Exploiting Temporal Context in CNN Based Multisource DOA Estimation

Alexander Bohlender, Ann Spriet, *Senior Member, IEEE*, Wouter Tirry, and Nilesh Madhu

***Abstract*—Supervised learning methods are a powerful tool for direction of arrival (DOA) estimation because they can cope with adverse conditions where simplified models fail. In this work, we consider a previously proposed convolutional neural network (CNN) approach that estimates the DOAs for multiple sources from the phase spectra of the microphones. For speech, specifically, the approach was shown to work well even when trained entirely on synthetically generated data. However, as each frame is processed separately, temporal context cannot be taken into account. This prevents the exploitation of interframe signal correlations, and the fact that DOAs do not change arbitrarily over time. We therefore consider two different extensions of the CNN: the integration of a long short-term memory (LSTM) layer, or of a temporal convolutional network (TCN). In order to accommodate the incorporation of temporal context, the training data generation framework needs to be adjusted. To obtain an easily parameterizable model, we propose to employ Markov chains to realize a gradual evolution of the source activity at different times, frequencies, and directions, throughout a training sequence. A thorough evaluation demonstrates that the proposed configuration for generating training data is suitable for the tasks of single-, and multi-talker localization. In particular, we note that with temporal context, it is important to use speech, or realistic signals in general, for the sources. Experiments with recorded impulse responses and noise reveal that the CNN with the LSTM extension outperforms all other considered approaches, including the plain CNN, and the TCN extension.**

***Index Terms*—Convolutional neural networks, direction-of-arrival, temporal context, training data generation.**

I. Introduction

THE locations of sound sources, which can be described by means of the DOAs of the sound captured by an array of microphones, are essential parameters for a variety of applications where spatial filtering techniques can be exploited, e. g., hands-free communication, teleconferencing, hearing aids as well as, more recently, voice-controlled smart devices, and hearables. As a result, DOA estimation has been a topic of interest for a long time, and a wide variety of approaches have been proposed. An overview can be found in [1].

Manuscript received September 26, 2020; revised January 13, 2021; accepted March 10, 2021. Date of publication March 18, 2021; date of current version May 7, 2021. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Huseyin Hacihabiboglu. *(Corresponding author: Alexander Bohlender.)*

Alexander Bohlender and Nilesh Madhu are with IDLab, Department of Electronics and Information Systems, Ghent University – imec, Ghent 9000, Belgium (e-mail: alexander.bohlender@ugent.be; nilesh.madhu@ugent.be).

Ann Spriet and Wouter Tirry are with the Goodix Technology (Belgium) B.V., Leuven 3000, Belgium (e-mail: aspriet@goodix.com; wtirry@goodix.com).

Digital Object Identifier 10.1109/TASLP.2021.3067113

Speech signals, the primary focus of this work, are known to exhibit a high degree of sparsity over time and frequency [2]. This being the case, an integration over frequency, as employed by methods based on generalized cross-correlation (GCC) [3], can be undesirable. In order to preserve the information from each individual frequency bin, narrowband DOA estimates can be acquired first. One popular method for which narrowband realizations, such as [4], are available is steered response power with phase transform (SRP-PHAT) [5]. It relies on the maximization of the output power of a beamformer over all possible directions. Various other narrowband methods use the decomposition into orthogonal speech and noise subspaces, e. g., MUSIC [6] and ESPRIT [7], the matching of interchannel phase differences [8]–[10], or the the sparsity of the true DOAs with respect to all possible directions [11]–[13].

Model based methods like these are derived based on strongly simplifying assumptions. In particular, disregarding the presence of acoustic reflections can lead to a poor performance when the direct path propagation does not dominate, i. e., for strong reverberation, or large source-array distances. Although attempts have been made to account for the presence of reverberation, e. g., [14], [15], it remains a challenge to find statistical models that are generic but not too complex, especially for the localization of multiple sources with an arbitrary array geometry. For this reason, it has become increasingly popular in recent years to employ supervised learning methods to address the problem of DOA estimation under adverse conditions. One common approach adopted by, e. g., [16], [17], is to utilize deep neural networks (DNNs) for the estimation of time-frequency masks, which can then enhance the robustness of classical methods like SRP-PHAT. Approaches like [18], in contrast, use classical methods for the feature extraction, followed by robust DOA estimation with DNN methods based on the provided features. Finally, it is also possible to employ learning methods both for extracting robust input features, and for the DOA estimation, as is done in [19], and [20].

Whereas the above approaches all make use of classical DOA estimation methods in some way, the CNN approach that was originally proposed for single-source DOA estimation in [21], and extended to multisource DOA estimation in [22], produces a vector of DOA probabilities directly from the unprocessed phase spectrum of the microphone signals. This allows the network to design a suitable feature representation for the DOA estimation task on its own. Moreover, for this CNN, the authors demonstrate that it is possible to generate suitable training data synthetically, requiring only that room impulse responses (RIRs) are simulated

in advance. Even for the source signals, it is possible to resort to white noise instead of realistic signals. In this case, the network can still learn to exploit sparsity in the time-frequency domain when this is explicitly incorporated into the training data generation process.

However, when each short-time Fourier transform (STFT) frame is processed individually, temporal characteristics of the source signals cannot fully be exploited, even if realistic signals are used for training. Moreover, in practice, the true DOAs do not change significantly between two consecutive frames most of the time, whereas occasionally, e. g., in a conference situation, a different speaker becomes active, or a previously active speaker stops talking, which results in a sudden change of the DOA. Therefore, as in [23]–[25], the combination of a CNN with, e. g., LSTM, or gated recurrent unit (GRU) can be considered, thereby giving the DOA estimator the ability to exploit temporal context. However, this complicates the generation of training data, as the evolution of the DOAs over time must be addressed as well. When considering only static sources, or sources that move in a predefined way, depending on the employed approach, a good generalization to different scenarios may not be ensured.

For the dataset used in the sound event localization and detection (SELD) task of the DCASE 2020 challenge [26], for example, the generated mixtures consist of a series of (overlapping) finite-length sound events. For each static event, a fixed DOA, and time of onset is selected at random. Additionally, moving sound events are simulated by means of a time-variant convolution, where the individual RIRs for each instant are obtained by interpolating between the closest grid points for which a RIR is available. The direction as well as the rate of motion are used to introduce more variability to the data.

In [27], the authors of [22] repurpose their CNN DOA estimator for the task of estimating a time-frequency mask that is then used to extract a single target speaker from the microphone signals. The authors deviate from the network architecture they used for DOA estimation by including an LSTM, or a bidirectional LSTM (BLSTM) layer in the network, but choose to limit the temporal context by using short sequences of frames in the immediate vicinity of the frame under consideration. Therefore, DOA changes in the training data are not a requirement.

In this work, we extend the multi-speaker CNN DOA estimation approach [22], including network architecture, and training data generation, such that long-term temporal context can be taken into account. The key contributions are: (i) a procedure for generating training data for a *dynamic* setting, i. e., where sources can be active at different locations at different times, as well as, (ii) based on experimental results, an analysis of how temporal context can best be exploited, and what benefit this gives.

For the evolution of the locations, and the activity of the sources throughout a sequence, we propose to use a Markov model. Through the probability for a transition between the model's states, we control how rapidly the acoustic scene changes. By not restricting the duration for which a source is active at a specific location, and letting the source positions change randomly throughout a sequence, we aim to guarantee a good generalization across a broad range of situations that may be encountered in practice.

We evaluate the benefit of incorporating temporal context based on two possible extensions of the considered CNN architecture, the one being based on the insertion of an LSTM layer, the other using a TCN [28], [29]. As part of the experimental analysis, we also study how the training set should be composed in terms of the number of active sources, and the additive noise. Furthermore, although it is possible to use noise for the source signals during training, as is done in [22], we show that for the LSTM and TCN extensions, it is beneficial to use realistic source signals, in this case speech, instead.

The remainder of the paper is structured as follows: Section II formulates DOA estimation as a classification problem, which is the viewpoint of the CNN DOA estimator that is presented in Section III. Subsequently, the LSTM and TCN extensions that will permit the incorporation of temporal context are introduced in Section IV. The resulting necessity of adjusting the training data generation procedure is addressed in Section V. Finally, the experimental validation is conducted in Section VI, followed by the conclusions.

II. DOA Estimation as a Classification Problem

We consider the problem of localizing J sound sources from the signal captured by an array of N microphones. The unknown locations of the sources with respect to the microphone array are described by the corresponding DOAs. As the estimation of the DOA of either of the sources can, if the array geometry allows it, encompass the estimation of the azimuth angle φ as well as the elevation angle ϑ, the generic notion of the DOA ϕ will be used in the following. Assuming a discrete grid for the source locations, the DOAs are represented by $\phi \in \{\phi_1, \ldots, \phi_I\}$, where I denotes the number of candidate DOAs. The problem can therefore be seen as a classification problem, where the one "DOA class" that corresponds to the true source location must be identified. For the case where multiple sources are active at the same time, more than one of the I classes should be selected. Ideally, in the vector of *probabilities of source activity* for the candidate directions returned by the classifier, a small number of entries should then be 1, whereas all other entries are 0 ("multi-hot").

We will consider the M-point STFT representation of the signals, where the frequency index is denoted by μ, and the frame index by λ. The microphone signal is composed of three parts: the direct path contribution $S^{\mathrm{dir}}_{j,n}(\mu,\lambda)$, and the corresponding reverberation components $S^{\mathrm{rev}}_{j,n}(\mu,\lambda)$, for the transmission from the j-th source to the n-th microphone, as well as an additive noise $V_n(\mu,\lambda)$. Consequently, the microphone signal is given by

$$\begin{aligned} Y_n(\mu,\lambda) &= \sum_j \left(S^{\mathrm{dir}}_{j,n}(\mu,\lambda) + S^{\mathrm{rev}}_{j,n}(\mu,\lambda)\right) + V_n(\mu,\lambda), \\ &= \sum_j S^{\mathrm{mic}}_{j,n}(\mu,\lambda) + V_n(\mu,\lambda). \end{aligned} \tag{1}$$

Further, $S^{\mathrm{dir}}_{j,n}(\mu,\lambda)$ can be modeled in terms of the convolution of a dry source signal with an impulse response that

describes the direct path propagation. Because it imposes only a time delay, and an attenuation factor, the frequency domain counterpart of this impulse response is a complex exponential. Whereas reverberation, and noise act as unwanted interference, $S_{j,n}^{\mathrm{dir}}(\mu,\lambda)$ carries the DOA information. This implies that, for the identification of the direct path, the *phases* $\angle Y_n(\mu,\lambda)$ of the magnitude-phase representation

$$Y_n(\mu,\lambda) = |Y_n(\mu,\lambda)|\, \mathrm{e}^{\jmath \angle Y_n(\mu,\lambda)} \tag{2}$$

of the microphone signals are of particular interest. The magnitude component, on the other hand, contains little information about the DOA when free field propagation is assumed, as the attenuation factors are very similar for all microphones when the source is located in the far-field of a compact array.

III. Single-Frame CNN DOA Estimation

This section summarizes the CNN approach for multisource DOA estimation proposed in [22].

A. Input Representation

Instead of manually assembling a set of potentially useful features from the microphone signals, it is left to the convolutional layers of the CNN to extract features that are optimally suited for the DOA classification. However, rather than using the microphone signals directly as the input of the CNN, only the phase component $\angle Y_n(\mu,\lambda)$ of (2) is retained. This is because the phase contains the information about the delays induced by the propagation from the source position to the microphone array and, thus, the DOA.

With these considerations, the phases of all N microphones at the frequencies up to the Nyquist frequency, i. e., $\mu = 0,\ldots,M'-1$ with $M' = M/2+1$, are used as input. This $N \times M'$ representation $\mathbf{\Phi}_\lambda$ of the microphone signals $Y_n(\mu,\lambda)$ is termed the phase map.

B. Architecture

The CNN architecture of [22] is illustrated in Fig. 1. Convolutions are applied across the channel dimension only. In [21], where the single-source case is considered, convolutions are also applied across frequency. For the localization of multiple (speech) sources, however, it is beneficial to take advantage of the commonly made assumption of W-disjoint orthogonality [2], i. e., there is only one dominant source in each time-frequency bin. Therefore, an improved robustness can be expected when performing a separate feature extraction for each frequency bin. The convolutions across the channel dimension are applied in the form of small filters with length 2, without zero-padding or pooling between the layers. To ensure that correlations between all combinations of two channels can be taken into account, the number of convolution layers is set to the resulting maximum of $N-1$.

After the feature extraction, two fully connected (FC) layers are used to aggregate the information from all frequency bins. The output layer with sigmoid activation comprises one node for each of the I DOA classes.

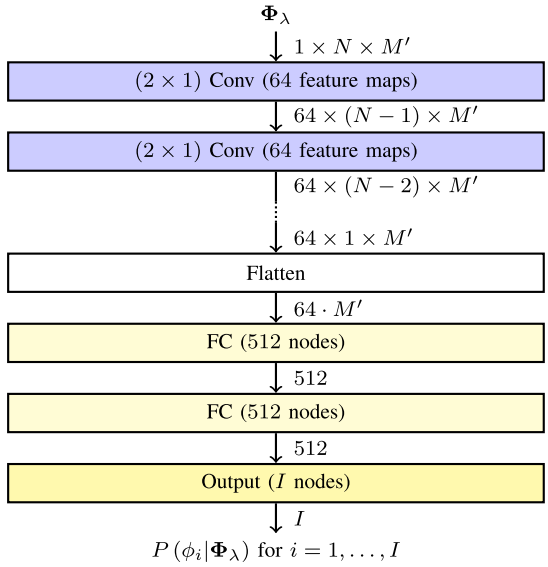


Fig. 1. CNN for multisource DOA estimation proposed in [22]. The phase spectrogram of the microphones signals is used as input, the output vector represents the probability of source activity for each of the I DOA classes.

C. Training

A large labeled dataset is required for training the CNN. By taking advantage of the signal model (1), there is no need to record and label a large number of microphone signals manually. Instead, dry source signals are convolved with multichannel RIRs and, subsequently, added up and mixed with an additive noise. The noise employed for this purpose is spatially and temporally uncorrelated, the mixing signal-to-noise ratio (SNR) is selected at random. The required RIRs can be simulated beforehand, e. g., using [30]. This approach is preferred, because it is hardly feasible in practice to record RIRs for a vast range of different acoustic conditions for just one array geometry. To improve the variability of the training data for a fixed set of simulated RIRs, different combinations of source DOAs, given a certain number of active sources, and acoustic conditions, i. e., room dimensions and reverberation time, can be considered.

To get the best performance for the localization of speech sources, the obvious choice would be to make use of a speech database for the dry source signals. However, it is pointed out in [21], [22] that (relatively) silent segments, which are inevitably present in any speech signal, make it difficult to set meaningful ground truth labels. When a low speech level in the training labels is not accounted for, the network will be trained to always look for a source in *some* direction, even when the additive noise is dominant. This can lead to unexpected results. The described problem does not apply to white noise, which can be used for the source signals instead of speech. Doing so is reasonable in this case because only the phase component is used, and the network is designed such that correlations of the signal across time and frequency are not taken into account in the convolutional part of the network anyway. Additionally, this removes the need for a source signal database altogether, given that noise can be generated easily.

For the case where multiple sources are active at the same time, care must be taken as to how the mixing of the contributions of these sources to the microphone signals is performed. For speech, a realistic mixture is obtained by simply adding up the convolved signals. However, this is not desirable when white noise source signals are used instead, because this would make it impossible to take advantage of the W-disjoint orthogonality property inherent to speech. Therefore, [22] proposes to enforce the disjointness in the training data by selecting exactly one dominant source at random. This random selection must be consistent for all channels, but may be different for each frequency.

Since, as outlined in Section II, we consider DOA estimation as a classification problem, only the target output vector entries that correspond to one of the active DOAs are set to 1, all other entries are 0. The binary cross-entropy (BCE) loss is used to optimize the weights.

To train the CNN, an Adam optimizer [31] is used with learning rate 0.001 and mini-batches of 512 frames. Dropout [32] with rate 0.5 is applied before each of the FC layers. Additionally, we make use of batch normalization [33]. All hidden layers use the ReLU [34] activation function.

D. DOA Extraction

Once trained, the output of the CNN may be interpreted as the posterior probabilities $P(\phi_i|\mathbf{\Phi}_\lambda)$ for each of the DOA classes. A decision regarding the DOAs for a block of L frames can be made by taking the J highest peaks of the averaged probabilities

$$P_\lambda(\phi_i) = \frac{1}{L} \sum_{\lambda'=\lambda-L+1}^{\lambda} P(\phi_i|\mathbf{\Phi}_{\lambda'}). \tag{3}$$

In the following, we will instead use recursive averaging to determine the probabilities

$$P_\lambda(\phi_i) = \alpha P_{\lambda-1}(\phi_i) + (1-\alpha) P_\lambda(\phi_i|\mathbf{\Phi}_\lambda), \tag{4}$$

where the averaging parameter is set to $\alpha = \mathrm{e}^{-1/L}$.

If not known, it is also possible to determine an estimate $\widehat{J}$ of the number of active sources J directly from the averaged probabilities.

IV. DOA Estimation on Frame Sequences

The described CNN architecture performs the DOA estimation independently for each frame by exploiting interchannel phase correlations across the entire frequency range. Although not modeling temporal changes simplifies the generation of useful training data, the inability to take temporal context into account is a considerable limitation. Whereas a long averaging duration helps at least with the localization of static sources, this simple post-processing prevents the quick detection of the activity of a new source, as well as the inactivity of a previously active source. In short, the CNN approach cannot account for: (i) the temporal evolution of the source DOAs, and (ii) spectro-temporal characteristics of the source signals.

We will consider two possible extensions of the CNN architecture that can address these shortcomings. For both, the feature extraction realized by the convolutional layers is preserved as it is. In Section IV-A, one of the FC layers is replaced by an LSTM layer, whereas in Section IV-B, a TCN is used for taking temporal context into account.

TABLE I
NUMBER OF TRAINABLE PARAMETERS FOR DIFFERENT CONFIGURATIONS OF THE TWO LAYERS FOLLOWING THE CONVOLUTIONAL LAYERS

	option	A	B	B′	C
first layer	type	FC	LSTM	LSTM	FC
	input size ($M'=257$)	16 448	16 448	16 448	16 448
	output size	512	512	128	512
	parameters $\times 10^6$	8.4	34.7	8.5	8.4
second layer	type	FC	FC	FC	LSTM
	input size	512	512	128	512
	output size	512	512	512	512
	parameters $\times 10^6$	0.3	0.3	0.1	2.1
Σ parameters $\times 10^6$		8.7	35.0	8.6	10.5
parameters relative to option A		1.00	4.03	0.98	1.21

A. CNN With LSTM Extension

A straightforward approach for enabling the network to take temporal context into account, without fundamentally changing the network structure, is to replace either the first, or the second FC layer by an LSTM layer. To decide which of these options is preferred, the effect on the complexity of the model in terms of the number of parameters and the computational effort requirements should be taken into account along with the resulting performance. For four different configurations, the number of parameters that need to be trained, for these two layers only, are shown in Table I. In the original CNN (option A in the table), the *first* FC layer contains by far the highest number of parameters due to the large number of input features ($64\,M' = 16\,448$ for $M' = 257$) provided by the convolutional layers. Therefore, replacing this layer by an LSTM (option B) would result in a significant increase of the number of parameters (303% more than the pure CNN), and thus the overall complexity of the network. To compensate for this, the output size of the LSTM could be reduced (option B′). This strategy is adopted for the time-frequency mask estimator in [27], as well as for the single-speaker DOA estimator based on GCC features proposed in [24]. The focus of this paper is on incorporating temporal context within the architecture depicted in Fig. 1. Since a reduction of the number of output features of either of the layers would equal a further deviation from this reference, we will not consider this option here.

Alternatively, the *second* FC layer could be replaced by an LSTM layer (option C). Given the significantly smaller number of input features, this only results in a moderate increase of the complexity (21% in terms of the number of parameters). Empirically, we find that neither of the configurations where one of the FC layers is replaced by an LSTM (options B and C) clearly outperforms the other. Therefore, to prevent a significant increase of the complexity without requiring the number of hidden features to be reduced, we select the second FC layer

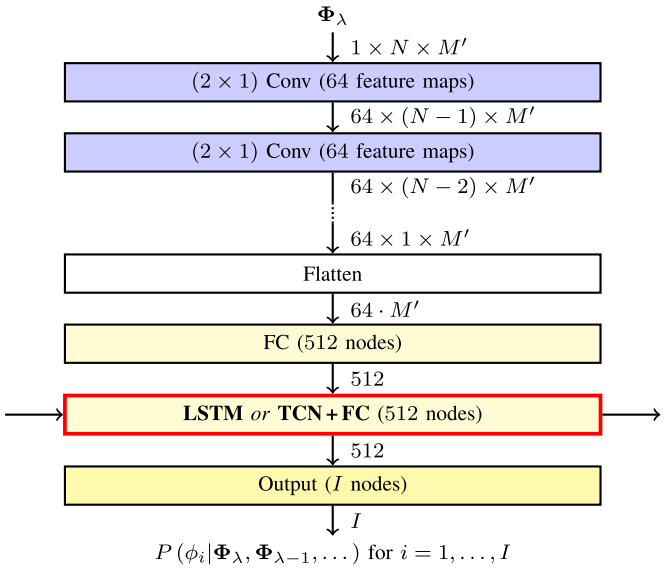


Fig. 2. An LSTM replaces the second FC layer (Section IV-A), or a TCN is added between the FC layers (Section IV-B). The arrows from the left and to the right indicate how information is passed on from one frame to the next.

for being replaced by an LSTM. The architecture of the resulting approach, which will be referred to as CNN/LSTM in the following, is shown in Fig. 2.

An important consequence of the modification of the architecture is that frame *sequences*, rather than individual (independent) frames, are now required as training data. For making it possible to exploit long-term temporal context, the LSTM must first learn *how* information from the past should be utilized. A plausible variability of the DOAs in the training data is therefore essential. If only static scenarios were considered during training, i. e., a fixed number of sources at fixed locations, it would not be possible for the CNN/LSTM approach to behave as intended when confronted with a realistic situation. On the other hand, for allowing the LSTM to take advantage of the consistency of the source positions in consecutive frames, the DOAs should not change too quickly either. In Section V-A, a model is proposed that introduces a parameter for controlling this trade-off.

B. CNN With TCN Extension

In contrast to LSTM, a temporal convolutional network (TCN) [28], [29] introduces temporal context by performing convolutions across time independently for each input feature. Even so, information from a relatively large (but defined) number of past frames can be taken into account, i. e., the TCN can be designed such that the *receptive field* r is sufficiently large. To achieve this without requiring long filters, a TCN is a concatenation of $\mathcal{L}$ convolutional layers with exponentially increasing dilations $2^0, 2^1, \dots, 2^{\mathcal{L}-1}$. This is illustrated in Fig. 3 for $\mathcal{L} = 4$, and a fixed filter length $K = 3$. Through sufficient zero-padding, it is ensured that the dimensionality does not change. The resulting receptive field of the entire TCN is

$$r = 1 + (K-1)\sum_{l=0}^{\mathcal{L}-1} 2^l = 1 + \left(K-1\right)\left(2^{\mathcal{L}}-1\right). \quad (5)$$

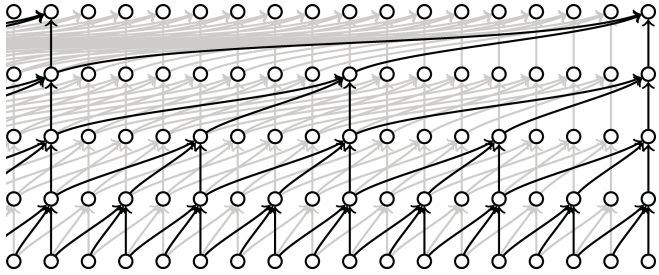


Fig. 3. Temporal Convolutional Network (TCN) comprising $\mathcal{L} = 4$ convolution layers (from bottom to top) with exponentially increasing dilations $2^0, 2^1, \dots, 2^{\mathcal{L}-1}$, all filters have length $K = 3$. A more detailed description of how dilated convolutions are useful for capturing long-term temporal dependencies can be found in [28], [29].

For $\mathcal{L} = 4$, and $K = 3$, this amounts to a receptive field of 31 frames, i. e., each output feature is a function of the corresponding input feature for the current frame, and the 30 directly preceding frames.

The insertion of a TCN instead of an LSTM is a promising alternative in case it is preferred to base the DOA estimation on a *defined* number of past frames. Without adding to the complexity, sequences of arbitrary length can still be passed through the network one frame at a time. The extension resulting from the addition of a TCN between the two FC layers will be referred to as CNN/TCN. The resulting architecture is also shown in Fig. 2.

Besides the easily adjustable receptive field, another advantage over CNN/LSTM is that the TCN allows for a higher degree of parallelization. Whereas the LSTM requires the previous frame to be fully processed first, the computations needed for one element in any one layer of the TCN only rely on the availability of K elements of the preceding TCN layer. This is helpful especially for training, and offline applications, but can to a limited extent also be exploited in realtime applications, when a few time frames are processed in parallel.

Due to the finite memory of the CNN/TCN approach, it is less critical how the evolution of the source DOAs over time is modeled in the training data. However, given the dedicated forgetting functionality of an LSTM, the CNN/LSTM architecture may be better suited for detecting sudden changes of the speaker position rapidly.

V. Training Data Generation

Both approaches, CNN/LSTM, and CNN/TCN, make it possible to take into account temporal context in a wide, or even in an unrestricted sense. Therefore, a meaningful model to describe how the considered scenario evolves over time is strictly required. This will be addressed in Section V-A. Although in [21], [22], it was deliberately chosen *not* to take into account source characteristics, the exploitation of temporal context makes it worthwhile to reevaluate this aspect. Therefore, the possibility of using speech source signals for training is discussed in Section V-B.

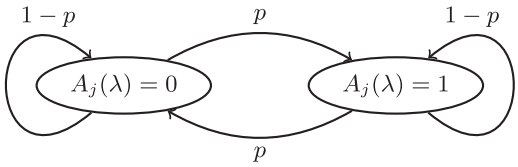


Fig. 4. The Markov chain with parameter p is used to model the activity of the source with index j: $A_j(\lambda) = 1$ represents activity whereas $A_j(\lambda) = 0$ represents inactivity.

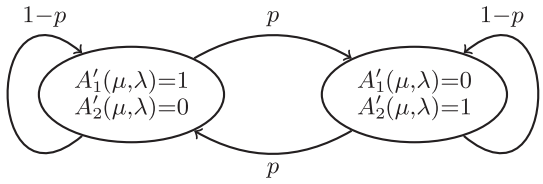


Fig. 5. When 2 sources are active in frame λ, a second Markov model is used to decide which of the sources contribute to the microphone signal at frequency bin μ. This is not required when speech is used for the source signals instead of noise.

A. Modeling Temporal Changes

1) Source activity and locations: If the source positions in the training data are static, the DOA estimator will not learn to forget previously active sources. On the other hand, if the considered scenarios are too dynamic, i. e., if there is a lack of consistency between the source positions in consecutive frames, it will not be able to properly exploit temporal information.

We aim for a good compromise by modeling the activity of a source with index j by a simple Markov chain $A_j(\lambda)$ with two states $A_j(\lambda) = 1$, and $A_j(\lambda) = 0$. For $A_j(\lambda) = 1$, the source is active in the frame with index λ, for $A_j(\lambda) = 0$ it is inactive. This is enforced in the signal model (1) by multiplying its contribution to the microphone signals with $A_j(\lambda)$, i. e.,

$$Y_n(\mu,\lambda) = \sum_j A_j(\lambda)\, S^{\text{mic}}_{j,n}(\mu,\lambda) + V_n(\mu,\lambda). \tag{6}$$

The transition probability between the two states is controlled by a single parameter p with $0 < p \ll 0.5$. The posterior probability of source activity is then given by

$$P\left(A_j(\lambda+1)=1|A_j(\lambda)\right) = (1-p)\cdot A_j(\lambda) + p\cdot\left(1-A_j(\lambda)\right), \tag{7}$$

and the probability of source inactivity

$$P\left(A_j(\lambda+1)=0|A_j(\lambda)\right) = p\cdot A_j(\lambda) + (1-p)\cdot\left(1-A_j(\lambda)\right). \tag{8}$$

This is illustrated in Fig. 4. Initially, $A_j(\lambda = 0) = 1$ for all j, i. e., all sources are active. For the duration of source activity, the location of the source is not changed, i. e., the DOA and, consequently, the RIR are time-invariant. Once a previously inactive source becomes active again, a new location (i. e., a new combination of DOA and distance to the array) is selected. The current DOAs of the remaining sources are excluded from this random selection, so that the sources remain spatially separated at all times.

By changing the source positions randomly, we do not make any assumptions about the nature of the change. For example, the case of a gradually moving source is inherently covered as well, as this corresponds to a transition between neighboring DOA classes. Alternatively, the model could be extended to incorporate slow source movements directly, although this will not be considered in the following.

2) Source mixing: Whenever more than one source is active at the same time, the sources are mixed such that only a single source contributes to the microphone signals in any time-frequency bin, as in [22]. When considering frame sequences, as with the source activity $A_j(\lambda)$, the allocation of the sources to each frequency should change gradually over time. Therefore, a similar strategy will be adopted as for the source activity model. When at most 1 source is active, there is no need to mix contributions from different sources, and (6) can still be applied. For the case where more than 1 source is active, i. e., $\sum_j A_j(\lambda) > 1$, a second random process $A'_j(\mu,\lambda)$ is used to decide whether the source with index j contributes to the microphone signals at frequency μ, i. e.,

$$Y_n(\mu,\lambda) = \sum_j A_j(\lambda)\, A'_j(\mu,\lambda)\, S^{\text{mic}}_{j,n}(\mu,\lambda) + V_n(\mu,\lambda). \tag{9}$$

To enforce an ideal W-disjoint orthogonality, $A'_j(\mu,\lambda)$ is 1 only for one source, and 0 for all others. For conciseness, we choose to consider a maximum of 2 sources that can be active at the same time in the training data. The index j for which $A'_j(\mu,\lambda) = 1$ is then determined by the current state of a second Markov chain model for which the same transition probability p is used here. This is illustrated in Fig. 5. The initial state of the model is selected at random, independently for each frequency.

One realization of the process $A_j(\lambda)$ is shown at the top of Fig. 6, the corresponding realization of $A_j(\lambda)\cdot A'_j(\mu,\lambda)$ is shown at the bottom. When fewer than 2 sources are active, i. e., $\sum_j A_j(\lambda) < 1$, $A'_j(\mu,\lambda)$ is formally set to 1 for all sources j and frequencies μ, so that (9) reduces to (6).

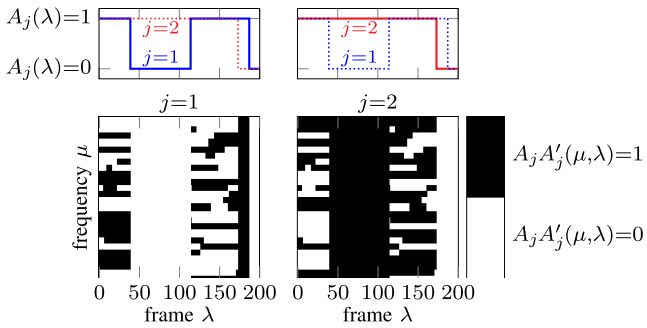


Fig. 6. Example illustrating how the activity of two sources changes over time, as described by the process $A_j(\lambda)$ (top), and over time-frequency, as described by the process $A_jA'_j(\mu,\lambda) = A_j(\lambda)\cdot A'_j(\mu,\lambda)$ (bottom). For $40 \le \lambda \le 114$ and $174 \le \lambda \le 187$, only one of the sources is active, so all time-frequency bins are allocated to this source. From $\lambda = 188$ on, neither of the sources is active, so $A_jA'_j(\mu,\lambda) = 0$ for both sources. In all other frames, each frequency is allocated to exactly one of the 2 active sources.

B. Training on Speech

The central argument in favor of using noise for training according to [21], [22] is the difficulty of assigning meaningful ground truth labels when realistic source signals are used. However, when considering sequences of frames, the exclusion of silent periods is no longer critical. During brief speech pauses, it can even be desirable to have the network preserve the position where an active speaker was last seen, given that the speaker's location may not have changed. When there is an absence of speech in the current frame, the availability of information from the past ensures that a high DOA probability target is always based on true source activity. It is only when a speaker has truly fallen silent, that the network should detect this, and no longer return significant probabilities for the corresponding direction.

Overall, this implies that it is more practicable to use speech as source signals for training when temporal context is used. Although this entails the requirement of a large database of source signals for training data generation, this is not a major limitation as speech databases that can be used for this purpose are available, e. g., [35]–[37].

It should be noted that the feature extraction is still done independently for each frequency bin to ensure robustness when multiple speakers are active. Nonetheless, it is possible to take spectro-temporal signal characteristics into account to some degree in the subsequent layers, where the information from different frames and frequencies is considered jointly.

When speech is used for the source signals instead of noise, it is no longer necessary to artificially impose the property of W-disjoint orthogonality on the training data. Therefore, the process $A'_j(\mu, \lambda)$ introduced in Section V-A2, which assigns each time-frequency bin to exactly one source, is not needed, and mixing equation (6) can be used instead of (9) in the multisource case as well.

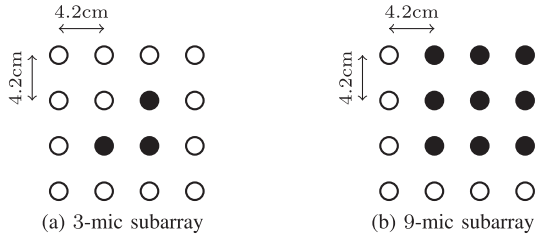
VI. Evaluation

All signals are sampled at $f_s = 16\,\mathrm{kHz}$ for the DOA estimation. For the transformation into the STFT domain, the frame length and frame shift are, respectively, set to 512 and 160 samples (one frame per $10\,\mathrm{ms}$), and a square-root Hann window function of length 512 samples is used.

A. Training Set and Hyperparameter Selection

For a closer examination of whether various aspects of the proposed training data generation procedure contribute to a more accurate DOA estimation, models are trained independently for different training configurations. The *default* setting is described in detail in this section, any deviation from this will be indicated explicitly.

For the source signals, we use dry speech taken from the TIMIT [35] and PTDB-TUG [36] databases, both of which consist of recordings of sentences from the TIMIT corpus. The division into training and validation sets is performed such that $7\,438$ utterances are available for training, and $2\,280$ for validation. A minor overlap between the spoken sentences exists, but the sets are nonoverlapping in terms of the speakers.



(a) 3-mic subarray (b) 9-mic subarray

Fig. 7. Two different subarrays (•) of a 16-microphone uniform rectangular array (URA) will be considered. The geometry corresponds to the miniDSP UMA-16 array [39].

Sufficiently long source signals are obtained by concatenating randomly selected combinations of utterances. Silent segments are truncated to a maximum duration of $10\,\mathrm{ms}$.

As described in Section V-A1, the sources are not necessarily active at all times, and it is possible for them to change their position. Effectively, the parameter p, that describes the probability of a state transition in the models depicted in Figs. 4 and 5, controls how trusting the network is when it comes to information from the past. For the choice of the sequence duration and the parameter p, it should be taken into consideration that: (i) the sequence duration should be sufficiently long so that the DOA estimator can learn to properly exploit information from the past for relatively static sources, (ii) it should be relatively common for sources to change their position within the selected sequence duration, so that the DOA estimator can learn to detect this quickly. Here, we choose a sequence duration of $2\,\mathrm{s}$, and set $p = 1/150$, which amounts to one state transition every $1.5\,\mathrm{s}$ on average when the frame shift is $10\,\mathrm{ms}$. For the TCN in the CNN/TCN architecture, the number of layers is set to $\mathcal{L} = 5$, and the filter length to $K = 3$. This selection was made by fixing the filter length K, and then choosing the number of layers $\mathcal{L}$ such that an appropriate number of past frames is taken into account. The resulting receptive field size is 63 frames ($630\,\mathrm{ms}$). Empirically, we find that these parameters deliver good, albeit not necessarily optimal, results for the considered test setup (as described in Section VI-B). As the fine-tuning is also a question of the application considered, we do not perform a dedicated optimization of these hyperparameters.

Multichannel RIRs are generated using [30]. Regarding the acoustic conditions, we choose to add some more variability compared to the parameter selection in [22]. As shown in Table II, the number of different rooms for the training data is $R = 10$, there are $P = 7$ different array positions in each room, and $D = 4$ different distances between the source and the array. For validation, an additional $R' = 3$ rooms are simulated for $P' = 4$ positions, and $D' = 4$ different distances. Two different array geometries are used that, respectively, comprise, 3 and 9 microphones. The relative positioning of the microphones for these arrays is depicted in Fig. 7. In order to optimally align the DOA estimator with the task that the evaluation setup poses, which consists of sources that are positioned at different azimuth angles between 0° and 180°, we choose to restrict the DOA estimation to the same range. For a resolution of 5°, this amounts to a total of $I = 37$ DOA classes.

TABLE II
DEFAULT CONFIGURATION FOR THE TRAINING AND VALIDATION DATA GENERATION, ANY DEVIATION WILL BE INDICATED EXPLICITLY

Source signals	Speech [35], [36] (unless stated otherwise)
Room dimensions in m	R1: $(6 \times 6 \times 2.5)$, R2: $(5 \times 4 \times 2.8)$, R3: $(10 \times 6 \times 2.4)$, R4: $(8 \times 3 \times 3.1)$, R5: $(8 \times 5 \times 2.9)$, R6: $(4 \times 9 \times 3.3)$, R7: $(7 \times 7 \times 2.3)$, R8: $(5 \times 6 \times 3.6)$, R9: $(9 \times 6 \times 3.2)$, R10: $(11 \times 7 \times 3)$, R11: $(5.5 \times 7.5 \times 2.65)$, R12: $(8.5 \times 4.5 \times 3.45)$, R13: $(6.5 \times 6.5 \times 2.25)$
Array positions in room	Training: 7 different positions in each of the rooms R1 to R10, Validation: 4 different positions in each of the rooms R11 to R13 Minimum distance to the nearest wall (excluding floor and ceiling) is $0.5\,\mathrm{m}$
Source-array distance	Training: 20%, 40%, 60% and 80% of the distance between array and wall for each DOA Validation: 30%, 50%, 70% and 85% of the distance between array and wall for each DOA
T_{60}	R1: $0.3\,\mathrm{s}$, R2: $0.2\,\mathrm{s}$, R3: $0.8\,\mathrm{s}$, R4: $0.4\,\mathrm{s}$, R5: $0.6\,\mathrm{s}$, R6: $0.5\,\mathrm{s}$, R7: $0.7\,\mathrm{s}$, R8: $0.45\,\mathrm{s}$, R9: $0.55\,\mathrm{s}$, R10: $0.75\,\mathrm{s}$, R11: $0.525\,\mathrm{s}$, R12: $0.625\,\mathrm{s}$, R13: $0.475\,\mathrm{s}$
Additive noise	Simulated (as described in [38]) for a spherically isotropic noise field (unless stated otherwise) SNR uniformly sampled from 0 to $30\,\mathrm{dB}$

TABLE III
NUMBER OF SOURCES PRESENT IN THE SCENARIOS INCLUDED IN THE TRAINING AND VALIDATION DATASETS

case	fraction of dataset	sources active at $\lambda = 0$	sources active at $\lambda > 0$
2 sources	$1/3$	$=2$	≤ 2
1 source	$1/3$	$=1$	≤ 1
0 sources	$1/3$	$=0$	$=0$

In [22], the authors show that their CNN generalizes reasonably well to the localization of more than 2 concurrently active sources even when exactly 2 sources are active at all times in the training set. This is mostly because, due to the temporal sparsity of the signals, different sources can be localized in different frames, so that the averaged probabilities (3) (or (4)) can still exhibit significant peaks for more than 2 DOA classes. For the proposed extensions, information from previous frames is already taken into account, so that further averaging is not strictly required. Therefore, and for generally improving the performance for a variable number of active sources, we choose to include cases with different numbers of sources in the training set in equal shares. Along with scenarios where 1 or 2 sources are active, we expect that it will be easier for the estimator to separate localized signal components from diffuse and spatially uncorrelated noise when a case with only noise is included as well. For this "0-sources" case, the ground truth probabilities are set to 0 for *all* DOA classes. A summary of the composition of the training set regarding the number of sources is presented in Table III.

With respect to the additive noise $V_n(\mu, \lambda)$, [22] simply makes use of temporally and spatially white noise. Although a good generalization to diffuse noise is reported, for a better match of the training data with realistic conditions, temporally uncorrelated noise simulated [38] for a spherically isotropic noise field is used here instead.

The mini-batches used for training the CNN/LSTM and CNN/TCN architectures consist of 20 sequences each (a total of $4\,000$ frames). For the remaining parameters, the same values are used as for the CNN (see Section III-C). Training and validation data are generated online during training time. The models are given plenty of time to converge, i. e., training is performed until the training and validation loss curves suggest that no further improvement can be expected. For CNN/LSTM and CNN/TCN, we found that this is the case for a total of 3.7 million training *sequences* (about $2\,072$ hours). At this point, for the single-source scenarios alone, each of the $R \cdot P \cdot D \cdot I = 10\,360$ impulse responses has been used 120 times. For the plain CNN, a total of 89.5 million independently generated training *frames* (about 796 hours) were found to be more than sufficient to guarantee convergence. The models are saved regularly while being trained, so that ultimately, the snapshot with the lowest validation loss can be selected for the evaluation.

B. Evaluation Setup and Performance Measures

Table IV shows an overview of the setup used to obtain microphone signals for the evaluation. Clean speech is convolved with room impulse responses that were measured using exponential sine sweeps [41] at $f_s = 48\,\mathrm{kHz}$. The recording setup consisted of a loudspeaker that was placed at azimuth angles $\varphi = 0^\circ, 20^\circ, \ldots, 180^\circ$ for distances $1\,\mathrm{m}$, $2\,\mathrm{m}$, or $3\,\mathrm{m}$ from the miniDSP UMA-16 array [39] in a meeting room with a reverberation time of about $660\,\mathrm{ms}$ (approximate room dimensions: $7.50\,\mathrm{m} \times 5.00\,\mathrm{m} \times 2.65\,\mathrm{m}$). For $3\,\mathrm{m}$, the size of the room permitted only a reduced set of angles $\varphi = 40^\circ, 60^\circ, \ldots, 140^\circ$. Our aim to incorporate more RIRs with larger source-array distances (to generate more challenging conditions for the localization) necessitated in us having to place the array closer to one end of the room. This meant, however, that we could not record RIRs for the full 360°-range.

After the convolution, noise is added at a fixed broadband SNR, and the resulting microphone signals are downsampled to $f_s = 16\,\mathrm{kHz}$. First, in Section VI-C, synthetic noise that is generated in the same way [38] as it is done for the training data will be used. For the benchmark in Section VI-D, recorded noise will be considered instead. The setup used for this recording consisted of one loudspeaker in each of the 4 corners of a lecture room ($T_{60} \approx 1\,\mathrm{s}$), and the array that was placed in a central position on a table. To obtain a good diffuseness, the pub noise signal from the ETSI background noise database [40] was played back by all loudspeakers simultaneously, with slightly different time delays.

Each source signal is obtained by concatenating 5 randomly selected utterances from the TSP speech database [37], which consists of $1\,444$ utterances in total. To simulate the dynamic

TABLE IV
CONFIGURATION USED TO OBTAIN MICROPHONE SIGNALS FOR THE EVALUATION

Source signals	Speech [37] (concatenation of 5 randomly selected utterances)
Impulse responses	Recorded (miniDSP UMA-16 array [39]) in a meeting room, approx. dimensions: $7.50\,\mathrm{m} \times 5.00\,\mathrm{m} \times 2.65\,\mathrm{m}$, $T_{60} = 0.66\,\mathrm{s}$
Additive noise	Sec. VI-C: simulated (as described in [38]) for a spherically isotropic noise field Sec. VI-D: recorded spatially (relatively) diffuse pub noise [40]
DOAs	Probability 50% for the DOA to change between two utterances Distances $1\,\mathrm{m}$ and $2\,\mathrm{m}$: azimuth angles $\varphi = 0^\circ, 20^\circ, \ldots, 180^\circ$ Distance $3\,\mathrm{m}$: azimuth angles $\varphi = 40^\circ, 60^\circ, \ldots, 140^\circ$
Scope of the evaluation	25 sets of microphone signals for each parameter setting (SNR, source-array distance, number of active sources)

nature of the sources, the position of a source is changed with probability 50% after each utterance. Whereas in the training data DOA and distance were changed at the same time, we now only select a different angle for the new source position, so that the influence of the source-array distance, and the number of sources can be independently assessed based on the evaluation results. It is again ensured that two sources are never at the same position, i. e., the difference in the angles corresponding to two neighboring sources is at least 20° at all times.

Following the computation of the posterior probabilities with either of the CNN architectures (plain CNN, CNN/LSTM, or CNN/TCN), frames with very low speech activity are removed from the evaluation. This is done to prevent a significant bias that could result from the inclusion of segments where the speakers to be localized are silent. The threshold is set to $10\,\mathrm{dB}$ below the global mixing SNR, assuming ideal knowledge of the additive noise, and the contribution of the speech signals to the microphones. Note that oracle knowledge of the mixed signals is used for this (optional) post-processing step only. Finally, the posterior probabilities can be averaged recursively according to (4). The time constant is set to $L = 30$ frames ($0.3\,\mathrm{s}$). The averaging is applied for the plain CNN and, as this was observed to still improve the results, for CNN/TCN. For CNN/LSTM, the averaging was found to be unnecessary, and is therefore omitted.

From the resulting probabilities $P_\lambda(\phi_i)$, the DOA estimates are extracted by taking the $\widehat{J}$ highest peaks. Because we aim to evaluate the approaches in terms of their *DOA estimation* performance, irrespective of how the number of sources is determined, we assume ground truth knowledge of J. Hence, the number of peaks taken into consideration is set to $\widehat{J} = J$ in Section VI-C. Before the estimation errors can be determined, the underlying assignment problem between the $\widehat{J}$ estimated, and the J true DOAs must first be solved. This is done by selecting the one permutation that yields the lowest sum of absolute errors. For each unique set of parameters (i. e., fixed architecture, training set composition, source-array distance, array geometry, SNR, and number of sources), a total of 25 evaluations is run. From all individual estimates (one estimate for each frame and each source of the 25 independent runs), the localization *accuracy* (acc) for this parameter setting is determined as the fraction of estimates where a fixed "tolerated" error threshold is not exceeded. In Section VI-C, this threshold is set to 5°. Additionally, we define the localization *inaccuracy* as $1 - \mathrm{acc}$.

As baselines, we consider a narrowband implementation of SRP-PHAT [5] as well as the informed phase unwrapping (IPU)-least-squares (LS) method [10]. Analogously to the CNN based approaches, frames with insufficient speech activity are not taken into consideration. To obtain an estimate of the power spectral density (PSD) matrix of the microphone signals, which is required for both approaches, recursive averaging with time constant $50\,\mathrm{ms}$ (5 frames) is used. Based on the narrowband estimates, the broadband DOAs are then determined by taking the $\widehat{J}$ highest peaks of the corresponding histogram. To ensure comparability, the histogram includes the estimates from the past 30 frames ($0.3\,\mathrm{s}$), and the bins have a width of 5°. As the localization is restricted to the range $0^\circ \leq \varphi \leq 180^\circ$, outliers (with a tolerance of 15°) are excluded. For IPU-LS, initial estimates are required to cope with spatial aliasing at higher frequencies. For each of the $\widehat{J}$ sources, one initial estimate is first determined following the same histogram approach. At the time-frequency bin with index (μ', λ'), only the final estimates for lower frequencies ($\mu < \mu'$) of the same frame ($\lambda = \lambda'$) are taken into account to obtain this histogram. From the resulting $\widehat{J}$ candidates, the one that yields the smallest LS error is used as the initial estimate.

C. Evaluation of the Proposed Approach

In Figs. 8 to 10, results for two different variants (referred to generically as variants "X" and "Y") of the CNN DOA estimator are plotted against each other. In each case, the x-axis represents the localization accuracies obtained with X, whereas the y-axis represents the accuracies obtained with Y. Consequently, a data point above the main diagonal, which is labeled $\pm 0\%$, indicates that Y performs better than X, a data point below the main diagonal indicates that X performs better than Y. For better illustration, we have inserted further diagonal grid lines (.....) which indicate an improvement of one method over the other in 10% steps, as indicated by the labels $\pm 10\%$, $\pm 20\%$, $\pm 30\%$, $\pm 40\%$, and $\pm 50\%$. Each displayed point corresponds to one set of parameters, i. e., a fixed SNR, source-array distance, and number of concurrently active sources. As the legend shows, the SNR is encoded by the *shape*, the distance by the *size*, and the number of sources by the *color* of the markers.

1) Effect of the composition of the training set: First, in Fig. 8, only the original architecture proposed in [22] (plain CNN) is considered, and white noise source signals are used for training instead of speech. This being the case, individual frames are generated independently for training, and the models introduced in Section V-A do not yet play a role. For the results on the y-axis (variant Y), all parameters were set exactly as in the "default configuration" described in Section VI-A, as far as applicable to the plain CNN. For variant X, in contrast, the data

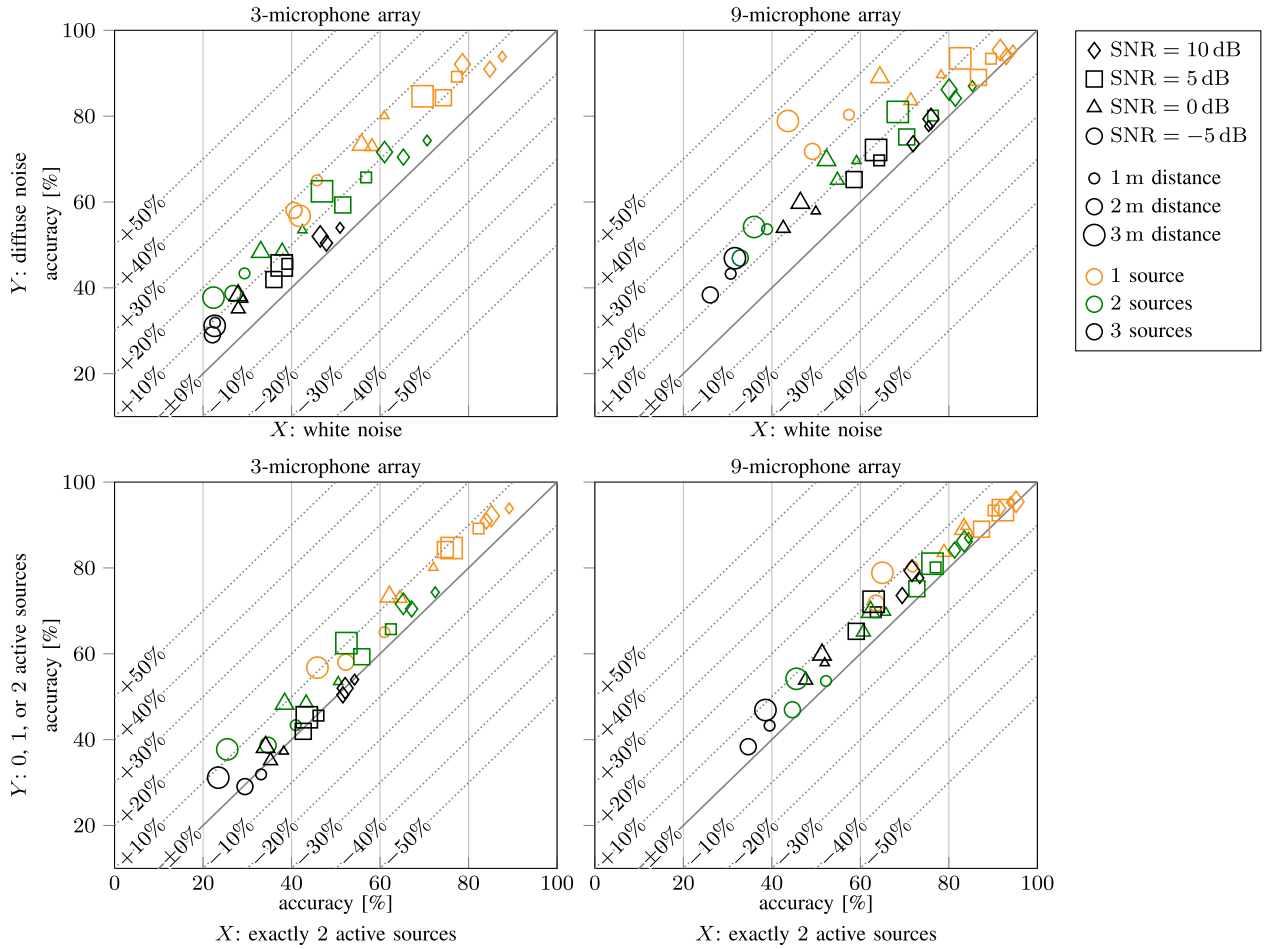


Fig. 8. The proposed modifications in the training data generation improve the localization accuracy attained with the plain CNN, which motivates their use in the following. This includes training on spatially diffuse instead of uncorrelated noise (top plots) and a varying number of active sources (bottom plots).

generation was aligned with the procedure used in [22] in terms of a single aspect only, as indicated by the x-axis label.

Because the evaluation is conducted using simulated diffuse noise, the properties of the additive noise seen during training and testing align favorably when the setting described in Section VI-A is used. The results shown in the figure confirm that the localization accuracies indeed reflect this. Although a satisfactory generalization to diffuse babble noise is reported in [22], we observe that the performance clearly improves throughout all conditions when spatially diffuse noise is used for training instead of uncorrelated noise. This applies to both the 3-microphone (top left plot), and the 9-microphone (top right plot) arrays. For most test parameter combinations, the improvement of the localization accuracy lies in the range 0% to 20%. An even greater improvement is observed in the single-source test case for the 9-microphone array, e. g., for $-5\,\mathrm{dB}$ SNR at a distance of 3 m (◯). Because directional interference and diffuse noise components (e. g., late reverberation) typically dominate over spatially uncorrelated noise (e. g., sensor noise) in a realistic scenario, we will use spatially diffuse noise for training in the following.

In the second row of the figure, the results for a training set where the number of active sources is fixed to 2 (X) are compared with the results for an equally large set where the number of active sources can be either 0, 1, or 2 (Y). The composition of the training set is then as shown in Table III (with the frame index fixed to $\lambda = 0$, as independently generated frames are used to train the plain CNN). Although the resulting difference is only in the range 0% to 10% for most test parameter combinations, the improvement is again fairly consistent throughout all conditions. Interestingly, this applies even to the 2-source test case, i. e., when a training set consisting only of scenarios with exactly 2 active sources should, in theory, be the optimal choice. This can likely be explained with the fluctuations in the level of the employed speech signals, as opposed to the noise used for the training data. The ability to generalize to the presence of 3 sources, on the other hand, only seems to improve marginally when sequences with a lower number of active sources are included in the training set.

2) Effect of training on speech: To complete the evaluation of the modifications in terms of the training data generation, it will now be investigated how it affects the performance when speech source signals are used for training instead of noise. For determining the ground truth labels in the training data, the presence of relatively silent segments was *not* taken into account, i. e., the target was set to 1 for the corresponding DOA class regardless

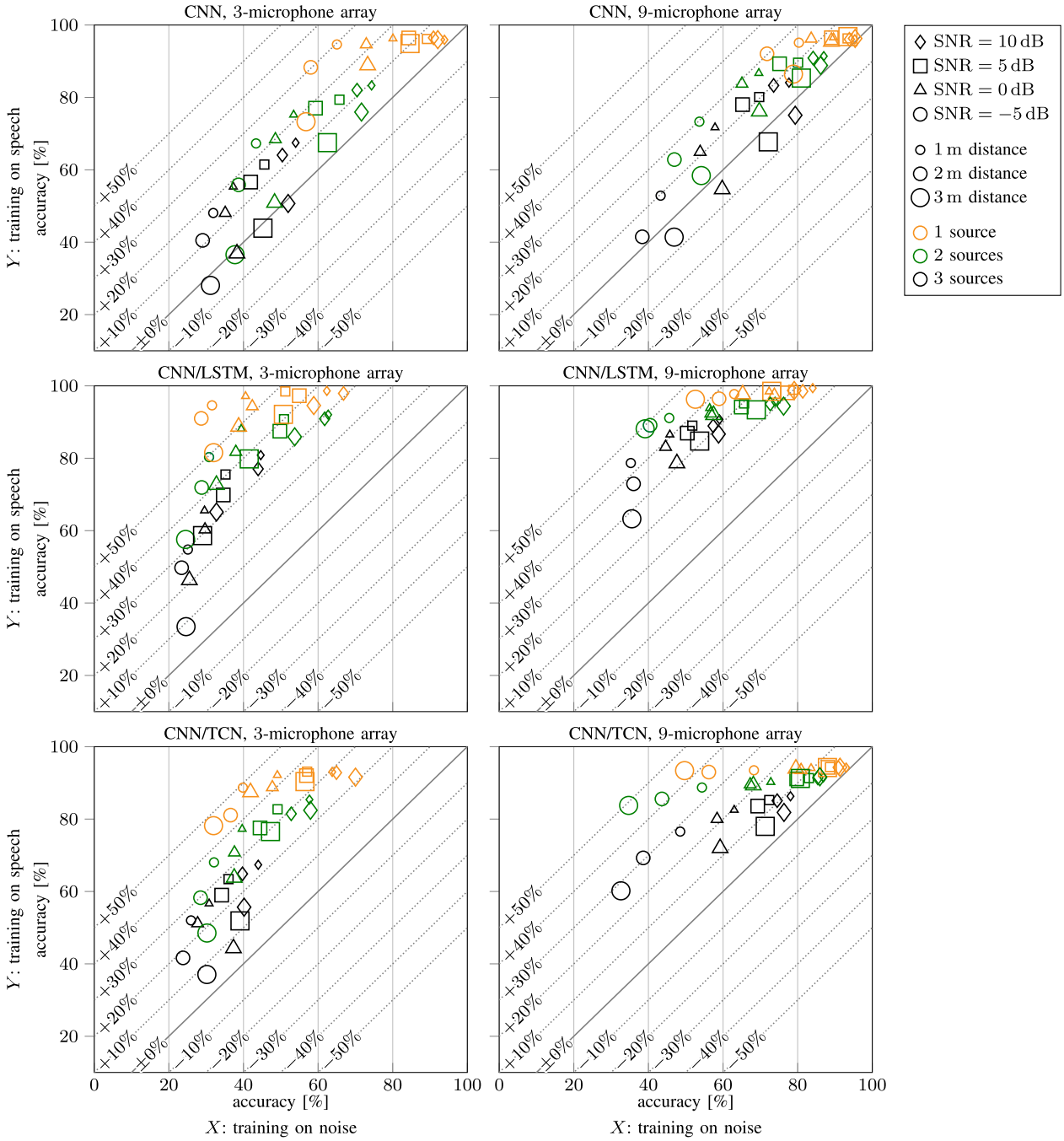


Fig. 9. Using realistic source signals (y-axis) instead of noise (x-axis) for the training data generation can be helpful for the plain CNN, even when the source signals are not taken into account for setting the ground truth labels. For the proposed extensions, it is even important to do so.

of how large the contribution of the source to the microphone signals is. As the proposed LSTM and TCN extensions are able to exploit temporal context as well, it is instructive to perform this evaluation for each architecture separately. When temporal information is available, it is not crucial for the ground truth labels to reflect the level of the source signal in the current frame. Additionally, the temporal properties of the employed source signals become relevant.

Despite the limitations of the pure CNN, as the top row of Fig. 9 shows, an improvement can be observed for most test parameter combinations when training is conducted on speech (Y), as opposed to training on noise source signals (X). Almost exclusively for the 3-source test case at the 3 m distance (◊□△◯), there is a slight deterioration.

The results for CNN/LSTM are shown in the middle row, and for CNN/TCN in the final row of the figure. Note that for training the architectures used in these two approaches, as opposed to the plain CNN, independently generated frames cannot be used as this would prevent an exploitation of temporal context. Further, if only static sources were considered for training, the results obtained for a dynamic scene would not be meaningful, as the DOA estimator would not learn to forget previously active sources.

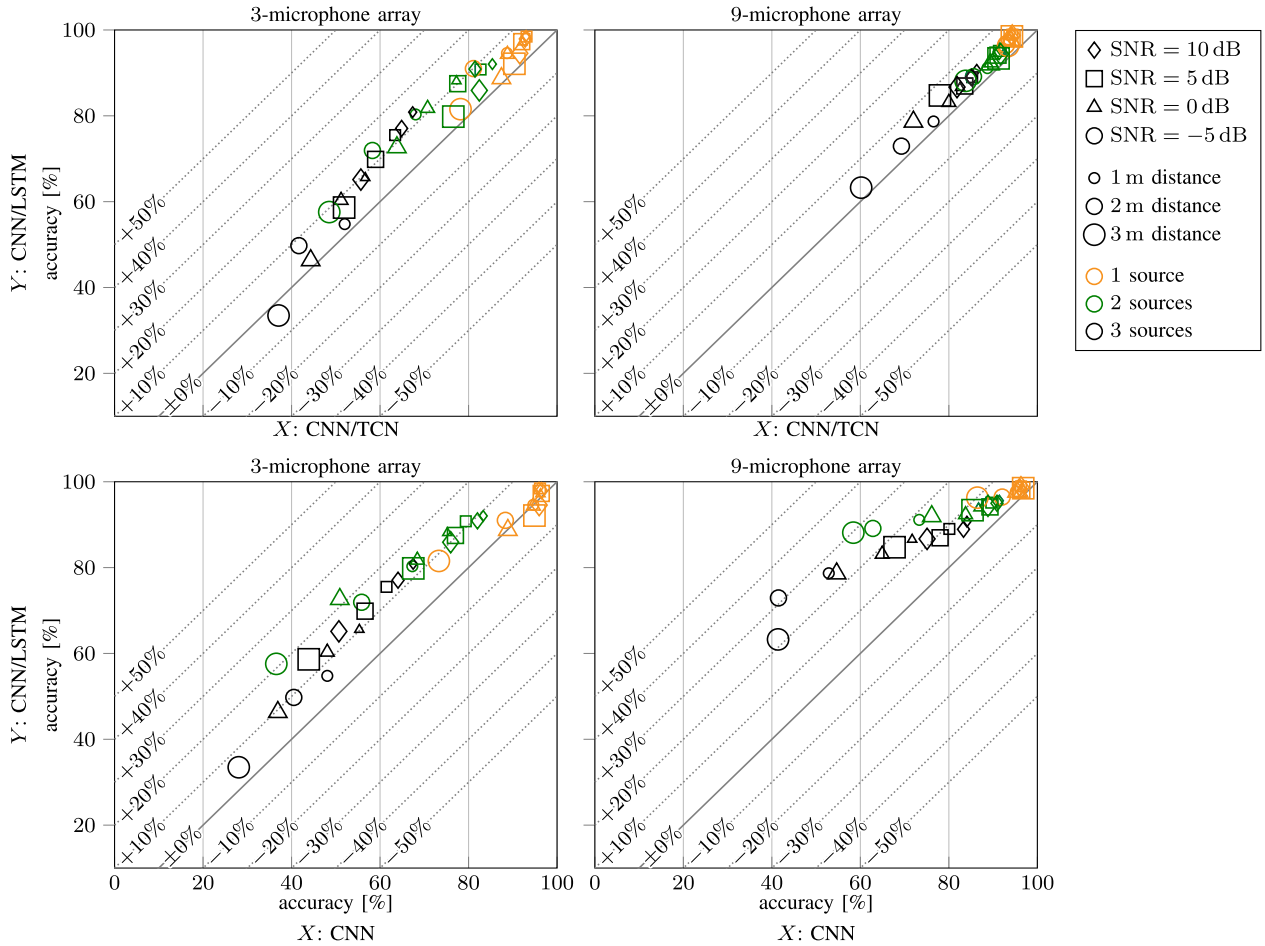


Fig. 10. The performance is slightly better for CNN/LSTM than for CNN/TCN (top plots). The differences are fairly small but consistent throughout different conditions. Compared to the plain CNN (bottom plots), the improvement is considerable especially for the 9-microphone array.

Therefore, we will always generate sequences as detailed in Section V for training the CNN/LSTM and CNN/TCN networks.

The benefit of using realistic source signals for training is clearly more pronounced for CNN/LSTM and CNN/TCN than for the plain CNN. This applies especially to the single-source test case, where the approximative nature of the model (Fig. 5) employed when 2 sources are active simultaneously in the training with noise source signals is not even relevant. For CNN/LSTM, the resulting difference in the localization accuracies even exceeds 30% for almost all of the considered conditions.

3) Effect of the exploitation of temporal context: The results shown in Sections VI-C1, and VI-C2 demonstrate the effectiveness of the proposed configuration for generating training data. In the following, a training set with a variable number of speech sources, and simulated diffuse additive noise will therefore be used for the plain CNN, as well as the LSTM and TCN extensions. Next, these extended architectures that incorporate temporal context will be compared against the plain CNN.

In the top row of Fig. 10, CNN/TCN (X), and CNN/LSTM (Y) are compared against each other. Although the performance is similar across all conditions for both subarrays, CNN/LSTM quite consistently yields marginally better localization accuracies (difference between 0% and 10%), at least for the configuration of the TCN chosen here ($\mathcal{L} = 5$, $K = 3$). One reason for the good performance of CNN/LSTM is that, because the averaging step (4) is omitted, it is able to detect DOA changes more rapidly.

Due to the slightly better performance of CNN/LSTM, this extension is also considered for the comparison with the plain CNN architecture in the bottom row of Fig. 10. For the 3-microphone array, improvements between 10% and 20% are observed for the 2- and 3-source localization. In terms of the single-source localization, the performance cannot improve significantly, as accuracies of 90% or more are already obtained with the CNN. Especially for more demanding conditions, the benefit of exploiting temporal context with CNN/LSTM is even greater for the 9-microphone than for the 3-microphone array. For up to 2 sources, the accuracies obtained with CNN/LSTM exceed 90% across all considered conditions.

The test case with 3 concurrently active sources is of particular interest because it gives an indication how well the approaches generalize to a greater number of active sources than seen during training. This may seem to put the plain CNN at an advantage since the independent processing of each frame favors

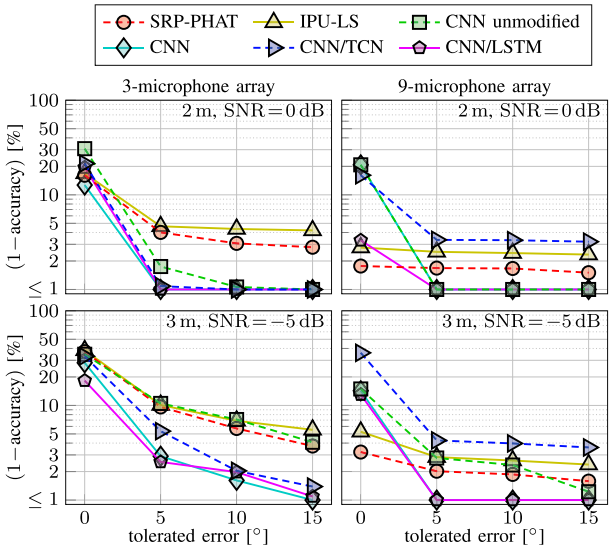


Fig. 11. Single-source localization *inaccuracy* (the lower the better). The DOA estimation is almost perfect, when tolerating an error of 5°, with both the "modified" plain CNN (i. e., training set as in Section VI-A as far as applicable to the plain CNN), and with CNN/LSTM for the considered conditions.

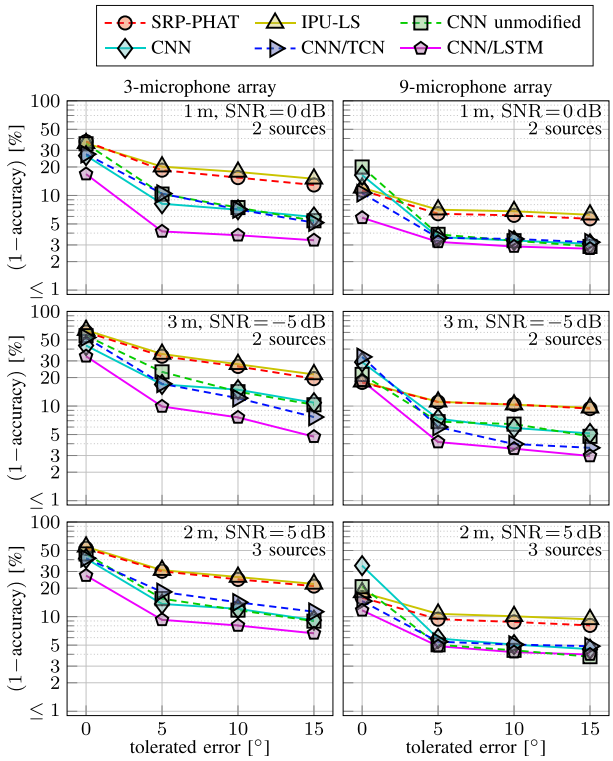


Fig. 12. Multisource localization performance. All variants of the CNN now perform better than the classical methods. CNN/LSTM is the best-performing of all of the considered methods.

the localization of different sources in different frames, and thus the successful detection of *all* present sources in a series of frames. Nevertheless, Fig. 10 clearly shows that the improvement over the plain CNN attained with the LSTM extension is still substantial, albeit less pronounced, even for the localization of 3 sources.

D. Comparison Against Baselines

The performance of the final system will now be compared against various baseline approaches. To demonstrate that the previously made observations are transferable to realistic conditions, the pub noise recording, as described in Section VI-B, is used instead of artificially generated diffuse noise. In order to account for incorrectly detected sources that could be related to the spatial characteristics of the recorded additive noise, the number of estimated DOAs is increased by 1, i. e., the next highest peak of the histogram, or of the (averaged) DOA probabilities is considered as well ($\widehat{J} = J + 1$). To determine the estimation errors that the accuracy metric is based on, the permutation that minimizes the sum of absolute errors is selected as outlined in Section VI-B. In this case, one of the estimates is therefore discarded.

Fig. 11 shows the performance for the single-source localization in terms of the localization *inaccuracy* for two selected combinations of the source-array distance, and the SNR (as indicated in the top right of each subplot). Note that it is easier to estimate the DOAs in the presence of *correlated* noise, as it is possible to benefit from time-frequency bins that are less affected by the noise. Therefore, the results (quantitatively) differ quite strongly from those in Figs. 8 to 10, and we have now selected a logarithmic scale. Along with SRP-PHAT (-○-), and IPU-LS (-△-), two variants of the plain CNN [22] are included as baselines in Fig. 11. For the "unmodified CNN" (-□-), the training procedure is adjusted to match that of [22], i. e., source signals are noise, the additive noise is spatially uncorrelated, and the number of sources in the training set is fixed to 2. In contrast, the results labeled as "CNN" (-◇-) are obtained *with* the modifications outlined in Section VI-A.

Both the CNN with the modified training and the CNN/LSTM approach (-⬠-) exhibit 5°-inaccuracies of less than 3% even for the most difficult conditions considered here, i. e., 3 m distance at SNR $= -5$ dB with only 3 microphones, whereas significant errors are still observed for the unmodified CNN. Given these results, a closer examination of the differences between CNN and CNN/LSTM based on test cases with only a single active source is hardly instructive. CNN/TCN (-▷-), on the other hand, performs only slightly worse for the 3-microphone array, but poorest of all compared methods for the 9-microphone array. A closer examination reveals that CNN/LSTM produces sharper peaks than CNN/TCN, which explains why sources tend to be localized incorrectly more commonly with CNN/TCN. This effect is more prominent for realistic noise, which does not exhibit an "ideal" diffuseness like the simulated noise used in Section VI-C. The localization accuracies attained with the classical methods are fairly good for the single-source test case as well (5°-inaccuracies of at most 10% for all of the considered conditions), but they are clearly outperformed by CNN, and CNN/LSTM.

In Fig. 12, results are displayed for 2 (first, and second row), or 3 (bottom row) active sources, again for selected combinations of the source-array distance, and the SNR. Under these conditions,

the performance advantage of the CNN based methods over the classical methods is more apparent. In contrast to the single-source test case where the CNN with the modified training set also performed very well, the results for CNN/LSTM are now clearly the best at least in the 2-source test case. It is the only method for which the 5°-inaccuracies are no higher than 10% for any of the conditions considered.

The results for the 3-source test case are mostly in line with Fig. 10. CNN/LSTM remains the best performing approach, but the improvement compared to other methods is less pronounced, especially for the 9-microphone array. Nevertheless, the potential disadvantage that a method incorporating temporal context has when it comes to coping with scenarios with a larger number of sources could straightforwardly be addressed by adjusting the composition of the training set accordingly.

As for CNN/TCN, the DOA estimation performance is decent, but a considerable improvement over the plain CNN is not observed. Unlike the single-source test case, the performance of CNN/TCN appears to fall behind especially for the 3-microphone array. This can again be attributed to the more distinct peaks that CNN/LSTM produces compared to CNN/TCN. In view of the promising results of Fig. 10, which do not translate to equally good results in comparison with the plain CNN in Figs. 11 and 12, the potential of the CNN/TCN approach may not be fully exploited yet.

Also, again in contrast to the results of Fig. 11, for the plain CNN we do not see that the changes to the composition of the training set produce a consistent improvement of the localization accuracies in Fig. 12. This conforms with the results of Fig. 9, which showed a better multisource DOA estimation performance of the "modified" CNN (that uses speech instead of noise source signals for training) only at small source-array distances. For CNN/LSTM and CNN/TCN, however, Fig. 9 indicates that realistic source signals remain a requirement.

Overall, the results demonstrate that, with the proposed framework for synthetically generating training data, particularly the CNN/LSTM architecture performs very well both in the presence of simulated, as well as recorded noise. If a hard limitation of the amount of past frames that are taken into account is desired, CNN/TCN can still be an interesting alternative. Due to its good performance without requiring the introduction of additional hyperparameters, the LSTM extension, however, appears to be the more suitable choice for most applications.

VII. Conclusion

In this work, we have studied how long-term temporal context can best be incorporated into CNN based multisource DOA estimation when a large recorded dataset is not available. The implication of this is twofold: On the one hand, the architecture of the CNN DOA estimator considered here, which operates on data from a single frame, must be extended so that information from previous frames can be taken in account. On the other hand, the procedure for generating training data must be adjusted to reflect that sequences are required instead of individual frames.

The exploitation of temporal context enables the DOA estimator to make use of the characteristic temporal properties of the underlying source signals, in our case speech, and to properly track sources over time. This includes taking into account that there is typically a large consistency of the DOAs in consecutive frames, whereas occasionally, they may change abruptly.

With regard to the network architecture, we consider two simple extensions that, respectively, consist of the integration of an LSTM layer, or of a TCN into the CNN. For the training data generation, we distinguish between training on noise source signals, which eliminates the need for a source signal database, and training on speech source signals. In both cases, Markov chains are employed to simulate the temporal evolution in the training data. Although we do not explicitly incorporate, e. g., gradually moving sources, we expect a relatively good generalization of the proposed generic model. On the other hand, the Markov framework could also straightforwardly be extended to include scenarios like this explicitly.

A series of evaluations was conducted based on realistic microphone signals. For the simultaneous localization of 1, 2, or 3 talkers, the LSTM extension, in particular, showed a strong performance. Under challenging conditions, it was superior to all of the considered baseline approaches, including the plain CNN, thereby demonstrating the benefit of incorporating temporal context in the proposed manner. In order to maximally exploit the potential of the approach, it was shown that the use of training data that reflect realistic conditions well is of high importance. A key aspect for achieving this, when synthetically generating microphone signals for training, is the use of realistic source signals, in this case speech.

References

[1] N. Madhu and R. Martin, "Acoustic source localization with microphone arrays," in *Proc. Adv. Digit. Speech Transmiss.*, R. Martin, U. Heute, and C. Antweiler, Eds., New York, USA: John Wiley & Sons, Ltd., 2008, pp. 135–170.

[2] S. Rickard and O. Yilmaz, "On the approximate W-disjoint orthogonality of speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, May 2002, pp. I–529–I–532.

[3] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, no. 4, pp. 320–327, Aug. 1976.

[4] M. Cobos, J. J. Lopez, and D. Martinez, "Two-microphone multi-speaker localization based on a Laplacian mixture model," *Digit. Signal Process.*, vol. 21, no. 1, pp. 66–76, 2011.

[5] J. H. DiBiase, "A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays," Ph.D. dissertation, Brown University, Providence, RI, USA, May 2000.

[6] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[7] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984–995, Jul. 1989.

[8] O. Thiergart, W. Huang, and E. A. P. Habets, "A low complexity weighted least squares narrowband DOA estimator for arbitrary array geometries," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2016, pp. 340–344.

[9] T. Kabzinski and E. A. P. Habets, "A least squares narrowband DOA estimator with robustness against phase wrapping," in *Proc. 27th Eur. Signal Process. Conf.*, Sep. 2019, pp. 1–5.

[10] A. Bohlender, A. Spriet, W. Tirry, and N. Madhu, "Least-squares DOA estimation with an informed phase unwrapping and full bandwidth robustness," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2020, pp. 4841–4845.

[11] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, Aug. 2005.

[12] A. Xenaki, P. Gerstoft, and K. Mosegaard, "Compressive beamforming," *J. Acoust. Soc. Amer.*, vol. 136, no. 1, pp. 260–271, 2014.
[13] P. Gerstoft, A. Xenaki, and C. F. Mecklenbräuker, "Multiple and single snapshot compressive beamforming," *J. Acoustical Soc. Amer.*, vol. 138, no. 4, pp. 2003–2014, 2015.
[14] J. R. Jensen, J. K. Nielsen, R. Heusdens, and M. G. Christensen, "DOA estimation of audio sources in reverberant environments," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2016, pp. 176–180.
[15] O. Nadiri and B. Rafaely, "Localization of multiple speakers under high reverberation using a spherical microphone array and the direct-path dominance test," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 10, pp. 1494–1505, Oct. 2014.
[16] P. Pertilä and E. Cakir, "Robust direction estimation with convolutional neural networks based steered response power," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2017, pp. 6125–6129.
[17] Z. Wang, X. Zhang, and D. Wang, "Robust speaker localization guided by deep learning-based time-frequency masking," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 1, pp. 178–188, Jan. 2019.
[18] 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 *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2015, pp. 2814–2818.
[19] P. Pertilä and M. Parviainen, "Time difference of arrival estimation of speech signals using deep neural networks with integrated time-frequency masking," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2019, pp. 436–440.
[20] S. Adavanne, A. Politis, and T. Virtanen, "Direction of arrival estimation for multiple sound sources using convolutional recurrent neural network," in *Proc. 26th Eur. Signal Process. Conf.*, Sep. 2018, pp. 1462–1466.
[21] S. Chakrabarty and E. A. P. Habets, "Broadband DOA estimation using convolutional neural networks trained with noise signals," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, Oct. 2017, pp. 136–140.
[22] S. Chakrabarty and E. A. P. Habets, "Multi-speaker DOA estimation using deep convolutional networks trained with noise signals," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 1, pp. 8–21, Mar. 2019.
[23] S. Adavanne, A. Politis, J. Nikunen, and T. Virtanen, "Sound event localization and detection of overlapping sources using convolutional recurrent neural networks," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 1, pp. 34–48, 2019.
[24] Q. Li, X. Zhang, and H. Li, "Online direction of arrival estimation based on deep learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 2616–2620.
[25] S. Kapka and M. Lewandowski, "Sound source detection, localization and classification using consecutive ensemble of CRNN models," in *Proc. Workshop Detection Classification Acoust. Scenes Events*, 2019, pp. 119–123.
[26] A. Politis, S. Adavanne, and T. Virtanen, "A dataset of reverberant spatial sound scenes with moving sources for sound event localization and detection," in *Proc. Workshop Detection Classification Acoust. Scenes Events*, 2020, pp. 165–169.
[27] S. Chakrabarty and E. A. P. Habets, "Time-frequency masking based online multi-channel speech enhancement with convolutional recurrent neural networks," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 4, pp. 787–799, Aug. 2019.
[28] A. van den Oord *et al.*, "Wavenet: A generative model for raw audio," 2016, *arXiv:1609.03499*.
[29] A. van den Oord *et al.*, "Wavenet: A. generative model for raw audio," in *Proc. 9th ISCA Speech Synth. Workshop*, 2016, pp. 125–125.
[30] E. A. P. Habets, "RIR generator," https://github.com/ehabets/RIR-Generator, Accessed: Nov. 4, 2020.
[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations*, 2015, pp. 1–13.
[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.
[33] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 448–456.
[34] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn., Ser.*, 2010, pp. 807–814.
[35] J. S. Garofolo *et al.*, "TIMIT acoustic-phonetic continuous speech corpus LDC93S1," Linguistic Data Consortium, Philadelphia, PA, USA, 1993. [Online]. Available: https://catalog.ldc.upenn.edu/LDC93S1
[36] G. Pirker, M. Wohlmayr, S. Petrik, and F. Pernkopf, "A pitch tracking corpus with evaluation on multipitch tracking scenario," in *Proc. 12th Annu. Conf. Int. Speech Commun. Assoc.*, 2011, pp. 1509–1512.
[37] P. Kabal, "TSP speech database," McGill Univ., Montreal, Quebec, Canada, Tech. Rep., 2002. Accessed: Apr. 3, 2021. [Online]. Available: http://www-mmsp.ece.mcgill.ca/Documents/Data/TSP-Speech-Database/TSP-Speech-Database.pdf
[38] E. A. P. Habets and S. Gannot, "Generating sensor signals in isotropic noise fields," *J. Acoust. Soc. Amer.*, vol. 122, no. 6, pp. 3464–3470, 2007.
[39] miniDSP, "UMA-16 USB microphone array," Accessed: Nov. 4, 2020. [Online]. Avilable: https://www.minidsp.com/products/usb-audio-interface/uma-16-microphone-array
[40] European Telecommunications Standards Institute, "Speech processing, transmission and quality aspects (STQ); speech quality performance in the presence of background noise; part 1: Background noise simulation technique and background noise database," Tech. Rep., ETSI EG 202 396-1, 2008.
[41] International Organization for Standardization, "Acoustics - application of new measurement methods in building and room acoustics," ISO 18233:2006, Geneva, Switzerland, Standard, Jun. 2006.