ENSEMBLE OF CONVOLUTIONAL NEURAL NETWORKS FOR WEAKLY-SUPERVISED SOUND EVENT DETECTION USING MULTIPLE SCALE INPUT

Donmoon Lee[1,2]*, Subin Lee*[1,2]*, Yoonchang Han*[2]*, Kyogu Lee*[1]

[1] Music and Audio Research Group, Seoul National University, Seoul, Korea,
[2] Cochlear.ai, Seoul, Korea,
{dmlee, sblee, ychan}@cochlear.ai, kglee@snu.ac.kr

ABSTRACT

In this paper, we propose to use an ensemble of convolutional neural networks to detect audio events in the automotive environment. Each of the networks is based on various lengths of analysis windows for multiple input scaling. Experiments showed that the structures with tagging different scales are complementary to each other on, i) detecting and ii) localizing sound events, therefore, an effective ensemble results in performance improvements for both tasks. The proposed model, an ensemble of the structures, achieved 0.4762 in the event-based F1-score and 0.7167 in the segment-based error rate on DCASE 2017 in development set. And it achieved 0.536 in the event-based F1-score and 0.66 in the segment-based error rate in evaluation set. Our model accomplished the 2nd place on audio tagging and the 1st place on sound event detection.

Index Terms— DCASE 2017, Weakly-supervised learning, Convolutional neural networks, Sound event detection

1. INTRODUCTION

Sound event detection (SED) aims to find sound objects and events from the audio content. SED has been studied in the contexts of various applications including acoustic scene analysis [1, 2], surveillance [3, 4, 5], health-care monitoring [6], and multimedia analysis [7, 8]. One of the applications is the SED for assisting car drivers which aims to help a driver to acknowledge the surroundings using audio content analysis.

Human drivers use cognitive abilities to recognize the surroundings such as the location and movements of nearby objects, e.g., cars and pedestrians. Obviously, a temporary decline in cognitive abilities reduces driving performance and increases the risk of accidents [9]. Object detection systems have been intensively studied over the years as an assistant system for human drivers. Especially, visual object detection systems have made significant improvements and deployed to the real system. However, visual sensors-based systems are heavily affected by the environmental condition such as lighting, shadows, and reflections, which limits the reliability of the system, therefore motivates using SED for the safer driving.

'Conventional' machine learning techniques have been proposed, e.g., using Mel-frequency cepstral coefficients (MFCCs) and non-negative matrix factorization-based features [10, 11, 12, 13]. Recently, deep learning-based methods such as recurrent neural networks (RNNs) [14] and convolutional neural networks (ConvNets) [15, 16] have been proposed. ConvNets showed the promising results in a number of computer vision tasks and have been actively adopted for audio content analysis such as SED [16] and music related tasks [17, 18].

An effective use of ConvNets on audio signal requires the specialized designs and domain-specific procedures. One of the design choices is the resolution/length of the audio input. Choosing the optimal size of audio input is usually task-specific and, to some extent, arbitrary. For example, a relatively long window (29-second) was used for music [19]. On the other hand, small window turned out to be more suitable for instrument identification [20]. SED also used a small length of window (below 100ms) as the optimal input size [14, 21]. A comprehensive approach is to use a multi-scale input and allows the network to learn to extract relevant information from inputs with various scale selectively [22, 23].

In this paper, we propose a sound event detection system that can recognize strong-labeled sound event from weakly-labeled data. This is a technical paper regarding out submission to the detection and classification of acoustic scene and events (DCASE) 2017 [24], *large-scale weakly supervised sound event detection for smart cars* which aims to simulate the SED problem in the real automotive environment by detecting 17 sound event categories including warning and vehicle sounds. Section 2 describes the proposed SED system. Section 4 shows and discusses the experiment results based on the results on the provided test set. Finally, Section 4 summarizes the final results of the competition.

2. PROPOSED SYSTEM

2.1. DCASE 2017 Dataset

The dataset of DCASE 2017 is a subset of AudioSet [25] that contains 17 warning and vehicle sounds that are related to the automotive environment. The dataset is divided into a training set and a test set, each with 51,172 and 488 audio clips. Each data sample may correspond to more than one sound event, and a binary decision is made for each class, i.e., the task is a multi-label classification problem. The audio signal is mono-channel and sampled at 44,100 Hz with a maximum duration of 10 seconds. The development set has only weak labels, i.e., only the presence of a given sound event is labeled without the exact time stamps while the test set is strongly labeled with both the categories of the existing sound events and their timestamps. There is a heavy class imbalance in the data set. The numbers of positive labels of the classes are between 180 and 25,077 and summarized in Fig. 1.

2.2. Audio Preprocessing

The amplitudes of the audio signal are normalized to the full-range. There are 10,785 signals that are shorter than 10-second, and they are zero-padded to equalize the length. 14 signals are excluded from the training set since they contain nothing. For separated-model, the

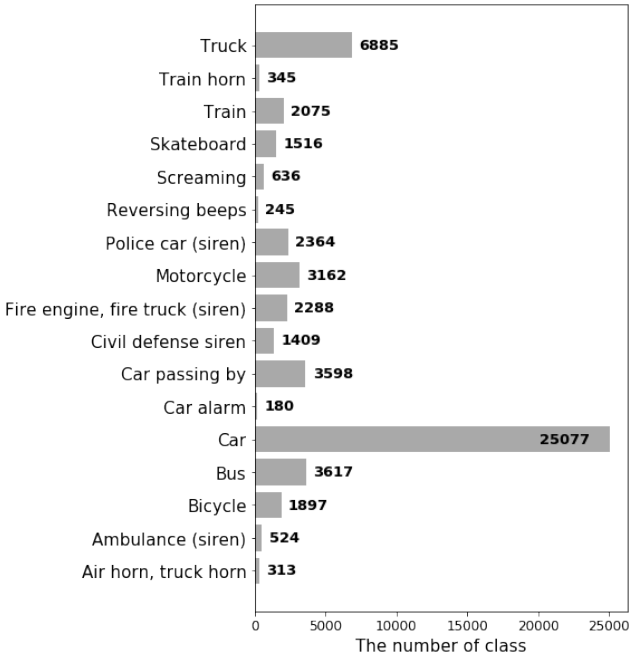

Figure 1: The class distribution in SED development set

waveform is segmented by 44,100 frames (1-second). It is chosen because 1-second is presumably long enough to contain a complete single sound event. The signal is converted into Mel-spectrogram with 2,048 FFT points (46 ms), 128 mel-bins, then its magnitudes are logarithmically mapped, i.e., $X \rightarrow \log 10X$. To simplify network design, we use the hop size of 431 and 460 for the global-input model and the separated-input model, respectively.

2.3. Background Noise Removal

An additional step in the audio preprocessing is performed to remove the background noise and enhance the target sound event in the Mel-spectrogram. For each Mel-spectrogram, the 128 median values are computed along the time axis and subtracted from it. It is known to have the effect of eliminating low-frequency background drift in continuous signals [26].

2.4. Network Architecture

The proposed system uses multiple models to predict audio events in a short-time segment. There are two networks: global-input model and separated-input model. It depends on whether the model uses the entire or a segmented audio clip. The system outline is illustrated in Fig. 2.

2.4.1. The global-input model

The details of the global-input model structure are illustrated in Fig. 3. It uses a 10-second waveform as input. It then converted to Mel-spectrogram with the shape of (1, 128, 1024) which correspond to the numbers of channels, mel-bins, and the frame. We use the homogeneous 2D (3×3) convolutional filters with the same number of feature map, 64, to form a fully convolutional network structure, which is similar to [19]. The double_conv block and the max-pooling layer alternates, learning features while reducing the sizes of the feature maps. The double_conv block is a stack of two sets of a convolution layers, batch normalization, and a ReLU (rectified linear unit) activation function. The global average pooling layer follows after the last convolution block. The output layer is a densely-connected layer with the sigmoid activation function since it is a multi-class classification problem. The position of batch normalization follows the recent study in [27]. The weights are initialized using 'He normal' [28].

In the training, Adam optimizer [29] is used for an adaptive learning rate control. We allocate 15 % of the development set as a validation set and the final model is selected based on the validation set performance.

2.4.2. The separated-input model

As mentioned earlier, the separated-input model predicts the occurrence of sound events in a short audio segment. It uses a n-second segmented waveform as input. It then converted to Mel-spectrogram with shape of (1, 128, $96 \times n$). The network structure is similar to that of the global-input model with changing the subsampling sizes as in Fig. 3.

Multiple models of the same structure are trained and correspond to inputs of 1, 2, 3, 4, and 5-second waveform with a 1-second sliding window. All the segments that make up the same clip are considered to have the same label. The other settings for training are the same as for the global-input model.

2.5. Predict Time Stamps

The proposed system is designed to predict the sound event probability of a given audio clip in seconds. This procedure primarily uses separated-input models. An input audio which has 10-second lengths is divided into pieces, and each segment is used in the separated-input model. Results from separated-input models are then converted to sound event occurrence probability matrix with the shape of (17×10) which correspond to the kind of events, and the time in seconds. When the length of input segment is 1, each result from an input segment is considered to the probability at that time window. If the length of input audio is longer than 1-second, a specific one-second can be contained input segment multiple times. That is, for each one-second, the system can have a maximum n prediction (for n-second of input). We then average all possibilities and determine the existence of the event in that one-second. Once the sound event occurrence probability matrix has been made, we can easily mix multiple models to predict timestamps. The ensemble of the individual models is computed by averaging the probabilities of that time.

The global-input model is expected to have higher performance because it uses the entire audio clips with the correct label. However, timestamps cannot be predicted using the global-input model alone. We use the global-input model in two ways. Firstly, we use it in the same way with the separated-input model (*ClipAvg*). In this case, it is assumed that predictions from the global-input model are spread evenly across the 10-second time windows. The probability is then averaged together with other models above. Also, we use the global-input model as a sound event detector and detect the location using separated-input model only for clips where the event occurs (*ClipGate*).

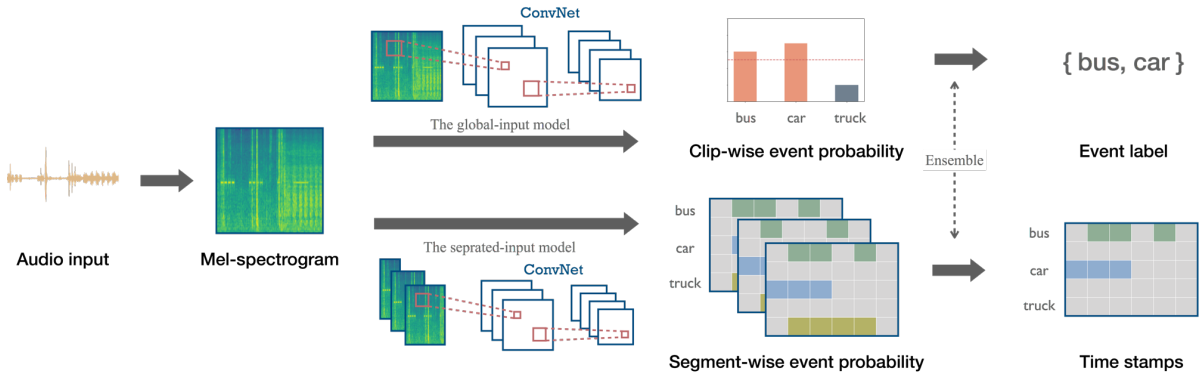

Figure 2: The overall system architecture of the proposed system. The global-input model takes the entire audio signal as an input and predicts the presence of an event in the signal while the separated-input model learns to find the presence of an event for given small segments. The final prediction is then given by ensembling both models probabilities.

The global-input model (data shape)

Audio Input	(1, 441000)
Mel-spectrogram	(1, 128, 1024)
Double_Conv. block	
4 x 4 Max-pooling	(64, 32, 256)
Double_Conv. block	
4 x 4 Max-pooling	(64, 8, 64)
Double_Conv. block	
2 x 4 Max-pooling	(64, 8, 16)
Double_Conv. block	
2 x 4 Max-pooling	(64, 4, 4)
Double_Conv. block	
GlobalAveragePooling	(1024)
Output	(17)

Double_Conv. Block

BatchNorm.
ReLU
3 x 3 Conv. (64)
BatchNorm.
ReLU
3 x 3 Conv. (64)

The separated-input model (data shape)

Audio Input	(1, 44100 x n)
Mel-spectrogram	(1, 128, 96 x n)
Double_Conv. block	
4 x 4 Max-pooling	(64, 32, 24 x n)
Double_Conv. block	
4 x 3 Max-pooling	(64, 8, 8 x n)
Double_Conv. block	
2 x 2 Max-pooling	(64, 4, 4 x n)
Double_Conv. block	
2 x 2 Max-pooling	(64, 2, 2 x n)
Double_Conv. block	
GlobalAveragePooling	(256)
Output	(17)

Figure 3: The detailed network structure of the global-input model and the separated-input model.

2.6. Ensemble Method

We apply the ensemble selection method to find the optimal combination of learned models, expecting a better combination than the empirically chosen one. The ensemble selection algorithm method proposed by Caruana et al. [30] is used since we can apply it to the probability matrix that our system uses in the ensemble procedure. It works by repeating iterations and adding a model that maximize performance at that point.

We think that timestamped data is insufficient and fitting too much into small data makes model vulnerable. Therefore, ensemble selection is performed for the entire test data. We used F1 or ER as the performance metric to choose the weights specialized to each subtask. In addition, we used F1-ER as the performance metric, because the process that satisfies both tasks is expected to work as a kind of regularization.

2.7. Evaluation Measures

In DCASE challenge, the performances of classification and detection are evaluated by the event-based F1-score and the segmented-based error rate, respectively. For the classification of the whole 10-second audio signal, F1-score is used:

$$F_1 = \frac{2 \cdot P \cdot R}{P + R} \tag{1}$$

, where P and R indicate the precision and recall respectively.

To evaluate the detected time stamps of sound events, the error rate (ER) is calculated in one-second segments over the entire test set. ER computes the percentage of all types of errors in every 1-second subsegment. For each data sample, ER is computed as below:

$$ER = \frac{S + D + I}{N} \tag{2}$$

, where N corresponds to the total number of event segments in the ground truth and S, D, I correspond to substitution/deletion/insertion errors.

3. EXPERIMENT RESULTS AND DISCUSSIONS

The experimental results are summarized in Table 1. For the audio tagging, the 10-second input model achieved the highest F-1 score. It suggests that for the global classification, using the entire audio in a single model and allowing the network to aggregate the prediction works better than manually aggregating the predictions from models with the shorter inputs. Among the models with various segment lengths, 3-second achieved the best performance, suggesting there exists the most suitable duration, probably depending on the types and intervals of the sound event.

Background subtraction improved the tagging performance of most systems, but a significant degradation was observed in the 5-second and 10-second input models, implying that our approach is not suitable for long time windows. The effect on the error rate is not clear, since the performance may be improved or decreased. However, by combining multiple separated-input models with or without BS, our system showed improved error rate than single models. The combined systems with 3, 4 s input models and 10-second input model (*ClipAvg*), show up to 0.7167 error. We assume that although the model with background subtraction shows similar error rates, behave differently, improving ensemble performance.

The result of ensemble selection is denoted in Fig. 4. The weight is used for the mean probability calculation. It could be interpreted as a kind of importance for each model. In this context, we

Networks	Subtask A *F-1*	Subtask B *ER*
Baseline (MLP)	.1310	1.0200
10-second input (w/BS)	**.4745** (.3378)	-
1s-segmented input (w/BS)	.4125 (.4373)	.7963 (.8362)
2s-segmented input (w/BS)	.4229 (.4316)	.8071 (.8007)
3s-segmented input (w/BS)	.4538 (.4561)	**.7546** (.7610)
4s-segmented input (w/BS)	.4304 (.4313)	.7633 (.7718)
5s-segmented input (w/BS)	.4335 (.3588)	.8028 (.8431)
MeanProb of 5 models (w/BS)	.4408 (.4448)	.7667 (.7688)
MeanProb of 10 models	.4430	.7475
ClipAvg in 5 best models	**.4762**	**.7167**
ClipGate in 5 best models	.4745	.7287
*Ensemble selection (*F1*)	.5139	.7477
*Ensemble selection (*ER*)	.4831	.7021
*Ensemble selection (*F1-ER*)	.4885	.7089

Table 1: SED performance on the test set. The performance of 12 single models is listed with multiple input scales and with background subtraction (BS). MeanPorb model results using the mean probabilities of ns-segmented input models with and without BS. *ClipAvg* and *ClipGate* are the result using 5 best models (a 10-second input model and the 3 and 4-second input model with and without BS). Ensemble selection algorithm used the performance metric in a bracket. Note that the result with * used test label which should not be directly compared to other approaches.

Networks	Subtask A *F-1*	Subtask B *ER*
Baseline (MLP)	.182	.930
ClipAvg in 5 best models	.523	.670
ClipGate in 5 best models	.523	.670
Ensemble selection (*F1*)	**.526**	-
Ensemble selection (*ER*)	-	.670
Ensemble selection (*F1-ER*)	.521	**.660**

Table 2: The results of our system in the DCASE 2017 competition.

can again guess the effect of background subtraction. The 5-second and 10-second models showed very low weights when using background subtraction, but the 1s-segmented models (w/BS) showed higher weights, although the lower performance. It suggests that the background subtraction works in a time window that is not too long.

4. DCASE2017 SUBMISSIONS AND RESULTS

Our submission 1, 2, and 4 used the same ensemble model for subtask A and B, only submission 3 used distinct models for subtask A and B. Details of submission are as follow: Submission 1 and Submission 2 are the ensembles of the top five models with the using of *ClipAvg* and *ClipGate*, respectively. Submission 3 and 4 are results of the ensemble selection method. We apply the ensemble selection to the entire 12 single models to find the best combinations of weights. Submission 3 is the result of ensemble selection over-fitting to test data specific to audio tagging and sound event detection, respectively. Submission 4 is the result of an ensemble selection suitable for both tasks.

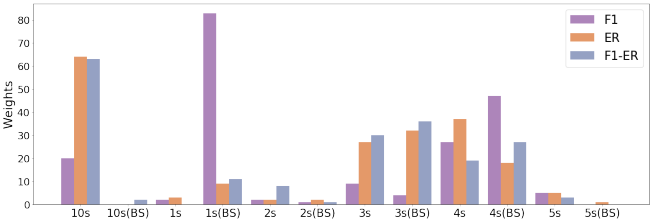

Figure 4: The weights of single models according to the performance metric of the ensemble selection.

There is no big performance difference between the submission when compared to the development set of the competition results. It suggests that the strategy that prevents overfitting into small data in ensemble method is valid, and the ensemble selection procedure does not significantly affect our system. In the DCASE competition, Submission 3 achieved second prize on audio tagging and Submission 4 achieved first prize on sound event detection.

5. CONCLUSIONS

In this paper, we used the ensemble of ConvNets with multiple analysis windows for the SED task. We segmented audio with duplicated labels to find the timestamps of weakly labeled data. The global-input model is superior to other single models when detecting the presence of the sound event in the entire audio clip, but there is a limitation to analyzing a small time window. Therefore, our system mixed the results from the global-input and separated-input models to predict the timestamps of the input audio and minimize errors using the ensemble selection methods.

We believe that there are potential improvements in our work. 1) In our experiments, the background subtraction is implemented on the entire time axis of the input audio, while it has more advantages in the short-time window. The ensemble of various models using short-time background subtraction can lead to improvements. 2) Experimental results show that the segmented-input is still useful for SED tasks, but all models in this study used the same structure regardless of the input shape. We think that using tailored network structures for analysis window in different lengths can improve the performance further.

6. ACKNOWLEDGEMENTS

We thank Keunwoo Choi for helpful comments that greatly improved the manuscripts. This research was supported by Korean government, MSIP provided financial support in the form of Bio & Medical Technology Development Program (2015M3A9D7066980).

7. REFERENCES

[1] D. Giannoulis, E. Benetos, D. Stowell, M. Rossignol, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events: An ieee aasp challenge," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.

[2] A. Mesaros, T. Heittola, and T. Virtanen, "Tut database for acoustic scene classification and sound event detection," in *Signal Processing Conference (EUSIPCO), 2016 24th European*. IEEE, 2016, pp. 1128–1132.

[3] S. Ntalampiras, I. Potamitis, and N. Fakotakis, "On acoustic surveillance of hazardous situations," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 165–168.

[4] C. Clavel, T. Ehrette, and G. Richard, "Events detection for an audio-based surveillance system," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 2005, pp. 1306–1309.

[5] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*. IEEE, 2007, pp. 21–26.

[6] Y.-T. Peng, C.-Y. Lin, M.-T. Sun, and K.-C. Tsai, "Healthcare audio event classification using hidden markov models and hierarchical hidden markov models," in *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*. IEEE, 2009, pp. 1218–1221.

[7] D. Zhang and D. Ellis, "Detecting sound events in basketball video archive," *Dept. Electronic Eng., Columbia Univ., New York*, 2001.

[8] M. Xu, C. Xu, L. Duan, J. S. Jin, and S. Luo, "Audio keywords generation for sports video analysis," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 4, no. 2, p. 11, 2008.

[9] K. J. Anstey, J. Wood, S. Lord, and J. G. Walker, "Cognitive, sensory and physical factors enabling driving safety in older adults," *Clinical psychology review*, vol. 25, no. 1, pp. 45–65, 2005.

[10] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Signal Processing Conference, 2010 18th European*. IEEE, 2010, pp. 1267–1271.

[11] X. Zhuang, X. Zhou, M. A. Hasegawa-Johnson, and T. S. Huang, "Real-world acoustic event detection," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1543–1551, 2010.

[12] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation," in *Machine Listening in Multisource Environments*, 2011.

[13] A. Mesaros, T. Heittola, O. Dikmen, and T. Virtanen, "Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 151–155.

[14] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 6440–6444.

[15] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on*. IEEE, 2015, pp. 1–6.

[16] H. Zhang, I. McLoughlin, and Y. Song, "Robust sound event recognition using convolutional neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 559–563.

[17] J. Schluter and S. Bock, "Improved musical onset detection with convolutional neural networks," in *Acoustics, speech and signal processing (icassp), 2014 ieee international conference on*. IEEE, 2014, pp. 6979–6983.

[18] W. Zhang, W. Lei, X. Xu, and X. Xing, "Improved music genre classification with convolutional neural networks." in *INTERSPEECH*, 2016, pp. 3304–3308.

[19] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *The 17th International Society of Music Information Retrieval Conference, New York, USA*. International Society of Music Information Retrieval, 2016.

[20] Y. Han, J. Kim, K. Lee, Y. Han, J. Kim, and K. Lee, "Deep convolutional neural networks for predominant instrument recognition in polyphonic music," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 25, no. 1, pp. 208–221, 2017.

[21] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[22] J. Lee and J. Nam, "Multi-level and multi-scale feature aggregation using pre-trained convolutional neural networks for music auto-tagging," *arXiv preprint arXiv:1703.01793*, 2017.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *European Conference on Computer Vision*. Springer, 2014, pp. 346–361.

[24] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "Dcase 2017 challenge setup: Tasks, datasets and baseline system."

[25] J. F. Gemmeke, D. P. 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 ICASSP*, 2017.

[26] A. W. Moore Jr and J. W. Jorgenson, "Median filtering for removal of low-frequency background drift," *Analytical chemistry*, vol. 65, no. 2, pp. 188–191, 1993.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[28] ——, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

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

[30] R. Caruana, A. Niculescu-Mizil, G. Crew, and A. Ksikes, "Ensemble selection from libraries of models," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 18.