WEAKLY SUPERVISED SOUND ACTIVITY DETECTION AND EVENT CLASSIFICATION IN ACOUSTIC SENSOR NETWORKS

Janek Ebbers, Lukas Drude, Reinhold Haeb-Umbach

Paderborn University
Department of Communications Engineering
33098 Paderborn, Germany
{ebbers,drude,haeb}@nt.upb.de

Andreas Brendel, Walter Kellermann

Friedrich Alexander University Erlangen-Nuernberg
Multimedia Communications and Signal Processing
91058 Erlangen, Germany
{andreas.brendel,walter.kellermann}@fau.de

ABSTRACT

In this paper we consider human daily activity recognition using an acoustic sensor network (ASN) which consists of nodes distributed in a home environment. Assuming that the ASN is permanently recording, the vast majority of recordings is silence. Therefore, we propose to employ a computationally efficient two-stage sound recognition system, consisting of an initial sound activity detection (SAD) and a subsequent sound event classification (SEC), which is only activated once sound activity has been detected. We show how a low-latency activity detector with high temporal resolution can be trained from weak labels with low temporal resolution. We further demonstrate the advantage of using spatial features for the subsequent event classification task.

Index Terms— acoustic sensor network, sound recognition

1. INTRODUCTION

Driven by the annual editions of the Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, sound recognition recently attracted increased interest not only in the research community. Typical data sets for sound recognition contain a collection of audio segments, which contain one or more events to be recognized. For example, the Google AudioSet [1] is a collection of $2\,\mathrm{M}$ video clips of $10\,\mathrm{s}$ each. The goal is to classify which of the 527 event classes are present in a clip.

While the AudioSet and similar datasets are certainly important to advance the state of the art, for many applications, the task of annotating segments of audio is only part of the challenge. In this contribution we are concerned with the use of an ASN to monitor human daily activities in order to support smart-home or ambient assisted living (AAL) applications. Since the monitoring system should be always running, for most of the time there will be no sound event of interest, e.g., because of absence of the person from home.

For such a scenario it is reasonable to devise a two-stage classification approach, where, in a first stage, a lightweight sound activity detector detects presence or absence of relevant sound events, and only if relevant sound activity is detected, the second stage is activated, which carries out the sound event classification. Such an approach is also followed in today's speech-controlled digital home assistants, where a comparatively small on-device voice activity detector listens for the wake word (e.g., "Alexa"), and the cloud-based recognizer is only employed if the wake word has been detected [2]. However, the concept of a wake word does not translate to general audio. Furthermore, there are no large-scale data sets available indicating on- and offsets of sounds at the desired temporal resolution that can be used to train a sound activity detector. However, there are data sets indicating whether there is activity or no activity in larger time segments, e.g., one minute, such as the SINS database [3] used in this study.

In this paper we show how a convolutional neural network (CNN) [4] can be trained to make accurate SAD predictions every $200\,\mathrm{ms}$, given only these low-resolution labels. As soon as activity is detected the detected sound is classified using another (larger) CNN. It was shown in [5], that the employed CNN architectures can be run on a Raspberry Pi 3 much faster than real-time, which makes the proposed system feasible for application in an ASN without the necessity of transmitting signals to a central server. This and the rather small receptive fields of the CNNs allow low-latency ($<1\,\mathrm{s}$) detection and classification compared to classifcation of longer time segments of, e.g., $10\,\mathrm{s}$ [6].

Further, when using an ASN for daily human activity monitoring, not only the spectral signature of a sound is indicative of an event, but also its source location. This is because certain sounds, e.g., cooking, occur only in certain places. Another objective of this paper is thus to evaluate the usefulness of spatial in addition to spectral features. Please note that our source code is publicly available on github.[1]

The remainder of the paper is organized as follows. First, the smart-home scenario is decribed in Sec. 2. After explaining the feature extraction in Sec. 3, our proposed two-stage recognition system is explained in Sec. 4 and Sec. 5. Experiments are described in Sec. 6, and conclusions are drawn in Sec. 7.

This work has been supported by Deutsche Forschungsgemeinschaft under contract no. HA 3455/15-1 and KE 890/10-1 within the Research Unit FOR 2457 (acoustic sensor networks).

[1] https://github.com/fgnt/sins

2. SCENARIO

We consider a smart-home scenario, where an ASN is distributed in a single-person apartment. Such a scenario is captured in the SINS database [3]. It contains real-life recordings from 12 sensor nodes distributed over several rooms taken over a period of one week. Each sensor node is equipped with a linear array of four microphones. One person lived in the apartment for a continuous duration of one week and annotated his daily activities such as working, cooking, eating. In this work, however, we only use the recordings of the seven sensor nodes in the combined living room and kitchen area.

3. FEATURE EXTRACTION

First, we perform a short-time Fourier transform (STFT) with $60\,\mathrm{ms}$ frames and $20\,\mathrm{ms}$ hops on the provided $16\,\mathrm{kHz}$ audio signals. For each frame we then extract 64 log-mel-band energy features [7] in the range of f_{min}=200 Hz to f_{max}=8 kHz yielding a feature map of shape $64\times T_n$, where T_n denotes the number of frames in the nth signal.

For the SEC task we further experiment with different spatial feature sets. The first set consists of inter-channel phase differences (IPDs) [8] between microphones d and d', which are calculated from the STFT observations $x_{d,t,f}$, where t and f denote the time frame and frequency bin index, respectively, as follows: $\varphi_{d,d',t,f} = \arg\{x_{d,t,f}\, x^*_{d',t,f}\}$. Inspired by their efficiency in multi-channel deep clustering [9] and due to their well-defined range of $[-1, 1]$, we here decided to select sine and cosine IPD features.

The second set consists of coherence-based features. The coherence between two channels is defined as the normalized cross power spectral density (CPSD) $\Phi_{d,d',t,f}$ between two channels d and d' [10, Eq. 2.17]. Here we use the magnitude and the sine and cosine of its phase, where CPSDs are estimated based on a few time frames. This features indicate, to some degree, how close a particular sound event is [11].

The third spatial feature set employs the complex Watson kernel-based Direction-of-Arrival (DoA) estimator [12]. Here, we calculate steering vectors $\mathbf{w}_{k,f}$ corresponding to K=17 candidate directions ranging from 10° to 170° (endfire positions are biased [13]). The likelihood that an observation $\mathbf{x}_{t,f}$ originates from a certain direction k is obtained as:

$$p(\mathbf{x}_{t,f}; \kappa, \mathbf{w}_{k,f}) = \frac{1}{c(\kappa)} \mathrm{e}^{\kappa |\mathbf{w}_{k,f}^{\mathsf{H}} \mathbf{x}_{t,f}|^2}, \qquad (1)$$

where $\mathbf{x}_{t,f} = (x_{d,t,f}; d = 1, \ldots, D)^{\mathsf{T}}$ is the vector of microphone signals, κ is a concentration parameter and $c(\kappa)$ is a normalization constant [14]. We obtain a feature map for each of the K=17 candidate directions.

The spatial features are extracted for a pair of adjacent microphones. However, classification scores of all pairs (with four microphones there are three adjacent pairs) from all sensor nodes may be fused (more details in Sec. 6). The extracted spatial feature maps, which have the same dimensionality as the STFT, are subsampled (at the maxima of the mel-filters) to fit to the dimensionality of the log-mel-band energies. We then stack the two log-mel spectrograms from the considered microphone pair with the subsampled versions of the spatial feature maps. Finally, we subtract the global mean of each feature over time and then divide each feature map by its global standard deviation. As the spatial signature depends on the position of the sensor, we further stack a one-hot representation of the node index when using spatial features. Hence, when, e.g., using IPDs as spatial features, we obtain a total of $C = 2$ log-mel + 2 IPD + #nodes one-hot feature maps.

Table 1. CNN Architecture with output shapes of each block. B, C and T denote the mini-batch size, the number of input feature maps and the number of input frames, respectively. Each ConvXd uses a kernel size of 3, a stride of 1 and includes BatchNorm [15], and ReLU. The parameter l controls the number of kernels in a layer.

Block	Output shape
Feature Extraction	$B\times C\times 64\times T$
2×Conv2d(16·l)	$B\times 16{\cdot}l\times 64\times T$
Pool2d(2×2)	$B\times 16{\cdot}l\times 32\times \lceil T/2\rceil$
2×Conv2d(32·l)	$B\times 32{\cdot}l\times 32\times \lceil T/2\rceil$
Pool2d(2×1)	$B\times 32{\cdot}l\times 16\times \lceil T/2\rceil$
2×Conv2d(64·l)	$B\times 64{\cdot}l\times 16\times \lceil T/2\rceil$
Pool2d(2×5)	$B\times 64{\cdot}l\times 8\times \lceil T/10\rceil$
Reshape	$B\times 512{\cdot}l\times \lceil T/10\rceil$
Conv1d(128·l)	$B\times 128{\cdot}l\times \lceil T/10\rceil$
Linear(V)	$B\times V\times \lceil T/10\rceil$

4. SOUND ACTIVITY DETECTION

In the given scenario, when an ASN is used in a home environment to monitor daily activities, most of the time there is actually no sound activity, e.g., because no one is at home. In such a scenario it is beneficial to run a lightweight SAD and only run the computationally more expensive SEC if sound activity was detected.

Compared to common voice activity detection systems there are two major challenges here. First, the relevant sounds usually have a very low Signal-to-Noise Ratio (SNR). This is especially true when the microphones used are of low quality, which is frequently the case for such devices to limit costs. Hence, it is difficult to use energy-based approaches [16] for activity detection. Second, we do not have any training data with (*strong*) labels, which indicate on- and offsets of sound activity at frame-level resolution, which makes training of an activity classification [17, 18] challenging. The given *weak* labels only indicate activity within a certain time period without providing the exact on- and offsets of the sound events.

Inspired by weakly labeled sound event detection [19], we propose an SAD system that can be trained by only using the information about presence or absence within longer time periods, which we refer to as sequences here. If a person is present it is very likely that there are at least some sounds during that sequence. If, on the other hand, the person is absent, it is certain that there are no relevant sounds in that sequence.

For SAD we propose a rather small CNN consisting of

eight layers as outlined in Table 1 with l=1 and a single output value (V=1). For this task, the input is the log-mel-band energy feature map from a single microphone (C=1) without additional spatial feature maps. However, decisions of multiple microphones and sensor nodes may be fused (more details in Sec. 6). Due to pooling, the network makes predictions $\tilde{z}_{n,m} = \sigma\big(\text{CNN}(\mathbf{X}_{n,m})\big)$ every tenth frame (200 ms) with $\mathbf{X}_{n,m}$ denoting the input features in the receptive field of the CNN at prediction step m of the nth sequence. Due to the sigmoid function $\sigma(\cdot)$, values between 0 and 1 are obtained with a high value indicating sound activity. At test-time we obtain binary decisions $\hat{z}_{n,m}$ by employing a decision boundary at 0.5. As we only have sequence-level targets, we adopt the training objective from [19]. A sequence-level score is computed as a weighted average as follows:

$$\tilde{z}_n = \sum_{m=1}^{M_n} w_{n,m}\tilde{z}_{n,m} \quad \text{with} \quad w_{n,m} = \frac{\exp(\alpha\tilde{z}_{n,m})}{\sum_{i=1}^{M_n}\exp(\alpha\tilde{z}_{n,i})} \qquad (2)$$

where M_n is the number of predictions in the nth sequence.

While in [19] α was learned automatically, it is considered as a hyperparameter here that can be used to control how aggressive the system is, i.e., to balance false positives and missed hits. If we choose α=0, Eq. (2) becomes the arithmetic mean of the individual scores and the network must produce a high scores on average in order to output a high sequence-level score. If $\alpha\to\infty$, Eq. (2) computes the maximum of the individual scores and the system needs only one high score at a single time step to reach a high sequence-level score. However, it also must not produce any single high score, if it is to predict no activity.

As we do have binary sequence-level targets z_n, with z_n=1 if the person is present and z_n=0 if the person is absent, the model can now be trained using binary cross entropy:

$$L(\tilde{z}_n, z_n) = -z_n\log(\tilde{z}_n) - (1-z_n)\log(1-\tilde{z}_n). \qquad (3)$$

5. SOUND EVENT CLASSIFICATION

After sound activity has been detected, the sound has to be classified. Here we classify a sound by the acoustic scene it belongs to such as working, cooking or eating, i.e., we consider a multi-label classification problem. We use the CNN outlined in Table 1 as classification network, this time with l=4, V=#classes and C=#feature maps:

$$\tilde{\mathbf{y}}'_{n,m} = \text{CNN}(\mathbf{X}_{n,m})\,, \quad \tilde{\mathbf{y}}_{n,m} = \text{softmax}\big(\tilde{\mathbf{y}}'_{n,m}\big)$$

with $\tilde{\mathbf{y}}_{n,m}$ denoting the V-dimensional vector of classification scores. The classification network has a total of 4.3 M parameters and is 16 times larger than the SAD network.

To train the event classifier we only use those segments of the training data where sound activity is detected. Assuming the detected sounds originate from the person's activity that is to be classified, we adopt the weak labels $\mathbf{y}_n$ as strong labels $\mathbf{y}_{n,m}$ at each prediction step m of the nth detected segment, where $\mathbf{y}_n$ is a V-dimensional one-hot encoding of the ground truth class label. Our training objective is then given by the cross-entropy between the predicted labels and target labels:

$$L(\tilde{\mathbf{y}}_{n,m}, \mathbf{y}_{n,m}) = -\sum_{v=1}^{V} y_{n,m,v}\log(\tilde{y}_{n,m,v})\,. \qquad (4)$$

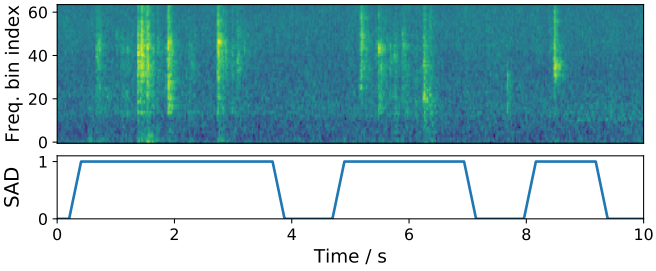

Fig. 1. Example sound activity detection.

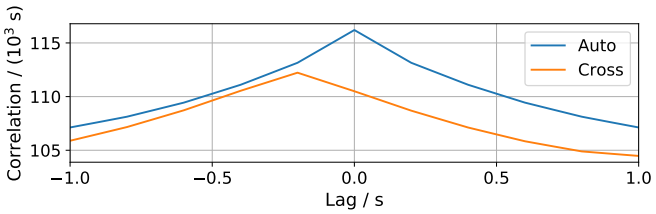

Fig. 2. Correlations of SAD.

6. EXPERIMENTS

For our experiments we use the recordings of the seven sensor nodes in the combined living room and kitchen area of the SINS Database [3] as explained in Sec. 2. We further divide the seven sensor nodes into the two groups $(1, 3, 7)$ and $(2, 4, 6, 8)$, which will be used to investigate matched and mismatched training and testing setups.

We randomly split the different sessions of the database into $\frac{1}{2}$, $\frac{1}{6}$ and $\frac{1}{3}$ for training, validation and evaluation, respectively. We discarded a small fraction of the absence sessions where the person is present in another room and is not sleeping to not be affected by sounds originating from the other rooms although the session is labeled as absence. We merged the classes "phone calling" and "visit" into a single class "social activity" [6], as each has only very few occurrences. This results in a total of eight classes (without absence).

The presented SAD system is trained using audio clips from the sensors $(1, 3, 7)$ with a maximum duration of 1 min, i.e., if a session is longer than 1 min, which is usually the case, it is split into smaller chunks. We then train the system as explained in Sec. 4 to classify if the person is present or absent during an audio clip. We train the system for 20 k iterations using Adam [20] with a mini-batch size of 16 and a learning rate of $3\cdot10^{-4}$. Training is started with α=1 which is linearly annealed to α=2 during the first 10 k iterations. The best checkpoint is determined based on the F_1-score on the validation set. An example of an SAD of the trained system is shown in Fig. 1.

A welcome byproduct of the channel-wise segmentation is that it can be used for rough synchronization of the signals from different nodes by correlating their SADs. Fig. 2 shows the autocorrelation of a single SAD as well as the cross corre-

Table 2. SAD performance in terms of F_1-Score [%] (high is good [22]) for different SNRs and setups where m indicates matched evaluation and f indicates decision fusion.

	Setup		SNR/dB			
System	m	f	-3	0	3	10
Oracle Thr.	✓		57.5	70.8	79.3	91.6
Proposed	✓		77.3	83.9	87.8	91.7
Oracle Thr.	✓	✓	66.1	76.6	83.7	93.1
Proposed	✓	✓	80.2	87.9	91.1	95.3
Oracle Thr.		✓	69.7	77.8	84.6	93.6
Proposed		✓	83.0	88.1	90.7	93.3

lation between the SADs from two unsyncronized nodes. We used this approach to align the signals from multiple nodes enabling decision fusion at frame level both for SAD as well as SEC evaluated later.

To quantitatively evaluate the system's activity detection performance, we artificially generate test samples by randomly mixing isolated sound events from the DCASE 2016 Challenge Task 2 [21] corpus into $20\,\mathrm{s}$ clips of absence recordings from the SINS Database serving as background noise. That way we know about the sounds' ground truth location in time which is required for evaluation purposes. We evaluate for different SNRs by first normalizing the variance of the sound to the variance of the background noise and then scaling the sound by $\sqrt{\mathrm{SNR}}$. We evaluate whether the system detects activity within the known active range of the sound, which would be counted as a true positive. Every other detected segment in the clip is counted as a false positive.

As a simple baseline we consider a thresholding-based approach where we sum the log-mel-band energy features in windows of $200\,\mathrm{ms}$ to obtain scores with the same resolution as the NN output. We then determine the optimal threshold on the evaluation set, i.e., the same set that we report our scores on, using a parameter sweep. Therefore, we refer to this baseline as Oracle Thresholding.

We consider four setups here: matched evaluation on nodes $(1, 3, 7)$ with and without decision fusion and mismatched evaluation on nodes $(2, 4, 6, 8)$ with and without decision fusion. For decision fusion, majority voting is applied at each time step using decisions from all microphones from all nodes. In the "without fusion" setting we evaluate the single-microphone decisions.

From the results shown in Table 2 it can be seen that our proposed system outperforms the Oracle Thresholding in almost all setups. It can further be observed that SAD greatly benefits from fusing decisions from multiple sensor nodes allowing to also achieve high F_1-scores for low SNRs. It is especially worth noting that the SAD performance for low SNRs does not decrease in the mismatched setting but actually increases a bit. This can be explained by the decision fusion of the channels from four rather than three sensor nodes.

Next, we evaluate the subsequent SEC. For this purpose we first perform SAD on all datasets using all seven sensor nodes. The resulting segments with sound activity are then used to train and evaluate the classification system in the following. If a detected sound segment exceeds the maximum duration of $4\,\mathrm{s}$, it is split into smaller chunks. We investigate the following three questions: 1) Considering that our classifier has a rather short receptive field of $<1\,\mathrm{s}$, how well can we classify the daily activity based on the few sounds being active therein? Note that performing classification with such a small receptive field allows low-latency processing such that a smart system can react much faster. 2) How much does the classification benefit from fusing decisions from multiple nodes? 3) How much does the classification benefit from using the spatial features proposed in Sec. 3?

Table 3. SEC performance in terms of F_1-Score [%] (high is good [22]) for different features and setups where m indicates matched training and f indicates decision fusion.

	Setup				
Features	m	f	$F_{1,\mathrm{min}}$	$F_{1,\mathrm{max}}$	$F_{1,\mathrm{mean}}$
Log-Mel			36.2	98.7	75.5
Log-Mel		✓	43.4	99.6	80.3
Log-Mel	✓	✓	45.2	99.8	84.0
Log-Mel+IPD	✓	✓	49.8	99.9	85.1
Log-Mel+Coh.	✓	✓	49.2	99.9	85.9
Log-Mel+Watson	✓	✓	48.5	99.9	85.5

The systems are trained for $100\,\mathrm{k}$ iterations using Adam with a mini-batch size of 48 segments and a learning rate of $3 \cdot 10^{-4}$. The system performance is measured in terms of $F_{1,\mathrm{mean}}$ which is the mean per-class F_1-score, i.e., we individually compute an F_1-score for each of the eight events under test and average these. The best checkpoint is determined based on the performance on the validation set. Evaluation is performed on the evaluation set using signals from the sensor nodes $(2, 4, 6, 8)$. Matched training is performed on the same sensor nodes and mismatched training on the nodes $(1, 3, 7)$.

Results are shown in Table 3. It can be seen that performance can be greatly improved by using decision fusion over multiple sensor nodes as well as spatial features. Note that spatial features depend on the sensor position and hence can only be trained in a matched setup. While matched labeled training is usually not available for a specific home environment we hypothesize that such a system may be trained using position-independent systems serving as teachers, which, however, is beyond the scope of this work.

7. CONCLUSIONS

In this paper we proposed a two-stage sound recognition system consisting of a sound activity detection and a sound event classification system. For both systems we performed experiments on realistic recordings and demonstrated their suitability for sound recognition. We further showed that both systems benefit from an ASN by fusing decisions over multiple nodes and using spatial features for classification.

8. REFERENCES

[1] J. F. Gemmeke, D. P. W. 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 *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.

[2] R. Haeb-Umbach, S. Watanabe, T. Nakatani, M. Bacchiani, B. Hoffmeister, M. Seltzer, H. Zen, and M. Souden, "Speech Processing for Digital Home Assistants," *accepted for IEEE Signal Processing Magazine*, 2019.

[3] G. Dekkers, S. Lauwereins, B. Thoen, M. W. Adhana, H. Brouckxon, T. van Waterschoot, B. Vanrumste, M. Verhelst, and P. Karsmakers, "The SINS database for detection of daily activities in a home environment using an acoustic sensor network," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017)*, 2017, pp. 32–36.

[4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[5] J. Ebbers, J. Heitkaemper, J. Schmalenstroeer, and R. Haeb-Umbach, "Benchmarking neural network architectures for acoustic sensor networks," in *Speech Communication; 13th ITG-Symposium*, 2018, pp. 1–5.

[6] G. Dekkers, L. Vuegen, T. van Waterschoot, B. Vanrumste, and P. Karsmakers, "DCASE 2018 Challenge-Task 5: Monitoring of domestic activities based on multi-channel acoustics," *arXiv preprint arXiv:1807.11246*, 2018.

[7] X. Huang, A. Acero, and H.-W. Hon, "Spoken Language Processing: A Guide to Theory, Algorithm, and System Development," *Prentice Hall PTR*, 2001.

[8] Y. Xu, Q. Kong, Q. Huang, W. Wang, and M. D. Plumbley, "Convolutional gated recurrent neural network incorporating spatial features for audio tagging," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 3461–3466.

[9] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[10] J. Bitzer and K. U. Simmer, "Superdirective microphone arrays," in *Microphone arrays*, pp. 19–38. Springer, 2001.

[11] A. Brendel and W. Kellermann, "Learning-based acoustic source-microphone distance estimation using the coherent-to-diffuse power ratio," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 61–65.

[12] L. Drude, F. Jacob, and R. Haeb-Umbach, "DOA-estimation based on a complex Watson kernel method," in *European Signal Processing Conference (EUSIPCO)*, 2015, pp. 255–259.

[13] F. Jacob and R. Haeb-Umbach, "On the bias of direction of arrival estimation using linear microphone arrays," in *Speech Communication; 12. ITG Symposium*. VDE, 2016, pp. 1–5.

[14] K. V. Mardia and I. L. Dryden, "The complex Watson distribution and shape analysis," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 4, pp. 913–926, 1999.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[16] A. Benyassine, E. Shlomot, H.-Y. Su, D. Massaloux, C. Lamblin, and J.-P. Petit, "ITU-T recommendation G.729 Annex B: a silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 64–73, 1997.

[17] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, 1999.

[18] R. Zazo Candil, T. N. Sainath, G. Simko, and C. Parada, "Feature learning with raw-waveform CLDNNs for voice activity detection," *Interspeech 2016*, 2016.

[19] 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.

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

[21] G. Lafay, E. Benetos, and M. Lagrange, "Sound event detection in synthetic audio: analysis of the DCASE 2016 task results," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017, pp. 11–15.

[22] A. Mesaros, T. Heittola, and T. Virtanen, "Metrics for Polyphonic Sound Event Detection," *Applied Sciences*, vol. 6, no. 6, pp. 162, 2016.