

# Near-Field Sensing: A Low-Complexity Wavenumber-Domain Method

Hao Jiang, *Graduate Student Member, IEEE*, Zhaolin Wang, *Graduate Student Member, IEEE*,  
and Yuanwei Liu, *Fellow, IEEE*

arXiv:2408.09592v1 [eess.SP] 18 Aug 2024

**Abstract**—A novel low-complexity wavenumber-domain method is proposed for near-field sensing (NISE). Specifically, the power-concentrated region of the wavenumber-domain channels is related to the target position in a non-linear manner. Based on this observation, a bi-directional convolutional neural network (BiCNN)-based approach is proposed to capture such a relationship, thereby facilitating low-complexity target localization. This method enables direct and gridless target localization using only a limited bandwidth and a single antenna array. Simulation results demonstrate that: 1) during the offline training phase, the proposed BiCNN method can learn to localize the target with fewer trainable parameters compared to the naive neural network architectures; and 2) during the online implementation phase, the BiCNN method can spend 100x less time while maintaining comparable performance to the conventional two-dimensional multiple signal classification (MUSIC) algorithms.

**Index Terms**—Deep learning, near-field sensing (NISE), target localization.

## I. INTRODUCTION

With the adoption of high-frequency bands and extremely large aperture arrays (ELAAs) in the sixth-generation communication technology (6G), near-field sensing (NISE) has become a hotshot of research [1], [2]. Compared to the conventional angle-only far-field sensing, NISE can facilitate a full capture of the target position, i.e., both angle and distance [3]–[5]. Specifically, the curvature of spherical wavefronts in the near-field region causes targets in the same angular direction to experience phase variances. This feature gives rise to an additional distance dimension hidden in near-field channels. In fact, by leveraging such a distance dimension, NISE can localize targets with limited bandwidth and a single antenna array [6].

Given these advantages, significant efforts have been dedicated to harnessing the potential of NISE. For example, the classic multiple signal classification (MUSIC) algorithm was first utilized for the near-field scenario by [7]. The core idea of the extended version of MUSIC was to sample both angle and distance to form two-dimensional (2D) grids. Then, a 2D MUSIC spectrum can be constructed by traversing the girds, where the target is estimated to be located at the grid with the highest peak. Based on the 2D MUSIC, the authors of [8], [9] proposed an angle-then-distance estimation method

Hao Jiang and Zhaolin Wang are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K. (e-mail: {hao.jiang; zhaolin.wang}@qmul.ac.uk).

Yuanwei Liu is with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: yuanwei@hku.hk).

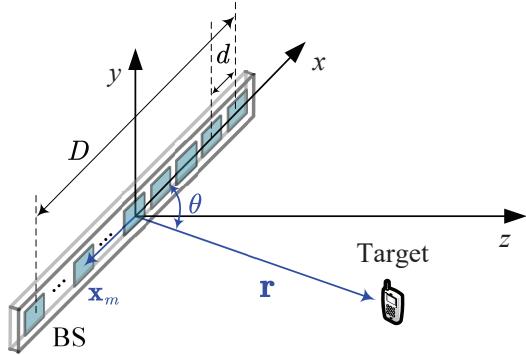


Fig. 1: An illustration of the system model.

thereby converting a 2D search into separated one-dimensional (1D) searches. As an alternative method, the authors of [10] turned 2D search into one 1D search by splitting the directional matrix. From the viewpoint of electromagnetic theory, authors of [11] exploited the electromagnetic propagation model to let the receiver find its location with low complexity. In fact, for most estimation algorithms in near-field sensing, the grid search is necessary due to the highly nonconvex and multimodal property of the resultant optimization problem [12], [13]. However, the grid search method can lead to high computational complexity, thus resulting in intolerable delays. To overcome this issue, low-complexity gridless methods are urgently needed for real-world implementations.

Motivated by the above, we propose a low-complexity gridless target localization algorithm for near-field systems. In particular, due to the positional information in space-domain channel representations being unapparent, we first resort to wavenumber-domain analysis, by which the positional information is related to the power-concentrated region of the channel. To capture a highly non-linear relationship between the target position and the power-concentrated region, we propose a bi-directional convolutional neural network (BiCNN) design, which enables the estimation of the target's location. Finally, the simulation results are presented to demonstrate the effectiveness of the proposed method, and further illustrate the superiority of the BiCNN approach over the traditional 2D-MUSIC algorithm in terms of complexity.

## II. SYSTEM MODEL

As illustrated in Fig. 1, we consider a near-field multiple-input single-output (MISO) system, which encompasses a base station (BS) with a  $M$ -antenna uniform linear array (ULA), where  $M = 2\tilde{M} + 1$  and a single-antenna target. Here,

the full-duplex technique is utilized at the BS, allowing for simultaneous transmission and reception. The transceivers are assumed to be located on the  $xz$  plane, and the ULA at the BS is placed along the  $x$ -axis. Letting the carrier frequency and the wavelength be  $f$  and  $\lambda$ , respectively, the array aperture of the ULA at the BS can be calculated by  $D = (M - 1)d$ , where  $d = 0.5\lambda$ . Here, letting  $r$  and  $\theta$  be the Euclidean distance between the BS and the target and the angle between the target and the  $x$ -axis respectively, it is assumed that the target is in the radiating near-field region of the BS, which is defined by  $r < \frac{2D^2}{\lambda}$ . Please note that the reactive near-field region is omitted here, since it is confined within a few wavelengths [14]. One narrow band with width  $B$  is utilized for localization of the target, and the channel reciprocity is assumed to hold in our system.

Due to the inaccuracy of planar-wave assumption in near-field region [4], we adopt a dedicated near-field channel to capture the characteristic of the channel. Without loss of generality, we set the origin of the coordinate system at the center of the ULA. Hence, the coordinate of the  $m$ -th antenna on the ULA and that of the target can be expressed as  $\mathbf{x}_m = [x_m^{(x)}, x_m^{(y)}, x_m^{(z)}] \in \mathbb{R}^{3 \times 1}$  and  $\mathbf{r} = [r^{(x)}, r^{(y)}, r^{(z)}] \in \mathbb{R}^{3 \times 1}$ , respectively. Subsequently, the near-field array response for a given position  $\mathbf{r}$  can be expressed as

$$\mathbf{a}(\mathbf{r}) = \left[ e^{-jk_0\|\mathbf{r}-\mathbf{x}_{-\bar{M}}\|_2}, \dots, e^{-jk_0\|\mathbf{r}-\mathbf{x}_{+\bar{M}}\|_2} \right]^T, \quad (1)$$

where  $\|\cdot\|_2$  denotes the two-norm of a vector, and  $k_0 \triangleq \frac{2\pi}{\lambda}$  denotes the wavenumber. Therefore, according to the channel reciprocity, a round-trip channel matrix for position  $\mathbf{r}$  can be expressed by

$$\mathbf{H} = \beta \mathbf{a}(\mathbf{r}) \mathbf{a}^T(\mathbf{r}). \quad (2)$$

where  $\beta$  denotes the channel gain for the line-of-sight (LoS) channel. Here, the LoS channel gain can be calculated by  $\beta \triangleq \zeta_{\text{pathloss}}(f, 2r) G_t G_r$ , where  $G_t$  and  $G_r$  denote the transmit and receive array gain, respectively, and  $\zeta_{\text{pathloss}}(f, d) = \sqrt{\frac{c}{4\pi f d}} \frac{1}{d}$  denotes the pathloss with  $c$  denotes the speed of light. It is noted that, since the target is assumed to be within the radiating near-field region, the pathloss variances among antenna elements can be ignored and replaced by the pathloss of the center link [3].

Based on the modeling above, letting  $s[n] \in \mathbb{C}$  be the probing signal at the BS during the  $n$ -th coherent time block, the transmitted signal can be expressed as

$$\mathbf{x}[n] = \sqrt{P} \mathbf{w} s[n], \quad (3)$$

where  $P$  denotes the transmit power at the BS, and  $\mathbf{w} \in \mathbb{C}^{M \times 1}$  is the precoding vector, with unit power constraint  $|\mathbf{w}^H \mathbf{w}| = 1$ . After propagating through the round-trip channel, the received echo signal at the BS can be expressed as

$$\mathbf{y}[n] = \mathbf{H} \mathbf{x}[n] + \mathbf{z}[n], \quad (4)$$

where  $\mathbf{z}[n] \in \mathcal{CN}(\mathbf{0}_M, \sigma^2 \mathbf{I})$  denotes the complex Gaussian noise with zero mean and power  $\sigma^2$ , and  $\mathbf{I}$  denotes a identity matrix.

At high-frequency bands, the wireless channels are LoS-

dominated and non-line-of-sight (NLoS) assisted, as a result of severe scattering loss [15]. Therefore, in this work, we deal with the LoS case. Since the position of the target is embedded in the LoS channel, target localization is possible. However, the position is highly spread and unapparent in the space-domain channel  $\mathbf{H}$ , thus entailing complex signal processing algorithms to extract the position. To circumvent the high computational complexity introduced by the adoption of these algorithms, we transform the space-domain channel representation into the wavenumber domain, where the position can be apparently observed from the wavenumber domain.

### III. WAVENUMBER-DOMAIN LOCALIZATION ALGORITHM

In this section, we first introduce how to transform the space-domain channel into the wavenumber-domain, where the power-concentrated region of the channel is related to the target's position. Then, we elaborate a learning algorithm to learn such a relation, thus facilitating localization.

#### A. Wavenumber-Domain Transform

The core idea of wavenumber-domain transformation is to approximate the spherical wavefronts in the near-field region with a summation of planar wavefronts, whose form is coincidental with the four-dimensional (4D) Fourier transformation. Deviated from the canonical wavenumber-domain transformation presented by [16], the transceivers are co-located at the same position for our case, due to the full-duplex antenna architecture. Therefore, by converting the 4D Fourier transformation into a 2D counterpart, the entry at the intersection of the  $i$ -th column and  $j$ -th row in  $\mathbf{H}$  can be written as

$$[\mathbf{H}]_{i,j} = \frac{1}{4\pi^2} \iint_{\mathcal{D}_k} a(\mathbf{k}, \mathbf{x}_m) h_a(k_x, k_y) a^*(\mathbf{k}, \mathbf{x}_m) dx dy, \quad (5)$$

where  $h_a(k_x, k_y)$  denoted the coupling coefficient between transmit/receive array response  $a(\mathbf{k}, \mathbf{x}_m)$ , and integral region  $\mathcal{D}_k$  can be specified by  $\mathcal{D}_k \triangleq \{(k_x, k_y) \in \mathbb{R}^2 : k_x^2 + k_y^2 \leq k_0^2\}$ . In addition, the array response are specified by  $a(\mathbf{k}, \mathbf{x}_m) = \exp\{-jk^T \mathbf{x}_m\}$ , where  $\mathbf{k} \triangleq [k_x, k_y, \gamma(k_x, k_y)]^T$  and  $\gamma(a, b) \triangleq \sqrt{k_0^2 - a^2 - b^2}$ . To make (5) more tractable, we discretize the integral region  $[-k_0, +k_0]$  by sampling it with an interval  $2\pi/D$ . Moreover, since the whole system is located on the  $xz$  plane, the  $y$ -coordinate can be discarded. Jointly considering the two factors, the discretized region can be expressed as

$$\mathcal{G}_k \triangleq \{\varepsilon \in \mathbb{Z} : -\lceil Dk_0/2\pi \rceil \leq \varepsilon \leq +\lfloor Dk_0/2\pi \rfloor\}, \quad (6)$$

where  $\lceil \cdot \rceil$  and  $\lfloor \cdot \rfloor$  denote the ceiling and flooring functions, respectively. Accordingly, by denoting the cardinality of  $\mathcal{G}_k$  as  $|\mathcal{G}_k|$ , the integral form in (5) can be converted to a summation form (equivalently matrix multiplication):

$$\mathbf{H} \stackrel{(a)}{\simeq} M \sum_{i=1}^{|\mathcal{G}_k|} \sum_{j=1}^{|\mathcal{G}_k|} \mathbf{a}(i) [\mathbf{H}_a]_{i,j} \mathbf{a}^*(j) = M \mathbf{A} \mathbf{H}_a \mathbf{A}^H, \quad (7)$$

where  $\mathbf{H}_a$  denotes the wavenumber-domain channel representation, and the array response can be written as

$$\mathbf{a}(i) = \frac{1}{\sqrt{M}} \left[ \exp \left\{ -j \frac{2\pi i}{D} x_{-\bar{M}}^{(x)} \right\}, \dots, \exp \left\{ -j \frac{2\pi i}{D} x_{+\bar{M}}^{(x)} \right\} \right]^T. \quad (8)$$

Please note that the approximation of (a) is very accurate for a normalized size  $D$  larger than a few wavelength  $\lambda$  [17]. Since the wavenumber-domain transformation matrix (WTM)  $\mathbf{A} \triangleq [\mathbf{a}(1), \dots, \mathbf{a}(|\mathcal{G}_k|)]$  is semi-unitary, indicating that  $\mathbf{A}^H \mathbf{A} = \mathbf{I}$  holds, we can obtain the wavenumber-domain channel via:

$$\mathbf{H}_a = \frac{1}{M} \mathbf{A}^H \mathbf{H} \mathbf{A}. \quad (9)$$

It is important to note that the wavenumber-domain channel  $\mathbf{H}_a$  is obtained by sampling wavenumber  $\mathbf{k}$ , which is directional. Therefore, the power-concentrated region of  $\mathbf{H}_a$  is correlated with the direction of the signal. More importantly, in near-field systems, the non-negligible aperture size causes the distance dimension to spread across different directions, which in turn facilitates the acquisition of distance information. Therefore, we will present a learning algorithm to learn this relation,

### B. Target Localization Algorithm

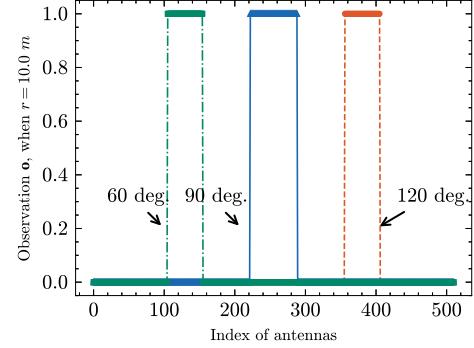
1) *Prepossessing*: In the first stage, the BS will transmit a probing signal that is specially designed to obtain the target's location. In this work, we utilize only a single snapshot (or, equivalently, a coherent time block) for localization. Therefore, for the brevity of notations, we drop the index of the coherent time block. It is worth noting that incorporating multiple snapshots can enhance the performance of our method, as averaging over several snapshots helps mitigate the impacts of noise. We utilize  $\mathbf{w}' = \mathbf{A}\mathbf{1}$  to transmit unit power probing signal  $s$ , where  $\mathbf{1} \in \mathbb{R}^{M \times 1}$  is an all-one vector. Under the unit power constraint, the transmit beamformer can be obtained by  $\mathbf{w} = \mathbf{w}' / \|\mathbf{w}'\|_2$ . After reception at the BS, the echo signal  $\mathbf{y}$  is processed by a matrix multiplication by  $\mathbf{A}^H$ , thus yielding

$$\begin{aligned} \tilde{\mathbf{o}} &= \frac{\mathbf{A}^H \mathbf{y}}{s} = \frac{\sqrt{P} \mathbf{A}^H \mathbf{H} \mathbf{w}'}{\|\mathbf{w}'\|_2} + \frac{\mathbf{A}^H \mathbf{z}}{s} \\ &= \frac{\sqrt{P} \mathbf{A}^H \mathbf{A} \mathbf{H}_a \mathbf{A}^H \mathbf{A} \mathbf{1}}{\|\mathbf{w}'\|_2} + \frac{\mathbf{A}^H \mathbf{z}}{s} \stackrel{(a)}{=} c \mathbf{H}_a \mathbf{1} + \tilde{\mathbf{z}}, \end{aligned} \quad (10)$$

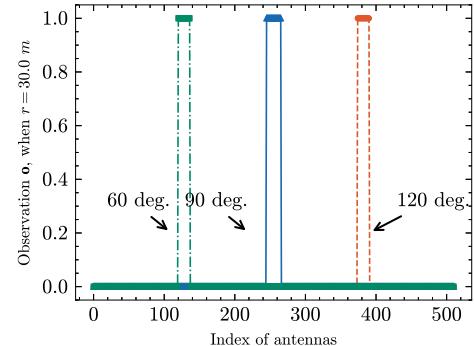
where  $\tilde{\mathbf{o}}$  is defined as the observation vector; step (a) is achieved by leveraging the semi-unitary property of WTM; and  $c = \frac{\sqrt{P}}{\|\mathbf{w}'\|_2}$  is a constant; and  $\tilde{\mathbf{z}} \triangleq \frac{\mathbf{A}^H \mathbf{z}}{s}$ . To extract power patterns in  $\tilde{\mathbf{o}}$  and normalize the observation to range  $[0, 1]$ , each entry of the normalized observation vector  $\mathbf{o} \in \mathbb{R}^{M \times 1}$  can be calculated using

$$[\mathbf{o}]_i = \mathbb{1} \left\{ \frac{[\tilde{\mathbf{o}}]_i - \min \{ [\tilde{\mathbf{o}}]_i \}}{\max \{ [\tilde{\mathbf{o}}]_i \} - \min \{ [\tilde{\mathbf{o}}]_i \}} > 0.5 \right\}, \quad (11)$$

where  $i \in \{1, 2, \dots, M\}$ ;  $\mathbb{1}\{\text{condition}\}$  is the indicator function, whose output is 1 when condition is true and 0 otherwise; and  $(\cdot)^{|\cdot|}$  denotes the element-wise modulus function. With normalization in (11), each of the entries is mapped to either



(a) Observation vector  $\mathbf{o}$  when  $r = 10 \text{ m}$



(b) Observation vector  $\mathbf{o}$  when  $r = 30 \text{ m}$

Fig. 2: An illustration of the received observation under different distances and angles of the target without noise and pathloss.

0 or 1, thus facilitating training by circumventing the bias introduced by pathloss and noise.

Fig. 2 depicts the received observation vector when the target is located at different positions. As we can see, the target's angle is related to where the power concentrates in  $\mathbf{o}$ , while the target's distance is correlated to the width of the power concentrated region. To capture this non-linear relationship between the power-concentrated region and the location of the target, we resort to machine learning tools.

2) *Neural Network (NN) Architecture*: Since this feature of the observation vector is localized in the power-concentrated region, we, therefore, employ a convolutional neural network (CNN)-based method to capture the feature's locality. The overall CNN-based algorithm is shown in Fig. 3. For the input layer, we propose a bi-directional architecture in order to let the NN learn the positional distribution better. Specifically, the input of the NN can be obtained by

$$\mathbf{O} = \text{Stack}\{\mathbf{o}, \mathbf{o}'\} \in \mathbb{R}^{2 \times M}, \quad (12)$$

where  $\text{Stack}\{\mathbf{a}, \mathbf{b}\}$  denotes the operation that same-dimension vectors  $\mathbf{a}$  and  $\mathbf{b}$  are stacked vertically, and  $\mathbf{o}'$  denotes the flipped version of  $\mathbf{o}$ . Then, the input will be fed to the Encoder, which is processed by a one-dimensional convolutional kernel. The Gaussian error linear units (GeLU) function is utilized for activation, which can avoid dead neuron issue of the conventional ReLU activation [18]. In addition, the MaxPool is utilized for downsampling the feature map, whose output is further flattened as a feature vector. In the sequel, the feature vector will be processed by the Decoder, which consists of

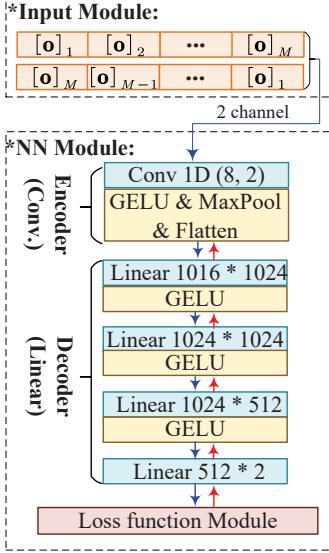


Fig. 3: An illustration of the proposed BiCNN method.

a fully connected structure, and finally mapped to a two-dimensional estimated target's location vector, i.e.,  $\hat{\mathbf{r}} = [\hat{x}, \hat{y}]^T$ . To train the NN, we utilize Huber loss as the loss function for estimation error, which is defined by

$$\mathcal{L}_\phi^{(1)}(\mathbf{r}, \hat{\mathbf{r}}; \rho) = \begin{cases} 0.5 \|\mathbf{r} - \hat{\mathbf{r}}\|_2^2, & \text{if } \|\mathbf{r} - \hat{\mathbf{r}}\|_2 \leq \rho \\ \rho \|\mathbf{r} - \hat{\mathbf{r}}\|_2 - 0.5\rho, & \text{otherwise} \end{cases} \quad (13)$$

where  $\rho$  is a hyper parameter and  $\phi$  denotes the vector containing all the trainable parameter. The core idea of (13) is: 1) when the estimation error is small, i.e.,  $\|\mathbf{r} - \hat{\mathbf{r}}\|_2 \leq \rho$ , a quadratic penalty will be imposed to force the NN to reduce the small errors; and 2) otherwise, a linear penalty will be imposed to make NN less susceptible to large outliers. By balancing the sensitivity to outliers via  $\rho$ , the optimization landscape can be smoother compared to the mean squared error (MSE) and the mean absolute error (MAE) criteria. Further, to circumvent the over-fitting problem, we introduce L2 regularization to the loss function, i.e.,

$$\mathcal{L}_\phi^{(2)} = \mu \sum_{i \in \mathcal{I}} [\phi]_i, \quad (14)$$

where  $\mu > 0$  denotes the weight of this term. In fact,  $\mathcal{L}_\phi^{(2)}$  in (14) will penalize more complexity of the NN, thus overcoming the overfitting issue. Jointly considering (13) and (14), the overall loss function and the corresponding parameter updating rule can be written as

$$\mathcal{L}_\phi = \mathcal{L}_\phi^{(1)}(\mathbf{r}, \hat{\mathbf{r}}; \rho) + \mathcal{L}_\phi^{(2)}, \quad (15)$$

$$\phi = \phi - w \nabla \mathcal{L}_\phi, \quad (16)$$

where  $w > 0$  denotes the learning rate. Therefore, the whole BiCNN can be seen as a function that maps the observations to the target location, i.e.,  $\mathcal{F}(\cdot; \phi) : \mathbb{R}^{2 \times M} \mapsto \mathbb{R}^{2 \times 1}$ .

3) *Training & Testing of BiCNN*: In general, we adopt the offline training and online implementation scheme, aiming at lowering the computational complexity. In the training process, we adopt an exponentially decaying learning rate, indicating the learning rate will decay by  $\alpha$  after each epoch. By doing

---

#### Algorithm 1: BiCNN-Based Target Localization

---

**Input:** Wavenumber-domain transformation matrix  $\mathbf{A}$  and pre-trained BiCNN  $\mathcal{F}(\cdot; \phi)$ .  
**Output:** Estimated position of the target  $\hat{\mathbf{r}}$ .  
// Transmit Probing Signal:  
1 Transmit symbol  $s$  using beamformer  
 $\mathbf{w} = \mathbf{A}\mathbf{1}/\|\mathbf{A}\mathbf{1}\|_2$ , and receive the echo signal  $\mathbf{y}$ ;  
// Echo Signal Prepossessing:  
2 Combine the echo signal  $\mathbf{y}$  using  $\mathbf{A}^H$  to obtain the raw observation  $\tilde{\mathbf{o}}$ ;  
3 Normalize  $\tilde{\mathbf{o}}$  using (11) to obtain  $\mathbf{o}$ ;  
4 Stack the  $\mathbf{o}$  and a flipped version  $\mathbf{o}'$  to form  $\mathbf{O}$ ;  
// BiCNN Estimation Step:  
5 Feed  $\mathbf{O}$  into the pre-trained BiCNN  $\mathcal{F}(\cdot; \phi)$  to obtain the estimated target's location  $\hat{\mathbf{r}}$ ;  
6 **return** Estimated target's location  $\hat{\mathbf{r}}$

---

so, the trainable parameters can be updated in a coarse-to-fine manner. Therefore, the learning rate transition from the  $k$ -th to the  $(k+1)$ -th epoch can be expressed as  $w_{k+1} = \alpha w_k$ . After being offline trained, the implementation procedure is summarized by **Algorithm 1**

#### IV. SIMULATION RESULTS

This section presents the numerical results that validate the effectiveness of the proposed BiCNN method. The following parameter settings are consistently applied throughout our simulations unless otherwise specified. For physical-layer parameters, the transmit power is set to  $P = 30$  dBm, while the noise power spectral density is  $-174$  dBm/Hz. The system functions at 28 GHz, with a 10 kHz bandwidth. The BS is assumed to be equipped with  $M = 511$  antennas with half-wavelength spacing, and the number of targets is one. Array gains are set to  $G_t = 10^{1.5}$  and  $G_r = 10^{0.5}$ . For the training process of BiCNN, its structure is illustrated by Fig. 3, in which Conv1D(8, 2) means a one-dimensional convolution operation with 8 output channels with a kernel size 2, and Linear  $M * N$  indicates a linear layer with  $M$ -dimensional input and  $N$ -dimensional output. We discrete an angle region  $[\pi/4, 3\pi/4]$  with an interval 0.01 and a distance region  $[8 \text{ m}, 35 \text{ m}]$  with an interval of 0.01 to generate dataset, in which 70%, 20%, and 10% are utilized for training, validation, and testing, respectively. Adaptive moment estimation (Adam) is used for optimization with a learning rate  $w = 0.001$  with a decaying factor  $\alpha = 0.98$ . The rest hyper-parameters are set to  $\mu = 10^{-5}$  and  $\rho = 1.0$ .

In Fig. 4, we compare the training process of the proposed BiCNN method with other benchmarks, i.e., ‘‘Naive CNN’’ and ‘‘Naive deep neural network (DNN)’’. In addition, ‘‘RC’’ in the legend of Fig. 4 represents the number of relative trainable parameters that each NN has compared to the DNN. This metric is directly related to the training complexity. Compared to the ‘‘Naive CNN’’, which doesn't include a flipped observation vector as an input channel, BiCNN can capture the locality of the observation and achieve a lower Huber loss (or equivalently higher accuracy). Compared to the ‘‘Naive DNN’’, which replaces the convolution encoder with a

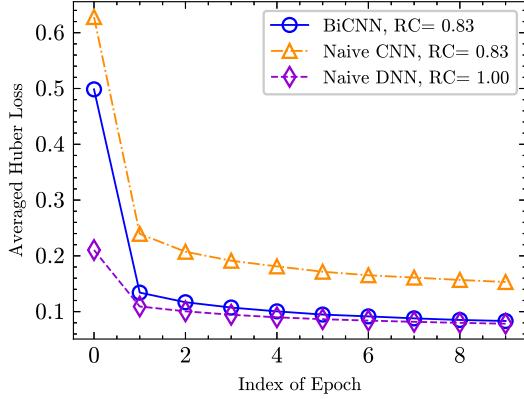


Fig. 4: Learning progress of BiCNN compared to the benchmarks

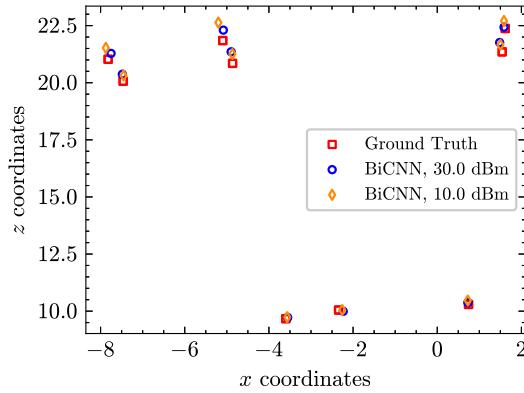


Fig. 5: An illustration of BiCNN's performance on the real-channel with different transmit power through 100 Monte Carlo simulation.

linear layer, BiCNN can compete with accuracy, while with a lower number of trainable parameters, thereby striking a good balance between performance and complexity.

In Fig. 5, the performance of the pre-trained BiCNN on the real channel, where noise and pathloss are included. This figure depicts the generalization ability of our method. As we can observe from this figure, the proposed BiCNN can achieve a high estimation accuracy, without traversing any pre-defined grids. However, as the distance between BS and the target grows larger, the estimation error will increase. In addition, as the transmit power increases by 20 dBm, the performance can be slightly elevated, which shows the good robustness of BiCNN to the variance of transmit power.

In Tab. I, we compare the conventional 2D MUSIC algorithm with the proposed BiCNN method. The MUSIC algorithm follows a two-step procedure: 1) obtaining the signal and noise subspaces with a computational complexity  $\mathcal{O}(M^3)$ , and 2) finding peaks on a discrete 2D coordinate system with a computational complexity dependent on the number of grids. In this table, RMSE stands for rooted mean square error of the localization, and all the results are obtained through 100 Monte Carlo simulations. As shown by Tab. I, the proposed BiCNN method can eliminate the eigendecomposition in the first step and the grid searching in the second step of MUSIC. More importantly, our method can achieve decent accuracy while significantly reducing the running time.

TABLE I: Comparison with 2D MUSIC algorithm with  $P = 30$  dBm.

Method	# Grids	RMSE	Avg. Run Time
MUSIC	10	3.8925 m	0.4510 s
	100	0.6786 m	0.4270 s
	1000	0.2109 m	0.5565 s
BiCNN	N/A	0.2439 m	0.0037 s

## V. CONCLUSION

A low-complexity wavenumber-domain method has been proposed for near-field localization. Specifically, gridless BiCNN is utilized to map the non-linear relationship between the observation and the target's location. In addition, a bi-directional structure is proposed to boost NN's performance with lower trainable parameters. Simulation results demonstrate that BiCNN can achieve a decent localization accuracy with much lower computational complexity, compared to conventional 2D MUSIC. The generalization of this method can be a promising direction for future research.

## REFERENCES

- [1] Z. Zheng *et al.*, “Localization of mixed near-field and far-field sources using symmetric double-nested arrays.” *IEEE Trans. Antennas Propag.*, vol. 67, no. 11, pp. 7059–7070, Jul. 2019.
- [2] Q. Yang *et al.*, “Beam focusing for near-field multi-user localization,” *arXiv:2407.20264*, 2024.
- [3] Z. Wang *et al.*, “Near-field integrated sensing and communications,” *IEEE Commun. Lett.*, vol. 27, no. 8, pp. 2048–2052, May 2023.
- [4] Y. Liu *et al.*, “Near-field communications: A tutorial review,” *IEEE Open J. Commun. Soc.*, vol. 4, pp. 1999–2049, Aug. 2023.
- [5] J. He *et al.*, “Mixed near-field and far-field source localization based on exact spatial propagation geometry,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3540–3551, Mar. 2021.
- [6] H. Luo *et al.*, “6D radar sensing and tracking in monostatic integrated sensing and communications system,” *arXiv:2312.16441*, 2023.
- [7] Y.-D. Huang *et al.*, “Near-field multiple source localization by passive sensor array,” *IEEE Trans. Antennas Propag.*, vol. 39, no. 7, pp. 968–975, Jul. 1991.
- [8] H. Li *et al.*, “Near-field integrated sensing, positioning, and communication: A downlink and uplink framework,” *IEEE J. Sel. Areas Commun.*, pp. 1–1, early access, 2024.
- [9] K. Qu *et al.*, “Two-stage beamspace MUSIC-based near-field channel estimation for hybrid XL-MIMO,” *IEEE Commun. Lett.*, vol. 28, no. 8, pp. 1949–1953, Aug. 2024.
- [10] X. Zhang *et al.*, “Localization of near-field sources: A reduced-dimension MUSIC algorithm,” *IEEE Commun. Lett.*, vol. 22, no. 7, pp. 1422–1425, Jul. 2018.
- [11] X. Cao *et al.*, “A low-complexity near-field localization method based on electric field model,” *IEEE Commun. Lett.*, vol. 28, no. 2, pp. 288–292, Feb. 2024.
- [12] M. Pesavento *et al.*, “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.
- [13] P. Ramezani, Özlem Tuğfe Demir, and E. Björnson, “Localization in massive MIMO networks: From near-field to far-field,” *arXiv:2402.07644*, 2024.
- [14] C. Ouyang *et al.*, “On the impact of reactive region on the near-field channel gain,” *IEEE Commun. Lett.*, early access, Jul. 2024.
- [15] A. Tang *et al.*, “Line-of-sight extra-large MIMO systems with angular-domain processing: Channel representation and transceiver architecture,” *IEEE Trans. Commun.*, vol. 72, no. 1, pp. 570–584, Oct. 2024.
- [16] A. Pizzo *et al.*, “Fourier plane-wave series expansion for holographic MIMO communications,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 6890–6905, Mar. 2022.
- [17] ———, “Spatially-stationary model for holographic MIMO small-scale fading,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 9, pp. 1964–1979, Jun. 2020.
- [18] D. Hendrycks *et al.*, “Gaussian error linear units (GELUs),” *arXiv:1606.08415*, 2023.