GCC-PHAT CROSS-CORRELATION AUDIO FEATURES FOR SIMULTANEOUS SOUND EVENT LOCALIZATION AND DETECTION (SELD) IN MULTIPLE ROOMS

Héctor A. Cordourier Maruri[1], *Paulo López Meyer*[1], *Jonathan Huang*[2], *Juan Antonio del Hoyo Ontiveros*[1], *Hong Lu*[2],

[1] Intel Corporation, Intel Labs, Zapopan, Jal., Mexico,
{hector.a.cordourier.maruri, paulo.lopez.meyer, juan.antonio.del.hoyo.ontiveros}@intel.com
[2] Intel Corporation, Intel Labs, Santa Clara, CA, USA,
{jonathan.huang, hong.lu}@intel.com

ABSTRACT

In this work, we show a simultaneous sound event localization and detection (SELD) system, with enhanced acoustic features, in which we propose using the well-known Generalized Cross Correlation (GCC) PATH algorithm, to augment the magnitude and phase regular Fourier spectra features at each frame. GCC-PHAT has already been used for some time to calculate the Time Difference of Arrival (TDoA) in simultaneous audio signals, in moderately reverberant environments, using classic signal processing techniques, and can assist audio source localization in current deep learning machines. The neural net architecture we used is a Convolutional Recurrent Neural Network (CRNN), and is tested using the sound database prepared for the Task 3 of the 2019 DCASE Challenge. In the challenge results, our proposed system was able to achieve 20.8 of direction of arrival error, 85.6% frame recall, 86.5% F-score and 0.22 error rate detection in evaluation samples.

Index Terms— GCC-PHAT, SELD, Polyphonic event detection, Sound source localization, CRNN, Sound event detection.

1. INTRODUCTION

Sound event detection (SED) or Audio Classification, refers to the task of automatically recognizing the type of sound that is being detected into some previously specified classes (like human voice, vehicle moving, music, etc.). Meanwhile, Sound Source Localization, or Sound Direction of Arrival (DoA) detection determines the location of the sound in some coordinate system, and in this task we use elevation and azimuth angles as its proxy. In most current published work, these two tasks have been approached as separate problems. However, there are many applications in which the simultaneous location and identification of the sound can be very useful, like detection of an intended user, observation and understanding of human activities, audio surveillance, autonomous agent navigation, among others[1]. In most of this real-life applications, it is reasonable to assume that sources sometimes will overlap in time. A detection pipeline of this kind of audio events is proposed in [1], and named as polyphonic SED. Such work proposes an interesting CRNN architecture to perform the task, and in this work, we present a system based in that architecture, but with additional features based in the Generalized Cross Correlation (GCC), and provide measurements of the benefits obtained.

2. SOUND EVENT DETECTION

In current literature, Convolutional Neural Networks (CNNs) have been proven to be very effective for image classification tasks. A natural next step was to use CNNs or similar systems for audio classification, providing audio features that resemble images, usually Fourier-based spectrograms, or similar representations. This approach has also been met with relative success[2], and in recent DCASE acoustic scene classification tasks, top submissions are mostly CNN-based or related[3].

2.1. Sound source localization

Estimating the location of a sound source is definitely not a new engineering problem. In classical signal-processing systems, source location is calculated from the Time Difference of Arrival (TDoA) of the signal in each element of a microphone array. Then, an analytic, regression formula, or a machine learning (ML) technique can be used to produce the source location. For the first step, regular cross-correlation can detect the time delay of two signals that contain little auto-correlation (i.e. low reverberation, rich frequency content sounds). In that sense, generalized cross-correlation with phase transform (GCC-PHAT) algorithm, developed in 1976 by Knapp and Carter [4], can reduce the effects of the auto-correlation of a signal, and make the system more robust to reverberation.

However, new machine learning techniques usually do not rely on mapping TDoA to spatial location. Instead, the trend is to directly relate some features of the audio signals to the source location[5, 6, 7]. In our work, we aim to get the best of both worlds: the tractability of a classical signal processing as feature, and the high accuracy and noise robustness of neural nets.

3. AUDIO FEATURE EXTRACTION

3.1. Development data set

For all this work, training and testing was performed using the development data set made available by the 2019 DCASE Challenge for the task 3[8], consisting of 400 recordings of roughly 60 seconds each. In order to identify the acoustic characteristics of the recordings, the average spectrum of all the development audio samples was calculated, and the result can be seen in Fig. 1. It can be noticed that there is practically no audio information above the 15.8 kHz frequency, even when the sample frequency (48 kHz) allows up

https://doi.org/10.33682/3re4-nd65

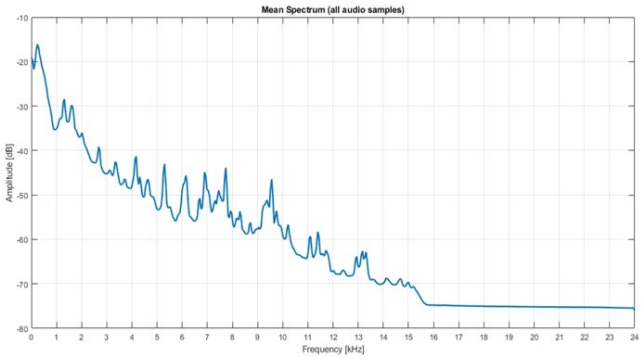

Figure 1: Average magnitude spectrum of all the database audio samples. Notice that there is practically no frequency content above 15.8 kHz.

to 24 kHz. Therefore, in all our feature extraction, we only include data information of frequency bins of up to 15.8 kHz.

The inputs of the feature extraction routine are 4-channel, 48kHz sample frequency, audio recordings. The signals are segmented in time frames, for which a time hop of 20ms is used, producing 3000 samples per each 60-second audio sample. The development data set consists of pre-defined four cross-validation splits of 200 one-minute samples for training, 100 samples for validation, and 100 for testing. The metrics shown in the results section refer to the average of all folds models in their corresponding testing set. The recordings include audio of 11 different classes, that can be located in azimuth angles of -180 to 170 degrees in 10 intervals, and elevations of -40 to 40 in 10 intervals. In preparation for the training routine, all the audio samples were padded or clipped to even out their run time to 60 seconds.

The first feature obtained from each time frame is the GCC-PHAT vector from each signal combination. Usually, the vector obtained from two signals from a microphone array shows a delta-like response close to the vector center, in which the maximum value of the vector has an offset from the center numerically equal to the amount of delay samples between the two time domain signals[4]. Therefore, it is the middle part of the vector which contains most of the useful information. We proposed to segment it, to generate time matrices we call GCC-grams, as an analogous name to spectrograms. An scheme of this process can be seen in Fig. 2.

We propose GCC-grams as an additional input feature, in which all the GCC-grams obtained from all channel combinations (6 in total, for the case of 4 input channels) are concatenated in one single feature matrix in which all the time frames are aligned. Our hypothesis is that this additional pre-processing of audio data can directly improve DoA detection performance. However, in order to keep all input information available for the system, we also feed the magnitude and phase spectra per each channel, properly synchronized. An scheme of this process can be seen in Fig. 3.

For the magnitude and phase spectrograms, the discrete Fourier transform was applied to time frames of 2048 samples, using a Hann window (these numbers were used also for the GCC vector calculation). From the positive frequencies of the output spectra (1024 samples), only the samples corresponding to frequencies below 15.8 kHz were extracted, which produced a vector of $1024 \times (15.8kHz/24kHz) = 672$ samples per each time frame.

In order to keep the same dimensions, the concatenated GCC-gram was fixed to have 672 samples wide too. Therefore, the middle section size of each of the six individual GCC-grams was set up to $672/6 = 112$ samples wide, which was large enough to contain the maximum value of the GCC vector in all recordings. Therefore, per each 1-minute recording, the feature extraction routine produces a 3-D tensor of 3000 time frames, by 672 frequency bins, by 9 components: 1 concatenated GCC-gram, 4 phase spectrograms, and 4 magnitude spectrograms. As a final step, all this 2D matrices were individually normalized.

To gather additional insight in the benefits of our proposed feature extraction technique, additional tests were performed in which only the GGC-grams, or only the magnitude and phase spectrograms, were fed into the proposed CRNN.

4. PROPOSED SYSTEM

4.1. CRNN Architecture

The CRNN architecture we used is based directly in the work proposed by S. Adavanne et al in [1], with some minimal modifications, as our approach was mainly focused on enhancing feature extraction. This architecture is characterized by taking a sequence of features in consecutive frames as input and predicting the sound event classes that are active for each of the input frames along with their respective spatial location (defined as a couple of output angles, azimuth and elevation), producing the temporal activity and DOA related information for each sound event class in parallel. An illustration of the final architecture proposed can be seen in Fig. 4.

The input of the neural network is composed of multiple 2D CNN layers. Each CNN layer has 64 filters of $3 \times 3 \times 9$ receptive fields with a ReLU activation function. After each CNN layer, the outputs are normalized using batch normalization, and the dimensions a reduced with average-pooling (MP_i) along the frequency axis. We preferred average-pooling over max-pooling on the hypothesis that average pooling carries more information from the whole kernel, in this case, the spectrograms and GCC-grams. The output after the final CNN is of dimension $T \times 2 \times 64$, in which T is the number of input time frames.

The output is reshaped to a T frame sequence of 128 feature vectors, fed to two GRU bidirectional layers of 128 nodes, followed by three identical branches of fully connected (FC) layers in parallel, one for SED, one for azimuth and other for elevation detection. The first FC layer consists of 256 input nodes with linear activation, followed by a Dropout layer, a SELU layer, and and finally a linear layer with 12 outputs, one per each audio class, plus an additional "garbage" class for the frames in which there is no audio event present.

4.2. Training parameters

In each frame, one-hot encoding target values were used for each of the active sound events in the event detection branch output. Since sound events can be overlapping in time, it is possible to have multiple ones at each time step. Similarly, for the azimuth and elevation branches, a 12 element vector is produced as output in which the active class (according with the event detection output) contains the numeric value of the angle in degrees, and the rest of the vector contains the garbage value of -181. A multi-classification hinge (margin-based) loss is used between the event detection predictions of our system and the reference sound class activities, while a mean square error (MSE) loss is used for both the azimuth and elevation

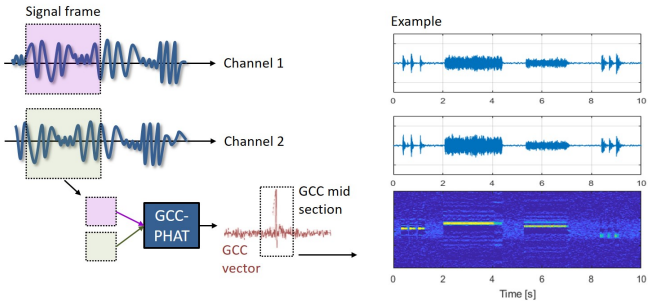

Figure 2: Scheme of the process in which a GCC-gram is produced from the audio signals.

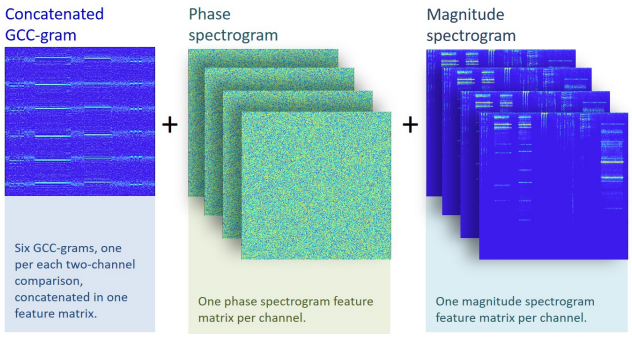

Figure 3: Scheme of the complete contents of the audio features extracted from the 4-channel signal.

outputs, for the whole matrices (including the garbage values). An additional MSE loss value was calculated for both angles, in which the garbage values were "masked out" of the MSE difference, using the target matrices. After some tests, the best result were obtained with a weighted sum of these three loss values. Additionally, we used spherical coordinates in all our calculations.

Training was performed by 150 epochs using Adam optimizer with batch size of 8 training samples, drop out rate of 0.5, and initial learning rate of 0.0001, using a cosine annealing scheduler. Early stopping is used to control the network from over-fitting to training split. The network was implemented using Pytorch.

5. RESULTS

5.1. In house tests on DCASE development data

As established in the DCASE Task 3 web page, four different metrics are taken onto account to assess the SELD system performance. Two are directly related with event detection: F-score and error rate (ER). The other two are related with DOA detection: DOA error and frame recall[9]. It must be highlighted that an ideal SELD method will have an error rate of 0, F-score of 100%, DOA error of 0 and frame recall of 100%. Also, as suggested in the page, the four cross-validation folds are to be treated as a single experiment, meaning that metrics are calculated only after training and testing all folds. The results are then compared with those from the baseline SELD system proposed by S. Adavanne et al[1], and the results of using only a part of the input features (GCC-gram only and magnitude-and-phase spectrums only). Such comparison of system results can

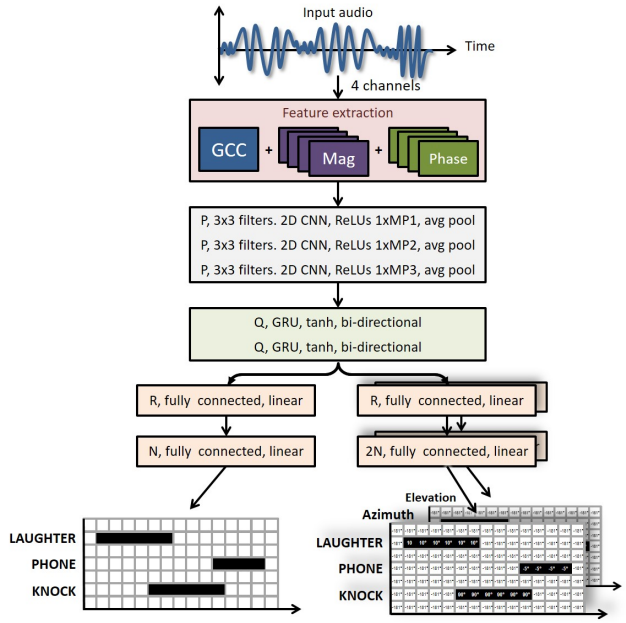

Figure 4: Overview of the architecture of the proposed sound event localization and detection (SELD) system

be seen in Table 1.

Model	ER	F-score	DOA error	Frame recall
Baseline (mic.)	0.35	80.0%	30.8	84.0%
GCC-Gram only	0.31	78.6%	29.9	80.9%
Spectrums only	0.25	83.8%	30.8	84.8%
Our system	**0.20**	**87.1%**	**20.4**	**86.4%**

Table 1: Results from the in-house tests.

As can be seen, using only GCC-grams or magnitude-and-phase spectrums as input features produce results in which the four metrics are very close to the baseline (with some slight improvement in the ER metric, that could be attributed to the small changes in the network architecture). However, it is only when the two feature types are fused in the input that the best results are obtained in all metrics. The best improvements over the baseline are present in the error rate (-0.15) and DOA error (-10.4). Then, we can attribute part of this performance jump to the additional prepossessing of audio events provided by GCC-grams.

5.2. Submission to the Task 3 of the DCASE Challenge with evaluation data

For the task 3 DCASE challenge, two sets were submitted, one with a CRNN trained with the fold with the best results in development data, and another with a fusion of the four folds. Table 2 shows the official results obtained in the four metrics considered, and the absolute ranking of each submission, in comparison with the two baseline options, and the best performing system.

Model	ER	F-score	DOA error	Frame recall	Rank
Baseline (mic.)	0.30	83.2%	38.1	83.4	58
Baseline (FOA)	0.28	85.4%	24.6	85.7	48
Our system (1-fold)	0.22	86.3%	19.9	85.6	46
Our system (fusion)	0.22	86.5%	20.8	85.7	45
Best system	**0.08**	**94.7%**	**3.7**	**96.8%**	**1**

Table 2: Results from the DCASE Challenge (Task 3).

These results ranked us in the 16th best out of 24 participating teams.

6. CONCLUSIONS

In this work we describe a system for the simultaneous audio event classification and location, based in the use of regular Fourier spectrograms and our proposed GCC-grams, in order to improve detection and localization performance over a previous baseline. Some additional changes in the CRNN architecture are also included, with the hypothesis of improved robustness over the baseline system. However, the main differentiation of our approach is clearly on the feature extraction side. The results obtained from the cross validation results show that our system performs better than the baseline in all the metrics proposed by the DCASE Challenge coordination team, which suggests that the additional processing at the feature extraction stage we proposed can produce significant additional benefits over an already properly functioning NN architecture.

7. 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, March 2019.

[2] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017. [Online]. Available: https://arxiv.org/abs/1609.09430

[3] A. Mesaros, T. Heittola, and T. Virtanen, "A multi-device dataset for urban acoustic scene classification," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, November 2018, pp. 9–13. [Online]. Available: https://arxiv.org/abs/1807.09840

[4] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, August 1976.

[5] X. Xiao, S. Zhao, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 2814–2818.

[6] R. Takeda and K. Komatani, "Discriminative multiple sound source localization based on deep neural networks using independent location model," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, Dec 2016, pp. 603–609.

[7] S. Chakrabarty and E. A. P. Habets, "Broadband doa estimation using convolutional neural networks trained with noise signals," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct 2017, pp. 136–140.

[8] S. Adavanne, A. Politis, and T. Virtanen, "A multi-room reverberant dataset for sound event localization and detection," *Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, 2019. [Online]. Available: https://arxiv.org/abs/1905.08546

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