Deep Learning-Based DOA Estimation via Grid Search Approach for Unknown and Variable Number of Signals

Qian Xu, Yulong Gao, *Senior Member, IEEE*, Ruoyu Zhang, *Member, IEEE*,
Jinshan Kong, and Chau Yuen, *Fellow, IEEE*

***Abstract*—Deep learning-based direction-of-arrival (DOA) estimators have displayed superior performance to classical model-based DOA estimators in many scenarios. However, most deep learning-based DOA estimators can only be used for a fixed signal number or pre-trained signal number ranges. This makes deep learning-based DOA estimators not practical and flexible for real-world applications. In this paper, we propose a deep learning-based spatial spectrum estimator (DeepSSE). It demonstrates remarkable generalization capability with an unknown and variable number of signals, not limited by the maximum signal number present in the training data. This capability is achieved by leveraging a novel angular grid search (AGS) process and the asymmetric loss (ASL). The AGS, which emulates that of the multiple signal classification (MUSIC) algorithm, enables DeepSSE to detect as many target DOAs as possible. And the ASL enhances the performance of DeepSSE under multiple signals by overcoming the imbalance of positive and negative angular grids. Furthermore, we introduce the optimal sub-pattern assignment (OSPA) metric to DOA estimation for the first time, to address the lack of performance evaluation metrics in scenarios with a variable number of signals. Extensive numerical results demonstrate that DeepSSE outperforms other DOA estimators across various scenarios, especially when the signal numbers are far beyond the maximum number in the training set.**

***Index Terms*—Array signal processing, direction-of-arrival (DOA) estimation, deep learning, multilabel classification.**

Received 26 January 2025; revised 18 April 2025; accepted 26 June 2025. Date of publication 1 July 2025; date of current version 29 December 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62171163 and 62201266, in part by Aeronautical Science Foundation (2023Z018077001), and in part by National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research & Development Programme FCP-NTU-RG-2024-025. The associate editor coordinating the review of this article and approving it for publication was A. Sciarrone. *(Corresponding author: Yulong Gao.)*

Qian Xu, Yulong Gao, and Jinshan Kong are with the School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China (e-mail: xuq3196@outlook.com; ylgao@hit.edu.cn; 23S005043@stu.hit.edu.cn).

Ruoyu Zhang is with the Key Laboratory of Near-Range RF Sensing ICs & Microsystems, Ministry of Education, School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: ryzhang19@njust.edu.cn).

Chau Yuen is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: chau.yuen@ntu.edu.sg).

Digital Object Identifier 10.1109/TCCN.2025.3584889

I. Introduction

DIRECTION of Arrival (DOA) estimation is an active area of array signal processing, finding applications across diverse fields such as automotive radar, mobile user or millimeter wave localization, and drone surveillance [1]. The advent of future wireless communication technologies, particularly joint sensing and communication, has further fueled research interest in DOA estimation [2], [3]. However, increasingly complex electromagnetic environments pose significant challenges to DOA estimation. For instance, the number of signals can be variable and unknown, making DOA estimation more difficult.

Over the past few decades, subspace-based estimators like MUSIC [4], Root-MUSIC [5], and ESPRIT [6] have achieved remarkable success in DOA estimation. Compressed sensing-based estimators like l_1-SVD [7], JLZA-DOA [8], and L_1-SRACV [9] have also been successful. These algorithms are based on an accurate mathematical description of signals and arrays, so we can refer to these algorithms as *model-driven* approaches [10]. Model-driven approaches, however, often suffer from sensitivity to the assumptions of array and signal models and high computational complexity. In particular, most of them heavily rely on the accurate estimation of the number of signals, which often leads to a significant decline in estimation accuracy in scenarios with a variable number of signals.

In recent years, numerous deep learning (DL)-based DOA estimators have emerged, which have superior robustness and performance compared to model-driven methods. In these studies, the DOA estimation problem is typically treated either as a regression task over DOA values [11], [12], [13], [14] or a classification task over angular grids [15], [16], [17], [18]. These methods learn the mapping between the signal and the DOA from the training data. We can refer to these algorithms as *data-driven* approaches. Data-driven methods have shown superior performance and robustness in certain scenarios. However, limited by training data and network architecture, data-driven methods are sensitive to variations in the number of signals. Most data-driven methods are based on the assumption that the number of signals is fixed, which severely restricts the application of the algorithms in practical scenarios, where the number of signals is often dynamic.

Some studies have attempted to apply DL-based DOA estimators to scenarios with a variable number of signals.

TABLE I
COMPARISON BETWEEN THIS WORK AND THE RELATED ONES

Method	Reference	Network Architecture	Training strategy & Loss Function	Limitations
Spatial spectrum subregion division	[19], [20]	Parallel DNNs/CNNs	Regression (MSE[1]) or classification (CE[2])	Cannot handle multiple signals in the same subregion
Signal-number-specific network models	[21], [22]	Mostly Multiple CNNs	Mostly regression with MSE	High training costs and runtime complexity
Angular grid multi-label classification	[23], [24]	Mostly CNN-based	Multi-label classification with BCE[3]	Limited accuracy; restricted to trained signal numbers
DNN-aided inference	[26], [27], [29]	RNNs/CNNs to substitute mathematical operations	Mostly Regression with RMSPE[4]	Limited to trained signal numbers; better tailored for static environments
Our proposed DeepSSE	-	CNNs with learnable angular grid search	Multi-label classification with ASL[5]	-

[1] MSE: Mean Squared Error; [2] CE: Cross Entropy; [3] BCE: Binary Cross Entropy; [4] RMSPE: Root Mean Squared Periodic Error; [5] ASL: Asymmetric Loss.

The authors of [19], [20] divide the spatial spectrum into multiple angular subregions. They estimate the DOA of each subregion separately using parallel neural networks. However, this approach cannot handle multiple signals existing in the same angular subregion. Other approaches train different models for different numbers of signals. In [21], independent network models are trained for different numbers of multipath signals. Similarly, [22] pre-stores network models trained for different signal numbers in a database. The appropriate network is then selected based on the number of signals during runtime. These methods lead to a significant increase in both training costs and runtime complexity. Moreover, the estimation accuracy strictly depends on the accuracy of the signal number estimation. Other methods such as those in [23], [24] model the DOA estimation as a multi-label classification problem and train the neural network with the binary cross-entropy loss. This method works well within the trained range of signal numbers. However, when the number of signals exceeds that in the training data, these algorithms' robustness decreases noticeably. Furthermore, because of the sparsity of the spatial spectrum, binary cross-entropy loss is not suitable for DOA estimation.

Recently, *model-based deep learning* methods have garnered significant attention in the DOA estimation field. Some of these are applicable to multi-signal scenarios. These algorithms generally outperform data-driven DL-based ones in terms of estimation accuracy [25], [26], [27], [28], [29] and computational complexity [30]. However, most methods simply replace certain intermediate mathematical steps with neural networks [25], [26], [27], [28], [29]. This approach is referred to as *DNN-aided inference* [31]. For example, [26] used Recurrent Neural Networks (RNN) to replace the covariance matrix computation in MUSIC. They also employed Deep Neural Networks (DNN) for noise subspace computation. The authors of [27] adopted an autoencoder as a front-end module that provides denoising and decoherence capabilities. These methods face several limitations. First, they usually incorporate matrix operations like singular value decomposition (SVD) into neural networks, which can cause gradient instability during backpropagation. When training data contains significant variations in signal parameters, the training process often becomes unstable. Thus, these methods work best in static and unchanging environments. Furthermore, they lack specific mechanisms for variable signal numbers. Although [26] and [29] designed a retroactively trained signal number classifier for varying signal numbers, they remain limited by training data and struggle with fluctuating signal scenarios.

To overcome the limitations, we propose the deep learning-based spatial spectrum estimator (DeepSSE), which is motivated by the model-based deep learning paradigm named *neural building blocks* [31]. Unlike existing approaches, DeepSSE handles scenarios with unknown and variable signal numbers. It works effectively even when signal numbers far exceed those in the training data. Our approach achieves this through two key innovations: First, DeepSSE's architecture incorporates parameter-learnable sub-networks that mimic the angular grid search flow of the MUSIC algorithm. This specifically designed mechanism enables DeepSSE to accurately detect all DOAs by searching at each angular grid. Second, we use the asymmetric loss (ASL) [32] to address the spatial spectrum sparsity. ASL can effectively balance the significant imbalance between positive angular grids (equal to actual DOAs) and negative angular grids. This strategy lets DeepSSE focus on searching for positive angular grids, while not strictly focusing on negative grids. These two innovations significantly enhance DeepSSE's performance in scenarios with varying signal numbers. By incorporating model-driven design with data-driven methods, and optimizing the spatial spectrum sparsity representation during training, DeepSSE provides a robust solution for DOA estimation in dynamic electromagnetic environments.

To the best of our knowledge, this is the first work to propose a DL-based DOA estimator that can be used in scenarios with a variable number of signals far exceeding the number present in the training data. The comparison between this work and the related ones is presented in Table I, and the main contributions of this paper are as follows:

- We propose DeepSSE, a DOA estimator based on model-based deep learning. It mimics the angular grid search process of MUSIC and handles unknown, variable numbers of signals. DeepSSE achieves high DOA estimation accuracy even when the number of signals far exceeds signals present in the training data, without prior knowledge of the signal number. It also significantly outperforms existing model-driven and data-driven DOA estimators. DeepSSE is designed for easy training and strong generalization. It is robust to variations in signal numbers, the number of snapshots, signal-to-noise ratio (SNR), and mismatched signal and array models. This makes DeepSSE suitable for DOA estimation under diverse scenarios.

- We use the asymmetric loss (ASL) to enhance the multi-signal DOA estimation performance of DeepSSE. ASL addresses a key training challenge in DOA estimation: the positive angular grids (equal to the target DOAs) are far fewer than the negative angular grids. ASL balances these two kinds of angular grids by focusing more on positive angular grids during the training process. It lets DeepSSE focus on searching for the positive angular grids, while not paying much attention to the negative angular grids. To the best of our knowledge, this is the first work that considers the unbalanced characteristic of training labels and applies ASL to a DL-based DOA estimator.
- We investigate the optimal sub-pattern assignment (OSPA) metric for evaluating DOA estimators in scenarios with a variable number of signals. OSPA measures the performance of multi-signal estimation more reasonably and effectively than other metrics used in prior works. It provides a benchmark for future performance evaluations of multi-signal DOA estimation.
- The performance of DeepSSE is evaluated over an extensive set of simulated experiments. We compare the performance of DeepSSE with state-of-the-art DL-based and model-driven DOA estimators in terms of estimation accuracy and computational complexity. The experiments are carried out under various conditions of signal numbers, snapshot numbers, and SNRs, and different kinds of model mismatch are considered. Experimental results demonstrate that DeepSSE possesses strong generalization ability and maintains excellent performance in varying scenarios. A comprehensive analysis of the influence of hyperparameters and different sub-networks is also presented.

The rest of this paper is organized as follows. In Section II, we introduce the signal model. In Section III, the proposed DOA estimator and the chosen ASL are described in detail. In Section IV, the performance metrics of DOA estimators in multi-signal scenarios are discussed, and the simulation results are shown. Finally, Section V concludes this paper and discusses the potential future research directions.

II. Signal Model

Consider a uniform linear array composed of M antenna elements with an inter-element spacing of l. Assuming K narrowband, uncorrelated far-field signals impinge on the array, which originate from angles $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_K\}$. At each time instance t from T snapshots (i.e., $t \in \{1, 2, \ldots, T\}$), the signals originating from K sources can be represented as $\mathbf{S}(t) = \left[s_1(t)\ s_2(t) \cdots s_K(t)\right]^T \in \mathbb{C}^{K\times 1}$. Then, the array received signal is modeled as

$$\mathbf{X}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{S}(t) + \mathbf{N}(t). \quad (1)$$

In (1), $\mathbf{X}(t) = \left[x_1(t)\ x_2(t) \cdots x_M(t)\right]^T \in \mathbb{C}^{M\times 1}$ is the signal received by M antennas of the array. $\mathbf{N}(t) = \left[n_1(t)\ n_2(t) \cdots n_M(t)\right]^T \in \mathbb{C}^{M\times 1}$ is the additive white Gaussian noise received by M antennas.

The matrix $\mathbf{A}(\boldsymbol{\theta}) = \left[\mathbf{a}(\theta_1)\ \mathbf{a}(\theta_2) \cdots \mathbf{a}(\theta_K)\right] \in \mathbb{C}^{M\times K}$ is the manifold matrix of the array, which consists of steering vectors $\mathbf{a}(\theta_k)$. If the first array element is taken as the reference element whose position is taken as the origin, and the carrier frequency of the K signals is f, then the kth steering vector can be expressed as

$$\mathbf{a}(\theta_k) = \left[1, e^{j2\pi f \frac{l}{c}\sin(\theta_k)}, \ldots, e^{j2\pi f \frac{(M-1)l}{c}\sin(\theta_k)}\right]^T. \quad (2)$$

If the number of snapshots T is sufficiently large and the SNR is high, we can accurately estimate the covariance matrix of the array received signal as

$$\begin{aligned}\mathbf{R} &= \mathbb{E}\left[\mathbf{X}(t)\mathbf{X}^H(t)\right] \\ &= \mathbf{A}(\boldsymbol{\theta})\mathbf{R}_S\mathbf{A}^H(\boldsymbol{\theta}) + \sigma^2\mathbf{I}_N,\ t = 1, 2, \ldots, T, \end{aligned} \quad (3)$$

where $\mathbf{R}_S = \mathbb{E}[\mathbf{S}(t)\mathbf{S}^H(t)]$ is the signal covariance matrix, σ^2 is the noise power, and $\mathbf{I}_N$ is the identity matrix of size M.

III. Proposed DeepSSE

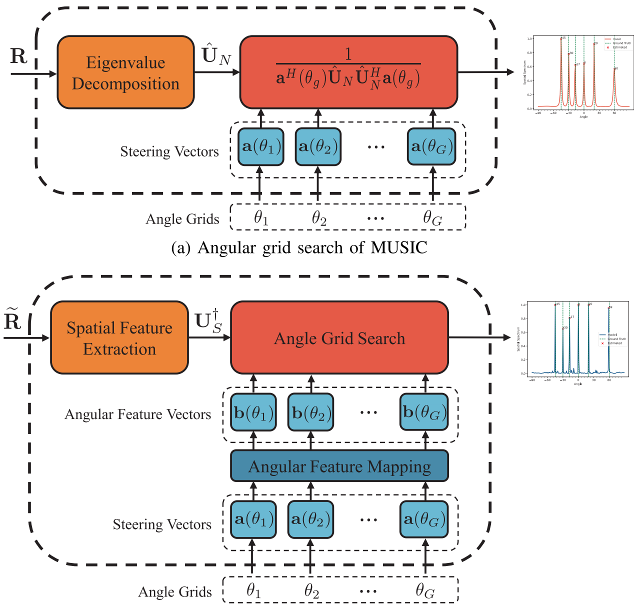
A. Motivation

DL-based DOA estimators depend heavily on training data, making them difficult to apply when signal numbers vary widely. It can be seen from (3) that, SNR and snapshot variations only cause ambiguity in $\mathbf{R}$. However, changes in the number of signals alter the characteristics of $\mathbf{R}_S$. This poses a challenge to the generalizability of DL-based DOA estimators. Most DL-based DOA estimation algorithms assume either a fixed number of signals, or that all variations in the signal numbers were considered during the training phase [15], [20], [23], [24], [25], [26], [27], [28], [29], [30]. This limitation restricts DL-based estimators in real-world applications with widely varying signal numbers.

However, the classical model-driven MUSIC [4] algorithm can effectively estimate all DOAs by finding orthogonality between steering vectors and noise subspace across angular grids. We refer to this process as angular grid search. As shown in Fig. 1a, the flow diagram of the MUSIC algorithm mainly consists of three blocks:

- *Noise subspace extraction:* Perform SVD on the covariance matrix $\mathbf{R}$ to extract the noise subspace $\hat{\mathbf{U}}_N$.
- *Steering vector calculation:* Generate corresponding steering vectors $\mathbf{a}(\theta_g)$ for different angular grids $\theta_g, g = 1, 2, \ldots, G$ via (2).
- *Spatial spectrum construction:* Search for the orthogonality between $\hat{\mathbf{U}}_N$ and $\mathbf{a}(\theta_g)$ at different angular grids via $p_g = 1/(\mathbf{a}^H(\theta_g)\hat{\mathbf{U}}_N\hat{\mathbf{U}}_N^H\mathbf{a}(\theta_g))$.

To handle a variable number of signals, the proposed DeepSSE utilizes a learnable angular grid search approach consisting of three specialized sub-networks. Each


$\mathbf{R}$
Eigenvalue Decomposition
$\hat{\mathbf{U}}_N$
$\frac{1}{\mathbf{a}^H(\theta_g)\hat{\mathbf{U}}_N\hat{\mathbf{U}}_N^H\mathbf{a}(\theta_g)}$
Steering Vectors
$\mathbf{a}(\theta_1)$ $\mathbf{a}(\theta_2)$ $\cdots$ $\mathbf{a}(\theta_G)$
Angle Grids
θ_1 θ_2 $\cdots$ θ_G
$\widetilde{\mathbf{R}}$
Spatial Feature Extraction
$\mathbf{U}_S^\dagger$
Angle Grid Search
Angular Feature Vectors
$\mathbf{b}(\theta_1)$ $\mathbf{b}(\theta_2)$ $\cdots$ $\mathbf{b}(\theta_G)$
Angular Feature Mapping
Steering Vectors
$\mathbf{a}(\theta_1)$ $\mathbf{a}(\theta_2)$ $\cdots$ $\mathbf{a}(\theta_G)$
Angle Grids
θ_1 θ_2 $\cdots$ θ_G

(a) Angular grid search of MUSIC

(b) Angular grid search of DeepSSE.

Fig. 1. Angular grid search flow diagram of MUSIC algorithm and DeepSSE.

sub-network performs a specific computation analogous to the three building blocks of MUSIC. As shown in Fig. 1b, DeepSSE consists of three sub-networks:

- *Spatial feature extraction (SFE) sub-network f_{SFE}.* This CNN-based sub-network extracts the pseudo-signal subspace (PSS) $\mathbf{U}_S^\dagger$ from the covariance matrix $\mathbf{R}$ of the array signals with $\mathbf{U}_S^\dagger = f_{SFE}(\mathbf{R})$. As calculating the reciprocal in MUSIC's angular grid search poses difficulties for gradient computation, DeepSSE utilizes the strong correlation between signal subspace and steering vectors instead of noise space to perform angular grid search.
- *Angular feature mapping (AFM) sub-network f_{AFM}.* The AFM consists of fully connected layers. It maps the steering vector $\mathbf{a}(\theta_g) \in \mathbb{C}^{M\times 1}$ of grid θ_g to a higher-dimensional angular feature vector (AFV) $\mathbf{b}(\theta_g) \in \mathbb{R}^{d\times 1}$. It can be expressed as $\mathbf{b}(\theta_g) = f_{AFM}(\mathbf{a}(\theta_g))$. The angular feature encoding of AFM enhances the correlation between $\mathbf{b}(\theta_g)$ and $\mathbf{U}_S^\dagger$. It also makes DeepSSE more robust to array imperfections.
- *Angular grid search (AGS) sub-network f_{AGS}.* The AGS sub-network searches for the target DOAs by evaluating the correlation between the PSS $\mathbf{U}_S^\dagger$ and the AFV $\mathbf{b}(\theta_g)$ of different angular grids. This process can be expressed as $p_g = f_{AGS}(\mathbf{b}(\theta_g), \mathbf{U}_S^\dagger)$. The estimated spatial spectrum peaks at the grid points where the true DOAs lie. We build the AGS sub-network with cross-attention mechanism, as the scaled dot product attention $softmax(\mathbf{b}(\theta_g)\mathbf{U}_S^{\dagger T})\mathbf{U}_S^\dagger$ can be seen as a soft version of $\mathbf{b}^H(\theta_g)\mathbf{U}_S^\dagger\mathbf{U}_S^{\dagger H}\mathbf{b}(\theta_g)$. The attention focusing process also enables dynamic energy focusing, which can achieve a certain degree of suppression of noise and interference components in the spatial features $\mathbf{U}_S^\dagger = f_{SFE}(\mathbf{R})$.

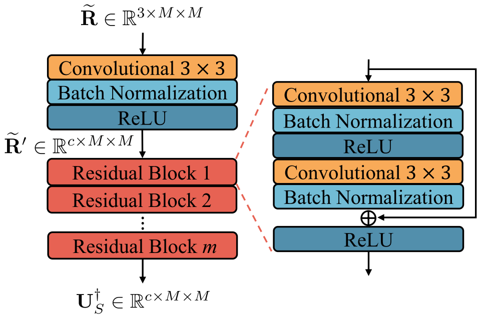

$\widetilde{\mathbf{R}} \in \mathbb{R}^{3\times M\times M}$
Convolutional 3 × 3
Batch Normalization
ReLU
$\widetilde{\mathbf{R}}' \in \mathbb{R}^{c\times M\times M}$
Residual Block 1
Residual Block 2
Residual Block m
$\mathbf{U}_S^\dagger \in \mathbb{R}^{c\times M\times M}$
Convolutional 3 × 3
Batch Normalization
ReLU
Convolutional 3 × 3
Batch Normalization
ReLU

Fig. 2. The architecture of the spatial feature extraction sub-network.

Furthermore, to enhance the estimation performance of DeepSSE under multi-signal conditions, we employ the asymmetric loss (ASL) to address the unbalanced labels caused by the sparsity of spatial spectrum. The ASL loss function effectively prevents the training process from overemphasizing negative angular grids and neglecting the detection of positive angular grids.

B. The Architecture of Proposed DeepSSE

DeepSSE consists of three sub-networks, the spatial feature extraction sub-network, the angular feature mapping sub-network, and the angular grid search sub-network. These three sub-networks work together to perform the angular grid search based DOA estimation.

To make the DOA estimator more robust to variations in the number of snapshots and signal features, the covariance matrix $\mathbf{R}$ of the array received signals in (3) is used as input to extract spatial features. To eliminate the influence of signal and noise amplitudes, the covariance matrix $\mathbf{R}$ is normalized as $\bar{\mathbf{R}} = \mathbf{R}/\|\mathbf{R}\|_F$, and then shaped into a $3 \times M \times M$ real matrix $\widetilde{\mathbf{R}}$. In particular, the three channels of $\widetilde{\mathbf{R}}$ are the real part, imaginary part, and phase entries of the normalized $\bar{\mathbf{R}}$, which can be denoted as

$$\widetilde{\mathbf{R}}_{1,:,:} = \Re(\bar{\mathbf{R}}), \tag{4}$$

$$\widetilde{\mathbf{R}}_{2,:,:} = \Im(\bar{\mathbf{R}}), \tag{5}$$

$$\widetilde{\mathbf{R}}_{3,:,:} = \arctan\big(\Im(\bar{\mathbf{R}})/\Re(\bar{\mathbf{R}})\big). \tag{6}$$

1) Spatial Feature Extraction: The SFE sub-network extracts PSS from the normalized real covariance matrix $\widetilde{\mathbf{R}}$. To avoid the backpropagation instability caused by SVD, we employ a purely neural network approach to extract the PSS $\mathbf{U}_S^\dagger$. The effectiveness of CNNs in extracting spatial features from the covariance matrix in DOA estimation has been demonstrated by numerous previous studies [18], [21], [23]. We select a residual CNN architecture to improve training efficiency and mitigate the vanishing gradient problem [33].

The structure of the SFE sub-network is shown in Fig. 2. The $\widetilde{\mathbf{R}}$ serves as input to the SFE sub-network to extract the desired PSS $\mathbf{U}_S^\dagger$. The first CNN block consists of a convolutional layer with c kernels, a batch normalization layer and a ReLU activation. It is used to expand $\widetilde{\mathbf{R}} \in \mathbb{R}^{3\times M\times M}$

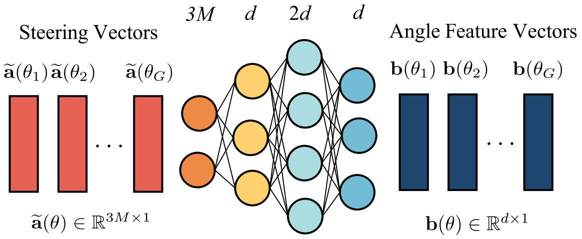

Steering Vectors
$3M$
d
$2d$
d
Angle Feature Vectors
$\widetilde{\mathbf{a}}(\theta_1)$ $\widetilde{\mathbf{a}}(\theta_2)$ $\widetilde{\mathbf{a}}(\theta_G)$
$\mathbf{b}(\theta_1)$ $\mathbf{b}(\theta_2)$ $\mathbf{b}(\theta_G)$
$\widetilde{\mathbf{a}}(\theta) \in \mathbb{R}^{3M\times 1}$
$\mathbf{b}(\theta) \in \mathbb{R}^{d\times 1}$

Fig. 3. The architecture of the angular feature mapping sub-network.

along the channel dimension to $\widetilde{\mathbf{R}}' \in \mathbb{R}^{c\times M\times M}$. This process can be represented by:

$$\widetilde{\mathbf{R}}' = \mathrm{ReLU}\Big(\mathrm{BN}(\mathrm{Conv}(\widetilde{\mathbf{R}}))\Big). \tag{7}$$

The channel dimension of the middle data remains unchanged in the subsequent *m* residual blocks. Each residual block contains two convolutional layers, both with c kernels. All residual blocks share the same structure shown in Fig. 2. The signal flow through these residual blocks can be represented as

$$\mathbf{X}_0 = \widetilde{\mathbf{R}}',\ \mathbf{X}_m = \mathbf{U}_S^{\dagger} \tag{8}$$

$$\mathbf{X}_i = \mathrm{ResBlock}(\mathbf{X}_{i-1}),\ i = 1, 2, \ldots, m, \tag{9}$$

where the calculation inside each residual block can be expressed as

$$\begin{aligned}&\mathrm{Res\,Block}(\mathbf{X}) = \\ &\quad \mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}(\mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}(\mathbf{X}))))) + \mathbf{X}).\end{aligned} \tag{10}$$

The kernel size of all the convolutional layers in the SFE sub-network is set to (3,3), with a stride of 1 and a padding of 1. Batch normalization and a ReLU activation function are applied after each CNN layer. Also, to avoid information loss, no pooling layer is used.

2) Angular Feature Mapping: The AFM sub-network maps the steering vectors corresponding to different angular grids to more feature-rich AFVs. Suppose G angular grids $\{\theta_1, \theta_2, \ldots, \theta_G\}$ are selected, the AFM sub-network maps the steering vector $\mathbf{a}(\theta_g)$ calculated according to (2) to the AFV $\mathbf{b}(\theta_g)$. The angle values are one-dimensional features. Therefore, we build the AFM sub-network using three fully connected layers with dimensions d, $2d$, and d, respectively. Each layer has a ReLU activation function. It can be expressed as

$$\mathbf{b}\big(\theta_g\big) = \mathrm{FC}_3\big(\mathrm{FC}_2(\mathrm{FC}_1(\widetilde{\mathbf{a}}(\theta_g)))\big),\ g = 1, 2, \ldots, G, \tag{11}$$

where $\widetilde{\mathbf{a}}(\theta_g)$ is a $3M \times 1$ vector consisting of the real, imaginary and phase parts of the steering vector $\mathbf{a}(\theta_g)$, namely, $\widetilde{\mathbf{a}}(\theta_g)_{0:M} = \Re(\mathbf{a}(\theta_g))$, $\widetilde{\mathbf{a}}(\theta_g)_{M:2M} = \Im(\mathbf{a}(\theta_g))$ and $\widetilde{\mathbf{a}}(\theta_g)_{2M:3M} = \arctan(\Im(\mathbf{a}(\theta_g))/\Re(\mathbf{a}(\theta_g)))$.

3) Angular Grid Search: The AGS sub-network searches for the target DOAs by evaluating the correlation between the PSS and the AFV at each angular grid. We build the AGS sub-network with cross-attention mechanism. This is because the scaled dot product attention $softmax(\mathbf{b}(\theta_g)\mathbf{U}_S^{\dagger T})\mathbf{U}_S^{\dagger}$ can be seen as a soft version of the PSS-AFV correlation $\mathbf{b}^H(\theta_g)\mathbf{U}_S^{\dagger}\mathbf{U}_S^{\dagger H}\mathbf{b}(\theta_g)$. Also, the feasibility of this correlation search has been demonstrated [34]. Alternatively, the angular grid search process can be described intuitively. In the cross-attention, each AFV is correlated with different blocks of the PSS, some of which contain features related to the target DOAs (the yellow blocks in the PSS shown in Fig. 4). When the gth angular grid θ_g matches a target DOA in $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_K\}$, the gth AFV $\mathbf{b}(\theta_g)$ shows a strong correlation with the blocks corresponding to the target DOAs. This results in a high power for the gth vector $\mathbf{o}_g$ in the cross-attention module's output.

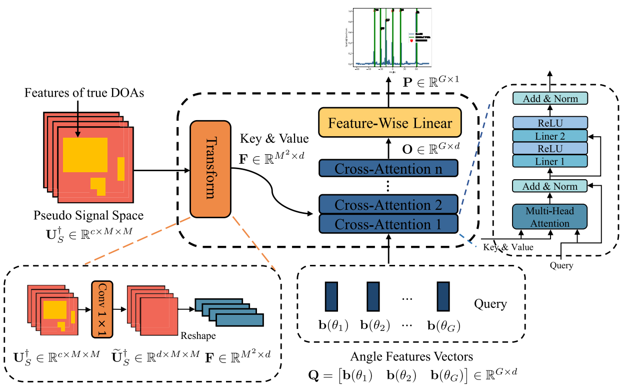

$\mathbf{P} \in \mathbb{R}^{G\times 1}$
Features of true DOAs
Pseudo Signal Space
$\mathbf{U}_S^{\dagger} \in \mathbb{R}^{c\times M\times M}$
Transform
Key & Value
$\mathbf{F} \in \mathbb{R}^{M^2\times d}$
Feature-Wise Linear
$\mathbf{O} \in \mathbb{R}^{G\times d}$
Cross-Attention n
Cross-Attention 2
Cross-Attention 1
Add & Norm
ReLU
Liner 2
ReLU
Liner 1
Add & Norm
Multi-Head Attention
Key & Value
Query
Conv 1 × 1
Reshape
$\mathbf{U}_S^{\dagger} \in \mathbb{R}^{c\times M\times M}$
$\widetilde{\mathbf{U}}_S^{\dagger} \in \mathbb{R}^{d\times M\times M}$
$\mathbf{F} \in \mathbb{R}^{M^2\times d}$
$\mathbf{b}(\theta_1)$ $\mathbf{b}(\theta_2)$ ... $\mathbf{b}(\theta_G)$
Query
Angle Features Vectors
$\mathbf{Q} = \big[\mathbf{b}(\theta_1)\quad \mathbf{b}(\theta_2)\quad \mathbf{b}(\theta_G)\big] \in \mathbb{R}^{G\times d}$

Fig. 4. The architecture of the angular grid search sub-network.

As illustrated in Fig. 4, the PSS $\mathbf{U}_S^{\dagger} \in \mathbb{R}^{c\times M\times M}$ is first passed through a single 2D convolutional layer with a kernel size of (1, 1), and an output channel of d. The output is expanded into a matrix of dimension $d \times M \times M$. It is then reshaped into a matrix $\mathbf{F} \in \mathbb{R}^{M^2\times d}$.

The AGS module contains n cross-attention blocks. Each block consists of a multi-head attention layer and a two-layer feed-forward network (FFN) with a residual connection. This module utilizes the G AFVs as *Query* for the attention operation, namely

$$\mathbf{Q}^T = \big[\mathbf{b}(\theta_1)\ \mathbf{b}(\theta_2)\ \cdots\ \mathbf{b}(\theta_G)\big] \in \mathbb{R}^{d\times G}. \tag{12}$$

Also, to preserve the positional information of elements within the PSS, we employ $\widetilde{\mathbf{F}}$ as the *Key*, which is $\mathbf{F}$ encoded with sinusoidal positional encoding, while $\mathbf{F}$ serves as the *Value*. For more effective correlation extraction, multiple cross-attention blocks can be used. In this case, the output of the preceding cross-attention block serves as the *Query* for the subsequent block. And $\mathbf{F}$ and $\widetilde{\mathbf{F}}$ continue to function as the *Value* and *Key*, respectively. This creates a cascaded n-layer cross-attention operation. The last cross-attention block outputs a matrix

$$\mathbf{O}^T = \big[\mathbf{o}_1\ \mathbf{o}_2\ \cdots\ \mathbf{o}_G\big] \in \mathbb{R}^{d\times G}, \tag{13}$$

where $\mathbf{o}_g \in \mathbb{R}^{d\times 1}$ is a vector corresponding to the output of the g-th angular grid.

If we let $\mathbf{O}_0 = \mathbf{Q}$, $\mathbf{O}_n = \mathbf{O}$, this cross attention process can be expressed as

$$\mathbf{O}_0 = \mathbf{Q},\ \mathbf{O}_n = \mathbf{O} \tag{14}$$

$$\mathbf{O}_i = \mathrm{FFN}\Big(\mathrm{Atten}(\mathbf{O}_{i-1}, \widetilde{\mathbf{F}}, \mathbf{F})\Big),\ i = 1, 2, \ldots, n. \tag{15}$$

To determine whether the g-th angular grid θ_g is present in the spatial features, we use a feature-wise linear layer to project $\mathbf{o}_g$ into a logistic value in $[0,1]$ in the g-th element in the output of DeepSSE $\hat{\mathbf{p}} \in \mathbb{R}^{G\times 1}$. For each $\mathbf{o}_g$, the feature-wise linear layer independently applies a linear transformation to it, which can be expressed as

$$\hat{p}_g = \text{Sigmoid}\Big(\mathbf{W}_g^T \mathbf{o}_g + b_g\Big),\ \ g = 1, 2, \ldots, G, \qquad (16)$$

where $\hat{p}_g$ is the g-th element in $\hat{\mathbf{p}}$. The $\mathbf{W}_g \in \mathbb{R}^{d\times 1}$ and $b_g \in \mathbb{R}$ represent the weight vector and bias of the feature-wise linear layer.

C. Training Strategy and Loss Function

DeepSSE is trained end-to-end in a supervised setting as a multi-label classification problem. The training procedure uses the dataset $\mathbf{D}$ consisting of U training examples $\mathbf{d}_u = \{\mathbf{R}_u(\boldsymbol{\theta}), \mathbf{p}_u(\boldsymbol{\theta})\}, u = 1, 2, \ldots, U$. $\mathbf{R}_u(\boldsymbol{\theta})$ is the covariance matrix corresponding to a set of target DOAs $\boldsymbol{\theta}$ and $\mathbf{p}_u(\boldsymbol{\theta})$ is the corresponding spatial spectrum. If we set G angular grids, then $\mathbf{p}_u(\boldsymbol{\theta})$ will be a G-dimensional vector. If the g-th angular grid is equal to one of the target DOAs in $\boldsymbol{\theta}$, the g-th element in $\mathbf{p}_u(\boldsymbol{\theta})$ will be 1, otherwise 0. For example, if we consider a set of 180 angular grids of 1° resolution in $[-90^\circ, 90^\circ)$ and the target DOAs are $\boldsymbol{\theta} = \{1^\circ, 10^\circ\}$, then $\mathbf{p}_u(\boldsymbol{\theta}) = [0 \cdots 1 \cdots 1 \cdots]$, where 1 appears in the 92st and 101st elements in $\mathbf{p}_u(\boldsymbol{\theta})$. Moreover, $\mathbf{D}$ consists of multiple subsets, each containing signal data under a specific signal number. i.e., $\mathbf{D} = \{\mathbf{D}_1, \mathbf{D}_2, \ldots\}$, and $\mathbf{D}_1$ is the subset corresponding to 1 signal, $\mathbf{D}_2$ is the subset corresponding to 2 signals, and so on.

The BCE is usually used when training a multi-label classification problem over angular grids, which is a common approach in prior works [21], [23], [24]. However, due to the sparsity of the target spatial spectrum, there is an imbalance between the number of positive angular grids and negative angular grids in each training spatial spectrum $\mathbf{p}_u(\boldsymbol{\theta})$. For instance, as shown in Fig. 5, when setting 180 angular grids and the number of target DOAs is 4, we will have 4 positive angular grids and 176 negative angular grids in $\mathbf{p}_u(\boldsymbol{\theta})$. This imbalance would lead to the network training focusing mainly on driving the probabilities of negative angular grids close to 0 but "forgetting" to let the probabilities of positive angular grids approach 1. This imbalance will make it difficult to accurately identify all the target DOAs. However, this issue has never been considered in prior works.

To address this issue, we use the novel ASL [32] to train our DeepSSE, which can be expressed as

$$\text{ASL}(\mathbf{p}, \hat{\mathbf{p}}) = \sum_{g=1}^{G} \mathcal{L}_{ASL}(p_g, \hat{p}_g), \qquad (17)$$

where

$$\begin{aligned}&\mathcal{L}_{ASL}(p_g, \hat{p}_g)\\ &= \begin{cases} -(1-\hat{p}_g)^{\gamma_+} \log(\hat{p}_g), & for \quad p_g = 1 \\ -(p_m)^{\gamma_-} \log(1-p_m), & for \quad p_g = 0 \end{cases},\end{aligned} \qquad (18)$$

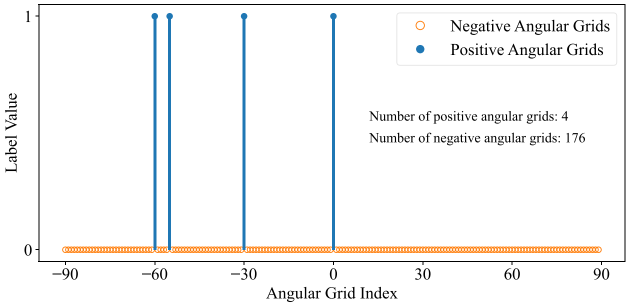


Fig. 5. An example of the unbalanced distribution of positive and negative angular grids in the target spatial spectrum.

where $p_m = \max(\hat{p}_g - \eta_m, 0)$, $p_g \in \{0, 1\}$ is the g-th element in the true spatial spectrum $\mathbf{p}$, and $\hat{p}_g \in [0, 1]$ is the probability of the g-th angular grid being present in the true DOA set $\boldsymbol{\theta}$. γ_+ and γ_- are the focusing parameters for positive and negative angular grids respectively. The smaller the value of γ_+, and the larger the value of γ_-, the greater the weight assigned to positive angular grids in the loss calculation. η_m is the probability margin that we can set, used to ignore negative angular grids with probabilities less than η_m.

It can be seen that ASL is similar to the binary cross entropy (BCE) loss

$$\mathcal{L}_{BCE}(p_g, \hat{p}_g) = \begin{cases} -\log(\hat{p}_g), & for \quad p_g = 1 \\ -\log(1-\hat{p}_g), & for \quad p_g = 0 \end{cases}. \qquad (19)$$

But ASL applies different weights to positive and negative angular grids. When γ_- is greater than γ_+, the gradient of the negative angular grids is much smaller than that of positive grids. Therefore the parameter updates are more influenced by the positive angular grids. In this case, the network's parameters are optimized in a direction that lets the estimated probability of all positive angular grids approach 1, while all negative angular grids are not strictly forced to approach 0. At the same time, p_m represents a probability margin applied to negative angular grids, allowing the network to essentially ignore negative angular grids with a low probability of occurrence. These negative angular grids can be easily predicted correctly and we can pay less attention to them.

IV. Simulation Results

In this section, we present simulation results to illustrate our proposed method. We first discuss the suitable performance evaluation metrics for DOA estimation in the multi-signal scenario. Then we evaluate the performance and computational complexity of the proposed DeepSSE across various setups, and compare it with state-of-the-art DOA estimators. A discussion on hyperparameter analysis and ablation studies of ASL and different sub-networks is also presented.

A. Performance Metrics

In the single-signal DOA estimation scenario, the root mean squared error (RMSE) is a commonly used performance evaluation metric. RMSE is frequently applied in prior studies to evaluate the performance of DOA estimation algorithms [21], [22], [24]. It can be expressed as

$$\mathrm{RMSE}\left(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}\right)=\sqrt{\frac{1}{K}\sum_{i=1}^{K}\left(\theta_i-\hat{\theta}_i\right)^2}. \tag{20}$$

However, RMSE is not suitable for the multi-signal scenario [23], [35]. First, in the multi-signal scenario, it is difficult to establish a one-to-one correspondence between the estimated results and the actual results. Second, it fails when there are cardinality errors. This is the case when the number of estimated DOAs $\hat{K}$ is not equal to the true number of DOAs K.

To address the correspondence problem in multi-signal scenarios, a variant of the Root Mean Square Periodic Error (RMSPE) is used [26], [27], [28], [29]:

$$\mathrm{RMSPE}\left(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}\right)=\min_{\mathbf{P}\in\mathcal{P}}\left(\frac{1}{K}\left\|\mathrm{mod}_{\pi}\left(\boldsymbol{\theta}-\mathbf{P}\hat{\boldsymbol{\theta}}\right)\right\|^2\right)^{\frac{1}{2}}, \tag{21}$$

where mod_{π} denotes the modulo operation with respect to π, and $\mathcal{P}$ is the set of all possible permutations of $\hat{\boldsymbol{\theta}}$. The RMSPE employs a combinatorial optimization algorithm (e.g., the Hungarian method) to find the optimal matching. But the cardinality error is still not considered.

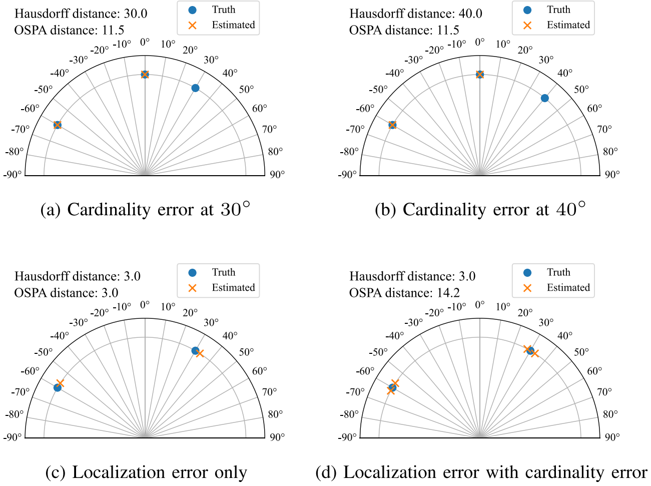
The number of signals $\hat{K}$ in the estimated results $\hat{\boldsymbol{\theta}}$ may not be the same as the actual signal number K ($\hat{K}\lesseqqgtr K$). In such cases, both RMSE and RMSPE are inapplicable. The authors of [23] proposed using the Hausdorff distance as an evaluation metric for multi-signal DOA estimation. It can measure how far two subsets of a metric space are from each other without considering the order and number of values in the subsets. It can be expressed as

$$\mathrm{Hausdorff}\left(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}\right)=$$
$$\max\left\{\max_{\theta\in\boldsymbol{\theta}}\min_{\hat{\theta}\in\hat{\boldsymbol{\theta}}}|\theta-\hat{\theta}|,\max_{\hat{\theta}\in\hat{\boldsymbol{\theta}}}\min_{\theta\in\boldsymbol{\theta}}|\theta-\hat{\theta}|\right\}. \tag{22}$$

However, the Hausdorff distance is not a consistent metric. It is only sensitive to localization errors but insensitive to cardinality errors. The optimal subpattern assignment (OSPA) metric, on the other hand, can balance these two errors [35], [36]. If $K\leq\hat{K}$, the OSPA metric is

$$\mathrm{OSPA}\left(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}\right)=$$
$$\left(\frac{1}{K}\left(\min_{\pi\in\Pi_K}\sum_{i=1}^{K}\min\left(c,|\theta_i-\hat{\theta}_{\pi(i)}|\right)^p+c^p\left(\hat{K}-K\right)\right)\right)^{\frac{1}{p}}. \tag{23}$$

If $\hat{K}>K$, we let $\mathrm{OSPA}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})=\mathrm{OSPA}(\hat{\boldsymbol{\theta}},\boldsymbol{\theta})$. In (23), p is the order which is usually set to $p=2$, and c is the cut-off. A smaller c tends to emphasize localization errors, and a larger c tends to emphasize cardinality errors.



(a) Cardinality error at 30°

(b) Cardinality error at 40°

(c) Localization error only

(d) Localization error with cardinality error

Fig. 6. The difference between the Hausdorff distance and OSPA metric ($p=2,\ c=20$) under different estimation results.

Fig. 6 shows the comparison of OSPA and Hausdorff distance in these two kinds of errors. We can see from Fig. 6a-6b that, given the same omission of one signal, Hausdorff distance is significantly affected by the DOAs of the original signal, while the OSPA metric is more stable. In performance evaluation, we expect the evaluation metric to be consistent. When the number of missed signals is the same, the metric value should also be the same. OSPA fulfills this requirement well. Also, as shown in Fig. 6c-6d, the Hausdorff distance is sensitive to localization errors but insensitive to cardinality errors, especially when the mis-estimated DOAs are close to the true DOAs. The Hausdorff distance struggles to reflect the performance of signal number estimation. In contrast, the OSPA metric can better reflect this error.

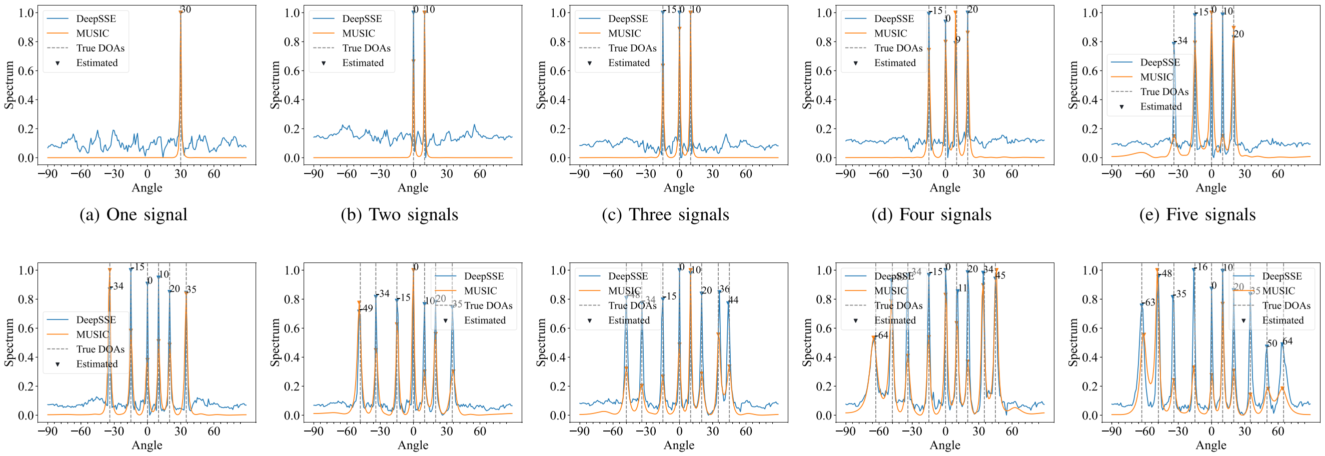
Therefore, in our experiments, we employ the OSPA metric to evaluate the algorithm's performance. We set $p=2$, $c=5$ to maintain a balance between localization and cardinality errors. It is also worth noting that, when $p=2$, the OSPA metric is equivalent to RMSE with a combinatorial optimization algorithm and the missed DOAs are substituted with a constant value. Since RMSE is widely used in the field of DOA estimation, this characteristic further validates the feasibility of applying the OSPA metric to DOA estimation.

B. Simulation Setup

1) Training Dataset Setup: During the training phase, we consider a ULA with $M=16$ antenna elements spaced at half-wavelength distance ($l=\lambda/2$). We consider narrowband, non-coherent signals with different intermediate frequencies (IF) transmitted from different sources. The same signal model is used to train and test different DOA estimators. The mathematical expression of the signals can be represented as

$$s_k(t)=\exp(2\pi(f_k+f_c)t+\phi_k),\ \ k=1,2,\ldots,K. \tag{24}$$

In (24), the carrier frequency of the signals is set to $f_c=$ 10MHz. The intermediate frequencies f_k and phase ϕ_k are randomly selected from uniform distributions $\mathcal{U}(0,500000)$ (Hz) and $\mathcal{U}(0,2\pi)$, respectively. The signals are sampled at the intermediate frequency with a sampling frequency of 2.5MHz and 300 sample points. Then we simulate the array received signal according to signal model (1) at different SNRs of $\{-10,-9,-8,-7,-6,-5,-4,-3,-2,-1,0,1,2,3,4,5\}$ dB.



(a) One signal (b) Two signals (c) Three signals (d) Four signals (e) Five signals

(f) Six signals (g) Seven signals (h) Eight signals (i) Nine signals (j) Ten signals

Fig. 7. Spatial spectrum estimated by DeepSSE (trained with a maximum of 4 signals) and MUSIC at -5 dB SNR using $T = 300$ snapshots under different number of signals. The spatial spectrum estimated by MUSIC is normalized to [0, 1] for better comparison with DeepSSE.

To train the DL-based DOA estimator, our generated training dataset $\mathbf{D}$ includes simulated data with varying numbers of signals. In particular, we set 180 angular grids spanning from -90° to 89° with a 1° spacing. Then we generate pairs of DOAs from all possible combinations of different angular grids under different numbers of signals (i.e., we can get ${}^3C_{180}$ pairs of DOAs under 3 signals). We generate datasets $\mathbf{D}$ with a mixed number of signals in $\{1, 2, 3, 4\}$ for training. That is, $\mathbf{D} = \{\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \mathbf{D}_4\}$. There are ${}^1C_{180} = 180$ training examples as described in Section III-C at each SNR in $\mathbf{D}_1$, ${}^2C_{180} = 16,110$ training examples at each SNR in $\mathbf{D}_2$. To keep the training dataset from becoming excessively large, in $\mathbf{D}_3$ and $\mathbf{D}_4$, we randomly select 37,500 training examples from all combinations at each SNR.

2) Algorithm Setup: As detailed in Section III, the proposed DeepSSE is comprised of three neural building blocks. In the SFE sub-network, we first expand the channel of $\widetilde{\mathbf{R}}$ to $c = 32$ and then pass it through $m = 4$ residual blocks. In the AFM sub-network, we set the dimension of AFV $\mathbf{b}(\theta)$ to $d = 128$. So the dimensions of the three fully connected layers in the AFM are set to 128, 256, and 128, respectively. In the AGS sub-network, we use $n = 2$ cross attention blocks. Within each cross-attention block, the number of heads in the multi-head attention layer is set to 8, and the dimensions of subsequent linear fully connected layers are set to 512 and 128, respectively. Finally, the dropout rate is set to 0.05.

During training, the ASL function discussed in (18) is used, and its hyperparameters are set according to the recommendations in [32], namely $\gamma_+ = 1$, $\gamma_- = 4$, and $\eta_m = 0.05$. DeepSSE is trained using the Adam optimizer [37] with a learning rate of $\mu = 0.001$.

We also compare DeepSSE[1] to other model-driven and DL-based DOA estimators:

- The DA-MUSIC[2] algorithm with a retroactively trained signal number classifier [26].
- The IQ-ResNet proposed in [24] trained under multiple-sources settings.
- The CNN-based DOA estimator proposed in [23] trained under multiple sources settings.
- The classical model-driven MUSIC[3] algorithm [4].
- The classical model-driven ESPRIT algorithm [6].

All DL-based algorithms are trained using the same training dataset $\mathbf{D}$ and the same Adam optimizer with the same learning rate $\mu = 0.001$ for 100 epochs.

C. DOA Estimation Under Varying Number of Signals

To demonstrate DeepSSE's DOA estimation capability in scenarios with a variable number of signals, we maintain the same array and source parameters as the training data described in Section IV-B1, set the SNR at –5 dB and the number of snapshots at 300, and then generate synthetic data with the number of signals varying from 3 to 10. We used both MUSIC and DeepSSE to estimate the spatial spectrum, and the experimental results are shown in Fig. 7. It can be seen that although the maximum number of signals in the training data was 4, DeepSSE can still accurately estimate DOAs of all signals even when the number of signals is large. By employing ASL, the negative grids in spatial spectrum maintain a low power but not strictly approach 0. The positive grids get peaks quite closely approach 1.

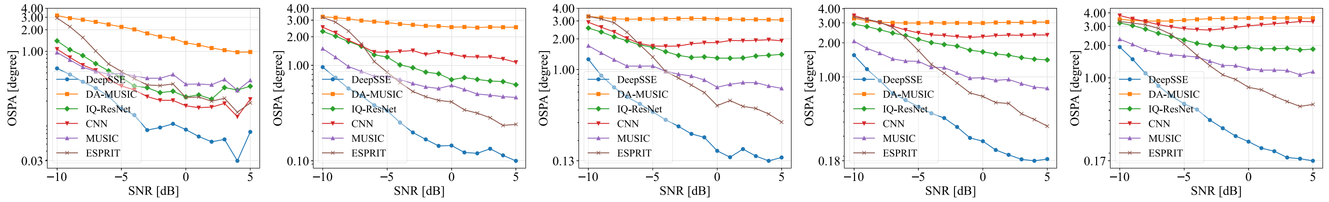
D. Performance With Number of Signals Known

1) Performance Evaluation Under Different Senerios: We performed DOA estimation using different algorithms under various numbers of signals, different SNRs, and different numbers of snapshots. We assumed prior knowledge of the number of signals. For DeepSSE, CNN, IQ-ResNet, and MUSIC algorithms, the estimation results are determined by the K largest peaks higher than 0.2 in the spatial spectrum. The DA-MUSIC algorithm directly utilizes the known number

[1] The source code of our method can be obtained from https://github.com/zhiim/deepsse.

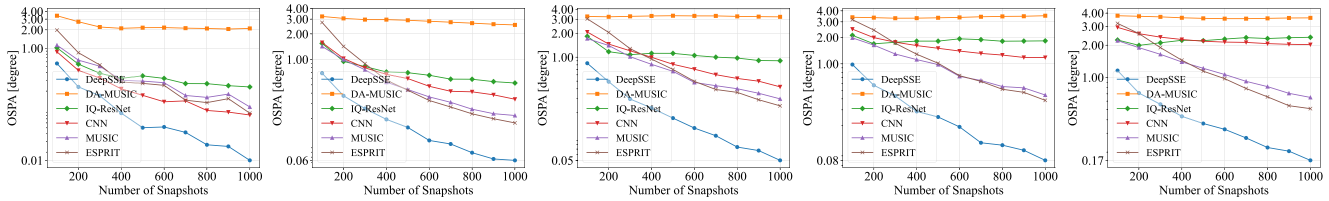
[2] The source code is obtained from https://github.com/DA-MUSIC/TVT23.

[3] Our implementation of MUSIC and ESPRIT can be obtained from https://github.com/zhiim/doa_py.



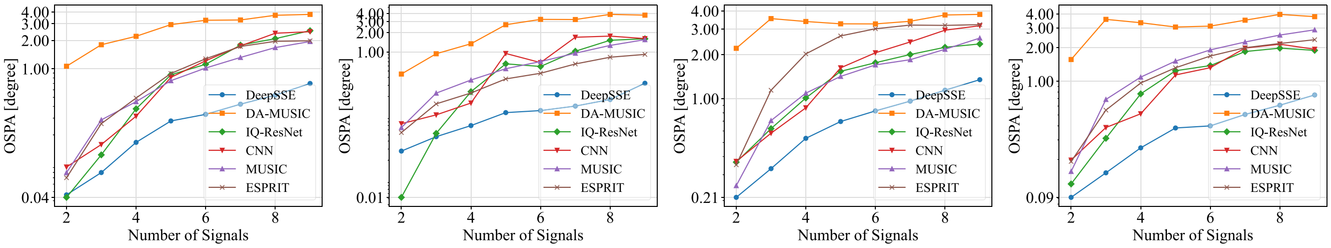
(a) 4 signals (b) 5 signals (c) 6 signals (d) 7 signals (e) 8 signals

Fig. 8. The OSPA metric of different algorithms using $T = 300$ snapshots under different number of signals vs. SNRs.



(a) 4 signals (b) 5 signals (c) 6 signals (d) 7 signals (e) 8 signals

Fig. 9. The OSPA metric of different algorithms at 0 dB under different number of signals vs. number of snapshots.



(a) -5 dB with 300 snapshots (b) 0 dB with 300 snapshots (c) -5 dB with 100 snapshots (d) 0 dB with 100 snapshots

Fig. 10. The OSPA metric of different algorithms vs. number of signals.

of sources for estimation. All results were obtained through 1000 Monte Carlo experiments.

First, we fixed the number of snapshots at $T = 300$, and conducted experiments under scenarios with {4, 5, 6, 7, 8} signals to calculate the OSPA metric of different DOA estimators at various SNRs. The experimental results are depicted in Fig. 8. It can be seen in Fig. 8a-8e that DeepSSE achieved the best estimation accuracy across different SNRs under varying numbers of signals. DA-MUSIC performs poorly in scenarios with varying numbers of signals. IQ-ResNet and CNN achieve good estimation accuracy when the signal number is equal to 4. When the number of signals exceeds the training data, the performance of IQ-ResNet and CNN degrades rapidly. DeepSSE, on the other hand, maintains the best estimation accuracy, and significantly outperforms other DL-based and model-based DOA estimators.

Next, keeping the SNR at 0 dB, we evaluated the performance using different numbers of snapshots under different numbers of signals. The experimental results are shown in Fig. 9. The experimental results demonstrate that DeepSSE consistently maintains the highest estimation accuracy across varying numbers of signals and different numbers of snapshots. IQ-ResNet and CNN estimate poorly when the numbers of signals is larger than 4, performing worse than MUSIC and ESPRIT algorithms. IQ-ResNet is sensitive to the number of snapshots, reaching its optimum around the number used in the training data.

Then, using different SNRs and number of snapshots, we did DOA estimation for each of the following numbers of signals: {2, 3, 4, 5, 6, 7, 8, 9}. We used different estimators and calculated the OSPA between the estimated DOA results and the ground truth. The experimental results are shown in Fig. 10. We can observe that DeepSSE maintains the highest estimation accuracy across different signal numbers. When the signal number exceeds 4, DeepSSE still maintains a high estimation accuracy, much better than other estimators.

2) Statistical Analysis: To validate the significance of the experimental results, we performed a statistical analysis of the OSPA metric for different algorithms under various experimental conditions. The OSPA metric involves a minimization over permutations and a constant penalty term. Consequently, the OSPA data did not conform to a normal distribution (this has been verified by the Shapiro-Wilk test [38], $p < 0.05$). Therefore, we first obtained data from 1000 Monte Carlo experiments. Then, we employed the Bootstrap method [39] to compute the confidence intervals of the OSPA metrics for different algorithms by resampling 9999 times. In addition, we compared DeepSSE to different baseline algorithms (MUSIC, CNN, and IQ-ResNet). We utilized the non-parametric Wilcoxon signed-rank test [40] with Holm-Bonferroni correction method [41] to calculate the p-value to verify the significance of the performance improvement achieved by DeepSSE. Since we focused on whether DeepSSE had improvements relative to other algorithms, we adopted

TABLE II
STATISTICAL ANALYSIS OF DEEPSSE'S IMPROVEMENT RELATIVE TO DIFFERENT BASELINE ALGORITHMS UNDER -5 dB, 300 SNAPSHOTS AND 6 SIGNALS

Baseline	95% CI[1]	p-value[2]	δ[3]	Improvement
MUSIC	(1.154, 1.297)	1.514×10^{-62}	0.321	25.226%
CNN	(0.785, 0.851)	1.873×10^{-86}	0.454	40.599%
IQ-ResNet	(0.975, 1.084)	2.594×10^{-48}	0.294	52.745%
DeepSSE	(0.462, 0.509)	-	-	-

[1] 95% CI: 95% confidence interval of OSPA for each baseline algorithm;
[2] p-value: The alternative hypothesis is set as: the distribution of the OSPA difference between the baseline algorithm and DeepSSE is stochastically greater than a distribution symmetric about zero;
[3] δ: Cliff's delta values for DeepSSE relative to the baseline algorithms.

TABLE III
STATISTICAL ANALYSIS OF DEEPSSE'S IMPROVEMENT RELATIVE TO DIFFERENT BASELINE ALGORITHMS UNDER -5 dB, 100 SNAPSHOTS AND 7 SIGNALS

Baseline	95% CI	p-value	δ	Improvement
MUSIC	(2.869 ,2.955)	1.027×10^{-146}	0.709	30.540%
CNN	(2.701 ,2.766)	2.822×10^{-142}	0.716	26.016%
IQ-ResNet	(2.432 ,2.500)	3.111×10^{-71}	0.458	18.011%
DeepSSE	(1.995 ,2.050)	-	-	-

TABLE IV
STATISTICAL ANALYSIS OF DEEPSSE'S IMPROVEMENT RELATIVE TO DIFFERENT BASELINE ALGORITHMS UNDER 0 dB, 100 SNAPSHOTS AND 4 SIGNALS

Baseline	95% CI	p-value	δ	Improvement
MUSIC	(1.047 ,1.233)	3.974×10^{-52}	0.229	73.763%
CNN	(0.447 ,0.520)	2.571×10^{-36}	0.235	38.132%
IQ-ResNet	(0.407 ,0.482)	6.966×10^{-11}	0.157	22.708%
DeepSSE	(0.270 ,0.329)	-	-	-

a one-tailed test. Furthermore, to quantify the improvement of the algorithm performance, we calculated the percentage increase in OSPA metric of DeepSSE relative to other algorithms. Simultaneously, we used Cliff's delta [42] value δ to measure the effect size, where $\delta > 0.147$ indicates small effect, $\delta > 0.33$ medium effect, and $\delta > 0.474$ large effect. The results are shown in Tables II-IV. We can see that DeepSSE achieves a significant performance improvement compared to baselines.

E. Performance With Number of Signals Unknown

DeepSSE can perform DOA estimation without prior knowledge of the number of signals. To demonstrate DeepSSE's ability to estimate DOA without knowing the number of signals, we used DeepSSE, DA-MUSIC, IQ-ResNet, and CNN to estimate the number of signals. DeepSSE, IQ-ResNet and CNN determine the number of signals by counting the detected peaks. In the experiment, we used a threshold of 0.5 for peak detection. And we counted the number of peaks as the estimated number of signals. DA-MUSIC trained a separate signal number classification network to estimate the number of signals [26]. We used the respective signal number estimation networks for comparison. In our experiment, we calculate the signal number estimation accuracy of different estimators, which is expressed as

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\hat{K}_i = K}, \tag{25}$$

where N is the number of Monte Carlo trials, $\hat{K}_i$ is the estimated number of signals in the i-th trial, and K is the true number of signals.

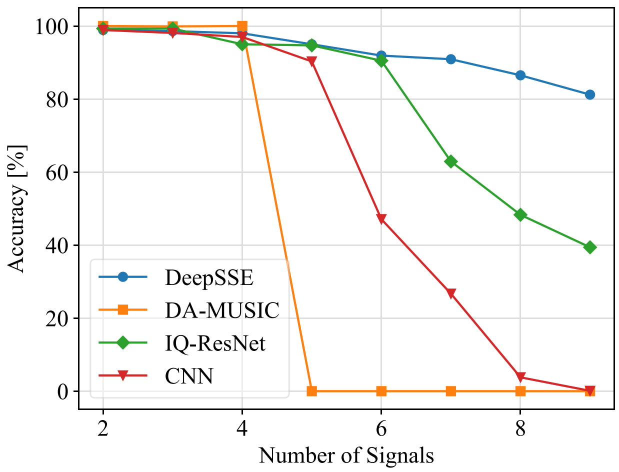
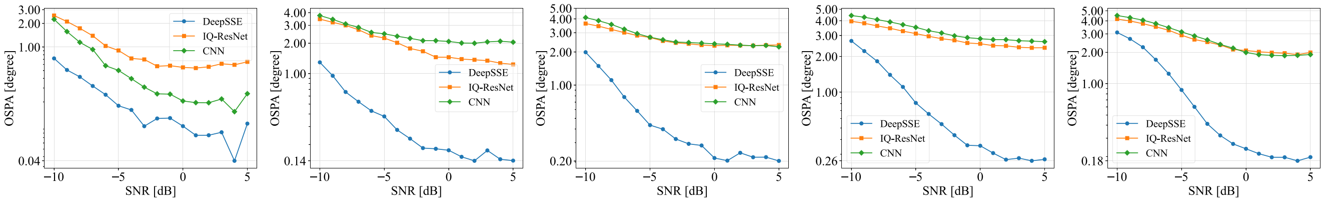


Fig. 11. The accuracy of signal number estimation of different DOA estimators at 0 dB SNR using 300 snapshots vs. number of signals.

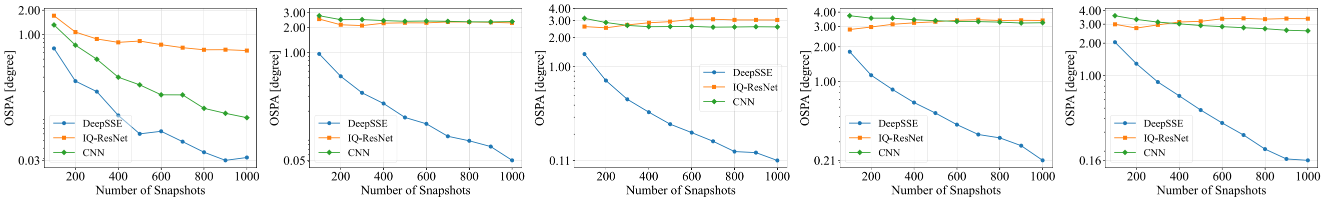
Under the conditions of 0 dB SNR and 300 snapshots, we estimated the signal number under different numbers of signals. The accuracy of signal number estimation was calculated through 1000 Monte Carlo trials. The experimental results are shown in Fig. 11. We can observe that DeepSSE maintains high accuracy in estimating the number of signals across different numbers of signals. However, when the number of signals exceeds the maximum number of 4 in the training data, the accuracy of IQ-ResNet and CNN's signal number estimation significantly decreases. Furthermore, DA-MUSIC cannot be used in scenarios where the number of signals is greater than 4.

To further verify the superior performance of DeepSSE in scenarios with an unknown number of signals, we compared OSPA metric of different estimators when the number of signals is unknown. The peak search threshold is set to 0.5. Given that MUSIC and ESPRIT require prior knowledge of the number of signals, and DA-MUSIC's signal number estimator fails when the number of signals is greater than 4, we only compare the performance of DeepSSE, IQ-ResNet, and CNN here. We conducted experiments under various numbers of signals, diverse SNRs, and different snapshot numbers to calculate the OSPA metric for different DOA estimators. The results are plotted in Fig. 12 and 13. As the number of signals increases, the estimation performance of DeepSSE consistently remains optimal and stable. When the number of signals is greater than 4, the performance of IQ-ResNet and CNN is significantly lower than that of DeepSSE, and the performance of IQ-ResNet is slightly better than CNN.



(a) 4 signals (b) 5 signals (c) 6 signals (d) 7 signals (e) 8 signals

Fig. 12. The OSPA metric of different DOA estimators at 300 snapshots under different number of signals vs. SNRs.



(a) 4 signals (b) 5 signals (c) 6 signals (d) 7 signals (e) 8 signals

Fig. 13. The OSPA metric of different DOA estimators at 0 dB SNR under different number of signals vs. number of snapshots.

F. Robustness to Model Mismatch

The parameterizable grid search and end-to-end data learning make DeepSSE highly robust. DeepSSE can effectively cope with various scenarios of mismatches between actual signals and ideal models. To verify DeepSSE's robustness, we trained DeepSSE using the ideal data and evaluated its estimation accuracy under different array and signal error conditions. We specifically considered four scenarios: sensor position errors, sensor mutual coupling errors, spatially correlated noise, and multipath signals. With 1000 Monte Carlo trials under various simulation conditions, we computed the OSPA metric to evaluate performance.

1) Sensor Position Errors: We model the sensor position error using the same model used in [19]. The real steering vector in the presence of sensor position errors can be expressed as

$$\mathbf{a}_{\text{pos}}(\theta) = \exp\Big(j2\pi f/c(\mathbf{L}^T + \Delta\mathbf{L}^T)\sin(\theta)\Big), \tag{26}$$

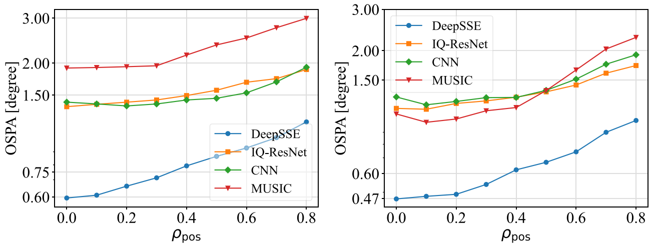
where $\mathbf{L} \in \mathbb{R}^{1\times M}$ is the ideal sensor position vector, $\Delta\mathbf{L} \in \mathbb{R}^{1\times M}$ is the sensor position error added to each sensor.

For each element in $\Delta\mathbf{L}$, we randomly generate the array position error value using a uniform distribution of $\mathcal{U}(-\rho_{\text{pos}}, \rho_{\text{pos}})$. By adjusting the size of parameter ρ_{pos}, we can control the degree of array error. The OSPA result is shown in Fig. 14. We can see that DeepSSE reaches the best performance when the sensor position error is present.

2) Sensor Mutual Coupling Errors: The mutual coupling model [19] can be expressed as

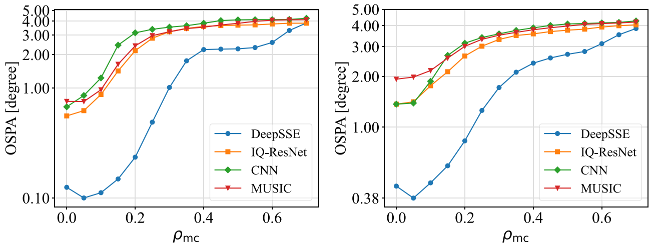
$$\mathbf{a}_{\text{mc}}(\theta) = (\mathbf{I}_M + \mathbf{E}_{\text{mc}})\mathbf{a}(\theta), \tag{27}$$

where $\mathbf{I}_M$ is the identity matrix, $\mathbf{E}_{\text{mc}}$ is the mutual coupling error matrix, and $\mathbf{a}(\theta)$ is the ideal steering vector. The mutual coupling error matrix is a Toeplitz matrix, which can be expressed as $\mathbf{E}_{\text{mc}} = \text{Toeplitz}([0, \gamma, \ldots, \gamma^{M-1}])$. And γ is set to $\rho_{\text{mc}} e^{j60^\circ}$ to maintain consistency with [19]. We can control the influence of coupling errors by setting the value of ρ_{mc}.



(a) 0 dB, 100 snapshots and 6 signals (b) -5 dB, 300 snapshots and 6 signals

Fig. 14. The OSPA metric of different DOA estimators at different scenarios when sensor position errors are present.



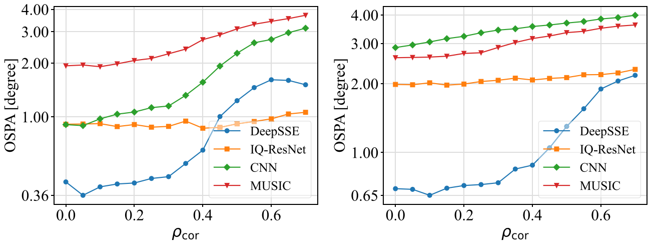
(a) 0 dB, 300 snapshots and 6 signals (b) 0 dB, 100 snapshots and 6 signals

Fig. 15. The OSPA metric of different DOA estimators at different scenarios when sensor mutual coupling errors are present.

We plot the OSPA curves of different algorithms as ρ_{mc} varies in Fig. 15. It is evident that DeepSSE achieves the best estimation accuracy when element mutual coupling errors exist.

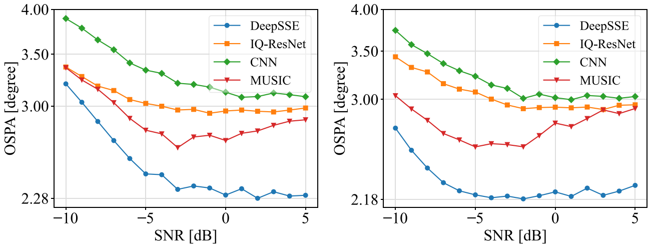
3) Spatially Correlated Noise: The spatially correlated noise received by array is a zero-mean Gaussian process which is spatially correlated and temporally uncorrelated [43], such that

$$\begin{aligned} \mathbb{E}[\mathbf{n}(t)] &= 0, \\ \mathbb{E}\Big[\mathbf{n}(t_1)\mathbf{n}^H(t_2)\Big] &= \delta(t_1 - t_2)\mathbf{R}_n. \end{aligned} \tag{28}$$



(a) 0 dB, 100 snapshots and 6 signals (b) 0 dB, 100 snapshots and 8 signals

Fig. 16. The OSPA metric of different DOA estimators at different scenarios when correlated noise is present.



(a) 100 snapshots and 6 signals (b) 300 snapshots and 6 signals

Fig. 17. The OSPA metric of different DOA estimators at different scenarios when multipath signals are present.

In (28), the noise covariance matrix $\mathbf{R}_n$ is an unknown positive definite Hermitian matrix representing the spatial correlation of noise across the array aperture.

The spatially correlated noise matrix is generated using the following formula:

$$[\mathbf{R}_n]_{l,k} = (\rho_{\text{cor}})^{|l-k|}\sigma_n^2 \exp\left(j\frac{\pi}{2}(l-k)\right), \tag{29}$$

where σ_n^2 is the noise power. We can control the degree of spatial correlation by adjusting the value of ρ_{cor}.

The result is shown in Fig. 16. The experimental results show that DeepSSE exhibits good robustness under spatially correlated noise. It is also worth noting that since IQ-ResNet does not use the covariance matrix of signal as an input feature, the impact of spatially correlated noise on it is relatively small.

4) Multipath Signals: We treat multipath signals as coherent interference signals of the original signal [44], [45]. In the presence of multipath, the real signals received by array in the presence of multi-path can be expressed as

$$\mathbf{y}(t) = \left(\mathbf{a}(\theta) + \sum_i A_i e^{j\phi_i}\mathbf{a}(\theta + \Delta\theta_i)\right)s(t) + \mathbf{n}(t), \tag{30}$$

where A_i is the relative amplitude of the i-th multipath signal, ϕ_i is the phase difference of the i-th multipath signal, and $\Delta\theta_i$ is the incident angle offset of the i-th multipath signal.

We assume that each target signal includes two additional multipath components. And we randomly select the amplitude variation relative to the original signal from a uniform distribution of $\mathcal{U}(0.3, 0.9)$, randomly set the phase difference relative to the original signal using a uniform distribution of $\mathcal{U}(0, 2\pi)$, and randomly select the incident angle offset relative to the original signal using a uniform distribution of $\mathcal{U}(-30^\circ, 30^\circ)$. Fig. 17 shows the OSPA of different estimators vs. SNRs. This result demonstrates the robustness advantage of DeepSSE compared to other algorithms when multipath signals are present. But it also shows that multipath signals will decrease performance of DeepSSE.

G. Complexity Analysis

As the superior estimation performance of DeepSSE is demonstrated in the previous sections, we further analyze the computational complexity of DeepSSE. The computational complexity of DeepSSE mainly comes from the SFE sub-network, the AFM sub-network and the AGS sub-network. The computational complexity of the SFE is mainly comes from the convolutional layers. All convolutional layers in SFE utilize the same number of channels c, and the same (3, 3) kernel, while maintaining the same height and width of the data throughout each layer. So the computational complexity of SFE is given by $C_{SFE} \sim O(mc^2M^2)$. The computational complexity of AFM arises from the fully connected layers. As $M < d$ normally, the computational complexity of AFM is $C_{AFM} = O(Gd^2)$. The computational complexity of AGS mainly depends on the attention mechanism and the subsequent fully connected layers and feature-wise linear layers, and it can be calculated as $C_{AGS} \sim O(nM^2Gd) + O(nGd^2) + O(Gd) \sim O(nM^2Gd + nGd^2)$. So the computational complexity of DeepSSE can be expressed as

$$\begin{aligned} C_{DeepSSE} &\sim C_{SFE} + C_{AFM} + C_{AGS} \\ &\sim O\Big(mc^2M^2\Big) + O\Big(Gd^2\Big) + O\Big(nM^2Gd + nGd^2\Big) \\ &\sim O\Big(mc^2M^2 + nM^2Gd + nGd^2\Big). \end{aligned} \tag{31}$$

In contrast, the computational complexity of MUSIC mainly comes from the SVD of the covariance matrix and angle grid search, which can be expressed as $C_{MUSIC} \sim O(M^3 + GM(M-K))$. The computational complexity of ESPRIT mainly comes from SVD and least squares calculation (we consider the total least squares method), which can be represented as $C_{ESPRIT} \sim O(M^3 + K^2M + K^3)$.

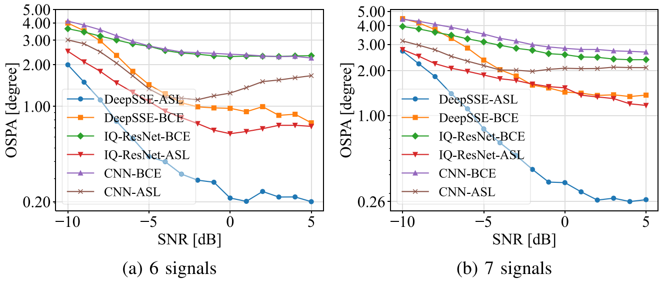
To evaluate the computational complexity of the DeepSSE algorithm fairly, we also conducted runtime experiments. All model-driven methods are implemented with NumPy (2.21), and data-driven methods are inferred with PyTorch (2.41 + CPU). The runtime cost of all methods is measured by running 1000 times on 12th Gen Intel(R) Core(TM) i5-12400. The results are shown in Table V. Data-driven methods generally suffer from limitations in computational complexity. It is worth noting that although DA-MUSIC has the smallest number of parameters and FLOPs, its runtime cost is the longest due to the sequential computation of RNN. Although DeepSSE has the largest number of FLOPs, its runtime is short due to the parallel computation advantage of the attention module. Our proposed DeepSSE not only significantly outperforms other methods in estimation accuracy, but also maintains a good balance in parameter size and runtime complexity.

H. Hyperparameter Sensitivity Analysis

To evaluate the sensitivity of the hyperparameters of DeepSSE, we conducted an analysis of the hyperparameters of DeepSSE. We analyzed the influence of hyperparameters from

TABLE V
RUNTIME OF DIFFERENT DOA ESTIMATORS UNDER 0 dB AND 1000 SNAPSHOTS

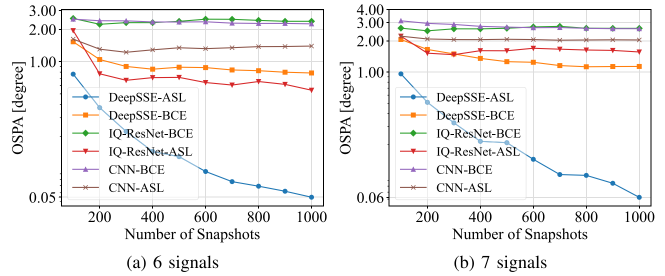
Algorithm	Params (M)	FLOPs (M)	Runtime (ms)
MUSIC	-	1.099	0.519
ESPRIT	-	2.938	0.426
CNN	28.251	96.078	9.132
IQ-ResNet	1.858	181.414	2.293
DA-MUSIC	0.035	12.633	51.251
DeepSSE	0.573	219.718	4.985



(a) 6 signals

(b) 7 signals

Fig. 18. The OSPA metric of different DOA estimators at 300 snapshots under different numbers of signals vs. SNR.



(a) 6 signals

(b) 7 signals

Fig. 19. The OSPA metric of different DOA estimator at 0 dB SNR under different number of signals vs. number of snapshots.

ASL and hyperparameters that influence the computational complexity. That is, the focusing parameters γ_+ and γ_- of ASL, the probability margin η_m of ASL, the number of residual blocks in SFE m, the number of kernels in each convolutional layer c, the number of cross attention blocks n, and the feature dimension d.

We first constructed a dataset with 100,000 data points. Each data point was generated with a randomly selected SNR from $\{-10, -9, \ldots, 5\}$ dB, a randomly selected number of snapshots from $\{100, 200, \ldots, 1000\}$, and a randomly selected number of signals from 1 to 9. We used the OSPA metric to measure the performance of DeepSSE under different hyperparameters using this dataset.

The experimental results are shown in Table VI. The results demonstrate that for the ASL, the performance of DeepSSE degrades when the difference between γ_+ and γ_- is either too small or too large, or when the threshold η_m is set too high or too low. Based on our experiments, it is recommended to set the parameters to $\gamma_+ = 1$, $\gamma_- = 4$, and $\eta_m = 0.05$ for better performance. In terms of network size, experimental results show that increasing m and d may improve model performance, while c and n should not be too large or too small. The selection of network hyperparameters should comprehensively consider the influence on model complexity. It seems that n and d have the most significant impact on the number of parameters, while c and d have a greater impact on inference time.

I. Ablation Study

1) The Contribution of Asymmetric Loss: To demonstrate the benefit of ASL, we trained different DL-based DOA estimators using both ASL and BCE. We refer to the DeepSSE trained with ASL as DeepSSE-ASL and the DeepSSE framework trained with BCE as DeepSSE-BCE. Similarly, we refer to the IQ-ResNet framework trained with ASL as IQ-ResNet-ASL, the IQ-ResNet framework trained with BCE (same as the IQ-ResNet in [24]) as IQ-ResNet-BCE, and the CNN framework trained with ASL as CNN-ASL, and the CNN framework trained with BCE (same as the CNN in [23]) as CNN-BCE.

Maintaining the same experimental conditions as in Section IV-D, we plotted the curves of OSPA metric for different DOA estimators. The experimental results are shown in Fig. 18 and 19. It can be observed that DeepSSE-ASL achieves the best estimation accuracy. DeepSSE-ASL maintains high and optimal estimation accuracy across different numbers of signals, while DeepSSE-BCE may experience performance degradation when the number of signals is large. All DOA estimators trained with ASL have an improvement in estimation performance. It should also be noted that only the combination of DeepSSE's novel network structure and ASL can achieve optimal estimation accuracy, significantly outperforming other algorithms trained with ASL.

2) The Contribution of Different Sub-Networks: We employ equivalent mathematical operations to replace DeepSSE's three sub-networks to discuss the contributions and impacts of different subnetworks. We replace the SFE sub-network with the original covariance matrix calculation, the AFM sub-network with the original steering vector calculation, and the cross-attention in the AGS sub-network with the dot product operation, respectively. Ablations on different sub-networks in DeepSSE are shown in Fig. 20. In the comparative experiments of the three sub-network architectures, the AFM has the least impact on the overall system performance, while the AGS sub-network has the most significant impact. It indicates that when DeepSSE lacks the AGS sub-network, DeepSSE experiences a significant performance degradation in scenarios with a large number of signals. This result further verifies the critical role of the angle grid search mechanism in DOA estimation tasks with variable numbers of signals. When the number of signals is 8, the lack of any sub-network will lead to significant performance degradation. This indicates that each sub-network is critical to the overall performance of DeepSSE.

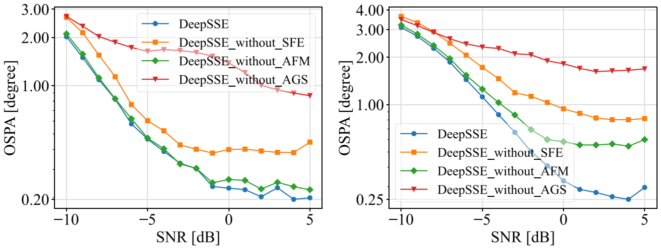
V. CONCLUSION AND FUTURE WORK

We propose a novel model-based deep learning motivated DOA estimation algorithm named DeepSSE for variable and unknown number of signals. DeepSSE features a learnable parametric learning angular grid search structure composed

TABLE VI
HYPERPARAMETER SENSITIVITY ANALYSIS OF DEEPSSE

Group	Loss Function			Network Architecture				Performance Metrics		Computational Complexity	
	γ_+	γ_-	η_m	m	c	n	d	OSPA[1]	p-value[2]	Params	Runtime[3] (ms)
Baseline	1	4	0.05	4	32	2	128	1.866 [1.857, 1.875]	-	572660	5.060
γ_+, γ_-	1	2	0.05	4	32	2	128	2.035 [2.026, 2.045]	1.000	572660	5.520
	1	6	0.05	4	32	2	128	1.958 [1.949, 1.967]	1.000	572660	5.243
	1	8	0.05	4	32	2	128	2.097 [2.070, 2.088]	1.000	572660	5.418
	0	2	0.05	4	32	2	128	1.934 [1.921, 1.939]	1.000	572660	5.208
	0	4	0.05	4	32	2	128	1.877 [1.881, 1.898]	1.000	572660	5.502
η_m	1	4	0.00	4	32	2	128	1.878 [1.869, 1.887]	1.000	572660	4.942
	1	4	0.10	4	32	2	128	1.879 [1.870, 1.888]	$\mathbf{5.200 \times 10^{-4}}$	572660	5.685
	1	4	0.20	4	32	2	128	1.914 [1.905, 1.923]	1.000	572660	4.891
	1	4	0.30	4	32	2	128	2.137 [2.128, 2.146]	1.000	572660	4.945
m	1	4	0.05	2	32	2	128	1.930 [1.921, 1.940]	1.000	535540	4.499
	1	4	0.05	6	32	2	128	1.861 [1.852, 1.870]	2.872×10^{-17}	609780	4.664
	1	4	0.05	8	32	2	128	**1.860 [1.852, 1.869]**	$\mathbf{5.197 \times 10^{-20}}$	646900	6.041
c	1	4	0.05	4	16	2	128	1.930 [1.921, 1.939]	1.000	514580	4.752
	1	4	0.05	4	64	2	128	1.904 [1.895, 1.912]	1.000	799412	5.771
	1	4	0.05	4	128	2	128	1.957 [1.948, 1.966]	1.000	1695284	8.579
n	1	4	0.05	4	32	1	128	1.937 [1.928, 1.946]	1.000	374388	3.151
	1	4	0.05	4	32	3	128	1.876 [1.867, 1.885]	$\mathbf{3.070 \times 10^{-4}}$	770932	6.919
	1	4	0.05	4	32	4	128	1.877 [1.868, 1.886]	1.000	969204	8.345
d	1	4	0.05	4	32	2	48	1.975 [1.967, 1.984]	1.000	216420	3.628
	1	4	0.05	4	32	2	64	1.898 [1.889, 1.907]	1.000	275380	3.800
	1	4	0.05	4	32	2	256	**1.859 [1.850, 1.858]**	$\mathbf{5.182 \times 10^{-7}}$	1462132	7.669

[1] OSPA: The OSPA values are presented in the format "mean [95% confidence interval]";
[2] p-value: The alternative hypothesis set as: the distribution of the OSPA difference between the current and baseline is stochastically greater than a distribution symmetric about zero. And the statistical test is done with the Wilcoxon signed-rank test with Holm-Bonferroni correction method;
[3] Runtime: The runtime is measured by inferring 1000 times using PyTorch on 12th Gen Intel(R) Core(TM) i5-12400.



(a) 300 snapshots and 6 signals

(b) 300 snapshots and 8 signals

Fig. 20. The OSPA metric of different estimators when SNR varies.

of three sub-networks. We use ASL to train DeepSSE, addressing the impact of spatial-spectral sparsity on network training. Meanwhile, we use OSPA as evaluation metric for scenarios with a variable and unknown number of signals to assess the algorithm's performance. In the simulation experiments, we conducted experimental analyses under varying conditions of signal numbers, SNRs, snapshot numbers, and model mismatches. DeepSSE outperforms other DL-based and model-based DOA estimators in different experimental setups. DeepSSE also maintains a good balance in parameter size and runtime complexity, while achieving extremely high estimation accuracy. Our hyperparameter analysis and ablation experiments also offer a reference for tuning DeepSSE in practical applications.

While DeepSSE demonstrates superior performance across various scenarios, there are still several directions for future work. The covariance matrix used in DeepSSE works well as an input feature representation for spatial features. Currently, other input features, such as IQ raw data [24] and the spectrogram of STFT [12], still have some limitations. A more efficient input feature will potentially further improve the performance and robustness of DeepSSE. Specifically designed network modules to reconstruct the ideal covariance matrix could also enhance robustness to model mismatch. Studies like [27] have explored this in depth. Finally, investigating lightweight variants of the attention mechanism or other more efficient network architectures would reduce computational complexity. This would especially benefit resource-constrained devices lacking parallel matrix computation capabilities. These improvements could greatly expand DeepSSE's applicability.

REFERENCES

[1] M. Pesavento, M. Trinh-Hoang, and M. Viberg, "Three more decades in array signal processing research: An optimization and structure exploitation perspective," *IEEE Signal Process. Mag.*, vol. 40, no. 4, pp. 92–106, Jun. 2023.

[2] R. Zhang et al., "Integrated sensing and communication with massive MIMO: A unified tensor approach for channel and target parameter estimation," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8571–8587, Aug. 2024.

[3] R. Zhang et al., "Channel estimation for movable-antenna MIMO systems via tensor decomposition," *IEEE Wireless Commun. Lett.*, vol. 13, no. 11, pp. 3089–3093, Nov. 2024.

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

[5] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 8, 1983, pp. 336–339.

[6] 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.

[7] 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.

[8] M. M. Hyder and K. Mahata, "Direction-of-arrival estimation using a mixed $l_{2,0}$ norm approximation," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4646–4655, Sep. 2010.

[9] J. Yin and T. Chen, "Direction-of-arrival estimation using a sparse representation of array covariance vectors," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4489–4493, Sep. 2011.

[10] R. Zhang, B. Shim, and W. Wu, "Direction-of-arrival estimation for large antenna arrays with hybrid analog and digital architectures," *IEEE Trans. Signal Process.*, vol. 70, pp. 72–88, 2022.

[11] Y. Guo, Z. Zhang, Y. Huang, and P. Zhang, "DOA estimation method based on cascaded neural network for two closely spaced sources," *IEEE Signal Process. Lett.*, vol. 27, pp. 570–574, 2020.

[12] 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. (EUSIPCO)*, 2018, pp. 1462–1466.

[13] T. N. T. Nguyen, D. L. Jones, K. N. Watcharasupat, H. Phan, and W.-S. Gan, "SALSA-lite: A fast and effective feature for polyphonic sound event localization and detection with microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2022, pp. 716–720.

[14] P. Sudarsanam, A. Politis, and K. Drossos, "Assessment of self-attention on learned features for sound event localization and detection," 2021, *arXiv:2107.09388*.

[15] 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.

[16] D. Hu, Y. Zhang, L. He, and J. Wu, "Low-complexity deep-learning-based DOA estimation for hybrid massive MIMO systems with uniform circular arrays," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 83–86, Jan. 2020.

[17] Y. Tian, S. Liu, W. Liu, H. Chen, and Z. Dong, "Vehicle positioning with deep-learning-based direction-of-arrival estimation of incoherently distributed sources," *IEEE Internet Things J.*, vol. 9, no. 20, pp. 20083–20095, Oct. 2022.

[18] J. Cong, X. Wang, C. Yan, L. T. Yang, M. Dong, and K. Ota, "CRB weighted source localization method based on deep neural networks in multi-UAV network," *IEEE Internet Things J.*, vol. 10, no. 7, pp. 5747–5759, Apr. 2023.

[19] 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 Propag.*, vol. 66, no. 12, pp. 7315–7327, Dec. 2018.

[20] A. M. Elbir, "DeepMUSIC: Multiple signal classification via deep learning," *IEEE Sens. Lett.*, vol. 4, no. 4, pp. 1–4, Apr. 2020.

[21] J. Yu and Y. Wang, "Deep learning-based multipath DoAs estimation method for mmWave massive MIMO systems in low SNR," *IEEE Trans. Veh. Technol.*, vol. 72, no. 6, pp. 7480–7490, Jun. 2023.

[22] H. Wang, X. Wang, X. Lan, and T. Su, "Effective high-resolution off-grid DOA estimation with mutual coupling via CNN framework," *IEEE Sensors J.*, vol. 25, no. 1, pp. 1133–1143, Jan. 2025.

[23] 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.

[24] S. Zheng et al., "Deep learning-based DOA estimation," *IEEE Trans. Cogn. Commun. Netw.*, vol. 10, no. 3, pp. 819–835, Jun. 2024.

[25] J. P. Merkofer, G. Revach, N. Shlezinger, and R. J. van Sloun, "Deep augmented MUSIC algorithm for data-driven DoA estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2022, pp. 3598–3602.

[26] J. P. Merkofer, G. Revach, N. Shlezinger, T. Routtenberg, and R. J. G. Van Sloun, "DA-MUSIC: Data-driven DoA estimation via deep augmented MUSIC algorithm," *IEEE Trans. Veh. Technol.*, vol. 73, no. 2, pp. 2771–2785, Feb. 2024.

[27] D. H. Shmuel, J. P. Merkofer, G. Revach, R. J. Van Sloun, and N. Shlezinger, "SubspaceNet: Deep learning-aided subspace methods for DoA estimation," *IEEE Trans. Veh. Technol.*, vol. 74, no. 3, pp. 4962–4976, Mar. 2025.

[28] X. Xu and Q. Huang, "MD-DOA: A model-based deep learning DOA estimation architecture," *IEEE Sensors J.*, vol. 24, no. 12, pp. 20240–20253, Jun. 2024.

[29] J. Ji, W. Mao, F. Xi, and S. Chen, "TransMUSIC: A transformer-aided subspace method for DOA estimation with low-resolution ADCS," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2024, pp. 8576–8580.

[30] M. L. L. de Oliveira and M. J. Bekooij, "Deep-MLE: Fusion between a neural network and MLE for a single snapshot DOA estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2022, pp. 3673–3677.

[31] N. Shlezinger, J. Whang, Y. C. Eldar, and A. G. Dimakis, "Model-based deep learning," *Proc. IEEE*, vol. 111, no. 5, pp. 465–499, May 2023.

[32] T. Ridnik et al., "Asymmetric loss for multi-label classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 82–91.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[34] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu, "Query2Label: A simple transformer way to multi-label classification," 2021, *arXiv:2107.10834*.

[35] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[36] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3452–3457, Jul. 2011.

[37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[38] S. Shaphiro and M. Wilk, "An analysis of variance test for normality," *Biometrika*, vol. 52, no. 3, pp. 591–611, 1965.

[39] B. Efron, "Bootstrap methods: Another look at the jackknife," in *Breakthroughs in Statistics: Methodology and Distribution*. New York, NY, USA: Springer, 1992, pp. 569–593. [Online]. Available: https://doi.org/10.1007/978-1-4612-4380-9_41

[40] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in Statistics: Methodology and Distribution*. Cham, Switzerland: Springer, 1992, pp. 196–202.

[41] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Stat.*, vol. 6, no. 2, pp. 65–70, 1979. [Online]. Available: http://www.jstor.org/stable/4615733

[42] N. Cliff, "Dominance statistics: Ordinal analyses to answer ordinal questions," *Psychol. Bull.*, vol. 114, no. 3, p. 494, 1993.

[43] M. Agrawal and S. Prasad, "A modified likelihood function approach to DOA estimation in the presence of unknown spatially correlated Gaussian noise using a uniform linear array," *IEEE Trans. Signal Process.*, vol. 48, no. 10, pp. 2743–2749, Oct. 2000.

[44] H. Xiang, B. Chen, T. Yang, and D. Liu, "Improved de-multipath neural network models with self-paced feature-to-feature learning for DOA estimation in multipath environment," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5068–5078, May 2020.

[45] N. Yuen and B. Friedlander, "DOA estimation in multipath: An approach using fourth-order cumulants," *IEEE Trans. Signal Process.*, vol. 45, no. 5, pp. 1253–1263, May 1997.

Qian Xu received the B.S. degree from the Harbin Institute of Technology, Harbin, China, in 2023, where he is currently pursuing the M.S. degree. His research interests include deep learning for signal processing, especially in the field of direction of arrival estimation.

Yulong Gao (Senior Member, IEEE) received the M.S. degree in communication and information systems from Harbin Engineering University, Harbin, China, in 2004, and the Ph.D. degree in information and communication engineering from the Harbin Institute of Technology, Harbin, in 2007, where he is currently a Professor with the School of Electronics and Information Engineering. From May 2012 to May 2013, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. His current research interests include intelligent communication and intelligent signal processing. He was a recipient of numerous awards, including the First Prize in Science and Technology Advancement Award of Ministry of Education of China in 2012 and 2013, respectively, and the First Prize of Technological Invention Award of Heilongjiang Province in 2012.

Ruoyu Zhang (Member, IEEE) received the B.E. and Ph.D. degrees in information and communication engineering from the Harbin Institute of Technology, Harbin, China, in 2014 and 2019, respectively. From 2017 to 2018, he was a Visiting Student with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada. He is currently an Associate Professor with the School of Electronic and Optical Engineering, Nanjing University of Science and Technology. His research interests include integrated sensing and communication, massive MIMO, millimeter-wave communications, and sparse signal processing.

Jinshan Kong received the B.S. degree from the Harbin Institute of Technology, Weihai, China, in 2023. She is currently pursuing the M.S. degree with the Harbin Institute of Technology, Harbin, China. Her research interests include deep learning-based signal processing, with a focus on spectrum prediction.

Chau Yuen (Fellow, IEEE) received the B.Eng. and Ph.D. degrees from Nanyang Technological University, Singapore, in 2000 and 2004, respectively. He was a Postdoctoral Fellow with the Lucent Technologies Bell Laboratories, Murray Hill, in 2005. From 2006 to 2010, he was with the Institute for Infocomm Research, Singapore. From 2010 to 2023, he was with the Engineering Product Development Pillar, Singapore University of Technology and Design. Since 2023, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University. He is currently the Provost's Chair of wireless communications, an Assistant Dean with Graduate College, and the Cluster Director for Sustainable Built Environment with ER@IN. He is listed as Top 2% Scientists by Stanford University. He has four U.S. patents and published more than 500 research articles at international journals. He received the IEEE Communications Society Leonard G. Abraham Prize in 2024, the IEEE Communications Society Best Tutorial Paper Award in 2024, the IEEE Communications Society Fred W. Ellersick Prize in 2023, the IEEE Marconi Prize Paper Award in Wireless Communications in 2021, the IEEE APB Outstanding Paper Award in 2023, and the EURASIP Best Paper Award for Journal on Wireless Communications and Networking in 2021. He is also a Highly Cited Researcher by Clarivate Web of Science from 2022. He serves as an Editor-in-Chief for *Computer Science* (Springer Nature); and an Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and the IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, where he was awarded as IEEE Transactions on Network Science and Engineering Excellent Editor Award in 2022 and 2024, respectively, and a Top Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY from 2009 to 2015. He also served as a Guest Editor for several special issues, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE WIRELESS COMMUNICATIONS MAGAZINE, IEEE COMMUNICATIONS MAGAZINE, IEEE VEHICULAR TECHNOLOGY MAGAZINE, the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, and *Applied Energy* (Elsevier).