Deep Learning Model Chaining for DOA Estimation Using Sparse Arrays

Mirza A. Haider, Saidur R. Pavel, and Yimin D. Zhang

Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 19122, USA
{mirza.asif.haider, pavel.saidur, ydzhang}@temple.edu

***Abstract*—Deep learning techniques have shown superior capability and performance for direction-of-arrival (DOA) estimation, especially in challenging scenarios, e.g., when the input signal-to-noise ratio is low or when only few snapshots are available. In this paper, we introduce a comprehensive deep neural network framework for sparse array-based DOA estimation. This is achieved by estimating the interpolated data covariance matrix of the sparse array followed by DOA estimation through a series of connected convolutional neural network models. The proposed approach incorporates a combined loss function comprising three individual loss terms, namely, mean squared error loss of the interpolated covariance matrices, that of the denoised covariance matrices, and binary cross-entropy loss of the estimated DOAs. The strategic combination of these loss functions significantly enhances the robustness and performance of DOA estimation, especially for detecting more sources than the number of sensors. Simulation results clearly demonstrate the advantages of the proposed chained deep neural network models for sparse array-based DOA estimation compared to existing models.**

***Index Terms*—Sparse array, machine learning, DOA estimation, deep neural network, interpolated covariance matrix.**

I. Introduction

Accurate direction-of-arrival (DOA) estimation finds a broad range of applications in wireless communication networks, radar systems, biomedical imaging, and defense surveillance [1]–[4]. In millimeter wave (mmWave) communication networks and massive multiple-input multiple-output (MIMO) systems, DOA estimation plays an increasingly important role in channel estimation and capacity analysis [5]–[8]. While uniform linear arrays (ULAs) are widely adopted due to the Nyquist sampling theorem, they suffer from a limited array aperture, a low number of degrees of freedom (DOFs), and high sensitivity to mutual coupling [9]–[11]. In contract, sparse arrays and sparse MIMO systems are attractive to enable enhanced DOA and channel estimation as a result of large array aperture, increased number of DOFs, and reduced mutual coupling effects [12]–[17].

In particular, by exploiting the difference coarray concept, a well-designed sparse array with M sensors can detect up to $\mathcal{O}(M^2)$ sources. Some sparse array designs, such as the nested array [13] and the maximum inter-element spacing constraint (MISC) array [10], can attain consecutive lags spanning the entire array aperture, whereas other sparse designs, such as the coprime array [14], the coprime array with displaced subarrays (CADiS) [15], and the optimized non-redundant array with minimum array aperture (ONRA) [18], contain holes in their difference coarrays. For such arrays, subspace-based DOA estimation methods utilizes only the uniform DOFs corresponding to the contiguous segment of the difference coarray [19]. To address this issue, DOA estimation with compressed sensing-based algorithms can utilize the full DOFs offered by all unique auto-correlation lags [15], [20], [21]. These methods assume that the signal DOAs fall into the discretized grid. When this assumption does not hold, additional processing is needed to perform off-grid DOA estimation [22], [23]. In addition, Coarray interpolation techniques fill holes in the difference coarray to enable gridless subspace-based DOA estimation and full-aperture use [24], [25]. However, their performance degrades under low snapshots or poor signal-to-noise ratio (SNR) and becomes computationally expensive for large arrays or dense grids.

Recent advancements in machine learning present promising alternatives for DOA estimation, utilizing techniques such as neural networks (NNs) [26], convolutional neural networks (CNNs) [27], [28], transformers [29], [30], reinforcement learning [31]–[33], algorithm unrolling [34], [35], and hybrid deep learning models [36], [37]. In [38], a stacked neural network exploiting denoising autoencoders was developed to predict a statistically improved version of the sampled matrix, which is then used for DOA estimation in virtually filled ULAs. Study [39] focused on a compression framework to reduce data traffic volume in a distributed array composed of multiple subarrays for DOA estimation, where each subarray utilized a neural network to compress its covariance matrix before transmitting it to a fusion center. At the fusion center, another neural network was employed to estimate the signal DOAs. A Mamba-based state-space modeling approach for time-varying DOA tracking was introduced in [40], leveraging Mamba's efficient state representation to capture and preserve long-term temporal dependencies in dynamic scenarios. In [41], a CNN was designed and tested with various alternative inputs (proxy spectra) to evaluate its DOA detection performance. The deep learning framework in [42] employs two neural networks that use the sample covariance matrix of a ULA as their input feature. Under the assumption that the

This work was supported in part by the National Science Foundation (NSF) under Grant No. ECCS-2236023 and in part by the Air Force Office of Scientific Research (AFOSR) under Grant No. FA9550-23-1-0255.

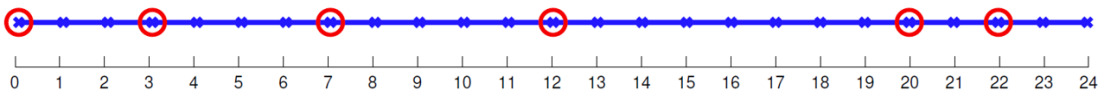


Fig. 1: ONRA array structure where red circles are the position of sensors on the array.

number of sources is smaller than the number of sensors, the model jointly performs DOA estimation and covariance reconstruction.

In this paper, we propose a comprehensive deep-learning chained framework for underdetermined DOA estimation using sparse arrays, where the number of sources is higher than the number of sensors. In such scenarios, given the nonlinear relationship between the sparse covariance matrix and signal DOAs, incorporating more detailed information about the input data into the network becomes increasingly important to enhance the precision of DOA estimates. The proposed approach takes a sampled sparse covariance matrix as input and sequentially estimates signal DOAs through interpolation and denoising.

The proposed framework consists of three stages. First, a deep neural network (DNN) predicts an interpolated covariance matrix from the sparse covariance matrix input. Next, an encoder DNN refines this matrix by removing noise. Finally, the denoised covariance matrix is fed into a classification network to generate DOA labels. Different loss functions are used at each stage, namely, mean squared error (MSE) for interpolation and refinement, and binary cross-entropy (BCE) for classification. In the proposed scheme, task-specific loss functions enable a modular learning design in which each block is both interpretable and independently trainable, thereby simplifying optimization and improving the reliability of the full DOA estimation pipeline. Additionally, separately implementing the sub-tasks of interpolation, denoising, and classification improves specialization for faster convergence, whereas their integration leads to more accurate DOA estimation. Simulation results demonstrate that the proposed chain model can accurately resolve more sources than sensors with high resolution and outperform existing interpolation-based techniques.

Notations: We denote vectors with lower-case bold characters and matrices with upper-case bold characters. Specifically, $\mathbf{I}_N$ represents the $N \times N$ identity matrix. Notations $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ are used to respectively indicate the transpose and Hermitian transpose of a matrix or a vector. Additionally, $\mathbb{E}[\cdot]$ denotes the expected value, $p(\cdot)$ represents probability, and $\mathrm{diag}(\cdot)$ constructs a diagonal matrix from a vector. The Frobenius norm of a matrix $\mathbf{A}$ is denoted as $\|\mathbf{A}\|_{\mathrm{F}}$. Finally, $\mathbb{C}^{M\times N}$ refers to the complex space with dimensions $M \times N$.

II. System Model

Consider a sparse linear array comprising $\bar{M}$ omnidirectional elements, located at $\mathbb{X} = \{p_0, p_1, \cdots, p_{\bar{M}-1}\}\lambda/2$, where λ denotes the signal wavelength. $p_0 = 0$ indicates the reference sensor position, whereas p_i, $i = 1, \cdots, \bar{M}-1$, takes an integer value. As an example, Fig. 1 illustrates a total of M =25 sensor positions on a half-wavelength grid, given as $\mathbb{U} = \{0, 1, \cdots, 24\}\lambda/2$. Among these, $\bar{M} = 6$ elements are physically present and are arranged in an ONRA configuration, positioned at $\mathbb{X} = \{0, 3, 7, 12, 20, 22\}\lambda/2$.

When L uncorrelated narrowband signals impinge from far field with distinct arrival angles θ_l, $l = 1, \cdots, L$, the baseband received signal vector at sampling time $t \in \{1, ..., T\}$ is described as

$$\bar{\mathbf{x}}(t) = \sum_{l=1}^{L} \bar{\mathbf{a}}(\theta_l) s_l(t) + \mathbf{n}(t) = \bar{\mathbf{A}}\mathbf{s}(t) + \mathbf{n}(t), \tag{1}$$

where $\bar{\mathbf{a}}(\theta_l) = [e^{-j\pi p_0 v_l}, e^{-j\pi p_1 v_l}, \cdots, e^{-j\pi p_{\bar{M}} v_l}]^{\mathrm{T}} \in \mathbb{C}^{\bar{M}\times 1}$ represents the steering vector of the $\bar{M}$-sensor physical array with $v_l = \sin(\theta_l)$, $\mathbf{s}(t)$ is the vector of the L received signals, and $\mathbf{n}(t)$ is the additive white Gaussian noise vector. We denote $\bar{\mathbf{A}} = [\bar{\mathbf{a}}(\theta_1), ..., \bar{\mathbf{a}}(\theta_L)]$ as the $\bar{M} \times L$ manifold matrix.

Define $\tilde{\mathbf{x}}(t)$ as an $M \times 1$ augmented array obtained from $\bar{\mathbf{x}}(t)$, i.e.,

$$\tilde{\mathbf{x}}(t) = \mathbf{U}\bar{\mathbf{x}}(t) = \mathbf{U}(\bar{\mathbf{A}}\mathbf{s}(t) + \mathbf{n}(t)) = \tilde{\mathbf{A}}\mathbf{s}(t) + \tilde{\mathbf{n}}(t), \tag{2}$$

where

$$[\mathbf{U}]_{m,\bar{m}} = \begin{cases} 1, & \text{if } p_{\bar{m}} = m, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

with $m = 0, 1, \ldots, M-1$ referring to the sensor position defined in $\mathbb{U}$ and $\bar{m} = 0, 1, \ldots, \bar{M}-1$ indexing the physical sensors. In addition, $\tilde{\mathbf{A}} = \mathbf{U}\bar{\mathbf{A}}$ is defined in the augmented sensor positions, and the elements corresponding to missing sensor positions are set to zero. The $M \times M$ covariance matrix of the augmented sparse array is represented as

$$\mathbf{R}_{sp} = \mathbb{E}[\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^{\mathrm{H}}(t)] = \tilde{\mathbf{A}}\mathbf{R}_s\tilde{\mathbf{A}}^{\mathrm{H}} + \sigma_n^2\mathbf{U}\mathbf{U}^{\mathrm{H}}, \tag{4}$$

where $\mathbf{R}_s = \mathrm{diag}(\sigma_1^2, \sigma_2^2, \cdots, \sigma_L^2)$ with $\sigma_l^2 = \mathbb{E}(|s_l(t)|^2)$ denoting the average power of the lth signal.

In practice, the covariance matrix is unknown, so we estimate the sample covariance matrix as

$$\tilde{\mathbf{R}}_{sp} = \frac{1}{T}\sum_{t=1}^{T} \tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^{\mathrm{H}}(t), \tag{5}$$

where T is the number of available snapshots.

III. Deep Learning Chain Model for DOA Estimation

The proposed DOA estimation approach based on a deep learning chain model is illustrated in Fig. 2. The deep learning chain model consists of three sequential sub-networks, each targeting a specific objective. The first sub-network interpolates the sparse sample covariance matrix $\tilde{\mathbf{R}}_{sp}$ to reconstruct a complete covariance matrix $\mathbf{R}_{ip}$. The second sub-network is trained to refine the resulting covariance matrix to more closely

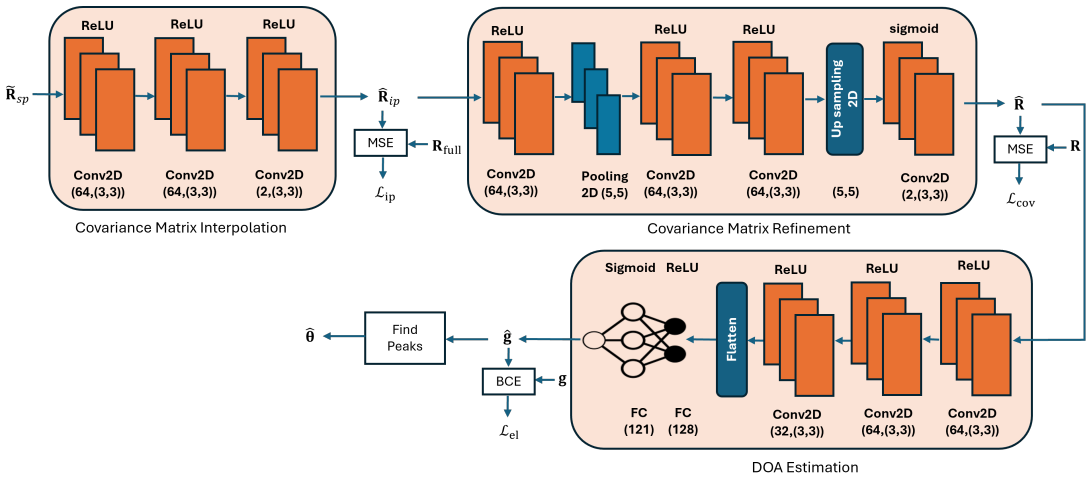


Fig. 2: Architecture of the deep learning chain model for DOA estimation.

approach the noise-free covariance matrix. The third sub-network estimates the signal DOAs by producing an output vector indicating the probability of having a signal at each discretized angle. These three sub-networks are separately described in Sections III-A through III-C.

A key innovation of this model lies in its chained network structure, where the first two sub-networks enable the direct use of the sparse covariance matrix, which is interpolated and refined to closely approximate the true, noise-free covariance matrix. During training, the covariance reconstruction, refinement, and DOA estimation sub-networks are jointly optimized. As detailed in Section III-D, their respective loss functions are minimized together to enhance the overall performance. As such, the proposed approach improves the DOA estimation performance and reduces the required training complexity.

A. Stage I: Covariance Matrix Interpolation

Because the sample covariance matrix, $\tilde{\mathbf{R}}_{sp}$, is sparse with missing entries, we feed it to the first-stage sub-network to perform covariance matrix interpolation. The real and imaginary parts of the complex-valued $\tilde{\mathbf{R}}_{sp}$ are separated and concatenated to form a two-channel input. This two-channel input is passed through two convolutional layers, each using 64 filters with a kernel size of (3,3). These convolutional layers are designed to extract and refine features from the sparse covariance matrix. In the final layer, two filters of size $(3,3)$ are used to match the dimension of the output interpolated covariance matrix, $\hat{\mathbf{R}}_{ip}$, which is then used as the input to the second stage. For supervised training, we require the ground-truth covariance matrix corresponding to a fully populated array. To obtain this, we first synthesize the received data vectors $\mathbf{x}(t) = \sum_{l=1}^{L} \mathbf{a}(\theta_l)s_l(t) + \mathbf{n}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t)$, where $\mathbf{a}(\theta_l) \in \mathbb{C}^{M\times 1}$ represents the steering vector and $\mathbf{A} \in \mathbb{C}^{M\times L}$ is the array manifold matrix for the fully populated sensor set $\mathbb{U}$. Using the standard array manifold model with generated DOAs, source signals, and noise realizations, we simulate a high number of $\dot{T}$ snapshots of the data matrix, $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(\dot{T})] \in \mathbb{C}^{M\times \dot{T}}$. The resulting full covariance matrix is then computed as

$$\mathbf{R}_{\text{full}} = \frac{1}{\dot{T}} \sum_{t=1}^{\dot{T}} \mathbf{x}(t)\mathbf{x}^{\text{H}}(t). \tag{6}$$

Because the data are synthesized, we can set $\dot{T} \gg T$ to ensure that $\mathbf{R}_{\text{full}}$ is an accurate, low-variance estimate that serves as ground truth during training.

The MSE loss at this stage, defined between the output of the first sub-network $\hat{\mathbf{R}}_{ip}$ and the full covariance matrix $\mathbf{R}_{\text{full}}$, is given as

$$\mathcal{L}_{\text{ip}} = \frac{1}{B} \sum_{b=1}^{B} \left\| \hat{\mathbf{R}}_{ip}^{(b)} - \tilde{\mathbf{R}}_{\text{full}}^{(b)} \right\|_{\text{F}}^{2}, \tag{7}$$

where B is the number of samples in each batch and $\tilde{\mathbf{R}}_{\text{full}}^{(b)}$ is the two-channel real-valued form of the bth full covariance matrix. This loss function is employed to evaluate the error between the interpolated covariance matrix and the true covariance matrix of the full ULA, ensuring accurate interpolation.

B. Stage II: Covariance Matrix Denoising and Refinement

In the second stage, the interpolated covariance matrix $\hat{\mathbf{R}}_{ip}$ generated in the first stage serves as the input. It passes through an initial convolutional layer with 64 filters with a kernel size of (3,3). The feature maps are then downsampled using a Max-Pooling layer with a pool size of (5,5) to reduce the dimensionality while preserving important features. Subsequently, two convolutional layers, each with 16 filters and a kernel size of (3,3), further refine the feature maps. An upsampling layer with a size of (5,5) restores the original dimensions of the feature maps by enlarging them using nearest-neighbor interpolation. The final convolutional layer has two filters with a kernel size of (3,3) and outputs $\hat{\mathbf{R}}$, a two-channel real-valued estimate of the noise-free covariance matrix for the full ULA. To provide the corresponding ground-truth target, the complex noise-free full-array covariance matrix is generated as

$$\mathbf{R} = \frac{1}{\dot{T}} \sum_{t=1}^{\dot{T}} \mathbf{A}\mathbf{s}(t)\mathbf{s}^{\text{H}}(t)\mathbf{A}^{\text{H}}. \tag{8}$$

This matrix represents the noise-free counterpart of the noisy covariance $\mathbf{R}_{\text{full}}$. For training, the target matrix $\tilde{\mathbf{R}}^{(b)}$ is formed by converting $\mathbf{R}^{(b)}$ into a two-channel real-valued representation and stacking its real and imaginary parts, matching the format of $\hat{\mathbf{R}}^{(b)}$.

The MSE loss for this refinement stage is then defined as

$$\mathcal{L}_{\text{cov}} = \frac{1}{B}\sum_{b=1}^{B}\left\|\hat{\mathbf{R}}^{(b)} - \tilde{\mathbf{R}}^{(b)}\right\|_{\text{F}}^{2}, \tag{9}$$

which quantifies the discrepancy between the estimated and target noise-free covariance matrices and drives the network to effectively denoise the interpolated covariance matrix.

C. Stage III: DOA Estimation

The final stage takes the denoised covariance matrix $\hat{\mathbf{R}}$ as its input, which undergoes a series of transformations, starting with two convolutional layers having 64 filters with kernel size (3,3), followed by another convolutional layer having two filters with size (3,3). These layers are responsible for extracting relevant features needed for DOA estimation. The feature maps are then flattened into a one-dimensional vector, which is accessed by two fully connected (FC) layers to produce the DOA labels, $\hat{\mathbf{g}} \in \mathbb{C}^{F_{\text{angle}}\times 1}$, where F_{angle} is the number of DOA grids.

Denoting $\mathbf{g}^{(b)} = [g_1^{(b)}, \cdots, g_{F_{\text{angle}}}^{(b)}]^{\text{T}}$ as the true labels corresponding to the F_{angle} angles in the bth sample, the BCE function describes the error between the predicted DOA labels and the true labels, defined as

$$\begin{aligned}\mathcal{L}_{\text{BCE}} = -\frac{1}{BF_{\text{angle}}}\sum_{b=1}^{B}\sum_{f=1}^{F_{\text{angle}}}\Big[g_f^{(b)}\log\Big(p(\hat{g}_f^{(b)})\Big) \\ +(1-g_f^{(b)})\log\Big(1-p(\hat{g}_f^{(b)})\Big)\Big],\end{aligned} \tag{10}$$

where $\hat{g}_f^{(b)}$ is output corresponding to the fth label in the bth sample. In post-processing, the predicted DOA labels are analyzed to identify peaks, which correspond to the estimated DOA angles, $\hat{\boldsymbol{\theta}}$.

D. Joint Loss Function Minimization

An online training model is used to train a dataset. Here, the DNN pipeline is optimized using a total loss that aggregates the three sub-network losses:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{ip}} + \mathcal{L}_{\text{cov}} + \mathcal{L}_{\text{BCE}}. \tag{11}$$

IV. Simulation Results

In this study, the synthetic dataset is generated using the signal model outlined in Section II with $\bar{M} = 6$ and $M = 25$. This simulated dataset contains a wide range of angles, input SNR values, and numbers of snapshots, enabling comprehensive testing. Specifically, the angle separation $\Delta\theta$ is set to $\{3^\circ, 5^\circ, 10^\circ\}$, the input SNR values range from -20 dB to 20 dB in 5-dB increments, and the number of snapshots N is selected from set $\{50, 100, 200, 300, 400, 600, 800, 1000\}$. For all cases, $\dot{T} = 10,000$ samples are used.

The models are trained on the entire dataset and evaluated using a separate, independently generated dataset, spanning a range of specific angle separations, input SNR values, and snapshot counts. Here, the training dataset contains all possible signal angles between -60° and 60° for eight uncorrelated signals. The grid scale was set to 1°, creating a total of $F_{\text{angle}} = 121$ angle values for the on-grid configuration. The proposed network undergoes training over 100 epochs with a batch size of $B = 64$, using the Adam optimizer at a learning rate of 0.05. This training process attempts to minimize the final BCE loss and improve the accuracy of predictions. The total DNN training experiment was conducted using Google Colab with L4 GPU.

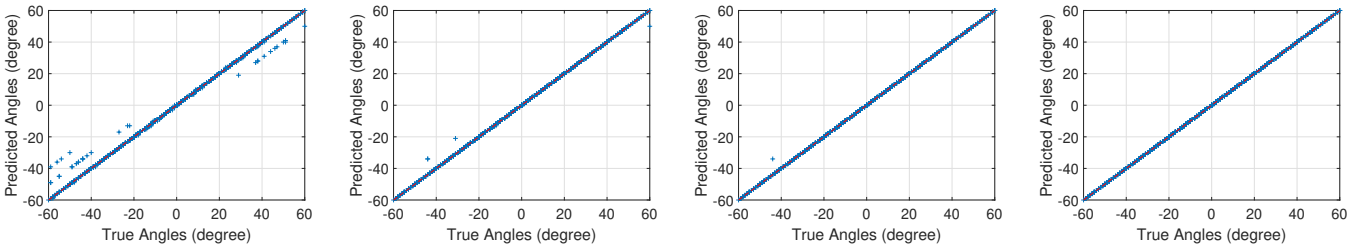
In the simulation, we randomly generated $1,000$ received data samples for each snapshot value, totaling 2.865×10^5 samples (number of angular separations $\times$ number of input SNR values $\times$ number of snapshot variations $\times$ number of samples). From the generated dataset, 80% of the data are used for training, and the remaining 20% are used for testing. Among the training data, 10% is used for validation purposes. The angle prediction performance is evaluated using the root-mean-square error (RMSE) of the estimated angles, defined as

$$\text{RMSE} = \sqrt{\frac{1}{QL}\sum_{q=1}^{Q}\sum_{l=1}^{L}\left(\hat{\boldsymbol{\theta}}_l^{(q)} - \boldsymbol{\theta}_l^{(q)}\right)^2}, \tag{12}$$

where $Q = 200$ is the number of independent trials dedicated for the set of each snapshot.

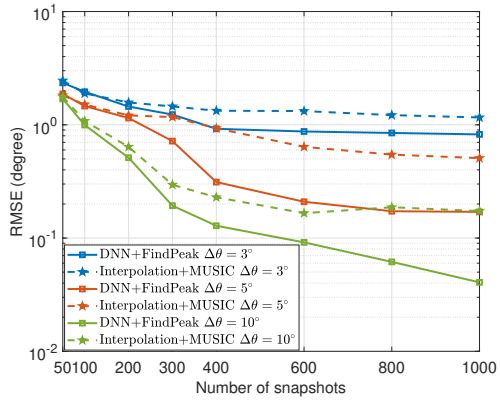
The scatter plots in Fig. 3 illustrate the estimated DOAs of eight sources separated by 10° using the proposed network, where the input SNR is 10 dB, and snapshot sizes of 50, 300, 600, and 1,000 are considered. Generally, as the number of snapshots increases, the DOA estimation accuracy improves. In Fig. 3(a), with 50 snapshots, there is a prominent scatter around the true diagonal trajectory, indicating higher estimation errors. Increasing the number of snapshots to 300 reduces these errors, as shown in Fig. 3(b). With further increasing to 600 and $1,000$ snapshots, Figs. 3(c) and 3(d), show that the detected angles align near perfectly with the true angles. This demonstrates the DNN's improved precision with more snapshots. This trend verifies that the network's DOA estimation performance becomes more reliable as the number of snapshots increases.

Fig. 4(a) shows the RMSE performance of the estimated DOAs versus the number of snapshots, ranging from 50 to 1,000, for three source separations of $\Delta\theta = \{3^\circ, 5^\circ, 10^\circ\}$. The comparison includes the proposed method (denoted as "DNN+FindPeak") and a baseline method using interpolation followed by MUSIC for peak detection (denoted as "Interpolation+MUSIC"). For the 3°-separation case, both methods exhibit relatively high RMSE at low snapshot counts. While the Interpolation+MUSIC method shows a gradual improvement as the number of snapshots increases, the DNN+FindPeak method achieves lower RMSE throughout the range being computed. However, both methods show limited improvement after the number of snapshots exceeds 400, indicating a floor

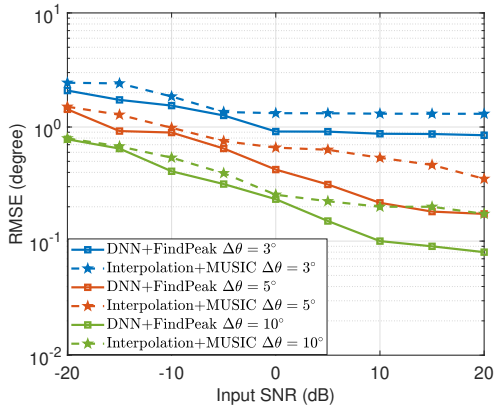


(a) 50 snapshots (b) 300 snapshots (c) 600 snapshots (d) 1,000 snapshots

Fig. 3: Scatter plots of estimated DOAs for varying numbers of snapshots (8 sources, 10° separation, and 10 dB input SNR).



(a) RMSE versus the number of snapshots (input SNR = 10 dB)



(b) RMSE versus the input SNR (600 snapshots)

Fig. 4: RMSE for varying snapshots and SNR values.

due to the small angular separation and grid resolution. In the 5°-separation scenario, the RMSE trends begin to diverge more clearly. The proposed DNN+FindPeak method rapidly improves with more snapshots and outperforms the Interpolation+MUSIC approach when more than 300 snapshots are available. The gap between the two methods remains consistent, showcasing the DNN's better generalization and noise-suppression capability in such scenario with a moderate angular separation. The 10°-separation case presents more pronounced advantage of the proposed method. As the number of snapshots increases, the RMSE drops significantly for DNN+FindPeak, reaching below 0.1° when the number of snapshots exceeds 600. In contrast, the RMSE of the Interpolation+MUSIC method improves more slowly and remains higher as compared to the proposed approach.

Fig. 4(b) illustrates the RMSE performance of DOA estimation as the input SNR varies from -20 dB to 20 dB for the same three angular separations of $\Delta\theta = \{3^\circ, 5^\circ, 10^\circ\}$. Overall, the RMSE decreases as the input SNR increases for both the proposed DNN+FindPeak and the Interpolation+MUSIC methods. Overall, the proposed DNN-based approach consistently achieves lower RMSE results across all SNR levels and angular separations. For the 3°-separation case, both methods perform badly at the low SNR. However, as the input SNR increases, the RMSE decreases slightly for both methods, and the DNN+FindPeak method achieves better performance, particularly in mid-to-high SNR ranges. Nonetheless, both methods appear to plateau, suggesting that finer angular separation presents a challenge due to limited resolution. In the 5°-separation case, the DNN+FindPeak method shows a prominent performance advantage. Its RMSE steadily falls with increasing SNR, whereas the Interpolation+MUSIC method improves in a slower manner and saturates when the input SNR reaches 0 dB. This indicates the DNN's greater robustness to noise and improved resolution in medium separation settings. For 10° separation, the advantage of the proposed method is more prominent. It consistently surpasses the Interpolation+MUSIC approach, with the RMSE falling below 0.2° as the input SNR exceeds 20 dB. Meanwhile, the RMSE of the Interpolation+MUSIC method levels off around 0.5°, displaying limited advancement when the input SNR exceeds 10 dB. This widening gap clearly underlines the superior noise resilience and high-resolution DOA estimation capability of the proposed DNN-based framework, particularly at the high SNR range with wider angular separations.

V. Conclusion

This paper proposed a novel chained DNN model for DOAs estimation exploiting sparse arrays. By integrating multiple sub-networks with combined loss functions, the model ensures accurate full covariance matrix recovery and high-resolution DOA estimations while enabling faster training. Simulation results confirmed that the proposed deep learning model can accurately resolve a high number of sources, even when the number of sources exceeds that of sensors. The model significantly outperforms well-known interpolation algorithms.

REFERENCES

[1] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Process. Mag.*, vol. 13, no. 4, pp. 67–94, 1996.

[2] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Wiley, 2002.

[3] T. E. Tuncer and B. Friedlander, *Classical and Modern Direction-of-Arrival Estimation*. Academic Press, 2009.

[4] M. G. Amin and Y. D. Zhang, "DOA estimation of nonstationary signals," in *Academic Press' Library in Signal Processing*, S. Theodoridis and R. Chellappa, Eds. Academic Press, 2013, pp. 110–127.

[5] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Vehi. Tech.*, vol. 67, no. 9, pp. 8549–8560, 2018.

[6] R. Shafin, L. Liu, J. Zhang, and Y.-C. Wu, "DOA estimation and capacity analysis for 3-D millimeter wave massive-MIMO/FD-MIMO OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 6963–6978, 2016.

[7] K. Hassan, M. Masarra, M. Zwingelstein, and I. Dayoub, "Channel estimation techniques for millimeter-wave communication systems: Achievements and challenges," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1336–1363, 2020.

[8] M. A. Haider, Y. D. Zhang, and E. Aboutanios, "ISAC system assisted by RIS with sparse active elements," *EURASIP J. Adv. Signal Process.*, vol. 2023, no. 20, pp. 1–22, 2023.

[9] Z. Ye, J. Dai, X. Xu, and X. Wu, "DOA estimation for uniform linear array with mutual coupling," *IEEE Trans. Aerosp. Electronic Syst.*, vol. 45, no. 1, pp. 280–288, 2009.

[10] Z. Zheng, W.-Q. Wang, Y. Kong, and Y. D. Zhang, "MISC array: A new sparse array design achieving increased degrees of freedom and reduced mutual coupling effect," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1728–1741, 2019.

[11] M. G. Amin, Ed., *Sparse Arrays for Radar, Sonar, and Communications*. Wiley-IEEE Press, 2024.

[12] R. Hoctor and S. Kassam, "The unifying role of the coarray in aperture synthesis for coherent and incoherent imaging," *Proc. IEEE*, vol. 78, no. 4, pp. 735–752, 1990.

[13] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4167–4181, 2010.

[14] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 573–586, 2011.

[15] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1377–1390, 2015.

[16] C. Zhou, Y. Gu, Y. D. Zhang, and Z. Shi, "Sparse array interpolation for direction-of-arrival estimation," in *Sparse Arrays for Radar, Sonar, and Communications*, M. G. Amin, Ed. Wiley-IEEE Press, 2024, pp. 41–74.

[17] S. Sun and Y. D. Zhang, "4D automotive radar sensing for autonomous vehicles: A sparsity-oriented approach," *IEEE J. Sel. Top. Signal Process.*, vol. 15, no. 4, pp. 879–891, 2021.

[18] A. Ahmed and Y. D. Zhang, "Generalized non-redundant sparse array designs," *IEEE Trans. Signal Process.*, vol. 69, pp. 4580–4594, 2021.

[19] P. Pal and P. P. Vaidyanathan, "Coprime sampling and the MUSIC algorithm," in *Proc. Digital Signal Process. and Signal Process. Edu. Meeting (DSP/SPE)*, 2011, pp. 289–294.

[20] Y. D. Zhang, M. G. Amin, and B. Himed, "Sparsity-based DOA estimation using co-prime arrays," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2013, pp. 3967–3971.

[21] M. Guo, Y. D. Zhang, and T. Chen, "DOA estimation using compressed sparse array," *IEEE Trans. Signal Process.*, vol. 66, no. 15, pp. 4133–4146, 2018.

[22] Z. Yang, L. Xie, and C. Zhang, "Off-grid direction of arrival estimation using sparse Bayesian inference," *IEEE Trans. Signal Process.*, vol. 61, no. 1, pp. 38–43, 2013.

[23] P. Chen, Z. Cao, Z. Chen, and X. Wang, "Off-grid DOA estimation using sparse Bayesian learning in MIMO radar with unknown mutual coupling," *IEEE Trans. Signal Process.*, vol. 67, no. 1, pp. 208–220, 2019.

[24] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5956–5971, 2018.

[25] S. Liu, Z. Mao, Y. D. Zhang, and Y. Huang, "Rank minimization-based Toeplitz reconstruction for DoA estimation using coprime array," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2265–2269, 2021.

[26] A. Barthelme and W. Utschick, "DOA estimation using neural network-based covariance matrix reconstruction," *IEEE Signal Process. Lett.*, vol. 28, pp. 783–787, 2021.

[27] T. N. T. Nguyen, W.-S. Gan, R. Ranjan, and D. L. Jones, "Robust source counting and DOA estimation using spatial pseudo-spectrum and convolutional neural network," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 2626–2637, 2020.

[28] G. K. Papageorgiou, M. Sellathurai, and Y. C. Eldar, "Deep networks for direction-of-arrival estimation in low SNR," *IEEE Trans. Signal Process.*, vol. 69, pp. 3714–3729, 2021.

[29] X. Zhao, J. Atli Benediktsson, Y. Yang, K.-S. Chen, and M. Örn Úlfarsson, "Exploring transformer-based direction-of-arrival estimation over sea surface: A BERT approach with physics-based loss function," *IEEE Trans. Geosci. Remote. Sens.*, vol. 62, pp. 1–13, 2024.

[30] W. Wang, L. Zhou, K. Ye, H. Sun, and S. Hong, "A DOA estimation method based on an improved transformer model for uniform linear arrays with low SNR," *IET Signal Process.*, vol. 2024, no. 6666395, pp. 1–25, 2024.

[31] A. M. Ahmed, A. A. Ahmad, S. Fortunati, A. Sezgin, M. S. Greco, and F. Gini, "A reinforcement learning based approach for multitarget detection in massive MIMO radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 5, pp. 2622–2636, 2021.

[32] M. S. R. Pavel and Y. D. Zhang, "Reinforcement learning-based weak signal detection from compressed measurements in massive MIMO systems," in *Proc. Asilomar Conf. Signals, Syst. Computers*, 2023, pp. 438–442.

[33] L. Xu, S. Sun, Y. D. Zhang, and A. P. Petropulu, "Reconfigurable beamforming for automotive radar sensing and communication: A deep reinforcement learning approach," *IEEE J. Sel. Areas Sensors*, vol. 1, pp. 124–138, 2024.

[34] R. Zheng, H. Liu, S. Sun, and J. Li, "Deep learning based computationally efficient unrolling IAA for direction-of-arrival estimation," in *Proc. European Signal Process. Conf. (EUSIPCO)*, 2023, pp. 730–734.

[35] Y. Hu, S. Sun, and Y. D. Zhang, "Model-based learning for DOA estimation with one-bit single-snapshot sparse arrays," *IEEE J. Sel. Top. Signal Process.*, in press.

[36] Z.-M. Liu, C. Zhang, and S. Y. Philip, "Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections," *IEEE Trans. Antennas Propagat.*, vol. 66, no. 12, pp. 7315–7327, 2018.

[37] W. Liu, "Super resolution DOA estimation based on deep neural network," *Scientific Reports*, vol. 10, no. 1, p. 19859, 2020.

[38] G. K. Papageorgiou and M. Sellathurai, "Fast direction-of-arrival estimation of multiple targets using deep learning and sparse arrays," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2020, pp. 4632–4636.

[39] S. R. Pavel and Y. D. Zhang, "Neural network-based compression framework for DOA estimation exploiting distributed array," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2022, pp. 4943–4947.

[40] S. R. Pavel, M. A. Haider, Y. D. Zhang, Y. Ding, D. Shen, K. Pham, and G. Chen, "Time-varying direction-of-arrival estimation exploiting mamba network," in *Proc. IEEE Radar Conf.*, 2025, pp. 1–5.

[41] P. Kulkarni and P. Vaidyanathan, "Feature engineering for DOA estimation using a convolutional neural network, for sparse arrays," in *Proc. Asilomar Conf. Signals Syst. Comput.*, 2021, pp. 246–250.

[42] A. M. Alam, C. O. Ayna, S. Biswas, J. T. Rogers, J. E. Ball, and A. C. Gurbuz, "Deep learning-based direction-of-arrival estimation with covariance reconstruction," in *Proc. IEEE Radar Conf.*, 2024, pp. 1–6.