SOUND EVENT DETECTION USING POINT-LABELED DATA

Bongjun Kim[1] *and Bryan Pardo*[2]

Northwestern University
Department of Computer Science, Evanston, IL, USA
[1]bongjun@u.northwestern.edu, [2]pardo@northwestern.edu

ABSTRACT

Sound Event Detection (SED) in audio scenes is the task that has been studied by an increasing number of researchers. Recent SED systems often use deep learning models. Building these systems typically require a large amount of carefully annotated, strongly labeled data, where the exact time-span of a sound event (e.g. the 'dog bark' starts at 1.2 seconds and ends at 2.0 seconds) in an audio scene (a recording of a city park) is indicated. However, manual labeling of sound events with their time boundaries within a recording is very time-consuming. One way to solve the issue is to collect data with weak labels that only contain the names of sound classes present in the audio file, without time boundary information for events in the file. Therefore, weakly-labeled sound event detection has become popular recently. However, there is still a large performance gap between models built on weakly labeled data and ones built on strongly labeled data, especially for predicting time boundaries of sound events. In this work, we introduce a new type of sound event label, which is easier for people to provide than strong labels. We call them 'point labels'. To create a point label, a user simply listens to the recording and hits the space bar if they hear a sound event ('dog bark'). This is much easier to do than specifying exact time boundaries. In this work, we illustrate methods to train a SED model on point-labeled data. Our results show that a model trained on point labeled audio data significantly outperforms weak models and is comparable to a model trained on strongly labeled data.

Index Terms— Sound event detection, Point labels, Weak labels, Deep learning

1. INTRODUCTION

Sound Event Detection (SED) is a task of identifying a class of sound events and estimating the time position (i.e. start and end) of each occurrence of that class in an audio recording. Automatic SED is an essential task in many areas that require audio-based understanding of the environment. Applications include detecting source of noise in cities [1], identifying bird species singing in nature recordings [2], and gunshot detection in city recordings [3]. Deep neural networks have been successfully applied to recent SED systems [4, 5, 6] and are the current state-of-the-art.

The typical approach to training automatic SED systems is supervised machine learning. For SED systems to be maximally effective at labeling sound events and their onset/offset times within a recording, it needs to be trained on audio data with time-coded labels that indicate start and stop times of sound events (strongly labeled data). However, manually annotating each sound's onset and offset within a recording is a very time-consuming task. It often requires repeated listening and adjusting of label time boundaries on a visual interface [7]. Moreover, building deep learning models usually requires lots of training data (e.g. tens of thousands of labeled audio files). This imposes a large burden on human annotators.

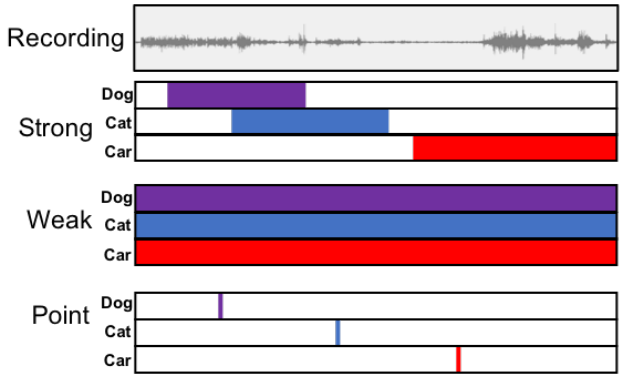

Figure 1: Examples of different types of audio annotation. Strong labels contain names of sound events and their time information. Weak labels only have clip-level presence or absence of events without their timing information. Point labels contain names of sound events at a single time point per sound event instance within a recording. The position of each point can vary within each instance.

As a result, models that can be trained on *weakly* labeled data have obtained much attention for sound event classification and detection [8, 4, 9, 10, 11]. Weakly labeled data names the sounds within an audio recording without specifying anything about onset or offset times (e.g. "there is a dog bark somewhere within this 30-second recording of a park scene"). Collecting weak labels is easier, since the human annotator does not need to indicate the exact time boundaries of events, which takes a lot of time. To collect weak labels, one might just need to listen to a sound clip once and record what events are in the clip [12]. Models trained on weak labels, however, typically do not achieve the performance of models trained on strongly labeled data.

In this paper, we introduce a new type of audio labeling, called *point labeling* which contains more information than weak labels, but still takes less human time to produce than strong labels. We also present a SED model that can be trained on *point-labeled* training data and show that its performance is similar to the performance of the strong model.

Our contributions are the following. First, we introduce a new type of sound event labeling, point labeling, which (to the best of

This work was funded, in part, by NSF Award Number:1617497

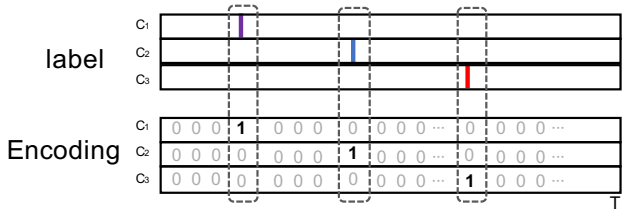

Figure 2: Examples of how to encode point labels to compute point-label loss. Given our model architecture, each time segment of encoded point labels covers $1/3$ seconds of an input recording.

our knowledge) has not been addressed in prior works on audio labeling. Second, we present a new method to train a machine learning model (in this case, a Fully Convolutional Network) on the point labeled audio data. Third, we present a strategy to automatically expand point labels so they can cover a greater portion of a sound event, which should lead to performance improvement on SED tasks. Finally, experimental results show that a model trained on point-labeled audio data significantly outperforms one trained on weak labels and achieves comparable results to a model trained on strongly labeled data.

2. POINT LABELS

Figure 1 shows an example of strong labels, weak labels, and point labels. Strong labels contain names of sound events and their temporal boundaries. Collecting strong labels is very time-consuming. Specifically, finding the correct onset and offset of each event in a polyphonic environment requires a lot of human effort [13]. Weak labels only indicate the presence or absence of sound events within an audio clip, without their temporal location within a recording. Weak labels are relatively easy to collect. A human annotator just needs to judge whether or not a target sound event is at all present in the recording. When modern SED systems are built on a weakly labeled dataset, a multiple-instance learning (MIL) formulation is typically applied, assuming each class of sound events is present during the entire clip if the event is present anywhere within the clip (see *Weak* in figure 1).

Point labels contain names of sound events at a single time point per sound event instance. A human annotator can indicate a sound occurred (e.g., by clicking a mouse button or hitting a key) anywhere within the area of the sound event. We believe that this user interaction takes much less time and effort than that required to find and mark the exact start and stop times for each labeled sound in a recording. In fact, in the right scenario, point labeling could take as little effort as weak labeling, since the time/effort difference between clicking at the time one hears a sound (providing a point label) and waiting until the end of a recording before indicating presence of a sound (weak labeling) may be very small, if the labeling task is structured appropriately.

2.1. Point-label loss

In this section, we present how to compute losses for a model when point labels are available. To compute the point-label loss, L_{point}, point labels of each recording needs to be encoded in the form of an output matrix of a SED model, $\hat{Y} \in \mathbb{R}^{C \times T}$ where C is the number of classes and T is the number of time segments. $\hat{Y}$ contains class probabilities for each segments. Let $Y^p \in \{0,1\}^{C \times T}$ be the encoded point labels of a recording. If a point label of the event c is located at the t-th time-segment, then 1 is assigned to $Y^p_{c,t}$, otherwise 0 is assigned to it. Figure 2 shows an example of the point label encoding when the number of classes C is 3.

The encoded point labels Y^p only contain information of the presence of an event, not absence of an event, which means that some of 0s in Y^p might be false-negative labels. Therefore, we should not compare Y^p directly to an estimate of the label matrix, $\hat{Y}$, to compute loss. Instead, we first multiply $\hat{Y}$ by Y^p to filter out any prediction probabilities that are not related to point labels (i.e., presence of events). As a result, we define point-label loss L_{point} as binary-cross entropy between $(\hat{Y} \odot Y^p)$ and Y^p, where $\odot$ indicates element-wise multiplication.

This has the effect of only computing loss on the time-segments where there is a point label. This means that most of the audio in a training example is not trained on. Rather than simply ignore that audio, we combine a weak loss L_{weak} function and point-label loss L_{point} in training. The weak loss L_{weak} is binary cross-entropy loss between weak labels, $y \in \{0,1\}^C$ and clip-level predictions, $\hat{y} \in \mathbb{R}^C$, where C is the number of classes. Weak labels are obtained by treating point labels as clip-level ground truth labels. The clip-level predictions from the model are obtained by applying time frame-wise max pooling operation on the model's output $\hat{Y}$ (i.e., segment-level predictions).

Finally, our model is trained by minimizing the following loss:

$$Loss = (1-\alpha)L_{weak}(y, \hat{y}) + \alpha L_{point}(Y^p, \hat{Y} \odot Y^p), \quad (1)$$

where α is a hyper-parameter to determine the contribution of each loss to the final loss.

2.2. Expanding point labels

While point labels contain time information of sound events which weak labels do not have, there is still a gap between strong labels and point labels. A single point label encoded for a SED model only covers a single time-step. In our experiment, given our model architecture, one time-step lasts $1/3$ of a second. However, many real-world sound events are longer than $1/3$ of a second. If we can expand point labels to label more adjacent time frames, we can improve the advantage point labels have over weak labels. However, if we expand too far, we may label adjacent segments where the labeled sound does not occur, creating false-positive "ground truth" labels that would harm learning. We now present a systemic way of expanding point labels. The idea is to measure similarities between a point labeled segment and its neighbor segments, and copy the point label only to similar neighbor segments.

To measure similarities between segments, we first build a SED model on a training set with only weak labels (i.e., weak model). Note that this training set is the one that the point model is trained on later. We then apply this *weak* model to label each audio clip in the *training set* at the segment level (1/3 of a second) to obtain segment-level class probabilities for each training example (i.e., $\hat{Y}$). The class probabilities for each segment can be thought of as feature embedding where the similarities between segments can be measured. Figure 3 shows an example of the proposed point label expanding method. For each point-labeled segment, we measure cosine similarity between that segment's class probabilities and the class probabilities of the segments immediately before and after it. If a neighbor's similarity is above a user-adjustable threshold (0.5

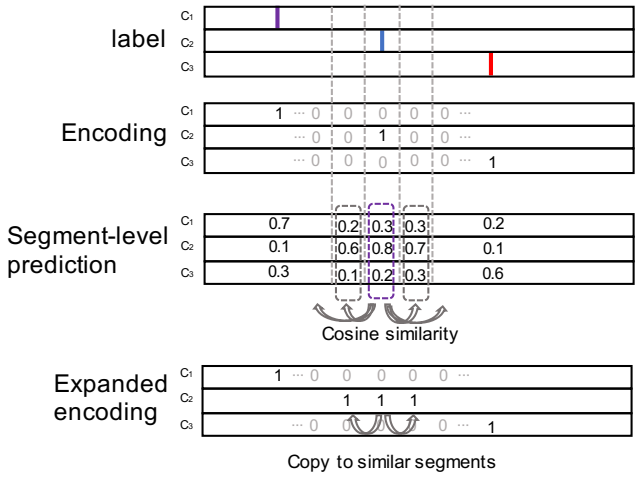

Figure 3: An example of expanding point labels to similar neighbor segments.

in our experiments), the point label is copied to that neighbor. This process is recursively applied to each neighbor segment that falls above the similarity threshold, until a segment with similarity less than a certain threshold is found.

This similar segment picking strategy is very useful, since the positions of point labels can vary with different human annotators. Regardless of the positions of point labels, this method successfully expands labels in a way that limits false-positive labels. In our experiments, we set the point labels at random locations within an event to reflect a real-world scenario.

3. EXPERIMENTS

We now present an experiment designed to shed light on the efficacy of point labels for providing a training signal to a deep model for sound event detection. If point labels prove more effective than weak labels, this indicates such labels have the potential to be an effective alternative labeling method that combines the advantage of strong labels (accurate labeling at fine time granularity) with weak labels (easy to create).

In the experiments, we train models on point-labeled data, weakly-labeled data or strongly-labeled data. We use weak loss L_{weak} for the weakly labeled SED model. For strong models, we encode the strong label matrix Y^s in the same way as point label encoding, but with the exact time boundary information of events and compute loss between $\hat{Y}$ and Y^s. We use binary cross-entropy to compute the loss for weak, strong, and point labels.

3.1. Model architecture

We use the same model architecture to learn from weak, strong, and point labels. Table 1 shows the architecture of our model. It is a fully convolutional network consisting of 8 convolutional layers and takes as input a log Mel-spectrogram of a variable length audio clip. Convolution operations for each layer are denoted as *Conv (the size of filters, the number of filters)* in the table. The number of filters on the last layer depends on the number of classes in the training data. Strides of all the convolutional layers are set to 1. Zero-padding with a size of 1 is applied to layer-1 to 6.

Table 1: Model architecture. *MP: 2D-Max Pooling (kernal size: 2×2, stride: 2), *N: the number of classes in the training dataset (N=10 in our experiment). The output shape column shows the size of tensor from each layer, given a 10-second recording as input.

Layers	Components	Output shape
Input	Mel-spectrogram	998×64
Layer-1	Conv (3×3, 64) → Relu → MP	499×32, 64
Layer-2	Conv (3×3, 128) → Relu → MP	249×16, 128
Layer-3	Conv (3×3, 256) → Relu	249×16, 256
Layer-4	Conv (3×3, 256) → Relu → MP	124×8, 256
Layer-5	Conv (3×3, 512) → Relu	124×8, 512
Layer-6	Conv (3×3, 512) → Relu → MP	62×4, 512
Layer-7	Conv (2×2, 1024) → Relu → MP	30×1, 1024
Layer-8	Conv (1×1, C) → Sigmoid	30×1, C

Given a recording, the network outputs a matrix $\hat{Y} \in \mathbb{R}^{C \times T}$ where C is the number of classes and T is the number of time segments, which represents class probabilities for each time segment. T depends on the input audio length. Table 1 also shows an example of output shapes from each layer. Given a 10-second recording, the network outputs $\hat{Y}$ ($C \times 30$ matrix) where each segment represents class probabilities for $1/3$ seconds of audio.

3.2. Dataset and performance metric

We evaluate models on URBAN-SED dataset which contains 10,000 soundscapes generated using the Scaper soundscape synthesis library [14]. We chose the dataset because it contains strong labels of all the soundscapes, which enables us to generate point labels as well as weak labels. Each file in the dataset is 10 second long (about 28 hours in total) and contains between 1 to 9 sound events from 10 classes in the UrbanSound8K dataset [15]. The 10 sound classes are the following: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gunshot, jackhammer, siren, and street music. The dataset is pre-divided into train, validation, and test sets containing 6000, 2000, and 2000 files respectively.

We generated point labels for training data. In this work, each time segment of the ground truth label matrix Y covers $1/3$ seconds of an input recording. We split the input audio into $1/3$ second segments and encoded the point labels for each segment. The position of point labels within a sound event was set randomly (random selection from a set of points on the $1/3$-second grid for an event) to reflect the real annotation scenario where different human annotators might choose different locations for a point label.

In this work, we assume that only a single point label per sound event is collected, although we believe that multiple point labels per event should not hurt the performance and could lead to further performance improvement.

To measure the performance of the sound event detection (SED) models, we compute segment-based F_1 score and Error Rate (ER) with the segment granularity of 1 second, which are official evaluation methods in the DCASE challenge [16], an annual evaluation of SED models. F_1 score is computed based on true-positive, false-positive, and false-negative values of every class at every second over the testing set. ER measures the amount of errors in terms

Table 2: Segment-based F_1 score, Precision, Recall (higher is better for all three), and Error Rate (lower is better) for each model. Segment size is 1 second.

Model	F1	Precision	Recall	ER
Weak	0.582	0.795	0.459	0.568
Strong	**0.639**	0.675	0.607	**0.519**
Point single	0.612	0.763	0.511	0.533
Point expanded	**0.638**	0.684	0.597	**0.523**
Strong (McFee et al.[4])	0.551	0.693	0.458	0.642

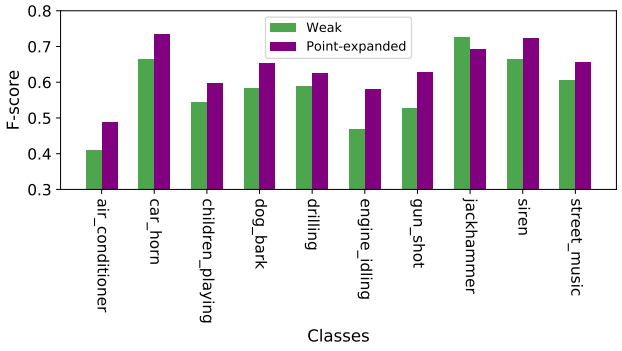

Figure 4: Class-wise F_1 scores.

of insertions , deletions and substitutions. More details about the metrics for SED can be found in [16].

3.3. Training and prediction

Each audio file was resampled to 16kHz mono and represented by a log-scale Mel-spectrogram with 64 Mel bins, a window size of 25 ms and hop size of 10 ms. All models were trained on mini-batches of 32 examples using Adam optimizer with a learning rate of 0.0001. The training stopped if the model performance on the validation set did not improve for 20 epochs. For the point label model, we also tested different α values in the loss function (see equation 1) and models with $\alpha = 0.8$ showed the best performance.

We also applied transfer learning because 6,000 training examples are relatively small dataset given our model architecture. When training our models, we initialized the network with the set of weights from a VGGish pre-trained model [17] that has been trained on 3,000 sound classes of 8 million YouTube videos. Layers 1 to 6 were initialized by with the weights from the VGGish model. The rest of the layers were randomly initialized. In training, the first three layers were fixed and the rest of the layers were fine-tuned on the training set. To obtain labels from the network output, we applied the likelihood threshold 0.5 to class probabilities to determine presences or absences of an event.

3.4. Results

We compared 2 variants of point models. *Point-single* model uses only a single point label at a random position of an event. *Point-expanded* uses the updated point labels expanded by our proposed methods in Section 2.2. Both models were trained by minimizing

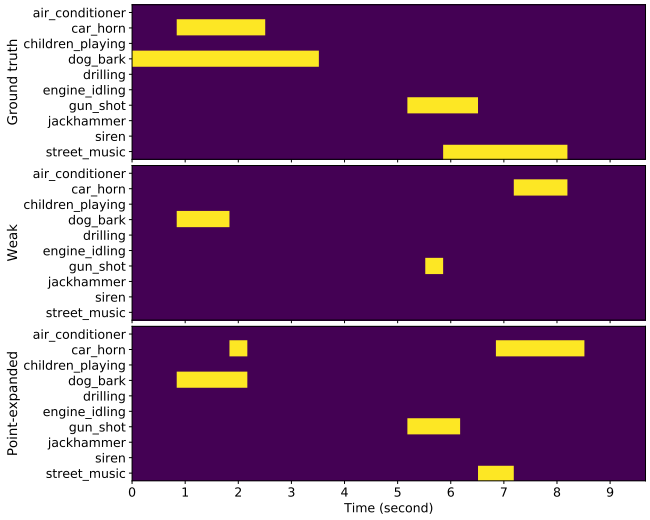

Figure 5: An example of SED performed by *weak* model and *Point-expanded* model

the loss function with α of 0.8 (see equation 1), which showed the best performance among point models. We also built a *Weak* model as our baseline and *Strong* model as the best possible model.

Table 2 shows segment-based F_1 score with precision and recall and ER for each model. We can see that our two point models outperform the weak model, which shows the point labels help models localize sound events more accurately. Finally, the result shows that the proposed point label expansion improve the performance even further. *Point expanded* model achieved $F_1 = 0.638$ and $ER = 0.523$ which is nearly identical to our strong model's score ($F_1 = 0.639$, $ER = 0.519$). We achieved this performance gain even though we randomly set the positions of point labels, which proves that the point models are robust to the position of point labeling which might vary in real annotation scenario. To provide the current state of the art as context, we also show results on the same data from a strong recent model by McFee et al. [4]. All models showed a better F1 score than McFee et al. We guess this is due to a more capable network architecture and transfer learning from a model pre-trained on a much larger dataset.

Figure 4 shows class-wise F_1 scores of weak and point-expanded models. We can see that the point-expanded model outperforms the weak model for most classes of the sound events. Figure 5 visualizes an example of the predictions performed by the weak and point-expanded model given a 10 second of recording from our testing set. The recording contains 4 different sound events. As shown in the figure, the point model made more accurate predictions of temporal boundaries of events.

4. CONCLUSION

We introduced a new way of labeling sound events, point labeling which is relatively easier to perform than collecting strong labels, but can provide more information about temporal locations of events than weak labels. We presented a training method of SED models on point-labeled audio data and showed that the model significantly outperforms weak models and is comparable to a model trained on strongly labeled data.

5. REFERENCES

[1] J. P. Bello, C. Silva, O. Nov, R. L. Dubois, A. Arora, J. Salamon, C. Mydlarz, and H. Doraiswamy, "Sonyc: A system for monitoring, analyzing, and mitigating urban noise pollution," *Commun. ACM*, vol. 62, no. 2, pp. 68–77, Jan. 2019.

[2] D. Stowell, M. Wood, Y. Stylianou, H. Glotin, *et al.*, "Bird detection in audio: a survey and a challenge," in *Proceedings of the 26th IEEE International Workshop on Machine Learning for Signal Processing*. IEEE Computer Society, 2016, pp. 1–6.

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

[4] B. McFee, J. Salamon, and J. P. Bello, "Adaptive pooling operators for weakly labeled sound event detection," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 26, no. 11, pp. 2180–2193, 2018.

[5] E. Cakır, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[6] Q. Kong, Y. Xu, I. Sobieraj, W. Wang, and M. D. Plumbley, "Sound event detection and time–frequency segmentation from weakly labelled data," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 27, no. 4, pp. 777–787, 2019.

[7] B. Kim and B. Pardo, "A human-in-the-loop system for sound event detection and annotation," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 8, no. 2, p. 13, 2018.

[8] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, "Audio set classification with attention model: A probabilistic perspective," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 316–320.

[9] A. Kumar and B. Raj, "Audio event and scene recognition: A unified approach using strongly and weakly labeled data," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 3475–3482.

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

[11] B. Kim and S. Ghaffarzadegan, "Self-supervised attention model for weakly labeled audio event classification," in *European Signal Processing Conference (EUSIPCO)*, 2019.

[12] M. Cartwright, G. Dove, A. E. M. Méndez, and J. P. Bello, "Crowdsourcing multi-label audio annotation tasks with citizen scientists," in *Proceedings of the ACM on Human-Computer Interaction*, 2019.

[13] M. Cartwright, A. Seals, J. Salamon, A. Williams, S. Mikloska, D. MacConnell, E. Law, J. P. Bello, and O. Nov, "Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations," *Proceedings of the ACM on Human-Computer Interaction*, vol. 1, no. CSCW, p. 29, 2017.

[14] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 344–348.

[15] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *22nd ACM International Conference on Multimedia (ACM-MM'14)*, Orlando, FL, USA, Nov. 2014, pp. 1041–1044.

[16] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for polyphonic sound event detection," *Applied Sciences*, vol. 6, no. 6, p. 162, 2016.

[17] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, "CNN architectures for large-scale audio classification," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 131–135.