StHCFormer: A Multivariate Ocean Weather Predicting Method Based on Spatiotemporal Hybrid Convolutional Attention Networks

Lianlei Lin, Zongwei Zhang, Hangyi Yu, Junkai Wang, Sheng Gao, Hanqing Zhao, and Jiaqi Zhang

***Abstract*—Ocean weather prediction is crucial for various applications, such as global climate prediction, marine environmental protection, and offshore production. However, current data-based marine weather prediction methods have limitations when predicting multiple variables in a particular area, failing to meet the efficiency and accuracy requirements of practical applications. In the realm of ocean weather variations, the presence of highly interconnected spatial and temporal continuations, coupled with the mutual influence of individual variables, underscores the utmost importance of effectively capturing dynamic correlations encompassing space, time, and variables to accurately predict ocean weather. To address this, we developed a novel approach called StHCFormer, which is a multivariate spatiotemporal hybrid convolutional attention network. The first key component of StHCFormer is the spatiotemporal hybrid convolutional attention (StHCA) module, which leverages a hybrid convolutional attention mechanism to explore both global spatial representations and local features. Additionally, the module incorporates temporal attention to capture the temporal dependence of weather records and effectively captures the dynamic correlations among multiple variables through channel deflation and weighted residuals. To ensure balanced variable losses, we introduced the concept of homoscedasticity uncertainty loss to dynamically adjust the multitask weights. This guarantees a global optimal solution and leads to more accurate multivariate ocean weather prediction. Finally, we conducted a comprehensive evaluation and comparison of the StHCFormer model with other state-of-the-art algorithms using the ERA5 dataset in the Philippine Sea. The results demonstrated that StHCFormer outperforms existing methods in marine multivariate field weather prediction.**

***Index Terms*—Deep learning, multivariate prediction, ocean weather forecast, spatiotemporal hybrid convolutional attention, spatiotemporal prediction.**

I. Introduction

THE ocean occupies 70% of the Earth's surface area. Its rich resources are important for human survival, and its variable climate always influences global climate change. For example, changes in sea surface temperature (SST) cause prominent global climate extremes, such as El Niño and global warming. Sea breeze levels and the formation of catastrophic-level waves are directly related and greatly impact the safety of offshore production. Therefore, accurately predicting ocean weather changes is of great significance for marine environmental protection, extreme climate prediction, and offshore production activities. However, accurately predicting marine weather in spatial and temporal environments is still greatly challenging because of the close spatial connections of the ocean, the rapid rate of weather changes, many influencing factors, and the intrinsic correlation between multiple regions and these factors, which impact and interact with each other.

The existing methods for ocean weather prediction are mainly divided into two directions: numerical models and data-driven models. Numerical models are constructed by a series of complex thermodynamic and physical equations representing the ocean and integrating the physical connections between various factors. Although these models exhibit high accuracy, they require enormous computational effort and long calculation times, and their general hardware conditions operating the models difficult [1], [2], [3], [4], [5].

Data-driven models construct predictions by learning the internal variation law of time series data to predict future records. These models are mainly categorized as statistical models, machine learning models, and deep learning models. The autoregressive integrated moving average model (ARIMA) is a classic statistical method [6] that has a simple model structure and requires only endogenous variables without other exogenous variables and can effectively extract time series of time relationships. However, the model has strict requirements for time series data stability. Commonly used machine learning methods include linear regression [7], support vector machine (SVM) [8], [9], and artificial neural network (ANN) [10]. In [11] used the K-nearest neighbor (KNN) algorithm to accurately forecast ocean surface currents 24 hours in advance. Khosravi et al. [12] used various machine learning algorithms to predict wind speed and wind direction, demonstrating the superior performance of the support vector regression (SVR) algorithm for wind field prediction. He et al. [13] derived a robust SVM-based SST prediction model, improving the SST trend prediction and significantly improving the nonstationary SST time series prediction of the model. However, these algorithms rely heavily on the

Manuscript received 12 September 2023; revised 12 December 2023; accepted 9 January 2024. Date of publication 16 January 2024; date of current version 29 January 2024. *(Corresponding author: Lianlei Lin.)*

Lianlei Lin, Zongwei Zhang, Hangyi Yu, Junkai Wang, Sheng Gao, and Hanqing Zhao are with the School of Electronic Information Engineering, Harbin Institute of Technology, Harbin 150000, China (e-mail: linlianlei@hit.edu.cn; 22b905067@stu.hit.edu.cn; 22s005058@stu.hit.edu.cn; karry@stu.hit.edu.cn; wsgs1415926@gmail.com; 23b305003@stu.hit.edu.cn).

Jiaqi Zhang is with the School of Physics, Northeast Normal University, Changchun 130000, China (e-mail: jqzhang@nenu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3354254

effectiveness of feature engineering. For small-scale data, such methods can use patterns to achieve better prediction results. However, traditional machine learning algorithms are somewhat inadequate when handling large-scale ocean weather data, such as weather prediction over multiple regions.

Benefiting from GPU computing power enhancements, deep learning techniques have been rapidly developed, and recurrent neural network (RNN) [14] have been proposed. However, RNN is prone to problems such as gradient disappearance and explosion when handling long time series prediction. To address gradient propagation, the long short-term memory network (LSTM) [15] and its variant network, the gated recurrent unit (GRU) [16], were proposed. These networks are capable of longer time predictions than RNN. Based on this, Zhang et al. [17] proposed the LSTM-CFCC algorithm, which successfully achieved regional SST prediction using multiple LSTM units to model the grid point data separately. Obara and Nakamura [18] combined migration learning and LSTM for significant wave height (SWH) prediction, modeling the SWH of several points separately and considering other influencing factors to improve the SWH prediction accuracy. Xie et al. [19] used an encoder–decoder structure and GRU as an encoder–decoder with a self-attention mechanism to assist in SST prediction. These methods are mostly for single-point or multipoint prediction of ocean climate factors, and they neglect spatial information, leading to low prediction accuracies for regional variables. For spatiotemporal feature extraction of spatiotemporal field data, convolutional neural networks (CNNs), and RNNs exhibit good coupling. Shi et al. [20] combined a CNN and LSTM and proposed the ConvLSTM, which used convolution to obtain spatial features and LSTM to obtain temporal dependence for rainfall prediction. Lin et al. [21] proposed SA-ConvLSTM, using self-attention to obtain an additional memory module M and thus enhance the long time series prediction of ConvLSTM. Zhou et al. [22] developed a multilayer fusion recurrent neural network (MLFrnn) to predict sea surface height anomaly (SSHA) by learning long-term dependencies within the SSHA time series and spatial correlations between neighboring and remote areas. The receptive field property of the convolutional kernel enables the time-series prediction network to extract spatial features. However, the perceptual field of the convolutional kernel is limited and cannot globally represent spatial features [23]. Moreover, networks such as LSTM alleviate RNN gradient disappearance and explosion, but in practice, stacking multiple layers for accurate long time series data prediction is difficult. The application of self-attention is a good solution to these problems. Transformers were first proposed for solving natural language processing (NLP) problems, and they have since been widely used in the computer vision [24], [25], [26], [27], [28], remote sensing [29], [30], [31], and temporal prediction fields [32], [33], among others. Zhou et al. [32] proposed informers based on transformers, effectively replacing traditional self-attention with ProbSpare self-attention and greatly reducing the computational effort. Meanwhile, the generative style decoder layer can generate the output of long sequences in only one step, which avoids propagating errors and enables long time series information to be accurately predicted. Ma et al. [34] proposed a spatiotemporal dependent learning network that uses an attention mechanism to extract the spatial features of multivariate time series. Google proposed TimeSformer to classify videos by computing temporal and spatial self-attention [35]. Self-attention networks have strong abilities to represent global temporal and spatial features and can capture global spatial and temporal representations. However, these networks are weak when capturing local spatial features, which makes handling the complex spatiotemporal variations in ocean weather prediction difficult [23]. Graph networks have also been studied in marine weather prediction. Graph networks build spatiotemporal models by constructing relational architectures from a graph perspective based on a priori knowledge to enable spatial relationships between points to be extracted [36], [37]. However, these graph models rely heavily on predefined graph structures to extract spatiotemporal features, limiting their applications.

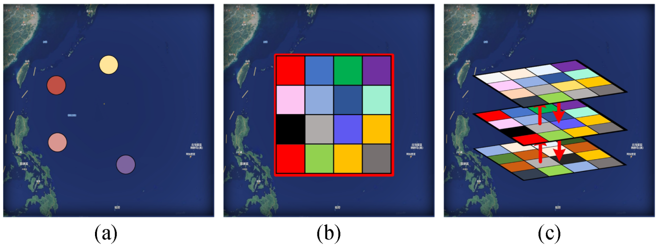

Fig. 1. Ocean weather forecasting methods. (a) Multipoint prediction method. (b) Unitary field prediction method. (c) Multifield prediction method.

As shown in Fig. 1, most existing marine weather forecasting methods are single-point or multipoint forecasts [17], [18], [19], [32], [36] and univariate regional forecasts [20], [21], [22], [38], [39], [40]. Single-point or multipoint forecasting methods build time-series models based on data from a certain point or several points, which makes comprehensively considering spatial correlation difficult, resulting in average forecasting effects. Regional prediction refers to the spatiotemporal modeling of a certain area. This prediction model can capture the temporal and spatial correlations of the area, but it is currently primarily a univariate model. For ocean weather prediction, univariate prediction is always one-sided. On the one hand, the multiple elements of ocean weather are interrelated and affect each other, and using effective means for exploring the correlations of multiple elements can improve the prediction efficiency of each element. Taking the formation of sea wind as an example, the uneven heating of the sea surface leads to the upward and downward movement of the air, and this vertical movement of the air changes the air pressure on the same horizontal plane, which generates the horizontal air pressure gradient force that prompts the atmosphere to horizontally flow from high-pressure areas to low-pressure areas. This movement leads to the formation of sea wind. The formation of sea wind is most directly related to air pressure and temperature. On the other hand, considering the hardware and facility conditions of practical applications, prediction models designed for a single ocean variable cannot meet the requirements for large ocean environment construction. However, the difficulty of simultaneously training and learning multiple variables is much greater than that of using univariate networks. Avoiding the negative impact of different scale losses on each task and balancing the training process of each variable to finally achieve multivariate cooptimal regression are difficult problems in the multitask domain [41], [42], [43], [44]. For multitask training, Chennupati et al. [45] proposed geometric losses to ensure losses were balanced. Chen et al. [46] proposed the grad-norm for adjusting the training gradient of the weight factor by the rate of change of each task loss to balance training among multiple tasks. To balance the learning training process of the ocean multivariate spatiotemporal field, we introduced the uncertainty correlation loss function into the ocean multivariate weather prediction domain to capture the optimal points through dynamically adjusting the loss weights of each variable. Notably, a multitasking mode, i.e., multiple inputs and multiple outputs, is adopted for marine weather data generation, and simultaneously predicting multiple variables can greatly reduce the required computational resources, which is of great significance for practical applications of data-driven marine weather prediction-based methods.

Therefore, multivariate ocean weather prediction has three main challenges: 1) exploring the comprehensive spatial linkages in the spatiotemporal field; 2) extracting the time dependence in the spatiotemporal fields; and 3) capturing the dynamic relevance among multivariate variables and ensuring multivariate prediction accuracy. Accordingly, we designed the spatiotemporal hybrid convolutional attention (StHCA) module. In this module, the spatial restoration and fusion layer fuses global and local spatial features to address the first problem, and the temporal self-attention mechanism extracts the temporal dependencies of spatiotemporal data to address the second problem. We solved the third problem by multivariate interacting and uncertainty loss functions. Our contributions are summarized as follows.

1) We proposed a novel multivariate spatiotemporal data generation model, StHCFormer, which achieves multivariate weather prediction for the ocean.
2) Fusing the advantages of self-attention and convolution, we proposed a StHCA module to mine the global representation and local features of space, obtain time-dependent relationships, and capture the dynamic correlation of multiple variables.
3) We introduced the homoscedasticity uncertainty loss function in the multitask domain into the marine multivariate weather prediction domain to balance the training process of multivariate tasks using the Gaussian distribution assumption of data and uncertainty correlation.
4) We compared StHCFormer with current state-of-the-art forecasting methods on publicly available datasets and demonstrated the effectiveness of StHCFormer. This shows that StHCFormer can effectively link spatiotemporal correlations and multivariate variables to provide more efficient ocean weather forecasts.

II. Methodology

A. Problem Definition

The sea surface is typically divided into grids based on spatial information (longitude and latitude). Each grid is an observation point for multivariate data, and each cycle contains a set of multivariate values. All grid areas form a $C \times H \times W$ matrix D_t, representing a set of weather values at a specific time t, where W and H correspond to the number of grid areas along the latitude and longitude, respectively, and C represents the number of variables. When three variables (i, j, k) are included, all matrices in the historical data records of an ocean region form a time series $D_1^{(i,j,k)}, D_2^{(i,j,k)}, \ldots, D_t^{(i,j,k)}, D_t^{(i,j,k)} \in R^{T \times 3 \times H \times W}$.

In practice, we usually try to learn from a historical period of weather records and use what we learn to predict future weather conditions. We follow this principle in the ocean weather prediction work we conduct. When predicting ocean weather based on meteorological knowledge, we default to 6 hours or less when the weather does not change dramatically, so our data point interval is chosen to be 6 hours. There are thus four time points in each day: 0:00, 6:00, 12:00, and 18:00. For a set area, given u multivariate history records $\{D_1^{(i,j,k)}, D_2^{(i,j,k)}, \ldots, D_u^{(i,j,k)}\}$, the future $v_{(i,j,k)}$ weather data are inferred from the history $u_{(i,j,k)}$, and the equation is expressed as

$$D_{u+1}^{(i,j,k)}, \ldots, D_{u+v}^{(i,j,k)} = \mathcal{F}\left(D_1^{(i,j,k)}, D_2^{(i,j,k)}, \ldots, D_u^{(i,j,k)}\right) \tag{1}$$

where $\mathcal{F}$ denotes the prediction network. For example, according to the previously set 6-hour fetching law, $u = 28$ and $v = 12$ indicate the prediction of future 3-day (i, j, k) records based on historical 7-day historical (i, j, k) records.

B. Model Structure

Fig. 2 shows the architecture of StHCFormer. The StHCFormer is designed with an encoder-decoder framework, and its architecture contains four parts: input, output, encoder, and decoder. The white area in Fig. 2 represents the network input, which contains the unfolding layer, the embedding layer, the location encoding module and the teacher forcing module, which is used to process the historical and target weather records. The green area in Fig. 2 represents the network encoder, which consists of multiple encoding blocks, each of which has two inputs and outputs. The first input is the token after encoding, and the second input is the convolutional feature layer after the initial convolution. The output of the encoder is the self-attention branch output and the convolutional branch output. Each encoding block contains two internal subconnected layers. The first sublayer is the encoder spatiotemporal hybrid convolutional attention module surrounded by the green dashed line, which contains a multiheaded attention layer, a local extractor and interacting layer, a spatial restoration and fusion layer, a normalization layer and a residual connection. The second sublayer connection contains a feed-forward fully connected layer, a normalization layer and a residual connection. The blue area represents the network decoder, which consists of a stack of multiple decoding blocks. Similar to the encoder, the decoder has two inputs and outputs, and its distribution is basically the same as that of the encoder. Each decoding block consists of three subconnected layers. The first and second sublayers are both decoder spatiotemporal hybrid convolutional attention modules surrounded by the blue dashed line. Both include a multiheaded self-attention layer, a local extractor and interacting layer, a

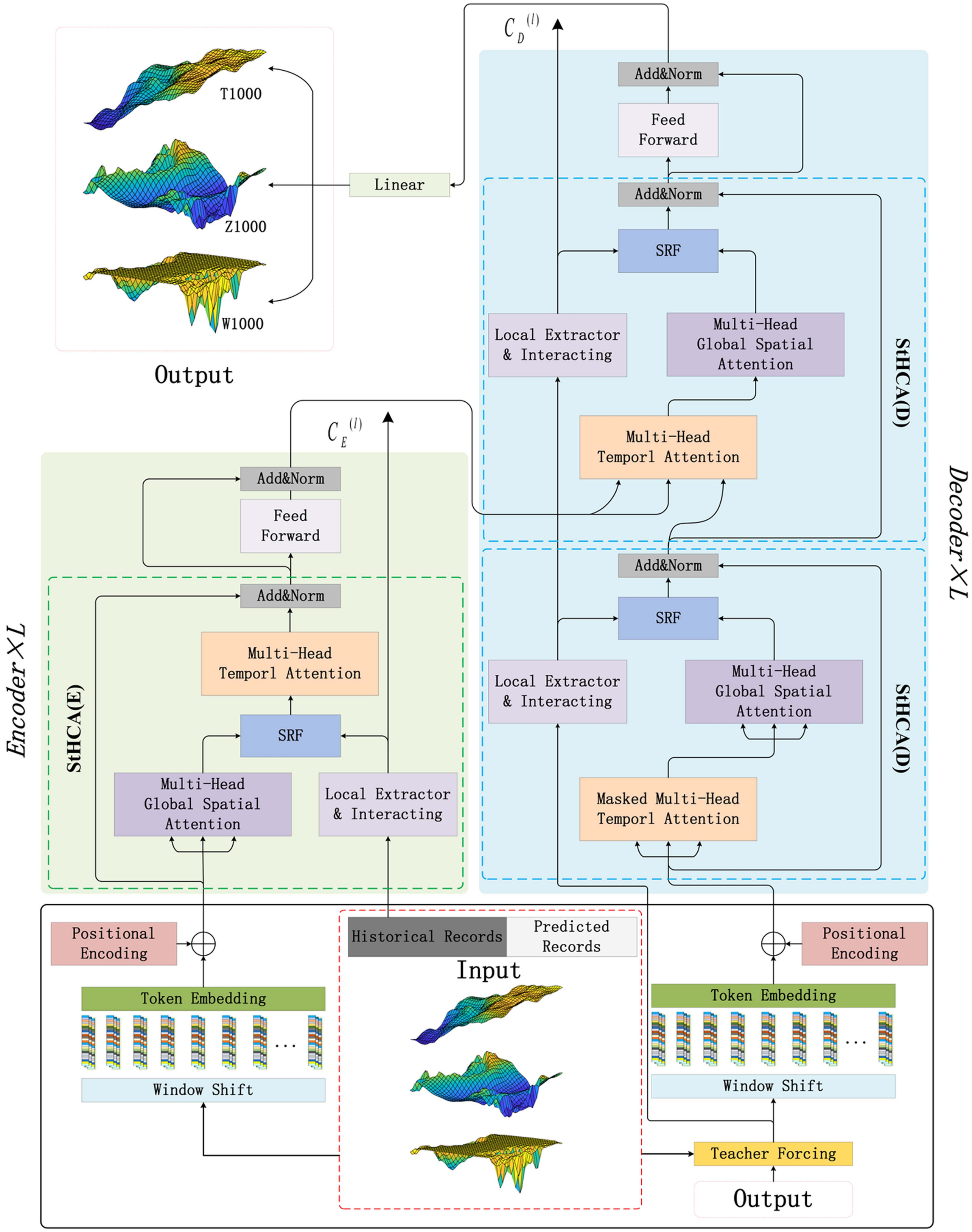

Fig. 2. Architecture of StHCFormer. The StHCFormer model contains four parts: input, output, encoder and decoder. The variables of the input historical records correspond to the data frames from day $t-7$ to day t. The records from day $t+1$ to day $t+n$ are used as predicted variables for the output. Each data frame contains three channels corresponding to the geopotential, temperature, and wind.

spatial restoration and fusion layer, a normalization layer and a residual connection. The third sublayer connection structure includes a feed-forward fully connected layer, a normalization layer and a residual connection.

The core of the StHCFormer network is the StHCA module, which includes a self-attention branch and a convolutional branch. The StHCA module in the encoder is slightly different from that in the decoder; the StHCA module in the encoder performs a multiheaded spatial attention computation followed by a temporal attention computation, whereas the StHCA module in the decoder performs a multiheaded temporal attention computation followed by a multiheaded spatial attention computation. The convolutional branch is used to extract local features of the grid information at each time step and capture dynamic correlations among multiple variables. Spatial attention is used to learn the global spatial representation of the grid points based on the elemental attention results at each time step. Temporal attention models temporal correlations at all time steps. The modules in the attention branch are bridged using a split-attention mechanism, which can significantly reduce the computational effort. StHCFormer accounts for the spatiotemporal features in comprehensive space and continuous time, as well as the heterogeneous information between multiple inputs, and can more accurately model the spatiotemporal evolution information of multiple field variables than traditional CNN models and RNN models.

1) Input Embedding: We use a sliding window to sample from the dataset and obtain input data frames of size $m \times H \times W \times T$, where m denotes the number of multivariate features, H and W denote the height and width of the input features, respectively, and T denotes the time length of the input features. In the convolution branch, the input data are $c_t \in R^{m \times H \times W}$, where $t = 1, 2, ..., T$. For the input data c_t containing T frames of data, we perform the initialized convolution operation on the input data separately according to the time index

$$c_t^{(0)} = SW\left(\left[g_{\text{ini}}(c_{t'})_{t'=1,2,\ldots,T}\right]\right) \tag{2}$$

where SW denotes the Swish activation function and g_{ini} denotes the initialized convolution calculation, which contains a set of $1 * 1$ convolution and $3 * 3$ convolution.

In the self-attention branch, the data are first processed by sliding window segmentation in the H and W dimensions, and divided to obtain N nonoverlapping patches of size h and w. $N = \frac{H \times W}{h \times w}$ contains the spatial domain information of all input data frames. The N patches are unfolded to obtain the vector $x_{(p,t)} \in R^{m \times h \times w}$, where $p = 1, 2, \ldots, N$ denotes the spatial location index of the divided patches and $t = 1, 2, \ldots, T$ denotes the data frame index. Then, we implemented a linear transformation using a learnable E matrix to map $x_{(p,t)} \in R^{m \times h \times w}$ to an embedding vector $z_{(p,t)}^{(0)} \in R^D$, which can be expressed by the following:

$$z_{(p,t)}^{(0)} = Ex_{(p,t)} + e_{(p,t)}^{\text{pos}} \tag{3}$$

where $e_{(p,t)}^{\text{pos}}$ denotes learnable location embedding, which is used to encode the spatiotemporal location for dividing patches. Above, the inputs are obtained as $z_{(p,t)}^{(0)} \in R^D$ and $c_t^{(0)} \in R^{m \times H \times W}$, which denote the inputs of the attention branch and the convolution branch, respectively. They are applied to both the encoder and the decoder.

2) Spatiotemporal Hybrid Convolutional Attention Model: Self-attention is strong for acquiring global spatial representations but weak for capturing local features. However, for ocean weather prediction, local feature interactions also play important roles in the dynamic evolution of variables, so exploring the temporal and global-local spatial feature linkages is especially important for ocean multivariate weather prediction. Therefore, we proposed a StHCA module to capture the historical dependence of variables in the temporal domain by temporal self-attention, obtain global spatial representations by spatial self-attention, and capture local features and multivariate dynamical linkages by convolutional networks to more accurately predict maritime weather. The network modules are connected in a split-attention manner, which greatly reduces the number of parameters and operations, making multivariate predictions possible.

Our model contains $\mathcal{L}$ encoder blocks. Each block has two inputs and outputs: the inputs are $z_{(p,t)}^{(\ell-1)}$ and $c_t^{(\ell-1)}$, and the outputs are $z_{(p,t)}^{(\ell)}$ and $c_t^{(\ell)}$. For the self-attention branch of the current coding block, the $q/k/v$ vector is obtained from the output of the previous coding block by the following:

$$\mathbf{q}_{(p,t)}^{(\ell,a)} = W_Q^{(\ell,a)} \text{LN}\left(z_{(p,t)}^{(\ell-1)}\right) \in R^{D_h} \tag{4}$$

$$\mathbf{k}_{(p,t)}^{(\ell,a)} = W_K^{(\ell,a)} \text{LN}\left(z_{(p,t)}^{(\ell-1)}\right) \in R^{D_h} \tag{5}$$

$$\mathbf{v}_{(p,t)}^{(\ell,a)} = W_V^{(\ell,a)} \text{LN}\left(z_{(p,t)}^{(\ell-1)}\right) \in R^{D_h} \tag{6}$$

where $LN()$ denotes LayerNorm, $a = 1, ..., \mathcal{A}$ is an index over multiple attention heads and $\mathcal{A}$ denotes the total number of attention heads. The latent dimensionality of each attention head is set to $D_h = D/\mathcal{A}$. The self-attention weights are calculated by DOT_PRODUCT and the global spatial self-attention weights $\alpha_{(p,t)}^{(\ell,a)space}$ are obtained by the following:

$$\alpha_{(p,t)}^{(\ell,a)space} = \sigma\left(\frac{\mathbf{q}_{(p,t)}^{(\ell,a)^{\top}}}{\sqrt{D_h}} \cdot \left[\left\{\mathbf{k}_{(p',t)}^{(\ell,a)}\right\}_{p'=1,\ldots,N}\right]\right) \tag{7}$$

where σ denotes the Softmax activation function. Multiplying the spatial self-attention weights with the corresponding v-vectors yields the single-headed attention output as

$$\mathbf{s}_{(p,t)}^{(\ell,a)space} = \sum_{p'=1}^{N} \alpha_{(p,t),(p',t)}^{(\ell,a)} \mathbf{v}_{(p',t)}^{(\ell,a)}. \tag{8}$$

By splicing the multiheaded attention, we can obtain the following spatial self-attention matrix by MLP and layer normalization:

$$\mathbf{z}_{(p,t)}^{(\ell)} = W_O \begin{bmatrix} \mathbf{s}_{(p,t)}^{(\ell,1)} \\ \vdots \\ \mathbf{s}_{(p,t)}^{(\ell,\mathcal{A})} \end{bmatrix}. \tag{9}$$

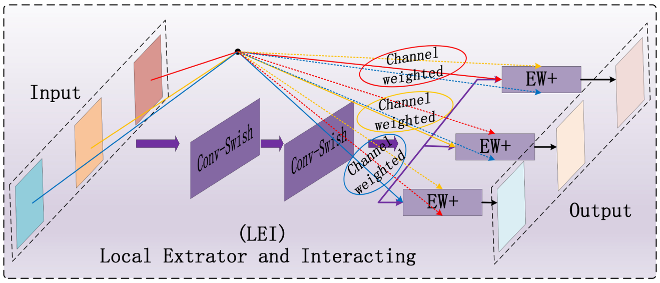

Fig. 3. Structure of the local extractor and interacting layer. $EW+$ denotes the elementwise summing operation.

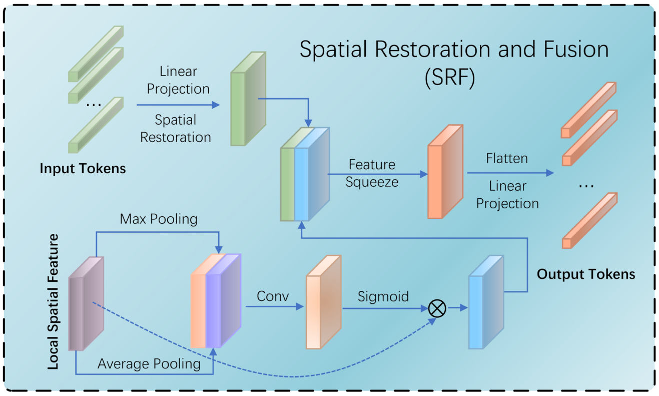

Fig. 4. Structure of the spatial restoration and fusion layer.

Fig. 3 shows the local extractor and interacting(LEI) structure, which is represented by the mauve structure in Fig. 2. The LEI consists of a 1 × 1 convolution layer, a 3 × 3 convolution layer, swish activation functions and a weighting residual structure. The LEI structure has two main roles. One is to extract local features, and the other is to achieve full communication of multivariate information and capture dynamic correlations among multivariates through convolutional channel transformation and channel weighting. The local spatial feature extractor can be expressed as

$$c^{(l)} = SW\left(g\left(c_t^{(\ell-1)}\right)\right) + CWc_t^{(\ell-1)} \tag{10}$$

where SW denotes the Swish activation function and g denotes the convolutional calculation process, and CW denotes the weights of different channels.

Fig. 4 shows the spatial restoration and fusion (SRF) structure, which is represented by the dark blue structure in Fig. 2. The SRF structure includes two inputs, which are the patch tokens of global attention and local features output by the LEI structure. Spatial feature restoration is first performed on the patch tokens to obtain a global spatial feature representation with scale $C \times T \times H \times W$ through reshape and fold operations

$$\hat{\mathbf{z}}_t^{(\ell)} = SR\left[\mathbf{z}_{(p,t)}^{(\ell)}\right] \tag{11}$$

where SR denotes the spatial restoration process. The maximum pooling and average pooling operations are performed on the local features output from the LEI structure, and the focus on the focal region is then obtained by convolving f_c

$$\hat{c}_t^{(\ell)} = SM\Big(f_c\Big(\Big[Avgpool\left(c_t^{(\ell)}\right); Maxpool\left(c_t^{(\ell)}\right)\Big]\Big)\Big) \odot c_t^{(\ell)} \tag{12}$$

where $Avgpool$ and $Maxpool$ stand for average pooling and maximum pooling, respectively. f_c denotes spatial attention convolution, whose convolution kernel is typically 5 or 7. SM denotes sigmoid activation function. $\hat{\mathbf{z}}_t^{(\ell)}$ and $\hat{c}_t^{(\ell)}$ are stitched, and a dimensionality reduction operation by 1 × 1 convolution to obtain the final spatial feature representation. Finally, it is repartitioned into patch token for temporal attention calculation

$$\phi_{(p,t)}^{(\ell,a)} = SW\left(f_\phi\left[\hat{c}_t^{(\ell)}; \mathbf{z}_t^{(\ell)}\right]\right) \tag{13}$$

where SW denotes the Swish activation function and f_ϕ denotes the reduced-dimensional convolution calculation process. Next, $\phi_{(p,t)}^{(l,a)}$ is fed into the temporal attention layer to obtain the $q/k/v$ matrix of temporal self-attention

$$\mathbf{q}_{(p,t)}^{(\ell,a)} = W_Q^{(\ell,a)}\mathrm{LN}\left(\phi_{(p,t)}^{(\ell,a)}\right) \in R^{D_h} \tag{14}$$

$$\mathbf{k}_{(p,t)}^{(\ell,a)} = W_K^{(\ell,a)}\mathrm{LN}\left(\phi_{(p,t)}^{(\ell,a)}\right) \in R^{D_h} \tag{15}$$

$$\mathbf{v}_{(p,t)}^{(\ell,a)} = W_V^{(\ell,a)}\mathrm{LN}\left(\phi_{(p,t)}^{(\ell,a)}\right) \in R^{D_h}. \tag{16}$$

Thus, the temporal self-attention weights are obtained as

$$\alpha_{(p,t)}^{(\ell,a)time} = SM\left(\frac{\mathbf{q}_{(p,t)}^{(\ell,a)^\top}}{\sqrt{D_h}} \cdot \left[\left\{\mathbf{k}_{(p,t')}^{(\ell,a)}\right\}_{t'=1,\ldots,T}\right]\right) \tag{17}$$

$$\mathbf{s}_{(p,t)}^{(\ell,a)time} = \sum_{t'=1}^{F} \alpha_{(p,t),(p,t')}^{(\ell,a)} \mathbf{v}_{(p,t')}^{(\ell,a)}. \tag{18}$$

The multiheaded temporal self-attention output is integrated and a residual connection is formed with the input of the self-attention branch to obtain the final output of the StHCA module

$$\phi'^{(\ell)}_{(p,t)} = W_O \begin{bmatrix} \mathbf{s}_{(p,t)}^{(\ell,1)} \\ \vdots \\ \mathbf{s}_{(p,t)}^{(\ell,\mathcal{A})} \end{bmatrix} + z_{(p,t)}^{(\ell-1)}. \tag{19}$$

The self-attention branch of the StHCA module is separately connected by computing the spatiotemporal self-attention modules separately and then connecting them in sequence; as a result, the input of the temporal attention module is the output of the refined spatial attention module. This connection reduces the computational complexity of the model from $O((FN)^2 d_{\text{model}})$ to $O((F^2 + N^2) d_{\text{model}})$. Notably, practical applications of transformer models are limited by the expensive computational resources of these models, which grow exponentially for multivariate model computations. Separating the self-attention connections greatly reduces the computational effort of the self-attention model, which allows us to investigate multivariate models using the self-attention model. Finally, the resulting vector is passed through the MLP to obtain the final encoding result $z_{(p,t)}^{(\ell)}$ for the l-encoding block. This result is

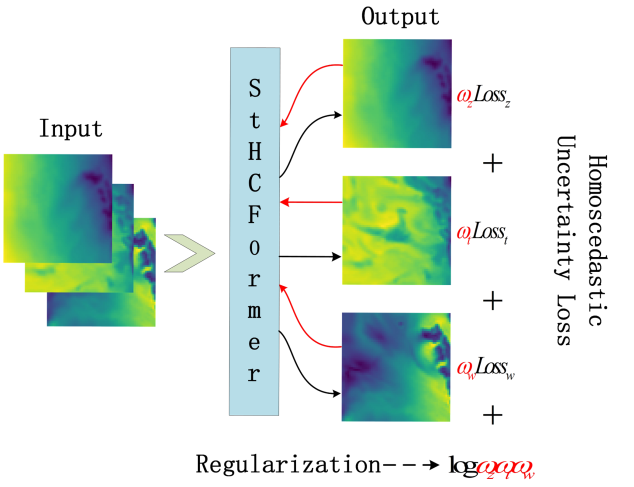

Fig. 5. Weighted and working schematic of homoscedastic uncertainty loss.

calculated by the following:

$$z_{(p,t)}^{(\ell)} = MLP\left(\mathrm{LN}\left(\phi_{(p,t)}^{\prime(\ell)}\right)\right) + \phi_{(p,t)}^{\prime(\ell)}. \tag{20}$$

3) Decoding: Like the encoding block, each decoding block has two inputs and outputs, which are the convolutional branch and the self-attention branch. However, each decoding block contains StHCA module parts. The StHCA module in the decoding block performs temporal self-attention calculation before spatial feature calculation. In the training phase, at the end of encoding, the last frame of the historical record is fed to the convolution branch. At the same time, this frame is divided into patches of size $h \times w$, after which it is transformed into tokens and position encoded, and finally fed into the spatial self-attention branch. After that, the decoder performs a cyclic decoding process based on the input tokens and the k/v vectors provided by the encoder until all the data frames to be predicted are obtained. As in the input section in Fig. 2, the output indicates the obtained data frames, and the output is input to the decoder to obtain the meteorological data frame at the next time point. Using teacher forcing in the training process can accelerate the model convergence. Specifically, the output generated by the model is replaced with the ground truth at time t in the training dataset as the input of the next time step. The output of the decoder part is a token representation of the ocean multivariate climate data at the future moment. Using a linear layer to transform the token, followed by data space restoration, finally yields the current prediction frame data of size $m \times H \times W$ is finally obtained. Afterward, the output of the model is used as input for a new test phase or with teacher forcing as input for the training phase.

C. Homoscedasticity Uncertainty Loss

To balance the training loss of multivariate variables, we introduced the homoscedasticity uncertainty loss function from the multitask learning domain into the marine multivariate climate prediction domain. The weighting process for multitask losses is shown in Fig. 5. In the multitask learning domain, we usually assign different weights to different tasks to balance the losses of multiple tasks. For example, the loss of the task in this article can be expressed by the following:

$$Loss = \omega_z Loss_z + \omega_t Loss_t + \omega_w Loss_w \tag{21}$$

where ω_z, ω_t, and ω_w denote the loss weights for the geopotential, temperature, and wind speed prediction tasks, and $Loss_z$, $Loss_t$, and $Loss_w$ denote the loss for the geopotential, temperature, and wind speed prediction tasks, respectively. However, constant weights require extensive experimental validation, and constant weights constrain the convergence direction of the model, which in turn affects the final performance of the model.

In multitask joint learning, task-dependent uncertainty can represent the relative difficulty between different tasks. The covariance uncertainty is independent of the input and dependent on the inherent uncertainty of the task. By transforming the homoscedastic uncertainty into the loss weight, the model can dynamically have the ability to adjust the loss. In this article, we assume that the ocean climate data follow a Gaussian distribution and derive the loss function based on homoscedasticity uncertainty. Assume that $f^w(x)$ denotes the output of the neural network when the weight is w and the input is x. For the regression task, the Gaussian likelihood is estimated as the following(σ is the observed noise scalar):

$$p\left(y \mid f^w x\right) = \mathcal{N}\left(f^w(x), \sigma^2\right). \tag{22}$$

In multitasking, after the maximum likelihood is decomposed into independent factors and $f^w(x)$ is defined as a sufficient statistic, we obtain the multitasking likelihood as

$$p\left(y_1, \ldots, y_k \mid f^w(x)\right) = p\left(y_1 \mid f^w(x) \ldots p\left(y_k \mid f^w(x)\right)\right. \tag{23}$$

where $y_1, \ldots, y_k$ are the outputs of the model task. Suppose the model has two outputs with y_1 and y_2, whose outputs follow the Gaussian distribution as follows:

$$\begin{aligned} &p\left(y_1, y_2 \mid f^w(x)\right) \\ &\quad = p\left(y_1 \mid f^w(x)\right) \cdot p\left(y_2 \mid f^w(x)\right) \\ &\quad = \mathcal{N}\left(y_1; f^w(x), \sigma_1^2\right) \cdot \mathcal{N}\left(y_2; f^w(x), \sigma_2^2\right). \end{aligned} \tag{24}$$

Taking the log-likelihood of the above function, the loss function can be expressed as

$$\begin{aligned} &= -\log p\left(y_1, y_2 \mid f^w(x)\right) \\ &\propto \frac{1}{2\sigma_1^2}\left\|y_1 - f^w(x)\right\|^2 + \frac{1}{2\sigma_2^2}\left\|y_2 - f^w(x)\right\|^2 + \log \sigma_1 \sigma_2 \\ &= \frac{1}{2\sigma_1^2}\mathcal{L}_1(w) + \frac{1}{2\sigma_2^2}\mathcal{L}_2(w) + \log \sigma_1 \sigma_2 \end{aligned} \tag{25}$$

where $\mathcal{L}_k(w) = \|y_k - f^w(x)\|^2$ denotes the loss function of the kth task. Therefore, the loss function in this article can be expressed as

$$\begin{aligned} &\mathcal{L}\left(w, \sigma_1, \sigma_2, \sigma_3\right) \\ &= \frac{1}{2\sigma_1^2}\mathcal{L}_1(w) + \frac{1}{2\sigma_2^2}\mathcal{L}_2(w) + \frac{1}{2\sigma_3^2}\mathcal{L}_3(w) + \log \sigma_1 \sigma_2 \sigma_3. \end{aligned} \tag{26}$$

When minimizing the objective function, σ_1 and σ_2 can be regarded as the relative weights of $L_1(w)$ and $L_2(w)$, respectively, and they can be adaptively adjusted according to the data. If the noise parameter σ_1 of variable y_1 increases, the weight of $L_1(w)$ decreases. Correspondingly, the corresponding loss function weights increase if the noise decreases. The last term is used as a regularization term to suppress excessive noise increases.

III. Experiment

All experiments ran on a Linux system with an Intel(R) Xeon(R) CPU E5-2680 v4 and Nvidia RTX 3090 GPU. The programming language was Python, and the training framework was PyTorch.

A. *Dataset*

In this article, we chose the ERA5 reanalysis dataset for experimental validation, which is available at https://cds.climate.copernicus.eu. For three-dimensional fields, ERA5 has 13 vertical horizontal scales: 50, 100, 150, 200, 250, 300, 400, 500, 600, 700, 850, 925, and 1000 hectopascals. Pressures in hectopascals are typically used as vertical coordinates rather than physical heights. This has practical advantages, such as reducing the number of required state variables and simplifying mass conservation [47]. In the processed dataset, 1000 hPa is approximately the pressure at sea level, 850 hPa is approximately the pressure at 1.5 km, and 500 hPa is approximately the pressure at 5.5 km. In this article, we select three variables, Z1000, T100, and W1000, for the multivariate weather prediction experiments at sea. Z1000 denotes the geopotential at a height of 1000 hPa, which is a common variable used to encode the pressure distribution at synoptic scales and is a continuous field variable. This variable is strongly correlated with air pressure, with larger values corresponding to higher air pressure. T1000 is the temperature at a height of 1000 hPa and approximates the surface temperature of the ocean. This directly reflects ocean climate change and is the most widely studied variable in the ocean weather forecasting field. It is also a continuous field variable. W1000 represents the speed of the sea wind at 1000 hPa altitude. The size of the wind speed directly affects the formation of waves at the sea surface, which is a more complex and variable continuous field variable. In most physical NWPs and climate models, geopotential, temperature, and wind are predicted state variables. Ideally, differences in temperature lead to different air pressure distributions, resulting in horizontal barotropic gradient forces that drive the atmosphere to move from high-pressure to low-pressure regions, forming sea wind. Therefore, the three variables we have chosen are closely related, and we hope to capture their intrinsic operation and development patterns through our modeling design for accurate weather prediction.

As shown in Fig. 6, we selected a portion of the Philippine Sea as the test area. This is one of the busiest offshore production and route areas in the world, and we hope that our model can accurately predict the SST, geopotential, and sea wind speed in this area to aid in offshore production activities. Our experimental region of interest has a latitude and longitude range of [9°45′N – 25°30′N, 128°E – 143°E]. The data are divided by 0.25° to obtain a grid of 60*60 points. The experimental time interval is chosen as 1980–2022, encompassing approximately 60 000 pieces of trivariate historical data. We selected 40 years of data from 1980–2020 as the training and validation set, and two years of data from 2021–2022 as the test set. The dataset is processed using a sliding window with a sliding interval of 1 day, i.e., 4 time points, corresponding to $s = 4$ in Fig. 7. We set up the model to predict weather data for the next 3 days (12 time points) and 5 days (20 time points) through 7 days of historical data (28 time points). For example, the first test time point of the test set after sliding window processing is 00 UTC on January 1, 2021, and the ending point is this time plus the historical time window, i.e., for the 5-day task, the starting point of the first test date is 00 UTC on January 8, 2020 and the ending point of the first test date is 23 UTC on January 12, 2020.

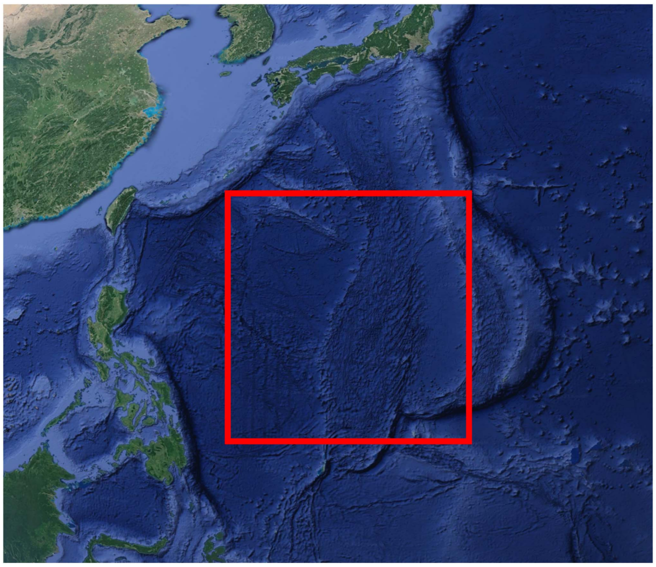

Fig. 6. Experimental area selection. The red box in the figure indicates the portion of the Philippine Sea selected for the experiments in this article.

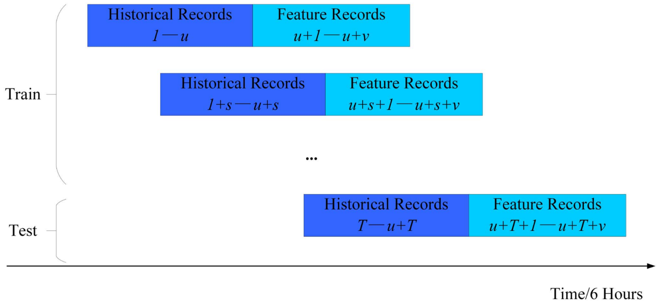

Fig. 7. Rules for dataset partitioning.

B. *Evaluation Metrics*

We chose to use the root-mean-square error (rmse), mean absolute error (MAE) and anomaly correlation coefficient (ACC)

TABLE I
RMSE of 3/5-Day Experiments With Different Algorithms

Model	3 Day			5 Day		
	Z1000 $[m^2s^{-2}]$	T1000[K]	W1000[m/s]	Z1000 $[m^2s^{-2}]$	T1000[K]	W1000[m/s]
CNN[47]	1046.50	6.54	4.45	1253.91	9.96	5.82
Conv-LSTM[20]	362.97	2.43	4.80	414.83	3.76	6.05
STGAN[33]	325.26	1.85	3.81	353.55	2.08	4.19
StHCFormer(single/5-day)	325.26	1.74	4.00	353.55	**1.91**	4.30
StHCFormer(5-day)	315.83	1.91	**3.58**	**348.83**	1.97	**3.77**
StHCFormer	**311.12**	**1.68**	3.73	**348.83**	1.97	**3.77**

The best model is highlighted in bold.

to evaluate the effectiveness of our algorithm. The rmse represents the standard deviation between the predicted and true values. It is a common loss function and evaluation metric in the data generation field. To reflect the position differences at different latitudes, we use the latitude-weighted rmse, which is calculated as follows:

$$\text{RMSE}=\frac{1}{N_{pre}}\sum_{i}^{N_{pre}}\sqrt{\frac{1}{N_{lat}N_{lon}}\sum_{j}^{N_{lat}}\sum_{k}^{N_{lon}}L(j)(f_{i,j,k}-t_{i,j,k})^2} \tag{27}$$

where f denotes the predicted result of our model and t denotes the truth value of the dataset. $L_{(j)}$ is the latitude weighting factor for the latitude at the jth latitude index

$$L(j)=\frac{\cos(lat(j))}{\frac{1}{N_{lat}}\sum_{j}^{N_{lat}}\cos(lat(j))}. \tag{28}$$

The MAE indicates the mean value of the absolute value of the error and is a commonly used error evaluation indicator. The MAE evaluates the degree of deviation between the predicted value and the true value, and the prediction error can be reflected by the positive effect of the absolute value. In this article, we use the latitude-weighted MAE, defined as follows:

$$\text{MAE}=\frac{1}{N_{\text{pre}}}\sum_{i}^{N_{\text{pre}}}\frac{1}{N_{\text{lat}}N_{\text{lon}}}\sum_{j}^{N_{\text{lat}}}\sum_{k}^{N_{\text{lon}}}L(j)\left\|f_{i,j,k}-t_{i,j,k}\right\|. \tag{29}$$

The anomaly correlation coefficient (ACC) is one of the most widely used measures for verifying spatial fields. It is the spatial correlation between forecast anomalies and verifying analysis anomalies relative to climatology. The ACC indicates how well the forecast anomalies represent the observed anomalies and shows how well the predicted values from a forecast model "fit" with the real-life data. The anomaly correlation coefficient can be defined as

$$\text{ACC}=\frac{\sum_{i,j,k}L(j)f'_{i,j,k}t'_{i,j,k}}{\sqrt{\sum_{i,j,k}L(j)f'^2_{i,j,k}\sum_{i,j,k}L(j)t'^2_{i,j,k}}}. \tag{30}$$

C. Experimental Results and Analysis

In our experiments, we used StHCFormer and other baseline models to generate the distributions of three continuous field variables, Z1000, T1000, and W1000, with lead times of 3 and 5 days, respectively, for some areas of the Philippine Sea. Based on the experimental results, we performed comprehensive performance comparison analysis, ablation experiment analysis, model complexity analysis, robustness analysis, and visualization analysis to demonstrate that our model performs better than other models under the above aspects.

1) Comprehensive Performance Comparison: Tables I and II compare the accuracy of the data generated by StHCFormer and multiple baseline models with lead times of 3 and 5 days. The rmse error values of the CNN predictions for Z1000 and T1000 with a three-day lead time are 1046.5 and 6.54, respectively, which are more than three times those of the StHCFormer model. CNN performs so poorly because it can extract only local spatial features and cannot explore temporal patterns. However, according to [47], marine weather does not exhibit large abrupt changes that occur in less than 6 hours, especially the geopotential and SST. Consequently, the distributions of weather elements in the previous frame and the next frame are tightly correlated, and the CNN model, which neglects temporal patterns, is not useful for weather prediction.

ConvLSTM performs better than CNN, with predicted rmses of 362.97, 2.43, and 4.80 for the three-day lead times Z1000, T1000, and W1000, respectively. ConvLSTM can capture spatiotemporal features, but due to the complex structure of LSTM itself and the gradient propagation defect of ConvLSTM, it cannot superimpose multiple layers to extract global spatial information and long-term dependencies, so it cannot achieve good prediction results in large spatiotemporal ranges. In addition, ConvLSTM cannot fuse and analyze multivariate relationships, so ConvLSTM does not perform as well as STDGN and StHCFormer. Fig. 8 also shows that no significant short-term prediction gap exists between the prediction effects of ConvLSTM and our method, whereas the long-term prediction gap gradually appears over time.

Both the STDGN and our model are able to capture longer temporal dependencies through the encoding-decoding structure, and both exhibit better long-term effect prediction performance. STDGN obtains the time-space features and multivariate coupling relationships through temporal attention, spatial attention and channel attention but neglects local features. Focusing on only global spatial features and neglecting local spatial features, especially the more drastic changing field variables, causes the prediction accuracy of STDGN to be lower than that of StHCFormer. StHCFormer outperforms all baseline models by extracting more comprehensive spatial features and capturing time-dependent relationships and multivariate channel coupling linkages through the StHCA module.

TABLE II
MAE OF 3/5-DAY EXPERIMENTS WITH DIFFERENT ALGORITHMS

Model	3 Day			5 Day		
	Z1000 $[m^2 s^{-2}]$	T1000[K]	W1000[m/s]	Z1000 $[m^2 s^{-2}]$	T1000[K]	W1000[m/s]
CNN[47]	1004.07	4.86	3.58	1216.20	7.47	4.72
ConvLSTM[20]	306.41	1.97	3.73	344.12	3.01	4.64
STDGN[33]	278.12	1.50	3.12	296.98	1.74	3.43
StHCFormer(single/5-day)	278.12	1.39	3.24	301.69	**1.56**	3.50
StHCFormer(5-day)	263.98	1.50	**2.89**	**292.27**	**1.56**	**3.08**
StHCFormer	**259.27**	**1.33**	3.01	**292.27**	**1.56**	**3.08**

The best model is highlighted in bold.

TABLE III
ACC OF 3/5-DAY EXPERIMENTS WITH DIFFERENT ALGORITHMS

Model	3 Day			5 Day		
	Z1000	T1000	W1000	Z1000	T1000	W1000
CNN[47]	0.41	0.59	0.12	0.24	0.45	0.06
ConvLSTM[20]	0.58	0.89	0.05	0.48	0.84	0.01
STDGN[33]	0.61	**0.92**	0.24	0.50	0.91	0.04
StHCFormer(single/5-day)	**0.68**	**0.92**	0.19	0.57	**0.92**	0.06
StHCFormer(5-day)	0.66	0.91	0.28	**0.59**	0.91	**0.16**
StHCFormer	0.66	**0.92**	**0.30**	**0.59**	0.91	**0.16**

The best model is highlighted in bold.

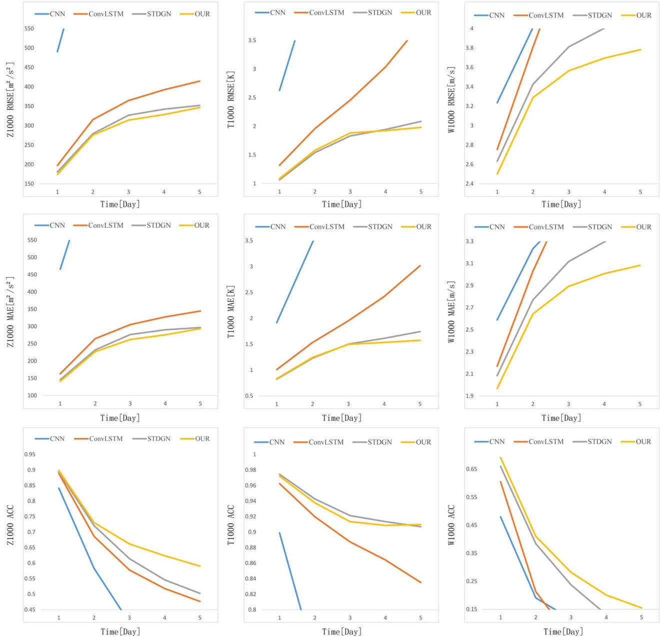

Fig. 8. Prediction of each algorithm for each day within a 5-day lead time. The three on the left are Z1000, the three in the middle are T1000, the three on the right are W1000, and the rows from top to bottom are rmse, MAE, and ACC.

General continuous field variables vary smoothly, so their prediction is relatively simple, and the rmse and MAE can adequately measure the prediction effect. However, the prediction of field variables with more dramatic fluctuations, such as wind fields with large fluctuations in a limited range, is more difficult. In addition, measuring the algorithm gap using rmse and MAE values is already difficult, so we instead used the ACC to further measure the robustness of the relevant algorithms for multivariate weather prediction fields. Table III presents the evaluation results of StHCFormer and other algorithms using the ACC as the metric. The rmse values of CNN and StHCFormer for W1000 are 4.45 and 3.73, and the MAE values are 3.58 and 3.01, respectively, when the lead time is 3 days. These values do not greatly differ, but the ACC values of CNN and StHCFormer are 0.12 and 0.3, respectively. This demonstrates that predicting a spatial-temporal field with such a sharp fluctuation as the wind field is truly difficult, and ACC metric can be used to better evaluate the predictive ability of the model. It also shows that our model achieves better wind field prediction results than other models. Comparing rows 4 and 5 of Tables I–III shows that integrating multivariate features, starting from the distribution of geopotential and temperature, can help the model better capture the evolution trend of the wind field. In terms of overall performance, StHCFormer outperforms all baseline models in marine multivariate weather prediction, and the results of model performance evaluation based on ACC and rmse/MAE are basically the same.

Fig. 8 shows the daily predictions of each model when the lead time is 5 days. The figure demonstrates that as the time increases, the prediction effect of each model gradually deteriorates because the temporal features of longer times are more difficult to capture than those of shorter times. CNN exhibits a much higher prediction error as the time increases because it does not capture the time dependence. ConvLSTM is able to compete with STDGN and StHCFormer in short-term prediction, but as the time increases, the prediction effect of ConvLSTM struggles to meet the demand due to ConvLSTM's own shortcomings. In the ACC graph of W1000 in particular, both CNN and ConvLSTM are out-of-control because predicting the wind field is extremely difficult in both temporal and spatial dimensions. Both our model and STDGN clearly perform better than CNN and ConvLSTM,

TABLE IV
ABLATION EXPERIMENTS

Model	Z1000 $[m^2s^{-2}]$			T1000[K]			W1000[m/s]		
	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
Encoder-Decoder	263.98	320.55	0.65	1.56	1.97	0.90	3.62	4.42	0.11
Hybrid ST	263.98	320.55	0.62	1.45	1.79	0.91	3.12	3.85	0.22
HUL	**259.27**	**311.12**	**0.66**	**1.33**	**1.68**	**0.92**	**3.01**	**3.73**	**0.30**

The best model is highlighted in bold.

which is consistent with the previous statement. Our model has an especially clear advantage over the other models in wind field prediction. This is because the wind field is more volatile than continuous spatiotemporal fields such as Z1000 and T1000, and obtaining sufficient spatiotemporal feature information from the wind field for accurate prediction as a single variable is difficult. However, StHCFormer achieves leading results in wind field prediction by separating the auxiliary components from Z1000 and T1000 through the built-in LEI module and then balancing the multivariate relationships through the HUL function.

2) Model Complexity Comparison: To verify that our model has low computational complexity, we designed experiments to illustrate the superiority of the model in terms of its number of parameters and weight size. Since the CNN we designed is simpler and includes base convolutional layers, it has a lower number of parameters, only approximately 2 M. However, even with a large model, improving the period prediction effect is difficult because the CNN lacks the ability to capture the time-scale features. ConvLSTM has a similar reason. Even if the depth of the model is adjusted and the number of network layers is increased, further improving the prediction effect is difficult due to the defects of the model itself. Self-attention-based models perform better than CNN and ConvLSTM because they can effectively capture spatial features and short and long temporal features. STDGN is a purely self-attention model with 18.84 M parameters and 26.7 M weights, while StHCFormer has 15.4 M parameters and 21.1 M weights. In terms of inference time, STDGN took 0.29 s and StHCFormer took 0.36 s when predicting weather conditions for the next 5 days. Our model has a slightly longer prediction time than STDGN because it uses a parallel strategy of self-attention and convolution for feature extraction and reconstruction, and the operation of its parallel structure adds additional runtime. However, our model outperforms STDGN both in terms of prediction accuracy and number of parameters. In some end devices, the restricted hardware facilities are very demanding on model design, whereas our multivariate prediction model can be explored for later practical applications.

3) Ablation Experiment: To visually demonstrate the effectiveness of our proposed method, we conducted ablation experiments for each module. This experiment was conducted based on the three-day lead time prediction. Table IV shows the ablation experiment results. The encoder–decoder model is the initial model, and we build the overall architecture of the encoder–decoder for spatiotemporal prediction based on the video classification model TimeSformer. The StHCA module is our proposed spatiotemporal hybrid convolutional attention module, and HUL is the homoscedastic uncertainty loss function we introduced. As shown in Table IV, after integrating the hybrid spatio-temporal attention mechanism, the rmse of temperature is reduced from 1.97 to 1.79 and the rmse of wind speed is reduced from 4.42 to 3.85, improving the prediction effect by approximately 9% and 13%, respectively, while the rmse of geopotential prediction remains unchanged. This indicates that our StHCA module can effectively capture the full range of spatial relationships while accounting for the coupling correlations among multiple variables to improve the prediction accuracy. Integrating HUL further improved the prediction accuracy of the three tasks by dynamically adjusting the task loss weights. This is because the losses of different variables have different numerical scales at different stages of model training. When simple summation is used as the loss treatment, the large numerical loss scales for certain channels suppress the impacts of small-scale loss. When HUL is used, HUL automatically weights the losses of each variable based on homoscedasticity uncertainty, thus unifying the losses of each variable in the same order of magnitude, ensuring losses with small gradients are not offset by losses with large gradients and improving the generalizability of the learned features.

4) Robustness Analysis: To demonstrate StHCFormer's robustness in multivariate prediction, we trained the model for each variable separately for testing, and the methods previously used for multivariate prediction, including the homoscedasticity uncertainty loss function and the multivariate fusion module, were of course eliminated. Rows 4 and 5 of Tables I–III demonstrate that our multivariate prediction model works better than the unitary prediction model in most cases. This indicates that our model extracts a positive influence relationship from the multivariate coupling, especially for the wind field, where the ACC improves by 47% and 167% for the 3-day and 5-day lead time cases, respectively. This also indicates that StHCFormer exhibits good robustness in the channel dimension and good generality for generating various multivariate spatiotemporal fields simultaneously.

The last two rows of Tables I–III show a comparison of the accuracies of the StHCFormer direct prediction results for three days and the direct prediction results for the third day at five days. The two methods produce essentially the same results when generating Z1000, T1000, and W1000 temporal and spatial field data at three-day lead times. The consistency of these three-day lead-time prediction results show that the proposed deep generative network has good robustness in the time dimension and can accurately predict short-term weather conditions while accurately generating long-term prediction results. This makes

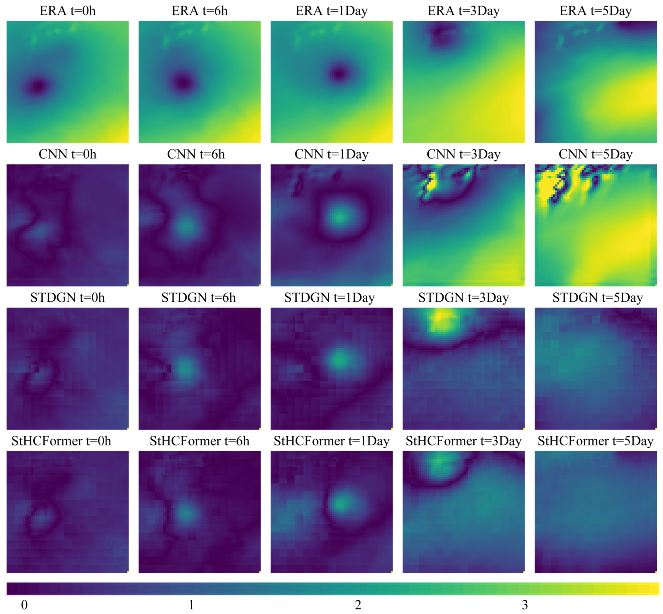

Fig. 9. Comparison of the results generated by different models for the Z1000 geopotential field. The top row shows the true values of the ERA data, including times $t = 0$ h, $t = 6$ h, $t = 1$ d, $t = 3$ d, and $t = 5$ d. The different time points can reflect the different performances of the model in the long and short terms. The subsequent three rows show the rmse loss of the prediction results of CNN, STDGN, and StHCFormer. Brighter colors indicate larger errors.

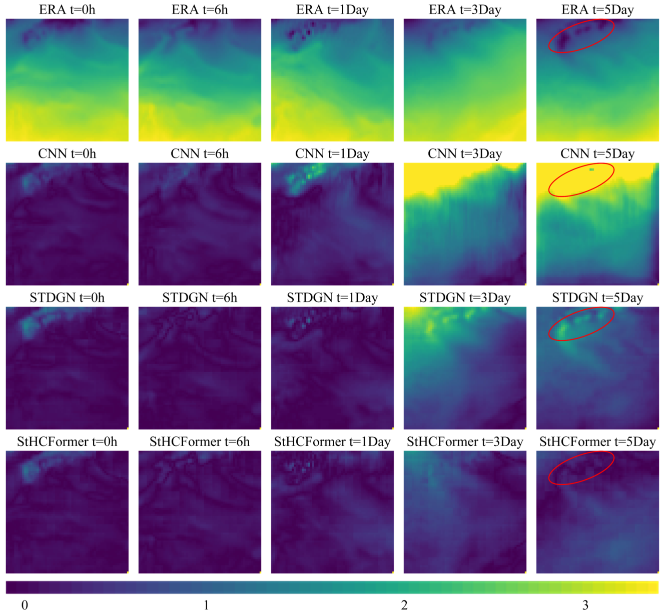

Fig. 10. Comparison of the results generated by different models for the T1000 temperature field. The top row shows the true values of the ERA data, including time $t = 0$ h, $t = 6$ h, $t = 1$ d, $t = 3$ d, and $t = 5$ d. The different time points can reflect the different performances of the model in the long and short terms. The subsequent three rows show the rmse loss of the prediction results of CNN, STDGN, and StHCFormer. In the rmse loss graph, a brighter color indicates a larger error. Red circles indicate some local hotspot locations.

obtaining short-, medium-, and long-term marine weather data in a single training session without training the model separately for each lead time possible.

The robustness of StHCFormer in the space-time and channel dimensions exceeds that of all baseline models. There are several possible reasons for this: first, StHCFormer considers more comprehensive information and is less susceptible to small-scale errors than the CNN. Second, in contrast to ConvLSTM, the StHCA module used in StHCFormer captures the long-range dependencies of the data in multiple dimensions, thus mitigating error accumulation during the generation process. The convolutional branch of StHCFormer enables more adequate local spatial features to be extracted than STDGN, forming a focused capture of local hotspots and their surrounding features. StHCFormer thus exhibits better robustness in the spatial domain.

5) Visualization Analysis: To further demonstrate the advantages of StHCFormer in marine multivariate climate prediction, we visualize and compare the prediction results of the CNN, STDGN, and StHCFormer algorithms on the Philippine Sea dataset. The images are divided into three parts, Z1000, T1000, and W1000. Each part contains 4 rows of images. The first row is the true value of ERA, and the subsequent three rows are the predicted rmse visualization results of CNN, STDGN, and StHCFormer. For each row, the corresponding schematics for $t = 0$ h, $t = 6$ h, $t = 1$ d, $t = 3$ d, and $t = 5$ d are depicted from left to right. Figs. 9–11 demonstrate that the loss of each algorithm increases with time, and CNN is the most unstable of the algorithms. This is because CNN is able to extract spatial features but cannot draw valid time-series information from historical data to help in prediction. Therefore, CNN performs fine in short-term prediction, but its long-term performance is poor. The STDGN and StHCFormer algorithms have better results in long-term prediction, which is consistent with the experimental results in the previous section. This is because StHCFormer's unique StHCA module allows it to better focus on local information and therefore better grasp salient features. As shown by the red circles in Figs. 10 and 11, StHCFormer predicts local hotspots well. Overall, our proposed model can better extract spatiotemporal features and thus more accurately predict ocean weather than other models.

IV. Discussion

The numerous trials stated above successfully highlight the benefits of our approach from a variety of angles. Traditional marine weather prediction methods require massive arithmetic support. Standard machine learning approaches are difficult to deal with massive data and fully utilize it to increase prediction accuracy. It is difficult for them to satisfy the expanding application demands. CNN cannot capture the temporal relationship based on deep learning methods, while ConvLSTM struggles with large time series when gradient propagation is used. As demonstrated in Figs. 8–11, the forecast accuracy of these two techniques drops rapidly as time passes. STDGN is one of the current SOTA algorithms in the field of multivariate ocean weather prediction, however the prediction impact is similarly poor due to the pure attention mechanism's lack of attention to local features. Our proposed model combines the strengths of convolution and self-attention: using self-attention to capture long-term temporal dependence and global spatial connection; and using convolution to capture local spatial hotspots. As shown

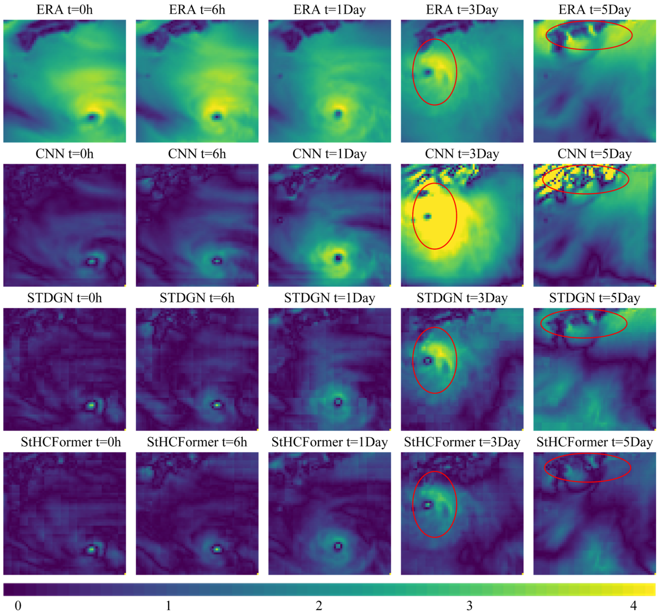

Fig. 11. Comparison of the results generated by different models for the W1000 wind field. The top row shows the true values of the ERA data, including time $t = 0$ h, $t = 6$ h, $t = 1$ d, $t = 3$ d, and $t = 5$ d. The different time points can reflect the different performances of the model in the long and short terms. The subsequent three rows show the rmse loss of the prediction results of CNN, STDGN, and StHCFormer. In the rmse loss graph, a brighter color indicates a larger error. Red circles indicate some local hotspot locations.

in Figs. 9–11, our model produces the best results, and it also produces more fantastic outcomes in specific local regions. In terms of algorithmic efficiency, and under the premise of ensuring prediction accuracy, our model has less parameters than STDGN. As shown in Tables I–III, our model is resilient in long and short time-series prediction, as well as multivariate prediction, thanks to the spatiotemporal hybrid convolutional attention module, LEI unit, and HUL loss. The ablation experiments also show the efficacy of our suggested strategy for multivariate marine weather prediction. Therefore, StHCFormer can better extract the spatiotemporal coupling aspects of marine weather, capture multivariate dynamic linkages, and improve multivariate marine weather forecast accuracy.

V. Conclusion

Accurate marine weather prediction is important for extreme weather warning, marine environmental protection and offshore production activities, while most of the existing data-driven models are multipoint prediction or univariate prediction methods, which have difficulty meeting the practical needs of marine climate prediction. In this study, from the perspective of multivariate space-time fields, we used ERA5 reanalysis data to forecast the weather in some sea areas of the Philippine Sea. Firstly, to predict multivariate ocean weather on a regional scale, we developed a prediction framework, called StHCFormer, based on space-time hybrid convolutional self-attention. In the convolution branch, we achieved multivariate information coupling by the LEI deflating feature channels and at the same time, enabled the network to acquire evolutionary features of local regions in space. In the self-attention branch, we used temporal self-attention to capture long-term dependencies and spatial self-attention to obtain global spatial feature representations. The SRF layer was then created in order to achieve an organic merging of the convolution branch and the self-attention branch, which is necessary to better capture the dynamic correlation of multivariate spatiotemporal data. Experiments demonstrate that our model can extract more appropriate spatio-temporal characteristics by space-time hybrid convolutional self-attention, which has a clear improvement effect on marine weather prediction. To balance the loss of multivariate networks in the training process, we introduced the homoscedasticity uncertainty loss function in the multitask training domain to realize the dynamic adjustment of multivariate loss weights to achieve the mutual positive effects of multivariate loss. These deliberate innovations allowed StHCFormer to capture more complete multivariate spatiotemporal variables and accomplish more accurate marine weather forecast. Experiments demonstrate that in 3-day and 5-day lead time forecasts, StHCFormer outperforms other classical methods.

Based on this research, we anticipate that StHCFormer will perform better in the real-world application of ocean climate prediction if we carry out further exploration in our follow-up work. These additional explorations should focus on avoiding the detrimental effects of cumulative errors and optimizing schemes for additional variables. For end-device applications, it is also critical to reduce the model's processing volume and speed up inference.

References

[1] T. N. Stockdale, M. A. Balmaseda, and A. Vidard, "Tropical atlantic SST prediction with coupled ocean–atmosphere GCMs," *J. Climate*, vol. 19, no. 23, pp. 6047–6061, 2006.

[2] J. Hurrell, M. Visbeck, and P. Pirani, "WCRP coupled model intercomparison project-phase 5-CMIP5," *Clivar Exchanges*, vol. 16, no. 56, pp. 1–52, 2011.

[3] B. B. Nardelli, C. Tronconi, A. Pisano, and R. Santoleri, "High and ultra-high resolution processing of satellite sea surface temperature data over southern European seas in the framework of myocean project," *Remote Sens. Environ.*, vol. 129, pp. 1–16, 2013.

[4] R. Noori, M. R. Abbasi, J. F. Adamowski, and M. Dehghani, "A simple mathematical model to predict sea surface temperature over the northwest Indian Ocean," *Estuarine, Coastal Shelf Sci.*, vol. 197, pp. 236–243, 2017.

[5] R. Allard et al., "The US navy coupled ocean-wave prediction system," *Oceanography*, vol. 27, no. 3, pp. 92–103, 2014.

[6] C. Li and J.-W. Hu, "A new ARIMA-based neuro-fuzzy approach and swarm intelligence for time series forecasting," *Eng. Appl. Artif. Intell.*, vol. 25, no. 2, pp. 295–308, 2012.

[7] J.-S. Kug, I.-S. Kang, J.-Y. Lee, and J.-G. Jhun, "A statistical approach to indian ocean sea surface temperature prediction using a dynamical ENSO prediction," *Geophysical Res. Lett.*, vol. 31, no. 9, pp. 399–420, 2004.

[8] I. D. Lins, M. Moura, M. Silva, E. L. Droguett, and C. M. C. Jacinto, "Sea surface temperature prediction via support vector machines combined with particle swarm optimization," *Proc. 10th Int. Probabilistic Saf. Assessment Manage. Conf.*, vol. 10, 2010.

[9] I. D. Lins, M. Araujo, M. das Chagas Moura, M. A. Silva, and E. L. Droguett, "Prediction of sea surface temperature in the tropical atlantic by support vector machines," *Comput. Statist. Data Anal.*, vol. 61, pp. 187–198, 2013.

[10] L. Wei, L. Guan, and L. Qu, "Prediction of sea surface temperature in the south China sea by artificial neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 4, pp. 558–562, Apr. 2020.

[11] A. Jirakittayakorn, T. Kormongkolkul, P. Vateekul, K. Jitkajornwanich, and S. Lawawirojwong, "Temporal KNN for short-term ocean current prediction based on HF radar observations," in *Proc. IEEE 14th Int. Joint Conf. Comput. Sci. Softw. Eng.*, 2017, pp. 1–6.

[12] A. Khosravi, R. Koury, L. Machado, and J. Pabon, "Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system," *Sustain. Energy Technol. Assessments*, vol. 25, pp. 146–160, 2018.

[13] Q. He et al., "Improved particle swarm optimization for sea surface temperature prediction," *Energies*, vol. 13, no. 6, 2020, Art. no. 1369.

[14] J. L. Elman, "Finding structure in time," *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, 1990.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.

[17] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, "Prediction of sea surface temperature using long short-term memory," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1745–1749, Oct. 2017.

[18] Y. Obara and R. Nakamura, "Transfer learning of long short-term memory analysis in significant wave height prediction off the coast of western tohoku, Japan," *Ocean Eng.*, vol. 266, 2022, Art. no. 113048.

[19] J. Xie, J. Zhang, J. Yu, and L. Xu, "An adaptive scale sea surface temperature predicting method based on deep learning with attention mechanism," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 5, pp. 740–744, May 2020.

[20] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[21] Z. Lin, M. Li, Z. Zheng, Y. Cheng, and C. Yuan, "Self-attention ConvLSTM for spatiotemporal prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 11531–11538.

[22] Y. Zhou, C. Lu, K. Chen, and X. Li, "Multilayer fusion recurrent neural network for sea surface height anomaly field prediction," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 4205111.

[23] Z. Peng et al., "Conformer: Local features coupling global representations for visual recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 367–376.

[24] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[25] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

[26] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[27] S. Zheng et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6881–6890.

[28] D. Jia et al., "Detrs with hybrid matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 19702–19712.

[29] D. Hong et al., "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5518615.

[30] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5607514.

[31] X. He, Y. Zhou, J. Zhao, D. Zhang, R. Yao, and Y. Xue, "Swin transformer embedding UNet for remote sensing image semantic segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4408715.

[32] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 11106–11115.

[33] J. Wang, L. Lin, S. Gao, and Z. Zhang, "Deep generation network for multivariate spatio-temporal data based on separated attention," *Inf. Sci.*, vol. 633, pp. 85–103, 2023.

[34] Q. Ma, S. Tian, J. Wei, J. Wang, and W. W. Ng, "Attention-based spatio-temporal dependence learning network," *Inf. Sci.*, vol. 503, pp. 92–108, 2019.

[35] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?," in *Proc. Int. Conf. Mach. Learn. Appl.*, 2021, Art. no. 4.

[36] Z. Gao, Z. Li, J. Yu, and L. Xu, "Global spatiotemporal graph attention network for sea surface temperature prediction," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 1500905.

[37] H. Lin, Z. Gao, Y. Xu, L. Wu, L. Li, and S. Z. Li, "Conditional local convolution for spatio-temporal meteorological forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 7470–7478.

[38] S. Hou et al., "D2CL: A dense dilated convolutional LSTM model for sea surface temperature prediction," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 12514–12523, 2021.

[39] Z. Liu, K. Hao, X. Geng, Z. Zou, and Z. Shi, "Dual-branched spatio-temporal fusion network for multihorizon tropical cyclone track forecast," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 3842–3852, 2022.

[40] L. Shi, N. Liang, X. Xu, T. Li, and Z. Zhang, "SA-JSTN: Self-attention joint spatiotemporal network for temperature forecasting," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9475–9485, 2021.

[41] M. Crawshaw, "Multi-task learning with deep neural networks: A survey," 2020, *arXiv:2009.09796*.

[42] F. Heuer, S. Mantowsky, S. Bukhari, and G. Schneider, "Multitask-centernet (MCN): Efficient and diverse multitask learning using an anchor free approach," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 997–1005.

[43] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7482–7491.

[44] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 525–536.

[45] S. Chennupati, G. Sistu, S. Yogamani, and S. A. Rawashdeh, "MultiNet: Multi-stream feature aggregation and geometric loss strategy for multi-task learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 1200–1210.

[46] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 794–803.

[47] S. Rasp, P. D. Dueben, S. Scher, J. A. Weyn, S. Mouatadid, and N. Thuerey, "Weatherbench: A benchmark data set for data-driven weather forecasting," *J. Adv. Model. Earth Syst.*, vol. 12, no. 11, 2020, Art. no. e2020MS002203.

Lianlei Lin was born in 1980. He received the B.S. degree in measurement control technology and instrument, the M.S. degree in measurement technology and instrument, and the Ph.D. degree in instrument science and technology from the Harbin Institute of Technology, Harbin, China, in 2002, 2004, and 2009, respectively.

He is currently a Professor/Doctoral Supervisor with the Harbin Institute of Technology. His research interests include joint test technology, virtual environment simulation and modeling technology, machine learning theory, and automatic test technology.

Zongwei Zhang was born in 1996. He received the M.Sc. degree in control science and engineering from the China University of Mining and Technology, Xuzhou, China, in 2022. He is currently working toward the Ph.D. degree in electronic and information with the Harbin Institute of Technology, Harbin, China.

His research interests include deep learning and environmental modeling.

Hangyi Yu was born in 2001. She received the bachelor's degree in measurement and control technology and instrumentation in 2022 from the Harbin Institute of Technology, Harbin, China, where she is currently working toward the master's degree in information and communication engineering.

Her research interests include deep learning and environmental modeling.

Junkai Wang was born in 1993. He received the master's degree in engineering in 2020 from the Harbin Institute of Technology, Harbin, China, where he is currently working toward the doctoral degree in information and communication engineering.

His research interests include deep learning, virtual environment modeling, and data generation.

Sheng Gao was born in 1997. He received the M.Sc. degree in mechanical engineering from Northeastern University, Liaoning, China, in 2022. He is currently working toward the Ph.D. degree in information and communication engineering with the Harbin Institute of Technology.

His research interests include deep learning and virtual environment modeling.

Hanqing Zhao was born in 1997. He received the B.S. degree in electronic information science and technology and the M.S. degree in information and communication engineering from the Lanzhou University, Gansu, China, in 2019 and 2022, respectively. He is currently working toward the Ph.D. degree in information and communication engineering with the Harbin Institute of Technology, Heilongjiang, China.

His research interests include hyperspectral imagery processing and pattern recognition.

Jiaqi Zhang was born in 1998. She received the bachelor's degree in electrical information engineering from the China University of Mining and Technology, Xuzhou, China, in 2020. She is currently working toward the master's degree in electronic and information with Northeast Normal University, Changchun, China.

Her research interests include deep learning and environment modeling.