SOUND EVENT LOCALIZATION AND DETECTION BASED ON CRNN USING DENSE RECTANGULAR FILTERS AND CHANNEL ROTATION DATA AUGMENTATION

Technical Report

Francesca Ronchini[1], *Daniel Arteaga*[1,2], *Andrés Pérez-López*[1,3]

[1] Universitat Pompeu Fabra, Barcelona
{francesca.ronchini01}@estudiant.upf.edu, {andres.perez, daniel.arteaga}@upf.edu
[2] Dolby Iberia, SL, Barcelona
[3] Eurecat, Centre Tecnologic de Catalunya, Barcelona

ABSTRACT

This technical report illustrates the system submitted to the DCASE 2020 Challenge Task 3: Sound Event Localization and Detection. The algorithm consists of a CRNN using dense rectangular filters specialized on recognize significant frequency features related to the task. In order to further improve the score and to generalize the system performance to unseen data, the training dataset size has been increased using data augmentation based on channel rotations and reflection on the *xy* plane in the First Order Ambisonic domain, which allow to improve Direction of Arrival labels keeping the physical relationships between channels. Evaluation results on the cross-validation development dataset show that the proposed system outperforms the baseline results, considerably improving Error Rate and F-score for location-aware detection.

Index Terms— Sound event detection, Direction of Arrival estimation, CRNN, First Order Ambisonic, data augmentation, SELD

1. INTRODUCTION

Sound event localization and detection **(SELD)** is the combined task of sound event detection **(SED)** and sound event localization **(SEL)**, which aim is the identification of the presence of independent or temporally-overlapped sound sources and their spatial location. In particular, SED requires to identify, at each time frame, the onset and offset of sound events and their correct classification, labeling the event. SEL is considered as the estimation of the sound event direction in space with respect to a microphone when an event is active, referred as direction-of-arrival (DOA) estimation.

Formerly, the SED and SEL have been explored as two standalone tasks. In fact, until 2018, the existing systems considering SELD as a unique task were limited and only one method was based on deep neural network [1], which localizes sound events but exclusively at a predefined grid of directions and a large number of output classes were required for an higher number of sound event labels and increased spatial resolution. In 2018, Advanne et al. introduced SELDnet [2], a convolutional recurrent neural network which simultaneously recognizes, localizes and tracks sound event sources in time, being the first method to address the localization and recognition of more than two concurrent overlapping sound events. The system has been proposed as baseline for the Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge 2019 Task 3 [3]. SELDNet has been presented as baseline system also for the DCASE Challenge 2020 Task 3 [3], with some modifications inspired by the highest ranked architectures of the previous year challenge submissions. Further information about the changes made in the baseline system for the Task 3 of this year challenge can be found at [4].

The methodology proposed in this report is based on Adavanne et al. SELDNet [2], including some of the adopted additions in the baseline algorithm of the DCASE 2020 Task 3, such as the use of log-mel spectral coefficients and acoustic intensity vector for the FOA format. However, the system proposed in this report differs from the baseline in different points such as (i) data augmentation, (ii) network architecture and (iii) training loss functions. With respect to (i), -90°, 90° and 180° channel rotations and reflection on the *xy* plane are used as data augmentation technique, implementing the *16 patterns* spatial augmentation as proposed by Mazzon et al. for the same task of last year's challenge [5]. The 16 patterns technique allows to augment DOA labels maintaining the physical relationships between channels. Regarding (ii), the network has been increased, adding 2 convolutional layers. Furthermore the receptive field has been expanded using dense rectangular filters (instead of squared ones) in order to make the network able to recognize frequency features relevant for the task. With regard to (iii), we used the same loss functions proposed in [2]. Binary cross-entropy loss is used for SED prediction task while mean square error (MSE) loss is used for DOA estimation.

Results on developments dataset are evaluated considering the evaluation metrics proposed by Mesaros et. al.n [6], considering the joint nature of localization-and-detection. The same are used as evaluation metrics for the challenge.

The rest of the report is structured as follows: Section 2 presents the methodology and the architecture of the proposed system. Section 3 describes the experiment setup. Sections 4 reports the development results compared with the baseline method. An holistic overview and conclusions are summarized in Sections 5. Code repository is openly available for reproducibility on Github [1].

2. METHODOLOGY

The method proposed for the DCASE Challenge Task 3 is based on Advanne et al. [2] system, with an alternative implementation. This section's purpose is to explain the details of the proposed system and how it differs from the baseline. Each step of the implementation will be deeper explained in its related sub-section.

[1] https://github.com/RonFrancesca/dcase2020-task3-fp

2.1. Features Extraction

Two formats of TAU-NIGENS Spatial Sound Events 2020 [7] are provided for DCASE 2020 task 3: First Order Ambisonic (FOA) and 4 channels from a Microphone Array (MIC) [4]. We only use Ambisonic format.

In this system, log mel magnitude spectrogram together with acoustic intensity vector are used as input features for the network. Both are represented in the log mel space to better concentrate the input information of the network, as proposed by Cao et. al in [8] and also implemented in the baseline system of the challenge.

FOA uses four channels to encode information of a directional sound field denoted as W, X, Y and Z. W corresponds to an omni-directional microphone recording the sound pressure. The signals X, Y and Z correspond to directional *figure-of-eight* microphones oriented along the components of the x, y and z axis, respectively, and measure the acoustic velocity of each directional components. The acoustic intensity vector expresses the power carried by sound waves per unit area in a direction perpendicular to that area. Since, as explained in [8], its inverse direction is the DOA, it makes sense to use it directly for DOA estimation. The intensity vector is computed as in [8].

2.2. The network

Figure 1 shows the overall architecture of our system with the relative parameters values used in the implementation of this method.

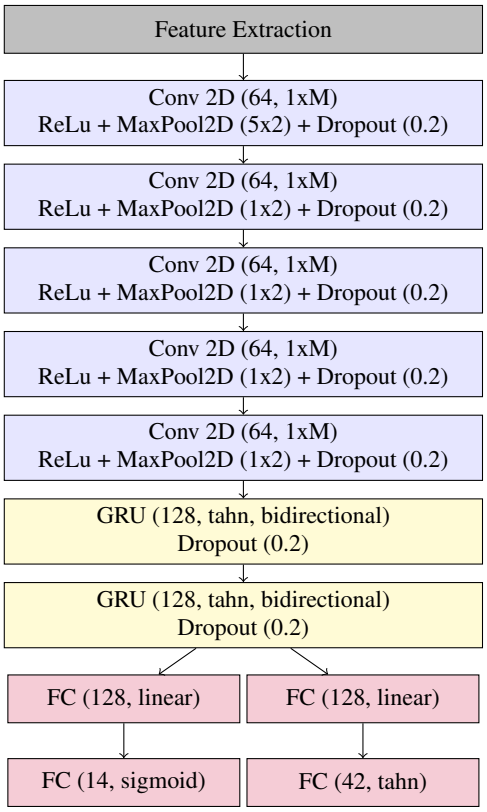

Figure 1: The proposed network architecture

The proposed system is based on Adavanne et al. SELDNet architecture as proposed in [2], with some modifications. Similarly to [2], it is a CRNN network using Gated Recurrent Units (GRU) as recurrent layers. This is followed by two parallel branches of Fully Connected (FC) layers, one for SED and one for DOA estimation, sharing weights along time dimension. The first FC layer of both branches uses linear activation while the last FC layer of each branch uses a different activation function according to the task. The last FC layer in the SED branch contains 14 nodes using sigmoid activation (one node for each sound event classes to be detected), while the last FC layer in the DOA branch consists of 42 nodes using tanh activation (each of the sound event classes is represented by 3 nodes relative to the sound event location in x, y, and z). We use binary cross-entropy as loss function for the SED branch and mean square error (MSE) loss for DOA estimation branch, keeping the two branches separate.

Regarding the changes made in this implementation, firstly, we added 2 CNN blocks in order to help the network to learn more features, increasing the number of CNN blocks from 3 to 5. Each CRNN block consists of a convolutional layer with rectified linear unit (ReLU) activation, Batch Normalization to normalize the activation output and MaxPooling along frequency axis to reduce the dimensionality. Although adding layers to a neural network use to help it to learn more features, it has the disadvantage of leading to possible overfitting, especially when the training dataset size is small as in this case. To prevent overfitting, we use Dropout in each convolutional block, after reducing the dimensionality. Secondly, we used dense rectangular filters instead of squared ones, mainly inspired by Pons et al. [9]. The authors tried to study how filters shape can influence and be used to proper model CNN motivated by musical aspects, reaching interesting results in music classifications. We propose the same concept, applying it to sound event detection. We use rectangular filters of shape *1 x M*, being *1* the time dimension and *M* the frequency dimension. We hypothesize that setting the time dimension to 1 would helped the network to better model frequency, helping the network to learn the presence or absence of an event, increasing the Error Rate (ER_{20°) and F-score (F_{20°) for location-aware detection.

The last addition is the use of data augmentation as described in Section 2.3 with the twofold scope of (i) preventing overfitting, (ii) increase the size of the dataset, expanding the number of DOA represented in it and consequently increase the scores related to Localization Error (LE_{CD}) and Localization Recall (LR_{CD}).

2.3. Data augmentation

With the aim of additionally increase the score and to reduce the overfitting of the system, the training dataset size has been increased using data augmentation based on channel rotations and reflection on the *xy* plane in the FOA domain. In particular, we implemented the *16 patterns* techniques proposed for the first time by Mazzon et. al in [5], with some small changes. This technique allows to improve DOA labels maintaining the physical relationships between channels. Moreover, a relevant advantage of this method is the possibility to be applied regardless of the number of overlapping sound sources [10]. We augmented the data following the transformations suggested in [10], considering only channel swapping and channel sign inversion. The suggested data manipulations correspond to rotations of 0, -90°, +90°, and +180° related to the azimuth angle, leading to 8 rotations about the z axis, and 2 reflections with respect to the xy plane (considering the opposite elevation angle), for a total of 15 new patterns plus the original one. Figure 2 shows an

example of channel rotation on intensity vector, after applying a reflection with respect to xy plane. On the figure, the 3 channels of the intensity vector are stacked into columns. The reader is referred to [10] for further details. In [5], Mazzon et. al computes the augmented dataset in time domain and extracted the features offline for each of the augmented waveforms. All the possible transformations are computed offline and the data generator randomly chooses between one of them at each iteration. In this system we implemented only 15 patterns, not considering the original one as data augmentation pattern. We also implement the data augmentation offline, but, for memory reason, instead of computing all the transformations for each audio file, we randomly select one out of the 15 patterns to augment the data during the feature extraction process. The same pattern selected for a particular audio file is used for augmenting the corresponding label. All the new generated files, together with the original ones, are used to train the network.

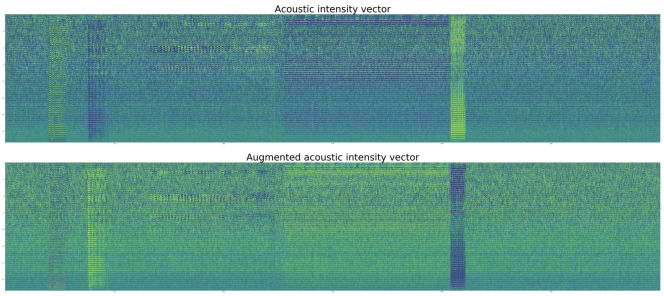

Figure 2: Example of augmented intensity vector. Pattern applied: reflection with respect to xy plane. The 3 channels of the intensity vector are stacked into columns. X and Y axes represent time and power dimension, respectively.

2.4. Hyper-parameters

We submitted different versions of the system, each with different hyper-parameters. More details are reported in Section 2.5.

All the dataset audio files are sampled at 24kHz. In all submission, we used a 960 point Hanning window with a 50% hop size. Considering that the temporal resolution of the label is at 100 ms, we interpolate the sub-frame of 20ms as suggested in the baseline system. The number of mel-band filter is set to 64.

For each audio file two channel rotations have been randomly selected between the 15 possible combinations, increasing the training development set from 400 to 1200 files.

With regard to the optimization technique, we use Adam method [11] as optimizer. Different hyper-parameters are considered in the different submissions for the learning rate and dimension of the rectangular filters used in the convolutional layer. More information is given in Section 2.5.

A sound event is considered to be active, and its respective DOA estimation considered, if the SED output exceeds a threshold of 0.5.

2.5. Variations of submitted versions

We submitted four different system outputs to the challenge. Each submission use different hyper-parameters of the network.

In submission *R_UPF_T3_1*, we use dense rectangular filters of dimension 1x48, being 1 the time dimension and 48 the frequency dimension. The learning rate is constant and set to 0.001. Submission *R_UPF_T3_2* uses 1x48 rectangular filters, using AveragePooling instead of MaxPooling. The learning rate is constant and set to 0.001 for the first 40 epochs, while it is decreased by 0.95% every next epoch. In submission *R_UPF_T3_3*, convolutional layers use MaxPooling and rectangular filters of 1x50, with constant learning rate of 0.001. In the last submission, submission *R_UPF_T3_4*, the architecture is configured with the same hyper-parameters used in submission 1, using only 3 convolutional layers instead of 5.

3. EXPERIMENTS

The dataset provided for this challenge is divided between development and evaluation set. The results presented in this report are based on the development set, which consists of 6 predefined cross-validation splits. In particular, we followed the same specification given in the task description, using split 1 for testing, split 2 for eval and split 3-6 for training.

The network predictions have been evaluated considering the joint nature of localization-and-detection, as proposed in [6]. In particular, ER_{20° and F_{20° are related to the SED task and they are location-dependent. A prediction is considered true positive only if under a distance threshold of 20° from the reference. LE_{CD} (localization error) and LR_{CD} (localization recall), are related to DOA estimation, being classification-dependent. Instead of considering all outputs, they are computed across each class only.

All metrics are computed in one-second non-overlapping frames. More information about the evaluation metrics can be found at [6].

Several architecture configurations, filter dimensions and data augmentation techniques have been explored before reaching the network architecture described in Section 2.2.
In particular, we trained the network with squared filters of size 3x3, 5x5, including different dilation rate to increase the receptive field. Those filters have been compared with rectangular filters *1xM* (1 is the time dimension and m the frequency dimension) of size 1x46, 1x48, 1x50, 1x52, 1x54, 1x56 and completely dense filters of dimension 1x64 (being 64 the size of mel-band filter). Rectangular filters always performed better than squared ones, with dimensions 1x48 and 1x50 being the ones that perform better.

Different data augmentation techniques have been tested, such as time stretching, pitch shifting and adding noise but only channel rotations helped to increase the accuracy of the network, considerably improving all metrics, especially SED metrics more than expected. This could be explained by the fact that applying data augmentation techniques such time stretching and pitch shifting affect the DOA in an unpredictable way. With channel rotation instead, the augmented data preserve physical relationships , not changing the signal but only its direction.
During all the experiments, the batch size has been set to 128 and the systems have been trained for 50 epochs at most. An early stopping strategy has been implemented, stopping the training if the validation loss does not improve during 50 epochs.

4. RESULTS AND DISCUSSION

Table 1 shows the evaluation results for the development dataset on the testing split, comparing it with the baseline. As it can be observed, all proposed methods outperform the baseline results, improving all the results, especially improving ER_{20° and F_{20°. The most significant contribution to the results has been the use of dense

rectangular filter, which suggests that using proper shape filters help the network to better model frequency features, learning the pattern to recognize when an event is active and when it is not, improving the SED metrics. The results also showed that increasing the architecture of the network adding 2 convolutional layer helps the network learn more features.

Regarding the use of adaptive learning rate, which in submission *R_UPF_T3_2* has been decreased by 0.95 each epoch for the last 10 epochs of train, we can conclude that it helps the DOA estimation, at the expense of SED task performance, comparing it with the best model proposed. Anyway, it is possible to confirm that the difference between the two is not drastic.

Method	ER_{20°	F_{20°	LE_{CD}	LR_{CD}	SELD
Baseline	0.72	37.4%	22.8°	60.7%	0.47
R_UPF_T3_1	**0.59**	**50.6%**	**17.6°**	**66.2%**	**0.38**
R_UPF_T3_2	0.60	49.9%	17.9°	66.8%	0.38
R_UPF_T3_3	0.61	48.7%	18.7°	65.2%	0.39
R_UPF_T3_4	0.61	48.4%	18.6°	65.6%	0.39

Table 1: Evaluation method on development set

5. CONCLUSIONS

This technical report illustrates the system submitted for the DCASE Challenge 2020 Task 3. The method is based on Adavanne et. al SELDNet [2], with some differences. A part of some hyper-parameters, the main changes are (i) data augmentation based on ambisonic rotation, (ii) network architecture and (iii) training loss functions. The main improvement of the proposed method is the use of dense rectangular filters. Data augmentation also helped to increase the evaluation score, especially ER_{20° and F_{20° related to the SED task. The proposed system considerably outperforms the state-of-the system presented as baseline before the submission deadline, significantly increasing the location-dependent metrics related to SED task.

6. REFERENCES

[1] T. Hirvonen, "Classification of spatial audio location and content using convolutional neural networks," in *Audio Engineering Society Convention 138*. Audio Engineering Society, 2015.

[2] 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, 2018.

[3] http://dcase.community/challenge2019/task-sound-event-localization-and-detection.

[4] http://dcase.community/challenge2020/task-sound-event-localization-and-detection.

[5] L. Mazzon, M. Yasuda, Y. Koizumi, and N. Harada, "Sound event localization and detection using foa domain spatial augmentation," in *Proc. of the 4th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2019.

[6] A. Mesaros, S. Adavanne, A. Politis, T. Heittola, and T. Virtanen, "Joint measurement of localization and detection of sound events," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, Oct 2019, accepted.

[7] 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.

[8] 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," *arXiv preprint arXiv:1905.00268*, 2019.

[9] J. Pons, T. Lidy, and X. Serra, "Experimenting with musically motivated convolutional neural networks," in *2016 14th international workshop on content-based multimedia indexing (CBMI)*. IEEE, 2016, pp. 1–6.

[10] L. Mazzon, Y. Koizumi, M. Yasuda, and N. Harada, "First order ambisonics domain spatial augmentation for dnn-based direction of arrival estimation," *arXiv preprint arXiv:1910.04388*, 2019.

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