AUDIO EVENT DETECTION AND LOCALIZATION WITH MULTITASK REGRESSION NETWORK

Technical Report

Huy Phan[*1], *Lam Pham*[2], *Philipp Koch*[3], *Ngoc Q. K. Duong*[4], *Ian McLoughlin*[5], *Alfred Mertins*[3],

[1] School of Electric Engineering and Computer Science, Queen Mary University of London, UK
[2] School of Computing, University of Kent, UK
[3] Institute for Signal Processing, University of Lübeck, Germany
[4] InterDigital R&D France, France
[5] Singapore Institute of Technology, Singapore
[*]Corresponding email: h.phan@qmul.ac.uk

ABSTRACT

This technical report describes our submission to the DCASE 2020 Task 3 (Sound Event Localization and Detection (SELD)). In the submission, we propose a multitask regression model, in which both (multi-label) event detection and localization are formulated as regression problems to use the mean squared error loss homogeneously for model training. The deep learning model features a recurrent convolutional neural network (CRNN) architecture coupled with self-attention mechanism. Experiments on the development set of the challenge's SELD task demonstrate that the proposed system outperforms the DCASE 2020 SELD baseline across all the detection and localization metrics, reducing the overall SELD error (the combined metric) approximately 10% absolute.

Index Terms— audio event detection, localization, regression, self-attention

1. INTRODUCTION

Extended from active research on sound (audio) event detection, sound event localization and detection (SELD) task [1, 2] entangles the *what* and *where* questions about occurring sound events. That is, it aims to determine the identities of the events and their spatial locations/trajectories simultaneously. Solving the SELD task would enable a wide range of novel applications in surveillance, human-machine interaction, bioacoustics, and healthcare monitoring, to mention a few.

The joint SELD task can be divided and conquered individually by two separate models, one for sound event detection (SED) [3, 4, 5] and the other for sound source localization (SSL) [6, 7]. Two-stage approach presented in [8] can be also considered to belong to this line of work. Dealing with the joint task in a single model has been known to be more challenging. Three main approaches have been proposed, including sound-type masked SSL [6], spatially mask SED [9], and joint SELD modeling [10, 2]. Joint sound event detection and localization modeling with multitask deep learning has been most commonly adopted in the latest DCASE challenge [11, 12, 13, 2], demonstrating encouraging results.

Typically, in the joint modeling approach with a multitask deep learning model, the binary cross-entropy loss is typically used for event detection (via classification) to handle possible multi-label due to occurrences of multiple events while the mean squared error (MSE) loss is often employed for direction of arrival (DOA) estimation (via regression). These two losses are usually associated with different weights and then combined to make the total loss for network training. However, there exist no established rules to set the weights for the losses; more often than not, they are set with some trivial weights without clear justification. For example, while the DCASE 2019 baseline weighted the MSE loss 50 larger than that of the binary cross-entropy loss, the current DCASE 2020 baseline even enlarges this multiplication to 1000 times. Furthermore, the two different types of loss functions might be progressing at different rates during the training and might not converge at the same time, making the fix weights suboptimal.

In order to avoid such an issue, we propose to formulate both the SED and SSL subtasks as regression problems and homogeneously use the MSE loss for both of them. The proposed network features a recurrent convolutional neural network (CRNN) architecture coupled with self-attention mechanism [14]. Experiments on the development set of the DCASE 2020 Task 3 show that our proposed network outperforms the DCASE 2020 SELD baseline across all the evaluation metrics, some with a large margin. Using the first-order Ambisonics (FOA) data, we achieve 19.0°, 65.6%, 0.60, and 49% on the localization error (LE_{CD}), localization frame recall (LR_{CD}), detection error (ER_{20°) and detection F1-score (F_{20°), respectively. The corresponding results obtained by using the tetrahedral capsule arrangement (MIC) data are 18.2°, 64.1, 0.59, and 0.38. In comparison with the DCASE 2020 SELD baseline [1], we achieve the combined SELD error rates of 0.39 and 0.38 using the FOA and MIC data, respectively, reducing 0.08 and 0.11 absolute from that of the baseline.

2. THE PROPOSED NETWORK

The proposed network is illustrated in Figure 1. The network receives time-frequency input $\mathbf{S} \in R^{T \times F \times C}$ of T frames, F frequency bins, and C channels. The convolutional part of the network consists of six convolutional layers each of which is followed by a max pooling layer except the first one. Since we assume that the early convolutional layers are crucial for feature learning, the network is designed to have the first two convolutional layers

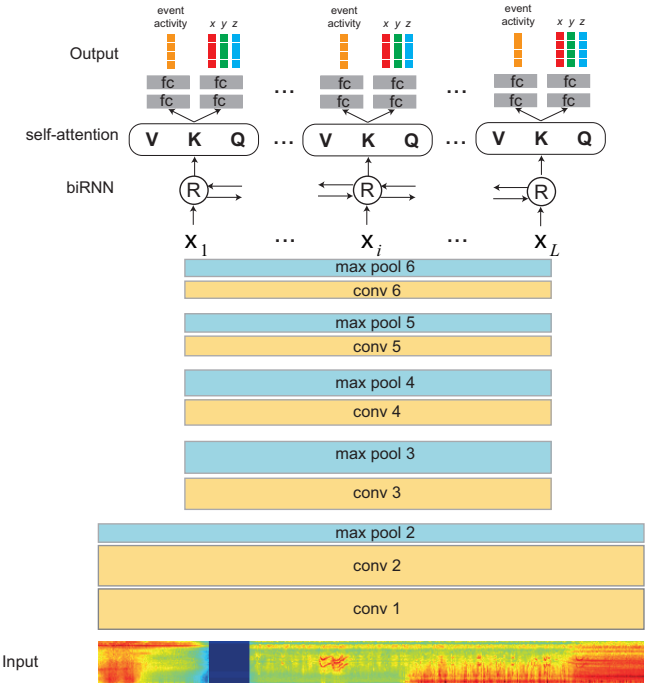

Figure 1: Overview of the proposed multitask regression self-attention CRNN.

back-to-back. In order, the six convolutional layers accommodate $\{64, 64, 128, 128, 256, 256\}$ filters, respectively, with a common kernel size of 3×3 and the stride of 1×1. The gradually larger numbers of filters in the later convolutional layers are to compensate for their smaller feature maps in the frequency dimension. Zero-padding (i.e. *SAME* padding) is used in order to reserve the temporal size. After convolution, batch normalization [15] is applied on the feature maps, followed by Rectified Linear Units (ReLU) activation [16].

The max pooling layers, except the first one, have a common kernel size of 1×2 to reduce the input size by half in the frequency dimension and, by doing so, gain frequency invariance in the induced feature maps while keeping the temporal size unchanged. Particularly, the pooling kernel size of the first max pooling layer (*max pool 2*, cf. Figure 1) is set to 5×2 in order to reduce the time dimension to $\frac{T}{5}$ to match the frame resolution (100 ms) for computing the evaluation metrics.

Passing through the convolutional block, the input is transformed into a feature map of size $\frac{T}{5} \times 2 \times 256$ which is reshaped to form a sequence of feature vectors $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{\frac{T}{5}})$ where $\mathbf{x}_i \in R^{512}$, $1 \leq i \leq \frac{T}{5}$. A bidirectional recurrent neural network (biRNN) is then employed to iterate through the sequence and encode it into a sequence of output vectors $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{\frac{T}{5}})$. The biRNN is realized by Gated Recurrent Unit (GRU) cells with the hidden size of 256. To further improve encoding the context around a feature $\mathbf{z}_i$, self-attention mechanism [14] is used. Viewing the vectors $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{\frac{T}{5}})$ as a set of *key-value* pair $(\mathbf{K}, \mathbf{V})$. In the context of this work, both the keys and values coincide to $\mathbf{Z}$ (the concatetation of the $\mathbf{z}$ vectors). We adopt the scaled dot-product attention as in [14], i.e. the attention output at a time index is a weighted sum of $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{\frac{T}{5}})$ where the weights are determined as:

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^{\mathsf{T}}}{\sqrt{d_k}})\mathbf{V}. \quad (1)$$

Here, $\mathbf{Q}$ is the *query* [14] and also coincides to $\mathbf{Z}$ in the context of this work, i.e. $\mathbf{Q} \equiv \mathbf{K} \equiv \mathbf{V} \equiv \mathbf{Z}$. d_k is the extra dimension into which $\mathbf{Q}$, $\mathbf{K}$ are transformed before the dot product to prevent the inner product from becoming too large. d_k is set to 64 in this work.

At each time index, the SED and SSL subtasks are accomplished via two network branches each consists of two fully connected layers with 512 units each. The first branch's output layer has C units with *sigmoid* activation to perform event activity regression of C classes. The second branch has $3C$ units with *tanh* activation to regress for the DOA trajectory of the C event classes. The network is trained to minimize the total MSE loss of the two network branches without weighting, which is averaged by the sequence length $\frac{T}{5}$.

3. EXPERIMENTS ON DCASE 2020 SELD DEVELOPMENT SET

3.1. DCASE 2020 SELD development set

Data of the DCASE 2020 SELD task is synthesized in two spatial sound formats: (1) MIC - 4-channel microphone array extracted from a subset of 32-channel Eigenmike format and (2) FOA - 4-channel first-order Ambisonics extracted from a matrix of 4×32 conversion filters. More information about the data synthesis can be found in [1]. The development set of the DCASE 2020 SELD task consists of six sets of 1-minute recordings. Following the Task's setup, the first set was used as the unseen data for testing purpose, the second set was used as the validation set for model selection, and the remaining four sets were used as the training data.

3.2. Feature extraction

Following the procedure of the DCASE 2020 SELD baseline, we extracted log-Mel magnitude spectrogram with a window size of 40 ms, 20 ms overlap, and 64 Mel-bands. To encode the phase information, for the FOA data, an acoustic intensity vector was extracted for each Mel-band, whereas, for the MIC data, generalized-cross-correlation with phase-transform (GCC-PHAT) features were computed for each Mel-band. Overall, multi-channel images of size $3000 \times 64 \times 7$ and $3000 \times 64 \times 10$ were resulted for one-minute FOA and MIC recordings, respectively.

3.3. Parameters

The proposed network was implemented using *Tensorflow* framework. We used spectrogram segments of size $T = 600$ (equivalent to 2 seconds) as inputs. *Dropout* rates of 0.5, 0.1, and 0.25 were employed to regularize the convolutional layers, the biRNN, and the fully-connected layers, respectively.

The network was trained using *Adam* optimizer [17] for 10000 epochs with a minibatch size of 64. Each spectrogram segment in a minibatch was randomly sampled from the 1-minute recording and augmented using spectrogram augmentation [18]. The learning rate was initially set to 2×10^{-4} and was exponentially reduced with a rate of 0.8 after 200, 600, and 1000 epochs. In addition, the first 10 epochs were used as a warmup period in which the network was trained with a small learning rate of 2×10^{-5}.

Table 1: Results obtained by the proposed system and the DCASE 2020 baseline on the development and evaluation sets.

	FOA					MIC				
	LE_{CD}	LR_{CD}	ER_{20°	F_{20°	$SELD$	LE_{CD}	LR_{CD}	ER_{20°	F_{20°	$SELD$
Development results										
Val (Baseline)	23.5°	62.0	0.72	37.7	0.46	27.0°	62.6	0.74	34.2	0.48
Test (Baseline)	22.8°	60.7	0.72	37.4	0.47	27.3°	59.0	0.78	31.4	0.51
Val	$\mathbf{17.7^\circ}$	**68.1**	**0.58**	**52.4**	**0.37**	$\mathbf{17.3^\circ}$	**66.0**	**0.56**	**53.9**	**0.37**
Test	$\mathbf{19.0^\circ}$	**65.6**	**0.60**	**49.2**	**0.39**	$\mathbf{18.2^\circ}$	**64.1**	**0.59**	**50.8**	**0.38**
Evaluation results										
System 1	$\mathbf{16.8^\circ}$	**69.8**	**0.52**	**57.8**	**0.33**	–	–	–	–	–
System 2	–	–	–	–	–	$\mathbf{14.6^\circ}$	**68.2**	**0.55**	**58.8**	**0.34**
System 3	$\mathbf{15.2^\circ}$	**72.4**	**0.49**	**61.7**	**0.31**	–	–	–	–	–
System 4	–	–	–	–	–	$\mathbf{14.6^\circ}$	**68.2**	**0.53**	**59.2**	**0.33**

During training, the network snapshot that achieved the lowest combined SELD error rate on the validation set was retained for evaluation. The retained network was then exercised on the test recordings with a 2-second segment at a time without overlap. To determine event activity from the corresponding regression output, a threshold of 0.5 was applied. No further post-processing was carried out.

3.4. Evaluation metrics

For sound event detection, the DCASE 2020 evaluates the performance of the SEL subtask using localization-aware detection error rate (ER_{20°) and F-score (F_{20°) with a threshold of 20° in one-second non-overlapping segments. For sound event localization, errors only between same-class predictions and references are considered. The class-aware localization error (LE_{CD}) and its corresponding recall (LR_{CD}) are employed for evaluating localization outputs and are also computed in one-second non-overlapping segments. In addition, we also computed the combined SELD error metric:

$$SELD = ER_{20^\circ} + (1 - F_{20^\circ}) + \frac{LE_{CD}}{180} + (1 - LR_{CD}) \quad (2)$$

to give an overall picture about a system.

3.5. Experimental results

The results obtained by the proposed system on the development set are shown in Table 1. In comparison to the DCASE 2020 SELD baseline the proposed system achieves better results across the evaluations metrics, some with a large margin. On the FOA data, the proposed system obtains 19.0°, 65.6%, 0.60, and 49% on the localization error (LE_{CD}), localization frame recall (LR_{CD}), detection error (ER_{20°) and detection F1-score (F_{20°), respectively. The corresponding results obtained using the MIC data are 18.2°, 64.1, 0.59, and 0.38. Overall, the proposed system reduces the combined SELD error by 0.08 and 0.11 absolute from that of the baseline.

4. DCASE 2020 SUBMISSION

In a similar procedure, we built four systems for DCASE 2020 SELD task submission.

- **System 1**: The network was trained using five recording sets (2-6) from the FOA data of the development set while the first recording set was used as the validation set.
- **System 2**: Similar to **System 1** but the MIC data was used.
- **System 3**: All six recording sets (i.e. without validation data for model selection) from the FOA data of development were used for training.
- **System 4**: Similar to **System 3** but the MIC data was used.

5. CONCLUSIONS

In this technical report, we presented the proposed system upon which four submission systems were built for the DCASE 2020 SELD task. We approach joint modeling for sound event detection and localization as a multitask regression problem so that the MSE loss can be used homogeneously for both the two subtasks. The proposed network features a CRNN architecture, which is popular for sound event detection, and self-attention mechanism. Experimental results on the development set of the the DCASE 2020 SELD task show significant improvements over the baseline across all the evaluation metrics.

6. REFERENCES

[1] 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 2006.01919*, 2020.

[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] H. Phan, O. Y. Chén, P. Koch, L. Pham, I. McLoughlin, A. Mertins, and M. D. Vos, "Unifying isolated and overlapping audio event detection with multi-label multi-task convolutional recurrent neural networks," in *Proc. ICASSP*, 2019.

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

[5] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, W. Xiao, and H. Phan, "Continuous robust sound event classification using time-frequency features and deep learning," *PLoS ONE*, vol. 12, no. 9, 2017.

[6] N. Ma, J. A. Gonzalez, and G. J. Brown, "Robust binaural localization of a target sound source by combining spectral

source models and deep neural networks," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 26, no. 11, p. 2122–2131, 2018.

[7] R. Chakraborty and C. Nadeu, "Sound-model-based acoustic source localization using distributed microphone arrays," in *Proc. ICASSP*, 2014, p. 619–623.

[8] Y. Cao, Q. Kong, T. Iqbal, F. An, W. Wang, and M. Plumbley, "Polyphonic sound event detection and localization usinga two-stage strategy," in *Proc. Detection and Classification ofAcoustic Scenes and Events 2019 Workshop (DCASE2019)*, 2019.

[9] I. Trowitzsch, C. Schymura, D. Kolossa, and K. Obermayer, "Joining sound event detection and localizationthrough spatial segregation," in *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 28, 2020, pp. 487–502.

[10] W. He, P. Motlicek, and J.-M. Odobez, "Joint localization and classification of multiple sound sources using a multi-task neural network," in *Proc. Interspeech*, 2018.

[11] F. Grondin, I. Sobieraj, M. Plumbley, and J. Glass, "Sound event localization and detection using crnn on pairs of microphones," in *Detection and Classification of Acoustic Scenesand Events 2019 Workshop (DCASE2019)*, 2019.

[12] H. Cordourier, P. L. Meyer, J. Huang, J. D. H. Ontiveros, and H. Lu, "Gcc-phat cross-correlation audio featuresfor simultaneous sound event localization and detection(seld) on multiple rooms," in *Proc. Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, 2019.

[13] S. Kapka and M. Lewandowski, "Sound source detection, localizationand classification using consecutive ensemble ofcrnn models," in *Procc. Detection and Classification of AcousticScenes and Events 2019 Workshop (DCASE2019)*, 2019.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.

[15] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.

[16] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. ICML*, 2010.

[17] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–13.

[18] 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. Interspeech*, 2019, pp. 2613–2617.