MotionUVRNN: A Motion Capture Recurrent Neural Network for Wind Field Forecast Correction via U and V Components Modeling

Zhan-Xin Gu, Jie Wu*, Wei Jin, Guo-Jing Han

***Abstract*—Wind field forecasting is crucial for human activities, but numerical weather prediction still has room to improve accuracy. In this paper, we formalize wind field forecast correction as a spatiotemporal sequence prediction task and propose MotionUVRNN, a recurrent neural network model suitable for wind field correction. The model consists of three elements: a feature decoupling module, an encoder, and a corrector. The feature decoupling module decouples the wind field into U-Wind (Wind component in x/longitude-direction) feature, V-Wind (Wind component in y/latitude direction) feature, and spatiotemporal feature. The encoder unifiedly models the motion trend and transient variation of the wind field. A novel long short-term memory dynamic capture unit (RS-LSTM) is proposed as the core part of the encoder to capture the different dynamic patterns of U-Wind and V-Wind. The corrector extracts and integrates multi-scale wind field information to capture regional features. By correcting the 3-12 hour wind field forecasts from the European Centre for Medium-Range Weather Forecasts (ECMWF; EC for short), the proposed model reduced the wind speed prediction error by 53.37% to 46.52%, and lowered the wind direction error by 21.99% to 13.26%. Moreover, this model surpasses the existing leading spatiotemporal sequence prediction models in wind field correction.**

***Index Terms*—Wind field forecast correction, Spatiotemporal prediction, Recurrent neural network, Motion trend modeling.**

I. Introduction

WIND, as an omnipresent natural phenomenon, exerts a significant impact on people's production and daily lives. The accuracy of wind forecasts is paramount in various domains, including aviation transportation, wind power generation planning, and the organization of outdoor sports events, among others [1], [2], [3]. Achieving precise predictions of wind speed and direction stands as a crucial pursuit for meteorologists, providing fundamental support across industries.

Manuscript received November 30, 2023; revised May 16, 2024.

This work was supported by the Liaoning Provincial Meteorological Service Talent Plan Science and Technology Activity Funding Project (RC202201), Key Technology Project of Liaoning Provincial Meteorological Service (LNGJ201901), Special Fund of the National Key Laboratory of Space Weather "The Correlation Impact of Geomagnetic Activity and Atmospheric Circulation Changes".

Zhan-Xin Gu is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China (email: gzx1109240926@163.com).

Jie Wu is an associate professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China (*corresponding author to provide email: wujieaa@163.com).

Wei Jin is a senior engineer of Anshan Meteorological Service, Anshan 114004, China (email: lnyk_jw@163.com).

Guo-Jing Han is an associate senior engineer of Anshan Meteorological Service, Anshan 114004, China (email: 314795969@qq.com).

Wind forecasts primarily rely on numerical weather prediction models [4], [5]. However, as numerical predictions are based on physical models, parameters and structure uncertainties may lead to results biases [6], [7]. Such biases could produce predictions that deviate from the underlying physical principles, thus generating inaccurate outcomes. Addressing and correcting these biases in forecast data is essential for improving forecast accuracy.

In recent years, various numerical bias correction methods have been proposed. For instance, He et al. employed the variational method along with historical data to correct biases in short-term numerical weather forecasts [8]. Rincón et al. introduced the WRF-ARW model, which utilizes the Kalman Filter with Model Output Statistics for correcting biases in solar irradiance forecasts [9]. Cho et al. proposed utilizing Random Forests and Support Vector Regression for temperature bias correction [10]. However, wind field data exhibits non-linear features, which cannot be handled effectively by traditional linear statistical methods [11]. Recent advancements in neural networks within atmospheric science have offered a viable solution for modeling these non-linear features [12], [13]. Due to the effectiveness of neural networks in capturing non-linear patterns, numerous deep learning models have been devised specifically for correcting biases in wind field data. For example, Han et al. introduced a deep learning model named CU-net to correct biases in wind field forecasts from the EC model [14]. Zhu et al. proposed a 3D-CNN model for decoupling the temporal and spatial features of wind speed prediction [15]. Liu et al. presented an LSTM network incorporating Variational Mode Decomposition and Singular Spectrum Analysis for multi-step wind speed forecasting [16]. However, despite addressing the limitations of traditional linear methods, these neural network models often overlook the unique spatiotemporal features present in numerical wind field forecasting data. These forecasts encapsulate spatial correlations between different locations and temporal evolutions over time, presenting challenges that standard convolutional or recurrent neural networks may struggle to handle effectively.

Wind field grid forecasts exhibit typical spatiotemporal sequence data features, which can be viewed as grid information sequences with significant spatial correlation that dynamically change over time. As a result, correcting biases in wind field forecasts has been widely recognized as a spatiotemporal sequence prediction problem. In order to gain a deeper understanding of the spatiotemporal relationships within wind fields, we introduce a novel model called MotionUVRNN. Our contributions are as follows:

1) We have introduced a novel RS-LSTM unit, which

skillfully captures the concealed dynamic patterns of U-Wind (Wind component in x/longitude-direction) and V-Wind (Wind component in y/latitude direction) through distinct internal gated units. This unit effectively extracts the fluctuating, intermittent, and chaotic features displayed by the wind field.

2) We decompose the spatiotemporal motion of the wind field into two components: motion trend and transient variation. An encoder is designed to model the motion trend of the wind field while simultaneously responding to transient variation in the wind field.

3) We have designed a corrector, which is a convolutional structure capable of extracting and integrating multi-scale information. It can further rectify the features extracted by the encoder.

4) Experimental results on the dataset demonstrate that this proposed method outperforms current mainstream spatiotemporal prediction models. Additionally, ablation studies validate the effectiveness of each designed component of MotionUVRNN.

II. Related Work

In recent years, numerous spatiotemporal sequence prediction models have been successively proposed. To address the precipitation prediction problem, Shi et al. proposed the ConvLSTM model [17]. Based on the FC-LSTM [18], this model replaces the original multiplication operation with a convolution operation, thereby better capturing spatiotemporal features. Shi et al. introduced the TrajGRU [19] model to compensate for the convolutional position invariance deficiency in ConvLSTM. Wang et al. proposed the PredRNN model [20] with a novel network structure to resolve the issue of previous models neglecting the influence of top-level output on lower levels. Moreover, they designed a new RNN unit, the Spatiotemporal LSTM (ST-LSTM), to facilitate separate transmission of temporal and spatial memory. Wang et al. introduced PredRNN++ [21], which builds on PredRNN by incorporating additional nonlinear units to capture short-term features and abrupt changes. Additionally, it includes a new spatiotemporal recursive structure called GHU to alleviate vanishing gradients. Fan et al. proposed CubicLSTM [22], which utilizes two ConvLSTMs to separately process temporal and spatial information while emphasizing ConvLSTM's position. Wang et al. proposed MIM [23], which introduces a differential operation into PredRNN's forget gate to extract stationary and non-stationary information. Lin et al. proposed the SA-ConvLSTM [24] model, which introduces a self-attention memory module into ConvLSTM to extract temporal dependencies, with the widespread application of attention mechanisms. Su et al. proposed Conv-TT-LSTM [25], which combines multi-step historical state gates to extend ConvLSTM to higher orders and uses tensor-train decomposition to reduce complexity and capture long-term dependencies. Wu et al. introduced MotionRNN [26] which can model overall motion trend as well as transient variation and can be combined with other models to improve their performance. The SimVp [27] model proposed by Tan et al. outperforms many other models despite its simple structure, such as PhyDNet [28], E3D-LSTM [29], and CrevNet [30].

Some spatiotemporal sequence prediction models have been recently applied in numerical wind field forecasting. For example, Ye et al. proposed DynamicNet [31], which employs an encoder-decoder framework to capture wind speed fluctuations, intermittency, and chaotic properties. Ye et al. also introduced SPLNet [32], which learns and captures long-range spatial dependencies through Spatiotemporal Dynamic Attention (STD-Atten) units. It uses a time-varying structure with different STD-Atten units for wind speed prediction at various time steps. Zhang et al. proposed the DETrajGRU [33] model, which can simultaneously correct both wind speed and direction. It uses a multi-task learning loss function, enabling a single model to accomplish bias correction for both speed and direction. Research has shown that decoupling wind into U-Wind and V-Wind can effectively enhance the correction accuracy of both wind direction and speed [34]. However, the decoupled U-Wind and V-Wind exhibit different dynamic patterns at the same location, necessitating separate capture of these distinct dynamic patterns [35]. The motion trend and transient variation of the wind field are not only complex and variable, but adjacent locations also exhibit similar wind field features, making it challenging to unifiedly model the entire wind field.

All the studies mentioned above have overlooked the distinct dynamic patterns exhibited by U-Wind and V-Wind at the same location in the wind field, as well as the inherent complexity and variability of the wind field itself.

III. Methodology

A. Problem Definition

Multi-step regional wind field correction is a problem that involves predicting a sequence of wind speed and direction of length T in the future. This prediction is based on a given wind field sequence X of length S, with the objective of minimizing the difference between the predicted values and the actual values. Typically, the wind field is viewed as a vector field and is decoupled into U-Wind and V-Wind to model and correct wind direction and wind speed. The wind field region is typically represented as a grid uniformly divided into M rows and N columns, where each grid point corresponds to the wind speed and direction within its designated area.

The mathematical formulation for wind field correction can be expressed as follows:

$$\widehat{Y} = F(X;\theta) \tag{1}$$

where $X \in R^{S\times P\times M\times N}$ represents the input wind field features, S denotes the length of the input spatiotemporal sequence, and P signifies the dimension of the input features. $\widehat{Y} \in R^{T\times R\times M\times N}$ represents the wind speed and wind direction sequence after correction, T denotes the length of the output sequence, R signifies the dimension of the output features. Additionally, θ denotes the model parameter.

B. The proposed MotionUVRNN

1) Model Framework: The MotionUVRNN model consists of three main components: a feature decoupling module, an encoder, and a corrector, as shown in Fig. 2. The input data X is first passed through the feature decoupling module to decompose it into separate features K, U, and V representing

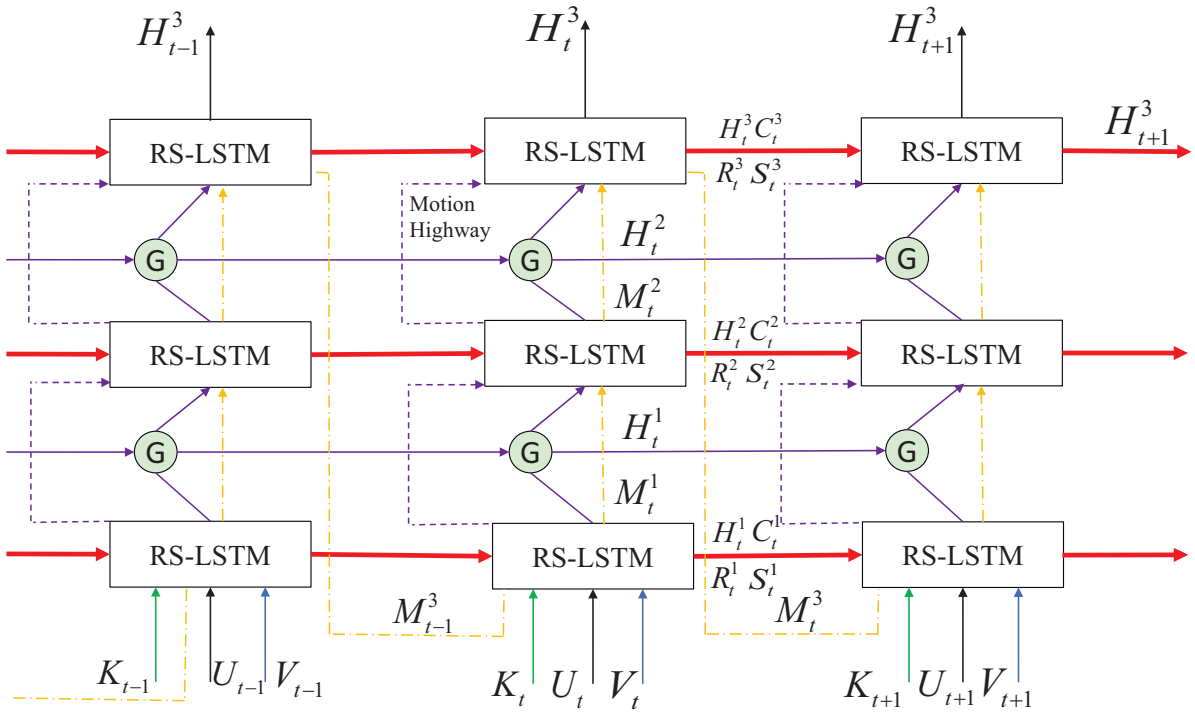


Fig. 1. Architecture of the Encoder: the MotionGRU is embedded between different layers of the RS-LSTM, and dashed lines between different layers represent the Motion Highway.

distinct features of the wind field. The encoder then extracts and integrates internal feature representations from K, U, and V to achieve unified modeling of the entire wind field. Finally, the corrector takes the encoded features F from the encoder and refines them to generate the final prediction $\widehat{U}\&\widehat{V}$ closer to the target values. Formally, the model operations are:

$$K, U, V = Decouple(X) \tag{2}$$

$$F = Encoder(K, U, V) \tag{3}$$

$$(\widehat{U}\&\widehat{V}) = Corrector(F) \tag{4}$$

where $Decouple$ denotes the feature decoupling operation, the $Encoder$ and $Corrector$ represent the encoder module and corrector module respectively. F represents the encoded features from the $Encoder$, and $\widehat{U}\&\widehat{V}$ is the final prediction after correcting.

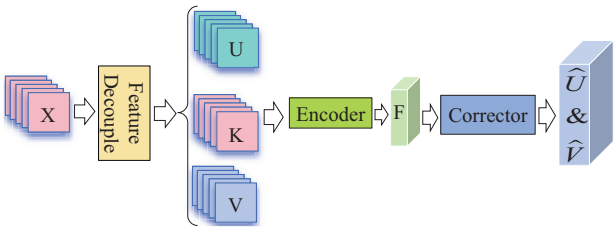


Fig. 2. The Framework of the MotionUVRNN

2) Feature decoupling module: The dynamic features of U-Wind and V-Wind are analogous, making it difficult for the model to differentiate them during training. The feature decoupling module helps enhance the model's ability to extract distinct information from the coupled input data X. It decouples the wind field data into separate components representing the overall spatiotemporal features, U-Wind dynamic features, and V-Wind dynamic features. After the feature decoupling operation, the model's ability to extract the various information contained in the wind field data can be enhanced. The formula specification of the feature decoupling module is as follows:

$$K = \tanh(X) \tag{5}$$

$$U = Slice_{\mathrm{u}}(X) \tag{6}$$

$$V = Slice_{v}(X) \tag{7}$$

where X denotes the input wind field information, $\tanh$ denotes the hyperbolic tangent activation function, K represents the input information containing the overall spatiotemporal features of the wind field, $Slice$ denotes the array slicing operation, and U and V contain the isolated dynamic patterns of U-Wind and V-Wind, respectively.

3) Encoder: Modeling the intricate spatiotemporal variations of wind fields presents a formidable challenge, given their propensity for phenomena such as dissipation, displacement, and deformation over time. Conventional spatiotemporal neural networks struggle to capture such complex wind field dynamic patterns while modeling the motion trend and transient variation of the wind field. We propose an effective encoder structure, as depicted in Fig. 1, to address these issues. Meanwhile, we employ a novel RS-LSTM unit (see section III-B4 for details) as the primary component of the encoder to capture the spatially diverse changes of U-Wind and V-Wind in the wind field. This unit can extract and integrate these features through specially designed gate units. To enhance the model's ability to handle wind field motion, we draw inspiration from the MotionRnn architecture[26], introducing the MotionGRU unit and Motion Highway. The MotionGRU unit is used to model the motion trend of the wind field and respond to transient variation in the wind field.

Additionally, the Motion Highway is utilized to balance the dynamic and static aspects of the grid in the wind field.

The overall formula for the l-th layer at time step t is as follows:

$$\mathcal{X}_t^l, \mathcal{F}_t^l, \mathcal{D}_t^l = \text{MotionGRU}\left(\mathcal{H}_t^l, \mathcal{F}_{t-1}^l, \mathcal{D}_{t-1}^l\right) \tag{8}$$

$$\mathcal{H}_t^{l+1}, \mathcal{C}_t^{l+1}, M_t^{l+1}, R_{t-1}^{l+1}, S_{t-1}^{l+1} = RS-LSTM(\mathcal{X}_t^l, \mathcal{H}_{t-1}^{l+1}, \mathcal{C}_{t-1}^{l+1}, M_t^l, R_{t-1}^{l+1}, S_{t-1}^{l+1}) \tag{9}$$

$$\mathcal{H}_t^{l+1} = \mathcal{H}_t^{l+1} + (1 - o_t) \odot \mathcal{H}_t^l \tag{10}$$

where, in Equation (8), the tensors $\mathcal{F}_t^l$ and $\mathcal{D}_t^l$ are used to model transient variation and motion trend, while $\mathcal{X}_t^l$ represents the input after the MotionGRU transformation. In Equation (9), $\mathcal{H}_{t-1}^{l+1}$ and $\mathcal{C}_{t-1}^{l+1}$ represent the hidden state and memory state from the previous time step, while M_t^l is the spatiotemporal state coming from the next layer. R_{t-1}^{l+1} and S_{t-1}^{l+1} represent the dynamic states of U-Wind and V-Wind from the previous time step, respectively. In Equation (10), o_t is the output gate based on the RNN prediction module, and $\odot$ denotes the Hadamard product. The Equation (10) represents the Motion Highway, which compensates for the output of the prediction block through the previous hidden state $\mathcal{H}_t^l$.

4) RS-LSTM: To effectively capture the distinct dynamic patterns of U-Wind and V-Wind at the same locations within the wind field, we propose the Long Short-Term Memory Dynamic Capture (RS-LSTM) Unit, as depicted in Fig. 3. In addition to possessing the hidden state H_t^l, spatiotemporal state M_t^l, and cell state C_t^l of ST-LSTM [20], also introduces the memory states R_t^l and S_t^l. Furthermore, R_t^l represents the dynamic state of U-Wind, and S_t^l represents the dynamic state of V-Wind. Given the strong correlation between dynamic states and temporal changes, both R_t^l and S_t^l exhibit horizontal movement. While the ST-LSTM captures spatiotemporal information, it does not take into account the features of wind field data and lacks the learning of inherent meteorological information within the wind field. The RS-LSTM unit's design addresses the ST-LSTM unit's limitations in discerning the dynamic patterns of U-Wind and V-Wind. RS-LSTM employs two sets of gating units, each comprising an input gate, a forget gate, and an input modulation gate. This allows for the independent capture of the hidden dynamic patterns of U-Wind and V-Wind within the data.

Specifically, in each RS-LSTM, the following calculations are performed:

$$\begin{aligned} i &= \sigma\left(W_{ix} * K_t + W_{ih} * H_{t-1}^l\right) \\ f &= \sigma\left(W_{fx} * K_t + W_{fh} * H_{t-1}^l\right) \\ g &= \tanh\left(W_{gx} * K_t + W_{g\mathrm{h}} * H_{t-1}^l\right) \\ C_t^l &= f \odot C_{t-1}^l + i \odot g \end{aligned} \tag{11}$$

$$\begin{aligned} i' &= \sigma\left(W_{i'x} * K_t + W_{i'h} * M_t^{l-1}\right) \\ f' &= \sigma\left(W_{f'x} * K_t + W_{f'h} * M_t^{l-1}\right) \\ g' &= \tanh\left(W_{g'x} * K_t + W_{g'\mathrm{h}} * M_t^{l-1}\right) \\ M_t^l &= f' \odot M_t^{l-1} + i' \odot g' \end{aligned} \tag{12}$$

$$\begin{aligned} i'' &= \sigma\left(W_{i''u} * U_t + W_{i''h} * S_{t-1}^l\right) \\ f'' &= \sigma\left(W_{f''u} * U_t + W_{f''h} * S_{t-1}^l\right) \\ g'' &= \tanh\left(W_{g''u} * U_t + W_{g''\mathrm{h}} * S_{t-1}^l\right) \\ S_t^l &= f'' \odot S_{t-1}^l + i'' \odot g'' \end{aligned} \tag{13}$$

$$\begin{aligned} i''' &= \sigma\left(W_{i'''v} * V_t + W_{i'''h} * R_{t-1}^l\right) \\ f''' &= \sigma\left(W_{f'''v} * V_t + W_{f'''h} * R_{t-1}^l\right) \\ g''' &= \tanh\left(W_{g'''v} * V_t + W_{g'''\mathrm{h}} * R_{t-1}^l\right) \\ R_t^l &= f''' \odot R_{t-1}^l + i''' \odot g''' \end{aligned} \tag{14}$$

$$o = \sigma(W_{ox} * X_t + W_{oh} * H_{t-1}^l + W_{oc} * C_t^l + W_{om} * M_t^l + W_{os} * \mathrm{S}_t^l + W_{or} * R_t^l) \tag{15}$$

$$H_t^l = o \odot \tanh\left(W_{1\times1} * \left[C_t^l, M_t^l, \mathrm{S}_t^l, R_t^l\right]\right) \tag{16}$$

where W represents the parameters that the module needs to learn, $*$ and $\odot$ denote the convolution operator and the hadamard product, respectively. K_t represents the input information of the overall spatiotemporal features of the wind field, while U_t and V_t represent the input information of the U-Wind dynamic pattern and V-Wind dynamic pattern. i, f and g denote the input gate, forget gate, and input modulation gate. σ and $\tanh$ represent the Sigmoid activation function and the hyperbolic tangent activation function, respectively. Equations 11, 12, 13, and 14 represent the calculation process of the four gating unit groups, which can respectively obtain the cell state C_t^l, the spatiotemporal state M_t^l, the U-Wind dynamic state S_t^l, and the V-Wind dynamic state R_t^l. In Equation (15), o represents the output gate, which determines the proportion of retention and discard of the new state. Equation (16) indicates that the final hidden state H_t^l of each node is determined by the combination of the cell state C_t^l, spatiotemporal state M_t^l, U-Wind dynamic state S_t^l, and V-Wind dynamic state R_t^l. In the calculation process, C_t^l, M_t^l, S_t^l and R_t^l are concatenated. Then, a 1*1 convolution layer is applied to reduce the dimension so that the final hidden state H_t^l has the same dimension as the other memory states. Finally, the Hadamard product is multiplied by the output gate to obtain the final result H_t^l.

5) Corrector: The Corrector excels in modeling local area trends within the wind field, effectively capturing local spatial correlations, as depicted in Fig. 4. It leverages the capabilities of multiple upsampling and downsampling techniques to extract information from varying scales embedded within the wind field data. Additionally, it integrates multi-scale information through a skip connection scheme, enhancing its effectiveness in capturing the complexity of the data. These mechanisms collectively aid the Corrector in refining the encoder's output, thereby amplifying the correction effect on areas in the wind field where the wind speed markedly surpasses the average value. Additionally, they equip the model with the capacity to discern when the wind direction aligns in both the overall and local areas. The Corrector's design draws inspiration from the UNet++ [36] model, progressively rectifying the encoder features, rendering the predicted wind field increasingly precise and detailed. The correction process can be analogized to the procedure of iteratively sharpening a blurry image.

The skip connection path of the Corrector can be described in the following way. $x^{i,j}$ represents the output of node $X^{i,j}$, i is the index number along the downsampling direction, and j is the index number in the horizontal direction. The stack of feature maps represented by $x^{i,j}$ is computed as follows:

$$x^{i,j} = \begin{cases} \mathcal{H}\left(\mathcal{D}\left(x^{i-1,j}\right)\right), & j = 0 \\ \mathcal{H}\left(\left[\left[x^{i,k}\right]_{k=0}^{j-1}, \mathcal{U}\left(x^{i+1,j-1}\right)\right]\right), & j > 0 \end{cases} \tag{17}$$

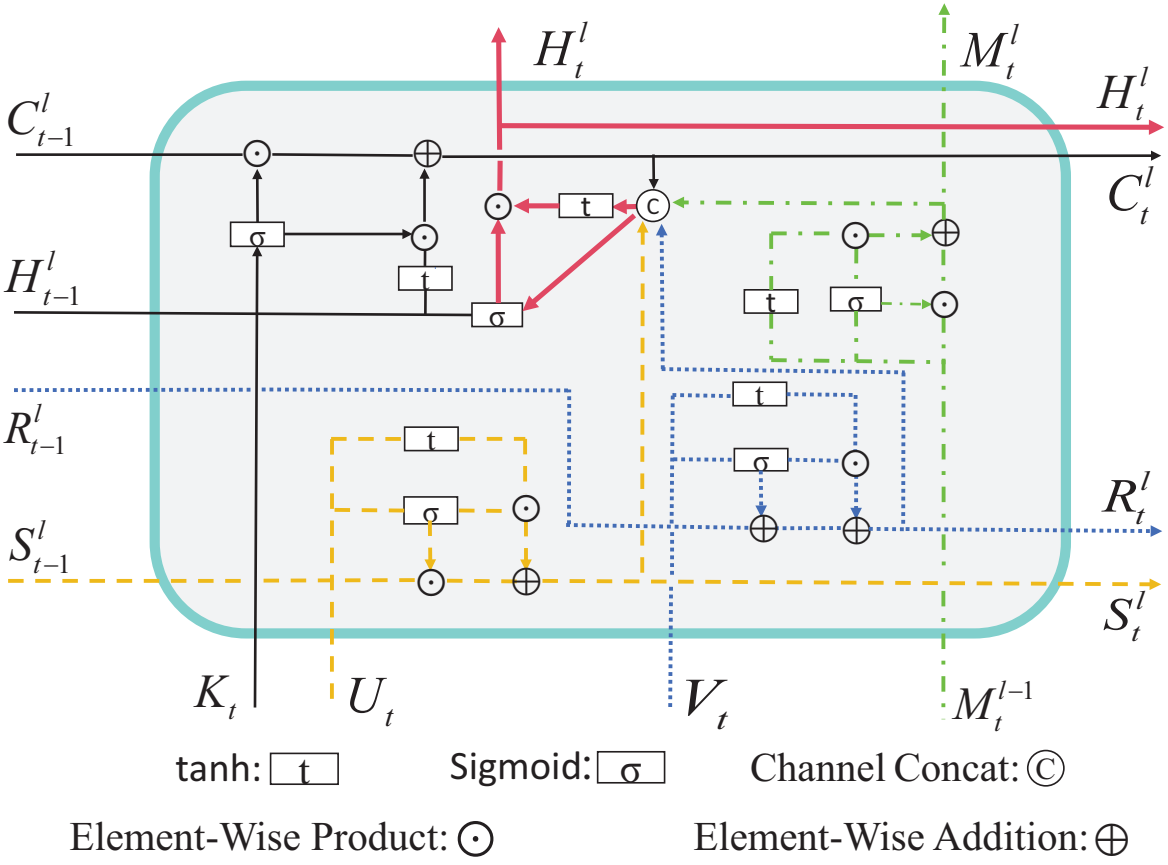


Fig. 3. Architecture of RS-LSTM

where the function $\mathcal{H}()$ is a convolution operation followed by an activation function, $\mathcal{U}()$ represents the upsampling layer, $\mathcal{D}()$ represents the downsampling layer, and [] represents the concatenation layer.

6) Loss Function: Wind field correction consists of two subtasks: wind speed correction and wind direction correction. Wind speed correction focuses primarily on the numerical magnitudes of U-Wind and V-Wind. In contrast, wind direction correction pays more attention to the sign correctness of U-Wind and V-Wind. Traditional regression loss functions like MAE and MSE cannot satisfy the diverse needs of these two different tasks. Zhang et al. proposed dedicated loss functions for wind direction and speed correction respectively [33]. Using task-specific losses facilitates rapid model convergence when training for these disparate tasks. The loss functions are defined as follows: MAE_{dir} represents the wind direction loss function, and MAE_{speed} represents the wind speed loss function.

$$MAE_{dir} = \frac{1}{L}\sum_{n=1}^{N}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(\begin{array}{l}\left|U_{n,i,j}-\widehat{U}_{n,i,j}\right| \\ + \\ \left|V_{n,i,j}-\widehat{V}_{n,i,j}\right|\end{array}\right) \tag{18}$$

$$MAE_{\text{speed}} = \frac{1}{L}\sum_{n=1}^{N}\sum_{i=1}^{H}\sum_{j=1}^{W}\left|\begin{array}{l}\sqrt{U_{n,i,j}^2+V_{n,i,j}^2} \\ - \\ \sqrt{\widehat{U}_{n,i,j}^2+\widehat{V}_{n,i,j}^2}\end{array}\right| \tag{19}$$

where L is the product of N, H, and W, N is the product of the number of samples per batch B, and the length of the output time sequence T. Furthermore, H and W represent the height and width of the model's output, respectively.

IV. Experiments

A. Experiment data

The experimental data used in this study originated from historical wind field data obtained from the Anshan region in Liaoning Province, China, covering the period from 2017 to 2019. The data from 2017 and 2018 formed the training set, while the 2019 data served as the test set. The data set comprised wind field numerical forecasts issued by EC and real wind field measurement data supplied by the China National Meteorological Center. The model training utilized the U-Wind, V-Wind, and surface pressure features from EC's wind field numerical forecast and the U-Wind and V-Wind features from the actual wind field data. The dataset had a temporal resolution of 3 hours and a spatial resolution of 64*64 grid cells based on longitude and latitude.

B. Data Preprocessing and Sample Construction

To address the issue of prediction errors accumulating with increasing time steps in the multi-step prediction process of the RNN model, a strategy is employed where multi-step predictions are decomposed into multiple single-step predictions [37]. A unique single-time-step rolling method is employed to construct samples based on the data characteristics. In this method, the EC input features closest to the prediction time point are provided at each step throughout the entire

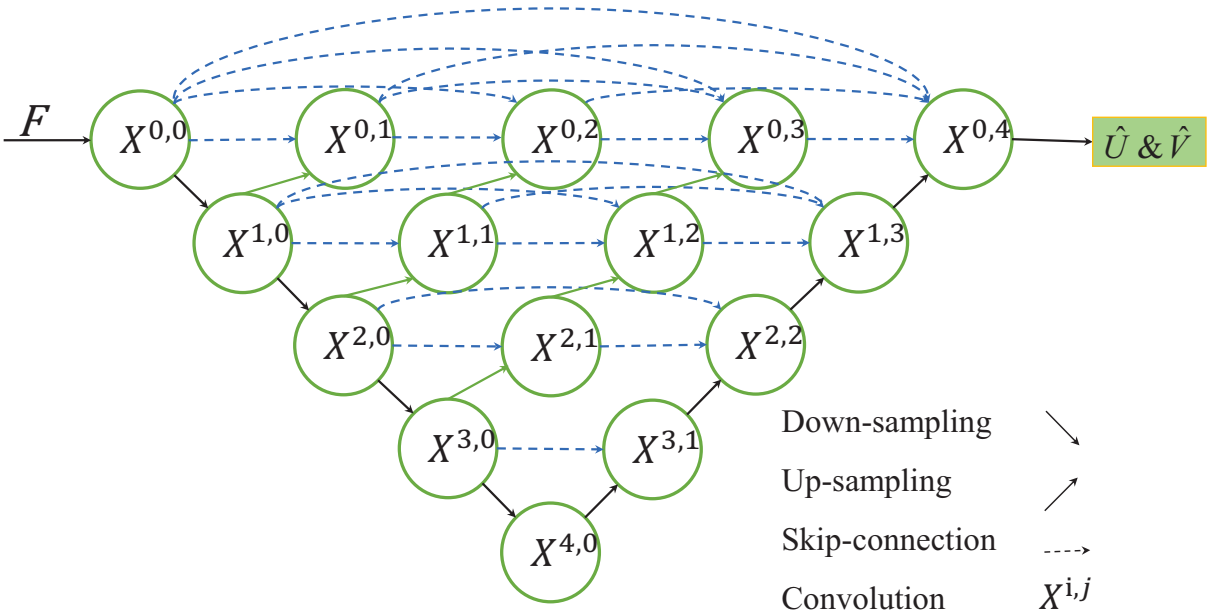


Fig. 4. Architecture of the Corrector, F represents the features extracted by the encoder, while $\hat{U}\&\hat{V}$ denotes the result after correction.

prediction horizon. These features include U-Wind and V-Wind data, along with other relevant features for wind field forecasting. This approach maximizes the utilization of the available data for accurate predictions.

During preprocessing, U-Wind and V-Wind are not normalized, as their signs hold wind direction information that would be lost if normalized. In contrast, surface pressure values are much larger in magnitude without directional significance, so they are normalized to facilitate learning. The normalization formula is as follows:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{20}$$

where X represents the original data, while $X_{\max}$ and $X_{\min}$ denote the maximum and minimum values of the data, respectively.

C. Evaluation MetricsEvaluation Metrics

The model's predictive targets are U-Wind and V-Wind, which need to be converted into wind speed and wind direction before error calculation can be performed. The functions for converting U-Wind and V-Wind into wind speed and wind direction are shown in Equation (21) and Equation (22), respectively.

$$\text{speed} = \sqrt{\text{U}^2 + \text{V}^2} \tag{21}$$

$$\text{dir} = \begin{cases} 270 - \arctan(V/U) * 180/\pi & \text{U} > 0 \\ 90 - \arctan(V/U) * 180/\pi & \text{U} < 0 \\ 180 & \text{U} = 0 \text{and} \text{V} > 0 \\ 360 & \text{U} = 0 \text{and} \text{V} < 0 \\ 0 & \text{U} = 0 \text{and} \text{V} = 0 \end{cases} \tag{22}$$

To intuitively evaluate the model's performance, two evaluation metrics are used to assess the model's performance in wind speed correction and wind direction correction, respectively. For wind speed correction, the Mean Absolute Error (MAE) function is used to calculate the error between the predicted wind speed and the actual wind speed, as defined in Equation (23). For wind direction correction, according to the "QXT 229–2014 Verification Method for Wind Forecast", the MAEd function is used to calculate the error between the corrected wind direction value and the actual value, as defined in Equation (24) [14].

$$MAE = \frac{1}{T * H * W} \sum_{t=1}^{T} \sum_{i=1}^{H} \sum_{j=1}^{W} |y_{t,i,j} - \widehat{y}_{t,i,j}| \tag{23}$$

$$MAEd = \frac{1}{T * H * W} \sum_{t=1}^{T} \sum_{i=1}^{H} \sum_{j=1}^{W} |E_{t,i,j}| \tag{24}$$

$$\text{E}_{\text{n,i,j}} \begin{cases} D_{n,i,j} & -180 \le D_{n,i,j} \le 180 \\ D_{n,i,j} + 360 & D_{n,i,j} < -180 \\ D_{n,i,j} - 360 & D_{n,i,j} > 180 \end{cases} \tag{25}$$

$$D_{n,i,j} = \widehat{\text{dir}}_{n,i,j} - \text{dir}_{n,i,j} \tag{26}$$

where T represents the length of the output time sequence, while H and W denote the height and width of the study area, respectively. In this context, T is set to 4, both H and W are set to 64.

D. Experimental Results and Analysis

TABLE I
COMPARISON MAE RESULTS OF DIFFERENT MODELS IN WIND SPEED CORRECTION TASK

Models	3h	6h	9h	12h
PredRNN[20]	0.5908	0.6493	0.6717	0.6797
PredRNN++[21]	0.5844	0.6482	0.6701	0.6755
CubicLSTM[22]	0.5833	0.6474	0.6670	0.6746
MIM[23]	0.5817	0.6478	0.6672	0.6720
Conv-TT-LSTM[25]	0.5883	0.6437	0.6636	0.6703
MotionPredRnn[26]	0.5815	0.6387	0.6608	0.6683
SimVP[27]	0.6043	0.6575	0.6762	0.6830
MotionUVRNN	0.5667	0.6252	0.6494	0.6499

TABLE II
COMPARISON MAED RESULTS OF DIFFERENT MODELS IN WIND DIRECTION CORRECTION TASK

Models	3h	6h	9h	12h
PredRNN	37.9151	41.4038	42.6962	42.8014
PredRNN++	37.8137	41.3412	42.5241	42.7533
CubicLSTM	37.6307	41.2683	42.3559	42.7225
MIM	37.7903	41.1996	42.4195	42.5823
Conv-TT-LSTM	37.7744	41.2044	42.3736	42.4729
MotionPredRnn	37.4989	41.1435	42.2130	42.3568
SimVP	36.8724	40.1874	41.3368	41.5858
MotionUVRNN	36.2234	38.9162	39.9073	40.2750

A quantitative analysis of the performance of MotionUVRNN in wind speed and wind direction correction tasks has been conducted. The analysis results are shown in Table I and Table II, respectively. The MAE evaluation metric of MotionUVRNN in the wind speed correction task outperforms all comparison models. Compared with MotionPredRnn, which performs best among the comparison models, the MAE for future 3, 6, 9, and 12 hours has been reduced by 0.0148, 0.0135, 0.0114, and 0.0184, respectively. In the wind direction correction task, the MAEd evaluation metric of MotionUVRNN also achieved the best results. Compared with SimVP, which performs best in the wind direction task among the comparison models, the MAEd loss values for future 3, 6, 9, and 12 hours have been reduced by 0.6490, 1.2712, 1.4295, and 1.3108, respectively. We calculated the error between the EC wind field forecast and the actual values, with a wind speed error (MAE) of 1.2153 and a wind direction error (MAEd) of 46.4317. It is evident that the error values of MotionUVRNN for future 3, 6, 9, and 12 hours are significantly lower than those before correction. The error in the wind speed correction task has decreased by 53.37% to 46.52%, and the error in the wind direction correction task has decreased by 21.99% to 13.26%.

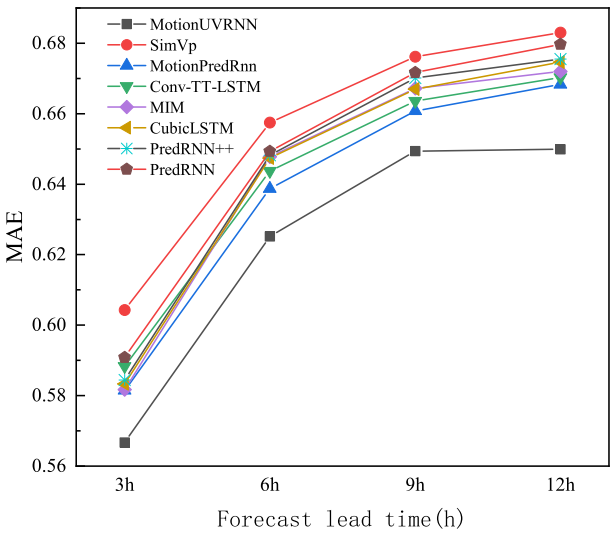


Fig. 5. Framewise MAE comparison

Figures 5 and 6 provide a more detailed step-by-step comparison of evaluation metrics. The figures show that the

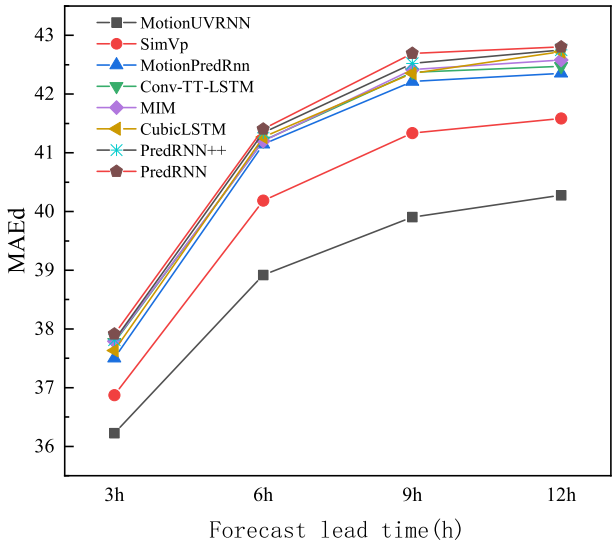


Fig. 6. Framewise MAEd comparison

MotionUVRNN model outperforms the current mainstream spatiotemporal sequence prediction models in both wind direction and wind speed correction tasks.

E. Ablation experiment

We conducted an ablation study on MotionUVRNN to evaluate the contribution of the Corrector module by comparing model performance with and without it. The results show that with the introduction of the Encoder and Corrector modules, the model's performance on these metrics has improved. The experimental results are shown in Table III and Table IV, respectively. In the tables, the symbols w and w/o represent with and without, respectively. Meanwhile E and C denote Encoder and Corrector, respectively.

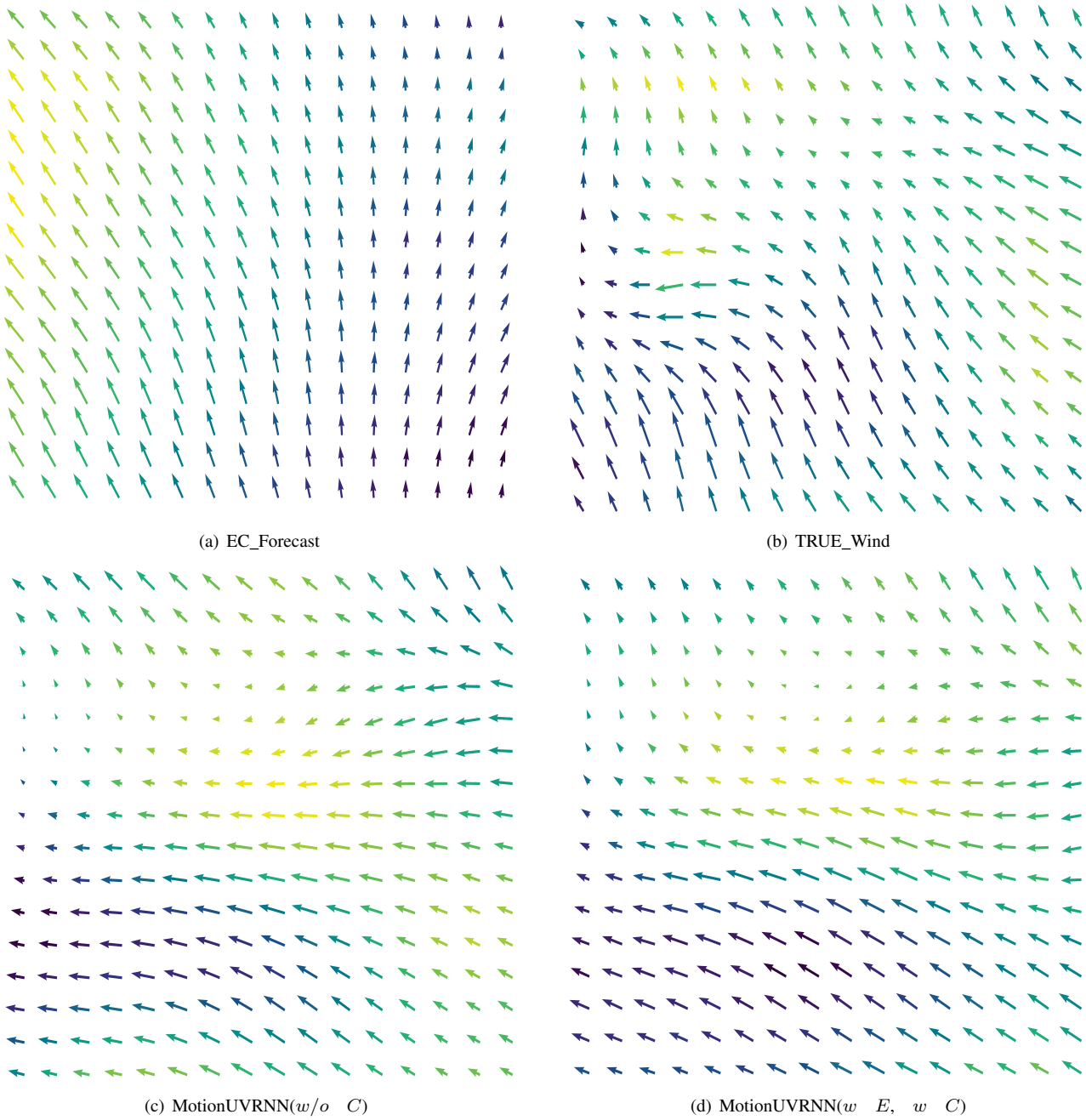
TABLE III
ABLATION EXPERIMENT OF MOTIONUVRNN ON WIND SPEED TASK

Module	3h	6h	9h	12h
$w/o\ \ C$	0.5695	0.6264	0.6507	0.6528
$w\ \ E,\ \ w\ \ C$	0.5667	0.6252	0.6494	0.6499

TABLE IV
ABLATION EXPERIMENT OF MOTIONUVRNN ON WIND DIRECTION TASK

Module	3h	6h	9h	12h
$w/o\ \ C$	36.8093	40.0757	41.0562	41.3456
$w\ \ E,\ \ w\ \ C$	36.2234	38.9162	39.9073	40.2750

To visually demonstrate the wind field correction results and ablation study, we have created visualizations of wind field data in the research area. These visualizations show the wind speed and wind direction correction results, as shown in Fig. 7. In the figures, the arrows indicate the wind direction at each location, with the length representing the wind speed magnitude.



(a) EC_Forecast (b) TRUE_Wind

(c) MotionUVRNN(w/o C) (d) MotionUVRNN(w E, w C)

Fig. 7. Example of Correction Results for Different Models

As shown in Fig. 7(b), the real wind field exhibits complex spatial patterns. In contrast, the EC forecast in Fig. 7(a) can only roughly capture the overall wind trends, with significant deficiencies in local details. After MotionUVRNN(w/o C) correction, the wind field in Fig. 7(c) shows greatly improved details compared to before correction. This enhancement stems from the Encoder, which can model the wind field's motion trend and transient variation in a unified manner. Its internal RS-LSTM units can also separately capture the dynamic patterns of U-Wind and V-Wind at the same location in the wind field.

Introducing the Corrector module further refines the local depictions in Fig. 7(d) to be closer to the real wind field versus Fig. 7(c). This is because the wind field features in local areas of the wind field are likely to be consistent. Since the spatiotemporal sequence prediction model composed of a combination of convolution and LSTM units is limited by its number of layers, it is difficult to fully exploit the deep information contained in the wind field. Therefore, it is necessary to use multi-layer convolutional models to further process the features generated after the spatiotemporal prediction model extraction. The convolution operation and up and downsampling operations in the Corrector module help the model learn the multi-scale information hidden in the wind field, thereby enhancing the model's performance in wind field correction.

F. Experimental Details

Given the transient and abrupt nature of wind field variations, we have chosen an input sequence length of S=7 and an output sequence length of T=4. Furthermore, the changes in the wind field are closely related to surface pressure [38].

The high and low pressure determine the wind direction, and the size of the pressure difference affects the wind speed. This study introduces surface pressure as an auxiliary feature to improve the accuracy of correcting the EC wind field forecast.

In order to ensure the fairness of the experimental results, this study adopts the same experimental settings for all models involved in the comparison. The hidden state size is configured as 48 for all RNN structure models, excluding the SimVP convolutional network, with a stacking of 4 layers. In the experiment, we employed Adam as the optimizer and initialized the learning rate to 0.00006 for the wind speed correction task and 0.00004 for the wind direction correction task. During training, we used a batch size of 16 and conducted training for 30 epochs before termination.

V. Conclusion

This work transforms wind field correction into a spatiotemporal sequence forecasting problem, proposing MotionUVRNN well suited for short-term wind field forecast correction. MotionUVRNN can effectively correct short-term EC wind field forecasts for the upcoming 12 hours simultaneously. Compared to the original EC forecasts, MotionUVRNN achieves a significant reduction in errors across the 3-12 hour horizons. Specifically, it decreases wind speed errors by 53.37% to 46.52%, and reduces wind direction errors by 21.99% to 13.26%.

Furthermore, MotionUVRNN outperforms current mainstream spatiotemporal models in wind field correction tasks, highlighting its effectiveness. For future work, we plan to incorporate physical constraints to further enhance the model's performance. In summary, this research lays the groundwork for further improving the accuracy of wind field forecasts, particularly by enhancing numerical weather prediction models using deep learning techniques.

References

[1] Y. Jin, S. Wang, Z. Ling, D. Wang, and G. Wang, "A hybrid prediction framework based on deep learning for wind power," in *2023 3rd International Conference on Neural Networks, Information and Communication Engineering (NNICE)*. IEEE, 2023, pp. 116–121.

[2] Yu Ye, Jinxing Che, and Heping Wang, "Optimal Component IGSCV-SVR Ensemble Model Improved by VMD for Ultra-short-term Wind Speed Forecasting," *Engineering Letters*, vol. 30, no.3, pp1166-1175, 2022

[3] W. J. Ross and M. Orr, "Predicting climate impacts to the olympic games and fifa men's world cups from 2022 to 2032," *Sport in Society*, vol. 25, no. 4, pp. 867–888, 2022.

[4] P. Bauer, A. Thorpe, and G. Brunet, "The quiet revolution of numerical weather prediction," *Nature*, vol. 525, no. 7567, pp. 47–55, 2015.

[5] X. Shen, J. Wang, Z. Li, D. Chen, and J. Gong, "Research and operational development of numerical weather prediction in china," *Journal of Meteorological Research*, vol. 34, no. 4, pp. 675–698, 2020.

[6] M. G. Schultz, C. Betancourt, B. Gong, F. Kleinert, M. Langguth, L. H. Leufen, A. Mozaffari, and S. Stadtler, "Can deep learning beat numerical weather prediction?" *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2194, p. 20200097, 2021.

[7] S. Vannitsem, J. B. Bremnes, J. Demaeyer, G. R. Evans, J. Flowerdew, S. Hemri, S. Lerch, N. Roberts, S. Theis, A. Atencia *et al.*, "Statistical postprocessing for weather forecasts–review, challenges and avenues in a big data world," *Bulletin of the American Meteorological Society*, pp. 1–44, 2020.

[8] D. He, Z. Zhou, Z. Kang, L. Liu *et al.*, "Numerical studies on forecast error correction of grapes model with variational approach," *Advances in Meteorology*, vol. 2019, 2019.

[9] A. Rincón, O. Jorba, M. Frutos, L. Alvarez, F. P. Barrios, and J. A. González, "Bias correction of global irradiance modelled with weather and research forecasting model over paraguay," *Solar Energy*, vol. 170, pp. 201–211, 2018.

[10] D. Cho, C. Yoo, J. Im, and D.-H. Cha, "Comparative assessment of various machine learning-based bias correction methods for numerical weather prediction model forecasts of extreme air temperatures in urban areas," *Earth and Space Science*, vol. 7, no. 4, p. e2019EA000740, 2020.

[11] S. Wang, J. Wang, H. Lu, and W. Zhao, "A novel combined model for wind speed prediction–combination of linear model, shallow neural networks, and deep learning approaches," *Energy*, vol. 234, p. 121275, 2021.

[12] S. Rasp and S. Lerch, "Neural networks for postprocessing ensemble weather forecasts," *Monthly Weather Review*, vol. 146, no. 11, pp. 3885–3900, 2018.

[13] Gonggui Chen, Mengyuan Zhu, Jing Huang, Yi Fu, Xiaochuan Xie, and Hongyu Long, "Short-term Wind Speed Prediction with Master-slave Performance Based on CNN-LSTM and Improved POABP," *Engineering Letters*, vol. 31, no.2, pp848-861, 2023

[14] L. Han, M. Chen, K. Chen, H. Chen, Y. Zhang, B. Lu, L. Song, and R. Qin, "A deep learning method for bias correction of ecmwf 24–240 h forecasts," *Advances in Atmospheric Sciences*, vol. 38, no. 9, pp. 1444–1459, 2021.

[15] X. Zhu, R. Liu, Y. Chen, X. Gao, Y. Wang, and Z. Xu, "Wind speed behaviors feather analysis and its utilization on wind speed prediction using 3d-cnn," *Energy*, vol. 236, p. 121523, 2021.

[16] H. Liu, X. Mi, and Y. Li, "Smart multi-step deep learning model for wind speed forecasting based on variational mode decomposition, singular spectrum analysis, lstm network and elm," *Energy Conversion and Management*, vol. 159, pp. 54–64, 2018.

[17] 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," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[18] Y. Yang, J. Dong, X. Sun, E. Lima, Q. Mu, and X. Wang, "A cfcc-lstm model for sea surface temperature prediction," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 207–211, 2017.

[19] X. Shi, Z. Gao, L. Lausen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo, "Deep learning for precipitation nowcasting: A benchmark and a new model," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[20] Y. Wang, H. Wu, J. Zhang, Z. Gao, J. Wang, S. Y. Philip, and M. Long, "Predrnn: A recurrent neural network for spatiotemporal predictive learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 2208–2225, 2022.

[21] Y. Wang, Z. Gao, M. Long, J. Wang, and S. Y. Philip, "Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5123–5132.

[22] H. Fan, L. Zhu, and Y. Yang, "Cubic lstms for video prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8263–8270.

[23] Y. Wang, J. Zhang, H. Zhu, M. Long, J. Wang, and P. S. Yu, "Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9154–9162.

[24] Z. Lin, M. Li, Z. Zheng, Y. Cheng, and C. Yuan, "Self-attention convlstm for spatiotemporal prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 531–11 538.

[25] J. Su, W. Byeon, J. Kossaifi, F. Huang, J. Kautz, and A. Anandkumar, "Convolutional tensor-train lstm for spatio-temporal learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 13 714–13 726, 2020.

[26] H. Wu, Z. Yao, J. Wang, and M. Long, "Motionrnn: A flexible model for video prediction with spacetime-varying motions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 435–15 444.

[27] Z. Gao, C. Tan, L. Wu, and S. Z. Li, "Simvp: Simpler yet better video prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3170–3180.

[28] V. L. Guen and N. Thome, "Disentangling physical dynamics from unknown factors for unsupervised video prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 474–11 484.

[29] Y. Wang, L. Jiang, M.-H. Yang, L.-J. Li, M. Long, and L. Fei-Fei, "Eidetic 3d lstm: A model for video prediction and beyond," in *International Conference on Learning Representations*, 2018.

[30] W. Yu, Y. Lu, S. Easterbrook, and S. Fidler, "Efficient and information-preserving future frame prediction and beyond," in *International Conference on Learning Representations*, 2019.

[31] R. Ye, X. Li, Y. Ye, and B. Zhang, "Dynamicnet: A time-variant ode network for multi-step wind speed prediction," *Neural Networks*, vol. 152, pp. 118–139, 2022.
[32] R. Ye, S. Feng, X. Li, Y. Ye, B. Zhang, and C. Luo, "Splnet: A sequence-to-one learning network with time-variant structure for regional wind speed prediction," *Information Sciences*, vol. 609, pp. 79–99, 2022.
[33] W. Zhang, Y. Jiang, J. Dong, X. Song, R. Pang, B. Guoan, and H. Yu, "A deep learning method for real-time bias correction of wind field forecasts in the western north pacific," *Atmospheric Research*, vol. 284, p. 106586, 2023.
[34] G. Turbelin, P. Ngae, and M. Grignon, "Wavelet cross-correlation analysis of wind speed series generated by ann based models," *Renewable Energy*, vol. 34, no. 4, pp. 1024–1032, 2009.
[35] R. Ye, S. Feng, X. Li, Y. Ye, B. Zhang, Y. Zhu, Y. Sun, and Y. Wang, "Wdmnet: Modeling diverse variations of regional wind speed for multi-step predictions," *Neural Networks*, vol. 162, pp. 147–161, 2023.
[36] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, 2018, pp. 3–11.
[37] K. Bi, L. Xie, H. Zhang, X. Chen, X. Gu, and Q. Tian, "Accurate medium-range global weather forecasting with 3d neural networks," *Nature*, vol. 619, no. 7970, pp. 533–538, 2023.
[38] T. Yu and R. Yang, "Temporal dynamic network with learnable coupled adjacent matrix for wind forecasting," *IEEE Geoscience and Remote Sensing Letters*, 2023.