

ENSEMBLE OF ACCDOA- AND EINV2-BASED SYSTEMS WITH D3NETS AND IMPULSE RESPONSE SIMULATION FOR SOUND EVENT LOCALIZATION AND DETECTION

Technical Report

Kazuki Shimada, Naoya Takahashi, Yuichiro Koyama, Shusuke Takahashi, Emiru Tsunoo, Masafumi Takahashi, Yuki Mitsufuji

Sony Group Corporation, Japan

ABSTRACT

This report describes our systems submitted to the DCASE2021 challenge task 3: sound event localization and detection (SELD) with directional interference. Our previous system based on activity-coupled Cartesian direction of arrival (ACCDOA) representation enables us to solve a SELD task with a single target. This ACCDOA-based system with efficient network architecture called RD3Net and data augmentation techniques outperformed state-of-the-art SELD systems in terms of localization and location-dependent detection. Using the ACCDOA-based system as a base, we perform model ensembles by averaging outputs of several systems trained with different conditions such as input features, training folds, and model architectures. We also use the event independent network v2 (EINV2)-based system to increase the diversity of the model ensembles. To generalize the models, we further propose impulse response simulation (IRS), which generates simulated multi-channel signals by convolving simulated room impulse responses (RIRs) with source signals extracted from the original dataset. Our systems significantly improved over the baseline system on the development dataset.

Index Terms— DCASE2021, Sound event localization and detection, Model ensemble, Impulse response simulation

1. INTRODUCTION

Sound event localization and detection (SELD) involves identifying both the direction of arrival (DOA) and the type of sound. Numerous methods have been tackling SELD through the DCASE challenge [1, 2, 3, 4, 5, 6]. The SELD system based on activity-coupled Cartesian direction of arrival (ACCDOA) representation enables us to solve a SELD task with a single target [6]. This ACCDOA-based system with RD3Net [7, 6] and data augmentation techniques outperformed state-of-the-art SELD systems in terms of localization and location-dependent detection [6].

In this study, we perform model ensembles of several systems trained with different conditions and model architectures. We use ACCDOA-based systems as a base and perform model ensembles by averaging outputs of the systems trained with various input features, training folds, and model architectures. The event independent network v2 (EINV2)-based system [4] is also used to increase the diversity of the model architectures. To increase the training data, we carry out four kinds of data augmentation: 1) equalized mixture data augmentation [8, 9], 2) rotation of first-order Ambisonic (FOA) signals [10], 3) the multichannel version of SpecAugment [11, 6], and 4) impulse response simulation (IRS), which is a novel approach for SELD to generate simu-

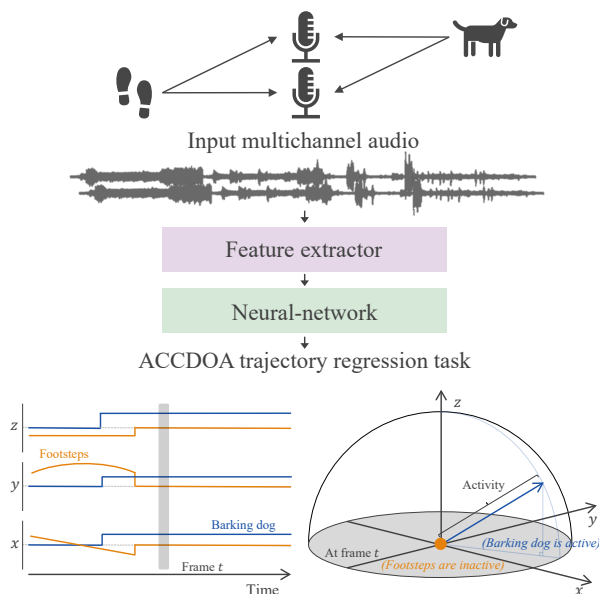


Figure 1: Illustration of an ACCDOA-based SELD system.

lated multi-channel signals by convolving simulated room impulse responses (RIRs) with source signals extracted from the original dataset. Experiments on the development dataset showed our system significantly improved over the baseline system.

2. ACCDOA-BASED SYSTEM

In this section, we first give the framework of ACCDOA-based systems. Then we explain the parts of our pipeline: the features, data augmentation, and network architecture.

2.1. ACCDOA framework

The ACCDOA representation assigns a sound event activity to the length of the corresponding Cartesian DOA vector, which enables us to handle SELD as a single task with a single network [6]. A schematic flow of the ACCDOA-based SELD system is shown in Fig. 1. After extracting the features, the network outputs frame-wise ACCDOA vectors for target sound events. The model is trained to minimize the Euclidean distance between the estimated and target coordinates in the ACCDOA representation. We solve the multi-output regression with a mean square error (MSE) loss [6].

2.2. Features

Multichannel amplitude spectrograms and inter-channel phase differences (IPDs) are used as frame-wise features. Here, $IPD_{t,f,p,q} = \angle x_{t,f,p} - \angle x_{t,f,q}$ is computed from the short-time Fourier transform (STFT) coefficients $x_{t,f,p}$ and $x_{t,f,q}$, where t, f, p , and q denote the time frame, the frequency bin, the Ambisonic channel p , and the channel q , respectively. We fix $p = 0$ to compute relative IPDs between all the other channels, $q \neq 0$. Since the input consists of four Ambisonic channel signals, we can extract four amplitude spectrograms and three IPDs. Some of the models use PCEN [12], cosIPDs, and sinIPDs [13] as input features, instead of the amplitude spectrograms and IPDs.

2.3. Data augmentation

To increase the generalizability of the model, we use three data augmentation techniques as described in our previous paper [6]: EMDA [8, 9], rotation of FOA signals [10], and the multichannel version of SpecAugment [11, 6]. We use the EMDA method, where up to two audio events are mixed with random amplitudes, delays, and the modulation of frequency characteristics, i.e., equalization [8, 9]. We also use the spatial augmentation method [10]. It rotates the training data represented in the FOA format and enables us to increase the numbers of DOA labels without losing the physical relationships between steering vectors and observations. Lastly, we use the multichannel version of SpecAugment [11, 6]. SpecAugment was extended to the channel dimension in addition to the time-frequency hard masking schemes applied on amplitude spectrograms [6].

In addition to these augmentation techniques, we further propose IRS. Fig. 2 describes the workflow of IRS augmentation. While source signals are extracted from the original reverberant dataset, RIRs for the FOA format are simulated. Then the simulated RIRs are convolved with the extracted source signals to obtain augmented multi-channel signals for the training dataset.

To extract source signals from the original dataset, we first extract non-overlapping and non-moving sound event segments from the original dataset on the basis of the provided sound event detection (SED) label. Since the dataset of DCASE2021 task3 contains unlabeled directional interference events, we heuristically eliminate the interference events on the basis of a two-stage approach. In the first stage, the sound event segments are eliminated when the segments are not recognized as any sound event classes by a preliminarily trained model. These segments can be regarded as the ones that dominantly include interference events. In the second stage, using the covariance matrix of the segment in the frequency bins under 2 kHz and their eigenvalues, the segments are removed when the second eigenvalue is larger than 30% of the first eigenvalue in most frequency bins. Finally, complex Gaussian mixture model (CGMM)-based minimum variance distortionless response (MVDR) beamforming is applied to the extracted segments to obtain the source signals of the target sound events [14, 3].

In order to obtain RIRs for the FOA format, RIRs for the Eigenmike are first simulated assuming that the Eigenmike can be regarded as a rigid baffle array [15, 16]. An image source method [17] is utilized for simulating reverberant conditions. The reverberation time (RT60) is randomly defined to be within 100 to 500 ms. Then the simulated RIRs are converted to FOA format on the basis of a higher-order Ambisonics (HOA) encoding process [18, 19].

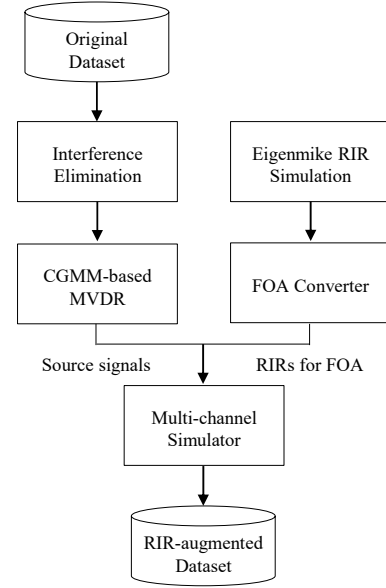


Figure 2: The workflow of IRS.

2.4. Network architecture

In this study, to increase the diversity of the model ensemble, we consider three variants of D3Net architecture [7], which has achieved the state-of-the-art performance in music source separation. Those architectures are illustrated in Fig. 3 (a)–(c).

The first variant is RD3Net [6]. To adapt D3Net architecture to the SELD problem, three modifications are made. First, dense blocks in the up-sampling path is omitted because high frame-rate prediction is not necessary for the SELD problem. Second, the bottleneck part is replaced with gated recurrent unit (GRU) cells. Third, the batch normalization is replaced with the network deconvolution [20]. The architecture is illustrated in Fig. 3 (a).

As the second variant, we propose to replace the GRU block with a time-frequency RNN (TFRNN) block, which is inspired by a dual-path recurrent neural network (DPRNN) block [21]. DPRNN was originally proposed to model long sequence. The long sequence is modeled by the intra- and inter-chunk RNN blocks (i.e., dual path). The dual path is considered to be able to efficiently integrate both time-domain context and frequency-domain context. Each RNN block is composed of bidirectional long short-term memory (LSTM), fully connected layers, and layer normalization. The intra- and inter-chunk RNN blocks are alternately stacked. As shown in Fig. 3 (b), TFRNN with four RNN blocks is applied to the output of fourth dense block, and followed by a frequency-domain pooling operation.

The last variant is inspired by the EINV2 architecture [4]. Soft parameter-sharing using cross-stitch and a transformer encoder are incorporated into D3Net. As shown in Fig. 3 (c), EINV2-D3Net consists of two parts, to which amplitude spectrograms features with and without IPDs are fed respectively. Each part has four D3 blocks and one track-wise transformer encoder block. An integration layer combines the outputs of the two parts, and outputs ACC-DOA vectors. Note that the differences from the original EINV2 [4] are twofold: 1) replacement of conv-block with D3 block and 2) the integration layer after the transformer encoder blocks.

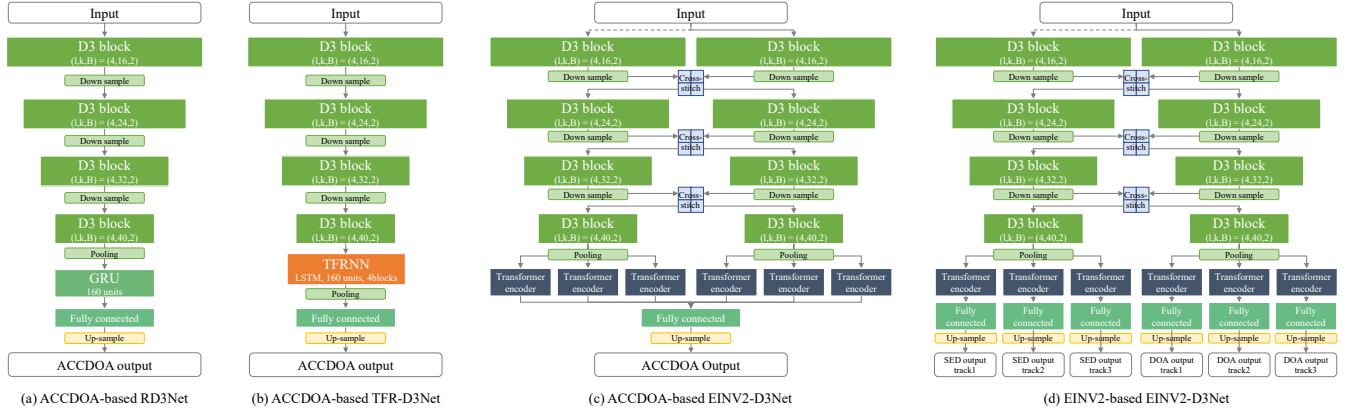


Figure 3: Illustration of D3Net architectures. (a) ACCDOA-based RD3Net, (b) ACCDOA-based TFR-D3Net (c) ACCDOA-based EINV2-D3Net (d) EINV2-based EINV2-D3Net. l , k , and B in D3 blocks denote the number of layers, growth rate, and number of blocks, respectively.

2.5. Post-processing

During the inference, we split the 60-second inputs into shorter segments that overlap. Subsequently, each segment is processed, and the results of overlapped frames are averaged. To further improve the performance, we conduct a post-processing by rotating the FOA data, estimating the ACCDOA vectors, rotating the vectors back, and averaging the vectors of different rotation patterns [6].

2.6. Hyper-parameters and training procedure

The sampling frequency is set to 24 kHz. The STFT is used with a 20 ms frame length and 10 ms frame hop. The frame length of input to the networks is 512 frames. The frame shift length is set to 20 frames during the inference. We use a batch size of 32. Each training sample is generated on-the-fly [22]. We gradually increase the learning rate to 0.001 with 50,000 iterations [23]. After the warm-up, the learning rate is decreased by 10% if the SELD score of the validation do not improve in 40,000 consecutive iterations. We use the Adam optimizer with a weight decay of 10^{-6} . We validate and save model weights every 10,000 iterations up to 400,000 iterations. Finally we average the model weights from the last 10 models as in [24].

3. EINV2-BASED SYSTEM

The EINV2 framework uses the track-wise output format and permutation-invariant training (PIT) [4]. The track-wise format assumes that the output of the model has several tracks, each with at most one predicted event with a corresponding DOA. Different tracks can detect events of the same class with different DOA, which enables us to detect overlaps of the same class. EINV2 is optimized with multiple targets using PIT by solving the track permutation problem. We use a binary cross entropy (BCE) for the SED classification task and a MSE for the DOA regression task [4].

The other components such as features, data augmentation, post-processing, hyper-parameters, and training procedure in the EINV2-based system are the same as in the ACCDOA-based systems. As shown in Fig. 3 (d), the network architecture use soft parameter-sharing using cross-stitch and a transformer encoder, the same as the original EINV2 architecture [4] except for replacement of conv-block with D3 block.

Table 1: Ensemble configuration. Th. means threshold.

| Ens. | Avg. | Th. | Base system |
|------|---------------|-----|---|
| #1 | Simple avg. | 0.3 | ACCDOA-based RD3Net \times 11 ACCDOA-based TFR-D3Net ACCDOA-based EINV2-D3Net EINV2-based EINV2-D3Net \times 2 |
| #2 | Simple avg. | 0.4 | ACCDOA-based RD3Net \times 11 ACCDOA-based TFR-D3Net ACCDOA-based EINV2-D3Net EINV2-based EINV2-D3Net \times 2 |
| #3 | Weighted avg. | 0.3 | ACCDOA-based RD3Net \times 11 ACCDOA-based TFR-D3Net ACCDOA-based EINV2-D3Net EINV2-based EINV2-D3Net \times 2 |
| #4 | Weighted avg. | 0.4 | ACCDOA-based RD3Net \times 15 ACCDOA-based TFR-D3Net \times 2 ACCDOA-based EINV2-D3Net \times 2 EINV2-based EINV2-D3Net \times 4 |

4. MODEL ENSEMBLE

A model ensemble is performed by averaging outputs of several models trained with different conditions such as input features, training folds, and model architectures. We convert the EINV2 format into the ACCDOA format for averaging by the following two steps: making pseudo track-wise ACCDOA outputs by multiplying SED outputs and DOA output for each track, and taking non track-wise ACCDOA outputs with the maximum activity among the tracks.

Here we use the simple average and the weighted average of the outputs predicted by different model. The weights are assigned to each model, thus the dimension of weights is M , where M is the number of models. We use 0.3 or 0.4 as a threshold. During the inference, if the norm of the ACCDOA vector of a sound event class exceeds the threshold, we consider the class to be active. The models used for the ensemble are listed in Table 1.

Table 2: SELD performance of our systems evaluated by using joint localization/detection metrics for the development set.

| System | Number of parameters | Validation fold | | | | Testing fold | | | |
|---------------------------------|----------------------|-----------------|----------------|-----------|-----------|-----------------|----------------|-----------|-----------|
| | | ER_{20° | F_{20° | LE_{CD} | LR_{CD} | ER_{20° | F_{20° | LE_{CD} | LR_{CD} |
| Baseline FOA [1] | 0.5 M | - | - | - | - | 0.73 | 30.7 | 24.5° | 40.5 |
| ACCDOA-based RD3Net | 1.7 M | 0.44 | 68.1 | 14.3° | 70.0 | 0.48 | 64.1 | 13.2° | 63.2 |
| ACCDOA-based TFR-D3Net w/ IRS | 3.0 M | 0.42 | 70.2 | 12.0° | 70.5 | 0.45 | 67.5 | 11.5° | 67.0 |
| ACCDOA-based EINV2-D3Net w/ IRS | 7.1 M | 0.43 | 69.2 | 12.0° | 71.7 | 0.44 | 67.9 | 11.5° | 69.5 |
| EINV2-based EINV2-D3Net w/ IRS | 7.1 M | 0.35 | 75.9 | 12.3° | 79.2 | 0.48 | 63.5 | 11.5° | 64.9 |
| Ensemble #1 | 42.6 M | 0.39 | 75.3 | 11.5° | 82.6 | 0.43 | 69.6 | 11.3° | 73.2 |
| Ensemble #2 | 42.6 M | 0.36 | 75.7 | 10.9° | 77.6 | 0.42 | 69.6 | 10.7° | 68.6 |
| Ensemble #3 | 42.6 M | 0.38 | 75.8 | 11.3° | 83.0 | 0.43 | 69.9 | 11.1° | 73.2 |
| Ensemble #4 | 73.6 M | 0.35 | 75.9 | 10.6° | 77.4 | 0.41 | 70.0 | 10.3° | 68.7 |

Table 3: Our preliminary experimental result without and with IRS for the development set.

| ACCDOA-based system | Testing fold | | | |
|---------------------|-----------------|----------------|-----------|-----------|
| | ER_{20° | F_{20° | LE_{CD} | LR_{CD} |
| Without IRS | 0.55 | 56.6 | 16.1° | 57.6 |
| With IRS | 0.54 | 57.6 | 15.2° | 58.5 |

5. EXPERIMENTAL EVALUATION

In this section, we show the experimental settings and our results on the development dataset.

5.1. Experimental settings

We evaluated our approach on the development set of TAU Spatial Sound Events 2021 - Ambisonic using the suggested setup [1]. The baseline was an ACCDOA-based system with a convolutional recurrent neural network (CRNN) [1]. In the setup, four metrics were used for the evaluation [25]. The first was the localization error LE_{CD} , which expresses the average angular distance between predictions and references of the same class. The second was a simple localization recall metric LR_{CD} , which expresses the true positive rate of how many of these localization predictions were correctly detected in a class out of the total number of class instances. The next two metrics were the location-dependent error rate (ER_{20°) and F-score (F_{20°), where predictions are considered as true positives only when the distance from the reference is less than 20° .

5.2. Experimental results

Table 2 shows the performance with the development set for our systems. As shown in the table, our ACCDOA-based RD3Net outperformed the baseline for each metric by a large margin. The result also suggested that D3Net variants and IRS data augmentation improved performance. Model ensemble improved F_{20° by 2.1 points from the single model in the testing fold. The result showed that model ensembles constantly performed better than single models. There were no significant difference between the simple average and the weighted average.

Table 3 shows our preliminary experiment result of an ACCDOA-based system without and with IRS. We observed that the performance with IRS was better than without IRS.

6. CONCLUSION

We presented our approach to DCASE2021 task 3: sound event localization and detection (SELD) with directional interference. Our systems use the activity-coupled Cartesian direction of arrival (ACCDOA) representation to solve both sound event detection (SED) and DOA estimation tasks in a unified manner. Using the ACCDOA-based system with RD3Net as a base, we perform model ensembles by averaging outputs of several systems trained with different conditions such as input features, training folds, and model architectures. We also use the event independent network v2 (EINV2)-based system to increase the diversity of the model ensembles. To generalize the models, we further propose impulse response simulation (IRS), which generates simulated multi-channel signals by convolving simulated room impulse responses (RIRs) with source signals extracted from the original dataset. Our systems performed superiorly to the baselines without model ensemble. Furthermore, we observed further improvement with ensembles of the ACCDOA- and EINV2-based systems.

7. REFERENCES

- [1] A. Politis, S. Adavanne, D. Krause, A. Deleforge, P. Srivastava, and T. Virtanen, "A dataset of dynamic reverberant sound scenes with directional interferers for sound event localization and detection." *arXiv preprint arXiv:2106.06999*, 2021.
- [2] A. Politis, S. Adavanne, and T. Virtanen, "A dataset of reverberant spatial sound scenes with moving sources for sound event localization and detection," *arXiv preprint arXiv:2006.01919*, 2020.
- [3] Q. Wang, J. Du, H.-X. Wu, J. Pan, F. Ma, and C.-H. Lee, "A four-stage data augmentation approach to ResNet-Conformer based acoustic modeling for sound event localization and detection," *arXiv preprint arXiv:2101.02919*, 2021.
- [4] Y. Cao, T. Iqbal, Q. Kong, F. An, W. Wang, and M. D. Plumbley, "An improved event-independent network for polyphonic sound event localization and detection," in *Proc. of IEEE ICASSP*, 2021, pp. 885–889.
- [5] T. N. T. Nguyen, N. K. Nguyen, H. Phan, L. Pham, K. Ooi, D. L. Jones, and W.-S. Gan, "A general network architecture for sound event localization and detection using transfer learning and recurrent neural network," in *Proc. of IEEE ICASSP*, 2021, pp. 935–939.

- [6] K. Shimada, Y. Koyama, N. Takahashi, S. Takahashi, and Y. Mitsufuji, “ACCDOA: Activity-coupled cartesian direction of arrival representation for sound event localization and detection,” in *Proc. of IEEE ICASSP*, 2021, pp. 915–919.
- [7] N. Takahashi and Y. Mitsufuji, “Densely connected multidilated convolutional networks for dense prediction tasks,” in *Proc. of IEEE CVPR*, 2021.
- [8] N. Takahashi, M. Gygli, B. Pfister, and L. V. Gool, “Deep convolutional neural networks and data augmentation for acoustic event detection,” in *Proc. of Interspeech*, 2016.
- [9] N. Takahashi, M. Gygli, and L. Van Gool, “AENet: Learning deep audio features for video analysis,” *IEEE Trans. on Multimedia*, vol. 20, pp. 513–524, 2017.
- [10] L. Mazzon, Y. Koizumi, M. Yasuda, and N. Harada, “First order ambisonics domain spatial augmentation for DNN-based direction of arrival estimation,” in *Proc. of DCASE Workshop*, 2019.
- [11] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” in *Proc. of Interspeech*, 2019, pp. 2613–2617.
- [12] V. Lostanlen, J. Salamon, M. Cartwright, B. McFee, A. Farnsworth, S. Kelling, and J. P. Bello, “Per-channel energy normalization: Why and how,” *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 39–43, 2018.
- [13] 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 *Proc. of IEEE ICASSP*, 2018, pp. 1–5.
- [14] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, “Robust MVDR beamforming using time-frequency masks for on-line/offline ASR in noise,” in *Proc. of IEEE ICASSP*, 2016, pp. 5210–5214.
- [15] A. Politis, “Microphone array processing for parametric spatial audio techniques,” PhD thesis, Aalto University, 2016.
- [16] Y. Mitsufuji, N. Takamune, S. Koyama, and H. Saruwatari, “Multichannel blind source separation based on evanescent-region-aware non-negative tensor factorization in spherical harmonic domain,” *IEEE ACM Trans. on Audio Speech Lang. Process.*, vol. 29, pp. 607–617, 2021.
- [17] R. Scheibler, E. Bezzam, and I. Dokmanić, “Pyroomacoustics: A python package for audio room simulation and array processing algorithms,” in *Proc. of IEEE ICASSP*, 2018, pp. 351–355.
- [18] S. Moreau, J. Daniel, and S. Bertet, “3D sound field recording with higher order Ambisonics—objective measurements and validation of a 4th order spherical microphone,” in *Proc. of Audio Engineering Society Convention*, 2006, pp. 20–23.
- [19] A. Politis and H. Gamper, “Comparing modeled and measurement-based spherical harmonic encoding filters for spherical microphone arrays,” in *Proc. of IEEE WASPAA*, 2017, pp. 224–228.
- [20] C. Ye, M. Evanusa, H. He, A. Mitrokhin, T. Goldstein, J. A. Yorke, C. Fermuller, and Y. Aloimonos, “Network deconvolution,” in *Proc. of ICLR*, 2020.
- [21] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation,” in *Proc. of IEEE ICASSP*, 2020, pp. 46–50.
- [22] H. Erdogan and T. Yoshioka, “Investigations on data augmentation and loss functions for deep learning based speech-background separation,” in *Proc. of Interspeech*, 2018, pp. 3499–3503.
- [23] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, “Accurate, large minibatch sgd: Training imagenet in 1 hour,” *arXiv preprint arXiv:1706.02677*, 2017.
- [24] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, *et al.*, “A comparative study on transformer vs RNN in speech applications,” in *Proc. of IEEE ASRU*, 2019, pp. 449–456.
- [25] A. Mesaros, S. Adavanne, A. Politis, T. Heittola, and T. Virtanen, “Joint measurement of localization and detection of sound events,” in *Proc. of IEEE WASPAA*, 2019.