



Article

Two-Dimensional Coherent Polarization–Direction-of-Arrival Estimation Based on Sequence-Embedding Fusion Transformer

Zihan Wu ^{1,2,3} , Jun Wang ^{1,2,3,*} and Zhiquan Zhou ^{1,2,3}¹ School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150000, China; 23b905043@stu.hit.edu.cn (Z.W.); zzq@hitwh.edu.cn (Z.Z.)² School of Information Science and Engineering, Harbin Institute of Technology, Weihai 264209, China³ Key Laboratory of Cross-Domain Synergy and Comprehensive Support for Unmanned Marine Systems, Ministry of Industry and Information Technology, Weihai 264209, China

* Correspondence: jwang@hit.edu.cn

Abstract: Addressing the issue of inadequate convergence and suboptimal accuracy in classical data-driven algorithms for coherent polarization–direction-of-arrival (DOA) estimation, a novel high-precision two-dimensional coherent polarization–DOA estimation method utilizing a sequence-embedding fusion (SEF) transformer is proposed for the first time. Drawing inspiration from natural language processing (NLP), this approach employs transformer-based multitasking text inference to facilitate joint estimation of polarization and DOA. This method leverages the multi-head self-attention mechanism of the transformer to effectively capture the multi-dimensional features within the spatial-polarization domain of the covariance matrix data. Additionally, an SEF module was proposed to fuse the spatial-polarization domain features from different dimensions. The module is a combination of a convolutional neural network (CNN) with local information extraction capabilities and a feature dimension transformation function, serving to improve the model’s ability to fuse information about features in the spatial-polarization domain. Moreover, to enhance the model’s expressive capacity, we designed a multi-task parallel output mode and a multi-task weighted loss function. Simulation results demonstrate that our method outperforms classical data-driven approaches in both accuracy and generalization, and the estimation accuracy of our method is improved relative to the traditional model-driven algorithm.



Citation: Wu, Z.; Wang, J.; Zhou, Z. Two-Dimensional Coherent Polarization–Direction-of-Arrival Estimation Based on Sequence-Embedding Fusion Transformer.

Remote Sens. **2024**, *16*, 3977. <https://doi.org/10.3390/rs16213977>

Academic Editor: Silvia Liberata Ullo

Received: 3 September 2024

Revised: 16 October 2024

Accepted: 17 October 2024

Published: 25 October 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: coherent polarization–DOA estimation; sequence-embedding fusion transformer; mechanism of multiple head self attention; convolutional neural network

1. Introduction

Direction of Arrival (DOA) estimation is a crucial task in array signal processing research. Many excellent estimation methods, such as the multiple signal classification (MUSIC) algorithm [1] and the estimating signal parameters via rotational invariance techniques (ESPRIT) algorithm [2], have been proposed and widely applied in modern radar, sonar, and wireless communication systems. While traditional scalar arrays have achieved a relatively mature state in DOA super-resolution angle estimation, polarization-sensitive arrays (PSAs) offer notable advantages in practical applications. Due to their strong resistance to interference and high separation probability, PSAs have garnered considerable interest in signal processing research over the past few decades [3–5]. At the same, various algorithms have been explored in depth for coherent polarization–DOA estimation, including polarization smoothing (PS) [6], improved polarization smoothing [7], signal-subspace fitting [8,9], and alternating projection (AP) [10]. In general, the aforementioned algorithms construct parametric models for signals or antenna arrays, and thus can be categorized as “model-driven”. Yet, the performance of model-based methods deteriorates rapidly when the actual received signals significantly deviate from the predetermined model, particularly in the presence of factors such as a low signal-to-noise ratio (SNR). Moreover, existing

algorithms often fail to account for the real-time requirements, which are critical in modern radar, wireless communication, and other applications where polarization–DOA estimation must frequently operate in high-speed, real-time environments.

Therefore, to address such issues, various data-driven learning methods have emerged, namely neural networks and support vector machines (SVM) [11–13]. Researchers have applied deep learning techniques to radar parameter estimation, resulting in significant improvements in estimation accuracy [14–16]. Meanwhile, with the rapid development of deep learning, coherent DOA estimation algorithms based on deep learning have gradually received attention. Various deep learning-based methods have been proposed to address the limitations of model driven approaches. Some methods include defect elimination algorithms [17], unsupervised learning strategies [18], and phase position learning [19–21]. These methods outperform traditional algorithms such as spatial smoothing and subspace fitting, particularly in complex environments. For instance, in one study [22], deeper data features of coherent signals were explored by introducing angle separation learning schemes. A more suitable learning model was established that overcame the limitations of traditional methods and improved the performance of coherent DOA estimation. At the same time, models such as the multitask autoencoder [23] and few-shot learning [24] have demonstrated superior performance in low SNR scenarios, handling fewer snapshots and array defects more effectively than typical super-resolution algorithms. These advancements offer new possibilities and directions for radar parameter estimation. Nonetheless, research on 2-D coherent polarization–DOA estimation using deep learning remains limited. In addition, although simulations show that a multilayer perceptron (MLP) can perform coherent polarization DOA estimation, its fully connected structure may limit its effectiveness. The lack of position and time modeling mechanisms in the multilayer perceptron may affect the effectiveness of extracting and fusing multi-dimensional information in the spatial polarization domain.

To address these issues, a high-precision 2-D coherent polarization–DOA estimation method based on a sequence-embedding fusion (SEF) transformer is proposed for the first time. Initially, inspired by natural language processing (NLP) techniques [25], the multi-dimensional data of the spatial-polarization domain, represented by the covariance matrix, is translated into textual tasks within the transformer model [26,27]. This transformation leverages the multi-head self-attention mechanism of the transformer, allowing it to capture the intricate multi-dimensional features of the spatial-polarization domain. Additionally, this paper introduces the SEF module, which integrates a convolutional neural network (CNN) with local information extraction capabilities and a dimension transformation function. This integration enhances the correlation within the spatial-polarization domain across different spatial and temporal sequences. The model effectively integrates spatial-polarization domain features from the covariance matrix, improving its ability to capture correlations. In addition, in order to improve the expressiveness and generalization ability of the model, we designed a multi-task parallel output model and a multi-task weighted loss function [28,29]. Simulation results show that the proposed method improves the estimation accuracy compared to the traditional model-driven algorithms such as the generalized subspace fitting (GSF) algorithm. Additionally, the method is superior to classical data-driven methods such as multilayer perceptron (MLP) in the estimation accuracy and generalization ability. Moreover, the proposed method has made significant advancements in both real-time performance and efficiency, and future research will further explore reconfigurable digital architectures. This challenge highlights the importance of reconfigurable digital architectures as a key research direction. By leveraging reconfigurable hardware architectures, such as field-programmable gate arrays (FPGA) and application-specific integrated circuits (ASIC), researchers can develop polarization–DOA estimation systems capable of real-time operation. These architectures offer flexible hardware resource allocation, which not only enhances processing speed but also significantly reduces power consumption, thereby meeting the demands of modern signal processing [30,31].

The rest of this paper is structured as follows: Section 2 introduces the signal model, including the detailed mathematical representation of the array model and the received signals. Section 3 presents the core methodology of the proposed sequence-embedding fusion (SEF) transformer for polarization-DOA estimation, including the transformer model, data preprocessing, and the SEF module for feature fusion. Section 3 explains the optimization strategies for the model, such as omitting positional encoding and implementing multi-task parallel processing, as well as introducing the angle pairing method and calculation principle of model efficiency. Section 4 details the numerical simulations and experiments conducted to evaluate the performance of the proposed method. Finally, Section 5 concludes the paper by summarizing the key findings and outlining potential directions for future research.

2. Signal Model

In this paper, as shown in Figure 1, a uniform circular array (UCA) of radius r is used to adopt M co-point biorthogonal short dipoles, and the two channels correspond to the directions of the x and y axes, respectively. Let the azimuth angle and elevation angle of the n -th incident signal be (φ_n, θ_n) , the auxiliary polarization angle and polarization phase difference be (γ_n, η_n) , the spatial steering vector of the corresponding signal can be expressed as:

$$\mathbf{a}_{s,n} = \left[e^{-j\frac{2\pi}{\lambda}(x_1\phi_{x,n}+y_1\phi_{y,n})}, \dots, e^{-j\frac{2\pi}{\lambda}(x_M\phi_{x,n}+y_M\phi_{y,n})} \right]^T \quad (1)$$

where $\mathbf{a}_{s,n} = \mathbf{a}_s(\theta_n, \varphi_n)$, $x_m = r \cos(\frac{2\pi(m-1)}{M})$ and $y_m = r \sin(\frac{2\pi(m-1)}{M})$, $m = 1, 2, \dots, M$ is the position coordinate of the m -th dipole, $\phi_{x,n} = \cos \theta_n \cos \varphi_n$, and $\phi_{y,n} = \sin \theta_n \cos \varphi_n$. The polarization steering vectors for the n -th signal of the array can be expressed as:

$$\mathbf{a}_{p,n} = \begin{bmatrix} -\sin(\varphi_n) & \cos(\theta_n) \cos(\varphi_n) \\ \cos(\varphi_n) & \cos(\theta_n) \sin(\varphi_n) \end{bmatrix} \begin{bmatrix} \cos(\gamma_n) \\ \sin(\gamma_n) e^{j\eta_n} \end{bmatrix} \quad (2)$$

where $\mathbf{a}_{p,n} = \mathbf{a}_p(\theta_n, \varphi_n, \gamma_n, \eta_n)$, and since the array element uses a co-point electromagnetic vector sensor, the manifold matrix consisting of N signals can be expressed as:

$$\mathbf{A}(\theta, \varphi, \gamma, \eta) = [\mathbf{a}_{s,1} \otimes \mathbf{a}_{p,1}, \dots, \mathbf{a}_{s,N} \otimes \mathbf{a}_{p,N}] \quad (3)$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{A}(\theta, \varphi, \gamma, \eta) \in \mathbb{C}^{2M \times N}$ is the manifold matrix. When the incident signal is coherent, the signal can be expressed as:

$$\mathbf{s}_n(t) = \alpha_n \mathbf{s}_1(t), n = 1, 2, \dots, N \quad (4)$$

where the complex gain α_n is the complex constant of $\mathbf{s}_n(t)$ with respect to $\mathbf{s}_1(t) \in \mathbb{C}^{1 \times T}$, and T is the number of snapshots. The received signal can be expressed as:

$$\mathbf{X}(t) = \mathbf{A}(\theta, \varphi, \gamma, \eta) \boldsymbol{\rho} \mathbf{s}_1(t) + \mathbf{N}(t) \quad (5)$$

where $\mathbf{N}(t) \in \mathbb{C}^{2M \times T}$ is the additive white Gaussian noise vector, and $\boldsymbol{\rho} = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_N]^T$ is the $N \times 1$ dimensional vector of attenuation coefficients. The covariance matrix of the received signal $\mathbf{X}(t)$ can be expressed as:

$$\mathbf{R} = E(\mathbf{X}(t) \mathbf{X}^H(t)) = \begin{bmatrix} \mathbf{r}_{ii} & \mathbf{r}_{ij} \\ \mathbf{r}_{ji} & \mathbf{r}_{jj} \end{bmatrix}, i, j = x, y \quad (6)$$

where $\mathbf{R} \in \mathbb{C}^{2M \times 2M}$, and $\mathbf{r}_{ij}$ is the covariance matrix associated with different polarization information.

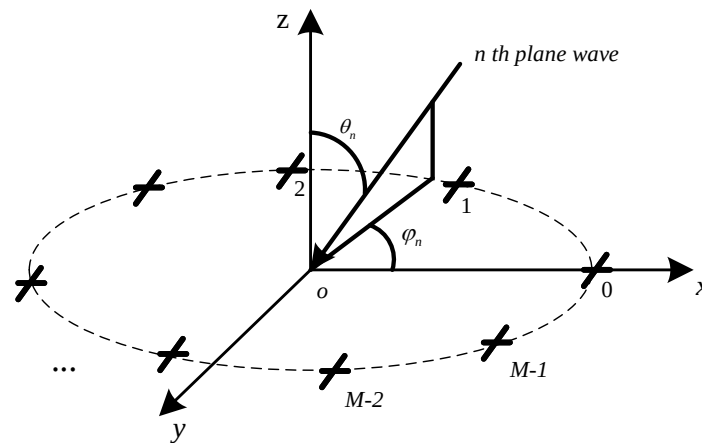


Figure 1. Array-model structure.

3. SEF Transformer Model for Polarization–DOA Estimation

As discussed in this section, using the signal model, the aim was to process the multi-dimensional feature data derived from coherent signals and their spatial-polarization domain information. However, the fully-connected structure of multilayer perceptrons (MLP) may limit their effectiveness in estimating 2-D coherent polarization–DOA. This is due to the absence of model mechanisms such as position and time, which impedes the extraction of spatial-polarization domain feature information. As such, efforts were made to transform the challenge of processing multi-dimensional data $\mathbf{R}$ in the spatial-polarization domain into a textual inference task within the transformer model [26,27], in accordance with the principles of NLP [25]. Since transformers typically process input in the form of tensors containing sequence information, the proposed model, as illustrated in Figure 2, utilizes a preprocessed $\mathbf{R}$. This matrix is divided into S -dimensional sequence data and E -dimensional embedding data, which serve as inputs to the model.

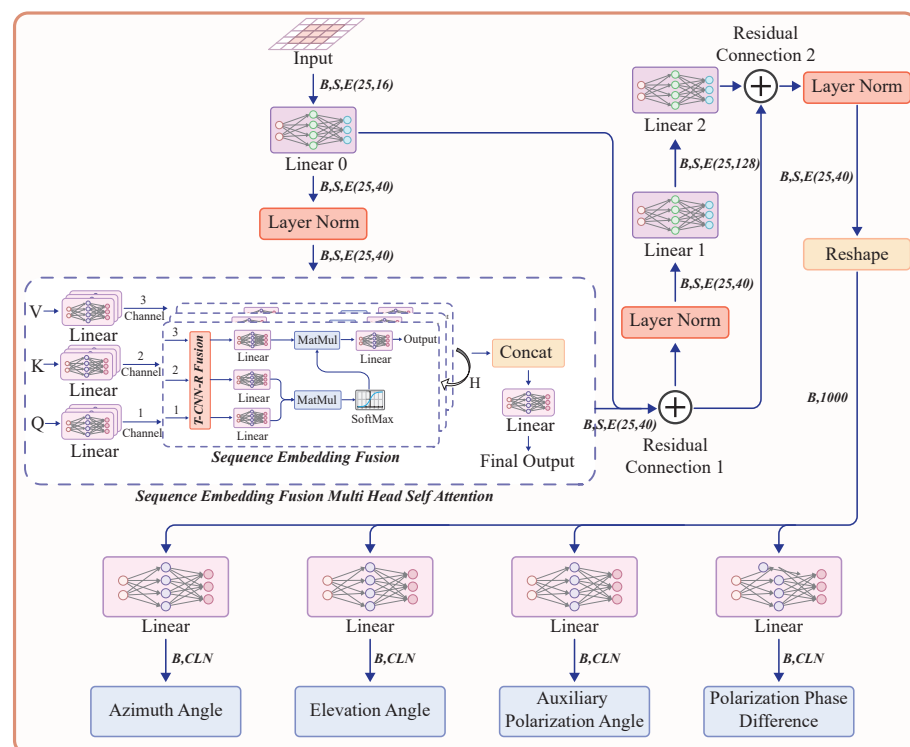


Figure 2. Proposed network model.

The generated signal can be represented as L -dimensional data. When the signal angle falls within a specific interval, the corresponding position is set to 1, while all other positions are set to 0. Assuming there are N incoming waves, the signal is represented by an L -dimensional vector of length with N positions are 1 and the remaining positions are 0. This representation is known as N -hot encoding. To address the classification problem, this paper employs N -hot encoding to explore all possible permutations, which corresponds to the number of permutations CLN for C_L^N . As such, one-hot encoding is used as a substitute for N -hot encoding. The model's output consists of four parallel angle predictions in one-hot format. Here, B denotes the batch size, S denotes the sequence dimension, and E denotes the embedding dimension.

The following section presents an overview of the fundamental concepts of the data preprocessing, sequence-embedding fusion, and optimization of the model within the context of the SEF transformer model architecture.

3.1. Data Preprocessing

3.1.1. Input Preprocessing

Considering that the covariance matrix is a conjugate symmetric matrix, it is generally necessary to utilize only the upper or lower triangular portion of $\mathbf{R}$ as an input:

$$\mathbf{r} = [r_{1,1}, \dots, r_{1,2M}, r_{2,2}, \dots, r_{2,2M}, \dots, r_{2M,2M}]^T \quad (7)$$

where $r_{p,q}$ is the (p, q) th element of $\mathbf{R}$.

Since $\mathbf{R}$ in the aforementioned equation contains complex numbers, except for the diagonal elements, it is necessary from a practical modeling perspective to separate these complex numbers into their real and imaginary components. These components are then normalized and used as input to the model:

$$\hat{\mathbf{r}} = \frac{[\Re(r_{1,1}), \Re(r_{1,2}), \Im(r_{1,2}), \dots, \Re(r_{2M,2M})]^T}{\|\mathbf{r}\|_2} \quad (8)$$

where $\hat{\mathbf{r}} \in \mathbb{C}^{4M^2}$, $\Re(\cdot)$ and $\Im(\cdot)$ denotes the operations of taking real and imaginary components, respectively, and $\|\cdot\|_2$ denotes the l_2 -norm of the vector.

3.1.2. Training and Testing Preprocessing

One of the main advantages of layer normalization [32] is its ability to mitigate the effects of internal covariate shift in neural networks. By using consistent means and variances during both the training and testing phases, layer normalization helps maintain stability in the model's data distribution, thereby enhancing the model's generalization ability. Additionally, this technique stabilizes the training process and accelerates convergence by keeping the inputs to the activation function within a small range. This helps prevent gradient vanishing, which further improves the model's sensitivity to input data. The mean and variance are first calculated during training, where layer normalization is normalized over feature dimensions, and we set feature input as $\mathbf{X} \in \mathbb{R}^{B \times H \times W}$, B is the batch size, $H \times W$ is the feature dimension, and calculated the mean variance of each sample in the B dimension,

$$\mu = \frac{1}{H \times W} \sum_{i=1}^{H \times W} x_i \quad (9)$$

$$\sigma^2 = \frac{1}{H \times W} \sum_{i=1}^{H \times W} (x_i - \mu)^2 \quad (10)$$

where μ is the mean value, and σ^2 is the variance. The standardized x_i value is,

$$\hat{x}_i = \frac{x_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} \quad (11)$$

$$\hat{y}_i = \gamma \hat{x}_i + \beta \quad (12)$$

where ε is a considerably small constant that is used to prevent the denominator from being zero. The scaling amount γ and offset amount β are learnable parameters and rescale and offset to obtain the final output $\hat{y}_i$.

During the testing phase, using the parameters learned during the training phase, the input is directly normalized:

$$\hat{y}_i = \gamma_{train} \odot \frac{x_i - \mu_{train}}{\sqrt{\sigma_{train}^2 + \varepsilon}} + \beta_{train} \quad (13)$$

where $\odot$ denotes the multiplication between elements, μ_{train} and σ_{train} are the mean and standard deviation of the entire training set calculated during the training phase, and γ_{train} and β_{train} are parameters learned through the gradient descent algorithm during the training process.

3.2. SEF Module for Feature Fusion

As discussed in this section, it is necessary to convert $\hat{r}$ into the tensor form of sequence-embedding in textual inference using the following expression:

$$\tilde{\mathbf{R}} = \begin{bmatrix} \hat{r}_{1,1} & \hat{\mathbf{R}}(r_{1,2}) & \hat{\mathbf{S}}(r_{1,2}) & \dots \\ \vdots & \vdots & \vdots & \vdots \\ \dots & \hat{\mathbf{R}}(r_{2M-1,2M}) & \hat{\mathbf{S}}(r_{2M-1,2M}) & \hat{r}_{2M,2M} \end{bmatrix} \begin{matrix} \text{Sequence} \\ \\ \\ \text{Embedding} \end{matrix} \quad (14)$$

where $\tilde{\mathbf{R}}$ multi-dimensional data matrix composed of S -dimensional sequences and E -dimensional embeddings. In order to reduce the influence of internal covariate shift, the following preprocessing, layer normalization, and linear transformation operations are performed,

$$\tilde{\mathbf{R}}^{pre} = \text{Preprocess}(\tilde{\mathbf{R}}) \quad (15)$$

$$\mathbf{X}^{[1]} = \text{Linear0}(\tilde{\mathbf{R}}^{pre}) = \tilde{\mathbf{R}}^{pre} \times \mathbf{W}^{[0]} + \mathbf{b}^{[0]} \quad (16)$$

$$\mathbf{X}_{LN}^{[1]} = \text{LayerNorm}(\mathbf{X}^{[1]}) \quad (17)$$

In the multi-head attention mechanism of the transformer model, let H represent the number of attention heads. For a given multi-head attention layer, the inputs consist of $(\mathbf{Q}, \mathbf{K}, \mathbf{V})_{(i)}$, where $i = 1, \dots, H$, $\mathbf{Q} = \mathbf{X}_{LN}^{[1]} \mathbf{W}_Q$, $\mathbf{K} = \mathbf{X}_{LN}^{[1]} \mathbf{W}_K$, $\mathbf{V} = \mathbf{X}_{LN}^{[1]} \mathbf{W}_V$. $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ is the weight matrix of learning. $\mathbf{Q}_{(i)}$ is the query, which can be represented as a certain representation of the input sequence, $\mathbf{K}_{(i)}$ and $\mathbf{V}_{(i)}$ are keys and values, respectively, used to calculate attention weights and aggregate features. $\mathbf{Q}_{(i)}$ and $\mathbf{K}_{(i)}$ perform matrix multiplication, and the equation is as follows:

$$\mathbf{D}_{(i)} = \mathbf{Q}_{(i)} \times \mathbf{K}_{(i)}^T \quad (18)$$

Then, a scaling operation is performed using the following equation:

$$\mathbf{C}_{(i)} = \frac{\mathbf{D}_{(i)}}{\sqrt{d^k}} \quad (19)$$

where d^k is the dimension of $\mathbf{K}_{(i)}$. Subsequently, a masking operation is required to mask certain regions that are not utilized for attention. These regions are multiplied by 0, then *Softmax* mapping and the matrix weights are projected to $[0, 1]$. The attention weights are then obtained using the following equation:

$$\mathbf{B}_{(i)} = \text{Softmax}(\mathbf{C}_{(i)}) \quad (20)$$

Finally, the attention weight is multiplied by $\mathbf{V}_{(i)}$ to obtain the output:

$$O_{(i)} = B_{(i)} \times V_{(i)} \quad (21)$$

Introducing this multi-head self-attention mechanism into the subsequent calculation, of $\tilde{R}$ leverages the fact that each attention head, denoted as $O_{(i)}$, can learn distinct feature representations of incoming signals. This diversity in feature learning aids the model in gaining a more nuanced understanding of signal distribution in space. In the context of 2-D coherent polarization-DOA estimation, which often involves multiple polarization channels, each channel offers a unique perspective on the signal's polarization characteristics. The multi-head self-attention mechanism enhances the model's ability to associate and integrate features from different channels, thereby improving the capture of the signal's polarization characteristics.

Given the spatial-polarization feature correlations in the spatial-polarization domain $\tilde{R}$, this paper incorporates a CNN with local information extraction capabilities into the SEF module. As illustrated in Figure 3, the CNN processes the queries, keys, and values to produce outputs $CNN(Q)$, $CNN(K)$, and $CNN(V)$, which are then fed into the multi-head attention mechanism. This approach enhances the correlation information between sequence and embedding in the spatial-polarization domain through the attention mechanism. Additionally, the multi-head attention mechanism integrates global information from the outputs $O_{(1)}, O_{(2)}, \dots, O_{(H)}$, which improves the model's ability to capture spatial-polarization domain relationships. This process contributes to better performance in coherent polarization-DOA estimation within the spatial-polarization domain. The specific operational process of the multi-head self-attention fusion CNN is detailed in Algorithm 1.

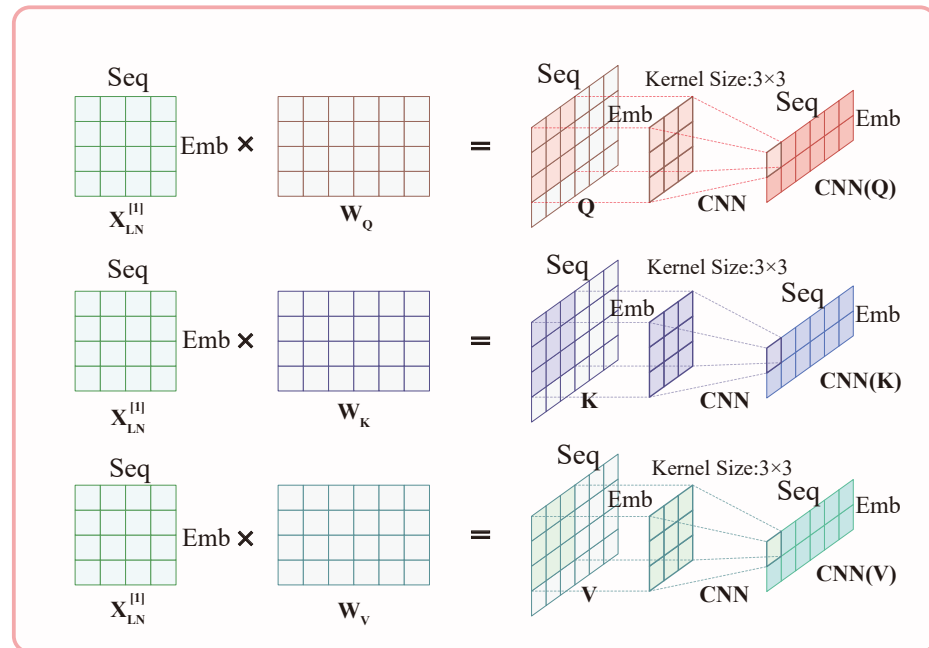


Figure 3. Structure of self attention mechanism fusion CNN.

In addition, as shown in Figure 4, the aim was to introduce representations on different dimensions by combining the (Q, K, V) dimensional transformation with CNN and local information extraction capabilities that will take the output with the $Reshape(CNN(Q^T))$, $Reshape(CNN(K^T))$, and $Reshape(CNN(V^T))$ of the SEF module as the input of the multi-head attention mechanism, which can be seen as introducing sequence information from different spaces and time series. Thus, the model enhances the spatial-polarization domain correlation from different spatial and temporal perspectives, which enables the model to perform better fuse the spatial-polarization domain features corresponding to the $\tilde{R}$ processed in this paper. Improving the model's understanding and utilization of the spatial-

polarization domain correlation information, namely better understand the contextual information input and improve the model's expressive power, thus improving the model's ability to model sequence diversity. The specific operation SEF multi-head self attention is shown in Algorithm 2.

Algorithm 1 Multi-Head Self Attention Fusion CNN Process

- 1: **Input:** Q, K, V (Query, Key, Value matrices)
- 2: $Q = X_{LN}^{[1]} W_Q, K = X_{LN}^{[1]} W_K, V = X_{LN}^{[1]} W_V$, where W_Q, W_K, W_V are learnable weight matrices
- 3: Perform linear mapping: $Linear(CNN(Q)), Linear(CNN(K)), Linear(CNN(V))$
- 4: Apply Scaled Dot-Product Attention (SDPA) as:

$$SDPA(Linear(CNN(Q)), Linear(CNN(K)), Linear(CNN(V)))$$

- 5: Perform parallel operations across heads. For the i -th head:

$$O_{(i)} = SDPA(Linear_{(i)}(CNN(Q)), Linear_{(i)}(CNN(K)), Linear_{(i)}(V))$$

- 6: Concatenate the outputs of the heads:

$$\hat{O} = Concat(O_{(1)}, O_{(2)}, \dots, O_{(H)})$$

- 7: Perform final linear mapping on the concatenated output:

$$O = Linear(\hat{O})$$

The aforementioned SEF module was added to each multi-head self-attention mechanism module $\{O_{(1)}, O_{(2)}, \dots, O_{(H)}\}$ to fuse multi-dimensional features $O_{(1), O_{(2)}, \dots, O_{(H)}}^{SEF}$ in the spatial-polarization domain, resulting in improved feature integration. This enhancement subsequently boosts the model's ability to understand and utilize the overall sequence relationships. The comprehensive modeling framework incorporating these elements is illustrated in Figure 5.

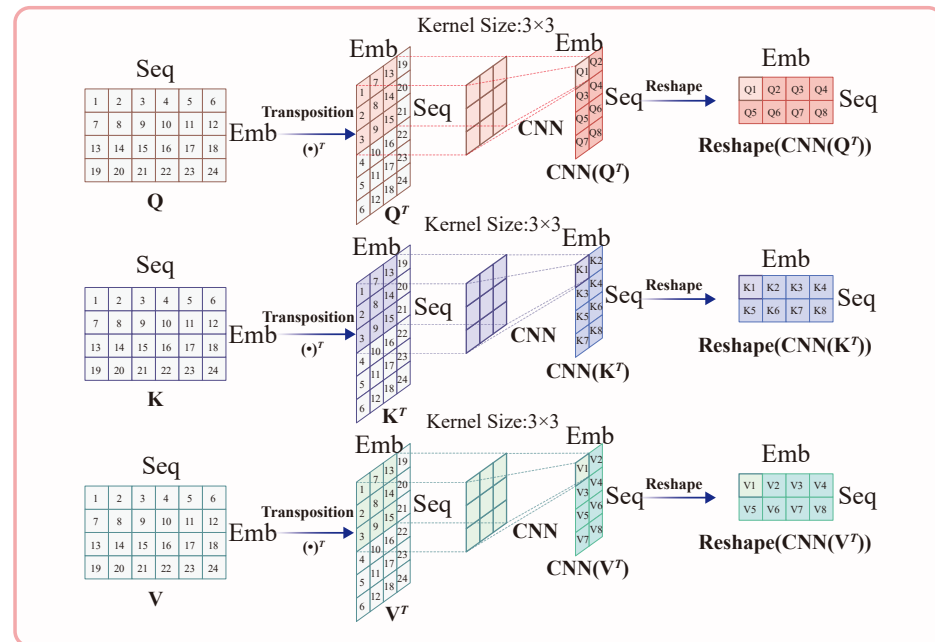


Figure 4. Structure of SEF self attention mechanism.

Algorithm 2 SEF Multi-Head Self Attention Process

- 1: **Step 1:** Perform dimension conversion of sequence and embedding as:

$$(Q^T, K^T, V^T)$$

- 2: **Step 2:** Perform feature fusion of sequence and embedding dimensions as:

$$CNN(Q^T), CNN(K^T), CNN(V^T)$$

where CNN represents convolution operations with a kernel size of 3×3 .

- 3: **Step 3:** Reshape the sequence and embedding dimensions:

$$Reshape(CNN(Q^T)), Reshape(CNN(K^T)), Reshape(CNN(V^T))$$

to complete feature fusion of sequence and embedding dimensions.

- 4: **Step 4:** Implement the self-attention mechanism after linear layer mapping. The overall equation is shown below:

$$O_{\{O_{(1)}, O_{(2)}, \dots, O_{(H)}\}}^{SEF} = Linear(Concat(SDPA(Linear_{(1)}(Reshape(CNN(Q^T))), Linear_{(1)}(Reshape(CNN(K^T))), Linear_{(1)}(Reshape(CNN(V^T))), \dots, SDPA(Linear_{(H)}(Reshape(CNN(Q^T))), Linear_{(H)}(Reshape(CNN(K^T))), Linear_{(H)}(Reshape(CNN(V^T))))))$$

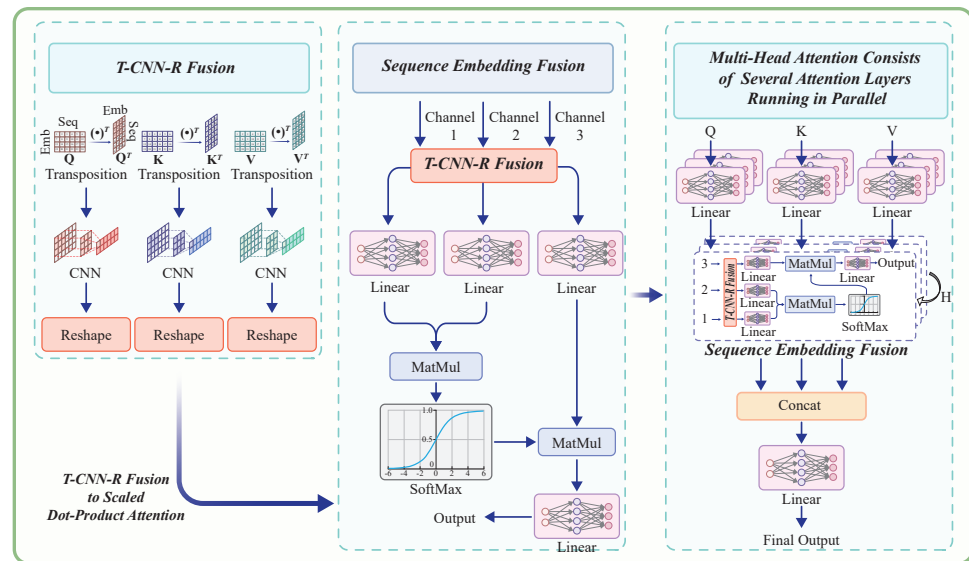


Figure 5. Structure of overall SEF multi-head self attention mechanism.

The goal was to enhance the model's expressive capability by learning the residual value between the input and output, as follows:

$$X^{[3]} = (((LayerNorm(X^{[1]} \oplus O_{\{O_{(1)}, O_{(2)}, \dots, O_{(H)}\}}^{SEF}) W^{[1]} + b^{[1]}) W^{[2]} + b^{[2]}) \oplus (X^{[1]} \oplus O_{\{O_{(1)}, O_{(2)}, \dots, O_{(H)}\}}^{SEF}))) \quad (22)$$

where $\oplus$ is used to indicate residual connections. Finally, layer normalization and reconstruction vector operations are performed to complete the output of the penultimate layer of the model, as follows:

$$\mathbf{G}^{[n-1]} = \text{Reshape}(\text{LayerNorm}(\mathbf{X}^{[3]})) \quad (23)$$

3.3. Optimization of Model

3.3.1. Positional Encoding Omission

In the transformer model, the positional embedding matrix $\mathbf{P}$ and word embedding are usually combined together. The equation for the combination of position and word embedding can be expressed as:

$$\mathbf{X}_{\text{InputEmbedding}} = \text{Embedding}(x) + \mathbf{P} \quad (24)$$

where $\mathbf{X}_{\text{InputEmbedding}}$ represents the input embedding matrix containing positional embeddings.

Equation (25) illustrates that, in the input phase of the model, the word embedding matrix and the positional embedding matrix are combined to produce an input that incorporates both semantic and positional information. However, the primary focus of this paper is on orientation information within the multi-dimensional features of the spatial-polarization domain in $\tilde{\mathbf{R}}$. This differs from the positional encoding used in NLP tasks, which captures the positional relationships between words in a sentence. In this context, the $\text{Embedding}(x)$ does not heavily rely on positional information [33], making positional encoding unnecessary for this model. Furthermore, incorporating encoding for longer sequence positions introduces additional parameters, which can increase computational and storage demands. Omitting positional encoding simplifies the model structure and significantly reduces the number of parameters, thereby alleviating the computational and storage burden. This reduction can improve both the training and inference efficiency of the model and make it easier to understand and debug.

3.3.2. Multi Task Parallel Processing

As discussed in this section, the concept that the CNN with shared model parameter [29] mechanism within the SEF module facilitates multi-dimensional information fusion in the spatial-polarization domain was leveraged. This approach ensures the correlation between multi-dimensional features and across multiple tasks. A parallel output structure for multi-tasks was designed and a multi-task weighted loss function was implemented to enhance the model's expressive power. This optimization allows the model to better fit the training data and improves its predictive performance and inference capabilities on unseen data, thereby enhancing the overall generalization ability of the model.

The following describes a parallel processing and updating model approach for the four multi-dimensional task designs in this study. Let the output of the second-to-last layer of the SEF transformer model be denoted as $\mathbf{G}^{[n-1]}$, with dimensions $S \times E$. The multi-task parallel output is represented by the following equation:

$$\begin{aligned} \mathbf{G}_{\theta}^n &= \text{Softmax}(\mathbf{W}_{\theta}^{[n]} \times \mathbf{G}^{[n-1]}) \\ \mathbf{G}_{\phi}^n &= \text{Softmax}(\mathbf{W}_{\phi}^{[n]} \times \mathbf{G}^{[n-1]}) \\ \mathbf{G}_{\gamma}^n &= \text{Softmax}(\mathbf{W}_{\gamma}^{[n]} \times \mathbf{G}^{[n-1]}) \\ \mathbf{G}_{\eta}^n &= \text{Softmax}(\mathbf{W}_{\eta}^{[n]} \times \mathbf{G}^{[n-1]}) \end{aligned} \quad (25)$$

where $\mathbf{W}_{\theta}^{[n]}$, $\mathbf{W}_{\phi}^{[n]}$, $\mathbf{W}_{\gamma}^{[n]}$, and $\mathbf{W}_{\eta}^{[n]}$ are the weight parameters of the final output layer, and $\mathbf{G}_{\theta}^n$, $\mathbf{G}_{\phi}^n$, $\mathbf{G}_{\gamma}^n$, and $\mathbf{G}_{\eta}^n$ are the output items.

Meanwhile, based on previous research [8], which indicated that polarization angle estimation typically performs worse than spatial angle estimation, a multi-task weighted

cross-entropy loss function was designed in the present study to address this issue. The multi-task loss function [28] is structured to account for the varying performance across different tasks. The expression for the loss function for a batch is given by:

$$\begin{aligned} Loss_{\theta} &= -\frac{1}{S_{bs}L} \sum_{j=1}^{S_{bs}} \sum_{i=1}^L y_{\theta_j^i} \log(G_{\theta_j^i}^n) + (1 - y_{\theta_j^i}) \log(1 - G_{\theta_j^i}^n) \\ Loss_{\varphi} &= -\frac{1}{S_{bs}L} \sum_{j=1}^{S_{bs}} \sum_{i=1}^L y_{\varphi_j^i} \log(G_{\varphi_j^i}^n) + (1 - y_{\varphi_j^i}) \log(1 - G_{\varphi_j^i}^n) \\ Loss_{\gamma} &= -\frac{1}{S_{bs}L} \sum_{j=1}^{S_{bs}} \sum_{i=1}^L y_{\gamma_j^i} \log(G_{\gamma_j^i}^n) + (1 - y_{\gamma_j^i}) \log(1 - G_{\gamma_j^i}^n) \\ Loss_{\eta} &= -\frac{1}{S_{bs}L} \sum_{j=1}^{S_{bs}} \sum_{i=1}^L y_{\eta_j^i} \log(G_{\eta_j^i}^n) + (1 - y_{\eta_j^i}) \log(1 - G_{\eta_j^i}^n) \end{aligned} \quad (26)$$

$$Loss_{\theta\varphi\gamma\eta} = \zeta_{\theta} \cdot Loss_{\theta} + \zeta_{\varphi} \cdot Loss_{\varphi} + \zeta_{\gamma} \cdot Loss_{\gamma} + \zeta_{\eta} \cdot Loss_{\eta} \quad (27)$$

where $y_{\theta_j^i}$, $y_{\varphi_j^i}$, $y_{\gamma_j^i}$, and $y_{\eta_j^i}$ are the label values of the i -th multilabel classification under the j -th sample, respectively, and $G_{\theta_j^i}^n$, $G_{\varphi_j^i}^n$, $G_{\gamma_j^i}^n$, and $G_{\eta_j^i}^n$ are the predicted values. S_{bs} is the size of batchsize; $Loss_{\theta}$, $Loss_{\varphi}$, $Loss_{\gamma}$, and $Loss_{\eta}$ represents the loss functions corresponding to the four tasks; ζ_{θ} , ζ_{φ} , ζ_{γ} , and ζ_{η} represent the elevation angle separately, azimuth angle, auxiliary polarization angle, polarization phase difference, and the weight of the loss function. The weights utilized in this paper are set to $\zeta_{\theta} = \zeta_{\varphi} < \zeta_{\gamma} = \zeta_{\eta}$.

Finally, the aforementioned multi-task weighted loss function $Loss_{\theta\varphi\gamma\eta}$ is used to back propagate the model and calculate the gradient. The model parameters are then updated using the Adam optimizer.

3.4. Angle Pairing for SEF Transformer

As mentioned above, a multi-task parallel module was designed to maintain the correlation between multi-dimensional features and multiple tasks by decoupling the problem into several parallel sub-problems, while this approach simplifies the overall model by reducing its complexity, it necessitates an additional step for angle pairing, since the outputs from multiple tasks consist of multi-class label results specific to their respective tasks. To address this, the GSF algorithm [8] is utilized in the angle pairing process. This process begins with representing the received data according to the signal reception model as:

$$X(t) = Q_{\theta,\varphi} H_{\gamma,\eta} S(t) + N(t) = Q_{\theta,\varphi} \tilde{S}_{\gamma,\eta}(t) + N(t) \quad (28)$$

where $Q_{\theta,\varphi} \in \mathbb{C}^{2M \times N}$ is the spatial response matrix, $H_{\gamma,\eta} \in \mathbb{C}^{N \times N}$ is the polarization response matrix, and $\tilde{S}_{\gamma,\eta}(t) \in \mathbb{C}^{N \times T}$ is the polarized source signal matrix. The signal covariance matrix is:

$$R = E(X(t)X^H(t)) = Q_{\theta,\varphi} R_{\tilde{S}}(\gamma, \eta) Q_{\theta,\varphi}^H + \sigma_n^2 I \quad (29)$$

When N' of N signals are coherent signals, then the rank of $R_{\tilde{S}}(\gamma, \eta)$ is $N - N' + 1$, there are $N - N' + 1$ large eigenvalues in R . Take the matrix composed of the $N - N' + 1$ eigenvectors corresponding to the large eigenvalues as the signal subspace U_s . In an ideal scenario, the signal subspace is a linear subspace of the array manifold tensor space, where there must exist a full rank matrix T_1 , such that:

$$U_s = Q_{\theta,\varphi} T_1 \quad (30)$$

where $\mathbf{U}_s \in \mathbb{C}^{2M \times (N-N'+1)}$ is the signal subspace, and $\mathbf{T}_1 \in \mathbb{C}^{N \times (N-N'+1)}$ is the transformation matrix. Moreover, according to Equation (30), the least squares solution of matrix $\mathbf{T}_1$ is:

$$\mathbf{T}_1 = \mathbf{Q}_{\theta,\varphi}^+ \mathbf{U}_s \quad (31)$$

where $(\cdot)^+$ represents the left inverse of a matrix. Additionally, $\mathbf{Q}_{\theta,\varphi}^+ = (\mathbf{Q}_{\theta,\varphi}^H \mathbf{Q}_{\theta,\varphi})^{-1} \mathbf{Q}_{\theta,\varphi}^H$ with $\mathbf{Q}_{\theta,\varphi}^+ \in \mathbb{C}^{(N-N'+1) \times 2M}$ is the pseudo-inverse matrix of $\mathbf{Q}_{\theta,\varphi}$.

When there is noise in space, the manifold of the signal array is not completely equal to the signal subspace. Further construction of fitting function is possible, and we estimate the spatial parameters of the signal by solving Equation (32):

$$\hat{\theta}, \hat{\varphi}, \hat{\mathbf{T}}_1 = \arg \min_{\theta, \varphi, \mathbf{T}} \|\mathbf{U}_s - \mathbf{Q}_{\theta,\varphi} \mathbf{T}_1\|_F^2 \quad (32)$$

where $\|\cdot\|_F^2$ represents the *Frobenius*-norm of a matrix. Additionally, Equation (32) can replace the matrix $\mathbf{T}_1$ in order to obtain:

$$\begin{aligned} \hat{\theta}, \hat{\varphi}, \hat{\mathbf{T}}_1 &= \arg \min_{\theta, \varphi} \|\mathbf{U}_s - \mathbf{Q}_{\theta,\varphi} \mathbf{Q}_{\theta,\varphi}^+ \mathbf{U}_s\|_F^2 \\ &= \arg \min_{\theta, \varphi} \|P_Q^\perp(\theta, \varphi) \mathbf{U}_s\|_F^2 \\ &= \arg \min_{\theta, \varphi} \text{tr} \{ P_Q^\perp(\theta, \varphi) \mathbf{U}_s \mathbf{U}_s^H P_Q^\perp(\theta, \varphi) \} \\ &= \arg \min_{\theta, \varphi} \text{tr} \{ P_Q^\perp(\theta, \varphi) \mathbf{U}_s \mathbf{U}_s^H \} \end{aligned} \quad (33)$$

where $P_Q^\perp(\theta, \varphi) = \mathbf{I} - \mathbf{Q}_{\theta,\varphi} \mathbf{Q}_{\theta,\varphi}^+$. Additionally, the estimation of spatial angle $(\hat{\theta}, \hat{\varphi})$ can be achieved through spatial domain search using Equation (33). $\mathbf{Q}_{\theta,\varphi}$ represents the array manifold composed of N signals. It requires a $2N$ dimensional search.

Similarly, the polarization parameters of the signal are estimated as follows Equation (34), and there is a full rank matrix $\mathbf{T}_2$, such that:

$$\mathbf{U}_s = \mathbf{Q}_{\theta,\varphi} \mathbf{H}_{\gamma,\eta} \mathbf{T}_2 = \mathbf{A}_{\theta,\varphi,\gamma,\eta} \mathbf{T}_2 \quad (34)$$

where $\mathbf{A}_{\theta,\varphi,\gamma,\eta} \in \mathbb{C}^{2M \times N}$ is the manifold matrix, and $\mathbf{T}_2 \in \mathbb{C}^{N \times (N-N'+1)}$ is the transformation matrix. Additionally, the least squares solution of $\mathbf{T}_2$ is:

$$\mathbf{T}_2 = \mathbf{A}_{\theta,\varphi,\gamma,\eta}^+ \mathbf{U}_s \quad (35)$$

where $\mathbf{A}_{\theta,\varphi,\gamma,\eta}^+ \in \mathbb{C}^{N \times 2M}$ is the pseudo-inverse matrix of $\mathbf{A}_{\theta,\varphi,\gamma,\eta}$. The polarization parameters can be estimated by Equation (36):

$$\hat{\gamma}, \hat{\eta}, \hat{\mathbf{T}}_2 = \arg \min_{\gamma, \eta} \|\mathbf{U}_s - \mathbf{Q}_{\hat{\theta}, \hat{\varphi}} \mathbf{H}_{\gamma, \eta} \mathbf{T}_2\|_F^2 \quad (36)$$

where $(\hat{\theta}, \hat{\varphi})$ is the estimated value of the spatial angle obtained from Equation (33). Therefore, estimated values of polarization parameters can be obtained:

$$\hat{\gamma}, \hat{\eta}, \hat{\mathbf{T}}_2 = \arg \min_{\gamma, \eta} \text{tr} \{ P_A^\perp(\hat{\theta}, \hat{\varphi}, \gamma, \eta) \mathbf{U}_s \mathbf{U}_s^H \} \quad (37)$$

where $P_A^\perp(\hat{\theta}, \hat{\phi}, \gamma, \eta) = I - A_{\hat{\theta}, \hat{\phi}, \gamma, \eta} A_{\hat{\theta}, \hat{\phi}, \gamma, \eta}^+$, polarization domain search can be achieved through Equation (37) obtain an estimate of polarization angle $(\hat{\gamma}, \hat{\eta})$, which completes the angle pairing process of spatial angle and polarization angle.

3.5. Model Efficiency

The computational complexity expressions for the linear transformation layer, layer normalization, CNN layer, multi-head self-attention, feedforward neural network, residual connection, parallel linear transformation layers, and generalized subspace fitting for angle pairing can be represented as follows:

$$\begin{aligned}
 \text{LinearComplexity} &= O(S \times E \times E_{\text{out}}) \\
 \text{LayerNormComplexity} &= O(S \times E) \\
 \text{CNNComplexity} &= O(S \times k^2 \times C_{\text{in}} \times C_{\text{out}}) \\
 \text{AttentionComplexity} &= O(H \times 3 \times S \times E^2) \\
 \text{FFNComplexity} &= O(S \times E \times E_{\text{out}}) \\
 \text{ResidualComplexity} &= O(S \times E) \\
 \text{ParallelLinearComplexity} &= O(N \times 4 \times S \times E \times L_n) \\
 \text{GSFComplexity} &= O((2M)^3 + 2M(N^4))
 \end{aligned} \tag{38}$$

where S is the sequence dimension of the input, E is the embedding dimension of the input, E_{out} is the embedding dimension of the output, the kernel size of the convolution is $k \times k$, the input channel is C_{in} , the output channel is C_{out} , H is the number of heads in the multi-head self-attention mechanism, N is the number of sources, and L_n is the dimension of the output of the last layer.

The formula for the model parameters of the linear transformation layer, layer normalization, CNN layer, multi-head self-attention, feed-forward neural network, and parallel linear transformation layer can be expressed as follows:

$$\begin{aligned}
 \text{LinearParameters} &= E \times E_{\text{out}} + E_{\text{out}} \\
 \text{LayerNormParameters} &= 2 \times E \\
 \text{CNNParameters} &= k^2 \times C_{\text{in}} \times C_{\text{out}} \\
 \text{AttentionParameters} &= H \times 3 \times (E \times E) \\
 \text{FFNParameters} &= 2 \times (E \times E_{\text{out}}) \\
 \text{ParallelLinearParameters} &= 4 \times (E \times L_n + L_n)
 \end{aligned} \tag{39}$$

4. Numerical Simulation

To evaluate the effectiveness of the model, numerical simulations were conducted. The experimental array structure in this paper is UCA. Additionally, due to the data-driven nature of the proposed method, the array structure is not limited to UCA and can be extended to other array structure. In the simulations, the following methods were compared: SEF Trans, which represents the transformer with sequence and embedding fusion; Trans, which denotes the transformer without sequence and embedding fusion; MLP, which stands for multilayer perceptron; and GSF, which refers to the generalized subspace fitting algorithm. The simulation dataset consists of 2,000,000 data points, with 80% allocated for training, 10% for validation, and 10% for testing. The experiment covers a spatial angle range of $[-30^\circ, 30^\circ]$ and a polarization angle range of $[0^\circ, 60^\circ]$, with a quantization interval of $\Delta\phi = 0.1^\circ$. The simulation assumes 2 incoming signal sources. The multi-task weighted loss function $\text{Loss}_{\theta\phi\gamma\eta}$ incorporates four angle weights: $\zeta_\theta = 0.2$, $\zeta_\phi = 0.2$, $\zeta_\gamma = 0.3$, and $\zeta_\eta = 0.3$. The experimental environment for this study was Python (version 3.7, Python Software Foundation, Wilmington, DE, USA), while data generation

was performed using MATLAB 2021b. The simulation parameters used in the experiments are listed in Tables 1 and 2.

Table 1. Array parameter table.

Array Parameters	Value Description
Number of elements	$M = 10$
Number of snapshots	$N = [100 : 100 : 500]$
Signal frequency	$f = 6 \text{ GHz}$
Wavelength	$\lambda = 0.05 \text{ m}$
Range of incoming wave direction	$[-30^\circ, 30^\circ]$
Polarization direction range	$[0^\circ, 60^\circ]$
Quantitative unit	$\Delta\phi = 0.1^\circ$
Uniform circular array radius	0.05×3.04
SNR	$[0 \text{ dB} : 3 \text{ dB} : 21 \text{ dB}]$

Table 2. Neural network parameter table.

Neural Network Parameter	Value Description
Learning rate	$\alpha = 0.001$
Epoch	25
Sequence length	25
Embedding size	16
Number of multi-head self-attention	8
Batch size	200
Overfitting factor (dropout)	0.05

To compare the training convergence and performance of various algorithms, the SNR of this experiment was set to 15 dB and 200 snapshots were used. The proposed methods—SEF Trans, Trans, and MLP—were trained over different epochs. The training loss curves for these methods are shown in Figures 6–8. As depicted, the SEF Trans method quickly achieved superior convergence compared to the other two methods and maintains relative stability after convergence. In this study, we employed the tree-structured parzen estimator (TPE) method [34] for hyperparameter optimization of the neural network. TPE is a Bayesian optimization technique that explores the hyperparameter space by constructing conditional probability models. In each trial, TPE dynamically balances exploration and exploitation by selecting the most promising hyperparameters likely to improve performance, thus reducing the number of trials required. We chose TPE because it dynamically balances exploration and exploitation, making it particularly well-suited for high-dimensional and complex hyperparameter search problems. Our experimental results demonstrate that this approach has led to satisfactory performance optimization.

To compare the testing accuracy of various algorithms, the SNR was set to 15 dB, and the DOA estimation error was evaluated under the condition that it is $\leq 1^\circ$ for each epoch of 200 snapshots. The test accuracy is computed using the following formula:

$$Accuracy = \frac{\sum_{i=1}^{S_N} \mathbb{I}(|\Delta\theta_{1,i}| \leq \varepsilon \text{ and } |\Delta\theta_{2,i}| \leq \varepsilon)}{S_N} \quad (40)$$

where S_N is the total number of test samples, $\Delta\theta_{1,i}$ and $\Delta\theta_{2,i}$ represent the errors in the two angles (the difference between the estimated and true angles) for the i -th sample, and ε is the error threshold; this experiment is set to 1° . $\mathbb{I}(\cdot)$ denotes the indicator function, which is 1 if the condition inside the parentheses is true (i.e., both angle errors are less than or equal to 1°) and 0 otherwise.

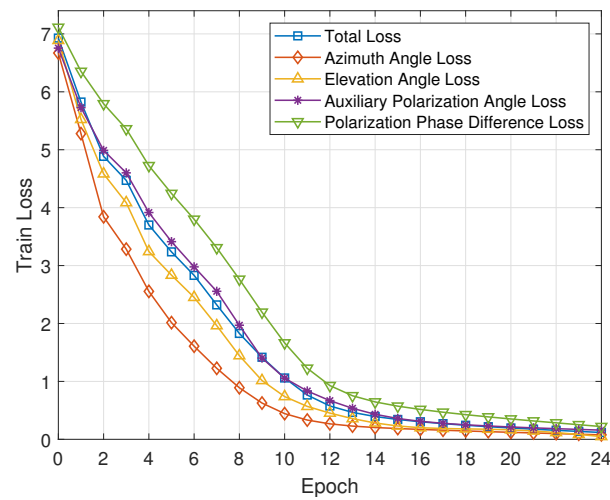


Figure 6. Relationship curve between epoch and loss (SEF Trans method).

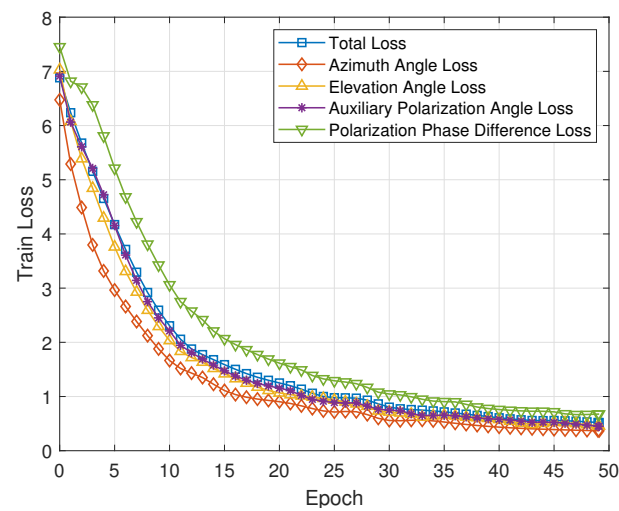


Figure 7. Relationship curve between epoch and loss (Trans method).

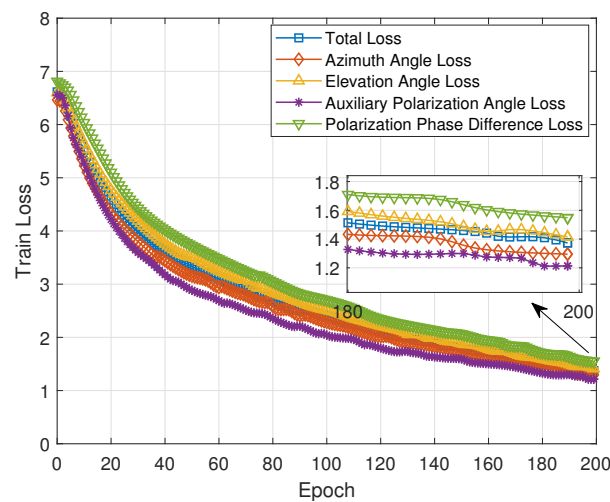


Figure 8. Relationship curve between epoch and loss (MLP method).

A comparative simulation was conducted to assess the test accuracy of the proposed SEF Trans method against the other two comparison algorithms. As shown in Figures 9–11, the test accuracy curves reveal that the SEF Trans method achieved the highest DOA

estimation accuracy with the fewest number of epochs. Additionally, the polarization phase difference, which involves processing signal phase information, is more susceptible to noise and interference. In contrast, the polarization assist angle, relying on signal amplitude or other more robust features, demonstrated better estimation performance. This was confirmed during simulation testing, where the accuracy of the auxiliary polarization angle was found to be superior to that of the polarization phase difference, consistent with the observed characteristics [8].

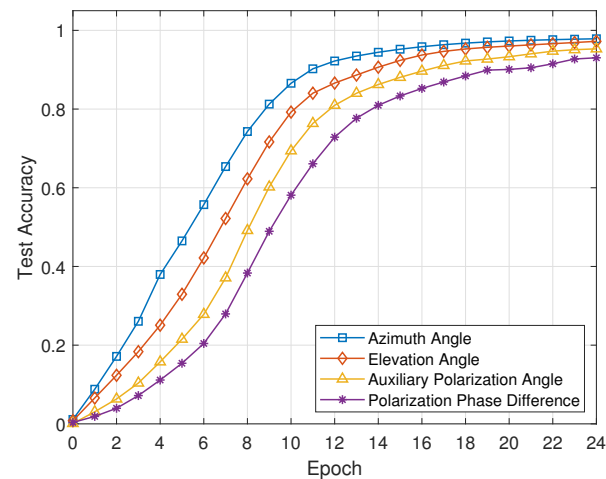


Figure 9. Relationship curve between epoch and test accuracy (SEF Trans method).

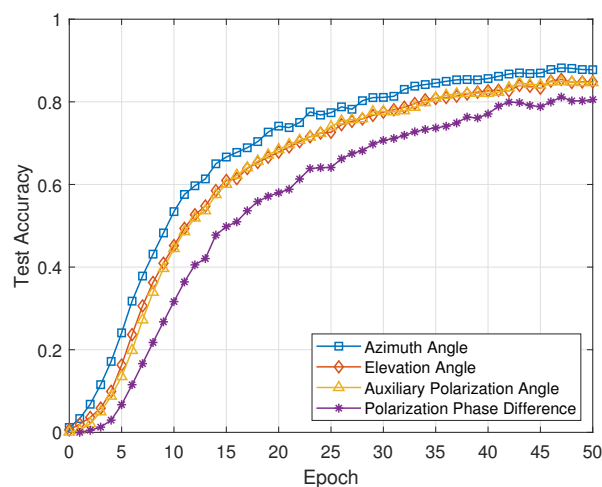


Figure 10. Relationship curve between epoch and test accuracy (Trans method).

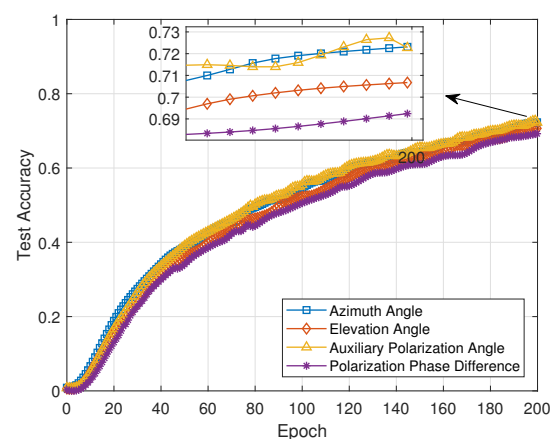


Figure 11. Relationship curve between epoch and test accuracy (MLP method).

To verify the feasibility of omitting positional encoding (PE), an experiment was conducted with an SNR of 15 dB and 200 snapshots. A comparative analysis was performed on the accuracy of the SEF Trans method with and without positional encoding, with the error criterion set to less than or equal to one degree. As shown in Figure 12, the results indicate that positional encoding does not significantly enhance performance. Therefore, it can be concluded that positional encoding can be omitted in this context.

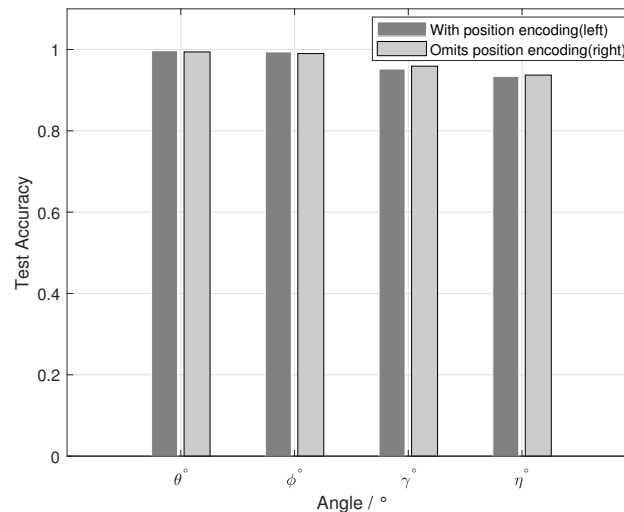


Figure 12. Testing accuracy comparison of SEF Trans method for position encoding.

To evaluate the model efficiency of the proposed method, the following table presents a comparison of different data-driven class methods. As shown in Table 3, it is observed that the SEF Trans method without PE outperforms other methods in terms of computational complexity, number of parameters, and inference time. This suggests that the model efficiency of the SEF Trans method, when positional encoding is omitted, is superior to that of the other methods.

Table 3. Model efficiency comparison.

Model	Computational Complexity (M)	Number of Parameters (M)	Inference Time (s)
SEF Trans with PE	30.31	29.79	1.705×10^{-2}
SEF Trans no PE	30.31	29.79	1.352×10^{-2}
Trans	33.15	32.79	1.999×10^{-2}
MLP	90.5	90.5	2.417×10^{-2}

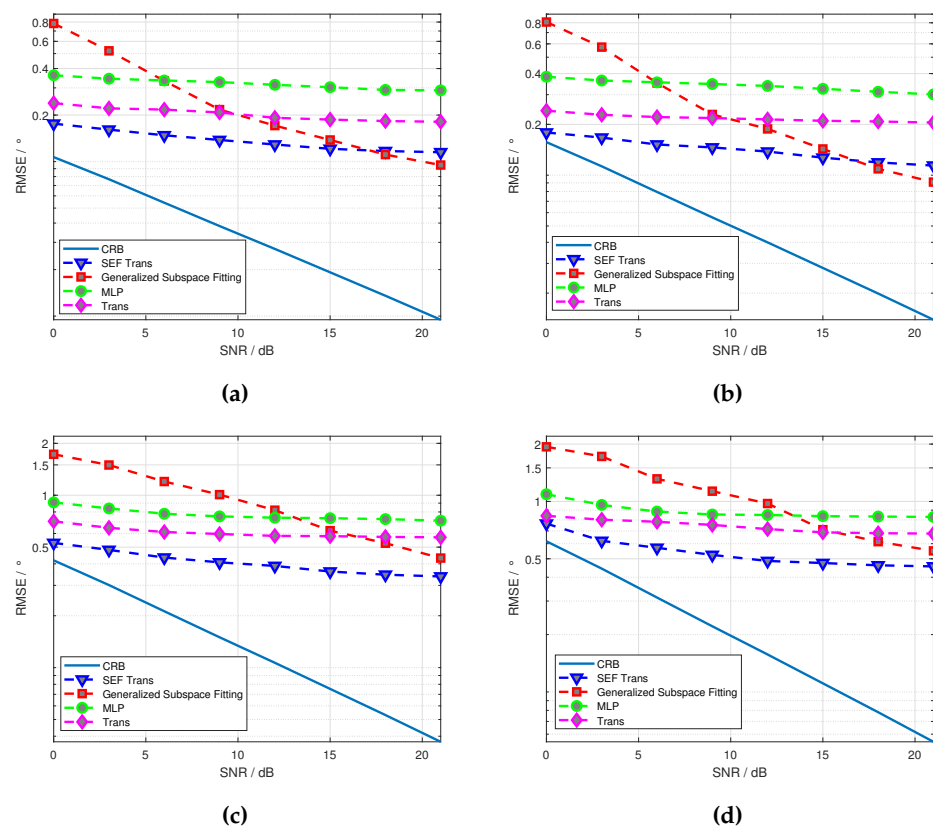
Table 4 shows that the comparison of inference time for different algorithms in the task of coherent polarization–DOA estimation under the same GPU test environment. In this test, the search range of the spatial angle for the model-driven algorithm is set to $[-5^\circ : 5^\circ]$, the search range of the polarization angle is $[0^\circ : 10^\circ]$, and the search step is 0.1° . The experiment demonstrates that the SEF Trans method outperforms traditional algorithms such as the polarization smoothing-oblique projection [6] and the generalized subspace fitting (GSF) [8] methods in terms of inference time. This indicates that the SEF Trans method not only significantly reduces inference time but also exhibits higher real-time performance and computational efficiency on the same hardware platform. This demonstrates the potential of the SEF trans method for real-time signal processing tasks.

Table 4. Real-time comparison of the same GPU test environment.

Methods	Test Environment	Inference Time (s)
SEF Trans	GPU	0.864×10^{-3}
Generalized Subspace Fitting	GPU	81.983
Polarization Smoothing and Oblique Projection	GPU	0.373

To verify estimation performance under varying SNR conditions, the SNR was set from 0 to 21 dB in 3 dB increments, with 200 snapshots used for comparison. The RMSE and CRB simulation results for each method are shown in Figure 13. As indicated in Figure 13a,b, while the GSF algorithm exhibited slightly better estimation accuracy than the SEF Trans method when the SNR was between 18 and 21 dB, the SEF Trans method outperformed the Trans, MLP, and GSF algorithms when the SNR was between 0 and 15 dB. Figure 13c,d further confirm that the proposed SEF Trans algorithm demonstrated superior estimation performance in both the spatial and polarization domains.

To evaluate estimation performance from the perspective of the number of snapshots, the experiment compared simulation performance under an SNR of 0 dB with snapshots ranging from 100 to 500 in increments of 100. The RMSE and CRB simulation results for each method are shown in Figure 14. As depicted in the RMSE curves for both spatial and polarization angles, the proposed SEF Trans method consistently demonstrated significant advantages as the number of snapshots increased. This indicates that the proposed method maintained strong learning and solving capabilities for multi-dimensional feature data in the spatial-polarization domain, as described in the present paper.

**Figure 13.** Relationship curve between RMSE and SNR (situation with matched SNR). (a) Azimuth angle. (b) Elevation angle. (c) Auxiliary polarization angle. (d) Polarization phase difference.

To analyze the generalization ability of the proposed algorithm concerning SNR, the SNR for the training set was set from 0 to 21 dB in 3 dB increments, with 200 snapshots. The

test set was set from 2 to 20 dB in 3 dB increments, resulting in a maximum SNR mismatch of 2 dB. Figure 15 illustrates the RMSE results for DOA estimation of both spatial and polarization angles. Compared with other data-driven algorithms, the SEF Trans method demonstrated robust performance even with SNR mismatches between the training and testing data sets. Although the GSF algorithm showed higher accuracy than the MLP and Trans algorithms as SNR increased, the accuracy of the SEF Trans method was comparable to that of the GSF algorithm. This indicates that the proposed SEF Trans algorithm exhibited superior generalization performance.

To evaluate the generalization ability of the algorithm concerning the number of snapshots, the training set was set with an SNR of 15 dB and snapshots ranging from 100 to 500 in increments of 100. The test set had snapshots ranging from 150 to 450 in increments of 100, resulting in a maximum difference of 50 snapshots. Figure 16 illustrates that, despite the mismatch in the number of snapshots between the training and testing datasets, the proposed algorithm maintains strong performance. Figure 16a,b show that the spatial angle estimation accuracy of this algorithm, after generalization, was comparable to the GSF algorithm at higher SNR levels, and the proposed method outperforms the Trans and MLP algorithms in terms of accuracy. Additionally, Figure 16c,d reveal that the generalized polarization angle estimation accuracy of the proposed algorithm generally surpassed that of the GSF algorithm. These results indicate that the proposed algorithm exhibited high generalization and robustness.

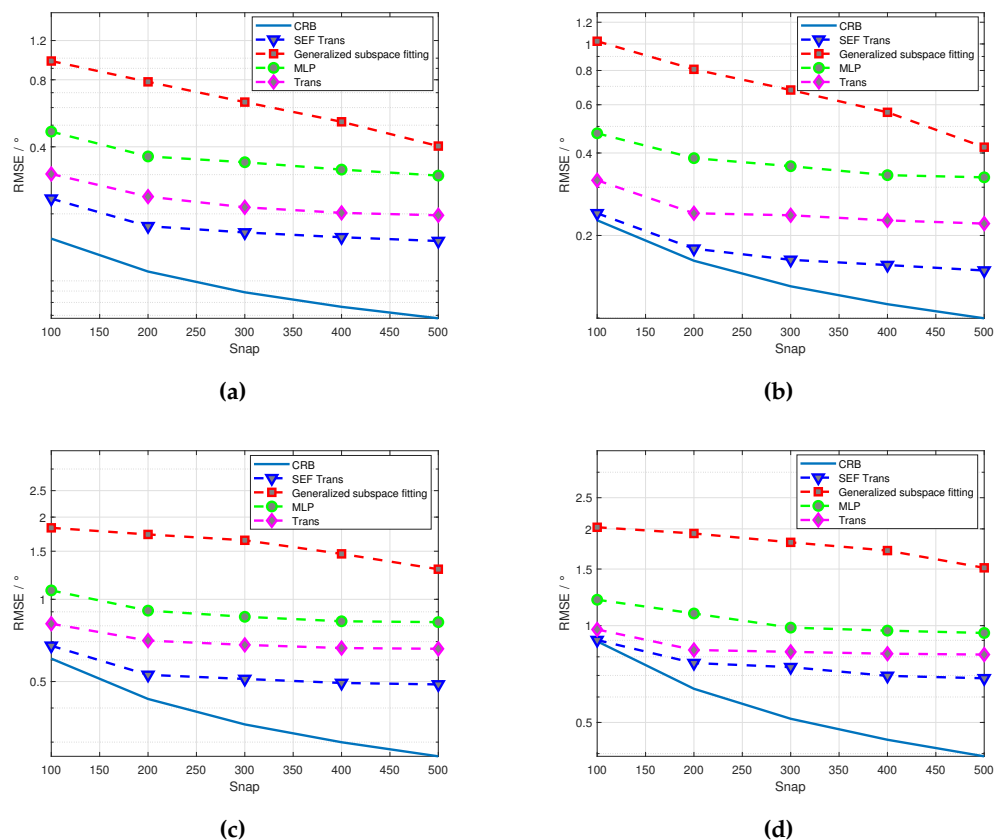


Figure 14. Relationship curve between RMSE and snapshots (situation with matched snapshots). (a) Azimuth angle. (b) Elevation angle. (c) Auxiliary polarization angle. (d) Polarization phase difference.

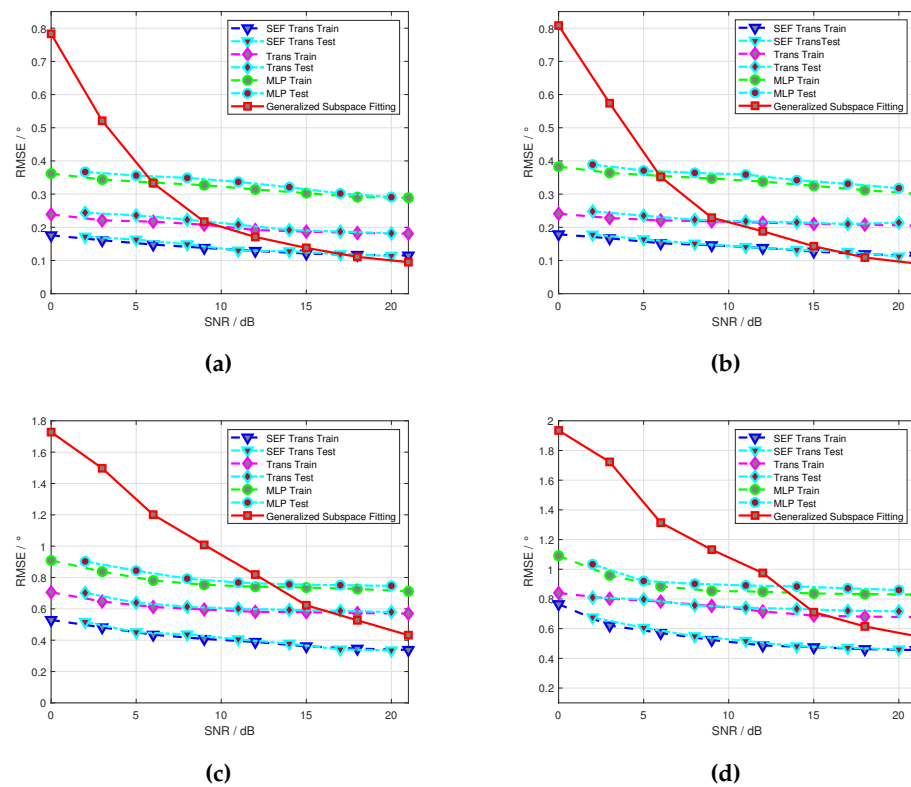


Figure 15. Relationship curve between RMSE and SNR (situation with mismatched SNR). (a) Azimuth angle. (b) Elevation angle. (c) Auxiliary polarization angle. (d) Polarization phase difference.

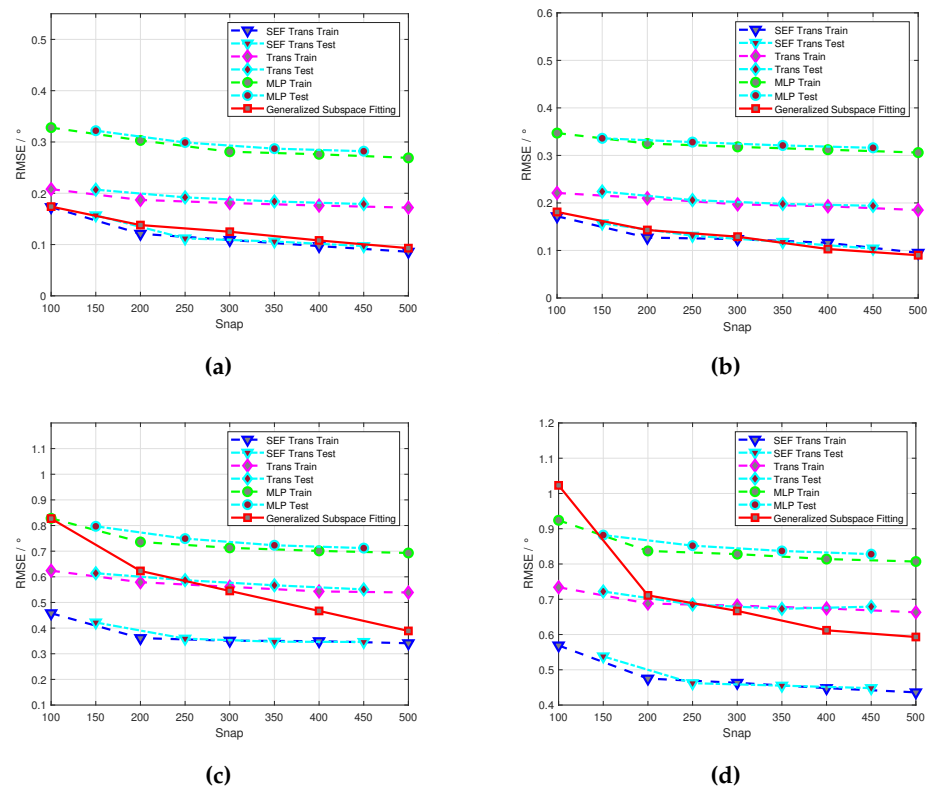


Figure 16. Relationship curve between RMSE and snapshots (situation with mismatched snapshots). (a) Azimuth angle. (b) Elevation angle. (c) Auxiliary polarization angle. (d) Polarization phase difference.

5. Conclusions

In this study, a 2D coherent polarization–DOA estimation method based on the SEF transformer was proposed. Drawing inspiration from the multi-head self-attention mechanism of the Transformer model, which excels in multi-level semantic recognition, the method utilizes Transformer multi-task text inference to achieve joint polarization–DOA estimation. The SEF module enhances the model’s capability to understand spatial-polarization domain correlations by fusing local sequence and embedding information from various dimensions, thereby improving polarization–DOA estimation accuracy. Additionally, the CNN-based shared model parameters within the SEF module are used to design a multi-task weighted loss function, optimizing the model’s expressive ability and generalization performance. The decision to omit positional encoding further enhances computational efficiency by focusing on spatial-polarization domain features rather than positional information. Simulation results indicate that the SEF Trans method outperformed other data-driven methods in terms of estimation accuracy, generalization ability, and computational efficiency, and showed superior performance to existing model-driven algorithms, particularly at low SNR. Future work will explore a more accurate gridless 2D coherent polarization–DOA estimation method using the Transformer model to address accuracy issues related to grid mismatch.

Author Contributions: Z.W.: Conceptualization, Methodology, and Writing—review and editing. J.W.: Data curation and Writing—original draft. Z.Z.: Investigation and Visualization. All authors have read and agreed to the published version of the manuscript.

Funding: This work was mainly supported by Major scientific and technological innovation projects of Shandong Province of China (Grant No. 2022ZLZX04 and 2021ZLZX05) and the National Natural Science Foundation of China (No. 62071144).

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Schmidt, R. Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas Propag.* **1986**, *34*, 276–280. [\[CrossRef\]](#)
- Roy, R.; Kailath, T. ESPRIT-estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust. Speech Signal Process.* **1989**, *37*, 984–995. [\[CrossRef\]](#)
- Costa, M.; Richter, A.; Koivunen, V. DoA and polarization estimation for arbitrary array configurations. *IEEE Trans. Signal Process.* **2012**, *60*, 2330–2343. [\[CrossRef\]](#)
- Weiss, A.J.; Friedlander, B. Analysis of a signal estimation algorithm for diversely polarized arrays. *IEEE Trans. Signal Process.* **1993**, *41*, 2628–2638. [\[CrossRef\]](#)
- Friedlander, B. The extended manifold for antenna arrays. *IEEE Trans. Signal Process.* **2020**, *68*, 493–502. [\[CrossRef\]](#)
- Xu, Y.; Liu, Z. Polarimetric angular smoothing algorithm for an electromagnetic vector-sensor array. *IET Radar Sonar Navig.* **2007**, *1*, 230–240. [\[CrossRef\]](#)
- Diao, M.; An, C. Direction finding of coexisted independent and coherent signals using electromagnetic vector sensor. *J. Syst. Eng. Electron.* **2012**, *23*, 481–487. [\[CrossRef\]](#)
- Xu, Y.; Liu, Z. Joint angle-polarization estimation via generalized signal-subspace fitting. *Trans. Beijing Inst. Technol.* **2010**, *30*, 835–839.
- Viberg, M.; Ottersten, B. Sensor array processing based on subspace fitting. *IEEE Trans. Signal Process.* **1991**, *39*, 1110–1121. [\[CrossRef\]](#)
- Fei, Y.; Cao, H.; Wu, Y.; Chen, X.; Chen, L. Doa estimation in non-uniform noise using matrix completion via alternating projection. *IEEE OJAP* **2021**, *2*, 281–285. [\[CrossRef\]](#)
- Pavel, S.R.; Chowdhury, M.W.T.; Zhang, Y.D.; Shen, D.; Chen, G. Machine learning-based direction-of-arrival estimation exploiting distributed sparse arrays. In Proceedings of the IEEE 2021 55th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 31 October–3 November; pp. 241–245.
- Donelli, M.; Viani, F.; Rocca, P.; Massa, A. An innovative multiresolution approach for DOA estimation based on a support vector classification. *IEEE T. Antenn. Propag.* **2009**, *57*, 2279–2292. [\[CrossRef\]](#)
- El Gonnouni, A.; Martinez-Ramon, M.; Rojo-Álvarez, J.L.; Camps-Valls, G.; Figueiras-Vidal, A.R.; Christodoulou, C.G. A support vector machine MUSIC algorithm. *IEEE T. Antenn. Propag.* **2012**, *60*, 4901–4910. [\[CrossRef\]](#)

14. Harkouss, Y. Direction of arrival estimation in multipath environments using deep learning. *Int. J. Commun. Syst.* **2021**, *34*, 1–23. [\[CrossRef\]](#)
15. Papageorgiou, G. K.; Sellathurai, M.; Eldar, Y. C. Deep networks for direction-of-arrival estimation in low SNR. *IEEE Trans. Signal Process.* **2021**, *69*, 3714–3729. [\[CrossRef\]](#)
16. Wu, L.L.; Huang, Z.T. Coherent SVR learning for wideband direction-of-arrival estimation. *IEEE Signal Process. Lett.* **2019**, *26*, 642–646. [\[CrossRef\]](#)
17. Ahmed, A.M.; Eissa, O.; Sezgin, A. Deep autoencoders for DOA estimation of coherent sources using imperfect antenna array. In Proceedings of the 2020 Third International Workshop on Mobile Terahertz Systems (IWMTS), Essen, Germany, 6–7 July, 2020; pp. 1–5.
18. Yuan, Y.; Wu, S.; Wu, M.; Yuan, N. Unsupervised learning strategy for direction-of-arrival estimation network. *IEEE Signal Process. Lett.* **2021**, *28*, 1450–1454. [\[CrossRef\]](#)
19. Xiang, H.; Chen, B.; Yang, T.; Liu, D. Improved de-multipath neural network models with self-paced feature-to-feature learning for DOA estimation in multipath environment. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5068–5078. [\[CrossRef\]](#)
20. Xiang, H.; Chen, B.; Yang, M.; Yang, T.; Liu, D. A novel phase enhancement method for low-angle estimation based on supervised DNN learning. *IEEE Access* **2019**, *7*, 82329–82336. [\[CrossRef\]](#)
21. Xiang, H.; Chen, B.; Yang, T.; Liu, D. Phase enhancement model based on supervised convolutional neural network for coherent DOA estimation. *Appl. Intell.* **2020**, *50*, 2411–2422. [\[CrossRef\]](#)
22. Xiang, H.; Chen, B.; Yang, M.; Xu, S. Angle separation learning for coherent DOA estimation with deep sparse prior. *IEEE Commun. Lett.* **2020**, *25*, 465–469. [\[CrossRef\]](#)
23. Liu, Z.M.; Zhang, C.; Philip, S.Y. Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections. *IEEE T. Antenn. Propag.* **2018**, *66*, 7315–7327. [\[CrossRef\]](#)
24. Wu, Z.; Wang, J. Small Sample Coherent DOA Estimation Method Based on S2S Neural Network Meta Reinforcement Learning. *Sensors* **2023**, *23*, 1546. [\[CrossRef\]](#)
25. Dogra, V.; Verma, S.; Woźniak, M.; Shafi, J.; Ijaz, M.F.; et al. Shortcut Learning Explanations for Deep Natural Language Processing: A Survey on Dataset Biases. *IEEE Access* **2024**, *12*, 26183–26195. [\[CrossRef\]](#)
26. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inform. Proc. Syst.* **2017**, *30*, 5998–6008.
27. Guo, Y.; Zhang, Z.; Huang, Y. Dual Class Token Vision Transformer for Direction of Arrival Estimation in Low SNR. *IEEE Signal Process. Lett.* **2023**, *31*, 76–80. [\[CrossRef\]](#)
28. Vandenhende, S.; Georgoulis, S.; Van Gansbeke, W.; Proesmans, M.; Dai, D.; Van Gool, L. Multi-task learning for dense prediction tasks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 3614–3633. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74. [\[CrossRef\]](#)
30. Hussain, A.A.; Tayem, N.; Soliman, A.H. FPGA hardware implementation of computationally efficient DOA estimation of coherent signals. In Proceedings of the 2021 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET) IEEE, Bali, Indonesia, 4–5 November 2021; pp. 103–108.
31. Jiang, R.; Ye, W. Hardware–Algorithm Codesigned Low-Latency and Resource-Efficient OMP Accelerator for DOA Estimation on FPGA. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2024**, *10*, 1–14. [\[CrossRef\]](#)
32. Qiu, Z.; Wei, P.; Yao, M.; Zhang, R.; Kuang, Y. Channel Pruning Method Based on Decoupling Feature Scale Distribution in Batch Normalization Layers. *IEEE Access* **2024**, *12*, 48865–48880. [\[CrossRef\]](#)
33. Li, X.; Li, S.; Zhang, X.L.; Rahardja, S. Transformer-Based End-to-End Speech Translation With Rotary Position Embedding. *IEEE Signal Process. Lett.* **2024**, *31*, 371–375. [\[CrossRef\]](#)
34. Valsecchi, C.; Consonni, V.; Todeschini, R.; Orlandi, M.E.; Gosetti, F.; Ballabio, D. Parsimonious Optimization of Multitask Neural Network Hyperparameters. *Molecules* **2021**, *26*, 7254. [\[CrossRef\]](#) [\[PubMed\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.