed multi-scale (MS) module. SA denotes split attention module.

Where CS denotes channel shuffle operation.

2.3. Channel Shuffle operation

In [17], channel shuffle operation can be used to improve the information flowing among the feature within different groups. In this paper, channel shuffle operation aims to enhance the cross-channel information communication among the features with different scales. A channel shuffle operation can be modeled as a process composed of “Reshape-Transpose-Reshape” operations. As shown in Figure 2 , the channel dimension of F' is reshaped to (g, c), where g is the number of groups, $c = 3C/g$. The channel dimension is further reshaped to (c, g) and then flatten back to $3C$. Through this operation, the channel information among different features can interact with each other.

2.4. Split Attention Module

As shown in Figure 3 , inspired by group convolution (GN) [18], SA module firstly adopts group convolution to learn different sub-features, which represent diverse semantic features such as different sound event patterns. Then, in order to measure the importance of different sub-features, a set of attention weights W_i corresponding to each sub-features are generated. This process can be abstracted into two parts: **Group**, **Attention**.

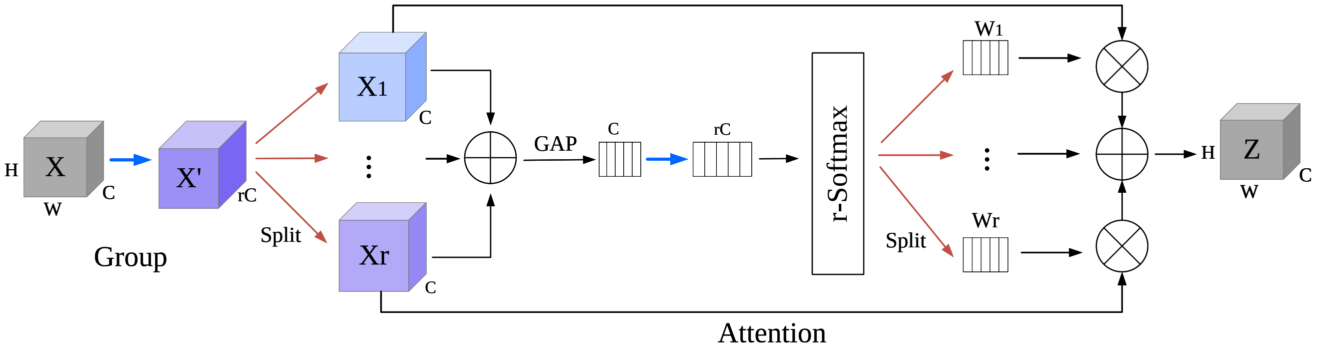

Figure 3: Illustration of the proposed split attention (SA) module. Blue arrow denotes group convolution operation, red arrow split operation along the channel dimension

Group: Assuming an input feature map $X\in \mathbb{R}^{C\times H\times W}$, we firstly adopt the group convolution to learn g sub-features in different groups separately. As a result, the C-channel feature map X is expanded into the rC-channel feature map $X'\in \mathbb{R}^{rC\times H\times W}$. Then the expanded feature map X' is split into r branches along the channel dimension that represented as $[X_1, ..., X_i..., X_r]$. $X_i\in \mathbb{R}^{C\times H\times W}$. $i\epsilon 1$, 2, ..., r. The number of group g and branch r will be discussed in the experiment.

Attention: Multiple sub-features $[X_1, ..., X_i..., X_r]$ are firstly fused via an element-wise summation $U = \sum_{i=1}^{r} X_i$. Then, global average pooling is calculated to squeeze the fused feature U into a channel-wise statistics $S\in \mathbb{R}^{C\times 1\times 1}$:

$$S = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} U(i', j') \tag{4}$$

Then, a simple attention mechanism with a Softmax function is performed on the channel-wise statistics S. The attention weight $W\in \mathbb{R}^{rC\times 1\times 1}$ can be obtained by:

$$W = r - Softmax((Conv(\delta(BN(Conv(S))))) \tag{5}$$

Where δ means the ReLU activation function, BN the batch normalization, $Conv$ the group convolution with the kernel size of 1×1 and group number is g. The attention weight $W\in \mathbb{R}^{rC\times 1\times 1}$ is then splitted into a set of attention weights $W_i\in \mathbb{R}^{C\times 1\times 1}$.

Finally, by applying the weights W_i to the sub-features X_i, the output feature map of SA module Z $\in \mathbb{R}^{C\times H\times W}$ is obtained by:

$$\mathbf{Z} = \sum_{i=1}^{r} S_i \times X_i \tag{6}$$

3. EXPERIMENTS

3.1. Dataset and Experimental setup

The audio samples in the DCASE2021 Task4 dataset are 10s clips recorded in domestic environment or synthesized to simulate a domestic environment. It contains 10 kinds of sound events. Three types data (i.e. the weakly labeled data (1578), unlabeled data (11412) and strong labeled data (10000)) are used for training. The ratio among them is 1:2:1 in each batch. The performance of the proposed method is evaluated on the validation data (1168).

Following the default experiment settings of DCASE2021 Task4 baseline, we also take the log-mel energies as input extracted with 128 mel-scale filters. The window length is 2048 with the hop size of 256. The audio is resampled to 16kHz. The training is set for 200 epochs using the Adam optimizer with an initial learning rate of 0.001. A learning rate exponential warmup [19] during the first 50 epochs is used. A detection threshold is fixed to 0.5 for each class. The binary SED predictions are further processed with a 7 frames median filter. For the 7 residual blocks in multi-scale extraction part, the number of channels for each residual block is [16, 32, 64, 128, 128, 128, 128], respectively and the pooling size is [[1, 2], [1, 2], [2, 2], [2, 2], [1, 2], [1, 2], [1, 2]], respectively. The dropout rate is 0.3. Note that due to the continuous pooling operation along the frequency dimension, its receptive field along the frequency dimension keeps increasing. Thus, in the last three residual blocks, the kernel sizes used in MS module are set to [[3, 3], [5, 3], [7,3]], that is the kernel sizes used in time dimension keep different values, while in frequency dimension the same.

3.2. Loss Function

The loss function for training the model is a sum of four loss components: two binary cross entropy (BCE) losses for supervised training and two mean square error (MSE) losses for consistency training, which are combined as follows:

$$\begin{aligned}\mathcal{L}(\theta) =& \mathcal{L}_{BCE}(sw_{out}, l_w) + \sigma(\lambda)\mathcal{L}_{MSE}(sw_{out}, tw_{out}) \\ & \mathcal{L}_{BCE}(ss_{out}, l_s) + \sigma(\lambda)\mathcal{L}_{MSE}(ss_{out}, ts_{out})\end{aligned} \tag{7}$$

Where sw_{out}, ss_{out} denote the AT output and SED output of the student model, respectively, tw_{out}, ts_{out} the AT output and SED output of the teacher model, l_w and l_s the weakly label and strong label of the labeled data.

3.3. Evaluation metrics

In DCASE2021 Task4, the evaluation metrics include PSDS-scenario1 (PSDS1), PSDS-scenario2 (PSDS2), Intersection-based F1 (IB-F1) and Collar-based F1 (CB-F1). The PSDS1 measures the model's capability of detecting the onset and offset of the event within an audio clip, and the PSDS2 measures that of avoiding confusion among the event classes. More details about PSDS evaluation metrics can refer to [20]. IB-F1 and CB-F1 are used as sup-

Table 1: Ablation experiments on multi-scale (MS) mechanism with different kernel sizes. We adopt vanilla convolution instead of SA module in MS module of all residual block in this experiment.

Network	PSDS1	PSDS2	IB-F1(%)	CB-F1(%)	Parameter
Base-2021	0.342	0.527	76.60	40.10	1.1M
MS-K=[3]	**0.358**	0.599	81.88	**44.48**	1.2M
MS-K=[3,5]	0.349	**0.602**	83.24	44.13	3.0M
MS-K=[3,5,7]	0.336	0.601	**83.50**	42.21	5.7M

Table 2: Ablation experiments on channel shuffle (CS) operation based on MS-K=[3,5] system. CS-g denotes the channel shuffle operation with g groups. g controls the fusion degree of features.

Network	PSDS1	PSDS2	IB-F1 (%)	CB-F1 (%)
MS-K=[3, 5]	0.349	0.602	**83.20**	44.13
+ CS-g=2	0.349	0.594	82.83	43.58
+ CS-g=4	**0.358**	**0.606**	82.98	**45.36**

plementary evaluation metrics to validate a model's performance in SED. For all these metrics, the value larger, the performance better.

4. RESULTS AND ANALYSIS

We separately investigate the contribution of each component to the overall network, including the multi-scale mechanism with different kernel sizes, the channel shuffle operation and the split attention module. All experiments are repeated 4 times and the average result of these experiments is reported.

Evaluations of MS mechanism

Table 1 shows the SED performance of the MS mechanism with difference kernel sizes. MS-K=[3] means there is only one branch with the kernel size of 3×3 in MS module, and MS-K=[3, 5] denotes there is two branches with the kernel sizes of 3×3 and 5×5. MS-K=[3, 5, 7] means exactly the processing depicted in Figure 2 but no channel shuffle operation. Experimental results show that our proposed MS network outperform the baseline of DCASE2021 Task4 [21] in terms of four evaluation metrics, demonstrating the effectiveness of the multi-scale mechanism for SED. However, compared with MS-K=[3], the performance of network applying two types of convolution kernels (MS-K=[3, 5]) or three types of convolution kernels (MS-K=[3, 5, 7]) has barely improved. The reason for this phenomenon may be that the network does not handle the features of different scales well.

Evaluations of CS operation

Table 2 lists the results of our proposed network with channel shuffle operation. In particular, compared with the network without CS operation denoted as MS-K=[3,5], the network applying CS operation with 4 groups achieve a better performance in terms of all evaluation metrics except IB-F1. This result demonstrates the effectiveness of channel shuffle operation.

Evaluations of SA module

Table 3 lists the results of network applied SA module. In this experiment, we only adopt vanilla convolution in MS module of the 1-th residual block, while SA module in MS module of the rest residual blocks. Compared with the results of first row, we can see that the network with split attention module achieves significantly improvement in terms of all evaluation metrics expect PSDS2 metric. The results demonstrate the effectiveness of SA module for SED. However, Table 3 shows that three are no significant difference among networks with different SA module on PSDS2 and IB-F1 metrics. Compared with the results between the second and fourth row or the third fifth row, we can see that the network applying group convolution with 2 group can achieve a better performance on PSDS1 and CB-F1 metrics than without applying group operation. This manifests that adopting group operation to learn sub-features is effective. Compared with the results between the fourth and fifth row, we can find that the performance of network splitting 2 sub-features (r=2) is better than without splitting operation in SA module. This indicates that generating attention weights to treat the learned sub-features differently is important. From the results of the last row, the performance of network applying channel shuffle get further improvements in terms of four metrics except PSDS1. It also shows the effectiveness of CS operation.

Table 3: Ablation experiments on split attention (SA) module based on MS-K=[3,5] system. SA(g, r) means the number of group is g, splitted sub-feaatures r in shuffle attention module.

Network	PSDS1	PSDS2	IB-F1(%)	CB-F1(%)	Parameter
MS-K=[3, 5]	0.349	0.602	83.24	44.13	3.0M
+ SA(1, 1)	0.354	0.598	**84.59**	47.99	3.2M
+ SA(1, 2)	0.350	0.602	84.40	46.64	5.5M
+ SA(2, 1)	0.367	**0.606**	83.80	48.59	1.9M
+ SA(2, 2)	**0.376**	0.599	83.63	49.02	3.3M
+ CS-g=4	0.373	0.602	83.98	**50.28**	3.3M

5. CONCLUSION

In this paper, we propose a multi-scale SED network based on split attention that it can deal with the short- or long- duration sound events. Multi-scale module can learn features with multiple scales in parallel. Specifically, channel shuffle operation is used to promote the cross-channel information flowing among the features with different scales. Split attention module can learn the different sub-features separately and generate attention used to weight the importance of sub-features. A set of experiments are conducted to verify their effective. The final results of the proposed network outperform the baseline of DCASE2021 Task4 significantly. In our future work, we would like to explore the issue that how to deal with the features with different scales in SED.

6. ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China (NSFC) (61761041,U1903213), Tianshan Innovation Team Plan Project of Xinjiang (202101642)

7. REFERENCES

[1] E. Wold, T. Blum, D. Keislar, and J. Wheaton, “Content-based classification, search, and retrieval of audio,” *IEEE Multim.*, vol. 3, pp. 27–36, 1996.

[2] J. P. Bello, C. Mydlarz, and J. Salamon, “Sound analysis in smart cities,” in *Computational Analysis of Sound Scenes and Events*. Springer, 2018, pp. 373–397.

[3] C. Debes, A. Merentitis, S. Sukhanov, M. Niessen, N. Frangiadakis, and A. Bauer, “Monitoring activities of daily living in smart homes: Understanding human behavior,” *IEEE Signal Processing Magazine*, vol. 33, no. 2, pp. 81–94, 2016.

[4] Q. Jin, P. Schulam, S. Rawat, S. Burger, D. Ding, and F. Metze, “Event-based video retrieval using audio,” in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[5] A. Tarvainen and H. Valpola, “Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results,” in *NIPS*, 2017.

[6] J. Zhang, W. Ding, J. Kang, and L. He, “Multi-scale time-frequency attention for acoustic event detection,” *arXiv preprint arXiv:1904.00063*, 2019.

[7] W. Ding and L. He, “Adaptive multi-scale detection of acoustic events,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 294–306, 2019.

[8] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944, 2017.

[9] J. Yan, Y. Song, W. Guo, L. Dai, I. Mcloughlin, and L. Chen, “A region based attention method for weakly supervised sound event detection and classification,” *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 755–759, 2019.

[10] J. Yan, Y. Song, L. Dai, and I. Mcloughlin, “Task-aware mean teacher method for large scale weakly labeled semi-supervised sound event detection,” *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 326–330, 2020.

[11] Y. Li, M. Liu, K. Drossos, and T. Virtanen, “Sound event detection via dilated convolutional recurrent neural networks,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 286–290.

[12] K. Drossos, S. I. Mimilakis, S. Gharib, Y. Li, and T. Virtanen, “Sound event detection with depthwise separable and dilated convolutions,” in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–7.

[13] K. Su, D. Yu, Z. Xu, X. Geng, and C. Wang, “Multi-person pose estimation with enhanced channel-wise and spatial information,” *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5667–5675, 2019.

[14] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, Z.-L. Zhang, H. Lin, Y. Sun, T. He, J. Mueller, R. Manmatha, M. Li, and A. Smola, “Resnest: Split-attention networks,” *ArXiv*, vol. abs/2004.08955, 2020.

[15] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, “Eca-net: Efficient channel attention for deep convolutional neural networks,” *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 531–11 539, 2020.

[16] Y. Wang, J. Li, and F. Metze, “A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 31–35.

[17] X. Zhang, X. Zhou, M. Lin, and J. Sun, “Shufflenet: An extremely efficient convolutional neural network for mobile devices,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6848–6856.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, pp. 84 – 90, 2012.

[19] P. Goyal, P. Dollár, R. B. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, “Accurate, large minibatch sgd: Training imagenet in 1 hour,” *ArXiv*, vol. abs/1706.02677, 2017.

[20] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, “A framework for the robust evaluation of sound event detection,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 61–65.

[21] http://dcase.community/challenge2021/task-sound-event-detection-and-separation-in-domestic-environments.