TRELLISNET-BASED ARCHITECTURE FOR SOUND EVENT LOCALIZATION AND DETECTION WITH REASSEMBLY LEARNING

Sooyoung Park, Wootaek Lim, Sangwon Suh, Youngho Jeong,

Electronics and Telecommunications Research Institute
Media Coding Research Section
218 Gajeong-ro, Yuseong-gu, Daejeon, Korea
{sooyoung, wtlim, suhsw1210, yhcheong}@etri.re.kr

ABSTRACT

This paper proposes a deep learning technique and network model for DCASE 2019 task 3: Sound Event Localization and Detection. Currently, the convolutional recurrent neural network is known as the state-of-the-art technique for sound classification and detection. We focus on proposing TrellisNet-based architecture that can replace the convolutional recurrent neural network. Our TrellisNet-based architecture has better performance in the direction of arrival estimation compared to the convolutional recurrent neural network. We also propose reassembly learning to design a single network that handles dependent sub-tasks together. Reassembly learning is a method to divide multi-task into individual sub-tasks, to train each sub-task, then reassemble and fine-tune them into a single network. Experimental results show that the proposed method improves sound event localization and detection performance compared to the DCASE 2019 baseline system.

Index Terms— DCASE 2019, sound event localization and detection, TrellisNet, convolutional recurrent neural network, reassembly learning

1. INTRODUCTION

A sound event localization and detection (SELD) [1, 2] is a new estimation problem that combines sound event detection (SED) and direction of arrival estimation (DOAE) into a single task. Hirvonen [3] suggested a classification approach for DOAE using convolutional neural networks (CNN). The disadvantage of DOAE using classification is that only discrete direction of arrival (DOA) can be predicted. Besides, applying this method to polyphonic sound events increases the number of target classes and requires a large amount of dataset to train the network. Therefore, the DCASE 2019 baseline [1, 2] tried to overcome these disadvantages using multi-output regression.

Multi-output regression is a method to fit regressors for all target classes. So the DCASE 2019 baseline estimates DOA for both active and inactive sound events. As a result, multi-output regression interrupts training for DOAE because of unnecessary inactive events. Multi-output regression forces the DOA label to have azimuth and elevation values for all classes. Therefore, DOA labels have true DOA values for active events and default DOA values for inactive events. Multi-output regression loss includes both DOA loss for active events and DOA loss for inactive events. As a result, the network output is trained to be closer to the true DOA for active events and to the default DOA for inactive events. In short, SELD with multi-output regression operates to estimate the array containing the default DOA values, rather than estimating DOA only for active events. To overcome this problem, Cao et al. [4] proposed a two-stage learning method to avoid loss from inactive events. The two-stage learning excludes DOA prediction for inactive events by masking using ground-truth event labels. As a result, the two-stage learning makes significant performance improvement.

The two-stage learning solves the problem of multi-output regression by excluding the inactive events from the DOA loss. However, the two-stage learning still has a problem with inactive events at the inference stage. The two-stage learning derives the final SELD output prediction by concatenating SED network prediction and DOA network prediction. Here, the DOA network of two-stage learning excludes the inactive events in training. Therefore, the DOA prediction values of the inactive events are random. As a result, there is a problem in the Hungarian algorithm used to calculate the DOA error for polyphonic sound events. This problem occurs when the SED network incorrectly predicts the event. Since the Hungarian algorithm is a method to find a combination that reduces the pair-wise cost, the DOA error is derived from the random DOA output of the inactive sound event in the above case. In short, the two-stage learning causes the irony that DOAE for mispredicted events is made with random DOA predictions excluded from training.

In this paper, we propose a reassembly learning method to partially alleviate the problem from inactive sound events. This method is based on the two-stage learning [4]. Reassembly learning is a method of reassembling pre-trained SED network and DOA network into a single SELD network, and training the reassembled SELD network. The key idea of reassembly learning is to reduce the influence of inactive sound events through fine-tuning.

A recurrent neural network (RNN) is widely used for sequence modeling. Theoretically, RNN can train an infinite length of the sequence. However, in actual RNN, the vanishing gradient occurs while repeating the sequential operation. It means that the stability of the RNN structure is not guaranteed. Therefore gating mechanism has been proposed such as LSTM [5] and GRU [6]. But the instability problem was not completely solved. There have been attempts to process sequence data using TCN, which is based on 1D convolution and showed good performance [7, 8]. Bai et al. [9] proposed TrellisNet, a new architecture that takes advantage of two advantages of CNN and RNN. TrellisNet is a special form of TCN structure that stacks multiple layers of TCNs and acts like a truncated RNN at a specific weight. TrellisNet outperforms several benchmarks, such as language modeling and long-term memory retention [9].

CRNN is used in DCASE 2019 baseline. Furthermore, many SED and DOAE studies [1, 2, 4, 10] choose CRNN as basic network

https://doi.org/10.33682/nzsm-jr50

architecture. CRNN is currently state-of-the-art in sound classifications and detection. CRNN architecture uses CNN for local feature extraction and RNN for a temporal summary of the output of the CNN. In this paper, we propose a new network architecture for sound classification and detection using a TrellisNet [9] based on the temporal convolutional network (TCN). The key idea of the proposed architecture is to take advantage of both CNN and RNN by replacing RNN with TrellisNet.

2. DATASET

DCASE 2019 challenge task 3 provides audio dataset for 11 classes of sound events. The sound event of DCASE 2019 dataset is synthesized using spatial room impulse response recorded in five indoor locations. The development dataset consists of 400 files. Each audio file is a one-minute duration with a sampling rate of 48000 Hz. The development dataset is provided as two different types: four channel tetrahedral microphone arrays and a first-order ambisonic (FOA) format. Besides, DCASE challenge task 3 targets polyphonic sound events with a maximum of two sound events overlap.

3. FEATURE

Our models use log mel-band energy (4 channels), mel-band acoustic active intensity (3 channels) and mel-band acoustic reactive intensity (3 channels). Log mel-band energy is extracted from the tetrahedral microphone dataset. On the other hand, mel-band acoustic active and reactive intensity are extracted from FOA dataset. The input feature configuration used in this paper is shown in Table 1.

3.1. Log mel-band energy

In the DCASE 2018 challenge task 4, many participants used the log mel-band energy as an input feature for SED [11–16]. Mel-band energy is a feature that applies a mel filter to an energy spectrogram. The mel filter mimics the non-linear human auditory perceptions. The results of DCASE 2018 challenge proved that this non-linear feature has strength for SED. Also, we expect to obtain information of time difference, loud difference for sound localization from a multi-channel log mel-band energy feature.

3.2. Mel-band acoustic intensity

Ambisonic is a coefficient of the spatial basis of the audio signal. Each spatial basis is expressed as spherical harmonics. Zero-order ambisonic signal (W) represents the component that is omnidirectional. First-order ambisonic signals (X, Y, Z) represent three polarized bidirectional components. In the presence of multiple sources or reverberant environments, it is impossible to express complex sound fields using FOAs (W, X, Y, Z). Therefore, we need additional methods to extract spatial information from FOAs for the reverberant environment. Acoustic intensity is one of these methods that extract spatial information from FOAs [17].

Acoustic intensity is one of the physical quantities representing the sound field. The acoustic intensity vector $\mathbf{I}(t, f)$ can be expressed by using FOA as equation (1). Active acoustic intensity vector I_a is a real part of acoustic intensity that represents the flow of sound energy. It is a physical quantity directly related to DOA. The active acoustic intensity is expressed as a real part of the product of the pressure $p(t, f)$ and the particle velocity $\mathbf{v}(t, f)$. Reactive intensity I_r is an imaginary part of acoustic intensity that represents a dissipative local energy transfer. It is a physical quantity dominated by direct sound from a single source. We expect to obtain spatial decomposed information and phase information from acoustic intensities of 6 channels of mel-band acoustic intensity.

$$\mathbf{I}(t, f) = p(t, f)\mathbf{v}^*(t, f) = -W(t, f) \begin{bmatrix} X^*(t, f) \\ Y^*(t, f) \\ Z^*(t, f) \end{bmatrix} \quad (1)$$

Finally, it is important that the size of the acoustic intensity feature and the size of mel-band energy feature are equal to deal with those features in the single network. Therefore, mel filter is applied to resize acoustic intensity features.

Table 1: Input features for single networks

Name	Feature configuration
MIC	8 channels, magnitude and phase spectrogram (Mic)
FOA	8 channels, magnitude and phase spectrogram (Foa)
Log-Mel	4 channels, log mel-band energy (Mic)
Log-Mel + I_a	7 channels, log mel-band energy (Mic) + mel-filtered active intensity
Log-Mel + I_a + I_r	10 channels, log mel-band energy (Mic) + mel-filtered active intensity + mel-filtered reactive intensity

4. NETWORK ARCHITECTURE

4.1. CNN Layers

In Figure 1(a), the CNN layers consist of two gated linear unit (GLU) blocks and global average pooling that compresses the frequency (mel-bin) axis. Both the SED network and the DOA network use the same CNN layers. The details of the GLU block are shown in Figure 1(b).

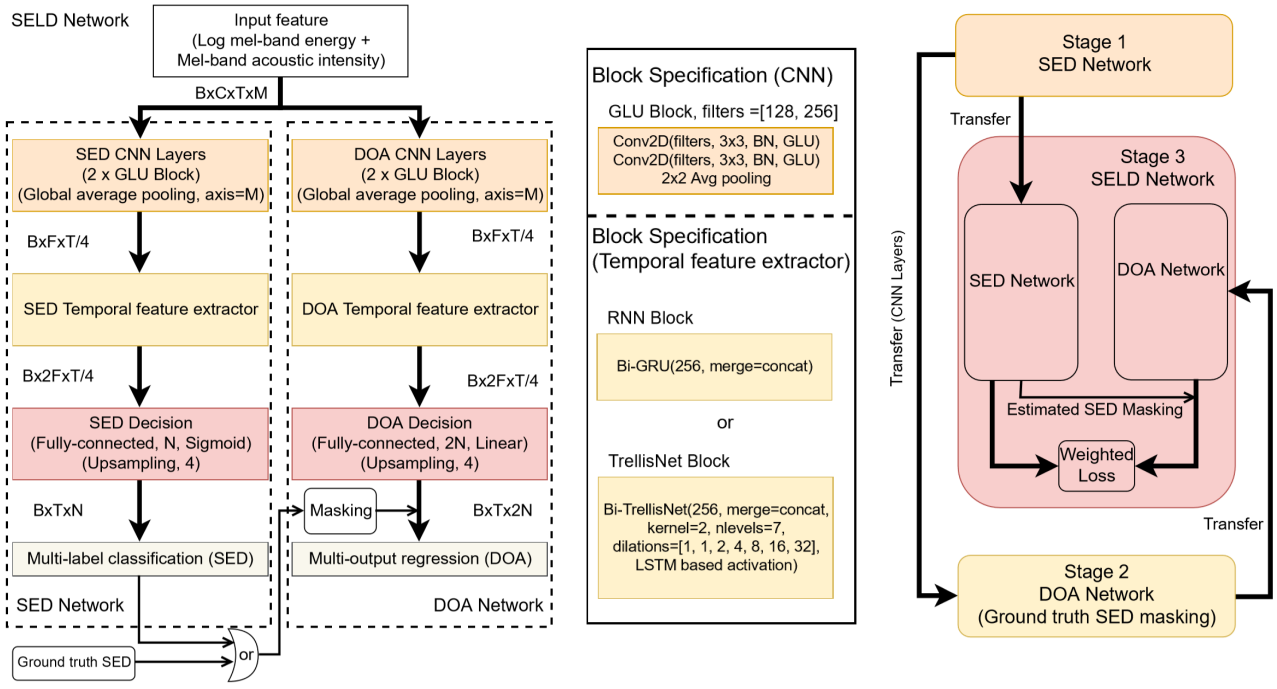
4.2. Temporal feature extractor

There are two different types of temporal feature extractors for the proposed model. One is Bidirectional-GRU, one of RNNs. The other is Bidirectional-TrellisNet which is a special form of TCNs. The details of the RNN block and the TrellisNet block are shown in Figure 1(b).

TrellisNet [9] tried to combine CNNs and RNNs through direct input injection and weight sharing among TCN layers. The key idea of TrellisNet is to implement a CNN that behaves like an RNN under certain conditions. When RNN is unfolded, a new input comes in every step and the same weight is applied. In TrellisNet, the temporal convolution layer with kernel size 2 corresponds to one RNN step. TrellisNet can replace the recurrent structure of the RNNs by stacking of multiple temporal convolution layers and adding the input injection and weight sharing techniques. Therefore TrellisNet can take advantage of both structural and algorithmic elements of CNN and RNN. In TrellisNet, LSTM is applied between each temporal convolution layer. We set the receptive field of TrellisNet to cover the input frame length for performance comparison with RNN.

4.3. Reassembly learning

We propose a reassembly learning to design a single network for a multi-task problem which consists of dependent sub-tasks. Re-



(a) Network architecture (b) Block specification (c) Process of reassembly learning

Figure 1: Proposed system for DCASE2019 task 3; B: batch size, C: channel, T: time, M: number of mel-bin, F: filters, N: number of classes

Table 2: Hyper-parameters for proposed system

Name	Value
Epoch for SED network	25
Epoch for DOA network	100
Epoch for SELD network	10
Learning rate for SED/DOA network	0.001
Learning rate for SELD network	0.0001
Weight for SELD network	SED:DOA = 0.2:0.8
Batch size (B)	32
Frame length (T)	200
Step size between two segment	100
Number of Mel-bin (M)	96
Number of the FFT	1024

assembly learning consists of three stages. The first stage is training the SED network. After then CNN layers of the trained SED network are transferred to the DOA network. In the second stage, the loss of the DOA network is calculated with masking inactive event by the ground truth SED labels. The last stage is the SELD stage. The SELD network initializes the whole parameter from the pre-trained SED network and the DOA network as shown in Figure 1(c). At this stage, we use estimated SED for masking instead of ground truth SED labels.

As mentioned in the introduction, using the ground truth SED label masking causes an irony that DOAE for mispredicted events is made with random DOA predictions excluded from training. Reassembly learning is a technique to make the random DOA value of the mispredicted event closer to the default value through additional training. In short, the reassembly learning is a learning method that reduces randomness through additional fine-tuning. The trained SED network has more than 98% F-score for the training dataset. Therefore, reassembly learning plays a role in adjusting less than 2% outliers that are not detected correctly.

5. EVALUATION RESULT

In this section, we will describe network architectures using the proposed technique and network layers in the previous section and its performance. The hyper-parameters for training are summarized in Table 2.

5.1. Single network

Table 3 shows the experimental results for single networks by using pre-defined four-fold cross-validation split for DCASE 2019. We combined training and validation split for training proposed models. The system name and its description stand for following,

- **Baseline**: Baseline network model for DCASE 2019 challenge task 3 by task organizers
- **Reassembly 'SED-DOA'**: Proposed architecture as shown in Figure 1(a); 'SED-DOA' specifies the network that corresponds to the temporal feature extractor in SED and DOA networks. Candidates for temporal feature extractor are RNN and TrellisNet. For example, RNN-TrellisNet means RNN block used as SED temporal feature extractor and TrellisNet block used as DOA temporal feature extractor
- **SELD RNN-TrellisNet**: Proposed network structure without reassembly learning
- **Two-stage RNN-TrellisNet**: Proposed network structure with two-stage learning
- **Reassembly v1**: 4 GLU block for CNN layers of proposed model; DCASE challenge submission model

Table 3: Experimental results for DCASE 2019 task 3 development dataset; ER: error rate, F: F-score, DOA: DOA error, FR: frame recall

System	Feature	ER	F	DOA	FR
Baseline	FOA	34	79.9	28.5	85.4
Baseline	MIC	35	80.0	30.8	84.0
Reassembly RNN-TrellisNet	Log-Mel	16	90.9	10.2	88.1
Reassembly RNN-TrellisNet	Log-Mel + I_a	15	91.4	7.6	88.2
Reassembly RNN-TrellisNet	Log-Mel + I_a + I_r	**15**	**91.4**	**6.4**	**88.4**
Reassembly RNN-RNN	Log-Mel + I_a + I_r	15	91.4	8.8	88.4
Reassembly TrellisNet-RNN	Log-Mel + I_a + I_r	30	84.5	10.4	86.6
Reassembly TrellisNet-TrellisNet	Log-Mel + I_a + I_r	29	84.6	7.0	86.6
SELD RNN-TrellisNet with ground truth masking	Log-Mel + I_a + I_r	18	89.7	10.3	87.5
SELD RNN-TrellisNet with estimated SED masking	Log-Mel + I_a + I_r	19	89.3	9.4	86.9
Two-stage RNN-TrellisNet	Log-Mel + I_a + I_r	15	91.4	6.7	88.4
Reassembly v1 (Development dataset)	Log-Mel + I_a + I_r	16	90.6	6.4	85.7
Reassembly v1 (Evaluation dataset)	Log-Mel + I_a + I_r	15	91.9	5.1	87.4
Avg ensemble (SED + DOA), Reassembly RNN-TrellisNet	Log-Mel, Log-Mel + I_a, Log-Mel + I_a + I_r	13	92.7	7.7	88.9
Avg ensemble (SED), Reassembly RNN-TrellisNet	Log-Mel, Log-Mel + I_a, Log-Mel + I_a + I_r	**13**	**92.7**	**6.4**	**88.9**

Table 3 shows the results of applying three different input features to Reassembly RNN-TrellisNet. As a result, the higher the dimension of the input feature used, the higher the DOA result. These results show that spatial information and phase information from FOA were important for DOAE. On the other hand, SED results showed no significant change. This means that the log mel-band energy feature has been used primarily for SED. While the intensity vector feature does not help improve SED performance. This is because the direction does not need to be considered for SED.

SED is a problem that infers time-varying patterns. While DOAE for static events is a problem for estimating static phase difference. Therefore, we assumed that RNN would be advantageous in inferring the time-varying pattern for SED. On the other hand, we assumed that CNN would be more appropriate than RNN for estimating the static phase difference. These assumptions are proved in the results of Table 3. RNN is strong for SED and TrellisNet has strong point for DOAE. This result is that the TCN based network has an advantage in DOAE and is the possibility of being applied to a variety of sound classification and detection applications.

Reassembly RNN-TrellisNet system using Log-Mel + I_a + I_r is the best performance in a single model in Table 3. We submitted a single network, Reassembly v1, using 4 GLU blocks on CNN layers for Reassembly RNN-TrellisNet to DCASE 2019 challenge task 3. Reassembly v1 ranked 10th in DCASE 2019 challenge task 3 challenge. The model proposed in this paper has slightly changed the DCASE challenge submission model. By reducing the number of GLU blocks, the time pooling is reduced. So it brings 1%, 1%, and 3% performance gain for error rate, F-score and frame recall respectively.

The proposed system has achieved performance improvement over DCASE 2019 baseline. Compared to the performance of the proposed network without reassembly learning, pre-training subtask networks has proven to significantly improve the performance of all metrics. In Table 3 the reassembly learning showed a 0.4° improvement in DOA error compared to the two-stage learning. Reassembly learning has led to a small improvement since it plays a role in reducing randomness for mispredicted outliers.

5.2. Ensemble network

We use the simple and powerful ensemble method, average ensemble, to Reassembly RNN-TrellisNet model of the three different input features used in Table 3. Following is descriptions of the ensemble systems:

- **Avg ensemble (SED + DOA)**: Average ensemble for both SED and DOA prediction results of Reassembly RNN-TrellisNet.
- **Avg ensemble (SED)**: Average ensemble for SED prediction results of Reassembly RNN-TrellisNet + DOA prediction results from Log-Mel + I_a + I_r feature.

Table 3 shows that the Avg ensemble (SED) has better performance than the Avg ensemble (SED + DOA). In the case of SED, the performance improvement was achieved by using the average value of probability used for classification. However, the DOA value is directly obtained from the regression, so the performance of the average ensemble for DOAE is almost equal to the average error of the three systems. The overall performance was improved by using the average ensemble. For the development dataset, the Avg ensemble network makes 3%, 2%, and 3% performance gain for error rate, F-score, and frame recall compared to Reassembly v1.

6. CONCLUSION

We proposed reassembly learning to solve the sound event localization and detection problem which consists of two dependent subtasks. Reassembly learning is a way to retrain network that consists of pre-trained sub-task networks. Through reassembly learning, we tried to alleviate the problem of multi regression loss used for continuous polyphonic SELD. As a result, the proposed models significantly improved both SED and DOAE performance compared to the baseline. Also, we proved that the log mel-band energy and mel-band intensity are helpful input features for SED and DOAE. Moreover, the DOAE network using TrellisNet showed better performance than CRNN. Thus TCN based architecture demonstrated the possibility for other sound classification and detection applications.

7. ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2017-0-00050, Development of Human Enhancement Technology for auditory and muscle support)

8. REFERENCES

[1] S. Adavanne, A. Politis, J. Nikunen, and T. Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 34–48, Mar 2019.

[2] S. Adavanne, A. Politis, and T. Virtanen, "Direction of arrival estimation for multiple sound sources using convolutional recurrent neural network," in *26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 1462–1466.

[3] T. Hirvonen, "Classification of spatial audio location and content using convolutional neural networks," in *Proc. Audio Eng. Soc. Conv. 138*, May 2015.

[4] Y. Cao, Q. Kong, T. Iqbal, F. An, W. Wang, and M. D. Plumbley, "Polyphonic sound event detection and localization using a two-stage strategy," 2019. [Online]. Available: http://arxiv.org/abs/1905.00268

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov 1997.

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

[7] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," 2016. [Online]. Available: https://arxiv.org/abs/1609.03499

[8] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018. [Online]. Available: http://arxiv.org/abs/1803.01271

[9] ——, "Trellis networks for sequence modeling," in *International Conference on Learning Representation (ICLR)*, 2019.

[10] L. JiaKai, "Mean teacher convolution system for DCASE 2018 task 4," DCASE2018 Challenge, Tech. Rep., Sep 2018.

[11] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. P. Shah, "Large-scale weakly labeled semi-supervised sound event detection in domestic environments," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 19–23.

[12] V. Morfi and D. Stowell, "Data-efficient weakly supervised learning for low-resource audio event detection using deep learning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 123–127.

[13] W. Lim, S. Suh, and Y. Jeong, "Weakly labeled semi-supervised sound event detection using crnn with inception module," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 74–77.

[14] K. Koutini, H. Eghbal-zadeh, and G. Widmer, "Iterative knowledge distillation in R-CNNs for weakly-labeled semi-supervised sound event detection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 173–177.

[15] R. Harb and F. Pernkopf, "Sound event detection using weakly labelled semi-supervised data with gcrnns, vat and self-adaptive label refinement," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 83–87.

[16] Y. Guo, M. Xu, i. Wu, Y. Wang, and K. Hoashi, "Multi-scale convolutional recurrent neural network with ensemble method for weakly labeled sound event detection," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, 2018, pp. 98–102.

[17] L. Perotin, R. Serizel, E. Vincent, and A. Guerin, "CRNN-based multiple DOA estimation using acoustic intensity features for ambisonics recordings," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 22–33, Mar 2019.