SOUND EVENT DETECTION WITH SEQUENTIALLY LABELLED DATA BASED ON CONNECTIONIST TEMPORAL CLASSIFICATION AND UNSUPERVISED CLUSTERING

Yuanbo Hou[1], *Qiuqiang Kong*[2], *Shengchen Li*[1] *and Mark D. Plumbley*[2]

[1] Beijing University of Posts and Telecommunications, Beijing, P. R. China
[2] Centre for Vision, Speech and Signal Processing, University of Surrey, UK
{hyb, shengchen.li}@bupt.edu.cn, {q.kong, m.plumbley}@surrey.ac.uk

ABSTRACT

Sound event detection (SED) methods typically rely on either strongly labelled data or weakly labelled data. As an alternative, sequentially labelled data (SLD) was proposed. In SLD, the events and the order of events in audio clips are known, without knowing the occurrence time of events. This paper proposes a connectionist temporal classification (CTC) based SED system that uses SLD instead of strongly labelled data, with a novel unsupervised clustering stage. Experiments on 41 classes of sound events show that the proposed two-stage method trained on SLD achieves performance comparable to the previous state-of-the-art SED system trained on strongly labelled data, and is far better than another state-of-the-art SED system trained on weakly labelled data, which indicates the effectiveness of the proposed two-stage method trained on SLD without any onset/offset time of sound events.

***Index Terms*—** Sound event detection, sequentially labelled data, convolutional recurrent neural network, connectionist temporal classification, unsupervised clustering

1. INTRODUCTION

Sound event detection (SED) aims to detect the class of acoustic events with the exact onset and offset time for the events. Classical applications of SED techniques include home monitoring and public security surveillance [1].

Many SED methods typically rely on strongly labelled data, also known as frame level labelled data [2, 3]. In strongly labelled data, each audio clip is labelled with both types of events in the audio clip and the onset/offset time of events. Based on strongly labelled data, the baseline system in [4] feeds a block of frames into a convolutional neural network (CNN) to learn high-level features and recurrent neural network (RNN) to learn temporal information. One of the classical SED systems proposed by Adavanne and Virtanen [5] (referred as *A&V* system in the context) uses stacked convolutional and recurrent neural network as the main architecture and predicts labels at the frame level with a median filter used. Another type of SED method is based on weakly labelled data, also known as clip level labelled data [6, 7]. In weakly labelled data, each audio clip is labelled with one or several tags of events in the audio clip without indicating the occurrence time and order information of events. Since no frame level information of sound events is provided in weakly labelled data, the whole audio clip is usually fed into models without dividing the clip into blocks [8]. Using weakly labelled data, another classical SED system proposed by Xu et al. [7] (referred as *XKWP* system in the context) uses intermediate variables of the model to infer the temporal locations of sound events to complete SED tasks. Due to the lack of frame level information, the SED algorithms with weakly labelled data cannot achieve a comparable performance with SED algorithms with strong labels.

Labelling strongly labelled data is time-consuming and labor expensive, so the size of strongly labelled dataset is often limited to a few minutes or hours [6]. Though there are many weakly labelled datasets on the Internet, they are difficult to use in SED due to insufficient temporal information. Thus we proposed sequentially labelled data (SLD) [9] inspired by the label sequence in speech recognition [10], where the sound events and the order of events are known in audio clips, without knowing the onset/offset time of events. With the help of connectionist temporal classification (CTC), SLD has been successfully applied for audio tagging [9, 11].

Previous work [12] used CTC to solve SED problem, using time boundaries of events as labels. However, the time positions of events predicted by CTC are not close to the actual time boundaries when time boundaries labels sequences are used in the training phase. To solve this problem, [12] uses the exact time boundaries of events as hints for the CTC model to find the actual time boundaries positions, which means strongly labelled data is used in [12] rather than SLD.

In this paper, we extend our previous works [9, 11] to do SED with SLD. The difficulty of solving SED problem based on SLD is to learn the location of time positions of different sound events in audio clips from the audio data without any onset/offset time of sound events.

This paper proposes a CTC-based SED system that uses SLD instead of strongly labelled data, with a novel unsupervised clustering stage. In Stage 1, sequential audio tagging is applied based on CTC using SLD to detect what events happen in audio clips and the order of events. In Stage 2, for each audio clip, frames of bottleneck features from the model trained in Stage 1 are clustered into either *background* cluster or *foreground* cluster using unsupervised clustering. The novelty of the proposed method is that event activity frames are obtained from the *foreground* cluster without using strongly labelled data. Then, combining sequential tags and events activity frames, the SED task can be completed.

This paper is organized as follows, Section 2 introduces the two-stage method in detail. Section 3 describes the dataset, baseline, experimental setup and analyzes the results. Section 4 gives conclusions.

2. TWO-STAGE DETECTION METHOD

A two-stage method is proposed for the SED problem. Stage 1 detects what events happened in audio clips and the order of events. In Stage 1, the events and the order of events in an audio clip are known, while their occurrence time remains unknown. Stage 2 detects the event activity frames, in which the event activity frames in an audio clip are known, without knowing what events are in the frame. Therefore, SED can be performed by combining the sequential tags in Stage 1 with the event activity frames in Stage 2, as shown in Fig. 1.

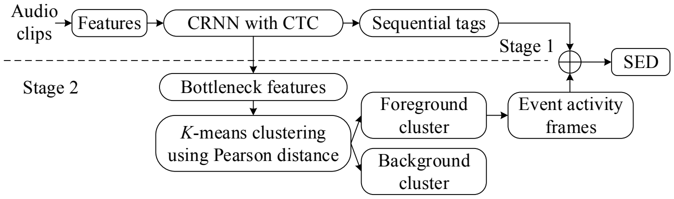

Fig. 1: Block diagram of the proposed method.

2.1. Stage 1: Sequential audio tagging with SLD

To detect what events happen in audio clips and the order of events, sequential audio tagging using SLD is proposed in Stage 1. For the good performance of convolutional recurrent neural network (CRNN) in audio tagging [8], the CRNN is used as the basic classification model in Stage 1. For sequential audio tagging using SLD, CTC is used to keep the sequential information of events in model prediction. CTC [13] redefines the loss function of a recurrent neural network (RNN) and allows the RNN to be trained for sequence-to-sequence tasks, without requiring any prior alignment between the input and target sequences *i.e.* the starting and ending time of sound events. As a result, it is sufficient to do audio tagging with SLD based on CRNN-CTC model.

Fig. 2 shows the basic CRNN model trained with the CTC loss function. The waveforms of audio clips are converted to log mel spectrograms. Convolutional layers are applied to learn local shift-invariant patterns from features. To preserve the time resolution of the input, pooling is applied to the frequency axis only [5]. Bidirectional gated recurrent units (BGRU) [14] are adopted to capture the temporal context information. The final prediction layer has (N+1) units, where N is the number of sound event classes and the extra '1' indicates the blank label for CTC loss function [13].

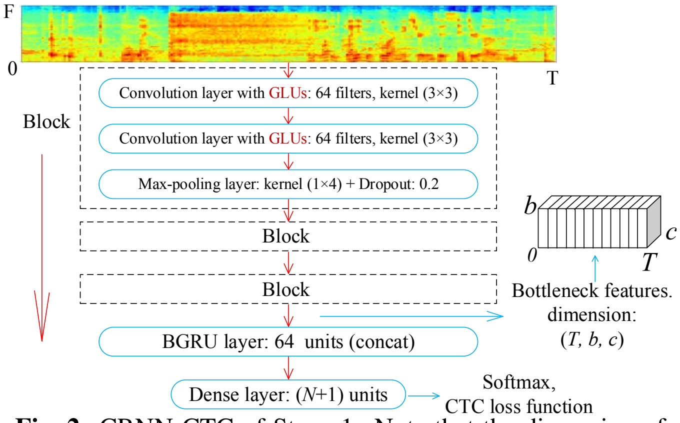
F
0
T
Block
Convolution layer with GLUs: 64 filters, kernel (3×3)
Convolution layer with GLUs: 64 filters, kernel (3×3)
Max-pooling layer: kernel (1×4) + Dropout: 0.2
Block
Block
BGRU layer: 64 units (concat)
Dense layer: (N+1) units
Softmax, CTC loss function
b
c
0
T
Bottleneck features. dimension: (T, b, c)

Fig. 2: CRNN-CTC of Stage 1. Note that the dimension of bottleneck features used in Stage 2 is (T, b, c), T is the number of input frames, b and c denote the frequency bins and the channel number of feature maps, respectively.

To reduce the gradient vanishing problem in deep networks, gated linear units (GLUs) [15] are proposed to replace the ReLU [16] activation in the CRNN model. These provide a linear path for gradients propagation while keeping nonlinear capabilities through the sigmoid operation [15]. GLUs can control the amount of information from a unit that flows to the next layer by sigmoid function. Given W and V are convolutional filters, b and c are biases, X denotes the input features in the first layer or the feature maps of the interval layers and σ is sigmoid function, the GLUs can be defined as:

$$Y = (W * X + b) \odot \sigma(V * X + c) \tag{1}$$

where the symbol $\odot$ is the element-wise product and $*$ is the convolution operator. Another benefit of using GLUs is that network can learn to attend to sound events and ignore the unrelated sounds. If the value of sigmoid function is close to 1, then the corresponding Time-Frequency unit is attended.

2.2. Stage 2: Unsupervised clustering

To obtain the event activity frames of each audio clip, the bottleneck features of each audio clip from Stage 1 are clustered to two clusters: a *background* cluster and a *foreground* cluster. The *background* means the background acoustic scene of the audio clip, and *foreground* cluster is regarded as a cluster of multiple sound events in the audio clip.

For each audio clip, suppose F is the bottleneck feature output from the CRNN trained in Stage 1, where $\{F_1, ..., F_T\}$ are the frames of F. For each frame F_i, if it contains target sound events, it will be regarded as a *foreground* frame; otherwise, it will be regarded as a *background* frame. Since there are no frame level labels in SLD, an unsupervised K-means clustering algorithm [17] is used to obtain the *background* cluster and *foreground* cluster from the bottleneck features of each audio clip, which means K equals 2 in K-means. Most K-means algorithms use Euclidean distance, also known as L_2 norm. However, Euclidean distance is sensitive to outliers [18]. To better measure the distance among frames, Pearson distance [19] is used as an alternative distance function in clustering. The performance of clustering based on Euclidean distance and Pearson distance will be investigated in experiments. The Pearson distance between two frames can be defined as:

$$d_{pearson}(F_m, F_n) = 1 - \rho(F_m, F_n) \tag{2}$$

where ρ is the Pearson Correlation Coefficient (PCC) [20] of m-th frame and n-th frame. Considering that the PCC falls between $[-1, 1]$, Pearson distance lies in [0, 2].

After unsupervised clustering, there are two clusters of frames. In real life, acoustic scene of audio recording is unlikely to change too suddenly or frequently, but different sound events in an audio clip may vary greatly. Therefore, the distance among frames of *background* cluster in the audio clip should be smaller, the *background* cluster should be more compact than the *foreground* cluster. To evaluate which cluster is more compact, the average Euclidean distance is calculated between each frame of the cluster and the cluster centroid, and the smaller one is regarded as the *background* cluster. Given frames $\{F_1, ..., F_n\}$ in cluster C, the average Euclidean distance is calculated by:

$$d_{avg} = \sum\nolimits_{i=1}^{n} d(F_i, p)/n \tag{3}$$

where d is the Euclidean distance and p is the cluster centroid.

By comparing the average Euclidean distance of two clusters in an audio clip, the cluster that has smaller average Euclidean distance is regarded as *background* cluster, and another is *foreground* cluster. From the *foreground* cluster, the frames of acoustic events are detected. Note that the event activity frames are obtained from *foreground* cluster, we only know that there are sound events in the event activity frames, without knowing what it is in each frame.

2.3. Combining sequential tags and event activity frames

Given $\{F_1, ..., F_N\}$ are the acoustic event frames detected by *foreground* cluster in Stage 2, $\{S_1, ..., S_K\}$ are used to present the event spikes sequence S in an audio clip whose corresponding time positions are presented as $\{Ts_1, ..., Ts_K\}$, respectively. The spikes S usually occur near the maximum posterior probability of the events, which is located within the period of acoustic event occurrence [13]. Therefore, the distance between each spike time position Ts_j and each activity frame index F_i is calculated, and each activity frame is assigned to the nearest event spike, which means the label S_j of each activity frame F_i can be defined as:

$$S_j = \arg\min_j \mid F_i - Ts_j \mid, j = 1, ..., K; i = 1, ..., N \tag{4}$$

At this point, both the event activity frames of audio clips and the classes of events in each frame are identified, which marks the completion of SED tasks.

3. EXPERIMENTS AND RESULTS

3.1. Dataset, Baseline and Experiments Setup

Previous work [12] tested the CTC method in SED on 17 types of sound events. To evaluate our proposed two-stage method on more types of sound events, the DCASE 2018 Task 2 dataset [21] is used in this paper. Task 2 contains 41 kinds of sound events from *Freesound*, and is larger than the datasets in other DCASE tasks [4, 22, 23]. These sound events are remixed with acoustic scenes into 10-second audio clips, where each audio clip contains 2 to 4 sound events mixed with TUT Urban Acoustic Scenes recordings [24]. The source code for synthesizing data can be found at [25], and the signal-to-noise ratio (SNR) of synthesizer is 0 dB. The mixed target events are non-overlapped. Target events and non-target events in acoustic scenes overlap.

For baselines, the *A&V* system [5] is trained on strongly labelled data and *XKWP* system [7] is trained on weakly labelled data. The proposed method is trained on SLD. SLD is derived from the strongly labelled data following [9], using the sequence of events as labels. For polyphonic audio clips such as *a dog barks while the bell rings*, although the two events overlap, the label of this audio clip can still be (*ringing*, *bark*). These methods are evaluated using the synthetic large-scale dataset totalling 33.4 hours.

In the training phase, log mel-band energy is extracted using STFT with a Hamming window of 64 ms length, which has a sufficient time and frequency resolution of spectrum. The overlap of 50% between the window is used to smooth the spectrogram. Then 64 mel filter banks are applied [25]. Dropout and Early-stopping are used to prevent over-fitting. The Adam [26] optimizer is used with a learning rate of 0.001. Four-fold cross-validation is used for model selection.

3.2. Results and Analysis

The metrics of *Precision* (P), *Recall* (R), *F-score* and *Error rate* (ER) [27] are based on segments of 1 second length. Higher P, R, *F-score* and lower ER indicate better performance. The audio tagging results can show the performance of CRNN model in Stage 1, and also indirectly reflect the quality of the bottleneck features learned by the CRNN. Only if the model learns better high-level acoustic features of audio clips, will the model better detect sound events. To provide an intuition on how well the approach performs on individual classes, the audio tagging accuracy is shown in Fig. 3. Due to the limitation of space, 11 classes of events are randomly selected in Fig. 3, for the detailed results of all 41 classes of events, please see here[1].

In Fig. 3, events such as ‘bark’ and ‘jangling’ have 100% classification accuracy but some classes like ‘squeak’ have

[1] https://github.com/moses1994/SED_based_on_SLD

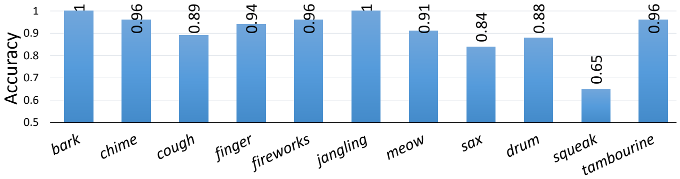

Fig. 3: Audio tagging class-wise accuracy.

poor performance. A reason may be that ‘squeak’ sound varies, depending on the objects caused them. In detail, we found that ‘squeak’ and ‘bass’ are often confused with many other classes, and ‘fart’ is often confused with ‘sax’. Another interesting pair is ‘computer keyboard’ and ‘drum’, it seems reasonable to confuse them as they sometimes sound to be similar. Confusion matrix for all 41 classes of events is available online[1]. Averaged *P*, *R* and *F-score* of audio tagging is *95.03%*, *88.81%* and *91.81%*, respectively. We see that the CRNN-CTC trained on SLD performs well in audio tagging.

To evaluate the classification accuracy of the *background* cluster and *foreground* cluster based on Euclidean distance and Pearson distance, the classification precision at the cluster level is calculated. According to the frame level ground-truth test data labels, the classification precision of unsupervised clustering based on Euclidean distance and Pearson distance is *70.33%* and *88.62%*, respectively. Consequently, the clustering results based on Pearson distance are used for SED.

For SED, the proposed method (referred as *HKLP*) is compared with *A&V* and *XKWP*. As shown in Table 1, the proposed method trained on SLD achieves performance close to *A&V* and is far better than *XKWP*. The standard deviation (*SD*) of *ER* for all 41 classes of *A&V*, *XKWP* and the proposed method is *0.24*, *0.22* and *0.11*, respectively. Although the *ER* of the proposed method is *10%* worse than the *A&V* trained on strongly labelled data, the proposed method reduced the standard deviation of *ER* by half. The proposed method is more stable for 41 different class events.

The SED class-wise *ER* are shown in Fig. 4, for all 41 classes of events are available here[1]. For some classes, the performance of our method (green) is similar to *A&V* (grey). For others like ‘bark’, ‘cough’ and ‘finger’, our method is better. For overall evaluation in Table 1, lower *ER*, *deletion (D) rate*, *insertion (I) rate* and *substitution (S) rate* [27] indicate better performance. *D rate* means the ratio of events that are not recognized to the total events in the ground-truth. In Table 1, the *D rate* is the main error rate in the overall *ER* of the proposed method. To study the reason for this phenomenon, the bottleneck features and event activity frames in an audio clip are shown in Fig. 5. Note that the *SNR* of synthesizer is 0 dB, so there are many other non-target events in the spectrogram. The event spike sequence is correct, but the event activity frames do not match well with the ground-truth red lines. Some frames where events occur are not recognized by the unsupervised clustering in Stage 2. This may be the reason for *D rate* is the main error rate in the *ER* of the proposed method.

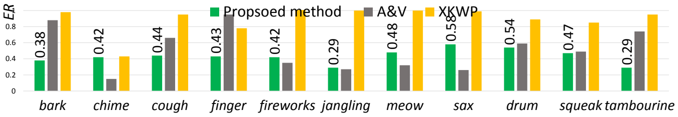

Fig. 4: The results of *ER* in frame level, the specific value of the proposed method is shown. Lower *ER* indicates better.

Table 1: Evaluation of SED among methods for 41 classes.

	ER	*D rate*	*I rate*	*S rate*	*F-score*	*SD of ER*	*Label type*
HKLP	0.46	**0.303**	0.101	0.058	70.98%	**0.11**	*sequential*
A&V	**0.40**	0.347	**0.030**	**0.021**	**75.05%**	0.24	*strong*
XKWP	0.94	0.791	0.102	0.044	25.02%	0.22	*weak*

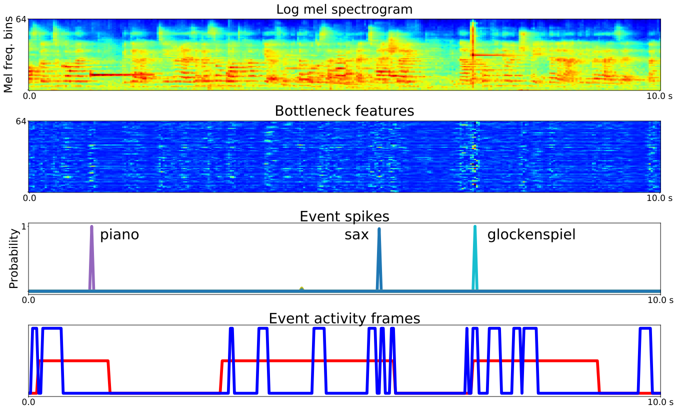

Fig. 5: From top to bottom: log mel spectrogram, bottleneck features, event spikes sequence and event activity frames. In the bottom subgraph, the red lines and blue lines denote the ground-truth event activity frames and the proposed event activity frames in Stage 2, respectively.

4. CONCLUSION

This paper proposed a CTC-based SED system that uses SLD instead of strongly labelled data, with a novel unsupervised clustering stage. Experimental results show the performance of the proposed method using SLD is comparable to the previous *A&V* system using strongly labelled data, and is far better than the *XKWP* system using weakly labelled data, indicating the effectiveness of the proposed method.

For the proposed method, the main error rate in *ER* is the *D rate* because the proposed event activity frames based on unsupervised clustering in Stage 2 are still inaccurate, and time location of sound events in audio clips is difficult in SED tasks. The future work will focus on improving the accuracy of proposed event onset/offset time in audio clips.

5. ACKNOWLEDGEMENTS

This research was partly supported by EPSRC grant EP/N014111/1 “Making Sense of Sounds” and a Research Scholarship from the China Scholarship Council (CSC) No. 201406150082.

6. REFERENCES

[1] G. Valenzise, L. Gerosa, M. Tagliasacchi, et al., “Scream and gunshot detection and localization for audio-surveillance systems,” in *Advanced Video and Signal Based Surveillance*, 2007, pp. 21–26.

[2] A. Mesaros, T. Heittola, and T. Virtanen, “TUT database for acoustic scene classification and sound event detection,” in *EUSIPCO*, 2016, pp. 1128–1132.

[3] G. Parascandolo, H. Huttunen, and T. Virtanen, “Recurrent neural networks for polyphonic sound event detection in real life recordings,” in *ICASSP*, 2016, pp. 6440–6444.

[4] A. Mesaros, T. Heittola, et al., “DCASE 2017 challenge setup: Tasks, datasets and baseline system,” in *Workshop on DCASE 2017*, Munich, Germany, 2017.

[5] S. Adavanne and T. Virtanen, “A report on sound event detection with different binaural features,” in *DCASE 2017 Challenge, Tech. Rep.*, September 2017.

[6] A. Kumar and B. Raj, “Audio event detection using weakly labeled data,” in *ACM on Multimedia Conference*, 2016, pp. 1038–1047.

[7] Y. Xu, Q. Kong, W. Wang, and M. D. Plumbley, “Large-scale weakly supervised audio classification using gated convolutional neural network,” in *ICASSP 2018, Calgary Canada*, 2018, pp. 121–125.

[8] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, “A joint detection-classification model for audio tagging of weakly labelled data,” in *ICASSP*, 2017, pp. 641–645.

[9] Y. Hou, Q. Kong, and S. Li, “Audio tagging with connectionist temporal classification model using sequentially labelled data,” in *2018 International Conference on Communications, Signal Processing, and Systems, Dalian, China*, 2018.

[10] A. Graves and N. Jaitly, “Towards end-to-end speech recognition with recurrent neural networks,” in *Proc. of ICML*, 2014.

[11] Y. Hou, Q. Kong, J. Wang, and S. Li, “Polyphonic audio tagging with sequentially labelled data using crnn with learnable gated linear units,” in *Workshop on DCASE 2018*, November 2018, pp. 78–82.

[12] Y. Wang and F. Metze, “A first attempt at polyphonic sound event detection using connectionist temporal classification,” in *ICASSP*, 2017, pp. 2986–2990.

[13] A. Graves and F. Gomez, “Connectionist temporal classification:labelling unsegmented sequence data with recurrent neural networks,” in *ICML*, 2006, pp. 369–376.

[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” in *Deep Learning and Representation Learning Workshop on NIPS 2014*.

[15] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, “Language modeling with gated convolutional networks,” in *Proceedings of ICML 2017*, 2017, pp. 933–941.

[16] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in *ICML*, 2010, pp. 807–814.

[17] J. A. Hartigan and M. A. Wong, “Algorithm as 136: A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C*, vol. 28, pp. 100–108, 1979.

[18] M. Asamoah-Boaheng, “Performance evaluation of some zero mean classification functions under unequal misclassification cost,” in *International Conference on Applied Science and Technology*, 2014.

[19] M. H. Fulekar, *Bioinformatics: Applications in Life and Environmental Sciences*, pp. 207–214, Springer, 2009.

[20] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Noise Reduction in Speech Processing*, pp. 4097–4098, Springer Berlin Heidelberg, 2009.

[21] F. Eduardo, P. Manoj, F. Frederic, et al., “General-purpose tagging of freesound audio with audioset labels: Task description, dataset, and baseline,” in *Workshop on DCASE 2018*, Surrey, UK, 2018.

[22] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, “Detection and classification of acoustic scenes and events,” *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1733–1746, 2015.

[23] M. Valenti, A. Diment, G. Parascandolo, et al., “DCASE 2016 acoustic scene classification using convolutional neural networks,” in *Workshop on DCASE 2016*, Budapest, Hungary, 2016.

[24] A. Mesaros, T. Heittola, and T. Virtanen, “A multi-device dataset for urban acoustic scene classification,” in *Workshop on DCASE 2018*, Surrey, UK, 2018.

[25] Q. Kong, X. Yong, S. Iwona, W. Wang, and M. D. Plumbley, “Sound event detection and time-frequency segmentation from weakly labelled data,” *IEEE/ACM Transection on Audio, Speech and Language Processing (Early Access)*, 2018.

[26] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proceedings of ICLR 2015*.

[27] A. Mesaros, T. Heittola, and T. Virtanen, “Metrics for polyphonic sound event detection,” *Applied Sciences*, vol. 6, no. 6, pp. 162, 2016.