Towards Domain Independence in CNN-based Acoustic Localization using Deep Cross Correlations

Juan Manuel Vera-Diaz
Department of Electronics
University of Alcalá.
Alcala de Henares, Madrid, Spain
ORCID 0000-0002-6152-5789

Daniel Pizarro
Department of Electronics
University of Alcalá.
Alcala de Henares, Madrid, Spain
ORCID 0000-0003-0622-4884

Javier Macias-Guarasa
Department of Electronics
University of Alcalá.
Alcala de Henares, Madrid, Spain
ORCID 0000-0002-3303-3963

***Abstract*—Time delay estimation is essential in *Acoustic Source Localization* (ASL) systems. One of the most used techniques for this purpose is the *Generalized Cross Correlation* (GCC) between a pair of signals and its use in *Steered Response Power* (SRP) techniques, which estimate the acoustic power at a specific location. Nowadays, *Deep Learning* strategies may outperform these methods. However, they are generally dependent on the geometric and sensor configuration conditions that are available during the training phases, thus having limited generalization capabilities when facing new environments if no re-training nor adaptation is applied. In this work, we propose a method based on an encoder-decoder *CNN* architecture capable of outperforming the well known *SRP-PHAT* algorithm, and also other *Deep Learning* strategies when working in mismatched training-testing conditions without requiring a model re-training. Our proposal aims to estimate a smoothed version of the correlation signals, that is then used to generate a refined acoustic power map, which leads to better performance on the ASL task. Our experimental evaluation uses three publicly available realistic datasets and provides a comparison with the *SRP-PHAT* algorithm and other recent proposals based on *Deep Learning*.**

***Index Terms*—Acoustic Source Localization, Generalized Cross Correlation, Steered Response Power, Convolutional Neural Networks, Deep Learning**

I. Introduction

One of the critical tasks in *Acoustic Source Localization (ASL)* is the time delay estimation [1], [2] between signals recorded by a pair of acoustic sensors that are generated by an unknown acoustic source. With at least three of these pairs, it is possible to estimate the position of the acoustic source by using hyperbolic trilateration techniques. However, this process is not reliable in everyday scenarios with signals contaminated with noise and multipath effects. Other *ASL* methods are more robust to these effects such as those based on the *Steered Response Power (SRP)* [3]–[7] or the *Minimum Variance Distortionless Response (MVDR)* [8], [9]; all of them based on the Generalized Cross Correlation (GCC) [10], [11].

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness under projects HEIMDAL-UAH (TIN2016-75982-C2-1-R) and ARTEMISA (TIN2016-80939-R); and by the University of Alcalá under project ACÚFANO (CCG19/IA-024).

In recent years, *ASL* methods based on *Deep Learning* techniques have appeared in the literature. In [12], they use raw acoustic signals to estimate the source position coordinates directly, and in [13], they use the signal spectra to estimate the *Direction of Arrival (DoA)* of the acoustic source. Their results are promising, reporting better accuracy than classical methods. However, they have significant limitations: *1)* they require a large amount of labeled data for training, whereas there is limited availability of large and public datasets for *ASL*, and *2)* learning techniques are highly dependent on the room and sensor geometry, and the training conditions. As a consequence, their accuracy dramatically degrades when used in other environments or outside the physical area used to generate training examples.

In this paper, we propose a method based on a *Convolutional Neural Network (CNN)*. It takes the *GCC* of a pair of signals in its input and estimates a likelihood function where its maximum appears at the time-delay between the two signals. We then combine these likelihood functions in a 3D spatial grid, as proposed in classical *SRP* techniques. The target is similar to that described in [14], in which they propose a method to obtain the time-delay between two signals using the GCC and multilayer perceptrons with a single hidden layer. However, their system is only tested using artificial signals (chirps), so that it is not possible to assess its applicability in realistic scenarios.

Our contributions are the following: *1)* our method is largely independent of the room and sensor geometry, *2)* we can use small size datasets to train the neural network and *3)* it is consistently more accurate than classical methods such as *SRP-PHAT* and also better than other *Deep Learning* methods in general conditions, where the testing room is physically different, or the source position significantly differs from those available in the training data.

II. Problem Statement

Let us consider an environment where we place M microphones at known positions $\vec{m}_k = (m_{x_k}, m_{y_k}, m_{z_k})^\top$ with $k = 0, \ldots, M-1$. An acoustic source emits a signal $s(t)$

which is received and sampled by each microphone, obtaining a discrete-time signal $x_k[n]$ at the k^{th} microphone.

In free-field conditions the signals received at microphones k and l from a source at position $\vec{q} = (q_x, q_y, q_z)^\top$ only differ in a time delay $\Delta\tau(\vec{q}, \vec{m}_k, \vec{m}_l)$:

$$\Delta\tau(\vec{q}, \vec{m}_k, \vec{m}_l) = \frac{\parallel \vec{q} - \vec{m}_k \parallel - \parallel \vec{q} - \vec{m}_l \parallel}{c}, \quad (1)$$

where $\parallel . \parallel$ represents the Euclidean norm and c is the sound propagation velocity ($340\ m/s$ at $20\ ^\circ C$). The *GCC-PHAT* between the two signals is defined as:

$$gcc_{kl} = \mathcal{F}^{-1}\left(\frac{X_k[\omega] \cdot X_l[\omega]^*}{\mid X_k[\omega] \mid \cdot \mid X_l[\omega] \mid}\right), \quad (2)$$

where $X_k[\omega]$ is the *DFT* of signal $x_k[n]$, $\cdot$ is the element-wise product operator, $(\ )^*$ is the conjugate operator, $\mid \cdot \mid$ is the magnitude and $\mathcal{F}^{-1}$ denotes the inverse *DFT*. Under ideal propagation conditions, and not considering fractional delay nor windowing effects, the *GCC-PHAT* is a unit impulse shifted according to the time-delay between the signals. In a real scenario, the microphone signals are affected by the distortion and reverberation introduced by the environment, and the GCC-PHAT signals do not easily allow for the recovery of accurate time delays. A common approach to overcome this issue is to compute the so-called *Acoustic Power Map* (APM), evaluated on a grid of possible source positions $\vec{q}_0, \ldots, \vec{q}_K$, by using the *SRP-PHAT* beamformer:

$$APM(\vec{q}) = \sum_{k=0}^{M-1} \sum_{l=k+1}^{M-1} gcc_{kl}\left(\lfloor \Delta\tau(\vec{q}, \vec{m}_k, \vec{m}_l)\, f_s \rceil\right) \quad (3)$$

The source position that maximizes APM is an estimate of the true source position.

Our proposal is a encoder-decoder *CNN*, represented by the mathematical function f_{net}, that takes as input the *GCC-PHAT* between two signals and produce a Gaussian-like signal with variance σ^2 and mean equal to the time-delay shift (see Figure 1):

$$f_{net}(gcc(x_k[n], x_l[n])) = e^{\frac{-(D - \Delta\tau(\vec{q}, \vec{m}_k, \vec{m}_l)\, f_s)^2}{2\sigma^2}} \quad (4)$$

with $D = -L/2, \ldots, L/2$, and L being the maximum possible sample delay according to the microphone topology. We can define the *APM* based on our method by using equation (4) as an estimation of the gcc function in equation (3). Our method produces a smoother APM (see Figure 3), and yields better *ASL* performance than *SRP-PHAT*, as described in section V.

III. Proposed CNN Architecture

The proposed CNN model that implements equation (4), from now on `DeepGCC`, is shown in Figure 1. `DeepGCC` uses an encoder-decoder architecture with 36025 parameters. Both the encoder and the decoder use four blocks composed of a 1D convolutional layer with kernels of size 4, Max-Pooling (encoder) or upsampling (decoder) layers with size of two samples, batch-normalization, and ReLU activations. Table I summarizes the input and output sizes of every block.

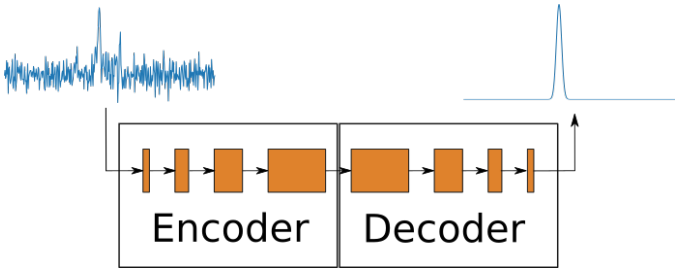

Fig. 1. `DeepGCC` layers scheme.

TABLE I
SUMMARY OF THE INPUT AND OUTPUT SIZES OF EACH NETWORK BLOCK.

Block	Input size	Output size
Encoder: Block 1	$L \times 1$	$L/2 \times 2$
Encoder: Block 2	$L/2 \times 2$	$L/4 \times 8$
Encoder: Block 3	$L/4 \times 8$	$L/8 \times 32$
Encoder: Block 4	$L/8 \times 32$	$L/16 \times 128$
Decoder: Block 1	$L/16 \times 128$	$L/8 \times 32$
Decoder: Block 2	$L/8 \times 32$	$L/4 \times 8$
Decoder: Block 3	$L/4 \times 8$	$L/2 \times 2$
Decoder: Block 4	$L/2 \times 2$	$L \times 1$

IV. Experimental Setup

A. Datasets

In order to evaluate to what extent the proposed method works properly under different acoustic and geometric conditions, we used three different datasets for the training and testing phases (refer to Figure 2 for graphical details):

- The *CAV3D* dataset [15] was recorded at 96kHz in a rectangular room of 4.77m×5.94m×4.50m and reverberation time of approximately $0.7s$, with a circular array of 8 microphones and 10cm radius, placed on top of a $73cm$ height table. All acoustic frames include the speaker's 3D mouth position coordinates. This dataset is composed of 10 single and moving speaker sequences, with varying user characteristics and moving patterns.
- The *AV16.3* dataset [16] was recorded in a 3.6m×8.2m×2.4m rectangular room that includes two circular arrays with the same geometry as the *CAV3D* array, also placed on top a 73cm height table. The room reverberation time is about 0.2s. This paper focuses on a subset of the dataset composed of 5 single speaker sequences recorded at 16kHz, comprising three static user and two moving user sequences. The recordings also include varying user characteristics and moving patterns. In our experimental work, we only use one circular array and upsample the signal to 96kHz, to provide the same array configuration and sampling rate as those of the *CAV3D* dataset.
- The *CHIL-CLEAR* dataset [17] contains sequences of "lecture seminars" with 3 to 8 participants (with no overlapped speech) recorded at five different rooms. In each room, eight sequences of 5 minutes were recorded. In this work, we focus only on the 8 *UPC* sequences, which

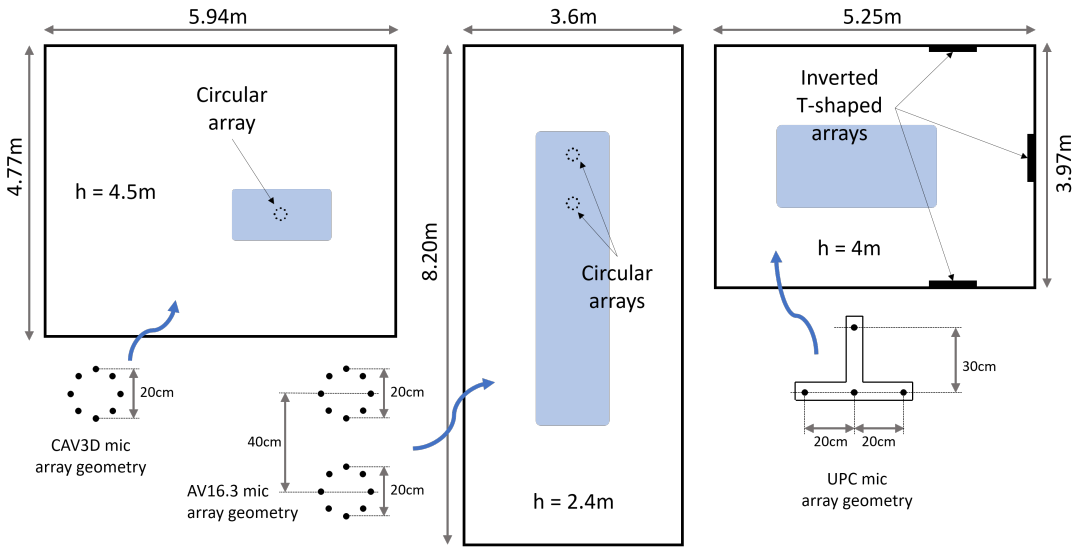

Fig. 2. Top view of the rooms used: CAV3D room (left), AV16.3 (center) and UPC (right).

were recorded in a $3.97\text{m}\times5.25\text{m}\times4.00\text{m}$ room, using three inverted T shaped 4-microphone arrays located on the walls at the height of 2.38m with a reverberation time of 0.6s. Again, we upsampled the microphone signals up to 96kHz.

Figure 2 shows the top view of the three rooms corresponding to the CAV3D (left), AV16.3 (center), and UPC (right), adequately scaled to provide the reader with a visual clue on the different geometrical and acoustical conditions for each database. It also shows the specific geometry of the microphone array configurations for each room.

TABLE II
CAV3D, *AV16.3*, *UPC* SEQUENCES USED TO TRAIN AND TEST OUR METHOD. THE 2^{nd}, 5^{th} AND 8^{th} COLUMNS INCLUDE THE NUMBER OF AVAILABLE ACOUSTIC FRAMES IN THE SEQUENCE, AND THE 3^{rd}, 6^{th} AND LAST COLUMNS REPRESENT THE CHARACTERISTICS OF EACH SEQUENCE.

CAV3D	#	Charact.	*AV16.3*	#	Charact.	*UPC*	#	Charact.
C06	393	M	A01	1328	$S+3$	U01	3869	$S+3$
C07	374	M	A02	1441	$S+3$	U02	4455	$S+3$
C08	412	M	A03	1552	$S+3$	U03	3794	$S+3$
C09	367	$M+1$	A11	297	$M+3$	U04	3780	$S+3$
C10	245	$M+1$	A15	284	$M+3$	U05	4348	$S+3$
C11	709	$M+1+2$	—	—	—	U06	3915	$S+3$
C12	725	$M+1+2$	—	—	—	U07	3135	$S+3$
C13	684	$M+3$	—	—	—	U08	3734	$S+3$
C20	422	M	—	—	—	—	—	—
C21	476	$M+2$	—	—	—	—	—	—

Table II summarizes the characteristics of the used sequences. We selected the sequences from *CAV3D* for training, validation, and testing, and the sequences from *AV16.3* and *UPC* for testing (see section IV-E for details). We code the characteristics of each sequence as follows: S refers to static speakers, M refers to moving speakers, 1 denotes noise present in the sequence, 2 denotes a speaker at two different heights, and 3 refers to the case of target positions not present in the rest of the sequences (thus not available in the training stages).

B. Dataset Processing

The process of generating the training, validation, and testing subsets is identical in all cases and consists of computing the *GCC-PHAT* for all the possible signals pairs, considering only those within each individual array. We extract each acoustic frame from the whole sequence using a 166ms signal frame with 50% overlapping and Blackman windowing. Assuming a 96kHz sampling rate, we use windows with 400 samples for the GCC signal, which implies that the network can process time-delays up to approximately $\pm$2ms. This delay is more than enough, given the maximum separation between microphones in the training data (20cm, as we only use the *CAV3D* dataset for training).

For each *GCC-PHAT* signal, we generate the supervised network output according to equation (4), computing the time-delay between the signals received by the two microphones and the labeled source position. This signal is also generated to be 400 samples in length and has a standard deviation of $\sigma = 5$ samples that we empirically selected in preliminary experiments.

C. Training Procedure

For the training phase, we followed the same procedure to evaluate all the methods. The loss function consists of the *Mean Squared Error (MSE)* between the network output and the target signal, generated with equation (4) using the *ground truth* source position. To minimize the loss, we used the *Adam* optimizer [18] with a *learning rate* of 10^{-4} and a *decay* of 10^{-8}, leaving the rest of the parameters at their default values. Batch size is equal to 100 samples, and we used validation data to stop training if the loss does not improve during 50 consecutive epochs. The *CAV3D* dataset was the only used for training, and due to the different features of each sequence, we run three different partitions, as shown in Table III, leaving the hardest sequences (C09, C10, C11, C12, and especially C13 (which contains speaker positions not available in the training

subset)) for validation and testing, and using the other simpler ones for training.

TABLE III
EVALUATED TRAINING/VALIDATION/TESTING PARTITIONS.

Partition #	Test	Val	Train
P1	C10, C12	C13	Rest of *CAV3D* sequences
P2	C09, C11	C13	Rest of *CAV3D* sequences
P3	C13	C11	Rest of *CAV3D* sequences

D. Algorithms Comparison

To compare the accuracy of our proposal, we evaluate it against three alternatives trained as explained in section IV-C. The first one is the well known SRP-PHAT algorithm, considered as the baseline system. The second method is the *Deep Learning* approach ASLNet [12], which estimates the cartesian coordinates of the acoustic source from the raw audio signals of a set of microphones. The last evaluated method is based on SELDnet [13], a recurrent neural network to estimate the azimuth and elevation of an acoustic source from the spectrogram of the audio signal of a microphone array. In our work, we have modified this architecture to estimate the source 3D cartesian coordinates directly, and we will refer to it as SELnetXYZ.

E. Experimental design

We carried out two different experiments to focus on two performance indicators: ASL precision performance, and environmental robustness:

- For testing the ASL precision performance, we used the C09, C10, C11, and C12 sequences because of the similarity of its labeled positions with the training ones. In this experiment, we expect better performance for the standard *Deep Learning* approaches ASLNet and SELnetXYZ. We also expect that our proposal will achieve better performance than the SRP-PHAT beamformer. We evaluated the ASL performance as the mean Euclidean distance between the labeled *ground truth* and the estimated position, provided the training and testing positions are similar.
- For testing the environmental robustness of the different proposals, we measured the localization accuracy when the room geometry, microphone array geometry, and the evaluated positions in the test data significantly differ from those in the training data. *IDIAP* and *UPC* sequences were used for this purpose, along with C13, which belong to the *CAV3D* dataset, but includes a full range of positions not found in the training subsets. In this case, we expect our DeepGCC proposal to roughly keep the same performance as in the first experiment, while the other *Deep Learning* methods exhibit a performance decrease due to mismatched evaluation conditions. Note that in the *UPC* sequences, only the SRP-PHAT and DeepGCC algorithms can be evaluated since the microphone array geometry has changed, and we will not retrain ASLNet or SELnetXYZ.

In all the experiments, we have used all the possible microphone pair combinations to build a volumetric acoustic power map with a 10cm×10cm×10cm grid resolution, generated from the DeepGCC model output. We then extract the acoustic source cartesian coordinates from the location of the maximum acoustic power.

V. RESULTS

Figure 3 (left) shows a particular example of the output obtained with DeepGCC and *GCC-PHAT* (SRP-PHAT). Figure 3 (right) shows the APMs using equation (3), which involves all microphone pairs. The map built with DeepGCC is considerably smoother than that built with SRP-PHAT.

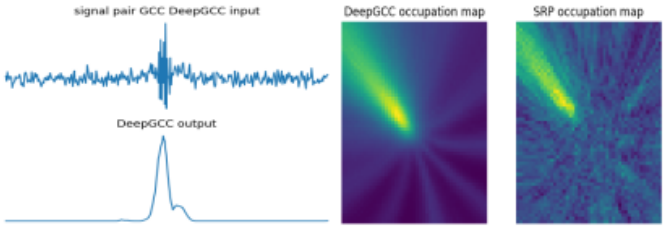

Fig. 3. Example of a *GCC-PHAT* input and the DeepGCC result and their respective Acoustic Power Maps.

Table IV shows the results of the first experiment (focused on ASL performance). The average MSE is shown in cm, and we also include the relative improvement achieved by each method as compared with the SRP-PHAT algorithm ($\Delta_r^{MSE} = \frac{MSE_{\text{SRP-PHAT}} - MSE_{algorithm}}{MSE_{\text{SRP-PHAT}}}$). SELnetXYZ obtains, as expected, the best results because testing and training positions are similar. Our DeepGCC proposal is the second-best strategy, getting better results than SRP-PHAT in all cases, and better than ASLNet in 3 out of 4 sequences.

TABLE IV
RESULTS FOR THE ASL PRECISION EXPERIMENTS (ERROR IN CM, Δ_r^{MSE} IS RELATIVE MSE IMPROVEMENT OVER THE SRP-PHAT ALGORITHM).

Sequence	SRP-PHAT	DeepGCC	ASLNet	SELnetXYZ
C10	101.3	94.5	92.5	**55.2**
Δ_r^{MSE}		6.71%	8.69%	45.51%
C12	114.5	101.5	103.5	**73.4**
Δ_r^{MSE}		11.35%	9.61%	35.89%
C09	94.9	82.1	87.6	**62.0**
Δ_r^{MSE}		13.49%	7.69%	34.67%
C11	117.5	106.4	133.3	**94.6**
Δ_r^{MSE}		9.45%	3.57%	19.49%

Table V shows the results of the second experiment (focused on environmental robustness), in which our DeepGCC proposal clearly outperforms all the other methods. The fact that we aim to estimate the GCC function (that mainly depends on the relative time delay) makes it more robust to changes in the room and array geometry or the source positions with respect to the microphone arrays. The other *deep learning* methods are not able to properly face the mismatched conditions, performing far worse than the standard SRP-PHAT. We remark the fact that the DeepGCC proposal has a 13.82% relative

improvement over the SRP-PHAT algorithm even though the microphone geometry (*UPC* sequences) changes. This is a difficult task for a *Deep Learning* method since it has to deal with unseen conditions in the *GCC-PHAT* signal.

TABLE V
RESULTS FOR THE ENVIRONMENTAL ROBUSTNESS EXPERIMENTS (ERROR IN CM, Δ_r^{MSE} IS RELATIVE MSE IMPROVEMENT OVER THE SRP-PHAT ALGORITHM).

Sequence	SRP-PHAT	DeepGCC	ASLNet	SELnetXYZ
C13	86.9	**79.5**	136.7	135.5
Δ_r^{MSE}		8.51%	-57.31%	-55.53%
A01	103.04	**84.4**	348.3	347.8
Δ_r^{MSE}		18.09%	−238.02%	−237.54%
A02	68.8	**64.6**	350.4	363.2
Δ_r^{MSE}		6.10%	−409.30%	−427.91%
A03	73.7	**57.7**	352.3	360.4
Δ_r^{MSE}		21.71%	−378.02%	389.01%
A11	84.1	**69.1**	274.3	284.1
Δ_r^{MSE}		17.84%	−226.16%	−237.81%
A15	140.3	**110.3**	387.1	386.4
Δ_r^{MSE}		21.38%	−175.91%	−175.41%
U01	96.8	**77.8**	—	—
Δ_r^{MSE}		19.63%		
U02	111.6	**91.9**	—	—
Δ_r^{MSE}		17.65%		
U03	128.8	**107.5**	—	—
Δ_r^{MSE}		16.54%		
U04	136.2	**115.3**	—	—
Δ_r^{MSE}		15.35%		
U05	130.0	**122.1**	—	—
Δ_r^{MSE}		6.08%		
U06	130.2	**113.6**	—	—
Δ_r^{MSE}		12.75%		
U07	103.2	**83.8**	—	—
Δ_r^{MSE}		18.80%		
U08	133.6	**123.1**	—	—
Δ_r^{MSE}		7.86%		

VI. CONCLUSIONS

In this paper, we have described DeepGCC, a method based on deep learning that transforms the *GCC-PHAT* of two signals, emitted from the same source and received in two microphones, into a Gaussian function whose maximum appears at the time difference between the two signals. We use DeepGCC to estimate a smoother and more accurate acoustic power map, as compared to that generated by the standard SRP-PHAT method. We obtain the acoustic source position finding the position where this map is maximum. In our experiments, DeepGCC yields more accurate localization than *GCC-PHAT* in all cases. We also compare DeepGCC with existing deep learning methods that estimate the source position from the microphone signals directly. Our approach is consistently more accurate than these methods when testing and training conditions vary significantly, which makes it better suited for deployment in real scenarios.

As future work, our method will be combined with sparse denoising [19] to improve localization accuracy based on acoustic maps. We also plan to extend DeepGCC to multiple speaker localization tasks.

REFERENCES

[1] Michael S. Brandstein and Harvey F. Silverman, "A practical methodology for speech source localization with microphone arrays," *Computer Speech & Language*, vol. 11, no. 2, pp. 91–126, 1997.

[2] Yiteng (Arden) Huang, Jacob Benesty, and Jingdong Chen, "Time delay estimation and source localization," in *Springer Handbook of Speech Processing*, Jacob Benesty, M. Mohan Sondhi, and Yiteng (Arden) Huang, Eds., pp. 1043–1063. Springer Berlin Heidelberg, 2008, 10.1007/978-3-540-49127-9_51.

[3] J.H. DiBiase, *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays*, Ph.D. thesis, Brown University, 2000.

[4] J. DiBiase, H. Silverman, and M. Brandstein, "Robust localization in reverberant rooms," *Microphone Arrays*, pp. 157–180, 2001.

[5] J.P. Dmochowski, J. Benesty, and S. Affes, "A generalized steered response power method for computationally viable source localization," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 8, pp. 2510 –2526, nov. 2007.

[6] Xinwang Wan and Zhenyang Wu, "Improved steered response power method for sound source localization based on principal eigenvector," *Applied Acoustics*, vol. 71, no. 12, pp. 1126 – 1131, 2010.

[7] Hoang Do and H.F. Silverman, "SRP-PHAT methods of locating simultaneous multiple talkers using a frame of microphone array data," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, march 2010, pp. 125 –128.

[8] Emanuël A. P. Habets, Jacob Benesty, Sharon Gannot, and Israel Cohen, "The MVDR beamformer for speech enhancement," in *Speech Processing in Modern Communication*, Israel Cohen, Jacob Benesty, and Sharon Gannot, Eds., vol. 3 of *Springer Topics in Signal Processing*, pp. 225–254. Springer Berlin Heidelberg, 2010, 10.1007/978-3-642-11130-3_9.

[9] D. Salvati, C. Drioli, and G. L. Foresti, "On the use of machine learning in microphone array beamforming for far-field sound source localization," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept 2016, pp. 1–6.

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

[11] M. Omologo and P. Svaizer, "Use of the cross-power-spectrum phase in acoustic event location," *IEEE Trans. on Speech and Audio Processing*, vol. 5, pp. 288–292, 1993.

[12] Juan Manuel Vera-Diaz, Daniel Pizarro, and Javier Macias-Guarasa, "Towards end-to-end acoustic localization using deep learning: From audio signals to source position coordinates," *Sensors*, vol. 18, no. 10, 2018.

[13] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," *CoRR*, vol. abs/1807.00129, 2018.

[14] Ludwig Houegnigan, Pooyan Safari, Climent Nadeu, Mike Schaar, Marta Solé, and Michel André, "Neural networks for high performance time-delay estimation and acoustic source localization," in *Proceedings of the Second International Conference on Computer Science, Information Technology and Applications*, 01 2017, pp. 137–146.

[15] X. Qian, A. Brutti, O. Lanz, M. Omologo, and A. Cavallaro, "Multi-speaker tracking from an audio–visual sensing device," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2576–2588, Oct 2019.

[16] Guillaume Lathoud, Jean-Marc Odobez, and Daniel Gatica-Perez, "AV16.3: An audio-visual corpus for speaker localization and tracking," in *Procceedings of the MLMI*, Samy Bengio and Hervé Bourlard, Eds. 2004, vol. 3361 of *Lecture Notes in Computer Science*, pp. 182–195, Springer-Verlag.

[17] Rainer Stiefelhagen, Keni Bernardin, Rachel Bowers, R Travis Rose, Martial Michel, and John Garofolo, "The CLEAR 2007 evaluation," in *Multimodal Technologies for Perception of Humans*, pp. 3–34. Springer, 2007.

[18] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] J. Velasco, D. Pizarro, J. Macias-Guarasa, and A. Asaei, "TDOA matrices: Algebraic properties and their application to robust denoising with missing data," *IEEE Transactions on Signal Processing*, vol. 64, no. 20, pp. 5242–5254, Oct 2016.