IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 6, 2020, accepted July 15, 2020, date of publication July 28, 2020, date of current version August 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3012582

Deep Neural Network for Estimation of Direction of Arrival With Antenna Array

MIN CHEN[1,2], YI GONG[2], (Senior Member, IEEE), AND XINGPENG MAO[1,3], (Member, IEEE)

[1]School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China
[2]Shenzhen Engineering Laboratory of Intelligent Information Processing for IoT, Southern University of Science and Technology, Shenzhen 518055, China
[3]Key Laboratory of Marine Environmental Monitoring and Information Processing, Ministry of Industry and Information Technology, Harbin 150001, China

Corresponding authors: Yi Gong (gongy@sustech.edu.cn) and Xingpeng Mao (mxp@hit.edu.cn)

This work was supported by the Key Program of National Natural Science Foundation of China under Grant 61831009.

ABSTRACT For many object tracking systems, how to quickly and efficiently estimate the direction of arrival (DOA) of radio waves impinging on the antenna array is an urgent task. In this paper, a new efficient DOA estimation approach based on the deep neural networks (DNN) is proposed, in which a nonlinear mapping that relates the outputs of the receiving antennas with its associated DOAs is learned by using the DNN-based network. The novel network architecture is divided into two phases, the detection phase and the DOA estimation phase. Additional detection network dramatically reduces the size of the training set and the process of the training data preparation is discussed in detail. After finishing the training phase, the corresponding DOAs can be identified based on current input data during testing phase. It has been shown that the proposed method can not only achieve reasonably high DOA estimation accuracy, but also reduce the computational complexity required by traditional superresolution DOA estimation methods such as multiple signal classification (MUSIC) and estimation of signal parameters via rotation invariance (ESPRIT). The computer simulation results are performed to investigate the generalization and effectiveness of the proposed approach in different scenarios.

INDEX TERMS Deep neural networks (DNN), detection network, direction of arrival (DOA) estimation network, testing process, training data preparation process.

I. INTRODUCTION

In modern electromagnetic research, there has been a growing research interest in the development of mobile communication devices [1]–[5]. One of the critical problems associated with this area is that the number of users that can actually interact at the same time with the base station is very high. Therefore, the need to develop efficient methods which are able to track the desired users is urgent. Massive multiple input multiple output (MIMO) system seems to be a powerful solution for theoretically enhancing the capacity of a communication system and mitigating inter-symbol interference by simply implementing additional antennas [6]–[10].

Adaptive array technique [11] generates radiation patterns of the array with maximum values toward the mobiles of interest while other sources of interference are null and the system is able to track these mobiles in real time. The first step in doing this task is the estimation of the direction of arrival (DOA) of the radio waves impinging on the antenna. Other applications of the DOA estimation include satellite mobile communication systems, astronomical observation and telemetry. A number of common methods have been proposed in the literature to solve the DOA estimation problem. Among them, the maximum likelihood [12] and the methods exploiting the eigen structure of the correlation matrix of received signals such as multiple signal classification (MUSIC) algorithm [13] and estimation of signal parameters via rotation invariance (ESPRIT) algorithm [14], [15] have been successfully applied. In particular, these methods although in general require high computational complexity due to the need for the eigenvectors of signal subspace and noise subspace to be estimated accurately, have been widely studied because they can be applied to arrays with arbitrary geometries and provide high resolution for signals with small angular separation. Meanwhile, many recent papers have derived different schemes based on the nonlinear optimization for superresolution DOA estimation [16]–[20]. The formulated optimization problems have generally been

The associate editor coordinating the review of this manuscript and approving it for publication was Yue Zhang.

non-convex, and they are usually solved using techniques that require a number of iterative computations, as a result, conventional techniques are not efficient enough to achieve superresolution DOA estimation. What is more, if a mapping from signal directions to sensor outputs during data collection is already known, the aforementioned methods can deal with the problem of DOA estimation as a inverse mapping from sensor outputs to signal directions by matching this inverse mapping with the preciously formed mapping. The consistency between these two mappings has a great influence on the performance of these methods.

In recent decades, the promising technique called machine learning (ML) has been proposed as a successful candidate to solve the DOA estimation problem in several array processing applications [21]–[26]. For example, methods based on the use of support vector regression (SVR) [21], [22] and radial basis function (RBF) [26] have also been efficiently applied for the problem of DOA estimation by establishing training data sets of the possible configurations of the impinging sources first, and then deriving a mapping from array outputs to signal directions. The derived mapping is then used on test data to estimate or predict signal directions not included in the learning phase through generalization. It has been shown that the RBF- and SVR-based DOA estimation methods have the capability to reduce the computation complexity, implement in real-time easily, and perform comparably with the superresolution methods in experiments [22]. However, a main drawback of the aforementioned supervised-based schemes is that the performance degrades rapidly or even model cannot work when dealing with unknown configurations, e.g. array number or source number changing. Therefore, it has to retrain the model with new data, which is a huge amount of work. To solve this problem, a generalization of the algorithm was introduced in [27], in such a way that the system would be able to track an arbitrary number of sources with any angular separation without prior knowledge of the number of sources. Furthermore, a method which combined the MUSIC algorithm with several ML techniques for the DOA estimation was also taken into account in [28]. Compared with the results obtained by using the MUSIC on its own, the DOA estimation performance improved with the help of ML.

Deep learning [29], a typical branch of the machine learning technique, has been applied to various application fields along with the fast developing intelligent algorithms, and its very high capability in solving complex nonlinear problem is now widely-known. However, the term ''deep learning'' did not come into common usage until around 2008. The recent growing interest has been sparked largely by the success of big data technique, in which deep learning allows automatically learning multiple levels of representations of the underlying distribution of the data to be modeled. More recently, deep learning which uses deep neural network (DNN) proposed by G. E. Hinton in 2006, has become a promising new area of work in the field of machine learning [30]. DNN has many advantages: DNN extracts features layer by layer and combines low-level features to form high-level features, which can find distributed expression of data; although the training of a DNN may take some time, it is well suited to real-time operations because a trained DNN model with moderate number of layers can be used with a low computation time. DNN enters into a flourishing period.

In the past few years, DNN has gradually become a promising tool in solving difficult wireless communication problems, such as resource allocation [31], [32], channel decoding [33], [34], MIMO [35], [36], cognitive radio network (CRN) [37], and channel estimation [38]–[40]. Furthermore, DNN algorithms have also been successfully applied to predictive data analysis and signal processing with many applications in varied areas of study. For instance, paper [41] applied DNN to DOA estimation and evaluated the estimation performance under a scenario where two equal-power and uncorrelated signals arriving on a uniform linear array. Authors in [42] combined classification with regression in a single DNN. Most of the signal processing algorithms have the inherent capability to solve nonlinear problems such as nonlinear classification, regression, information retrieval, and so on. Though DNN has acquired some success, it is still in its infancy, and many problems have not been solved yet.

In this paper, a novel framework that performs both detection and DOA estimation, which is based on DNN, is presented and the design aspects are studied under a scenario where multiple uncorrelated narrowband signals arriving from different directions are incident on a uniform linear array. From the perspective of machine learning, the DOA estimation problem is dealt with as a nonlinear mapping that relates the outputs of the receiving antennas with its associated DOAs by means of the DNN-based network. The research is not related to a particular application, which means that no exact performance criteria are specified. The main contributions of this paper are summarized as follows.

1) In our research, the DOA estimation approach based on DNN that does not need to provide any additional information is a combination of the detection network and the DOA estimation network. Additional detection network attached in our structure makes it possible to reduce the size of the training sets and to individually train several DNNs that correspond to different position sectors.
2) Starting from the knowledge of several input/output samples, the internal parameters of the framework are determined in the offline training phase. After that, the network yields excellent generalization performance. For example, our approach performs well in response to input signals that have not been initially included in the learning phase through generalization and the DOA estimations of new signals are performed in a very short time.
3) The results of simulation indicate that the DNN-based DOA estimation not only has excellent performance on DOA estimation, but also fast convergence rate. Subsequently, the estimation results are compared with those

obtained by applying other robust algorithms mentioned above on the same measurements. The extensive simulation results and comparison also demonstrate that the proposed scheme has made a major breakthrough in terms of detection accuracy and running speed.

The remaining sections of this paper are organized as follows. Section II presents the mathematical formulation of the received signal model and the DNN structure. Section III reports the DNN-based DOA estimation system model design. Section IV describes the training process used in this proposed algorithm. Section V provides several numerical simulation results to evaluate the performance of the proposed scheme by comparing with traditional DOA estimation schemes. Section VI summarizes this paper.

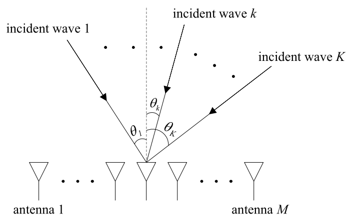

FIGURE 1. Configuration of the antenna array.

II. MATHEMATICAL FORMULATION

A. RECEIVED SIGNAL MODEL

As shown in Fig. 1, a uniform linear array composed by M elements with inter-element spacing d is used to receive K electromagnetic waves (which are supposed to be narrowband plane waves with a complex amplitude $s_k(t)$ and a wavelength λ) transmitted by sources located at angles $\theta_k, k = 1, \ldots, K$ and a time t. The received signal at the ith array element, $i = 1, \ldots, M$, can be expressed as

$$x_i(t) = \sum_{k=1}^{K} s_k(t) e^{-j\frac{2\pi}{\lambda}(i-1)d sin\theta_k} + n_i(t) \tag{1}$$

where $n_i(t)$ is the noise signal received at the ith element of the array.

Defining a $M \times K$ steering matrix $\boldsymbol{A}$ whose generic element is given by

$$a_{i,k} = e^{-j\frac{2\pi}{\lambda}(i-1)d sin\theta_k} \tag{2}$$

Using vector notation, we can rewrite (1) as

$$\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{n}(t) \tag{3}$$

where $\boldsymbol{x}(t)$, $\boldsymbol{n}(t)$, and $\boldsymbol{s}(t)$ are given by

$$\boldsymbol{x}(t) = [x_1(t), x_2(t), \ldots, x_M(t)]^T \tag{4}$$
$$\boldsymbol{n}(t) = [n_1(t), n_2(t), \ldots, n_M(t)]^T \tag{5}$$
$$\boldsymbol{s}(t) = [s_1(t), s_2(t), \ldots, s_K(t)]^T \tag{6}$$

In (4)-(6) the superscript $[\cdot]^T$ denotes the transpose of the matrix.

As widely recognized in most of literatures [43], [44], array processing algorithms do not work directly on the actual outputs of the array elements for direction of arrival estimation purposes, but they use the correlation matrix. Therefore, the $M \times M$ spatial correlation matrix of the received signals can be expressed as

$$\boldsymbol{R}_x = E\left[\boldsymbol{x}(t)\boldsymbol{x}^H(t)\right] = \boldsymbol{A}\boldsymbol{R}_s\boldsymbol{A}^H + \boldsymbol{R}_n \tag{7}$$

where $E[\cdot]$ and $[\cdot]^H$ indicate the statistical expectation and the Hermitian transpose, respectively. $\boldsymbol{R}_s$ and $\boldsymbol{R}_n$ are $K \times K$ source signal and $M \times M$ noise signal correlation matrices, respectively, and are given by

$$\boldsymbol{R}_s = E\left[\boldsymbol{s}(t)\boldsymbol{s}^H(t)\right] \tag{8}$$
$$\boldsymbol{R}_n = E\left[\boldsymbol{n}(t)\boldsymbol{n}^H(t)\right] \tag{9}$$

The noise signals received at the different array elements are assumed to be samples from statistically independent Gaussian white noise with zero mean and variance σ^2 and also independent of $\boldsymbol{s}(t)$. Thus, we have

$$\boldsymbol{R}_n = \sigma^2\boldsymbol{I} \tag{10}$$

where $\boldsymbol{I}$ is the $M \times M$ identity matrix. Then, we can rewrite (7) as

$$\boldsymbol{R}_x = \boldsymbol{A}E\left[\boldsymbol{s}(t)\boldsymbol{s}^H(t)\right]\boldsymbol{A}^H + \sigma^2\boldsymbol{I} \tag{11}$$

In this paper, the correlation matrix $\boldsymbol{R}_x$ properly preprocessed the outputs of the array is used as input of our DNN.

Since the correlation matrix $\boldsymbol{R}_x$ is a Hermitian matrix, the upper and lower triangular parts have the same information. What's more, it has been found [21] that the upper triangular part of this matrix is sufficient to estimate the DOAs of the arriving waves with a quite good accuracy. In our design, the input vector to the input layer of the DNN is the elements of $\boldsymbol{R}_x$ belonging to the upper triangular part that can be organized in an M^2–dimensional vector denoted by $\boldsymbol{y}$, given by

$$\boldsymbol{y} = [r_{1,1}, r_{2,2}, \ldots, r_{M,M}, R(r_{1,2}), I(r_{1,2}), R(r_{1,3}), I(r_{1,3}), \ldots, I(r_{M-1,M})]^T \tag{12}$$

where $r_{h,k}$ is the (h, k)th element of the correlation matrix $\boldsymbol{R}_x$, and $R[\cdot]$ and $I[\cdot]$ denote the real and imaginary parts of a complex-valued entity, respectively.

Since the neural network does not deal with complex numbers directly, the each complex-valued entity of $\boldsymbol{R}_x$ is considered to be two dimensional real values except for the diagonal elements.

B. DEEP NEURAL NETWORK STRUCTURE

The simplest neural network model is the single layer neural network that involves J linear combinations of the input variable $c_1, \ldots, c_D$ in the form

$$a_j = \sum_{i=1}^{D} w_{ji}c_i + b_j \tag{13}$$

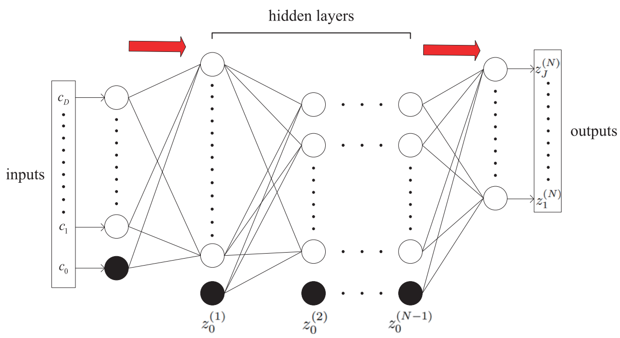

FIGURE 2. **Architecture of a multiple-layered deep neural network.**

where w_{ji}, b_j, a_j are referred to as weight, bias, and activation respectively. Each of activations is then transformed using an appropriate activation function $h(\cdot)$ to give a set of network outputs z_j,

$$z_j = h(a_j), \quad j = 1, \ldots, J \tag{14}$$

where J is the total number of outputs.

The bias parameters in (13) can be absorbed into the set of weight parameters by defining an additional input variable c_0 whose value is clamped at $c_0 = 1$, and the corresponding weight $w_{j0} = b_j$, so that (13) takes the following formulation

$$a_j = \sum_{i=0}^{D} w_{ji} c_i \tag{15}$$

We can combine these relations to give the overall network function which takes the form

$$z_j = h\left(\sum_{i=0}^{D} w_{ji} c_i\right) \tag{16}$$

Using vector notation, we can rewrite (14) and (15) in a matrix form as

$$\boldsymbol{z} = h\left(\boldsymbol{a}\right) \tag{17}$$

$$\boldsymbol{a} = \boldsymbol{W}\boldsymbol{c} \tag{18}$$

where

$$\boldsymbol{z} = [z_1, z_2, \ldots, z_J]^T \tag{19}$$

$$\boldsymbol{a} = [a_1, a_2, \ldots, a_J]^T \tag{20}$$

$$\boldsymbol{c} = [1, c_1, c_2, \ldots, c_D]^T \tag{21}$$

$$\boldsymbol{W} = \begin{bmatrix} w_{1,0} & w_{1,1} & \cdots & w_{1,D} \\ w_{2,0} & w_{2,1} & \cdots & w_{2,D} \\ \vdots & \vdots & & \vdots \\ w_{J,0} & w_{J,1} & \cdots & w_{J,D} \end{bmatrix} \tag{22}$$

The actual neural network has two or more hidden layers as shown in Fig. 2, we can similarly absorb the nth layer bias into the nth layer weights, so that the output of the nth layer can be expressed as

$$\boldsymbol{z}^{(n)} = h^{(n)}\left(\boldsymbol{a}^{(n)}\right) \tag{23}$$

$$\boldsymbol{a}^{(n)} = \boldsymbol{W}^{(n)} \boldsymbol{z}^{(n-1)} \tag{24}$$

in which the initial parameter $\boldsymbol{z}^{(0)} = \boldsymbol{c}$ and the overall network function becomes

$$\boldsymbol{z}^{(N)} = h^{(N)}\left(\boldsymbol{W}^{(N)}\left(\cdots h^{(2)}\left(\boldsymbol{W}^{(2)}\left(h^{(1)}\left(\boldsymbol{W}^{(1)}\boldsymbol{c}\right)\right)\right)\cdots\right)\right) \tag{25}$$

III. PROPOSED DEEP NEURAL NETWORKS-BASED DIRECTION OF ARRIVAL ESTIMATION SYSTEM DESIGN

In this paper, a comprehensive study is conducted to carry out the DOA estimation based on the DNNs. The computational problem of the DOA estimation can be treated from a different point of view. In this work, a fully connected DNN architecture is exploited to address the DOA estimation problem, where the aim is to learn a mapping from the observed antenna array signals to the DOAs of the impinging waves. In order to achieve a significant gain, a massively huge amount of training data is required in general. To overcome this problem, we introduce a detection network which may reduce the size of training set. The network architecture is divided into two phases, the detection phase and the DOA estimation phase. We will describe the details of our work in the following section.

A. DETECTION NETWORK

The detection network is used to divide the search area of the antenna array into different position sectors to detect signals radiating from sources in each sector. In our case, a Multiple Perception (MLP)-based detection network is considered because it can handle any dimensional inputs without making any assumption to the input data distribution. The detection network structure is composed of three layers, one input layer (the number of nodes is the same as the input vector dimension in our case), only one hidden layer, and an output layer, as shown in Fig. 3.

The elements of correlation matrix belonging to the upper triangular part of the received signals, $\boldsymbol{y}$, are directly fed into the detection network. The weight matrix between input vector and hidden layer is $\boldsymbol{W}_1^{(d)} \in \mathbb{R}^{H \times M^2}$, where H denotes the number of hidden nodes. For the detection outputs, the MLP-based detection network aims to make a discrete decision about the presence of a source in a certain direction. The outputs $\boldsymbol{h}$ of the hidden layer are processed by using the weight matrix $\boldsymbol{W}_2^{(d)} \in \mathbb{R}^{Q \times H}$, where Q denotes the number of output nodes, and then output a temporary vector $\boldsymbol{t}$. After that, $\boldsymbol{t} = \left[\boldsymbol{t}_1^T, \ldots, \boldsymbol{t}_P^T\right]^T$ is divided into P groups in which the pth group is $\boldsymbol{t}_p = [t_p^1, \ldots, t_p^q]^T$. In order to follow the additive rule, the activation functions are both defined as identity functions. The above description can be expressed using the following equations

$$\begin{cases} \boldsymbol{h} = \boldsymbol{W}_1^{(d)}\boldsymbol{y} + \boldsymbol{b}_1^{(d)}, \\ \boldsymbol{t} = \boldsymbol{W}_2^{(d)}\boldsymbol{h} + \boldsymbol{b}_2^{(d)} \end{cases} \tag{26}$$

where $\boldsymbol{b}_1^{(d)} \in \mathbb{R}^{H \times 1}$ and $\boldsymbol{b}_2^{(d)} \in \mathbb{R}^{Q \times 1}$ represent bias vectors at input layer and hidden layer, respectively.

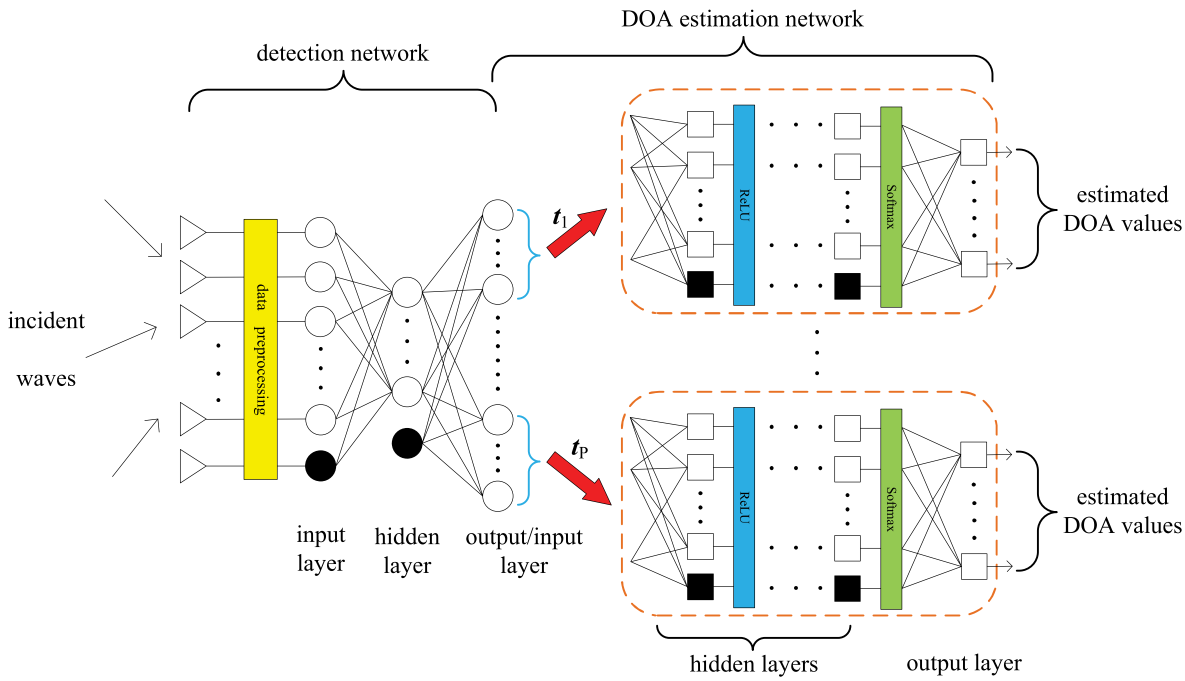

FIGURE 3. Block diagram of the considered DNN-based DOA estimation algorithm with detection network and DOA estimation network.

B. DOA ESTIMATION NETWORK

Once the detection process is accomplished, the corresponding DOA estimation networks can be activated in order to estimate the DOAs of the signals. The problem of DOA estimation is formulated as a K-class classification problem, where each class corresponds to a possible DOA value in the set $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$.

The structure of the DNN-based DOA classifier is shown in Fig. 3. The input of the DOA estimation network is formed by aligning the detection part outputs as a column vector, and it has the same number of nodes as the detection part output nodes in our case. The number of output nodes of the DOA estimation network depends on the array geometry, a required angle resolution and search range. Each output corresponds to a certain unique angle within the search range. For example, for an uniform linear array the DOA range lies between $[\theta_{min}, \theta_{max}]$, and with an angle resolution of $\Delta\theta$, the total number of output nodes is $Z = (\theta_{max} - \theta_{min}) / \Delta\theta + 1$. Supposing the DOA estimation network has $L-1$ fully feed forward connected hidden layers from one layer to the next layer, $\mathcal{L} = \{0, \ldots, L\}$ represents the set of layers, where $l = 0$ and $l = L$ denote the input layer and output layer respectively. The number of nodes of each layer $l \in \mathcal{L}$ is denoted by n_l, and we have $n_0 = Q$ and $n_L = Z$. For each hidden layer, the output $\boldsymbol{o}_l \in \mathbb{R}^{n_l \times 1}$ is calculated as follows

$$\boldsymbol{o}_l = \begin{cases} ReLU\left(\boldsymbol{W}_l^{(e)}\boldsymbol{t}+\boldsymbol{b}_l^{(e)}\right), & l=1 \\ ReLU\left(\boldsymbol{W}_l^{(e)}\boldsymbol{o}_{l-1}+\boldsymbol{b}_l^{(e)}\right), & l \in \{2, \ldots, L-1\} \end{cases} \tag{27}$$

where $\boldsymbol{o}_{l-1} \in \mathbb{R}^{n_{l-1} \times 1}$ is the output of the $(l-1)$th layer, $\boldsymbol{W}_l^{(e)} \in \mathbb{R}^{n_l \times n_{l-1}}$ and $\boldsymbol{b}_l^{(e)} \in \mathbb{R}^{n_l \times 1}$ are respectively the weight matrix and bias vector at layer l. The activation function at the output of each hidden layer is the Rectified Linear Unit function, $ReLU\,(x) = \max\,(x, 0)$, which introduces nonlinearity to the network in order to estimate more than one DOAs for a given position sector. However, the output layer has a softmax activation function at the end, namely $\boldsymbol{o}_L = softmax\left(\boldsymbol{W}_L^{(e)}\boldsymbol{o}_{L-1} + \boldsymbol{b}_L^{(e)}\right)$. It is worth noting that the transformation from the input space to the hidden-unit space is nonlinear, whereas the transformation from the hidden layer to the output space is linear.

In conclusion, the array outputs are preprocessed to serve as the input vectors for the detection phase. The vectors processed during the detection phase are sent to the estimation phase which can reproduce the DOA estimate values. Note that the weights and bias vectors are represented in different forms just to make them easier to distinguish.

IV. SUPERVISED LEARNING POLICY OF THE PROPOSED ALGORITHM

Unlike the existing methods which mainly rely on the array geometry, the proposed DNN-based framework uses a set of given samples of the input/output values to learn the relationship between the received signals and the DOAs of radio waves as well as possible. It should be mentioned here that the optimal training scheme can be obtained by exhaustively enumerating all possible combinations of angles of arrival. However, this naive design causes the number of training data patterns to increase in proportion to $\mathcal{O}(M^K)$. To avoid this problem, we introduce the detection network and divide the DOA estimation network into multiple DNNs, which makes the estimated value of the DOA closer to the hypothesis value in a smaller position sector. In the following section, detection network training strategy and DOA estimation network training strategy will be described in details.

A. DETECTION NETWORK TRAINING POLICY

In this part, we consider the learning scheme for training the detection network and the parameters required for detection process are determined during the training phase. After the training phase, the network based on DNN is performed on the testing phase online. The training pairs consist of a set of given samples of the input/output values of the detection network, which are created by

$$\Phi = \left\{ \boldsymbol{y}\left(\theta_{\phi}\right), \boldsymbol{t}\left(\theta_{\phi}\right) \right\}_{\theta_{\phi}=\theta_{min}}^{\theta_{max}} \tag{28}$$

where θ_{ϕ} is assumed to be uniformly distributed in the range $[\theta_{min}, \theta_{max}]$ and $\boldsymbol{y}\left(\theta_{\phi}\right)$ is its corresponding correlation vector which can be calculated according to (11) and (12). When the signal from direction θ_{ϕ} belonging to the pth group is incident on the antenna array, the output of the detection network, $\boldsymbol{t}\left(\theta_{\phi}\right)$, is expressed as

$$\begin{aligned} \boldsymbol{t}\left(\theta_{\phi}\right) &= \left\{ \boldsymbol{t}_1^T, \ldots \boldsymbol{t}_p^T \ldots, \boldsymbol{t}_P^T \right\}^T \\ &= \left\{ \underbrace{[0, \ldots, 0]}_{q}, \ldots \boldsymbol{y}\left(\theta_{\phi}\right)^T \ldots, \underbrace{[0, \ldots, 0]}_{q} \right\}^T \end{aligned} \tag{29}$$

that is to say, the pth group output of the detection network is expected to be $\boldsymbol{y}\left(\theta_{\phi}\right)$, while the outputs of the other groups expected to be $\mathbf{0} = [0, \ldots, 0]^T \in \mathbb{R}^{q\times 1}$.

Starting from these training sample pairs, the weight matrices $\boldsymbol{W}^{(d)}$ and bias vectors $\boldsymbol{b}^{(d)}$ of the detection network are determined during the training phase by minimizing the loss function which can be defined as the difference between the hypothesis output and its predicted value, that is

$$R\left(\theta_{\phi}\right) = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\|^2 \tag{30}$$

where N is noted as the number of samples at each direction, and $\hat{\boldsymbol{t}}\left(\theta_{\phi}\right)$ is the predicted value. It is possible to evaluate the gradients of the error function $R\left(\theta_{\phi}\right)$ efficiently by means of the background procedure. In order to make the formulations look more compact, we omit the contants.

$$\begin{aligned} \frac{\partial R(\theta_{\phi})}{\partial \boldsymbol{W}_1^{(d)}} &= \sum_{n=1}^{N} \boldsymbol{W}_2^{(d)} \left[\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\right] \boldsymbol{y}\left(\theta_{\phi}\right) \\ \frac{\partial R(\theta_{\phi})}{\partial \boldsymbol{W}_2^{(d)}} &= \sum_{n=1}^{N} \boldsymbol{W}_1^{(d)} \left[\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\right] \left[\boldsymbol{y}\left(\theta_{\phi}\right) + \boldsymbol{b}_1^{(d)}\right] \\ \frac{\partial R(\theta_{\phi})}{\partial \boldsymbol{b}_1^{(d)}} &= \sum_{n=1}^{N} \boldsymbol{W}_2^{(d)} \left[\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\right] \\ \frac{\partial R(\theta_{\phi})}{\partial \boldsymbol{b}_2^{(d)}} &= \sum_{n=1}^{N} \left[\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\right] \end{aligned} \tag{31}$$

The sequential gradient descent is adopted to use the obtained gradients information, that is

$$\begin{aligned} \boldsymbol{W}^{(\tau+1)} &= \boldsymbol{W}^{(\tau)} - \eta \nabla_{\boldsymbol{W}} \|\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\|^2 \\ \boldsymbol{b}^{(\tau+1)} &= \boldsymbol{b}^{(\tau)} - \eta \nabla_{\boldsymbol{b}} \|\boldsymbol{t}\left(\theta_{\phi}\right) - \hat{\boldsymbol{t}}\left(\theta_{\phi}\right)\|^2 \end{aligned} \tag{32}$$

where the parameter $\eta>0$ is known as the learning rate and the τ labels the iteration step. After each such update, the gradients are re-evaluated for the new weight matrices $\boldsymbol{W}^{(\tau+1)}$ and new bias vectors $\boldsymbol{b}^{(\tau+1)}$ and the process repeated.

Once training process of the detection network is completed, the testing phase can be constructed by giving a specific received signal correlation vector and the estimated detection output can be obtained without requiring iterations.

B. DOA ESTIMATION NETWORK TRAINING POLICY

In this part, we would like to investigate how to use a set of training data to achieve reliable DOA estimation. In the training phase, a DOA classifier, namely a DNN in our case, is trained from sets of G training pairs, given by

$$\Omega\left(\theta_k\right) = \left\{ \boldsymbol{t}\left(\theta_k^g\right), \theta_k^g \right\}_{g=0}^{G-1} \tag{33}$$

where θ_k^g is the training angle of $\theta_k, k = 1, \ldots, K$ corresponding to the gth sample and $\boldsymbol{t}\left(\theta_k^g\right)$ is its detection output vector. Therefore, the nonlinear relationship between the detection network output and the corresponding DOA can be learned by training the DOA estimation network. In order to generalize the estimation capabilities of the DNN to unknown angular separations, in the learning phase, different angular separations between the training angles are considered. Because the detection network which can divide the search area of antenna array into smaller position sectors to reduce the training angles combinations has been introduced, we only consider two impinging signals (θ_1 and θ_2) with one angular separation ($\Delta\theta_{12} = \theta_2 - \theta_1$) in each classifier. Therefore, $\Omega\left(\theta_k\right)$ can be defined by the following angles of incidence

$$\begin{aligned} \theta_1^g &= \theta_{min} + \delta_\theta\, g\, \frac{\pi}{180} \\ \theta_2^g &= \theta_{min} + \delta_\theta\, g\, \frac{\pi}{180} + \Delta\theta_{12} \end{aligned} \tag{34}$$

where δ_θ is the angle distance between two adjacent DOAs of the training set. The DOA estimation network training is divided into two stages:

The first stage, forward propagation stage: taking an input vector $\boldsymbol{t}\left(\theta_k^g\right)$ from the training sample set, passing forward it through the network, and calculating the output of each layer.

The second stage, back propagation stage: through calculating the difference between the above network output $\hat{\theta}_k^g$ and the desired output θ_k^g, successively utilizing the background propagation algorithm to find the error derivatives of all the network parameters, and consequently updating the weights and biases values of the interconnections based on the sequential gradient descent method. The gradients of all the network parameters, which are needed for the back propagation based learning, can be obtained following the same calculation process as the detection network training.

The above training process continually adjusts all network parameters to minimize the value of loss function until convergence. The mean square error (MSE) loss function is taken into account during the training, which is defined as

$$E\left(\theta_k^g\right) = \frac{1}{N}\sum_{n=1}^{N} \|\theta_k^g - \hat{\theta}_k^g\|^2 \tag{35}$$

During testing, the detection network output vector is fed into the trained DNN-based DOA estimation network, which returns evaluated DOA values, from which the DOAs can be obtained. Our proposed scheme can also be used when the angular separations are not included in training set.

It is worth noting that after the training phase (performed offline), the real-time determination of DOA is less time consuming and the memory space is saved because only the coefficients of the DNNs need to be stored. The training process of the proposed algorithm is supervised and steps can be summarized in appendix.

V. SIMULATION RESULT AND ANALYSIS

In this section, the estimation performance of the proposed method has been evaluated by means of several computer simulations. The powerful Tensorflow [45] is introduced to design and process the DNN. All the computer simulations are performed in a PC equipped with 3.6 GHz processor and RAM of 8 GB.

A. SIMULATION SETUP

In the following simulations, a uniform linear array composed by $M = 10$ elements with half wavelength inter-element spacing was employed to receive signals from multiple sources. The center frequency was set to 2 GHz. Simulation data used for training and testing the proposed framework is created using the data model (3), as no real-world measurements were conducted within this paper. The received signals were supposed to be uncorrelated narrowband signals and had equal signal-to-noise ratio (SNR) of 10 dB. The correlation matrix was calculated from 400 snapshots of received signals.

For the detection neural network, the dimension of the input layer was set to 100 nodes, and that of the hidden layer was 50 nodes. The number of output layer units was the same as that of the input layer. The DOAs used for training were random variables of the uniformly distribution in the range $[-60°, +60°]$ with a sampling interval of $1°$ and the corresponding correlation matrices were computed according to (11). The search range of DOA was divided into $P = 2$ position sectors with equal spatial regions. Next, the training pairs which composed by the calculated correlation vectors and detection network outputs were used to train this neural network to detect the presence of signals in each spatial region. As the neural network was trained by considering 1000 samples per direction, the training set comprised 121000 samples in the simulations.

For the DOA estimation network, the dimension of the input layer was the same as the output layer size of the detection neural network because the signals processed by the detection neural network were subsequently presented to the DOA estimation network to estimate the DOA. The number of hidden layers was two where the first hidden layer had 50 nodes and the number of nodes on the second hidden layer was 30 in each classifier. It was assumed that the DOA of each plane wave fell within the range of $[-60°, +60°]$, and the required angle resolution was set to $1°$. Therefore, the number of output layer nodes was 121. In each classifier, the training samples were created by considering two signals separated by $\Delta = 2°$ where the DOAs of the first signal (denoted by θ) were generated by uniformly sampling in the search range $[-30°, +28°]$ with a sampling interval of $1°$, and the DOAs of the second signal were $\theta + \Delta$. For both networks, the learning rate was set to 0.001 while the batch size was 16.

B. DOA ESTIMATION

In this section, the simulation results confirm the generalization capabilities of the proposed method in several different cases by comparison with classical DOA estimation methods.

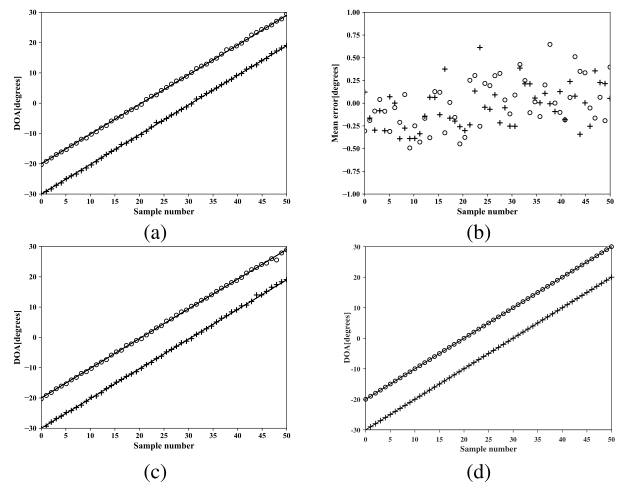

FIGURE 4. Comparison between the actual and estimated DOA values of two signals separated by $\Delta = 10°$ (not included in the training set). (a) DNN-based estimation approach. (b) Mean errors on the DNN-based estimation approach. (c) SVR-based estimation approach. (d) MUSIC-based estimation approach.

As described before, the each DOA estimation network was trained with two uniformly distributed random variables separated by $\Delta = 2°$ in the search range $[-60°,+60°]$. However, in the first experiment, the trained network has been tested with two uncorrelated narrowband signals arriving on the antenna array with an angular separation given by $\Delta = 10°$ (not included in the training set) where the DOAs are assumed to be uniformly distributed from $-30°$ with a step of $1°$ to $+30°$. Both signals have equal SNR of 10 dB. In Fig. 4, we use dashed lines to represent theoretical angles of arrival, and symbols to indicate their estimated data. Note that the abscissa in Fig. 4 refers to the number of samples not for the training phase but for the estimation one. The above two rules are true for Figs. 5-9 as well. As can be seen from Fig. 4(a), the estimated DOA values of our proposed method and the true values are very close. The results obtained from the MUSIC and SVR algorithms in the same scenarios are also shown in Fig. 4(c) and (d). The training data set of the SVR is the same as our defined classifiers. Also, Fig. 4(b) shows the behavior of mean errors versus the number of samples for assessing the accuracy of the DOA estimation and each reported simulation result is averaged over 200 Monte Carlo trials. It is evident that our proposed algorithm is able to estimate the DOAs, which are not included in the learning phase, with a quite good accuracy. Fig. 4 shows that the performance of our proposed method approaches that of the MUSIC and SVR algorithms and most of the mean square errors are lower than 0.5°.

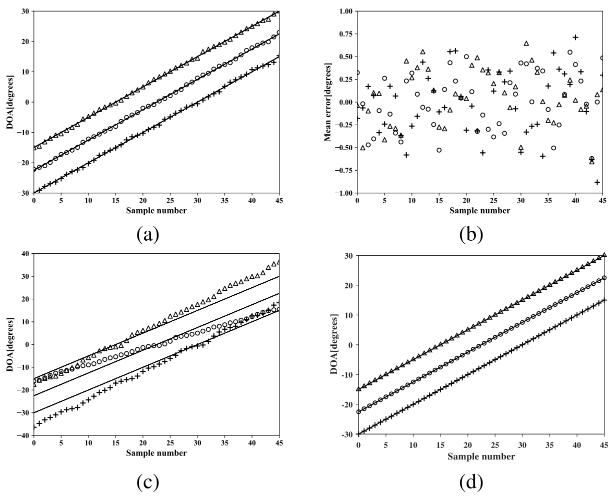

FIGURE 5. **Comparison between the actual and estimated DOA values of three signals separated by $\Delta = 7.5°$ (not included in the training set). (a) DNN-based estimation approach. (b) Mean errors on the DNN-based estimation approach. (c) SVR-based estimation approach. (d) MUSIC-based estimation approach.**

In the second experiment, Fig. 5 reports the estimated DOA values when the network is tested with three uncorrelated narrowband signals which are uniformly distributed random variables in the position sector [−30°, +30°] with the same angular separation $\Delta = 7.5°$ and same SNR. The actual values are also reported. From the result, we note that our approach provides very good DOA estimates and the estimation accuracy can be comparable with those drawn by the high resolution MUSIC algorithm without requiring the critical information of the number of active signals. However, the SVR algorithm does not work well when the number of signals is unknown in advance in the simulations. It is supposed that the SVR in general requires a good estimate of the number of active sources, which is often unavailable or difficult to obtain. Otherwise, the model has to be retrained overall to get the accurate DOA estimates.

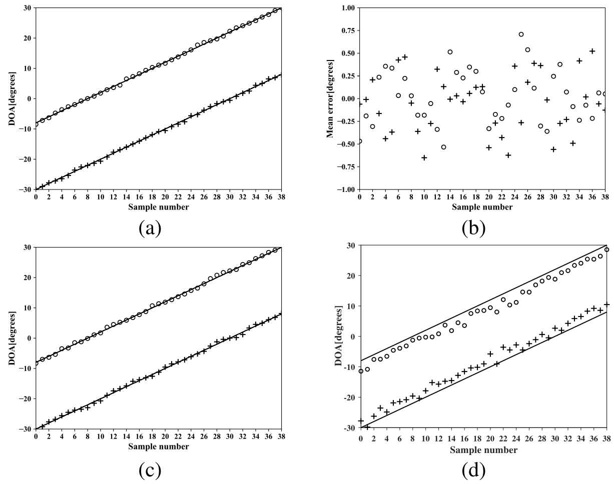

FIGURE 6. **Comparison between the actual and estimated DOA values of twelve signals (larger than the number of array elements) separated by 2°. For clarity, only show the first and the last signals. (a) DNN-based estimation approach. (b) Mean errors on the DNN-based estimation approach. (c) SVR-based estimation approach. (d) MUSIC-based estimation approach.**

In the third experiment, Fig. 6 shows a comparison between the estimated and true values for this array receiving twelve uncorrelated narrowband sources uniformly distributed in the sector [−30°,+30°] with the same 2° angular separation and same SNR. In this experiment, for clarity, we only show the first signal and the last signal. As can be seen from this figure, the MUSIC algorithm has a poor performance when the number of array elements is less than the number of sources, and significant errors occur. In comparison, the proposed network is basically able to resolve the DOAs of multiple sources although the results are a little bit of fluctuation. This behavior can be attributed to the fact that the MUSIC algorithm is based on signal subspace decomposition and it needs a full-rank estimate of the correlation matrix. Therefore, the maximum number of signals that an array can resolve is bounded by the number of its elements. The DNN-based approach is fully data-driven, therefore may not be puzzled by such problem.

It can be concluded from Figs. 4-6 that the proposed network is able to successfully retrieve the DOAs of the sources (not included in the training phase) with a quite good accuracy through generalization and yield satisfactory results without any extra information or any need to adopt a mathematical method. Meanwhile, the simulation results also illustrate that the proposed method provides better performance than traditional DOA estimation methods and other machine learning methods, especially when dealing with unknown configurations.

To study the effect of noise on the performance of the neural network, for comparison, the root mean square error (RMSE) of the DOA estimates of four kinds of algorithms (which are the standard MUSIC algorithm, SVR-based DOA estimation algorithm, RBF-based DOA estimation algorithm, and our proposed DNN-based DOA estimation algorithm)

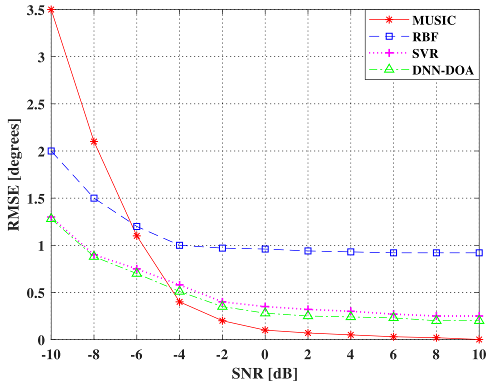

FIGURE 7. RMSE in DOA estimation (degrees) of the proposed DNN-based DOA estimation algorithm, standard MUSIC algorithm, SVR-based DOA estimation algorithm, and RBF-based DOA estimation algorithm as a function of SNR.

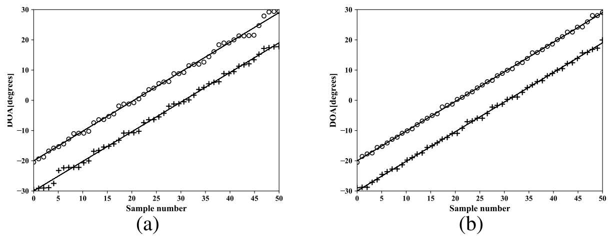

FIGURE 8. Comparison between the actual and estimated DOA values with smaller SNR values compared with the training settings. (a) SNR=-8 dB. (b) SNR=-4 dB.

trained and tested with different sets of SNR is shown in Fig. 7. The DNN-based DOA estimation algorithm is called DNN-DOA for short in this figure. The RMSE is defined as

$$RMSE = \sqrt{\frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}\left(\hat{\theta}_{k,n} - \theta_k\right)^2} \tag{36}$$

where N denotes the number of trials and $\hat{\theta}_{k,n}$ is the estimated value of the θ_k of the nth Monte Carlo trial for the kth electromagnetic wave. In order to observe the performance of our proposed DNN-based DOA estimation algorithm under small SNR in a more detailed way, the specific estimation results of our algorithm are also given in Fig. 8. In particular, the range of SNR is set from -10 dB to 10 dB in Fig. 7, and the estimation results obtained by DNN-DOA algorithm when the SNR $= -8$ dB and SNR $= -4$ dB are shown in Fig. 8. The other training settings are the same as the first experiment. From the results, the RMSE of all the DOA estimation algorithms reduces significantly as SNR increases. Fig. 7 clearly indicates that the MUSIC algorithm achieves a slightly higher estimation accuracy when SNR>-4 dB, but when SNR becomes small, the performances of the learning-based DOA estimation methods are much better. As can be seen in Fig. 8, although there was a slight fluctuation when SNR $= -8$ dB, the estimation results have been greatly improved when SNR $= -4$ dB. It can be observed from these figures that our proposed method shows a good robustness against noise even at low SNR and the results obtained by our proposed approach are also comparable with those provided by the MUSIC algorithm at the SNR of 10 dB. It is worth noting that the RMSE of the DOA estimates of all these algorithms is tending to be stable when SNR is higher than 2 dB.

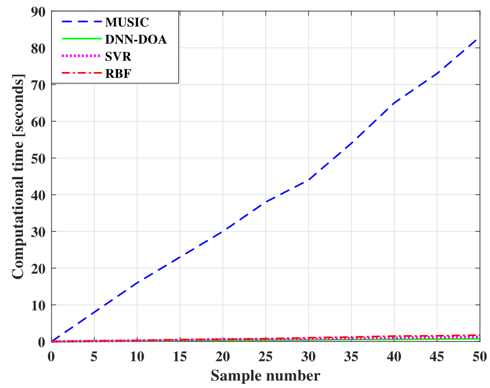

FIGURE 9. Computational time required by the learning-based DOA estimation algorithms and the standard MUSIC algorithm versus sample numbers.

C. COMPUTATIONAL COMPLEXITY

Finally, the computational time during the testing phase required by above mentioned four kinds of DOA estimation algorithms versus the sample numbers is plotted in Fig. 9. Note that, our proposed scheme requires an average of 2.5 minutes to train the detection network and about 5 minutes to train the DOA estimation network. After the training phase, the computation time to estimate a new DOA value is measured at 0.015 seconds which is sufficiently low to be used in practice. As reported in [21], the CPU time of the SVR-based method is 0.025 seconds which can be comparable with our approach. As can be seen from Fig. 9, the CPU time required by the MUSIC algorithm for DOA simulation is about 85 seconds when the number of samples is equal to 50, which is much longer than that of learning-based schemes. These learning-based schemes need less than a second in the same case. That is to say, the learning-based DOA estimation methods dramatically reduce the computational complexity and the memory space. Simulation results for run time and precision show that our DNN-based DOA estimation approach has made a major breakthrough in both detection accuracy and running speed.

VI. CONCLUSION

In this paper, we have presented a new approach based on DNNs for detecting and estimating the DOAs of radio waves. The network architecture has been constructed to address the DOA estimation problem by learning a mapping that relates the observed antenna array signals with its associated DOAs of the impinging waves. The detection network divided the search area of the antenna array into several sectors to detect signals radiating from sources in each sector, each of which

Algorithm 1 Training Steps in the Detection Stage:

- **Step 1**: Generate the set of DOAs which is uniformly distributed in the range $[-60°, +60°]$;
- **Step 2**: Calculate the observed signals using (1), estimate the correlation matrices of the array output vectors using (11), and form the detection network input vectors using (12);
- **Step 3**: Present input vectors to the detection network and evaluate the detection network outputs using (26);
- **Step 4**: Form detection network training pairs using (28);
- **Step 5**: The background propagation algorithm is used to evaluate the gradients of the error function using (31);
- **Step 6**: Update the weights and biases by the sequential gradient descent algorithm;
- **Step 7**: The weights and biases are determined by minimizing the error function.

Algorithm 2 Training Steps in the DOA Estimation Stage:

- **Step 1**: Generate the set of DOAs using (34);
- **Step 2**: Calculate the detection network output vectors by passing the set of DOAs through the detection network;
- **Step 3**: Form the DOA estimation network training pairs using (33);
- **Step 4**: Calculate the DOA estimation network outputs by forward propagation;
- **Step 5**: Through the above network outputs, the background propagation algorithm is used to find the error derivatives of the network parameters;
- **Step 6**: Update the weights and biases by the sequential gradient descent algorithm;
- **Step 7**: The weights and biases are determined by minimizing the error function.

corresponded to one DOA estimation network. According to the detection results, one or more DOA estimation networks could be activated to perform the DOA estimation. The detection network introduced in our architecture dramatically reduced the size of the training set. Starting from the knowledge of several input/output samples, the internal parameters of the framework were determined in the offline training phase, and then the corresponding DOA output values could be obtained based on the current input data during testing phase. The performance of this approach has been compared with the conventional DOA estimation algorithms in a couple of different scenarios. The simulation results have shown high estimation accuracy and the excellent generalization capabilities of our approach. The main advantage of the proposed approach was that it greatly reduced the computation time for the DOA estimation, which further verified its efficiency. The framework was evaluated on synthetically generated uncorrelated narrowband data, and a scenario with two correlated signals impinging on the array showed that the performance of the framework decreased gradually. Further research is required to determine whether alternative learning algorithms and advanced techniques can handle this problem.

APPENDIX A
TRAINING STEPS IN THE DETECTION STAGE

See Algorithm 1.

APPENDIX B
TRAINING STEPS IN THE DOA ESTIMATION STAGE

See Algorithm 2.

REFERENCES

[1] J. H. Winters, "Smart antennas for wireless systems," *IEEE Pers. Commun.*, vol. 5, no. 1, pp. 23–27, Feb. 1998.

[2] S. D. Blostein and H. Leib, "Multiple antenna systems: Their role and impact in future wireless access," *IEEE Commun. Mag.*, vol. 41, no. 7, pp. 94–101, Jul. 2003.

[3] K. Kyungjunq, T. K. Sarkar, H. Wang, and M. Salazar-Palma, "Direction of arrival estimation based on temporal and spatial processing using a direct data domain (D3) approach," *IEEE Trans. Antennas Propag.*, vol. 52, no. 2, pp. 533–541, Feb. 2004.

[4] N. Celik, W. Kim, M. F. Demirkol, M. F. Iskander, and R. Emrick, "Implementation and experimental verification of hybrid smart-antenna beamforming algorithm," *IEEE Antennas Wireless Propag. Lett.*, vol. 5, pp. 280–283, 2006.

[5] S.-S. Jeng, G. T. Okamoto, G. Xu, H.-P. Lin, and W. J. Vogel, "Experimental evaluation of smart antenna system performance for wireless communications," *IEEE Trans. Antennas Propag.*, vol. 46, no. 6, pp. 749–757, Jun. 1998.

[6] N. Garcia, H. Wymeersch, E. G. Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive MIMO," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2475–2487, May 2017.

[7] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[8] Y. Peng, F. Al-Hazemi, R. Boutaba, F. Tong, I.-S. Hwang, and C.-H. Youn, "Enhancing energy efficiency via cooperative MIMO in wireless sensor networks: State of the art and future research directions," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 47–53, Nov. 2017.

[9] H. Xie, B. Wang, F. Gao, and S. Jin, "A full-space spectrum-sharing strategy for massive MIMO cognitive radio systems," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 10, pp. 2537–2549, Oct. 2016.

[10] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?" *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 160–171, Feb. 2013.

[11] M. Mozingo, *Introduction to Adaptive Arrays*. New York, NY, USA: Wiley, 1980.

[12] A. G. Jaffer, "Maximum likelihood direction finding of stochastic sources: A separable solution," in *Proc. Int. Conf. Acoust., Speech, Signal Process. ICASSP*, Apr. 1988, pp. 2893–2896.

[13] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and cramer-rao bound," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 5, pp. 720–741, May 1989.

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

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

[16] R. O. Schmidt, "A new approach to geometry of range difference location," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-8, no. 6, pp. 821–835, Nov. 1972.

[17] H. Schau and A. Robinson, "Passive source localization employing intersecting spherical surfaces from time-of-arrival differences," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, no. 8, pp. 1223–1225, Aug. 1987.

[18] M. Wax and T. Kailath, "Optimum localization of multiple sources by passive arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 5, pp. 1210–1217, Oct. 1983.

[19] C. E. Chen, F. Lorenzelli, R. E. Hudson, and K. Yao, "Stochastic maximum-likelihood DOA estimation in the presence of unknown nonuniform noise," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3038–3044, Jul. 2008.
[20] F.-M. Han and X.-D. Zhang, "An ESPRIT-like algorithm for coherent DOA estimation," *IEEE Antennas Wireless Propag. Lett.*, vol. 4, pp. 443–446, 2005.
[21] M. Pastorino and A. Randazzo, "A smart antenna system for direction of arrival estimation based on a support vector regression," *IEEE Trans. Antennas Propag.*, vol. 53, no. 7, pp. 2161–2168, Jul. 2005.
[22] A. Randazzo, M. A. Abou-Khousa, M. Pastorino, and R. Zoughi, "Direction of arrival estimation based on support vector regression: Experimental validation and comparison with MUSIC," *IEEE Antennas Wireless Propag. Lett.*, vol. 6, pp. 379–382, 2007.
[23] A. Rawat, R. N. Yadav, and S. C. Shrivastava, "Neural network applications in smart antenna arrays: A review," *AEU Int. J. Electron. Commun.*, vol. 66, no. 11, pp. 903–912, Nov. 2012.
[24] K. Terabayashi, R. Natsuaki, and A. Hirose, "Ultrawideband direction-of-arrival estimation using complex-valued spatiotemporal neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 9, pp. 1727–1732, Sep. 2014.
[25] Y. Gao, D. Hu, Y. Chen, and Y. Ma, "Gridless 1-b DOA estimation exploiting SVM approach," *IEEE Commun. Lett.*, vol. 21, no. 10, pp. 2210–2213, Oct. 2017.
[26] A. H. El Zooghby, C. G. Christodoulou, and M. Georgiopoulos, "A neural network-based smart antenna for multiple source tracking," *IEEE Trans. Antennas Propag.*, vol. 48, no. 5, pp. 768–776, May 2000.
[27] A. H. El Zooghby, C. G. Christodoulou, and M. Georgiopoulos, "Performance of radial-basis function networks for direction of arrival estimation with antenna arrays," *IEEE Trans. Antennas Propag.*, vol. 45, no. 11, pp. 1611–1617, Nov. 1997.
[28] A. Khan, S. Wang, and Z. Zhu, "Angle-of-arrival estimation using an adaptive machine learning framework," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 294–297, Feb. 2019.
[29] G. E. Hinton, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[30] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1544, Jul. 2006.
[31] W. Lee, M. Kim, and D.-H. Cho, "Deep power control: Transmit power control scheme based on convolutional neural network," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1276–1279, Jun. 2018.
[32] F. Liang, C. Shen, W. Yu, and F. Wu, "Towards optimal power control via ensembling deep neural networks," 2018, *arXiv:1807.10025*. [Online]. Available: http://arxiv.org/abs/1807.10025
[33] F. Liang, C. Shen, and F. Wu, "An iterative BP-CNN architecture for channel decoding," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 144–159, Feb. 2018.
[34] M. Kim, N.-I. Kim, W. Lee, and D.-H. Cho, "Deep learning-aided SCMA," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 720–723, Apr. 2018.
[35] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.
[36] T. Wang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based CSI feedback approach for time-varying massive MIMO channels," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 416–419, Apr. 2019.
[37] S. Ayoubi, N. Limam, M. A. Salahuddin, N. Shahriar, R. Boutaba, F. Estrada-Solano, and O. M. Caicedo, "Machine learning for cognitive network management," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 158–165, Jan. 2018.
[38] K. Kim, J. Lee, and J. Choi, "Deep learning based pilot allocation scheme (DL-PAS) for 5G massive MIMO system," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 828–831, Apr. 2018.
[39] D. Neumann, T. Wiese, and W. Utschick, "Learning the MMSE channel estimator," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2905–2917, Jun. 2018.
[40] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.
[41] Y. Kase, T. Nishimura, T. Ohgane, Y. Ogawa, D. Kitayama, and Y. Kishiyama, "DOA estimation of two targets with deep learning," in *Proc. 15th Workshop Positioning, Navigat. Commun. (WPNC)*, Oct. 2018, pp. 1–5.
[42] O. Bialer, N. Garnett, and T. Tirer, "Performance advantages of deep neural networks for angle of arrival estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 3907–3911.
[43] M. S. Choi, G. Grosskopf, D. Rodhe, B. Kuhlow, G. Przyrembel, and H. Ehlers, "Experiments on DOA-estimation and beamforming for 60 GHz smart antennas," in *Proc. IEEE VTC Spring*, Jeju, South Korea, vol. 2, Apr. 2003, pp. 1041–1045.
[44] M. Pastorino and A. Randazzo, "Real-time SVM-based approach for localization of sources," in *Proc. IEEE Int. Workshop Imag. Syst. Techn. (IST)*, May 2004, pp. 2–6.
[45] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467

MIN CHEN received the B.S. degree from Shanxi Normal University, Linfen, China, in 2016, and the M.S. degree from the Harbin Institute of Technology, Harbin, China, in 2018, where she is currently pursuing the Ph.D. degree.

Her research interests include digital signal processing and device-free indoor localization.

YI GONG (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2002.

He was with the Hong Kong Applied Science and Technology Research Institute, Hong Kong, and Nanyang Technological University, Singapore. He is currently a Professor with the Southern University of Science and Technology, Shenzhen, China. His research interests include cellular networks, mobile computing, and signal processing for wireless communications and related applications.

Dr. Gong served on the Editorial Board for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, from 2006 to 2018.

XINGPENG MAO (Member, IEEE) was born in Liaoning, China, in 1972. He received the B.S. degree in radio electronics from Northeast Normal University, Changchun, China, in 1993, and the M.Eng. and Ph.D. degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 1999 and 2004, respectively.

He joined the Harbin Institute of Technology, as a Lecturer, in 1993, working on the design and development of radar and other electronic system. From 2005 to 2006, he was a Postdoctoral Fellow with the University of Waterloo, Waterloo, ON, Canada. He is currently a Professor with the Department of Electronic and Information Engineering, Harbin Institute of Technology. His current interests include signal processing in wireless communication and radar.

Dr. Mao is a Senior Member of the Chinese Institute of Electronics.