

DOA estimation of array signals based on convolutional sparse autoencoder under sparse prior

REN Jing, TAN Xiuhui, BAI Yanping*, WANG Peng, CHENG Rong, ZHANG Feng, XU Ting

School of Mathematics, North University of China, Taiyuan 030051, China

***Corresponding author:** BAI Yanping (baiyp666@163.com)

Received: December 22, 2025 **Revised:** January 31, 2026 **Accepted:** April 14, 2026

Abstract: The application of deep learning to direction of arrival (DOA) estimation is of great significance in the field of array signal processing. The use of deep learning for DOA estimation of vector hydrophone array usually directly inputs the covariance matrix of the signal as the signal feature into the network, but this method has limitations such as high data requirements and high computational complexity. This paper proposes a DOA estimation method for vector hydrophone array based on a convolutional sparse autoencoder under sparse prior conditions. This method adds an L1 norm regularization term to the convolutional layer of a convolutional autoencoder to achieve sparsity constraints, and establishes a convolutional sparse autoencoder. At the same time, a residual compensation mechanism is introduced to avoid overfitting and loss of details during the training process. Subsequently, the columns of the signal covariance matrix of the vector hydrophone array are treated as under-sampled noisy linear measurements of the spatial spectrum, and are input into a convolutional sparse autoencoder for feature extraction and reconstruction. Finally, the obtained features are used as inputs for training a convolutional neural network to achieve multi-source DOA estimation. Furthermore, to address the shortcomings of classification methods in off-grid situations, we propose a DOA regression estimation method based on the convolutional sparse autoencoder. The simulation results show that under complex conditions such as low signal-to-noise ratio and a small number of snapshots, the classification method proposed in this paper outperforms various deep learning algorithms and traditional algorithms mentioned in the literature in terms of estimation performance. In addition, the proposed regression method can further improve the DOA estimation performance in off-grid scenarios.

Key words: vector hydrophone array; direction of arrival estimation; sparse representation; convolutional sparse autoencoder; L1-norm regularization; regression

0 Introduction

Direction of arrival (DOA) estimation is a core research direction in the field of array signal processing. It has become a widely used technology in modern society, and is applied in fields such as navigation, military, aviation, and daily life^[1-3].

Early DOA estimation algorithms primarily focused on beamforming techniques, such as the conventional beamforming (CBF) algorithm^[4]. These algorithms applied the traditional Fourier transform to spatial spectrum estimation, but the angular resolution was limited by the Rayleigh limit. Due to the aperture limitation of the array, these algorithms cannot accurately distinguish targets within the same beamwidth. Schmidt proposed the multiple signal classification (MUSIC) algorithm^[5], while Roy et al. introduced the rotational invariance technique (ESPRIT) algorithm^[6]. Both are representative algorithms for subspace

classification. The MUSIC algorithm utilizes the orthogonality between the signal subspace and the noise subspace to decompose the signal covariance matrix, enabling rapid development of DOA estimation research. The ESPRIT algorithm estimates signal parameters by utilizing the rotational invariance between subspaces to obtain DOA. Many scholars have made improvements on the basis of the two subspace classification algorithms^[7-8]. Despite the implementation of super-resolution estimation techniques, accurately constructing spatial spectra remains a challenge when processing small snapshot data. In the DOA estimation algorithm based on subspace fitting, the maximum likelihood (ML) algorithm is representative^[9], which obtains the spatial spectrum by solving the non-linear direction estimation likelihood function, but requires a large amount of computation. With the development of sparse representation technology, many researchers transformed the DOA estimation problem into a sparse signal

reconstruction problem and proposed various DOA estimation methods based on sparse representation^[10]. However, the sparse representation class of methods also has issues, such as the difficulty of grid discretization.

Traditional DOA estimation methods are mostly model-driven, and model mismatch may occur under non-ideal conditions. With the advancement of deep learning theory, many researchers have applied deep learning to DOA estimation^[11-15]. Compared with traditional methods, data-driven deep learning DOA estimation methods avoid model mismatch under non-ideal conditions. By utilizing the powerful nonlinear modeling and automatic feature extraction capabilities of neural networks, as well as their adaptability to complex environments, a nonlinear relationship between received array signals and signal source DOA is established. Liu et al. used the real and imaginary parts of the signal covariance matrix as input data for DOA estimation in the network^[11]. Papageorgiou et al. input the real, imaginary, and phase information of the sample covariance matrix into a 2D convolutional neural network for feature extraction, resulting in a significant improvement in DOA estimation performance under low signal-to-noise ratio conditions^[12]. Ren et al. used a complex-valued convolutional denoising autoencoder for denoising and combined it with a complex-valued convolutional neural network for DOA estimation. They improve the estimation accuracy and generalization ability in low signal-to-noise ratio and low snapshot scenarios^[13]. Qin replaced convolutional neural networks with residual neural networks for feature extraction, which can learn more complex feature representations^[14]. Zheng et al. proposed using the original in-phase and quadrature components of the signal as inputs to the network, which simplifies the process of calculating the covariance matrix^[15]. Due to the fact that DOA estimation is considered as a sparse linear inverse problem and its application in compressive sensing^[16-18], Wu et al. regarded the DOA estimation problem as a sparse linear inverse problem in compressive sensing and proposed a deep inverse network that utilized 1D convolutional layers to learn the inverse transform from array output to DOA spectrum^[19]. Qin used 1D convolution to capture local features of spatial spectra based on sparse prior, and used fully connected layers to estimate DOA. This method is superior to DOA estimation methods that input covariance matrices in complex environments^[20], but it does not consider the sparsity of the incident sources and the possible overfitting phenomenon in feature extraction.

We propose a method for DOA estimation of vector hydrophone arrays using 1D convolutional sparse

autoencoder (CSAE) under sparse prior. The simulation results show that our proposed classification method outperforms convolutional neural network (CNN) algorithms, CM-CNN algorithms, and traditional algorithms such as MUSIC, CBF, and OMP in DOA estimation under low signal-to-noise ratio and a small number of snapshots. The proposed regression method overcomes the limitations of classification methods in off-grid scenarios, and the DOA estimation performance of the regression method is better than that of classification methods in off-grid scenarios.

1 Signal model

1.1 Vector hydrophone array signal model

Suppose that there are M elements in the vector hydrophone array, and the adjacent-element spacing is d . There are K far-field narrowband signal sources $s_k(t)$, the angle of incidence is θ , $\theta = [\theta_1, \theta_2, \dots, \theta_K]$. Fig. 1 shows the signal reception model of a uniform linear array (ULA) of vector hydrophones when only the horizontal azimuth is considered.

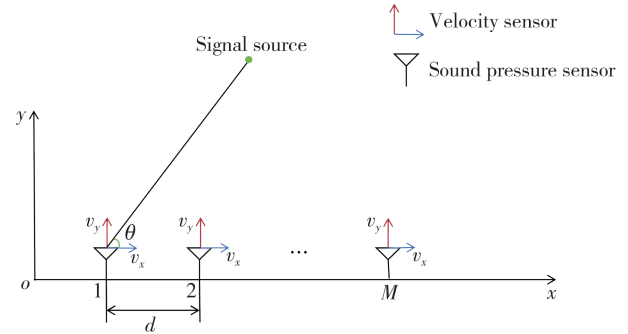


Fig. 1 ULA signal reception model for vector hydrophone

In Fig. 1, v_x, v_y is the component of the vibration velocity vector $\mathbf{v}(t)$ along the x axis and y axis. At time t , the sound pressure scalar $p(t)$ and the vibration velocity vector $\mathbf{v}(t)$ are related as

$$\mathbf{v}(t) = [\xi \cos \theta + \eta \sin \theta] p(t), \quad (1)$$

where ξ, η are unit vectors of orthogonal coordinates. The connection between sound pressure $p(t)$ and v_x, v_y is expressed as

$$\begin{cases} p(t) = s(t), \\ v_x(t) = p(t) \cos \theta, \\ v_y(t) = p(t) \sin \theta. \end{cases} \quad (2)$$

Thus, the output yielded by one vector hydrophone is

$$\begin{aligned} \mathbf{z}(t) &= [p(t), v_x(t), v_y(t)]^T = \\ &= [s(t), s(t) \cos \theta, s(t) \sin \theta]^T = \\ &= s(t) [1, \cos \theta, \sin \theta]^T. \end{aligned} \quad (3)$$

Considering the vector hydrophone array received signal model, if the first element is regarded as the reference element, the output $\mathbf{x}_m(t) \in \mathbb{C}^{3 \times 1}$ of the m -th element at time t is

$$\mathbf{x}_m(t) = \sum_{k=1}^K a_m(\theta_k) [1, \cos \theta_k, \sin \theta_k]^T s_k(t) + \mathbf{n}_m(t), \quad (4)$$

where $a_m(\theta_k) = \exp(-j2\pi d(m-1)(\cos \theta_k)/\lambda)$; λ is the signal wavelength; $s_k(t)$ indicates the k -th received signal; $\mathbf{n}_m(t) \in \mathbb{C}^{3 \times 1}$ stands for the received noise of the m -th array element.

The vector hydrophone array output $\mathbf{X}(t)$ is

$$\mathbf{X}(t) = \mathbf{A}(\theta)\mathbf{S}(t) + \mathbf{N}(t), \quad (5)$$

where $\mathbf{X}(t) = [\mathbf{x}_1(t), \mathbf{x}_2(t), \dots, \mathbf{x}_M(t)]^T \in \mathbb{C}^{3M \times 1}$; $\mathbf{S}(t) = [s_1(t), s_2(t), \dots, s_K(t)]^T \in \mathbb{C}^{K \times 1}$ is called the source vector; $\mathbf{N}(t) = [\mathbf{n}_1(t), \mathbf{n}_2(t), \dots, \mathbf{n}_M(t)]^T \in \mathbb{C}^{3M \times 1}$ is the array-received noise vector; $\mathbf{A}(\theta) \in \mathbb{C}^{3M \times K}$ is the signal steering matrix defined by

$$\mathbf{A}(\theta) = [\mathbf{a}(\theta_1) \otimes \mathbf{u}_1, \mathbf{a}(\theta_2) \otimes \mathbf{u}_2, \dots, \mathbf{a}(\theta_K) \otimes \mathbf{u}_K], \quad (6)$$

where $\otimes$ denotes the Kronecker product; $\mathbf{u}_k = [1, \cos \theta_k, \sin \theta_k]^T$; $k = 1, 2, \dots, K$; $\mathbf{a}(\theta_k) = [a_1(\theta_k), a_2(\theta_k), \dots, a_M(\theta_k)]^T = [1, \exp(-j2\pi d(\cos \theta_k)/\lambda), \dots, \exp(-j2\pi d(M-1)(\cos \theta_k)/\lambda)]^T$.

1.2 Vector hydrophone array sparse signal model

Let $\mathbf{f} = [-\phi_{\max}, \dots, -\phi_a, 0, \phi_a, \dots, \phi_{\max}]^T$ be the discrete set of directions obtained by sampling the incident signal, with a total of L grids, where ϕ_a denotes the sampling step, $L \gg K$. The real source θ is contained in $\mathbf{f}$. The output matrix $\mathbf{X}(t)$ of the vector hydrophone array can be reformulated in an over-complete manner and represented as

$$\mathbf{X}(t) = \sum_{l=1}^L (\mathbf{a}(\phi_l) \otimes \mathbf{u}_l) \bar{s}_l(t) + \mathbf{n}(t) = \bar{\mathbf{A}}(\mathbf{f})\bar{\mathbf{S}}(t) + \mathbf{N}(t), \quad (7)$$

where $\bar{\mathbf{A}}(\mathbf{f}) = [\mathbf{a}(\phi_1) \otimes \mathbf{u}_1, \mathbf{a}(\phi_2) \otimes \mathbf{u}_2, \dots, \mathbf{a}(\phi_L) \otimes \mathbf{u}_L] \in \mathbb{C}^{3M \times L}$ is the spatial over-complete dictionary set. $\bar{\mathbf{S}}(t) =$

$$\mathbf{W}_m = [(\mathbf{a}(\phi_1) \otimes \mathbf{u}_1)(\mathbf{a}(\phi_1) \otimes \mathbf{u}_1)^H \mathbf{e}_1, \dots, (\mathbf{a}(\phi_L) \otimes \mathbf{u}_L)(\mathbf{a}(\phi_L) \otimes \mathbf{u}_L)^H \mathbf{e}_L] \in \mathbb{C}^{3M \times L}.$$

Arrange the columns $\mathbf{R}$ sequentially from left to right to obtain a new $(3M)^2 \times 1$ dimensional column vector $\mathbf{z}$, that is

$$\mathbf{z} = [\mathbf{z}_1; \dots; \mathbf{z}_M] = \text{vec}(\mathbf{R}) = \mathbf{W}\boldsymbol{\eta} + \sigma^2 \mathbf{e}, \quad (12)$$

where $\mathbf{W} = [\mathbf{W}_1; \mathbf{W}_2; \dots; \mathbf{W}_M] \in \mathbb{C}^{(3M)^2 \times L}$; $\text{vec}(\cdot)$ denotes the straightening of the matrix by columns; $\mathbf{e} = [\mathbf{e}_1; \mathbf{e}_2; \dots; \mathbf{e}_M] \in \mathbb{R}^{(3M)^2 \times 1}$; $[\cdot; \cdot; \dots; \cdot]$ denotes the vertical

$[\bar{s}_1(t), \bar{s}_2(t), \dots, \bar{s}_L(t)]^T \in \mathbb{C}^{L \times 1}$ represents the vector of the received sparse signal. If $\phi_l = \theta_k$, then $\bar{s}_l(t) = s_k(t)$; otherwise, it is 0.

The over-complete representation of the array output for the multiple snapshots condition corresponding to Eq. (7) is given by

$$\hat{\mathbf{X}} = \bar{\mathbf{A}}(\mathbf{f})\hat{\mathbf{S}} + \hat{\mathbf{N}}, \quad (8)$$

where $\hat{\mathbf{S}} = [\bar{\mathbf{S}}(t_1), \bar{\mathbf{S}}(t_2), \dots, \bar{\mathbf{S}}(t_N)] \in \mathbb{C}^{L \times N}$ is the signal waveform matrix under the multiple snapshots condition; $\hat{\mathbf{N}} = [\mathbf{n}_1(t_1), \mathbf{n}_2(t_2), \dots, \mathbf{n}_M(t_N)]^T \in \mathbb{C}^{3M \times N}$ is the Gaussian white noise under the multiple snapshots condition, and the positions of non-zero elements in the signal waveform vector $\bar{\mathbf{S}}(t_n)$ at different time instants are all associated with the direction θ of the incident signal.

When the signals are independent of each other, the value of the signal covariance matrix $\mathbf{R}_{\hat{\mathbf{S}}}$ is obtained by

$$\mathbf{R}_{\hat{\mathbf{S}}} = \mathbb{E}[\hat{\mathbf{S}}\hat{\mathbf{S}}^H] = \text{diag}[\eta_1, \dots, \eta_L], \quad (9)$$

where if $\phi_l = \theta_k$, there is $\eta_l = \mathbb{E}[s_k(t)s_k^H(t)]$; otherwise, $\eta_l = 0$. The results show that the spatial distribution power spectrum $\boldsymbol{\eta} = [\eta_1, \dots, \eta_L]^T \in \mathbb{R}^{L \times 1}$ of the signal has the same sparse structure as the signal waveform matrix $\hat{\mathbf{S}}$ in the spatial domain. Therefore, reconstructing $\hat{\mathbf{S}}$ is equivalent to reconstructing the spatial spectrum $\boldsymbol{\eta}$. The reduction in the dimension of the optimization objective ($\mathbb{C}^{L \times N} \rightarrow \mathbb{R}^{L \times 1}$) is beneficial for reducing the difficulty of reconstruction.

The covariance matrix $\mathbf{R}$ of the array output is

$$\begin{aligned} \mathbf{R} &= \mathbb{E}[\hat{\mathbf{X}}\hat{\mathbf{X}}^H] = \\ &= \sum_{l=1}^L (\mathbf{a}(\phi_l) \otimes \mathbf{u}_l)(\mathbf{a}(\phi_l) \otimes \mathbf{u}_l)^H \eta_l + \sigma^2 \mathbf{I}_{3M} = \\ &= \bar{\mathbf{A}}(\mathbf{f})\mathbf{R}_{\hat{\mathbf{S}}}\bar{\mathbf{A}}^H(\mathbf{f}) + \sigma^2 \mathbf{I}_{3M}. \end{aligned} \quad (10)$$

So the m -th column vector of $\mathbf{R}$ is

$$\begin{aligned} \mathbf{z}_m &= \mathbf{R}\mathbf{e}_m = \\ &= \sum_{l=1}^L (\mathbf{a}(\phi_l) \otimes \mathbf{u}_l)(\mathbf{a}(\phi_l) \otimes \mathbf{u}_l)^H \eta_l \mathbf{e}_m + \sigma^2 \mathbf{I}_{3M} \mathbf{e}_m = \\ &= \mathbf{W}_m \boldsymbol{\eta} + \sigma^2 \mathbf{e}_m, \end{aligned} \quad (11)$$

where $\mathbf{e}_m$ is a vector of dimension $3M \times 1$; the m -th element is 1, and the rest are 0, and

splice of the matrix or vector in parentheses; $\mathbf{z}$ is the covariance vector.

Eqs. (11) and (12) show that the covariance vector $\mathbf{z}$ can be considered as a linear observation of the spatial power spectrum $\boldsymbol{\eta}$, and $\mathbf{W}$ is the observation matrix.

In practice, the array output's covariance matrix is usually estimated using a limited set of T observations

with the expression

$$\hat{\mathbf{R}} = \frac{1}{T} \sum_{t=1}^T \mathbf{X}(t) \mathbf{X}(t)^H. \quad (13)$$

Affected by the limited observation snapshots, there is a certain error between the estimated value $\hat{\mathbf{R}}$ of the array output covariance matrix and the theoretical value $\mathbf{R}$.

$$\hat{\mathbf{R}} = \mathbf{R} + \Delta\mathbf{R}, \quad (14)$$

where $\Delta\mathbf{R}$ is the covariance estimation error matrix. The corresponding covariance vector estimation $\hat{\mathbf{z}}$ is

$$\begin{aligned} \hat{\mathbf{z}} &= \text{vec}(\hat{\mathbf{R}}) = \text{vec}(\mathbf{R}) + \text{vec}(\Delta\mathbf{R}) = \mathbf{z} + \Delta\mathbf{z} = \\ &\mathbf{W}\boldsymbol{\eta} + \sigma^2 \mathbf{e} + \Delta\mathbf{z} = \mathbf{W}\boldsymbol{\eta} + \boldsymbol{\xi}, \end{aligned} \quad (15)$$

where $\hat{\mathbf{z}} \in \mathbb{C}^{(3M)^2 \times 1}$; $\Delta\mathbf{z} = \text{vec}(\Delta\mathbf{R})$ is the estimation error of the covariance vector, which is obtained by straightening the estimation error $\Delta\mathbf{R}$ of the covariance matrix column by column; $\boldsymbol{\xi} = \sigma^2 \mathbf{e} + \Delta\mathbf{z}$ represents the disturbance term plus the estimation error.

On this basis, the task of retrieving the spatial spectrum $\boldsymbol{\eta}$ from a linear measure $\hat{\mathbf{z}}$ is essentially a sparse linear inverse problem due to the sparse nature of $\boldsymbol{\eta}$ and the linear relationship between $\hat{\mathbf{z}}$ and $\boldsymbol{\eta}$. The key to solving this problem is to utilize the sparsity of $\boldsymbol{\eta}$ to get a more accurate $\boldsymbol{\eta}$ as possible from the limited measurement data $\hat{\mathbf{z}}$. To realize the inverse mapping from $\hat{\mathbf{z}}$ to $\boldsymbol{\eta}$, the observation matrix $\mathbf{W}$ is utilized to obtain the spectral proxy $\tilde{\mathbf{z}} = \mathbf{W}^H \hat{\mathbf{z}}^{[19]}$.

2 Autoencoder and convolutional autoencoder

Autoencoder (AE) ^[21] is constructed using fully connected layers, and it is an unsupervised learning model mainly used for dimensionality reduction or feature learning of data. The AE maps the input data to the hidden layer using an encoder part and then decodes the output of the hidden layer using a decoder, thus completing the data reconstruction. Fig.2 illustrates the structure of the AE.

From the structure diagram of the AE, assuming an input signal $\mathbf{p}$, the encoder of the AE first encodes this input signal $\mathbf{p}$. The input data is then mapped to the hidden layer via the encoder part, and this process can be denoted as

$$\mathbf{h} = s_1(\mathbf{W}_1 \mathbf{p} + \mathbf{b}_1), \quad (16)$$

where $s_1(\bullet)$ is the activation function; $\mathbf{W}_1$ is the weight matrix that connects the hidden layer to the output layer, and $\mathbf{b}_1$ is the bias vector of the hidden layer. Then the decoder part of AE decodes the output $\mathbf{h}$ of the hidden

layer, and the decoding process is denoted as

$$\mathbf{g} = s_2(\mathbf{W}_2 \mathbf{h} + \mathbf{b}_2), \quad (17)$$

where $s_2(\bullet)$ is the activation function; $\mathbf{W}_2$ is the weight matrix for the decoder part connecting the hidden layer to the output layer, while $\mathbf{b}_2$ represents the bias vector. The final output vector $\mathbf{g}$ is of the same size as the input vector $\mathbf{p}$.

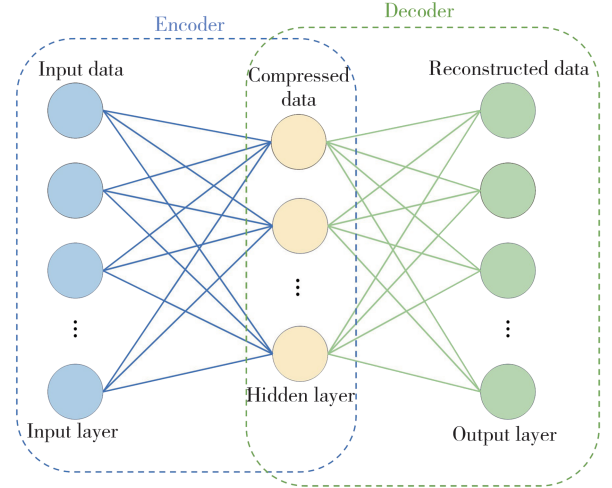


Fig. 2 AE structure diagram

The structure of AE limits its application scope, and this is especially true when the input data is large in scale or has a high dimension. The use of AE does not learn the features of such data well^[22]. Convolutional neural networks are excellent at processing data such as images, so CNNs can be combined with AE by replacing the fully connected (FC) layers of the encoder and decoder parts of AE with convolutional, pooling, and upsampling layers to construct a convolutional autoencoder (CAE) ^[23] to learn signal-related features. The structure of CAE is shown in Fig.3.

The encoder part of CAE consists of multiple convolutional layers and pooling layers. The encoder part learns and reduces the dimension of the characteristics of the input data, and inputs the obtained feature data into the decoder part of the CAE. In the decoder part, upsampling layers are usually inserted to recover the dimension of the feature data, and after iteration of multiple upsampling layers and convolutional layers, the final realization is that the input data and output data have the same dimension.

In the CAE, assuming that there is input data $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L]$, it undergoes convolution and max pooling operations through the encoder to obtain the compressed data $\mathbf{c}$. This process can be expressed as

$$\mathbf{c} = \text{Max pooling}(\mathbf{s}_3(\mathbf{W}_3 * \mathbf{x} + \mathbf{b}_3)), \quad (18)$$

where $*$ denotes the convolution process; Max pooling is the maximum pooling; $\mathbf{s}_3(\bullet)$ is the activation function; the

convolution kernel weight matrix of the convolutional layer in the encoder is denoted by W_3 ; b_3 represents the

bias vector of the same convolutional layer in the encoder.

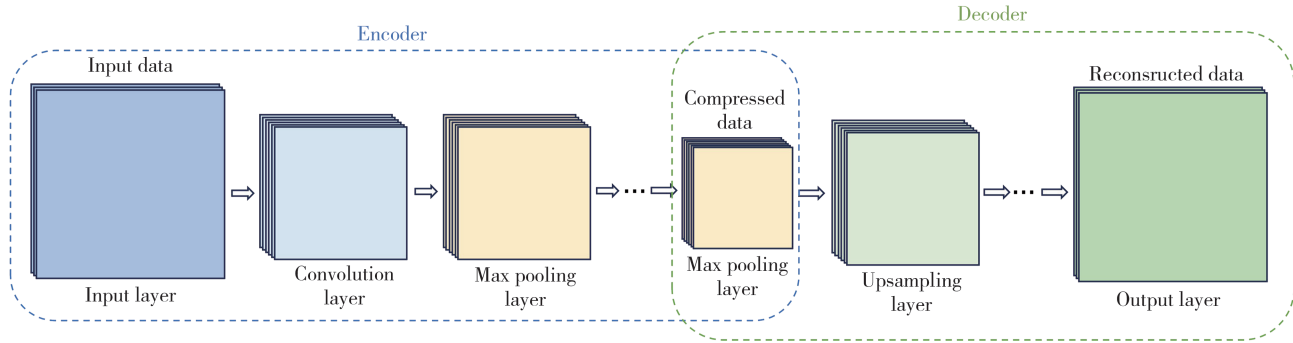


Fig. 3 CAE structure diagram

After multiple convolutional and pooling layers, the encoder part of the CAE compresses the input data to a specified dimension, and then decodes the compressed data through multiple upsampling layers to obtain the reconstructed data $r = [r_1, r_2, \dots, r_L]$. The process can be represented as

$$r = \text{upsampling}(c), \quad (19)$$

where upsampling denotes the upsampling process; the loss function is minimized by optimizing the weights W_3 and bias b_3 . The mean squared error (MSE) is generally adopted as the loss function in CAE, shown as

$$\{W_3, b_3\} = \text{argmin } L_{\text{CAE}}, \quad (20)$$

$$L_{\text{CAE}} = \frac{1}{L} \sum_{l=1}^L \|x_l - r_l\|_2^2. \quad (21)$$

3 DOA estimation method for vector hydrophone array based on CSAE-CNN

3.1 Convolutional sparse autoencoder

Considering the sparsity of the signal sources impinging on the vector hydrophone array, and the overfitting phenomenon that occurs during feature extraction, the L1-norm regularization term^[24] is added

to the convolutional layer of the CAE. Regularization constrains the model's parameters by introducing an additional penalty term into the model's loss function, thereby reducing the complexity of the model. This attains the sparsity constraint, and in the process, it strengthens the model's robustness and increases its generalization ability. The expression for the penalty term is

$$L_{L1} = \lambda \sum_{i=1}^I \|W_i\|_1, \quad (22)$$

where $\|\cdot\|_1$ is the L1-norm; $\lambda \in [0, 1]$ is the regularization parameter, which is used for controlling the strength of the regularization; and W_i denotes the weight matrix of the i -th convolutional kernel in the convolutional layer.

The penalty term is added to the loss function of the CAE model to form the loss function $\text{LOSS}_{\text{CSAE}}$ of the CSAE.

$$L_{\text{CSAE}} = L_{\text{CAE}} + L_{L1} = \frac{1}{L} \sum_{l=1}^L \|x_l - r_l\|_2^2 + \lambda \sum_{i=1}^I \|W_i\|_1. \quad (23)$$

The goal of CSAE is to optimize the parameters in the model so that L_{CAE} and L_{L1} become minimum and the CSAE model can learn more features.

The CSAE network constructed is shown in Fig.4.

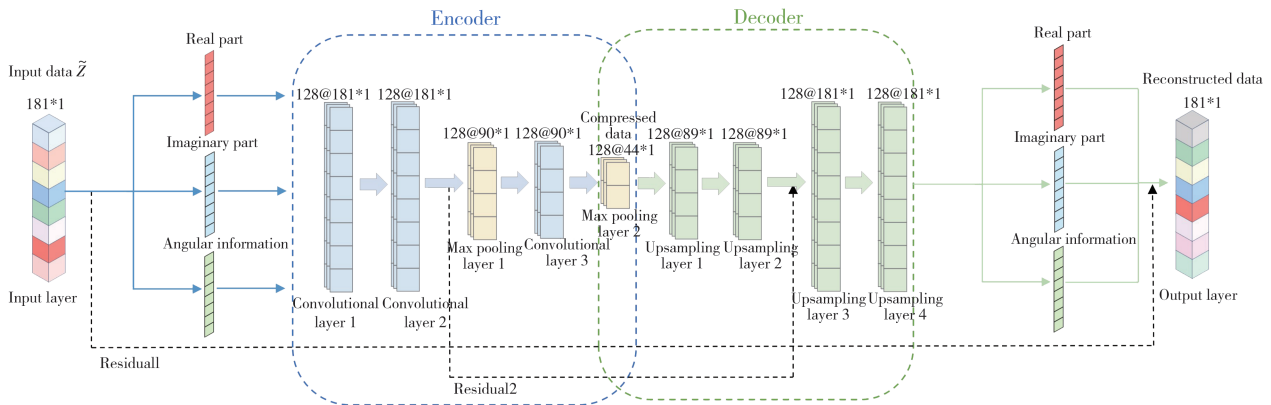


Fig. 4 CSAE structure diagram

The encoder portion is made up of three convolutional layers and two max pooling layers. In the three convolutional layers, the convolution kernel has a size of 5 and a stride of 1. For the two max-pooling layers, the convolution kernel size is 3 with a stride of 2. The decoder portion is composed of five upsampling layers. The convolution kernel of the first upsampling layer has a size of 3, while the convolution kernels of the remaining upsampling layers are of size 5. The size of the convolution kernel of the first and third upsampling layers has a stride of 2, and the rest of the upsampling layers have a stride of 1. All the convolutional and upsampling layers are subjected to the ReLU function and batch normalization after convolution. In addition, two residual join sections are added to the network.

The upsampling layer of the CSAE network can recover details. But as the network layers deepen, the information loss accumulates, causing recovery to become difficult, and the deep network tends to fall into the problem of gradient vanishing. To solve these problems, a residual compensation mechanism is introduced, inspired by deep residual learning^[25]. The core idea of residual connection is to let the network learn the residuals between the inputs and outputs instead of directly learning the mapping relationship from inputs to outputs. This approach is capable of efficiently mitigating the issue of gradient vanishing, retaining feature information, and enhancing the convergence speed of the network. In the residual connection, the input is set to be I , which undergoes a series of nonlinear transformations to obtain $F(I)$, which is then added to the input I to obtain the final output O , expressed by

$$O = F(I) + I, \quad (24)$$

where O is the final output after residual concatenation. The specific structure is shown in Fig.5.

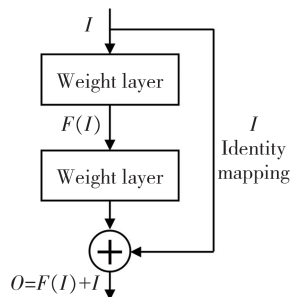


Fig. 5 Structural diagram of residual compensation mechanism

In the CSAE constructed in this paper, the first residual connection is to store the input data as residual1. And after the data has gone through the encoding process and five upsampling layers, the pre-saved residual1 is added to the current feature map. This residual

connection incorporates the original information of the input data directly into the output of the decoder, which enables the reconstructed data to be closer to the original input data. The second residual connection is to save the data after two convolutions as residual2, which is added to the current feature map after the encoding process and three up-sampling operations. The function of this residual connection is to re-inject the feature information extracted by the encoder in the intermediate stage into the decoder, which helps the decoder to better recover the detailed features of the data. These two residual mappings are established to transform the original direct mapping problem into a residual mapping problem, which can effectively alleviate the gradient vanishing problem and retain more features.

3.2 Classification method for DOA estimation based on Class-CSAE-CNN

The CSAE model is first trained to save the information such as weights and bias when the LOSS is smallest. Then the decoder part of the CSAE network is removed and replaced with the flatten layer and four FC layers. The Sigmoid activation function is applied after the FC layers. The Sigmoid function is utilized to obtain the probability of the presence of the signal source at the current angle. The structure of the Class-CSAE-CNN classification network designed is shown in Fig.6. After inputting the data into the network, it first goes through the saved trained CSAE encoder part, then goes through the FC layer to classify the angles. This step is taken because the CSAE network aims to reconstruct data. Its objective is to make the reconstructed data closely approximate the original input data, through reducing the reconstruction error. The encoder continuously adjusts the parameters to extract features that are more representative of the original data, and the process also removes some noise and redundant information.

The specific process of the DOA estimation for the vector hydrophone array using the classification method based on Class-CSAE-CNN proposed in this paper is as follows.

1) Simulate and generate datasets for training CSAE and Class-CSAE-CNN. Binarize and partition the angles in the datasets. Input the CSAE dataset into the CSAE network for feature extraction and data reconstruction, and save the trained model. Then remove the decoder part of the CSAE network, replace it with a flattening layer and a fully connected layer, and add a Sigmoid function after the FC layer to form the Class-CSAE-CNN network required for classification.

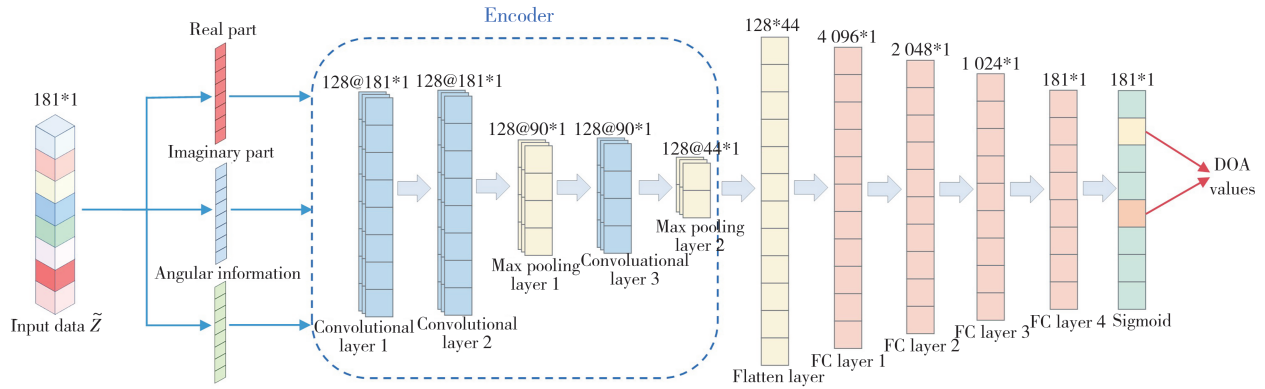


Fig. 6 Structure diagram of Class-CSAE-CNN classification network

2) The dataset used to train Class-CSAE-CNN is directly subjected to feature extraction using the trained CSAE encoder model. It then passes through a flatten layer and a FC layer, and finally the probability of the source's presence at various angles is obtained through the Sigmoid function. The binary cross-entropy (BCE) loss function is used to train the model.

3) After the model training is completed, the test samples are input into the trained Class-CSAE-CNN model, and then the output probabilities of the presence of the signal source at each angle are arranged in descending order. If there are K sources, the labels corresponding to the first K possibilities in the ranking are used as the prediction results for the K sources, thus realizing the DOA estimation of the vector hydrophone array.

In the proposed framework, the CSAE is first employed to perform feature extraction and signal reconstruction. Through minimizing the reconstruction error, the encoder learns representative latent features from the input spectral data while suppressing noise and redundant information. These learned features provide a more robust representation of the spatial characteristics of the received signals. The extracted features are then fed into the CNN-based module for DOA estimation. This two-stage architecture combines the denoising capability of the autoencoder with the strong nonlinear mapping ability of CNNs, enabling the network to learn more discriminative features for accurate angle estimation.

3.3 Regression method for DOA estimation based on Reg-CSAE-CNN

The DOA estimation classification approach based on Class-CSAE-CNN requires discretizing the continuous angular space into a finite number of grid points. This grid limitation may cause the estimation results to be only the values at the pre-defined grid points, resulting in quantization errors. To handle this matter, a DOA estimation regression approach based on Reg-CSAE-

CNN is put forward. This method directly outputs the estimated DOA, bypassing the need for grid-based discretization. Compared with classification-based DOA estimation, regression-based estimation is more suitable for scenarios requiring higher angular resolution. In practical applications, the direction of arrival is a continuous variable, and discretizing the angular space into grids may introduce quantization errors. By adopting the Reg-CSAE-CNN structure, the network directly learns the nonlinear mapping between the extracted features and the continuous DOA value, which improves estimation accuracy and avoids the limitations caused by predefined angular grids.

The model for the feature extraction part of this method is the same as the classification method. The difference is that the classification method outputs a Sigmoid function, while the regression method outputs a tanh activation function with the expression

$$\tanh x = \frac{\exp x - \exp(-x)}{\exp x + \exp(-x)}. \quad (25)$$

The tanh function restricts the output to $[-1, 1]$. Assuming that the range of incident angles is $\theta = [\theta_{\min}, \theta_{\max}]$, the angle corresponding to the labeling equation is

$$l = 2 \times \frac{\alpha - \theta_{\min}}{\theta_{\max} - \theta_{\min}} - 1, \quad (26)$$

where α is the current angle, and the Eq. (26) causes the labels to be scaled down during training to $[-1, 1]$. The main reason for designing the network in this way is to match the model output with the range of the labels, which in turn improves the convergence speed and stability of the model. Fig.7 shows the network for DOA estimation that employs the Reg-CSAE-CNN regression method. The regression method uses the MSE as the loss function to measure the difference between the predicted value and the actual value. After completing the training of the model, the model is

utilized for prediction, and the output obtained is in the interval $[-1, 1]$, which still needs to be converted into

the estimated angle using the inverse transformation of the above equation.

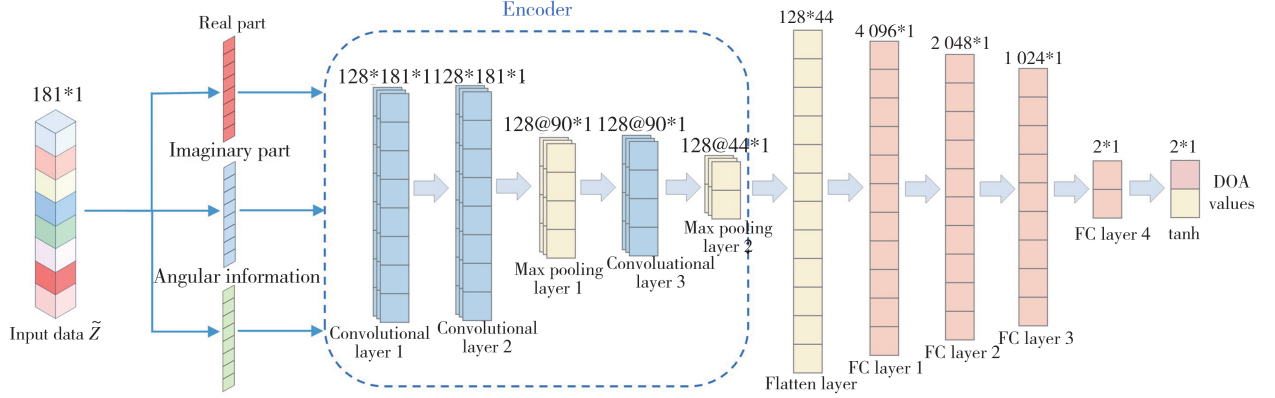


Fig. 7 Structure diagram of Reg-CSAE-CNN regression network

4 Simulation experiment

We compared the algorithm proposed in this paper with the CM-CNN algorithm^[12], CNN algorithm^[20], MUSIC algorithm^[5], CBF algorithm^[4], and OMP algorithm^[26] through simulation experiments. Where the CM-CNN algorithm denotes the use of a covariance matrix as an input to the network. For MUSIC, CBF, and OMP algorithms, the search step size is set to 0.1° . The experimental platform of the thesis is a 64-bit Windows 10 system running on a 3.3 GHz Intel(R) Core (TM) i7-5820K processor and NVIDIA GeForce GTX 1080 Ti, and the experimental software is Matlab 2022b and PyTorch.

4.1 Simulation settings

To train the proposed network, the resolution is defined as $\rho = 1^\circ$, the spatial domain range $G = [-90^\circ, 90^\circ]$, where $|G| = 181$ grid points. Thus, after the binary transformation of DOA, the output dimension of the CNN is 181. The number of vector hydrophone array elements is $M = 10$, the interval of array elements is 0.5 m, and the number of sources is $K = 2$. There are $C_{181}^2 = 16\,290$ chances for all possible angle combinations. Thus, there are a total of $7 \times 16\,290 = 114\,030$ training samples at $S = \{-20, -15, -10, -5, 0, 5, 10\}$ dB. For each signal-to-noise ratio (SNR) place, the network takes the spectral proxy $\tilde{z} = W^H \hat{z}$ as its input.

Since the neural network cannot handle complex numbers directly, it is necessary to obtain information from the input data $\tilde{z}$, where $\tilde{Z}_{:,1} = \text{Re}(\tilde{z})$, $\tilde{Z}_{:,2} = \text{Im}(\tilde{z})$, $\tilde{Z}_{:,3} = \angle(\tilde{z})$. $\tilde{Z}_{:,3}$ represent the angular information of the spectrum proxy $\tilde{z}$. The inputs to the network are formed from D data, and the data set is noted as $\tilde{Z} =$

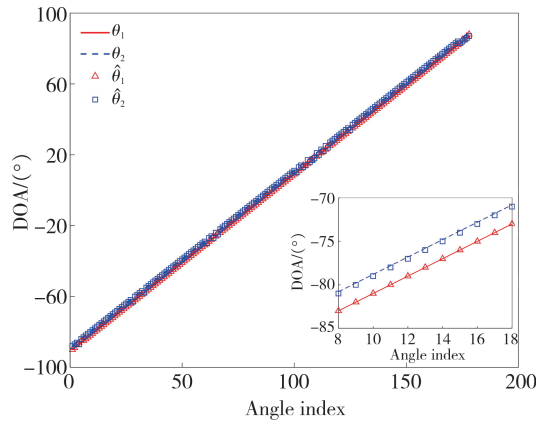
$[\tilde{Z}_1, \tilde{Z}_2, \dots, \tilde{Z}_D]$. The angles of each $\tilde{Z}_i$ in the classification network need to be binarized. The angle corresponding to the location label is 1, and the rest are 0. The binarized I_i is used as the network label. Therefore, the input of the network is $D = \{(\tilde{Z}_1, I_1), (\tilde{Z}_2, I_2), \dots, (\tilde{Z}_D, I_D)\}$. The dataset D undergoes division, the training set constitutes 80% of the data, whereas the validation set makes up 20%.

The network hyperparameters are set as follows. Train the CSAE model with a batch size of 64, a learning rate of 0.000 1, a regularization coefficient of 0.000 1, and 100 epochs of training, using the Adam optimizer and the MSE loss function. Train the Class-CSAE-CNN model with a batch size of 64, an initial learning rate of 0.001, and 100 epochs of training, with the learning rate changing to the previous rate after every 10 epochs 0.5 times of the previous one, using the Adam optimizer and the BCE loss function.

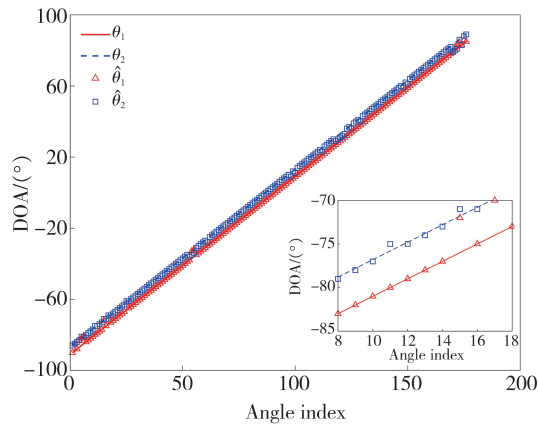
4.2 DOA estimation results

In the experiment, two signal sources were incident into the ULA of vector hydrophones with an SNR of -10 dB. The direction of the first signal source θ_1 is uniformly distributed from -90° to 85° , with a sampling interval of 1° and an angular interval of $\Delta\theta = 4.2^\circ$ between the two sources. The direction of the second signal source $\theta_2 = \theta_1 + \Delta\theta$ is uniformly distributed from -85.8° to 89.2° (not included in the learning phase) and with 2 000 snapshots set. The estimated directions of the method proposed in this paper are shown in Fig. 8(a). The estimated directions of the CNN are shown in Fig. 8(b). The estimated directions of the CM-CNN are shown in Fig. 8(c). For the proposed Class-CSAE-CNN algorithm, the root mean square error (RMSE) in DOA estimation is 0.68° , the RMSE of the CNN is 0.95° , and the RMSE of the CM-CNN is 2.41° . The

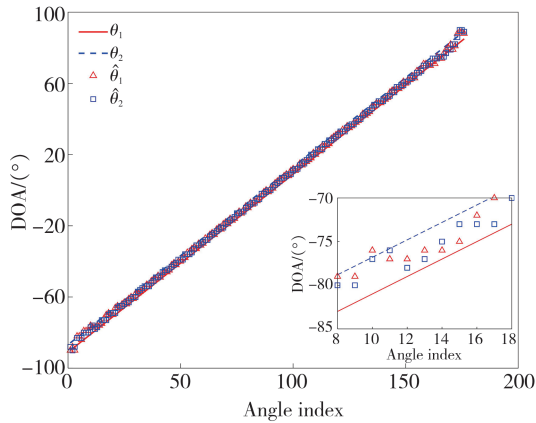
DOA estimation ability of the Class-CSAE-CNN algorithm surpasses that of other algorithms.



(a) Class-CSAE-CNN



(b) CNN



(c) CM-CNN

Fig. 8 DOA estimation results with an SNR of -10 dB, a snapshot number of 2 000, and an angular interval of 4.2°

4.3 Spatial spectrum estimation for coherent sources

Since some traditional algorithms are vulnerable to coherent signal sources in practical environments, to further validate the algorithm's experimental performance proposed in this paper, the number of snapshots in the

experiment is set to 5. The SNR is -5 dB. The incident signal source angles are -30° , 12° , and 34° , and the frequency is 1 000 Hz. Fig.9 illustrates the experimental results.

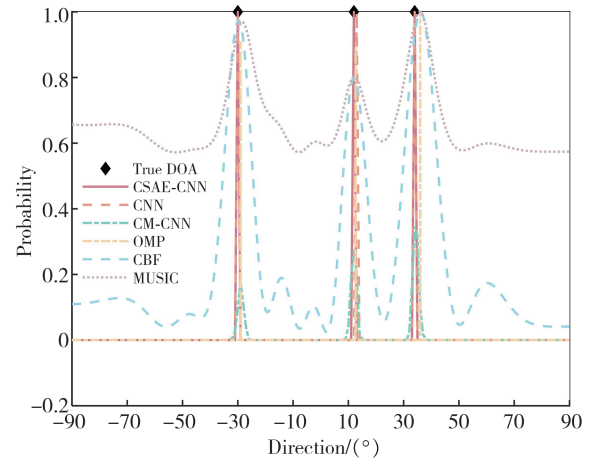


Fig. 9 DOA estimation of various algorithms under coherent source conditions

Table 1 displays the DOA estimation values of various algorithms under the condition of coherent signal sources. It can be seen that the estimation result of the Class-CSAE-CNN algorithm is the most accurate. The incident angles of the two signals can be precisely estimated by the CNN and CM-CNN algorithms. The MUSIC algorithm accurately estimates the result at point -30° , while the CBF and OMP algorithms accurately estimate the result at point 12° . Errors occur in the environment of other coherent signal sources. In conclusion, the Class-CSAE-CNN algorithm performs better in DOA estimation under coherent signal sources, low SNR, and a few snapshots.

Table 1 DOA estimation angles for each algorithm under coherent source conditions

Algorithm	DOA ₁ /($^\circ$)	DOA ₂ /($^\circ$)	DOA ₃ /($^\circ$)
TRUE	-30	12	34
Class-CSAE-CNN	-30	12	34
CNN ^[20]	-30	13	34
CM-CNN ^[12]	-29	12	34
MUSIC ^[5]	-30	12.4	34.9
CBF ^[4]	-30.2	12	34.9
OMP ^[26]	-30.3	12	34.7

4.4 Total number of parameters for deep learning algorithms

Parameter quantity is one of the core indicators for measuring the computational complexity, memory usage, and deployment cost of deep learning models. This section verifies the advantage of the proposed CSAE-CNN in computational complexity by comparing the total parameter count of different deep learning DOA

estimation algorithms. The total parameter count of each algorithm is shown in Table 2.

By comparison, it can be seen that the total parameter count is much smaller when the input is spectral agents (CSAE-CNN, CNN) than when the input is covariance matrix.

Table 2 Total number of parameters for deep learning algorithms

Algorithm	Total number of parameters/M
Class-CSAE-CNN	33.75
CNN ^[19]	33.91
CM-CNN ^[12]	338.56

4.5 Statistical performance

To explore the performance of the approach put forward in this paper, DOA estimation is performed at different SNRs, numbers of snapshots, and the performance is evaluated by analyzing the obtained RMSE. The RMSE used in the experiment is defined by

$$\text{RMSE} = \sqrt{\frac{1}{HK} \sum_{k=1}^K \sum_{v=1}^V (\theta_k^{(v)} - \hat{\theta}_k^{(v)})^2}, \quad (27)$$

where K indicates the number of signal sources; H stands for the number of Monte Carlo experiments; $[\theta_1^{(v)}, \theta_2^{(v)}, \dots, \theta_K^{(v)}]^T$ is the true value of DOA, and $[\hat{\theta}_1^{(v)}, \hat{\theta}_2^{(v)}, \dots, \hat{\theta}_K^{(v)}]^T$ is the estimated value of DOA.

In this experiment, the number of snapshots is configured as 50, and the DOAs of the two signal sources in the directions of $\theta_1=25.2^\circ$ and $\theta_2=28.7^\circ$ are estimated, and the SNR is set to $\{-20, -15, -10, -5, 0, 5, 10, 15, 20\}$ dB. A total of 1 000 Monte Carlo experiments are conducted at each SNR. The algorithms are compared under the same conditions, and the changes of the RMSE of each algorithm with the SNR are shown in Fig.10.

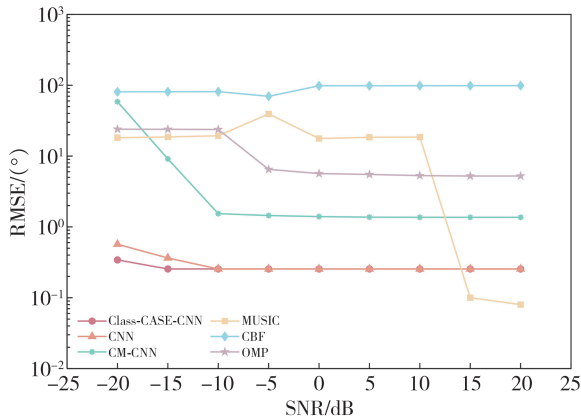


Fig. 10 RMSE under different SNR when number of snapshots is 50

Under different SNR, the RMSE of the Class-CSAE-CNN is basically stable and at a relatively low level,

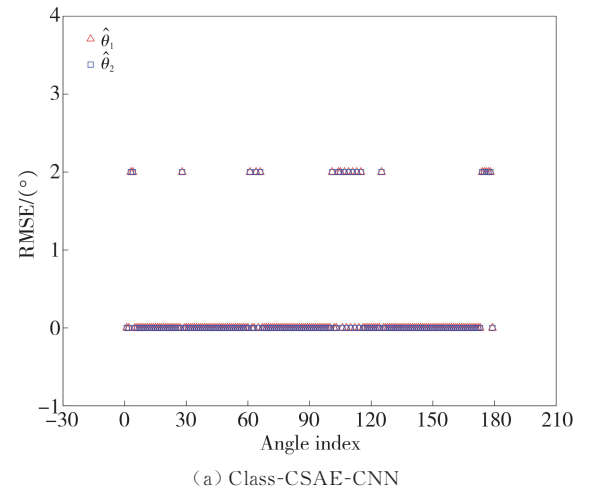
indicating that this algorithm has remarkable robustness and can keep high-precision estimation. However, the RMSE of the CNN and CM-CNN shows a decreasing trend as the SNR rises. When the SNR reaches -10 dB, the RMSE tends to be stable. After stabilization, the RMSE of the CNN is not much different from that of the Class-CSAE-CNN, while the RMSE of the CM-CNN is higher than that of the Class-CSAE-CNN. The RMSE of the MUSIC algorithm is relatively large and stable when the SNR is between -20 dB and 10 dB. Once the SNR exceeds 10 dB, the RMSE drops rapidly. After the SNR reaches 15 dB, the MUSIC algorithm outperforms the Class-CSAE-CNN algorithm. The performance of the CBF is poor, and the RMSE always remains around 100° . The RMSE of the OMP decreases when the SNR is -10 dB, and it outperforms the CBF algorithm in overall performance.

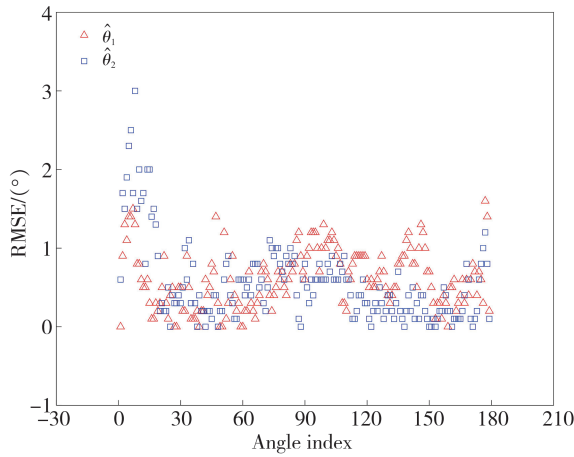
4.6 Performance of CSAE-CNN based regression approach

This section focuses on discussing the performance of the regression approach. The capability of the classification and regression approach in the cases of on-grid and off-grid are compared.

First, consider the on-grid situation. The experiment is carried out under the conditions that the first signal source's incident angle covers the interval from -90° to 88° , and the interval between the two signal sources is 2° . The second signal source's incident angle covers the interval from -88° to 90° , and the sampling intervals of both signal sources are 1° , with an SNR of 10 dB and the number of snapshots amounts to 200 .

Fig.11 shows the experimental results. For the classification approaches, the RMSE corresponding to the two signal sources is almost all 0° , and the RMSE only increases significantly to 2° at some angles.





(b) Reg-CSAE-CNN

Fig. 11 Performance comparison of classification and regression DOA estimation approaches for two signal sources in on-grid situation

For the regression approach, the RMSE of the first signal source is between $[0^\circ, 1.5^\circ]$, and the RMSE of the second signal source is between $[0^\circ, 3^\circ]$. The RMSE values corresponding to the regression approach are more scattered, and the overall estimation accuracy is not as stable as that of the classification approach. The outcomes indicate that in the on-grid situation, the classification approach based on Class-CSAE-CNN is significantly superior to the regression approach in DOA estimation performance.

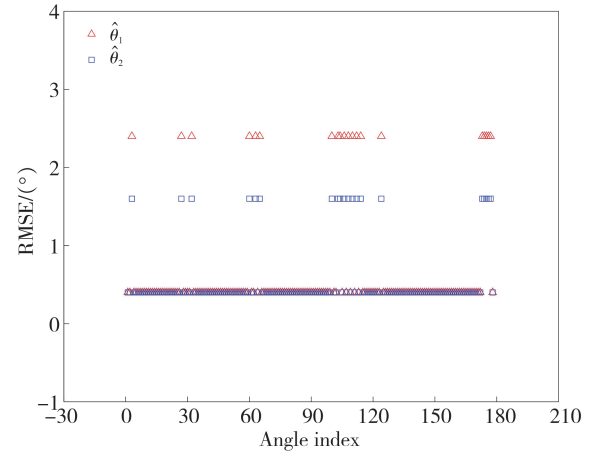
Next, consider the off-grid situation. The experiment is conducted under the conditions that the incident angle of the first signal source ranges from -89.4° to 87.6° , and the interval between the two signal sources is 2° . The incident angle of the second signal source ranges from -87.4° to 89.6° , and the sampling intervals of both signal sources are 1° , with an SNR of 10 dB, and the number of snapshots amounts to 200.

Fig. 12 presents the experimental results. The RMSE of the two signal sources for the classification approach is between $[0.4^\circ, 2.4^\circ]$.

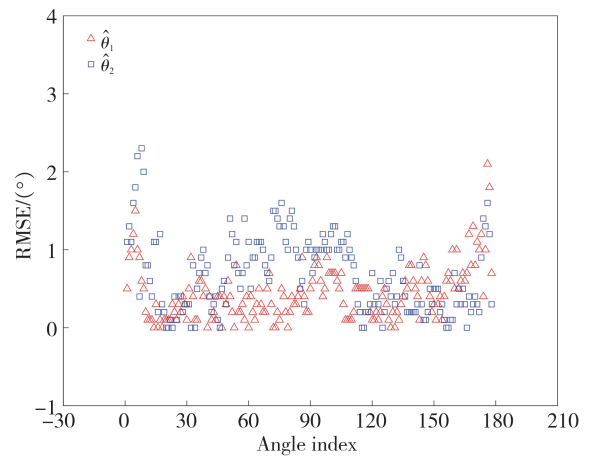
This indicates that in the off-grid situation, the precision of the DOA estimation of the classification approach is considerably affected, and the error increases. For the regression approach, the RMSE of the two signal sources is between $[0^\circ, 2.3^\circ]$. Although there are fluctuations, at most angles, the error does not show high values compared with the classification approach and is relatively scattered. This shows that in the off-grid scenario, although the regression approach of Reg-CSAE-CNN also has estimation errors, its stability and adaptability are relatively better than those of the classification approach.

The experimental results show that under on-grid

condition, the Class-CSAE-CNN classification approach outperforms the regression method in DOA estimation, while under off-grid situation, the Reg-CSAE-CNN regression method outperforms the classification method in DOA estimation.



(a) Class-CSAE-CNN



(b) Reg-CSAE-CNN

Fig. 12 Performance comparison of classification and regression DOA estimation approaches for two signal sources in the off-grid situation

5 Conclusions

This paper proposes a DOA estimation method for vector hydrophone array based on sparse prior CSAE, which includes two methods: Class-CSAE-CNN and Reg-CSAE-CNN. The core results of the simulation experiment show that the proposed Class-CSAE-CNN classification method outperforms CM-CNN, CNN, and traditional algorithms in estimation accuracy, especially under complex conditions such as low signal-to-noise ratio and few snapshots. The Reg-CSAE-CNN regression method effectively breaks through grid limitations and achieves performance improvement compared to classification methods in off-grid scenarios. Future research will focus on exploring the fusion path of

deep learning and traditional algorithms to estimate DOA. Although the proposed CSAE-CNN method demonstrates superior performance under low SNR and small snapshot conditions, several limitations should be acknowledged. The proposed method relies on simulated datasets for training, which may lead to a performance gap when applied to real-world underwater acoustic environments. The classification-based approach still depends on predefined angular grids, which may cause quantization errors in highly off-grid scenarios, although the regression model partially alleviates this issue. Future work will focus on lightweight model design, adaptive grid-free estimation, and robustness enhancement under array model mismatch.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (No. 61774137) and the Scientific Activities of Selected Returned Overseas Professionals in Shanxi Province (No. 20240011).

Declaration of conflicting interests

The authors have no conflict of interests related to this publication.

References

- [1] WEN F Q, SHI J P, GUI G, et al. Joint DOD and DOA estimation for NLOS target using IRS-aided bistatic MIMO radar. *IEEE Transactions on Vehicular Technology*, 2024, 73(10): 15798-15802.
- [2] TIAN Y, FENG Y, LIU W, et al. Reconfigurable intelligent surface aided DOA estimation by a single receiving antenna. *IEEE Transactions on Communications*, 2025, 73(2): 983-994.
- [3] NIU R Z, ZHANG Z Y. Harmonic frequency enhanced direction of arrival estimation using non-uniform linear arrays. *IEEE Access*, 2024, 12: 36942-36950.
- [4] BARTLETT M S. Properties of sufficiency and statistical tests. *Proceedings of the Royal Society of London Series A, Mathematical and Physical Sciences*, 1937, 160(901): 268-282.
- [5] SCHMIDT R. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, 1986, 34(3): 276-280.
- [6] ROY R, PAULRAJ A, KAILATH T. ESPRIT: a subspace rotation approach to estimation of parameters of cisoids in noise. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1986, 34(5): 1340-1342.
- [7] MERKOFER J P, REVACH G, SHLEZINGER N, et al. DA-MUSIC: data-driven DoA estimation via deep augmented MUSIC algorithm. *IEEE Transactions on Vehicular Technology*, 2024, 73(2): 2771-2785.
- [8] LI J Z, YANG G L, REN X Q, et al. Modified ESPRIT and unitary-transformation form for computationally efficient DOA estimation on FPGA. *IEEE Transactions on Instrumentation and Measurement*, 2025, 74: 2003209.
- [9] ZISKIND I, WAX M. Maximum likelihood localization of multiple sources by alternating projection. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1988, 36(10): 1553-1560.
- [10] MEI Z Q, LI X, LI H F, et al. Optimization algorithm for DOA estimation accuracy of coherent signal sources based on sparse Bayesian theory. *Signal, Image and Video Processing*, 2025, 19(3): 249.
- [11] LIU Y J, CHEN H X, WANG B. DOA estimation based on CNN for underwater acoustic array. *Applied Acoustics*, 2021, 172: 107594.
- [12] PAPAGEORGIOU G K, SELLATHURAI M, ELDAR Y C. Deep networks for direction-of-arrival estimation in low SNR. *IEEE Transactions on Signal Processing*, 2021, 69: 3714-3729.
- [13] REN J, TAN X H, BAI Y P, et al. Vector hydrophone DOA estimation method based on Complex-Valued convolutional denoising autoencoder denoising. *Journal of Test and Measurement Technology*, 2025, 39(4): 475-482.
- [14] QIN Y H. Residual neural network for direction-of-arrival estimation of multiple targets in low SNR. *IET Signal Processing*, 2024, 2024(1): 4599954.
- [15] ZHENG S L, YANG Z, SHEN W G, et al. Deep learning-based DOA estimation. *IEEE Transactions on Cognitive Communications and Networking*, 2024, 10(3): 819-835.
- [16] DAI J S, XU X, ZHAO D A. Direction-of-arrival estimation via real-valued sparse representation. *IEEE Antennas and Wireless Propagation Letters*, 2013, 12: 376-379.
- [17] LIU Z M, HUANG Z T, ZHOU Y Y. Sparsity-inducing direction finding for narrowband and wideband signals based on array covariance vectors. *IEEE Transactions on Wireless Communications*, 2013, 12(8): 1-12.
- [18] BECK A, TEBOULLE M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2009, 2(1): 183-202.
- [19] WU L L, LIU Z M, HUANG Z T. Deep convolution network for direction of arrival estimation with sparse prior. *IEEE Signal Processing Letters*, 2019, 26(11): 1688-1692.
- [20] QIN Y H. Deep networks for direction of arrival estimation with sparse prior in low SNR. *IEEE Access*, 2023, 11: 44637-44648.
- [21] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors. *Nature*, 1986, 323(6088): 533-536.
- [22] LI J M, ZHANG J, BAH M J, et al. An auto-encoder with genetic algorithm for high dimensional data: towards accurate and interpretable outlier detection. *Algorithms*, 2022, 15(11): 429.

- [23] MASCI J, MEIER U, CIREŞAN D, et al. Stacked convolutional auto-encoders for hierarchical feature extraction//Artificial Neural Networks and Machine Learning-ICANN 2011. Berlin, Heidelberg: Springer, 2011: 52-59.
- [24] EFRON B, HASTIE T, JOHNSTONE I, et al. Least angle regression. The Annals of Statistics, 2004, 32(2): 407-451.
- [25] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition//2016 IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE, 2016: 770-778.
- [26] AICH A, PALANISAMY P. On-grid DOA estimation method using orthogonal matching pursuit//2017 International Conference on Signal Processing and Communication, July 28-29, 2017, Coimbatore, India. New York: IEEE, 2018: 483-487.

稀疏先验下基于卷积稀疏自编码器的阵列信号 DOA 估计

任 晶, 谭秀辉, 白艳萍*, 王 鹏, 程 蓉, 张 峰, 续 婷

中北大学 数学学院, 山西 太原 030051

摘 要: 深度学习在 DOA 估计中的应用在阵列信号处理领域具有重要意义。使用深度学习对矢量水听器阵列进行 DOA 估计通常直接将信号的协方差矩阵作为信号特征输入到网络中, 但这种方法存在数据要求高、计算复杂度高等局限性。本文提出了一种在稀疏先验条件下基于卷积稀疏自编码器的矢量水听器阵列 DOA 估计方法。该方法在卷积自编码器的卷积层中添加 L1 范数正则化项, 以实现稀疏约束, 并建立卷积稀疏自编码器。同时, 引入了残差补偿机制, 以避免训练过程中的过拟合和细节丢失。接下来, 矢量水听器阵列的信号协方差矩阵的列被视为空间谱的欠采样噪声线性测量值, 并输入到卷积稀疏自编码器中进行特征提取和重建。最后, 将获得的特征用作训练卷积神经网络的输入, 以实现多源 DOA 估计。此外, 为了解决离网情况下分类方法的缺点, 我们提出了一种基于卷积稀疏自编码器的 DOA 回归估计方法。仿真结果表明, 在低信噪比和少量快照等复杂条件下, 本文提出的分类方法在估计性能方面优于各种深度学习算法和文献中提到的传统算法。所提出的回归方法可以进一步提高离网场景下的 DOA 估计性能。

关键词: 矢量水听器阵列; 到达方向估计; 稀疏表示; 卷积稀疏自动编码器; L1 范数正则化; 回归

引用格式: REN Jing, TAN Xiuhui, BAI Yanping, et al. DOA estimation of array signals based on convolutional sparse autoencoder under sparse prior. Journal of Measurement Science and Instrumentation, 2026, 17(2): 254-266. DOI: 10.62756/jmsi.1674-8042.2026022