

# Temporal downscaling meteorological variables to unseen moments: Continuous temporal downscaling via Multi-source Spatial-temporal-wavelet feature Fusion and Time-Continuous Manifold

Sheng Gao <sup>a,b</sup> , Lianlei Lin <sup>a,b</sup> , \* Zongwei Zhang <sup>a,b</sup> , Jiawei Wang <sup>c</sup>

<sup>a</sup> School of Electronic Information Engineering, Harbin Institute of Technology, Harbin, 150000, Heilongjiang, China

<sup>b</sup> Technological Innovation Center of Littoral Test, Harbin, 150000, Heilongjiang, China

<sup>c</sup> Beike Findfang Technology Co., Ltd, Beijing, 100085, China



## ARTICLE INFO

### Keywords:

Meteorological data  
Temporal downscaling  
Deep learning  
Infomation fusion  
Attention mechanism

## ABSTRACT

Accurate modeling of meteorological variables with high temporal resolution is crucial for simulations and decision-making in aviation, aerospace, and other engineering sectors. Conventional meteorological products typically have temporal resolutions exceeding one hour, hindering the characterization of short-term nonlinear evolutions in meteorological variables. Current temporal downscaling methods encounter challenges of insufficient multi-source data fusion, limited extrapolation capabilities of data distributions, and inadequate learning of spatiotemporal dependencies, leading to low modeling accuracy and difficulties in modeling meteorological environments with higher temporal resolutions than those in the training data. To address these issues, this study proposes MSF-TCMA (Multi-source Spatial-temporal-wavelet feature Fusion and Time-Continuous Manifold-based Algorithm) for continuous temporal downscaling. The algorithm introduces multiscale deep-wavelet feature extraction branch for integrating spatial dependence and the cross-modal spatiotemporal information fusing branch for fusing multi-source information and learning temporal dependence. The time-continuous manifold sampling branch is used to solve the problem of data distribution extrapolation. Finally, the algorithm's continuous downscaling performance is optimized by employing multi-moment weighted meteorological state estimation-energy change deduction joint loss. Two regional case studies demonstrate that MSF-TCMA achieves modeling errors of less than 0.65 K for 2-meter temperature, less than 36.24 Pa for surface pressure, and less than 0.38 m/s for wind speed over a 6-hour time interval, with errors reduced by 3.99–99.64% compared to the comparison methods. Furthermore, two engineering experiments demonstrate that the method realizes continuous downscaling of multiple moments in a time interval (including for unseen moments during the algorithm training phase), and downscaling prediction of future meteorological states based on GFS forecast data. This study provides a new paradigm for high-precision and high-temporal resolution reconstruction of meteorological data, which is of great application value for optimization and risk control of complex engineering activities. The code is available at: <https://github.com/shermo1415/MSF-TCMA/>.

## 1. Introduction

Accurately simulating meteorological variables within specific spatial and temporal ranges is crucial for assessing the potential impact of the meteorological environment on human activities and provides a scientific basis for the optimization and safe management of complex systems in aviation, aerospace, and other engineering sectors. For instance, simulating and reconstructing aviation accidents require modeling of meteorological variables such as air pressure, temperature, wind speed, and humidity during the accident. Spacecraft launches require predicting wind speed and air pressure at the launch site within

the launch window. These scenarios usually last for only a few minutes, so these tasks require not only obtaining high-precision meteorological variables to accurately reflect the environmental impact on activities but also ensuring that the data has a high temporal resolution to simulate the real-time interaction between the environment and human activities. However, existing meteorological products usually have a low temporal resolution, (e.g., 6 h for GFS forecast products [contributor, 2015](#) and 1 h for ERA5 reanalysis data [Hersbach et al., 2023](#)). In these application scenarios that require rapid response to environmental changes, low temporal resolution data are difficult to characterize

\* Corresponding author at: School of Electronic Information Engineering, Harbin Institute of Technology, Harbin, 150000, Heilongjiang, China.  
E-mail addresses: [sheng1997@stu.hit.edu.cn](mailto:sheng1997@stu.hit.edu.cn) (S. Gao), [linlianlei@hit.edu.cn](mailto:linlianlei@hit.edu.cn) (L. Lin).

short-term environmental changes, limiting their practical application. Therefore, it has become an important challenge for meteorological environmental modeling to investigate how to infer high temporal resolution meteorological states from low temporal resolution data to enhance the temporal continuity, so as to model the dynamic changes of meteorological data and provide realistic environmental support for engineering applications.

Temporal downscaling techniques have been widely studied in recent years as an effective means of improving the temporal resolution of meteorological data. Its core objective is to estimate the meteorological state of other moments in the temporal interval by meteorological state at the boundary moment, so as to map the low-temporal-resolution meteorological data to a higher resolution output. Temporal downscaling can help to recover high-frequency dynamical details of meteorological data to meet the demand for high-resolution meteorological environmental data for engineering applications. The problem of temporal downscaling within a spatiotemporal region can be formulated as the problem of temporal super-resolution of spatiotemporal series, i.e., given a spatiotemporal segment, to estimate the spatial states of the moments within the time interval by frame interpolation (Dong et al., 2023). Since the evolution of meteorological data involves the coupling of spatial, temporal, and multiple meteorological factors (Zhang et al., 2025), and the availability of high temporal resolution meteorological data is limited (Sun et al., 2024b), accurate and effective temporal downscaling is a complex problem, the main challenges are as follows:

- **Insufficient multi-source data fusion.** Previous studies focus only on the spatial and temporal evolution of meteorological variables themselves, ignoring the important role of heterogeneous data, which are closely related to meteorological variables, in the characterization and modeling of meteorological evolution (Yi et al., 2025). For example, the temporal evolution of wind speed can not only be represented by the u and v-components of spatial observation points. Satellite observation data, as heterosource data can reflect the association between atmospheric motion and wind field evolution in a more intuitive and comprehensive way. Therefore, downscaling methods with a single data source often fail to capture the multisource features of meteorological evolution, limiting the improvement of downscaling accuracy.
- **Data distribution extrapolation constraints.** Most of the current temporal downscaling studies adopt a data-driven model, relying on pre-collected high-temporal resolution data for model construction, which leads to the fact that the existing methods can only estimate the meteorological state within the existing data distribution. Specifically, existing methods can only estimate the discrete meteorological states at anchor moments within the temporal intervals during the training phase, but cannot extrapolate the states at unseen moments outside the data distribution (i.e., unrecorded non-training anchor moments within the interval). This means that the upper limit of the actual temporal resolution of the algorithm's output is constrained by the resolution of the collected training data, making it difficult to overcome the limitations imposed by the temporal resolution of the training dataset.
- **Spatial dependence on meteorological variables.** Meteorological variables show different local spatial dependence across different spatial scales, and the local spatial state is correlated with the global spatial state (Sun et al., 2025). However, existing convolution-based downscaling methods typically focus on extracting local spatial features of meteorological variables and fail to capture global spatial dependencies. Flow-based methods assume a fixed spatial characterization pattern in the downscaling interval, making it difficult to fully represent the spatial correlation of meteorological variables. The inadequate learning of spatial dependence restricts the accuracy of temporal downscaling.

• **Temporal dependence on meteorological state.** Meteorological variables evolve nonlinearly in the temporal interval, they show randomness and fluctuation at different moments in the relative temporal interval (Sun et al., 2024a). At the same time, there is a deterministic correlation between their evolutionary trend and the absolute time in which they are located, i.e., the meteorological variables show periodicity with respect to months, days, and hours (Gao et al., 2024). Although the existing downscaling methods consider the correlation of temporal boundary states, they ignore the relative time nonlinearity of the meteorological parameters and the correlation between the meteorological variables and the absolute time, makes it difficult for these methods to model the time-deterministic and uncertain evolution of the meteorological variables, and exacerbates the downscaling errors.

Existing temporal downscaling methods for meteorological variables can be categorized into three groups: interpolation-based methods, machine learning-based methods, and deep learning-based methods. Interpolation-based methods assume the meteorological variables evolve linearly, then estimate the meteorological state in the interval by linear interpolating the temporal boundary states. Buster et al. (2021) uses linear interpolation to downscale solar radiation from 30 min to 5 min. McDowell and Kummert (2016) and McDowell et al. (2018) applied modified linear interpolation algorithms for downscale the series into sub-hourly resolution. Compared to linear interpolation, spline interpolation provides smoother edges and accurate interpolation results. Ruiz-Arias (2022) proposed a second-order cubic polynomial to reduce the downscaling error. Balog et al. (2023) explored the effectiveness of the spline interpolation method in temporal downscaling tasks for temperature, humidity, and solar radiation, increasing the temporal resolution from 1 h to sub-hourly. The advantage of the interpolation method is that the mathematical form is explicit and unconstrained by the data distribution, enabling continuous estimation at any moment. However, the lack of ability to fit nonlinear temporal evolution makes it applicable only to temporal downscaling of deterministic time series, and it is difficult to fit the complex temporal dependence of meteorological variables such as temperature and wind speed, which are essentially nonlinearly evolving (Omoyele et al., 2024). Especially as the temporal range extends, the evolution of meteorological variables becomes increasingly nonlinear and prone to high fluctuations. As a result, the interpolation method is no longer adept at capturing the temporal nonlinear dependence, leading to a significant degradation in accuracy. On the other hand, the interpolation method only considers the relative temporal dependence of a single spatial observation point, and cannot integrate the spatial dependence of different spatial observation points in the spatiotemporal series.

Machine learning-based methods break through the linear constraints by constructing nonlinear models, which theoretically can capture the nonlinear temporal dependence of meteorological variables. For example, Al Kajbaf et al. (2022) downscaled the precipitation in climate models based on multiple machine learning models. In Jennings et al. (2010), the master-target scaling approach is used to improve the temporal resolution of precipitation data. Nguyen and Chen (2022) used Copula and Modified Huff Rainfall Curves to achieve high temporal resolution meteorological data reconstruction. Additionally, based on the weighted random weather generator, Fatichi et al. (2013) downscaled meteorological variables, include precipitation, temperature, and wind speed at different time scales. In Hou et al. (2017), they use support vector machines and Markov chain to perform temporal downscaling of Daily Precipitation. Li (2019) integrates autoencoder networks, XGBoost, and Random Forest for spatiotemporal downscaling of wind speed. Compared to interpolation-based methods, machine learning methods are more likely to capture the relative temporal nonlinear dependence of meteorological variables due to their ability to fit nonlinear features. However, limited by their shallow structure, their fewer learnable parameters make them unable

to accurately fit the relative temporal evolution patterns when facing massive data. And since machine learning methods usually consider only single or several spatial observation points, it is difficult to model the global spatial dependence.

With the development of global meteorological big data, deep learning-based methods have been widely studied and applied in meteorological prediction and modeling (Gao et al., 2024; Wang et al., 2021; Zhang et al., 2024; Gao et al., 2025). Among them, the convolution-based deep learning model significantly improves the local spatial dependence learning ability through convolutional computation, while the deep model structure brings a strong ability to fit complex nonlinear features. Serifi et al. (2021) proposed a U-Net algorithm for temporal downscaling and improved the time resolution of meteorological data. Wang et al. (2024) proposed an enhanced residual U-Net for time downscaling the meteorological data. In the study of Xiang et al. (2022), the transformer model is used for time downscaling of 2-m temperature and 10-m wind speed. These methods take into account the local spatial dependence of meteorological variables by introducing convolutional computation and realize the fusion of temporal boundary information by simultaneously dealing with the time boundary information. Experimental results prove that their downscaling accuracy is better than that of the machine learning methods, but the local receptive field still restricts these methods from learning the global spatial dependence of meteorological data. Moreover, limited by the end-to-end data-driven training mode, convolution-based deep learning methods lack the ability to extrapolate the state outside the data distribution, i.e., the unseen moments during the training process. In recent years, flow-based deep learning methods have realized data distribution extrapolation by modeling the spatial evolution within the time interval (Lee et al., 2020; Xu et al., 2022; Zhang et al., 2023). Such as Tatsubori et al. (2022) treating meteorological spatiotemporal data as videos and using optical flow methods to estimate the meteorological evolution process. In Chen et al. (2022), the Zooming Slow-Mo algorithm is improved to obtain the evolution estimation of meteorological data. Although these works demonstrate the feasibility of flow-based deep learning methods in solving the problem of data distribution extrapolation. The flow-based method assumes that the temporal resolution of the original series is less than 1/24 s, and the spatial motion is tiny in the interval, which can be approximated as a linear change. In contrast, the temporal boundaries of meteorological data are often higher than 1 h, and the spatial scales of meteorological state evolution are large and do not obey the linearity assumption. Secondly, flow-based methods assume the meteorological state is spatially constant within the interval, whereas the meteorological variable is of relative temporal randomness. Therefore, flow-based deep learning methods are difficult in learning the nonlinear temporal dependence and complex spatial flow in the temporal downscaling tasks, which prevents these methods from achieving high downscaling accuracy.

In general, although current methods try to solve the above challenges faced by temporal downscaling, they only try to solve the single or serval problems among local spatial dependence, relative temporal nonlinearity, and data distribution extrapolation from a one-sided perspective, but neglect the comprehensive consideration of the above challenges faced by temporal downscaling. Moreover, few works have considered the problems of multi-source information fusion, global spatial dependence, and absolute time dependence in temporal downscaling, resulting in insufficient downscaling accuracy and limited output temporal resolution of current temporal downscaling studies, which is difficult to satisfy the needs of engineering activities for high-precision and high temporal resolution meteorological data.

To solve the above problems, this study proposes a Multi-source Spatial-temporal-wavelet feature Fusion and Time-Continuous Manifold-based Algorithm, MSF-TCMA, for meteorological data temporal downscaling. MSF-TCMA consists of a multi-scale depth-wavelet feature extraction branch, a cross-modal spatial-temporal information fusion branch, and the time-continuous manifold sampling branch.

Among them, the multiscale depth-wavelet feature extraction branch extracts multiscale spatial features of heterogeneous meteorological data from the convolutional domain and wavelet domain, respectively, and then integrates the spatial features to fully learn the spatial dependencies. The cross-modal spatiotemporal information fusion branch firstly fuses the meteorological and satellite observation information, then integrates the absolute time information and the relative time information, and finally, it integrates the boundary meteorological state to construct the time-continuous manifold. The time-continuous manifold sampling branch samples the manifold at the moment to be solved, and maps it to the meteorological state to realize the data distribution extrapolation for continuous downscaling. The main contributions of this paper are as follows:

- **High temporal-resolution meteorological state extrapolation algorithm:** MSF-TCMA maps the nonlinear meteorological state evolution into a linear time-continuous manifold space, and obtains the meteorological state at any moment in the time interval by interpolating the manifold, which overcomes the constraint of data distribution extrapolation in the temporal downscaling task and realizes the continuous temporal downscaling with high temporal resolution.
- **A spatiotemporal and multisource information fusion paradigm for high-accuracy temporal downscaling:** MSF-TCMA extracts and integrates the multi-scale spatial features from multi-source data through the multi-scale depth-wavelet feature extraction branch. And fuses the cross-modal information, and the relative and absolute time dependencies by cross-modal spatial-temporal information fusion branch. This solves the problem of inadequate spatiotemporal dependencies learning and insufficient multi-source data fusion.
- **Multi-Moment Weighted Meteorological State Estimation-Energy Change Deduction Joint Loss:** During the algorithm training stage, the proposed composite loss function combines both the meteorological state estimation loss and energy change deduction loss to jointly optimize the temporal downscaling accuracy. Additionally, the moment-weighting mechanism is introduced to enhance the performance of multi-moment downscaling.
- Experiments based on public datasets demonstrate the anchor moment temporal downscaling accuracy of MSF-TCMA and analyze the contribution of multisource data to the downscaling task. Two engineering application experiments validate the effectiveness of MSF-TCMA in the multi-moments (including unseen moments) continuous temporal downscaling task and future multi-moment downscaling tasks, respectively.

The rest of this article is organized as follows: In Section 2, we introduce the principle of time-continuous manifold-based data distribution extrapolation and the architecture of MSF-TCMA. Section 3 introduces the study area, dataset, evaluation metric, experimental, and analysis. Section 4 concludes this paper.

## 2. Proposed algorithm

### 2.1. Time-continuous manifold for data distribution extrapolation

As the spatiotemporal series, the meteorological state  $M_t$  at moment  $t$  in the spatial region is expressed as:

$$M_t = \Phi(t) \quad (1)$$

where  $\Phi(\cdot)$  describes the evolution of meteorological states over time.

Meteorological products provide meteorological states with a specified time resolution by sampling the  $\Phi(\cdot)$ . For example, the time resolution of GFS (contributor, 2015) is 6 h. In the temporal downscaling task, to obtain meteorological states with higher resolution, it is necessary to model the evolution of meteorological variables within

the 6 h. For each 6-h interval, the meteorological states at the start and the end times provided by the meteorological product are called the temporal boundary meteorological states, denoted as  $M_0$  and  $M_1$ .

Temporal downscaling attempts to establish a mapping mode  $\varphi(\cdot)$  to estimate the meteorological state  $M_t$  based on the boundary meteorological states, where  $t \in (0, 1)$ . To make the estimated value  $\hat{M}_t$  in this mode has the minimum error with  $M_t$ :

$$\begin{cases} \underset{\varphi}{\operatorname{argmin}} \mathcal{L}(\hat{M}_t, M_t) \\ \hat{M}_t = \varphi(M_0, M_1, t) \end{cases} \quad (2)$$

The linear interpolation-based method regards  $\Phi(\cdot)$  in the time interval as a linear model. In this method, the estimated meteorological state  $\hat{M}_t$  is expressed as

$$\hat{M}_t = (1-t)M_0 + tM_1 \quad (3)$$

However, as described in Section 1, the evolution of meteorological variables does not satisfy the linear mode, resulting in large estimation errors. In this study, we aim to find a mapping pattern  $\varphi_w(\cdot)$  to map the meteorological evolution  $\Phi(\cdot)$  to the time-continuous manifold  $\mathcal{F}$ , in which the latent features evolve linearly. Therefore, the latent feature  $F_t$  at moment  $t$  can be obtained by linear interpolation of the manifold, then mapped  $F_t$  back to the meteorological state space by inverse transformation to obtain an estimation of the meteorological state  $M_t$ . The simple form of this process can be represented as:

$$\begin{cases} F_0 = \varphi_w(M_0) \\ F_1 = \varphi_w(M_1) \end{cases} \quad (4)$$

$$F_t = (1-t)F_0 + tF_1 \quad (5)$$

$$\hat{M}_t = \varphi_w^{-1}(F_t) \quad (6)$$

Rewrite Eq. (5), let  $F_{0 \rightarrow 1} = F_1 - F_0$ ,  $F_{1 \rightarrow 0} = F_0 - F_1$ , we have:

$$\begin{cases} F_t = tF_{0 \rightarrow 1} + F_0 \\ F_t = (1-t)F_{1 \rightarrow 0} + F_1 \end{cases} \quad (7)$$

Combining the two equations, we have:

$$F_t = \alpha(tF_{0 \rightarrow 1} + (1-t)F_{1 \rightarrow 0}) + \beta \quad (8)$$

where  $F_{0 \rightarrow 1}$ ,  $F_{1 \rightarrow 0}$  are the forward and the backward evolution estimation,  $\alpha$  is a constant, and  $\beta$  depends on  $F_0$ ,  $F_1$ .

As described in Eq. (8),  $F_t$  is obtained by linearly interpolating the front and back boundary features  $F_0$  and  $F_1$  of the temporal interval. Therefore, it is possible to obtain the latent features at any moment within the interval and then derive the meteorological state  $\hat{M}_t$  at that moment through inverse transformation, thereby enabling continuous temporal downscaling.

## 2.2. Overview

**Fig. 1** illustrates the overall framework of the proposed temporal downscaling method. In the data acquisition and preprocessing stages, raw meteorological data and satellite remote sensing data are first obtained from meteorological data suppliers and satellite data platforms, respectively. Concurrently, the timestamp code is generated based on the timestamp (month, day, and hour) of the acquired samples. To build the original multisource dataset. Subsequently, for different meteorological variable downscaling tasks, the meteorological variables and satellite observation channels of interest are extracted, and the data are spatially sampled according to the study area. Finally, the heterogeneous data are aligned by their timestamp to achieve cross-modal spatiotemporal alignment.

In the algorithm construction and training stage, the proposed MSF-TCMA is initialized, and training samples are extracted for algorithm optimization. This paper proposed to jointly optimize the algorithm

using the meteorological state estimation loss  $\mathcal{L}_{Est}$  and the energy change deduction loss  $\mathcal{L}_{dEn}$  to accurately estimate the meteorological state and meteorological state evolution within the time interval. For the continuous temporal downscaling task, the moment-weighting mechanism is proposed to weigh the optimization tasks of different moments.

Finally, the test set is used to evaluate the performance of the trained MSF-TCMA in the temporal downscaling task. The details of the proposed method will be discussed in the following subsections.

## 2.3. Architecture of proposed MSF-TCMA

This section provides a detailed description of the proposed MSF-TCMA, its architecture is shown within the gray box in **Fig. 1**. It consists of three main components: the multiscale deep-wavelet feature extraction branch, the information fusion branch, and the manifold sampling branch. These branches are responsible for spatial feature extraction of meteorological and satellite data at the time boundary, multimodal fusion and temporal dependence integrating, and estimation the meteorological states at specific moments. By extracting multiscale deep-wavelet features, fusing information from heterogeneous spatiotemporal data, as well as jointly optimizing the objectives, the proposed MSF-TCMA achieves high-precision estimation of meteorological states within the temporal interval.

### 2.3.1. MultiScale deep-wavelet feature extraction branch

The multiscale deep-wavelet feature extraction branch is proposed based on the spatial characteristic of meteorological variables. Meteorological variables exhibit complex spatial variation characteristics, specifically, the meteorological variables of a single observation point are highly correlated with its surrounding states, in different spatial scales, the dependency varies. Therefore, the feature extraction branch uses two feature streams: wavelet and convolution, to fully capture the multiscale local spatial dependency, then integrates the global multiscale feature by multi-scale feature extraction and fusion mechanism.

**Fig. 2(a)** illustrates the structure of the multiscale deep-wavelet feature extraction branch. It consists of two independent feature extractors, designed to process meteorological data and satellite data at the time boundaries, respectively. In each feature extractor, a convolutional feature extraction layer and a wavelet feature extraction layer process the input data to obtain the depth features and wavelet features. Concatenate these features to fuse the Deep-wavelet information. Then, Res2Blocks (Gao et al., 2021) are used for multiscale feature extraction. Finally, the multiscale feature fusion layer integrates the multiscale deep features from different layers.

The structure of the wavelet feature extractor is shown in **Fig. 2(b)**, where the db1 wavelet (Wang et al., 2006) is used to perform a two-level the feature map is halved after the wavelet transform, the deconvolution-BN-ReLU structure is used to upsample the feature map. Finally, the multi-scale wavelet feature fusion is realized by channel concatenation. The calculation can be represented as:

$$\begin{cases} X_{wave} = X_{wave}^1 \oplus X_{wave}^2 \\ X_{wave}^{l+1} = \text{TConv}(\text{WAVELET}_{db1}(X_{wave}^l)) \end{cases} \quad (9)$$

where  $\oplus$  presents channel concatenation,  $\text{WAVELET}_{db1}(\cdot)$  represents the two-dimensional discrete wavelet transformation by db1 wavelet.  $\text{TConv}(\cdot)$  represents the deconvolution layer. The choice of the db1 wavelet is motivated by the characteristics of the studied region, where the meteorological fields are represented on relatively sparse grids ( $64 \times 64$  over a  $15.75^\circ \times 15.75^\circ$  area at  $0.25^\circ$  resolution). In such cases, retaining low-frequency information is crucial for preserving the integrity of large-scale spatial structures. Accordingly, the wavelet feature extractor is primarily designed to capture coarse spatial components of the input. To supplement high-frequency information, a convolutional feature extractor is employed in parallel. By combining wavelet-based

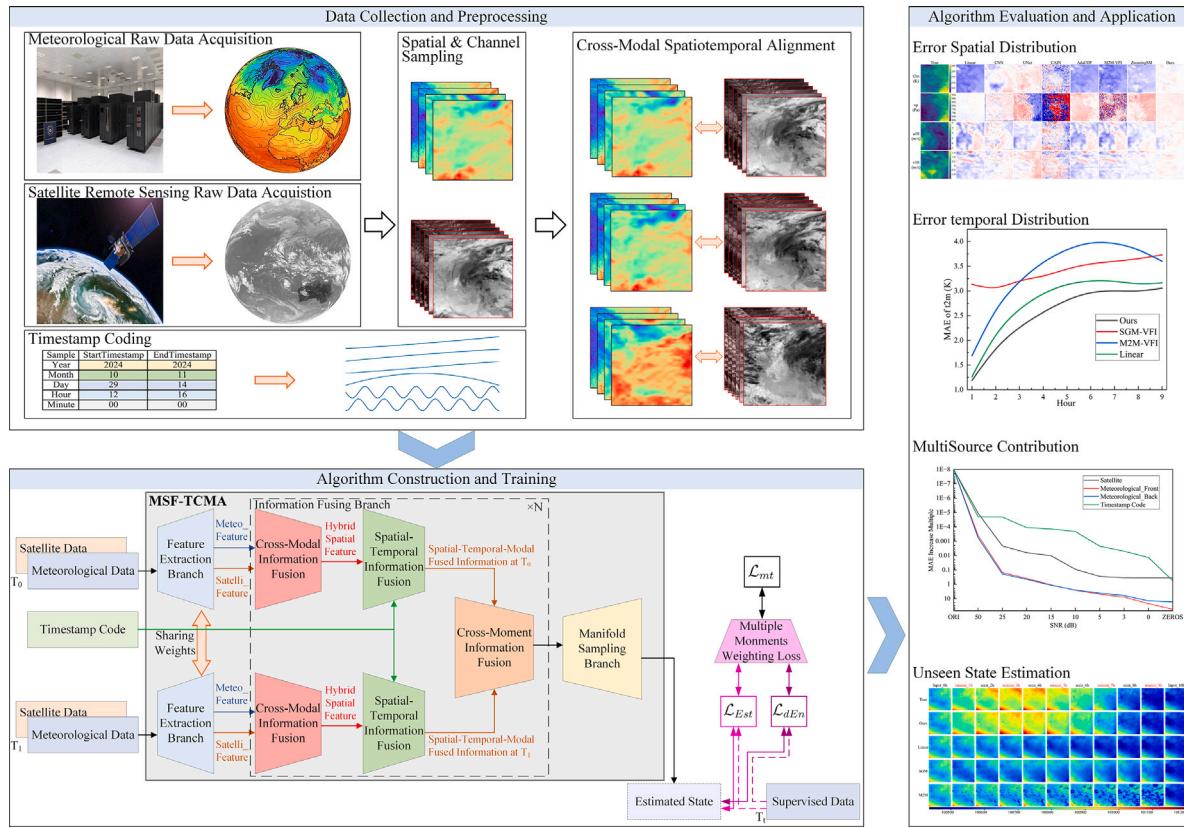


Fig. 1. The overall framework of the proposed temporal downscaling method.

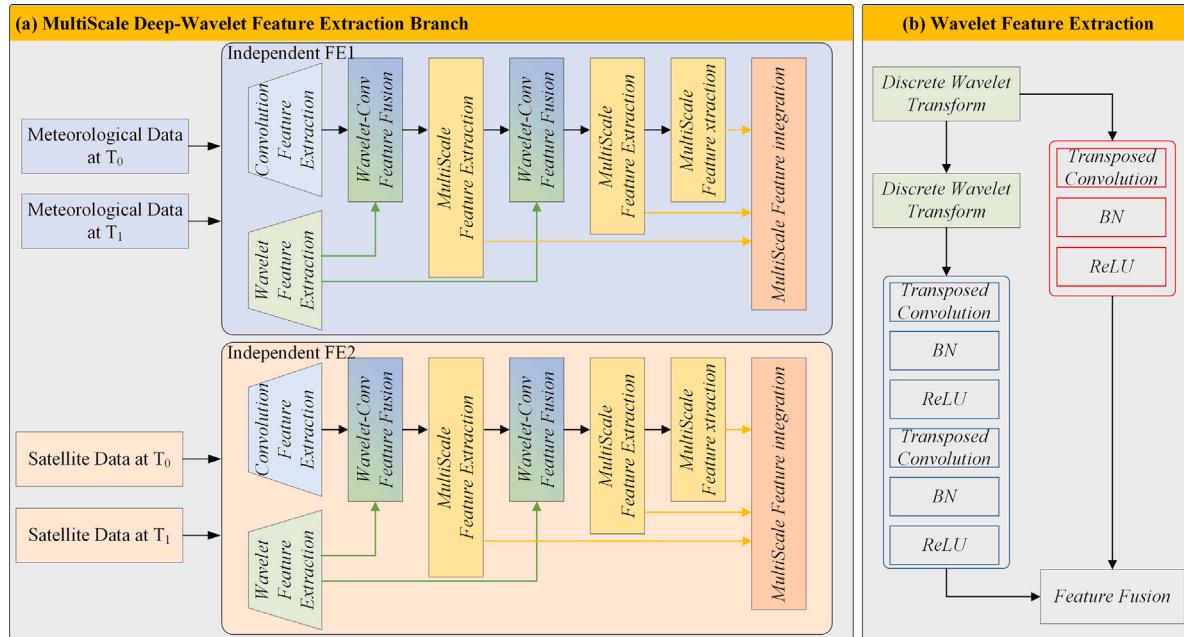


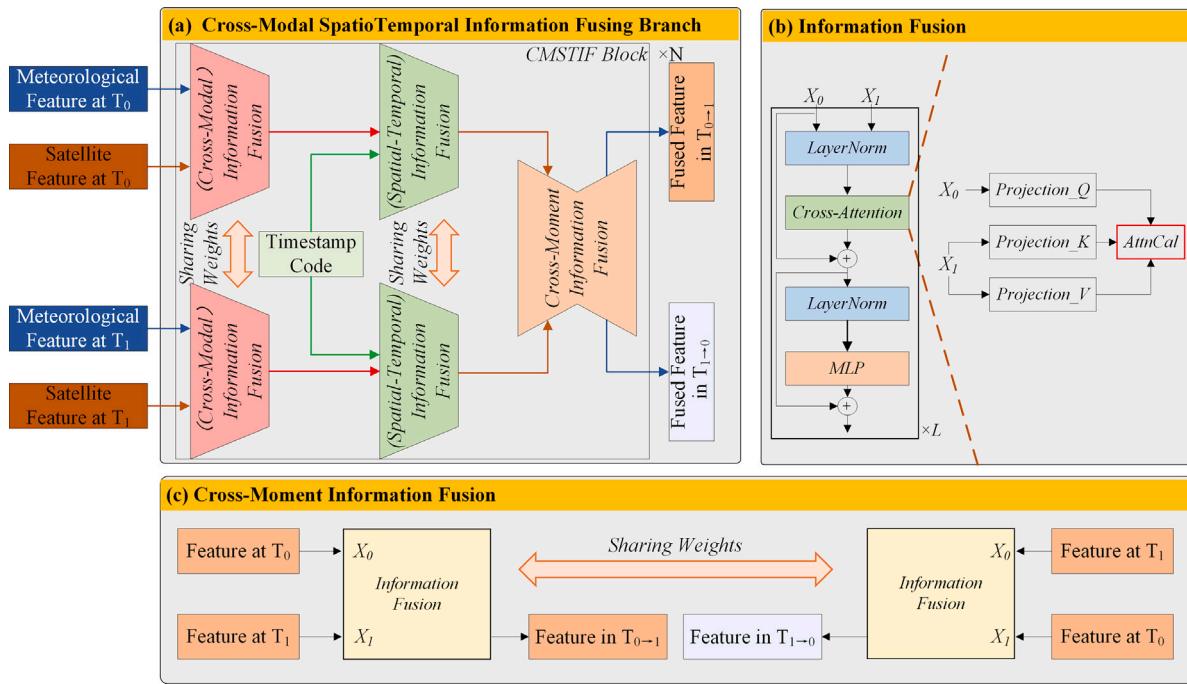
Fig. 2. The structure of the multiscale deep-wavelet feature extraction branch.

low-frequency modeling with convolution-based high-frequency enhancement, the model achieves a balanced representation of both the global background and localized variability within meteorological fields.

The multi-scale feature extraction layer is realized by Res2Block. Its structure is not described in detail here. Its calculation can be

represented as:

$$X_{MFB}^l = \begin{cases} \text{Res2Block}^l(X_{wave} \oplus X_{conv}), l = 1 \\ \text{Res2Block}^l(X_{wave} \oplus X_{MFB}^{l-1}), l = 2 \\ \text{Res2Block}^l(X_{MFB}^{l-1}), l \geq 3 \end{cases} \quad (10)$$



**Fig. 3.** The structure of the cross-modal spatiotemporal information fusion branch.

The multi-scale feature integration layer first downsamples the deep features from different layers by a convolution layer, then there is a channel attention layer for channel weighting. Finally, a single convolution layer is used to integrate the multi-layer features. The calculation can be represented as:

$$X_{MFI} = \text{Conv}(\hat{X}_{MFB}) \quad (11)$$

$$\hat{X}_{MFB} = \text{CA}\left(\sum_{\oplus} \text{Conv}(X_{MFB}^l)\right) \quad (12)$$

where  $\sum_{\oplus}(\cdot)$  represents channel concatenate from  $l = 1$  to  $L$ .  $\text{CA}(\cdot)$  is the channel attention calculation, which is realized by global average pooling of the input feature map and a two-layer neural network.

Use the proposed multiscale deep-wavelet feature extraction branch to extract the features from the initial input of the algorithm: meteorological data and satellite data at the boundary time. Note that independent feature extractors are used for heterogeneous data, and feature extractors for homologous data share weights.

### 2.3.2. Cross-modal spatiotemporal information fusing branch

The information fusing branch is proposed for fusing the deep features. Specifically, considering the contribution of different data sources to the downscaling task, the cross-modal information fusion mechanism is used to integrate heterogeneous information. Considering the deterministic correlation of meteorological state evolution over absolute time, the spatial-temporal information fusion mechanism is used to integrate the absolute time information. Finally, the cross-moment information fusion mechanism learns the relative temporal dependency and builds the time-continuous manifolds. The structure of the cross-modal spatiotemporal information fusion branch is shown in Fig. 3(a), which is composed of  $N$  Cross-Modal Spatial-Temporal Information Fusion blocks (CMSTIF blocks). Each CMSTIF block begins with the cross-modal information fusion layer, which is responsible for fusing meteorological features and satellite features. Following this, the spatial-temporal information fusion layer integrates static spatial features with temporal information. Finally, the cross-moment information fusion layer fuses the boundary features, enabling the branch to obtain the temporal evolution information within the time interval.

**Fig. 3(b)** shows the structure of the cross-modal information fusion layer and the spatial-temporal information fusion layer. They are implemented by cross-attention calculation. The calculation process can be represented as:

$$X_{fuse} = \sigma\left(\frac{QK}{\sqrt{d}}\right)V \quad (13)$$

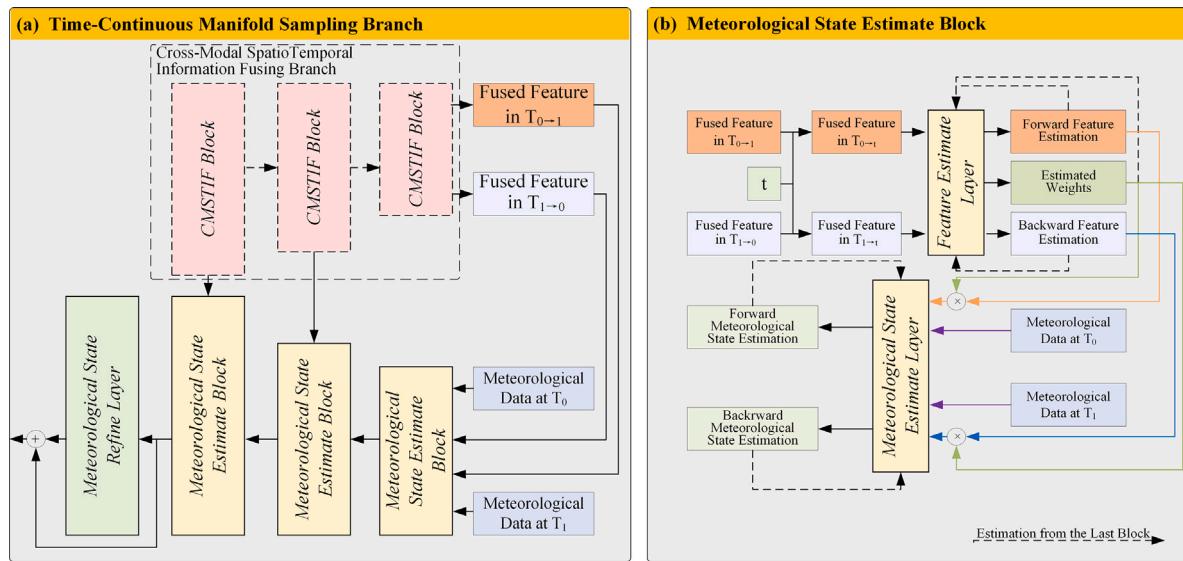
$$\begin{cases} Q = \text{Proj}_q(X_0) \\ K, V = \text{Proj}_{k,v}(X_1) \end{cases} \quad (14)$$

where  $\sigma(\cdot)$  is the Softmax function,  $\text{Proj}(\cdot)$  is implemented by a single-layer perceptron. Note that in the cross-modal information fusion layer, the meteorological data is taken as  $X_0$ , and the satellite data is  $X_1$ . In the spatial-temporal information fusion layer, the fused feature is  $X_0$ , and the timestamp code is  $X_1$ .

The input of the spatial-temporal information fusion layer is the timestamp code, which decomposes the time information of the time boundary into month, day, and hour. Calculate the month, day, and hour differences between the front and back time boundaries, and embed them in the sine and cosine functions with a period of 12 months, the number of days in the month, and 24 h respectively. The calculation process can be represented as:

$$\begin{cases} \text{code} = \text{concat}(t_m^1, t_m^2, t_d^1, t_d^2, t_h^1, t_h^2) \\ t_m^1 = \sin\left(\frac{2\pi t}{12}\right), t \in (m_1, m_2) \\ t_m^2 = \cos\left(\frac{2\pi t}{12}\right), t \in (m_1, m_2) \\ t_d^1 = \sin\left(\frac{2\pi t}{\text{days}}\right), t \in (d_1, d_2) \\ t_d^2 = \cos\left(\frac{2\pi t}{\text{days}}\right), t \in (d_1, d_2) \\ t_h^1 = \sin\left(\frac{2\pi t}{24}\right), t \in (h_1, h_2) \\ t_h^2 = \cos\left(\frac{2\pi t}{24}\right), t \in (h_1, h_2) \end{cases} \quad (15)$$

The structure of the cross-moment information fusion layer is shown in Fig. 3(c). Its core is based on the information fusion layer illustrated



**Fig. 4.** The structure of the time-continuous manifold sampling branch.

in Fig. 3(b). The key difference is that it employs two information fusion layers with shared weights to compute the forward and backward feature evolution respectively. Specifically, when the feature at  $T_0$  is used as  $X_0$ , and the feature at  $T_1$  is used as  $X_1$ , its output is the forward feature evolution  $F_{0 \rightarrow 1}$ . Conversely, when the feature at  $T_1$  is used as  $X_0$ , and the feature at  $T_0$  is used as  $X_1$ , the output represents the backward feature evolution  $F_{1 \rightarrow 0}$ .

It should be noted that the calculation described above applies to the first CMSTIF block. For subsequent blocks, the forward and backward feature evolution from the previous block is used to replace the meteorological features at the time boundaries.

### 2.3.3. Time-continuous manifold sampling branch

Building upon the principle of time-continuous manifold for data distribution extrapolation, the manifold sampling branch is used to estimate the latent feature at any given moment within the temporal interval. This estimated feature is then used to infer the corresponding meteorological state, ensuring the high-resolution reconstruction of meteorological evolution over time.

The structure of the time-continuous manifold sampling branch is shown in Fig. 4(a). It utilizes multiple meteorological state estimate blocks to form a U-like structure in conjunction with the cross-modal spatial-temporal information fusion branch. By fusing the evolved features from different CMSTIF blocks and incorporating the refine layer, the branch ultimately provides an estimation of the meteorological state at the given moment  $t$ .

The computational principle of the meteorological state estimation block is shown in Fig. 4(b). First, the feature estimation layer interpolates the forward and backward feature evolution by  $t$ , and obtains the feature evolution from the front and back time boundaries to the moment respectively. The feature estimation layer then combines the feature evolution and the feature estimation of its previous layer (if applicable) to obtain the feature estimates from the front and back time boundaries to the  $t$  moment, along with the corresponding combined weights of the two estimates. Subsequently, the meteorological state estimation layer integrates the feature estimation at  $t$ , the boundary meteorological state, and the forward and backward meteorological state estimation of its previous layer (if applicable) to obtain the meteorological state estimation at  $t$  moment. Finally, the final meteorological state estimation is obtained by computing the weighted sum of the

estimates. The computational process is represented as:

$$\begin{cases} F_{0 \rightarrow t}^l, F_{1 \rightarrow t}^l = t F_{0 \rightarrow 1}^l, (1-t) F_{1 \rightarrow 0}^l \\ F_{tF}^{l+1}, F_{tB}^{l+1}, w^{l+1} = \text{FEL}(F_{0 \rightarrow t}, F_{1 \rightarrow t}, F_{tF}^l, F_{tB}^l, w^l) \\ M_{tF}^{l+1}, M_{tB}^{l+1} = \text{MSEL}(M_0, M_1, w^{l+1} M_{tF}^l, (1-w^{l+1}) M_{tB}^l) \\ \hat{M}_t^l = w^l M_{tF}^l + (1-w^l) M_{tB}^l \\ \hat{M}_t = \hat{M}_t^l + \text{MSRL}(\hat{M}_t^l) \end{cases} \quad (16)$$

where  $\text{FEL}(\cdot)$ ,  $\text{MSEL}(\cdot)$ , and  $\text{MSRL}(\cdot)$  are the feature estimation layer, meteorological state estimation layer, and Refine layer respectively. They are all three-layer convolutional network structures. The alignment of the feature map is achieved via Pixelshuffle upsampling.

### 2.4. Algorithm optimization

We propose a composite optimization objective consisting of two components: the meteorological state estimation loss  $\mathcal{L}_{Est}$  and the energy change deduction loss  $\mathcal{L}_{dEn}$ . This composite objective ensures that the algorithm accurately estimates the meteorological state at single-moment  $t$  and the energy change from the time boundary to  $t$ .

The meteorological state estimation loss  $\mathcal{L}_{Est}$  describes the numerical error between the actual meteorological state and the estimated meteorological state, computed as their L1 loss:

$$\mathcal{L}_{Est} = \|\hat{M}_t - M_t\| \quad (17)$$

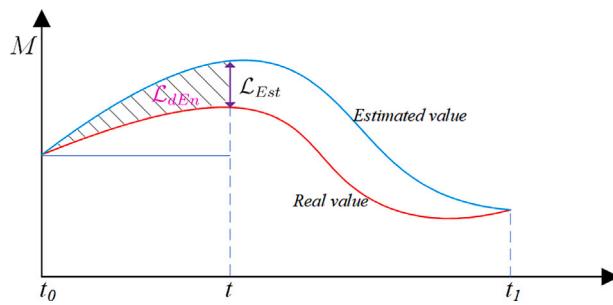
The energy change deduction loss  $\mathcal{L}_{dEn}$  describes the estimated error in the energy change of the meteorological state from the time boundary to the  $t$  moment. It is calculated as:

$$\mathcal{L}_{dEn} = \|(\hat{M}_t - M_0)^2, (M_t - M_0)^2\| + \|(\hat{M}_t - M_1)^2, (M_t - M_1)^2\| \quad (18)$$

where the first term describes the energy change from the front boundary to the  $t$  moment, and the second term represents the change from  $t$  moment to the back boundary. Here,  $(M_t - M_0)^2$  describes the perturbation energy change of the true meteorological state at  $t$  moment relative to the initial state  $M_0$ , while  $(\hat{M}_t - M_0)^2$  denotes the estimated perturbation energy change. Calculating their L1 loss as the optimal objective to constrain the algorithm in learning the energy evolution. The principle is shown in Fig. 5.

Therefore, the optimization objective of the single-moment downscaling task can be expressed as:

$$\mathcal{L}_t = \mathcal{L}_{Est} + \mathcal{L}_{dEn} \quad (19)$$



**Fig. 5.** The principle of the meteorological state estimation loss energy change deduction loss.

To ensure that the latent features at different moments within the interval follow a linear evolution pattern on the time-continuous manifold, multiple meteorological states within the interval are sampled to construct a multi-moment composite loss as a supervisory constraint for algorithm optimization. Meanwhile, considering that the meteorological state near the time boundary has a strong correlation with the boundary state. As the distance from the time boundary increases, the correlation between the meteorological state and the boundary state also weakens. Therefore, the time weighting coefficient is used to weigh the estimation loss at different moments. Given that real meteorological states are available for  $T$  moments as supervision, the weighting coefficient for each moment  $t$  is defined as:

$$w_t = \frac{e^{\sin(\pi t/T)}}{\sum_i e^{\sin(\pi i/T)}} \quad (20)$$

Finally, for the multi-moment task, the optimization objective is:

$$\mathcal{L}_{mt} = \sum_{t=t_0}^T w_t \mathcal{L}_t \propto \sum_{t=t_0}^T w_t L(t F_{0 \rightarrow 1} + (1-t) F_{1 \rightarrow 0}) \quad (21)$$

where both  $F_{0 \rightarrow 1}$  and  $F_{1 \rightarrow 0}$  are produced by the algorithm. By computing the losses at different moments within the interval as the optimization objective, the  $F_{0 \rightarrow 1}$  and  $F_{1 \rightarrow 0}$  are constrained to evolve linearly within the time interval. This enables accurate estimation of the latent features at any moment through linear interpolation.

### 3. Experiment

#### 3.1. Data description

##### 3.1.1. Dataset 1: ERA5 reanalysis meteorological dataset on northern Indian Ocean

This study takes the northern Indian Ocean as the research area, with a latitude and longitude range of  $-15.75^\circ$  S– $0^\circ$ ,  $80$ – $95.75^\circ$  E. The spatial area involved is shown in Fig. 6(a), where the red box indicates the study area.

In this study, the meteorological data comes from the ERA5 reanalysis data product (Hersbach et al., 2023), which has a spatial resolution of  $0.25^\circ$  and provides global meteorological data from 1941 to the present at a temporal resolution of 1 h. We selected 2m temperature (t2m), surface pressure (sp), 10m meridional wind speed (u10), and 10m zonal wind speed (v10) as the meteorological variables of the study to construct the meteorological data set.

##### 3.1.2. Dataset 2: ERA5-land reanalysis meteorological dataset on southwestern China

This study takes southwestern China as the main research area, the latitude and longitude range is  $31.5$ – $37.8^\circ$ N,  $100$ – $106.3^\circ$ E. The spatial area involved is shown in Fig. 6(b), where the red box indicates the study area.

In this study, the meteorological data is acquired from the ERA5-land (Muñoz-Sabater et al., 2021) reanalysis data product, which has a spatial resolution of  $0.1^\circ$  and a temporal resolution of 1 h. We also take t2m, sp, u10, and v10 as the meteorological variables to form the meteorological dataset.

#### 3.1.3. FY-4B satellite remote sensing dataset

The satellite observation data comes from the FY-4B meteorological satellite (Min et al., 2017), which provides full-disk earth observation data centered at  $105^\circ$  E with a spatial resolution of 4 km and a temporal resolution of 15 min. Six spectral channels are selected to form the satellite observation data, with spectrum bands of  $0.47\text{ }\mu\text{m}$ ,  $0.65\text{ }\mu\text{m}$ ,  $0.825\text{ }\mu\text{m}$ ,  $2.225\text{ }\mu\text{m}$ ,  $6.95\text{ }\mu\text{m}$ , and  $10.80\text{ }\mu\text{m}$ . The observation data were transformed into longitude and latitude coordinate system by projection, and spatial sampling the data to  $0.25^\circ \times 0.25^\circ$  and  $0.1^\circ \times 0.1^\circ$  resolutions in the study areas of data 1 and 2, respectively, to align heterogeneous observation data.

### 3.2. Evaluation metrics

This paper evaluates the effectiveness of the temporal downscaling methods from three perspectives: numerical accuracy, spatial distribution similarity, and correctness of evolution direction. The numerical accuracy is described by Mean Absolute Error (MAE) and Root Mean Square Error (RMSE):

$$MAE = \frac{1}{LAT \times LON} \sum_{i,j}^{LAT, LON} |\hat{M}_t(i, j) - M_t(i, j)| \quad (22)$$

$$RMSE = \sqrt{\frac{1}{LAT \times LON} \sum_{i,j}^{LAT, LON} (\hat{M}_t(i, j) - M_t(i, j))^2} \quad (23)$$

Numerical accuracy describes the extent to which the outputs of the downscaling method deviate from the actual meteorological field in terms of numerical values. Specifically, MAE measures the average numerical error between the estimated and true field, while RMSE emphasizes extreme numerical errors. Lower values for both metrics indicate more accurate downscaling performance.

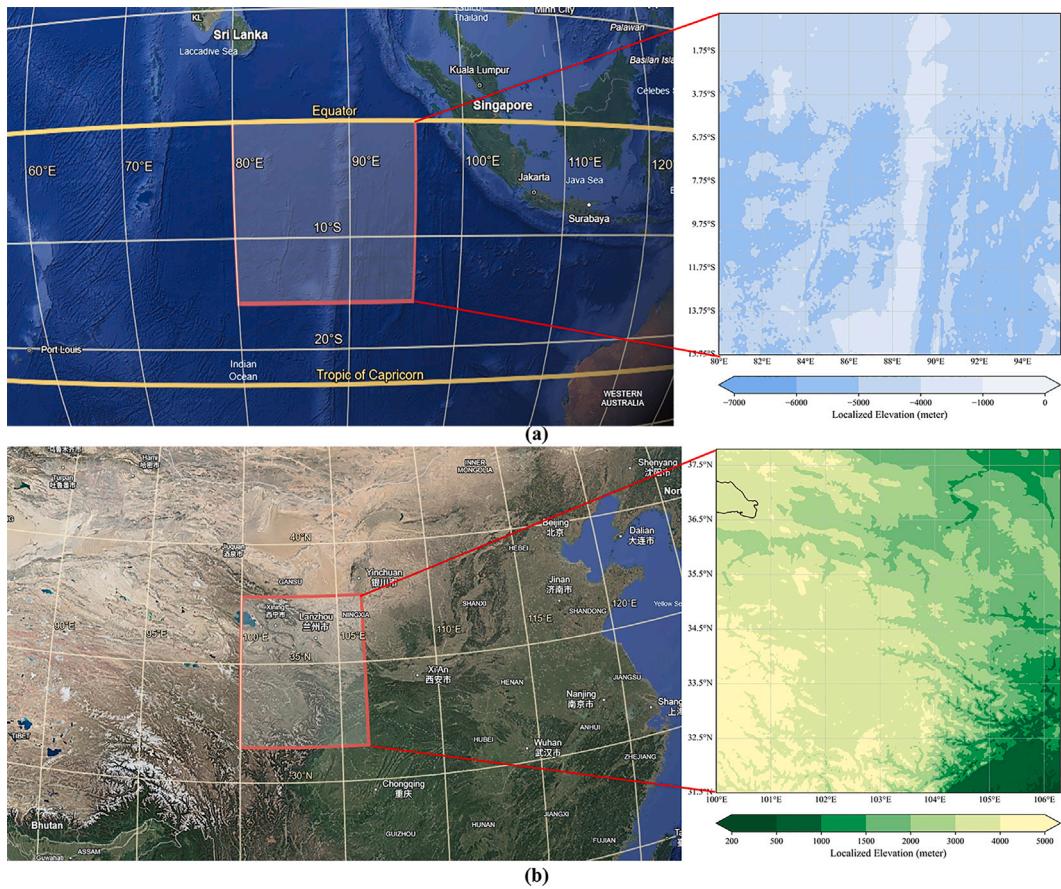
In addition to the numerical accuracy of the estimated meteorological state, we also use Anomaly Correlation Coefficient (ACC) and Peak Signal-to-Noise Ratio (PSNR) to evaluate the similarity in spatial distribution between the estimated and the actual meteorological field. The calculation is represented as:

$$\left\{ \begin{array}{l} ACC = \frac{(\hat{M}'_t - \bar{M}'_t)(M'_t - \bar{M}'_t)}{S_{\hat{M}'_t} S_{M'_t}} \\ \hat{M}'_t = \hat{M}_t - \bar{M}_t \\ S_{\hat{M}'_t} = (\hat{M}'_t - \bar{M}'_t)^2 \end{array} \right. \quad (24)$$

$$PSNR = 10 \lg \frac{1}{\sum_{i,j}^{LAT, LON} (\text{Normal}(\hat{M}_t(i, j)) - \text{Normal}(M_t(i, j)))^2} \quad (25)$$

where  $\text{Normal}(\cdot)$  represents the normalization. ACC measures the degree of spatial distribution matching between the estimated field and the true field, while PSNR reflects the consistency of spatial details and textures between the estimated field and the actual field. Higher values of both metrics indicate better downscaling performance.

In addition, we propose to evaluate the matching degree between the algorithm estimation results and the actual meteorological field in the evolution direction over time by evolution direction accuracy. The



**Fig. 6.** The research area of dataset 1 and dataset 2.

calculation is represented as:

$$\left\{ \begin{array}{l} EDA = \frac{D_{0t} + D_{t1}}{2 \times LAT \times LON} \\ LAT, LON \\ D_{0t} = \sum_{i,j} [\hat{M}_t(i,j) > M_0(i,j)] \odot [M_t(i,j) > M_0(i,j)] \\ D_{t1} = \sum_{i,j} [\hat{M}_t(i,j) > M_1(i,j)] \odot [M_t(i,j) > M_1(i,j)] \end{array} \right. \quad (26)$$

where  $\odot$  is the XNOR operation. EDA evaluates the ability of the algorithm to accurately capture the evolution trend of meteorological variables within the temporal interval. It emphasizes whether the direction of numerical changes is correct, rather than the magnitude of the errors, thereby supplementing numerical accuracy metrics by assessing aspects of dynamic process rationality that they cannot reveal. A higher EDA value indicates that the algorithm is more effective in characterizing the evolution trend of meteorological variables within the interval.

### 3.3. Comparison results and analysis on single moment downscaling tasks

To evaluate the performance of the proposed method in temporal downscaling tasks, we conducted comparative experiments with three types of algorithms.

- **Interpolation-based algorithm:** we used a linear interpolation method, simplifying the meteorological state evolution process into a linear model, and estimated the meteorological state at t moment by temporal interpolation for reference.
- **CNN-based downscaling algorithm:** We selected general models including U-Net (Ronneberger et al., 2015; Wang et al., 2024),

and a four-layer CNN to learn the spatial dependency, and directly build the mapping pattern from the low-temporal resolution input to the high-temporal resolution output as baselines.

- **Flow-based downscaling algorithm:** Including EMA-VFI (Zhang et al., 2023), which estimates optical flow through inter-frame attention, including RIFE (Huang et al., 2022), which uses a block-based deep learning model to learn the inter-frame optical flow. M2M-VFI (Hu et al., 2022), which interpolates the frame within the time interval by estimating bidirectional flow. And SGM (Liu et al., 2024), which captures the large spatial motion by sparse global matching. And TemDeep (Wang et al., 2025), which employs two encoder-decoder structures with multiple convolutional layers to estimate the meteorological states and flow, respectively, thereby enabling temporal downscaling.
- **Other SOTA methods:** Including CAIN (Choi et al., 2020), which introduces CNN to capture the local spatial dependency, and use channel attention to integrate the multi-variable information. SWinIR (Liang et al., 2021; Xiang et al., 2022), builds an end-to-end network based on transformer architecture to capture the local and global spatial dependence. AdaCOF (Lee et al., 2020) uses adaptive convolution for complex spatial dependency extraction and is used for complex motion estimation. And zooming slow-mo (Chen et al., 2022; Xiang et al., 2020), which uses deformable ConvLSTM to improve the effect of learning temporal dependency. And SpateGAN (Glawion et al., 2025), which performs spatiotemporal downscaling using 3D convolution and generates realistic high-resolution meteorological fields through a generative adversarial training mode.

Note that U-Net, Zooming Slow-Mo, SpateGAN, and TemDeep have been reported in temporal downscaling tasks and have demonstrated

good performance. Taking into account their common application scenarios, we evaluated the effectiveness of these comparison methods in the single-moment (i.e., the anchor moment in the training phase) downscaling task. All algorithms were trained on a device equipped with 16 vCPUs (Intel(R) Xeon(R) Gold 6430) and a single RTX 4090 (24 GB) GPU for accelerated inference. MSF-TCMA was implemented using PyTorch 2.5.1, optimized with the Adam optimizer, with the learning rate set to 0.0001, a batch size of 8, and trained for 100 epochs.

### 3.3.1. Estimating intermediate single moment state on dataset 1

**Table 1** show the downscaling results of the proposed method and the comparative methods on single-moment tasks for 2 h, 4 h, and 8 h time intervals. Analyzing the experimental data, the interpolation-based method achieves high accuracy in the 2 h task due to the low fluctuation probability of the meteorological variables in a short time range. Whereas as the temporal range extends, the accuracy of the interpolation method degrades significantly. Specifically, from 2 h to 8 h task, the downscaling MAE for SP increased by a factor of 8.7. Although flow-based methods have achieved SOTA performance in the field of video frame interpolation, they are less accurate than the other compared methods in the temporal downscaling task. In particular, in the 2 h task, the accuracy of RIFE and M2M-VFI is similar to that of other methods, while as the time range extended, the accuracy of the flow-based methods is much lower than that of the other deep learning methods. Despite the architecture of CNN and UNet being simpler than flow-based methods, without energy and streaming constraints, these methods achieve better results by directly establishing a mapping pattern from low-resolution data to high-resolution. Among the other SOTA methods, CAIN learns the local spatial correlation of meteorological variables and the coupling characteristics between multiple meteorological fields, but does not consider the fusion of meteorological fields with heterogeneous data. SWinIR considers local and global spatial correlation through the Transformer architecture but neglects the consideration of time dependence due to the simultaneous processing of front and back boundary information. AdaCOF considers complex spatial correlations in the input data by adaptive convolution, SpateGAN utilizes 3D convolution to capture local spatiotemporal correlations, but they overlook other problems faced by downscaling tasks. Zooming slow-mo fully learns the complex spatial-temporal dependencies of meteorological variables through the ConvLSTM architecture, but does not consider information fusion. Therefore, although these methods are more accurate than other comparative methods. None of them comprehensively considers the challenges faced in temporal downscaling tasks and performs worse than the proposed MSF-TCMA.

Compared with other methods, the proposed MSF-TCMA, by integrating multi-scale deep-wavelet spatial features and relative-absolute temporal dependencies to fully learn the spatial-temporal dependencies. And integrates the heterogeneous information by fusing meteorological and satellite observation data. MSF-TCMA achieves the best performance in the comparison experiment. For the 2 h time interval task, the estimation MAE for different variables was reduced by 3.99–35.10% compared to the best-performing comparison method. For the 4 h time interval task, it was reduced by 10.80–30.61%, and for the 8 h time interval tasks, it was reduced by 5.34–46.09%. Therefore, the proposed algorithm effectively improves the accuracy of single-moment downscaling tasks.

### 3.3.2. Estimating intermediate single moment state on dataset 2

**Table 2** shows the downscaling performance of comparison methods on single moment tasks of 2 h, 4 h, and 6 h time intervals on Dataset 2. In contrast to Dataset 1, dataset 2 focuses on land areas, where the meteorological state undergoes more significant changes in a short period due to the ground effect. Additionally, the spatial range covered is smaller, and the spatial resolution is higher. As a result, all methods

perform worse compared to that on Dataset 1. Nevertheless, the proposed algorithm still achieves the best performance among comparative methods. For the 2 h time interval task, the MAE of t2m is reduced by 17.05–94.14% compared with the comparative method, sp is reduced by 7.46–99.64%, u10 is reduced by 4.76–90.98%, and v10 is reduced by 7.73–88.03%. For the 4 h time interval task, the MAE of the four variables is reduced by 29.28–89.64%, 38.64–97.72%, 8.49–80.96%, and 12.85–58.54%, respectively. For the 6 h task, the MAE of the four variables decreased by 30.17–85.56%, 56.03–97.37%, 6.86–67.76%, and 11.18–66.02%, respectively. These experimental results further demonstrate that the proposed method can achieve more accurate downscaling results compared to existing methods in time downscaling tasks across study areas, time intervals, and variables.

**Fig. 7** shows the spatial distribution of the errors between the estimated values and the actual meteorological state for all samples in the test set at a 6-h time interval. Note that due to the large estimation errors of EMA and SGM, their result is not shown in the figure. It can be observed that for the four meteorological variables, the proposed method exhibits the smallest estimation error, with the error image appearing nearly white. This indicates that the estimation error at different spatial locations fluctuates around zero. In contrast, the other methods show significant differences in their error distributions. It can also be seen that the estimation errors of most methods align with the spatial distribution of the real meteorological state. Specifically, at locations where the real meteorological values are low, the estimation errors are often positive, while at locations with high real meteorological values, the estimation errors tend to be negative. This suggests that the error in meteorological downscaling primarily stems from the poor fitting ability of high-frequency real meteorological spatial changes. These algorithms generally produce more conservative and stable state estimates, which makes it challenging to accurately fit the extreme values of the spatial meteorological state.

## 3.4. Further analysis

### 3.4.1. Analysis of the importance of different modalities

This subsection evaluates the importance of multi-source data input in enhancing algorithm performance. For the multimodal data utilized by the proposed algorithm, including meteorological data, satellite observation data, and timestamp encoding, we conducted an experiment on the 4-h task of Dataset 2. Noise with varying signal-to-noise ratio (SNR) of 0, 3, 5, 10, 15, 25, and 50 dB was injected separately into the input meteorological data, satellite observation data, and timestamp encoding to assess their impact on algorithm performance. For meteorological data, noise was separately added to the meteorological states of the front and back boundaries, denoted as Meteorological\_Front and Meteorological\_Back. Additionally, for each source, both a zero matrix and the original data were used as reference inputs. Taking the estimated MAE for t2m as the evaluation metric, the performance degradation curve of the algorithm is shown in **Fig. 8**.

As observed in **Fig. 8**, the decrease in the SNR of different input sources has resulted in a decline in algorithm performance. This highlights the crucial role of meteorological state data, satellite observation data, and timestamp data in the downscaling task. Among them, the meteorological state is the most important source. When the SNR of a single boundary meteorological state is reduced to 25 dB, the downscaling MAE increases by more than 10%. Reducing the SNR to 15 dB nearly doubles the MAE. And at 0 dB, the MAE increases by almost an order of magnitude. Furthermore, at the same SNR, the meteorological states at the front and back boundaries have nearly equal influence on the downscaling task. Prove that the algorithm takes the boundary meteorological state as the main source for temporal downscaling, and estimates the intermediate meteorological state by fully leveraging the correlation between boundaries. Satellite observation data and timestamp encoding also contribute to the algorithm's performance. Notably, when the SNR dropped from 50 to 0 dB, the

**Table 1**

Comparison result on single moment task of 2h, 4h, and 8h time interval.

Variable	2h					4h					8h					
	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	
t2m (K)	Linear	0.0781	0.1333	0.9759	34.8680	0.8581	0.1543	0.2333	0.9226	30.0038	0.8188	0.2752	0.3771	0.7898	25.5239	0.7766
	CNN	0.0694	0.1140	0.9823	36.2206	0.8649	0.1398	0.2115	0.9374	30.8544	0.8360	0.2455	0.3387	0.8238	26.4566	0.8096
	U-Net	0.0856	0.1289	0.9778	35.1544	0.8229	0.1516	0.2201	0.9314	30.5104	0.8163	0.2552	0.3473	0.8179	26.2402	0.7992
	SWinIR	0.0652	0.1020	0.9867	37.1876	0.8588	0.1440	0.2080	0.9458	31.0010	0.8207	0.2697	0.3624	0.8191	25.8691	0.7881
	CAIN	0.0918	0.1355	0.9752	34.7261	0.8087	0.1715	0.2404	0.9188	29.7432	0.7881	0.2744	0.3683	0.7933	25.7302	0.7805
	AdaCOF	0.0685	0.1146	0.9822	36.1791	0.8688	0.1436	0.2143	0.9349	30.7399	0.8345	0.2859	0.3733	0.7955	25.6124	0.7731
	RIFE	0.0820	0.1221	0.9822	35.6295	0.8246	0.1781	0.2481	0.9237	29.4683	0.7785	0.2874	0.3884	0.7741	25.2691	0.7728
	M2M-VFI	0.0891	0.1420	0.9724	34.3156	0.8296	0.1988	0.2811	0.8864	28.3864	0.7657	0.3524	0.4599	0.6884	23.8009	0.7193
	EMA-VFI	0.5780	0.5908	0.9794	21.9342	0.5119	0.5589	0.6016	0.9222	21.7756	0.5473	0.3837	0.4678	0.7884	23.6527	0.6656
	SGM-VFI	0.4799	0.4943	0.9800	23.4832	0.5192	0.4301	0.4771	0.9243	23.7904	0.5799	0.3070	0.4096	0.7799	24.8064	0.7544
Zooming S-M	0.0708	0.1077	0.9850	36.7191	0.8478	0.1431	0.2145	0.9354	30.7345	0.8301	0.2745	0.3752	0.7806	25.5674	0.7877	
	SpateGAN	0.1336	0.1771	0.9598	32.4003	0.7299	0.1869	0.2533	0.9090	29.2891	0.7660	0.2953	0.3900	0.7697	25.2331	0.7633
	TemDeep	0.0759	0.1226	0.9795	35.5894	0.8516	0.1425	0.2101	0.9380	30.9157	0.8324	0.2537	0.3451	0.8168	26.2954	0.8035
	<b>Ours</b>	<b>0.0579</b>	<b>0.0951</b>	<b>0.9877</b>	<b>37.7973</b>	<b>0.8815</b>	<b>0.1247</b>	<b>0.1947</b>	<b>0.9466</b>	<b>31.5749</b>	<b>0.8541</b>	<b>0.2324</b>	<b>0.3332</b>	<b>0.8339</b>	<b>26.5991</b>	<b>0.8201</b>
	Linear	17.1597	21.5017	0.9954	40.9112	0.9199	44.3688	52.0457	0.9807	33.1296	0.8542	149.3973	163.1273	0.9375	22.6566	0.6739
	CNN	16.3340	20.7672	0.9955	41.2131	0.9372	29.7821	37.7003	0.9851	35.9304	0.9357	54.9618	70.1304	0.9324	29.9890	0.9075
sp (Pa)	U-Net	21.5463	27.8448	0.9922	38.6658	0.9112	26.6354	34.9128	0.9856	36.5976	0.9371	54.4329	74.4461	0.9221	29.4703	0.9170
	SWinIR	14.6291	18.4604	0.9962	42.2359	0.9426	24.0952	30.2275	0.9921	37.8493	0.9516	51.1267	63.8914	0.9545	30.7983	0.9217
	CAIN	26.5872	34.3312	0.9880	36.8469	0.8798	31.2634	40.0954	0.9827	35.3954	0.9155	58.4404	74.3109	0.9215	29.4861	0.8959
	AdaCOF	18.2148	23.0041	0.9947	40.3246	0.9198	41.0635	48.8443	0.9832	33.6810	0.8659	104.0015	126.6151	0.9111	24.8574	0.7440
	RIFE	54.7037	59.3152	0.9945	32.0974	0.7204	77.3824	88.0232	0.9708	28.5654	0.7310	212.6664	222.9871	0.9348	19.9415	0.5029
	M2M-VFI	21.0965	26.9400	0.9921	38.9527	0.9164	43.2074	53.3088	0.9719	32.9213	0.8658	117.2715	140.9310	0.8566	23.9270	0.7202
	EMA-VFI	510.4843	511.0353	0.9944	13.3917	0.4572	655.1423	656.6727	0.9767	11.1103	0.3608	650.3096	653.6924	0.9363	10.5996	0.1740
	SGM-VFI	18.8958	24.2612	0.9945	39.8624	0.9224	38.5883	46.2227	0.9763	34.1602	0.8969	214.5813	224.5727	0.9363	19.8800	0.5033
	Zooming S-M	20.8697	27.3487	0.9952	38.8220	0.9228	25.9031	32.6688	0.9916	37.1746	0.9447	71.1306	91.9363	0.9279	27.6374	0.8960
	SpateGAN	36.7022	48.1862	0.9755	32.9022	0.8370	47.8817	62.3394	0.9533	31.5621	0.8814	116.5397	144.2870	0.7113	23.7226	0.8503
u10 (m/s)	TemDeep	17.0599	21.7572	0.9949	40.8086	0.9282	30.8520	39.4468	0.9857	35.5371	0.9417	54.7233	71.0865	0.9364	29.8714	0.9060
	<b>Ours</b>	<b>14.0451</b>	<b>17.6595</b>	<b>0.9965</b>	<b>42.6211</b>	<b>0.9478</b>	<b>16.7192</b>	<b>21.1643</b>	<b>0.9948</b>	<b>40.9452</b>	<b>0.9663</b>	<b>27.5644</b>	<b>35.0632</b>	<b>0.9828</b>	<b>36.0101</b>	<b>0.9657</b>
	Linear	0.1430	0.2148	0.9983	43.6934	0.8428	0.2976	0.4107	0.9940	38.0273	0.7932	0.5385	0.7258	0.9821	32.5599	0.7525
	CNN	0.1139	0.1765	0.9989	45.3954	0.8772	0.2344	0.3272	0.9962	40.0011	0.8425	0.4708	0.6355	0.9854	33.7130	0.7891
	U-Net	0.2328	0.3152	0.9970	40.3617	0.7507	0.3179	0.4357	0.9933	37.5134	0.7858	0.5235	0.7061	0.9833	32.7987	0.7734
	SWinIR	0.1050	0.1507	0.9992	46.7677	0.8752	0.2929	0.3767	0.9966	38.7778	0.7964	0.4505	0.6113	0.9863	34.0513	0.8008
	CAIN	0.1921	0.2659 v	0.9975	41.8380	0.7899	0.3175	0.4358	0.9932	37.5114	0.7866	0.6161	0.8172	0.9760	31.5291	0.7392
	AdaCOF	0.1353	0.1995	0.9986	44.3323	0.8467	0.3051	0.4231	0.9935	37.7691	0.7933	0.5557	0.7657	0.9786	32.0946	0.7553
	RIFE	1.2603	1.2719	0.9989	28.2434	0.5179	1.6209	1.6654	0.9944	25.8669	0.5401	1.8834	2.0003	0.9813	23.7539	0.5988
	M2M-VFI	0.1857	0.2846	0.9971	41.2472	0.8041	0.4237	0.6117	0.9863	34.5658	0.7325	0.7344	1.0251	0.9612	29.5601	0.6922
	EMA-VFI	2.6728	2.6787	0.9988	21.7738	0.5141	3.0339	3.0623	0.9936	20.5763	0.5329	2.8192	2.9077	0.9808	20.5049	0.5880
	SGM-VFI	2.7664	2.7717	0.9989	21.4775	0.5140	3.3845	3.4048	0.9949	19.6555	0.5327	1.7649	1.8888	0.9819	24.2522	0.5996
Zooming S-M	TemDeep	0.1474	0.1960	0.9991	44.4885	0.8280	0.2539	0.3556	0.9957	39.2769	0.8312	0.5226	0.7019	0.9829	32.8497	0.7731
	SpateGAN	0.5616	0.7522	0.9800	32.8061	0.6212	0.6320	0.8471	0.9736	31.7380	0.6710	1.0517	1.3679	0.9451	27.0545	0.6642
	TemDeep	0.1480	0.2130	0.9984	43.7642	0.8334	0.2848	0.3938	0.9944	38.3915	0.8050	0.5030	0.6838	0.9841	33.0778	0.7793
	<b>Ours</b>	<b>0.0800</b>	<b>0.1196</b>	<b>0.9995</b>	<b>48.7812</b>	<b>0.9095</b>	<b>0.2029</b>	<b>0.2839</b>	<b>0.9971</b>	<b>41.2339</b>	<b>0.8639</b>	<b>0.4250</b>	<b>0.5792</b>	<b>0.9879</b>	<b>34.5195</b>	<b>0.8128</b>
	Linear	0.1139	0.1765	0.9989	45.3954	0.8772	0.2344	0.3272	0.9962	40.0011	0.8425	0.4708	0.6355	0.9854	33.7130	0.7891

(continued on next page)

**Table 1** (continued).

	Linear	0.1503	0.2223	0.9966	43.7181	0.8498	0.3277	0.4481	0.9861	37.5725	0.7972	0.6132	0.8165	0.9557	32.0177	0.7480
	CNN	0.1225	0.1849	0.9977	45.3172	0.8765	0.2627	0.3584	0.9919	39.5118	0.8402	0.5264	0.6996	0.9658	33.3607	0.7906
	U-Net	0.2031	0.2858	0.9944	41.5333	0.7981	0.3629	0.4807	0.9859	36.9622	0.7800	0.5529	0.7405	0.9611	32.8671	0.7830
	SWinIR	0.1226	0.1696	0.9980	46.0682	0.8692	0.2801	0.3626	0.9931	39.4105	0.8253	0.6096	0.7936	0.9684	32.2647	0.7557
	CAIN	0.1744	0.2473	0.9957	42.7902	0.8253	0.3261	0.4454	0.9862	37.6248	0.8042	0.6386	0.8430	0.9535	31.7410	0.7457
	AdaCOF	0.1457	0.2146	0.9968	44.0245	0.8520	0.3437	0.4685	0.9849	37.1848	0.7913	0.6975	0.9468	0.9428	30.7324	0.7281
v10 (m/s)	RIFE	1.1751	1.1877	0.9978	29.1618	0.5115	1.4021	1.4560	0.9883	27.3362	0.5402	1.5176	1.6740	0.9572	25.7820	0.6131
	M2M-VFI	0.1943	0.2886	0.9942	41.4501	0.8125	0.4491	0.6260	0.9724	34.6672	0.7388	0.8111	1.0869	0.9181	29.5337	0.6878
	EMA-VFI	2.5511	2.5574	0.9978	22.4997	0.5058	2.7904	2.8263	0.9858	21.5751	0.5228	1.6000	1.7632	0.9534	25.3314	0.6026
	SGM-VFI	3.2850	3.2895	0.9979	20.3130	0.5057	4.3237	4.3422	0.9887	17.8452	0.5223	4.3667	4.4359	0.9564	17.3175	0.5795
	Zooming S-M	0.1446	0.2031	0.9980	44.5035	0.8491	0.2634	0.3633	0.9912	39.3952	0.8418	0.6085	0.8110	0.9553	32.0765	0.7627
	SpateGAN	0.5134	0.7192	0.9635	33.5191	0.6536	0.6249	0.8111	0.9556	32.4179	0.6869	0.9264	1.1912	0.8960	28.7377	0.6844
	TemDeep	0.1390	0.2078	0.9970	44.3024	0.8609	0.3201	0.4400	0.9870	37.7302	0.8043	0.6039	0.8070	0.9594	32.1195	0.7578
	<b>Ours</b>	<b>0.0795</b>	<b>0.1172</b>	<b>0.9990</b>	<b>49.2737</b>	<b>0.9192</b>	<b>0.2107</b>	<b>0.2911</b>	<b>0.9941</b>	<b>41.3180</b>	<b>0.8746</b>	<b>0.4764</b>	<b>0.6382</b>	<b>0.9715</b>	<b>34.1581</b>	<b>0.8145</b>

Note: Results of MSF-TCMA are highlighted in red, the best results are highlighted in bold.

**Table 2**

Comparison result on single moment task of 2h, 4h, and 6h time interval.

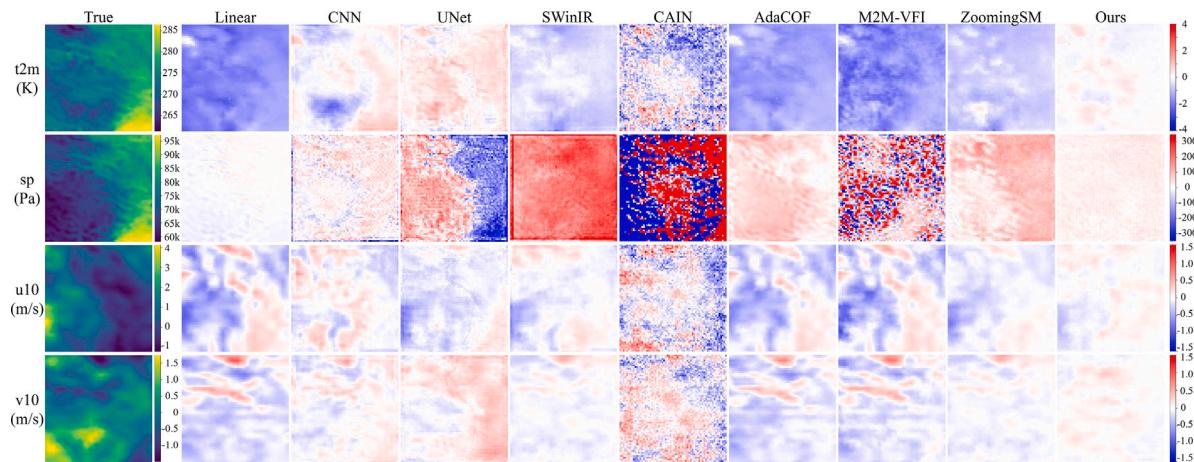
Variable	2h						4h						6h					
	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	MAE↓	RMSE↓	ACC↑	PSNR↑	EDA↑	MAE↓	RMSE↓	ACC↑
t2m (K)	Linear	0.4006	0.5767	0.9964	40.9523	0.9114	0.9642	1.1847	0.9884	34.4834	0.8572	1.8920	2.2108	0.9741	28.5741	0.7987		
	CNN	0.3729	0.5564	0.9964	41.2644	0.9162	0.6737	0.9618	0.9882	36.2941	0.9154	1.1021	1.5280	0.9689	31.7829	0.9153		
	U-Net	0.4725	0.6659	0.9948	39.7034	0.8803	0.6985	0.9859	0.9878	36.0788	0.9114	0.9339	1.2695	0.9782	33.3927	0.9236		
	SWinIR	0.3008	0.4823	0.9973	42.5054	0.9284	0.6828	0.9856	0.9904	36.0811	0.9301	0.9626	1.3704	0.9796	32.7281	0.9289		
	CAIN	0.8082	1.0621	0.9864	35.6486	0.8017	0.9527	1.2487	0.9802	34.0261	0.8651	1.4230	1.8112	0.9582	30.3058	0.8569		
	AdaCOF	0.3782	0.5543	0.9965	41.2958	0.9124	0.8653	1.0911	0.9886	35.1982	0.8580	1.6224	1.9486	0.9719	29.6705	0.8007		
	RIFE	2.1647	2.2206	0.9967	29.2421	0.5289	2.3209	2.4567	0.9899	28.1484	0.6145	2.8960	3.1467	0.9761	25.5079	0.6564		
	M2M-VFI	0.4330	0.6344	0.9954	40.1237	0.9081	1.0027	1.3371	0.9835	33.4320	0.8569	1.9771	2.5942	0.9490	27.1849	0.8022		
	EMA-VFI	4.2597	4.2901	0.9969	23.5222	0.5519	4.5874	4.6680	0.9902	22.5728	0.6632	4.5167	4.6964	0.9759	22.0298	0.7314		
	SGM-VFI	4.1786	4.2140	0.9965	23.6777	0.5502	4.1758	4.2766	0.9888	23.3336	0.6720	1.6554	1.9568	0.9749	29.6343	0.8558		
Zooming S-M	0.4010	0.5511	0.9970	41.3470	0.8919	0.6722	0.9409	0.9907	36.4851	0.9192	1.1318	1.6793	0.9729	30.9626	0.9150			
	SpateGAN	0.5396	0.7465	0.9934	38.7107	0.8586	0.8562	1.1546	0.9841	34.7072	0.8947	1.1110	1.5087	0.9702	31.8930	0.9100		
	TemDeep	0.4410	0.6494	0.9951	39.9214	0.8960	0.7434	1.0673	0.9861	35.3896	0.9049	1.1378	1.6160	0.9669	31.2962	0.9088		
	<b>Ours</b>	<b>0.2495</b>	<b>0.4340</b>	<b>0.9977</b>	<b>43.4226</b>	<b>0.9398</b>	<b>0.4754</b>	<b>0.7073</b>	<b>0.9935</b>	<b>38.9635</b>	<b>0.9378</b>	<b>0.6521</b>	<b>0.9081</b>	<b>0.9884</b>	<b>36.3027</b>	<b>0.9449</b>		
	Linear	16.0965	20.1414	0.9986	66.2524	0.8784	55.2731	61.9621	0.9860	56.4917	0.7755	93.7625	106.9542	0.9556	51.7434	0.7606		
	CNN	48.7395	64.6392	0.9850	56.1243	0.7327	60.7976	77.9980	0.9772	54.4925	0.7938	83.0664	106.8714	0.9573	51.7502	0.8039		
	U-Net	117.2253	164.6906	0.9092	48.0008	0.6178	118.1896	151.5541	0.9284	48.7228	0.6919	173.7163	220.1271	0.8559	45.4739	0.6818		
44 sp (Pa)	SWinIR	22.7068	31.3814	0.9967	62.4007	0.8513	49.7333	65.5761	0.9846	55.9993	0.8379	175.2380	204.3193	0.9482	46.1212	0.6504		
	CAIN	913.9760	1239.2457	0.2762	30.4710	0.5178	959.0927	1289.0216	0.2664	30.1290	0.5257	957.0020	1280.9154	0.2500	30.1770	0.5301		
	AdaCOF	18.5592	25.6668	0.9976	64.1468	0.8687	52.5332	61.7423	0.9860	56.5225	0.7989	89.2424	109.4814	0.9616	51.5406	0.8049		
	RIFE	1260.8196	1261.0180	0.9982	30.3198	0.4985	1341.0797	1342.5571	0.9855	29.7755	0.4927	1380.1958	1384.3164	0.9557	29.5027	0.4784		
	M2M-VFI	62.7897	123.8019	0.9469	50.4796	0.7621	174.4585	287.8780	0.7781	43.1500	0.6822	292.3047	458.4746	0.6094	39.1011	0.6554		
	EMA-VFI	4153.6074	4153.6736	0.9980	19.9655	0.5015	1247.1051	1248.7672	0.9847	30.4045	0.5073	559.5090	569.6001	0.9556	37.2160	0.5217		
	SGM-VFI	947.5455	947.8783	0.9977	32.7991	0.5015	599.8273	603.2105	0.9850	36.7248	0.5073	654.1701	663.9967	0.9496	35.8841	0.5216		
	Zooming S-M	93.2928	124.1561	0.9510	50.4548	0.6443	75.6360	101.9811	0.9608	52.1638	0.7793	82.4210	106.2381	0.9729	51.8018	0.8268		
	SpateGAN	179.2035	232.9199	0.8054	44.9900	0.5826	282.1987	358.5829	0.6519	41.2424	0.5886	323.3663	411.7020	0.4904	40.0357	0.6240		
	TemDeep	86.8060	115.0968	0.9505	51.1129	0.6442	95.9916	123.6307	0.9434	50.4917	0.7166	123.9664	161.1041	0.9119	48.1853	0.7453		
	<b>Ours</b>	<b>14.8949</b>	<b>18.7194</b>	<b>0.9988</b>	<b>66.8884</b>	<b>0.8953</b>	<b>30.5186</b>	<b>40.1887</b>	<b>0.9946</b>	<b>60.2521</b>	<b>0.9008</b>	<b>36.2425</b>	<b>45.9391</b>	<b>0.9935</b>	<b>59.0837</b>	<b>0.9207</b>		
u10 (m/s)	Linear	0.1502	0.2210	0.9898	40.5289	0.8640	0.3601	0.5136	0.9452	33.2033	0.8154	0.5628	0.7975	0.8603	29.3497	0.7771		
	CNN	0.1408	0.2028	0.9911	41.2728	0.8642	0.3095	0.4326	0.9620	34.6937	0.8345	0.4547	0.6239	0.9132	31.4812	0.8203		
	U-Net	0.1869	0.2531	0.9864	39.3483	0.8138	0.3074	0.4300	0.9604	34.7463	0.8358	0.4202	0.5780	0.9288	32.1458	0.8326		
	SWinIR	0.1239	0.1842	0.9925	42.1084	0.8827	0.2837	0.3953	0.9664	35.4764	0.8469	0.4007	0.5638	0.9302	32.3619	0.8452		
	CAIN	0.2542	0.3318	0.9764	36.9982	0.7609	0.4441	0.5889	0.9247	32.0146	0.7633	0.6311	0.8272	0.8428	29.0321	0.7523		
	AdaCOF	0.1440	0.2127	0.9904	40.8586	0.8656	0.3533	0.5029	0.9466	33.3860	0.8165	0.5558	0.7840	0.8646	29.4977	0.7785		
	RIFE	1.0188	1.0389	0.9907	27.0835	0.5252	1.0292	1.1142	0.9569	26.4756	0.5739	1.0559	1.2248	0.8992	25.6230	0.6046		
	M2M-VFI	0.1652	0.2474	0.9866	39.5489	0.8606	0.3630	0.5248	0.9402	33.0151	0.8151	0.5704	0.8057	0.8551	29.2611	0.7787		
	EMA-VFI	1.3084	1.3245	0.9907	24.9744	0.5134	1.3632	1.4290	0.9593	24.3143	0.5284	1.1575	1.3104	0.9069	25.0363	0.5652		
	SGM-VFI	0.9684	0.9912	0.9900	27.4917	0.5284	0.7168	0.8322	0.9535	29.0101	0.6432	0.5804	0.7844	0.9036	29.4935	0.7636		
Zooming S-M	0.1448	0.2068	0.9910	41.1053	0.8579	0.3010	0.4175	0.9629	35.0027	0.8352	0.4329	0.6100	0.9205	31.6777	0.8301			
	SpateGAN	0.1969	0.2737	0.9839	38.6690	0.8072	0.3435	0.4680	0.9525	34.0104	0.8107	0.5114	0.6975	0.8918	30.5127	0.7961		
	TemDeep	0.1511	0.2146	0.9900	40.7833	0.8519	0.3102	0.4290	0.9606	34.7656	0.8308	0.4618	0.6283	0.9146	31.4205	0.8178		
	<b>Ours</b>	<b>0.1180</b>	<b>0.1721</b>	<b>0.9935</b>	<b>42.6982</b>	<b>0.8863</b>	<b>0.2596</b>	<b>0.3664</b>	<b>0.9710</b>	<b>36.1350</b>	<b>0.8608</b>	<b>0.3732</b>	<b>0.5126</b>	<b>0.9422</b>	<b>33.1878</b>	<b>0.8544</b>		

(continued on next page)

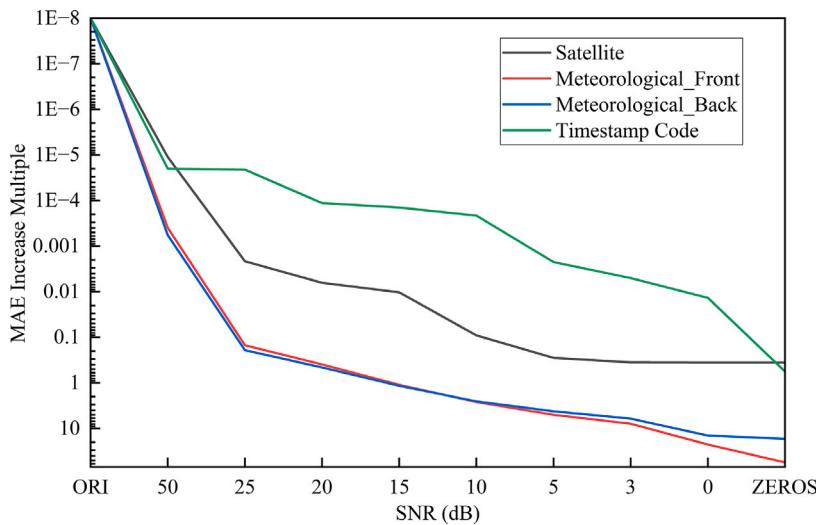
**Table 2** (continued).

	Linear	0.1503	0.2223	0.9966	43.7181	0.8498	0.3277	0.4481	0.9861	37.5725	0.7972	0.6132	0.8165	0.9557	32.0177	0.7480
	CNN	0.1225	0.1849	0.9977	45.3172	0.8765	0.2627	<b>0.3584</b>	0.9919	<b>39.5118</b>	0.8402	0.5264	0.6996	0.9658	<b>33.3607</b>	0.7906
	U-Net	0.2031	0.2858	0.9944	41.5333	0.7981	0.3629	0.4807	0.9859	36.9622	0.7800	0.5529	0.7405	0.9611	32.8671	0.7830
	SWinIR	0.1226	0.1696	0.9980	<b>46.0682</b>	0.8692	0.2801	0.3626	<b>0.9931</b>	39.4105	0.8253	0.6096	0.7936	<b>0.9684</b>	32.2647	0.7557
	CAIN	0.1744	0.2473	0.9957	42.7902	0.8253	0.3261	0.4454	0.9862	37.6248	0.8042	0.6386	0.8430	0.9535	31.7410	0.7457
	AdaCOF	0.1457	0.2146	0.9968	44.0245	0.8520	0.3437	0.4685	0.9849	37.1848	0.7913	0.6975	0.9468	0.9428	30.7324	0.7281
v10 (m/s)	RIFE	1.1751	1.1877	0.9978	29.1618	0.5115	1.4021	1.4560	0.9883	27.3362	0.5402	1.5176	1.6740	0.9572	25.7820	0.6131
	M2M-VFI	0.1943	0.2886	0.9942	41.4501	0.8125	0.4491	0.6260	0.9724	34.6672	0.7388	0.8111	1.0869	0.9181	29.5337	0.6878
	EMA-VFI	2.5511	2.5574	0.9978	22.4997	0.5058	2.7904	2.8263	0.9858	21.5751	0.5228	1.6000	1.7632	0.9534	25.3314	0.6026
	SGM-VFI	3.2850	3.2895	0.9979	20.3130	0.5057	4.3237	4.3422	0.9887	17.8452	0.5223	4.3667	4.4359	0.9564	17.3175	0.5795
	Zooming S-M	0.1446	0.2031	<b>0.9980</b>	44.5035	0.8491	0.2634	0.3633	0.9912	39.3952	0.8418	0.6085	0.8110	0.9553	32.0765	0.7627
	SpatGAN	0.2009	0.2657	0.9874	37.0147	0.7967	0.3674	0.4877	0.9617	31.7412	0.7920	0.5073	0.6722	0.9220	28.9532	0.7890
	TemDeep	0.1668	0.2279	0.9913	38.3473	0.8309	0.3079	0.4188	0.9684	33.0623	0.8235	0.4600	0.6091	0.9374	29.8099	0.7930
	<b>Ours</b>	<b>0.1169</b>	<b>0.1670</b>	<b>0.9947</b>	<b>41.0469</b>	<b>0.8827</b>	<b>0.2617</b>	<b>0.3617</b>	<b>0.9764</b>	<b>34.3360</b>	<b>0.8530</b>	<b>0.3751</b>	<b>0.5096</b>	<b>0.9039</b>	<b>31.3582</b>	<b>0.8393</b>

Note: Results of MSF-TCMA are highlighted in red, the best results are highlighted in bold.



**Fig. 7.** The error spatial distribution of comparative methods.



**Fig. 8.** The performance degradation curve.

influence of satellite observation data on the algorithm's performance is greater than that of timestamp encoding. When the original data for either of these two inputs is replaced by a zero matrix, the algorithm's performance drops by 35%–41%.

#### 3.4.2. Hyperparameter sensitivity analysis

This subsection investigates the impact of key hyperparameters in MSF-TCMA on downscaling performance. Specifically, we examine the number of multiscale feature extraction modules ( $B$ ) in the multiscale deep-wavelet feature extraction branch; the number of CM-STIF blocks ( $N$ ) in the cross-modal spatiotemporal information fusing branch (which also determines the number of meteorological state estimate blocks in the time-continuous manifold sampling branch); and the number of cross-moment information fusion layers within each CMSTIF block ( $L$ ). We evaluated the impact of different hyperparameter settings on both the downscaling error and inference efficiency of MSF-TCMA using Dataset 2. Considering the different units of the meteorological variables, all four variables were normalized, and the mean MAE and RMSE were used as the evaluation metrics for downscaling error. In addition, inference time for each sample was measured on a 16 vCPU Intel(R) Xeon(R) Gold 6430 platform without GPU acceleration. The experimental results are shown in Table 3.

It can be seen from the table that increasing  $N$ ,  $L$ , and  $B$  significantly improves the downscaling accuracy for  $t2m$  and  $sp$ . However, these adjustments do not result in notable improvements for  $u10$  and

$v10$ . Consequently, in the experiments on  $N$  and  $B$ , further increasing these hyperparameters does not reduce the average error of the algorithm. When  $N$  is set to 2 and  $B$  to 3, the algorithm achieves optimal downscaling accuracy. As for  $L$ , although increasing its value slightly enhances performance, it also leads to a substantial decrease in inference efficiency. Therefore, considering the trade-off between downscaling accuracy and efficiency, we set  $L$  to 4.

#### 3.5. Engineering application experiment 1—Estimating multiple moment unseen meteorological states on dataset 2

##### 3.5.1. Experiment setting and comparison results

The proposed method can not only estimate the meteorological state at a single moment within a time interval with higher accuracy than other methods, but also provide continuous estimation of the meteorological state at multiple moments within the interval. This includes both seen anchor moments and unseen moments during the training stage. To evaluate the multi-moment downscaling capability, we conducted 6-h, 10-h, and 12-h multi-moment downscaling experiments on Dataset 2. For comparison, we selected the SGM-VFI, M2M-VFI, and linear interpolation methods, all of which support this multi-moment downscaling task. The specific experimental details are shown in Table 4. Note that the term downscaling scale refers to the ratio by which the dataset resolution is downsampled, rather than the input resolution to the algorithm. For the comparison methods, we

**Table 3**

Hyperparameter sensitivity experimental result.

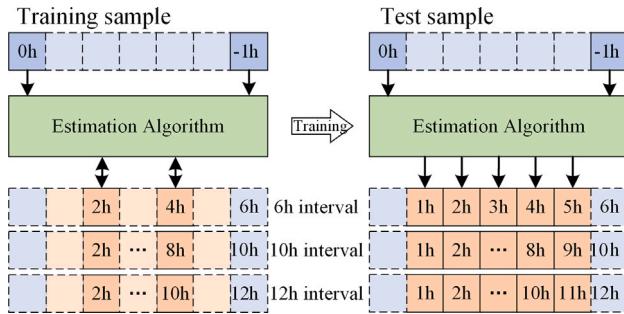
HyperParameter	MAE↓					RMSE↓					Time↓
	t2m	sp	u10	v10	Avg	t2m	sp	u10	v10	Avg	
N	1	0.4775	34.4683	0.2648	0.2666	0.006965	0.7180	43.6701	0.3746	0.3678	0.009859
	2	0.4754	30.5186	<b>0.2596</b>	<b>0.2617</b>	<b>0.006836</b>	<b>0.7073</b>	40.1887	<b>0.3664</b>	<b>0.3617</b>	<b>0.009665</b>
	3	<b>0.4739</b>	<b>28.6124</b>	0.2599	0.2662	0.006872	<b>0.7073</b>	<b>36.8448</b>	0.3670	0.3669	0.009708
L	2	0.4714	31.3801	0.2621	0.2641	0.006878	0.7056	40.4854	0.3693	0.3631	0.009703
	4	0.4754	30.5186	<b>0.2596</b>	<b>0.2617</b>	<b>0.006836</b>	0.7073	40.1887	<b>0.3664</b>	<b>0.3617</b>	0.009665
	6	<b>0.4580</b>	<b>27.6677</b>	0.2628	0.2654	0.006837	<b>0.6839</b>	<b>34.9723</b>	0.3706	0.3648	<b>0.009639</b>
B	2	0.4753	31.7436	0.2628	0.2670	0.006928	0.7138	40.8432	0.3692	0.3700	0.009801
	3	0.4754	30.5186	<b>0.2596</b>	<b>0.2617</b>	<b>0.006836</b>	0.7073	40.1887	<b>0.3664</b>	<b>0.3617</b>	<b>0.009665</b>
	4	<b>0.4665</b>	<b>24.4748</b>	0.2638	0.2677	0.006882	<b>0.6933</b>	<b>31.1861</b>	0.3709	0.3685	0.009689

Note: The best results are highlighted in bold.

**Table 4**

Experimental details of engineering application experiment 1.

Symbol	Time interval	Seen moments	Unseen moments	Training time resolution	Test time resolution	Downscaling scale
6h-1m	6 h	3h	1h, 2h, 4h, 5h	3 h	1 h	3x
6h-2m		2h, 4h	1h, 3h, 5h		1 h	2x
6h-3m		1h, 3h, 5h	2h, 4h		1~2 h	1~2x
6h-5m		1h, 2h, 3h, 4h, 5h	–		1 h	1x
10h-1m	10 h	5 h	1h, 2h, 3h, 4h, 6h, 7h, 8h, 9h	5 h	1 h	5x
10h-2m		4h, 8h	1h, 2h, 3h, 5h, 6h, 7h, 9h		1 h	2~4x
10h-3m		3h, 6h, 9h	1h, 2h, 4h, 5h, 7h, 8h		1~3 h	1~3x
10h-4m		2h, 4h, 6h, 8h	1h, 3h, 5h, 7h, 9h		2 h	2x
12h-1m	12 h	6h	1h, 2h, 3h, 4h, 5h, 7h, 8h, 9h, 10h, 11h	6 h	1 h	6x
12h-2m		4h, 8h	1h, 2h, 3h, 5h, 6h, 7h, 9h, 10h, 11h		1 h	4x
12h-3m		3h, 6h, 9h	1h, 2h, 4h, 5h, 7h, 8h, 10h, 11h		3 h	3x
12h-5m		2h, 4h, 6h, 8h, 10h	1h, 3h, 5h, 7h, 9h, 11h		2 h	2x

**Fig. 9.** The training and testing mode of engineering application experiment 1.

evaluated their performance at a 2 × downscaling scale, as shown in Fig. 9. Furthermore, the proposed MSF-TCMA was trained with different anchor points to explore its ability to estimate meteorological states at unseen moments.

Due to the limited availability of datasets, the testing objective was to provide meteorological states at a 1-h temporal resolution within each interval, while during the training stage, only meteorological states at a few selected moments within the interval were used to supervise the optimization process. In this way, we evaluated both the algorithm's ability to downscale data with a temporal resolution coarser than 1 h to 1 h, as well as its estimation performance for unseen moments within the interval.

Tables 5–7 present the MAE of different methods for downscaling four meteorological variables under the same training condition. Additionally, we calculated EDA and SSIM (Wang et al., 2004) for each moment within the interval to describe the temporal and spatial consistency between the downscaled results and the actual meteorological fields. The SSIM value ranges from –1 to 1, higher SSIM values indicate that the estimated meteorological field is more similar to the actual field in terms of spatial structure distribution.

In the 6h-2m, 10h-4m, and 12h-6m tasks, the proposed method consistently achieved lower downscaling errors at all moments within the interval compared to the other methods, while generally obtaining higher SSIM and EDA values. This demonstrates that the proposed method can accurately estimate the meteorological evolution at multiple moments within the interval, with the estimated dynamics closely matching the real situation. Taking the 10h-4m task as an example, Fig. 10 shows the estimation errors for the test set at different moments within the interval. The meteorological states at 2, 4, 6, and 8 h were selected as anchor data for supervision the algorithm training as the seen moments, while 1, 3, 5, 7, and 9 h are unseen moments. Experimental results indicate that the proposed method achieves the lowest estimation errors among all methods. Specifically, the MAE for t2m is reduced by 6.3–88.09%, the error for sp is reduced by 1.26–98.82%, and the errors for u10 and v10 are reduced by up to 68.40% and 51.35%, respectively. These results demonstrate that the proposed method achieves excellent estimation accuracy in multi-moment downscaling tasks.

As shown in Fig. 10, for comparative methods, the estimation error within the temporal interval increases as the prediction moment moves further away from the temporal boundaries. This indicates that the correlation between the meteorological states at a given moment and those at the boundaries weakens with increasing distance, making it more difficult for the algorithms to achieve accurate estimations. Notably, the proposed method yields estimation errors at unseen moments that are almost identical to those at seen moments, and the distribution of estimation errors remains smooth throughout the interval. This demonstrates that the proposed method can infer meteorological states at unseen moments with high accuracy, even when trained on datasets with limited temporal resolution.

Consider the hourly estimation results for wind speed between UTC 0:00–10:00 on November 30, 2024, as shown in Fig. 11. The trend of wind speed first increases and then decreases within the time interval. However, the linear model and the M2M fail to capture this nonlinear change, resulting in large errors. This demonstrates that the evolution of meteorological states cannot be viewed as a linear model. Among the

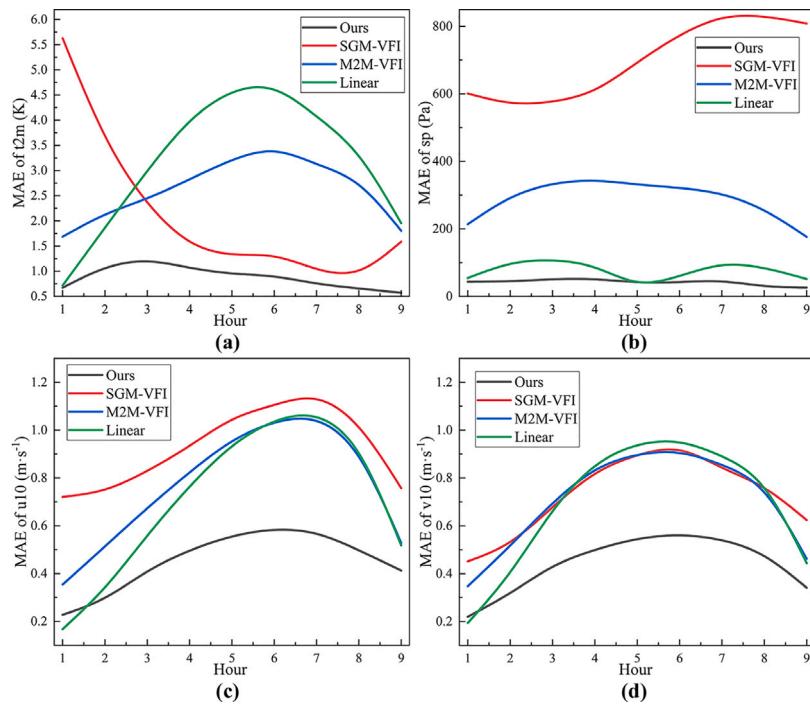


Fig. 10. The estimation error at different moments within the interval.

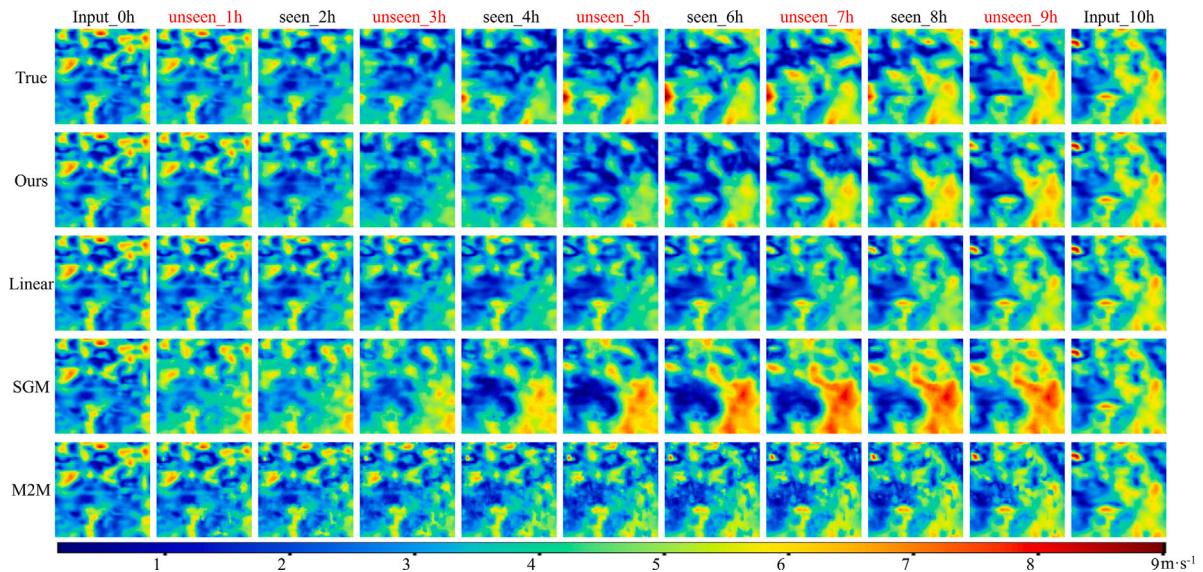


Fig. 11. The hourly downscaling results for wind speed comparative methods.

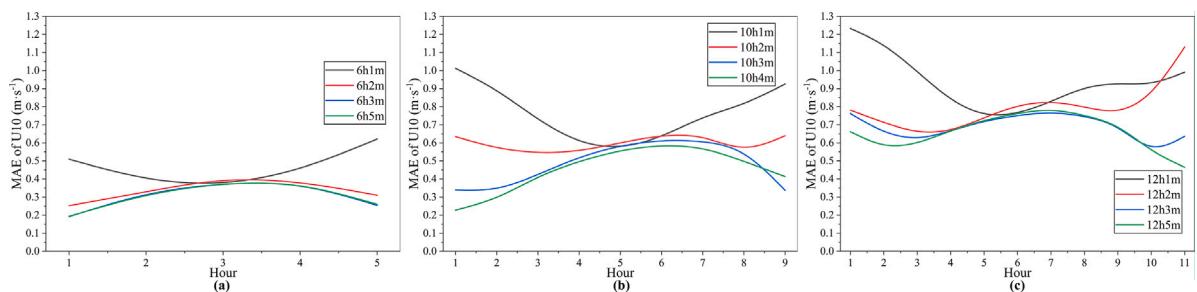


Fig. 12. MAE curve of MSF-TCMA for across different train modes.

**Table 5**

Comparison result on multi moment task of 6h time interval.

Variable	Moment	SGM-6h2m			M2M-6h2m			Linear-6h			Ours-6h2m		
		MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑
t2m	1	4.8023	0.9040	0.4963	1.4295	0.8609	0.8363	1.1089	0.9493	0.8936	<b>0.6762</b>	<b>0.9703</b>	<b>0.9095</b>
	2	3.3097	0.9195	0.5788	1.9787	0.7797	0.6360	1.6901	0.9129	0.6636	<b>0.7626</b>	<b>0.9541</b>	<b>0.8821</b>
	3	3.0538	0.9021	0.5798	2.0342	0.7347	0.5798	1.8920	0.9024	0.5748	<b>0.7408</b>	<b>0.9485</b>	<b>0.7923</b>
	4	2.8064	0.9325	0.6014	1.7285	0.7505	0.5475	1.7083	0.9242	0.5183	<b>0.5683</b>	<b>0.9594</b>	<b>0.7733</b>
	5	2.6321	0.9317	0.6075	1.1676	0.8280	0.6360	1.1050	<b>0.9681</b>	0.5736	<b>0.5127</b>	0.9640	<b>0.7769</b>
sp	1	565.8928	1.0000	0.5963	223.9863	0.9767	0.5989	47.4666	1.0000	0.6646	<b>44.2526</b>	<b>1.0000</b>	<b>0.7630</b>
	2	599.8738	0.9996	0.5751	307.7769	0.9627	0.5557	81.4581	0.9997	0.6051	<b>46.3244</b>	<b>1.0000</b>	<b>0.7746</b>
	3	620.7300	0.9989	0.5043	334.0273	0.9571	0.5531	93.7625	0.9991	0.5967	<b>56.2596</b>	<b>1.0000</b>	<b>0.7361</b>
	4	654.8742	0.9995	0.4312	304.0647	0.9610	0.5611	82.3053	0.9996	0.6050	<b>47.5090</b>	<b>1.0000</b>	<b>0.7457</b>
	5	691.6220	1.0000	0.4057	219.4221	0.9743	0.5984	49.0867	1.0000	0.6713	<b>43.2723</b>	<b>1.0000</b>	<b>0.7610</b>
u10	1	0.4900	0.8800	0.6239	0.3742	0.8374	0.7123	0.2783	0.9150	0.7446	<b>0.2524</b>	<b>0.9375</b>	<b>0.7571</b>
	2	0.5205	0.8262	0.6316	0.5619	0.7303	0.6020	0.4668	0.8004	0.6410	<b>0.3292</b>	<b>0.8897</b>	<b>0.7154</b>
	3	0.6124	0.7715	0.5954	0.6482	0.6821	0.5810	0.5628	0.7390	0.6082	<b>0.3910</b>	<b>0.8526</b>	<b>0.6749</b>
	4	0.6021	0.7694	0.5972	0.6069	0.7141	0.5753	0.5419	0.7582	0.6023	<b>0.3779</b>	<b>0.8667</b>	<b>0.6824</b>
	5	0.5417	0.7912	0.6328	0.4238	0.8296	0.6827	0.3685	0.8649	0.6909	<b>0.3101</b>	0.9151	<b>0.7410</b>
v10	1	0.5351	0.5149	0.6105	0.3710	0.8069	0.7104	0.2873	<b>0.8878</b>	0.7370	<b>0.2622</b>	0.8850	<b>0.7612</b>
	2	0.5770	0.5318	0.5947	0.5452	0.6736	0.6004	0.4715	0.7366	0.6278	<b>0.3317</b>	<b>0.8414</b>	<b>0.7062</b>
	3	0.6511	0.4924	0.5716	0.6146	0.6260	0.5808	0.5560	0.6699	0.5969	<b>0.3976</b>	<b>0.7971</b>	<b>0.6650</b>
	4	0.6111	0.5524	0.5800	0.5659	0.6763	0.5744	0.5217	0.7104	0.5903	<b>0.3730</b>	<b>0.8202</b>	<b>0.6697</b>
	5	0.5559	0.6089	0.6166	0.3935	0.8103	0.6803	0.3469	0.8525	0.6836	<b>0.3066</b>	<b>0.8679</b>	<b>0.7326</b>

Note: The best results are highlighted in **bold**, the 1h, 3h, and 5h are the unseen moments during the training phase.**Table 6**

Comparison result on multi moment task of 10h time interval.

Variable	Moment	SGM-10h4m			M2M-10h4m			Linear-10h			Ours-10h4m		
		MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑
t2m	1	5.6300	0.9497	0.4494	1.6855	0.7477	0.7793	0.7158	<b>0.9660</b>	<b>0.8977</b>	<b>0.6707</b>	0.9622	0.8737
	2	3.6880	0.8926	0.5970	2.1222	0.6695	0.7692	1.8648	0.8562	0.8173	<b>1.0581</b>	<b>0.9011</b>	<b>0.9132</b>
	3	2.3698	0.8504	0.6832	2.4492	0.6481	0.7137	2.9895	0.7883	0.6245	<b>1.1930</b>	<b>0.8834</b>	<b>0.8722</b>
	4	1.5925	0.8520	0.7466	2.8253	0.6673	0.6613	3.9714	0.7817	0.5396	<b>1.0707</b>	0.9032	<b>0.8353</b>
	5	1.3390	0.8549	0.7159	3.2033	0.7014	0.5751	4.5455	0.7847	0.4934	<b>0.9570</b>	0.9221	<b>0.7719</b>
	6	1.2921	0.8653	0.5717	3.3798	0.7419	0.3630	4.6048	0.8135	0.3262	<b>0.8926</b>	0.9312	<b>0.6424</b>
	7	1.0475	0.9106	0.5999	3.1311	0.7927	0.4048	4.0762	0.8754	0.3910	<b>0.7580</b>	0.9528	<b>0.6244</b>
	8	1.0155	0.9333	0.6830	2.7128	0.8374	0.3621	3.2833	0.9171	0.3507	<b>0.6565</b>	0.9622	<b>0.7357</b>
	9	1.5885	0.9563	0.6038	1.8022	0.8878	0.5960	1.9514	0.9593	0.5063	<b>0.5706</b>	0.9707	<b>0.8522</b>
sp	1	601.1546	0.9995	0.5514	213.8882	0.9789	0.5857	54.4142	1.0000	0.5579	<b>43.3134</b>	<b>1.0000</b>	<b>0.7488</b>
	2	574.0437	1.0000	<b>0.7066</b>	291.1883	0.9665	0.4743	95.9910	1.0000	0.3097	<b>45.2443</b>	<b>1.0000</b>	0.6301
	3	577.7322	0.9995	<b>0.6853</b>	332.0390	0.9586	0.4924	106.1931	1.0000	0.3588	<b>50.4454</b>	<b>1.0000</b>	0.5900
	4	612.8599	0.9995	0.5415	342.1919	0.9538	0.5865	85.2841	1.0000	0.6814	<b>50.4404</b>	<b>1.0000</b>	<b>0.8187</b>
	5	692.4006	0.9996	0.4998	331.8958	0.9528	0.6310	43.0137	1.0000	0.9219	<b>42.4724</b>	<b>1.0000</b>	<b>0.9249</b>
	6	772.3997	0.9993	0.4552	321.0012	0.9551	0.6067	60.5968	1.0000	0.8132	<b>42.6370</b>	<b>1.0000</b>	<b>0.8731</b>
	7	823.8331	0.9995	0.3436	301.9203	0.9605	0.4881	92.0035	1.0000	0.4530	<b>44.0127</b>	<b>1.0000</b>	<b>0.6640</b>
	8	828.0718	0.9995	0.3129	254.3469	0.9687	0.4313	83.4018	1.0000	0.3403	<b>31.1195</b>	<b>1.0000</b>	<b>0.6137</b>
	9	808.3522	0.9997	0.4213	176.1976	0.9804	0.5637	51.4781	1.0000	0.5369	<b>25.7217</b>	<b>1.0000</b>	<b>0.7888</b>
u10	1	0.7206	0.8969	0.5296	0.3541	0.8414	0.6662	<b>0.1675</b>	<b>0.9707</b>	<b>0.7365</b>	0.2277	0.9615	0.6938
	2	0.7519	0.8562	0.5720	0.5140	0.7628	0.6389	0.3427	0.8930	0.6926	<b>0.2989</b>	0.9164	<b>0.7288</b>
	3	0.8301	0.7923	0.5837	0.6732	0.6961	0.6096	0.5580	0.7855	0.6236	<b>0.4082</b>	0.8697	<b>0.7025</b>
	4	0.9345	0.7407	0.5677	0.8230	0.6435	0.5817	0.7632	0.6907	0.5758	<b>0.4956</b>	0.8350	<b>0.6612</b>
	5	1.0434	0.6942	0.5269	0.9528	0.6010	0.5444	0.9320	0.6200	0.5365	<b>0.5547</b>	0.8073	<b>0.6318</b>
	6	1.1053	0.6683	0.5409	1.0307	0.5789	0.5269	1.0354	0.5877	0.5103	<b>0.5825</b>	0.8035	<b>0.6140</b>
	7	1.1292	0.6622	0.4875	1.0382	0.5912	0.4852	1.0537	0.5989	0.4878	<b>0.5661</b>	0.8222	<b>0.6156</b>
	8	1.0113	0.7222	0.5702	0.8881	0.6793	0.5037	0.9032	0.6896	0.5023	<b>0.4973</b>	0.8650	<b>0.6667</b>
	9	0.7572	0.8518	0.6548	0.5280	0.8496	0.6447	0.5177	0.8725	0.6185	<b>0.4129</b>	0.9001	<b>0.7440</b>
v10	1	0.4516	0.5686	0.6029	0.3477	0.7575	0.6811	<b>0.1937</b>	<b>0.9229</b>	<b>0.7413</b>	0.2197	0.8968	0.7184
	2	0.5329	0.4752	0.6358	0.5177	0.6553	0.6473	0.4067	0.7658	0.6728	<b>0.3195</b>	0.8345	<b>0.7335</b>
	3	0.6778	0.3672	0.6026	0.6932	0.5619	0.6012	0.6605	0.5808	0.5886	<b>0.4283</b>	0.7717	<b>0.6970</b>
	4	0.8171	0.2750	0.5467	0.8318	0.4925	0.5536	0.8492	0.4645	0.5344	<b>0.4990</b>	0.7442	<b>0.6532</b>
	5	0.8945	0.2287	0.5398	0.8957	0.4578	0.5359	0.9364	0.4090	0.5242	<b>0.5441</b>	0.7146	<b>0.6139</b>
	6	0.9157	0.2298	0.5402	0.9043	0.4577	0.5433	0.9482	0.4002	0.5334	<b>0.5600</b>	0.6873	<b>0.6183</b>
	7	0.8433	0.3255	0.5533	0.8540	0.4858	0.5136	0.8903	0.4472	0.5132	<b>0.5394</b>	0.7053	<b>0.5943</b>
	8	0.7578	0.3635	0.5749	0.7395	0.5502	0.4973	0.7571	0.5372	0.4957	<b>0.4731</b>	0.7560	<b>0.6288</b>
	9	0.6235	0.3879	0.6404	0.4616	0.7027	0.6458	0.4436	0.7355	0.6177	<b>0.3409</b>	0.8377	<b>0.7522</b>

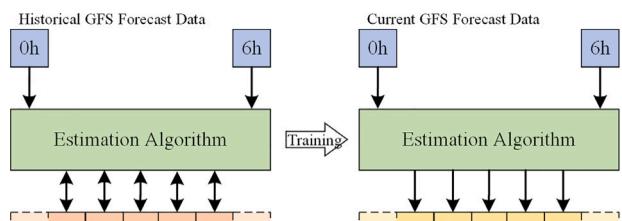
Note: The best results are highlighted in **bold**, the 1h, 3h, 5h, 7h, and 9h are the unseen moments during the training phase.

**Table 7**

Comparison result on multi moment task of 12h time interval.

Variable	Moment	SGM-12h5m			M2M-12h5m			Linear-12h			Ours-12h5m		
		MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑	MAE↓	SSIM↑	EDA↑
t2m	1	7.4920	0.6642	0.1131	5.4816	0.6354	0.3279	5.5391	0.6706	0.3352	<b>2.1016</b>	<b>0.7110</b>	<b>0.8134</b>
	2	6.4914	0.7258	0.4421	4.6174	0.6905	0.5056	4.6575	0.7242	0.5029	<b>1.9994</b>	<b>0.7576</b>	<b>0.7228</b>
	3	4.3045	0.7921	0.5557	2.7061	0.7532	0.7074	2.7031	0.7790	0.7081	<b>1.6146</b>	<b>0.8088</b>	<b>0.8430</b>
	4	2.5290	0.8235	0.6749	1.7649	0.7761	0.7921	1.7111	0.8001	0.7985	<b>1.4192</b>	<b>0.8407</b>	<b>0.8404</b>
	5	1.5096	0.8424	0.7670	1.9122	0.7836	0.7228	1.8432	0.8087	0.7288	<b>1.2811</b>	<b>0.8662</b>	<b>0.8072</b>
	6	1.4982	0.8504	0.6884	2.6539	0.7821	0.5823	2.5923	0.8137	0.5811	<b>1.2123</b>	<b>0.8896</b>	<b>0.7311</b>
	7	1.8296	0.8572	0.4456	3.2469	0.7852	0.3606	3.1904	0.8239	0.3586	<b>1.1916</b>	<b>0.9027</b>	<b>0.5412</b>
	8	1.8757	0.8835	0.4602	3.3497	0.8108	0.4038	3.3014	0.8540	0.3982	<b>1.0213</b>	<b>0.9221</b>	<b>0.5433</b>
	9	1.7915	0.8931	0.4606	3.2182	0.8291	0.3709	3.1688	0.8753	0.3644	<b>0.9303</b>	<b>0.9309</b>	<b>0.6269</b>
	10	1.3012	0.9091	0.7463	2.5758	0.8560	0.5379	2.5093	0.9027	0.5321	<b>0.8019</b>	<b>0.9367</b>	<b>0.8278</b>
	11	<b>0.7923</b>	0.9480	<b>0.8719</b>	1.4219	0.9084	0.7410	1.3075	0.9550	0.7414	0.9775	<b>0.9554</b>	0.8201
sp	1	856.7831	0.9964	0.3293	273.3334	0.9695	0.6126	162.0204	0.9963	<b>0.7505</b>	<b>156.2760</b>	<b>0.9977</b>	0.5189
	2	818.3229	0.9967	0.5435	316.7810	0.9609	0.5415	169.9293	0.9967	0.5606	112.6351	0.9981	<b>0.6230</b>
	3	793.4729	0.9968	<b>0.7096</b>	347.2539	0.9542	0.4582	179.5937	0.9974	0.4110	<b>93.3652</b>	<b>0.9989</b>	0.6028
	4	801.4712	0.9975	<b>0.6888</b>	358.4987	0.9504	0.4787	170.6698	0.9975	0.4190	<b>85.8436</b>	<b>0.9993</b>	0.5562
	5	843.3404	0.9979	0.5430	353.4687	0.9485	0.5746	143.0028	0.9980	0.6001	<b>78.8076</b>	<b>0.9994</b>	<b>0.7243</b>
	6	929.4317	0.9979	0.4998	340.8591	0.9486	0.6209	104.3467	0.9987	0.7294	<b>69.3656</b>	<b>0.9996</b>	<b>0.8239</b>
	7	1017.6462	0.9986	0.4534	340.6456	0.9504	0.5862	98.7113	0.9993	0.6970	71.9306	1.0000	0.7577
	8	1078.0106	0.9985	0.3441	341.3575	0.9541	0.4678	119.7444	0.9993	0.4702	<b>77.7959</b>	<b>0.9999</b>	<b>0.5342</b>
	9	1093.1340	0.9988	0.3081	317.8155	0.9595	0.4180	118.8121	0.9996	0.3569	<b>63.7493</b>	1.0000	<b>0.4626</b>
	10	1083.6385	0.9995	0.4277	271.7191	0.9670	0.4988	96.6742	1.0000	0.4796	<b>40.5005</b>	1.0000	0.7152
	11	1056.9491	0.9993	0.4713	199.7330	0.9764	0.6255	57.0590	1.0000	0.7265	<b>44.3816</b>	1.0000	<b>0.7790</b>
u10	1	1.2846	0.5661	0.4398	1.0677	0.4577	0.5374	1.0935	0.4500	0.5497	<b>0.6613</b>	<b>0.6414</b>	<b>0.6777</b>
	2	1.2586	0.5828	0.5215	1.0228	0.4753	0.5413	1.0333	0.4726	0.5424	<b>0.5893</b>	<b>0.6945</b>	<b>0.5980</b>
	3	1.2591	0.5610	0.5465	0.9876	0.4785	0.5829	0.9890	0.4778	0.5859	<b>0.6020</b>	<b>0.6987</b>	<b>0.6358</b>
	4	1.2788	0.5579	0.5558	0.9980	0.4837	0.5959	1.0053	0.4807	0.5954	<b>0.6650</b>	<b>0.6962</b>	<b>0.6350</b>
	5	1.3207	0.5551	0.5412	1.0668	0.4794	0.5760	1.0829	0.4748	0.5719	<b>0.7226</b>	<b>0.6934</b>	0.6183
	6	1.3822	0.5406	0.5090	1.1633	0.4671	0.5377	1.1828	0.4639	0.5329	<b>0.7615</b>	<b>0.6880</b>	0.6026
	7	1.4091	0.5320	0.5299	1.2320	0.4636	0.5189	1.2476	0.4676	0.5152	<b>0.7792</b>	<b>0.6933</b>	<b>0.5788</b>
	8	1.3930	0.5401	0.4745	1.2471	0.4806	0.4827	1.2555	0.4909	0.4785	<b>0.7507</b>	<b>0.7237</b>	<b>0.5899</b>
	9	1.2797	0.6017	0.5536	1.1274	0.5518	0.4959	1.1188	0.5742	0.4919	<b>0.6857</b>	<b>0.7729</b>	<b>0.6067</b>
	10	1.0290	0.7333	0.6088	0.8224	0.6935	0.5394	0.7839	0.7393	0.5278	<b>0.5616</b>	<b>0.8337</b>	<b>0.6821</b>
	11	0.8527	0.8688	0.6128	0.4452	0.8611	0.6845	0.3586	<b>0.9266</b>	0.6965	0.4644	0.8824	<b>0.7095</b>
v10	1	1.3170	-0.1687	0.5522	1.0943	0.1679	0.5590	1.1404	0.1465	0.5728	<b>0.7048</b>	<b>0.4288</b>	<b>0.6723</b>
	2	1.3224	-0.1899	0.5402	1.0437	0.1913	0.5650	1.0681	0.1726	0.5674	<b>0.6617</b>	<b>0.4612</b>	<b>0.5988</b>
	3	1.3788	-0.2041	0.5457	1.0215	0.2235	0.5955	1.0332	0.2072	0.5965	<b>0.6872</b>	<b>0.4741</b>	<b>0.6336</b>
	4	1.4688	-0.2039	0.5271	1.0611	0.2537	0.5800	1.0736	0.2386	0.5777	<b>0.7453</b>	<b>0.4890</b>	<b>0.6253</b>
	5	1.5504	-0.1836	0.4936	1.1216	0.2643	0.5395	1.1319	0.2513	0.5342	<b>0.7901</b>	0.5051	<b>0.5918</b>
	6	1.5462	-0.1419	0.4970	1.1408	0.2736	0.5256	1.1441	0.2643	0.5222	<b>0.7959</b>	0.5087	0.5721
	7	1.4665	-0.0913	0.4909	1.1218	0.2942	0.5339	1.1151	0.2894	0.5333	<b>0.7769</b>	0.5141	<b>0.5858</b>
	8	1.2691	0.0129	0.5359	1.0746	0.3326	0.5083	1.0539	0.3341	0.5057	<b>0.7164</b>	<b>0.5818</b>	<b>0.5657</b>
	9	1.1295	0.0603	0.5576	0.9868	0.3743	0.4927	0.9490	0.3941	0.4886	<b>0.6462</b>	<b>0.6321</b>	<b>0.5879</b>
	10	0.9951	0.0711	0.5954	0.7641	0.4680	0.5358	0.7011	0.5133	0.5309	<b>0.5241</b>	<b>0.7017</b>	<b>0.6761</b>
	11	0.9827	-0.0166	0.5588	0.4481	0.6801	0.6882	<b>0.3498</b>	<b>0.7558</b>	<b>0.7025</b>	0.4575	0.6963	0.6854

Note: The best results are highlighted in bold, the 1h, 3h, 5h, 7h, 9h, and 11h are the unseen moments during the training phase.

**Fig. 13.** The training and apply mode of engineering application experiment 2.

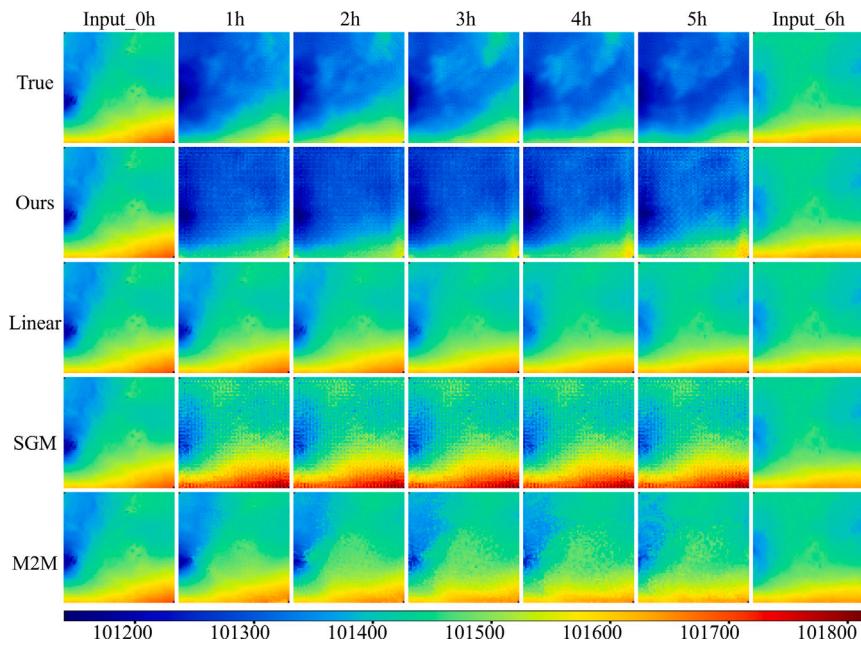
comparison methods, the proposed method and the SGM estimates this trend from the 5th to the 9th hour, but SGM produces large estimation error, proving the effectiveness of the proposed MSF-TCMA.

### 3.5.2. Sensitivity analysis of MSF-TCMA

To evaluate the sensitivity of the proposed MSF-TCMA to different temporal intervals and input datasets with varying temporal resolutions

in multi-moment downscaling tasks, **Tables 8–10** report the downscaling MAE results of the proposed method under various training configurations. **Fig. 12** presents the MAE curves of U10 across the temporal intervals for the three downscaling tasks under different training conditions.

**Fig. 12(a), (b), and (c)** show the U10 MAE curves of MSF-TCMA for the 6h, 10h, and 12h tasks, respectively, under different training conditions. First, it can be observed that the overall downscaling MAE of the MSF-TCMA increases as the temporal interval becomes longer. This suggests that as the interval increases, it becomes more difficult for the time-continuous manifold to accurately fit the meteorological evolution within the interval. Second, more seen moments during the training phase (i.e., improving the temporal resolution of the input dataset) significantly enhance the estimation accuracy. On one hand, more seen moments provide the model with additional information during training, ensuring that the latent features are mapped into a space exhibiting linear evolution. On the other hand, increasing the number of seen moments effectively reduces the temporal downscaling ratio, thus lowering the task complexity. Third, despite introducing multiple anchor moments as optimization targets, the estimation accuracy for intermediate moments within the interval is nearly the same as the single-moment downscaling results for the 6h task shown in



**Fig. 14.** The future hourly downscaling results for sp by comparative methods.

**Table 8**  
MAE results of the sensitivity experiment on the 6h multi-moment task.

Training mode	Variable	Moment				
		1	2	3	4	5
Ours-6h1m	t2m	<b>2.6212</b>	<b>1.3040</b>	<b>0.6387</b>	<b>0.9357</b>	<b>1.4275</b>
	sp	<b>101.9375</b>	<b>60.2816</b>	<b>31.3900</b>	<b>58.8675</b>	<b>107.5420</b>
	u10	0.5094	0.4050	0.3803	0.4604	0.6211
	v10	0.5088	0.4148	0.3805	0.4375	0.5613
Ours-6h2m	t2m	<b>0.6762</b>	0.7626	<b>0.7408</b>	0.5683	<b>0.5127</b>
	sp	<b>44.2526</b>	46.3244	<b>56.2596</b>	47.5090	<b>43.2723</b>
	u10	<b>0.2524</b>	0.3292	<b>0.3910</b>	0.3779	<b>0.3101</b>
	v10	<b>0.2622</b>	0.3317	<b>0.3976</b>	0.3730	<b>0.3066</b>
Ours-6h3m	t2m	0.5304	<b>0.7013</b>	0.6512	<b>0.5276</b>	<b>0.3616</b>
	sp	<b>25.6663</b>	<b>43.5041</b>	49.5610	<b>45.3849</b>	<b>25.0675</b>
	u10	<b>0.1914</b>	<b>0.3125</b>	0.3702	<b>0.3603</b>	<b>0.2536</b>
	v10	<b>0.1915</b>	<b>0.3154</b>	0.3722	<b>0.3613</b>	<b>0.2468</b>
Ours-6h5m	t2m	<b>0.5255</b>	<b>0.6755</b>	0.6458	<b>0.5235</b>	0.3671
	sp	25.7088	<b>40.3360</b>	46.1776	<b>40.3001</b>	25.1633
	u10	0.1934	<b>0.3079</b>	<b>0.3697</b>	0.3610	0.2597
	v10	0.1955	<b>0.3100</b>	<b>0.3694</b>	<b>0.3543</b>	0.2529

Note: The best results are highlighted in bold, Blue font denotes unseen moments during the training phase.

Section 3.3.1. This validates the rationality of employing the time-continuous manifold for continuous temporal downscaling. Finally, in the 12h task, the MSF-TCMA trained with only two anchor points achieves better downscaling results than comparative methods trained with six anchor points, further demonstrating the superiority of the proposed method.

### 3.6. Engineering application experiment 2—Estimating future meteorological states by GFS data on dataset 1

In practical engineering applications, such as weather forecasting, route planning, and activity rehearsal, it is necessary to obtain future meteorological states with high temporal resolution. However, the GFS can only provide forecast data with a temporal resolution of 6 h. This experiment developed a mapping model to convert the 6-h temporal resolution GFS forecast products into 1-h temporal resolution ERA5 reanalysis data, enabling high-resolution estimates of future meteorological states. Specifically, GFS forecast data, which is released daily at

12:00 to predict weather conditions at 0:00, 6:00, 12:00, and 18:00 of the following day was input into the algorithm. The algorithm's output is hourly reanalysis data for the next day, provided by ERA5. This allows for the estimation of future meteorological evolution using the available GFS forecast data. The process is shown in Fig. 13.

Table 11 shows the performance of the four methods in this experiment. Fig. 14 shows the estimation results of different methods for the samples from 0:00 to 6:00 on November 12 in the test set. Since the GFS and ERA5 differ in data sources and assimilation methods, their data distributions also vary. As a result, the linear interpolation method cannot effectively capture the mapping relationship between the two datasets. Similarly, SGM and M2M also failed to deliver satisfactory results. The proposed method, however, demonstrated high accuracy, with the estimated MAE for t2m less than 0.4702 K, the estimated error for sp less than 70 Pa, and the estimated errors for u10 and v10 less than 1.44 m/s. These results demonstrate that the proposed method can provide high-accuracy, high-temporal-resolution meteorological evolution over the next 48 h, thereby providing valuable

**Table 9**

MAE results of the sensitivity experiment on the 10h multi-moment task.

Training mode	Variable	Moment								
		1	2	3	4	5	6	7	8	9
Ours-10h1m	t2m	<b>6.9376</b>	<b>4.9024</b>	<b>3.0942</b>	<b>1.6250</b>	1.0247	<b>1.0953</b>	<b>1.1175</b>	<b>1.1807</b>	<b>1.3541</b>
	sp	<b>122.2555</b>	<b>137.9244</b>	<b>133.3715</b>	<b>98.6896</b>	<b>40.9839</b>	<b>69.9806</b>	<b>120.0603</b>	<b>129.5136</b>	<b>122.1625</b>
	u10	<b>1.0123</b>	<b>0.8878</b>	<b>0.7332</b>	<b>0.6130</b>	0.5813	<b>0.6388</b>	<b>0.7379</b>	<b>0.8186</b>	<b>0.9255</b>
	v10	<b>0.8519</b>	<b>0.7302</b>	<b>0.6107</b>	<b>0.5547</b>	0.5453	<b>0.5805</b>	<b>0.6611</b>	<b>0.7352</b>	<b>0.8136</b>
Ours-10h2m	t2m	<b>4.1673</b>	<b>2.6166</b>	<b>1.6473</b>	1.1637	<b>1.1824</b>	<b>1.2333</b>	<b>0.9304</b>	0.6979	<b>0.7204</b>
	sp	<b>296.5085</b>	<b>174.5012</b>	<b>104.3347</b>	74.5673	<b>55.0269</b>	<b>53.9648</b>	<b>52.0440</b>	35.7104	<b>49.5658</b>
	u10	<b>0.6345</b>	<b>0.5745</b>	<b>0.5473</b>	0.5583	<b>0.5998</b>	<b>0.6369</b>	<b>0.6284</b>	0.5757	<b>0.6390</b>
	v10	<b>0.8748</b>	<b>0.6580</b>	<b>0.5307</b>	0.5441	0.6000	<b>0.6252</b>	<b>0.5843</b>	0.4995	<b>0.3767</b>
Ours-10h3m	t2m	<b>1.5327</b>	<b>1.2435</b>	1.2354	<b>1.0966</b>	<b>0.9926</b>	0.9051	<b>0.7729</b>	<b>0.6428</b>	<b>0.5052</b>
	sp	<b>128.3693</b>	<b>81.2814</b>	71.0030	<b>68.3131</b>	<b>49.3753</b>	43.3467	<b>48.0763</b>	<b>39.1247</b>	<b>20.8740</b>
	u10	<b>0.3394</b>	<b>0.3492</b>	0.4249	<b>0.5169</b>	<b>0.5804</b>	0.6107	<b>0.6054</b>	<b>0.5379</b>	<b>0.3373</b>
	v10	<b>0.3837</b>	<b>0.3648</b>	0.4381	<b>0.5172</b>	<b>0.5551</b>	0.5632	<b>0.5366</b>	<b>0.4653</b>	<b>0.2842</b>
Ours-10h4m	t2m	<b>0.6707</b>	<b>1.0581</b>	<b>1.1930</b>	<b>1.0707</b>	<b>0.9570</b>	<b>0.8926</b>	<b>0.7580</b>	0.6565	<b>0.5706</b>
	sp	<b>43.3134</b>	<b>45.2443</b>	<b>50.4454</b>	<b>50.4404</b>	<b>42.4724</b>	<b>42.6370</b>	<b>44.0127</b>	<b>31.1195</b>	<b>25.7217</b>
	u10	<b>0.2277</b>	<b>0.2989</b>	<b>0.4082</b>	<b>0.4956</b>	<b>0.5547</b>	<b>0.5825</b>	<b>0.5661</b>	<b>0.4973</b>	<b>0.4129</b>
	v10	<b>0.2197</b>	<b>0.3195</b>	<b>0.4283</b>	<b>0.4990</b>	<b>0.5441</b>	<b>0.5600</b>	<b>0.5394</b>	0.4731	<b>0.3409</b>

Note: The best results are highlighted in bold, Blue font denotes unseen moments during the training phase.

**Table 10**

MAE results of the sensitivity experiment on the 12h multi-moment task.

Training mode	Variable	Moment										
		1	2	3	4	5	6	7	8	9	10	
Ours-12h1m	t2m	<b>6.3295</b>	<b>5.5641</b>	<b>3.5645</b>	<b>1.9841</b>	<b>1.2513</b>	1.5072	<b>1.9297</b>	<b>1.9709</b>	<b>1.8585</b>	<b>1.4611</b>	<b>1.2644</b>
	sp	<b>114.9391</b>	<b>124.5962</b>	<b>138.3755</b>	<b>133.2224</b>	<b>106.4907</b>	75.4923	<b>97.2392</b>	<b>134.9866</b>	<b>141.6206</b>	<b>130.6427</b>	<b>115.7604</b>
	u10	<b>1.2331</b>	<b>1.1378</b>	<b>0.9935</b>	<b>0.8449</b>	0.7619	0.7680	<b>0.8296</b>	<b>0.9006</b>	0.9257	0.9326	0.9907
	v10	<b>1.2690</b>	<b>1.1526</b>	<b>0.9892</b>	<b>0.8581</b>	0.8119	0.8076	<b>0.8300</b>	<b>0.8635</b>	0.8697	0.8017	0.7106
Ours-12h2m	t2m	<b>3.2244</b>	<b>3.0150</b>	<b>1.9203</b>	1.4676	<b>1.4274</b>	<b>1.5122</b>	<b>1.5284</b>	1.2720	<b>1.0914</b>	1.0868	<b>1.7886</b>
	sp	<b>195.9919</b>	<b>136.2277</b>	<b>97.3759</b>	85.9108	<b>80.4271</b>	<b>73.0213</b>	<b>76.7644</b>	<b>76.7917</b>	<b>58.2716</b>	64.6343	<b>132.8017</b>
	u10	<b>0.7811</b>	<b>0.7134</b>	<b>0.6640</b>	0.6735	0.7380	<b>0.8017</b>	<b>0.8231</b>	0.7978	<b>0.7805</b>	0.8835	1.1303
	v10	<b>0.9766</b>	<b>0.8767</b>	<b>0.7713</b>	0.7482	0.7864	<b>0.8096</b>	<b>0.8031</b>	0.7437	<b>0.6848</b>	0.6846	0.8745
Ours-12h3m	t2m	<b>2.4060</b>	<b>2.3534</b>	1.6835	<b>1.4520</b>	<b>1.3147</b>	<b>1.2016</b>	<b>1.1712</b>	<b>1.0417</b>	0.9821	<b>0.8469</b>	<b>1.0307</b>
	sp	<b>152.3971</b>	<b>118.0500</b>	100.1561	<b>91.5643</b>	<b>81.8105</b>	75.5292	<b>82.0994</b>	<b>85.6221</b>	65.9620	<b>41.9667</b>	<b>59.3377</b>
	u10	<b>0.7620</b>	0.6645	0.6289	<b>0.6667</b>	<b>0.7174</b>	<b>0.7504</b>	<b>0.7650</b>	<b>0.7450</b>	<b>0.6822</b>	0.5801	0.6360
	v10	<b>0.7680</b>	<b>0.7171</b>	0.7045	<b>0.7464</b>	<b>0.7814</b>	<b>0.7883</b>	<b>0.7818</b>	<b>0.7360</b>	0.6572	<b>0.5741</b>	0.6281
Ours-12h5m	t2m	<b>2.1016</b>	<b>1.9994</b>	<b>1.6146</b>	<b>1.4192</b>	1.2811	1.2123	<b>1.1916</b>	<b>1.0213</b>	<b>0.9303</b>	0.8019	<b>0.9775</b>
	sp	<b>156.2760</b>	<b>112.6351</b>	<b>93.3652</b>	<b>85.8436</b>	<b>78.8076</b>	<b>69.3656</b>	<b>71.9306</b>	77.7959	<b>63.7493</b>	40.5005	<b>44.3816</b>
	u10	<b>0.6613</b>	<b>0.5893</b>	<b>0.6020</b>	0.6650	0.7226	0.7615	<b>0.7792</b>	0.7507	<b>0.6857</b>	0.5616	<b>0.4644</b>
	v10	<b>0.7048</b>	<b>0.6617</b>	<b>0.6872</b>	0.7453	0.7901	0.7959	<b>0.7769</b>	0.7164	<b>0.6462</b>	0.5241	<b>0.4575</b>

Note: The best results are highlighted in bold, Blue font denotes unseen moments during the training phase.

meteorological references for practical applications such as mid-term flight route planning.

#### 4. Conclusion

To overcome the challenge of limited data distribution extrapolation capability, insufficient multi-source data fusion, and inadequate learning of spatiotemporal dependencies, which cause the limited temporal resolution and low accuracy of existing downscaling methods. This paper proposes the MSF-TCMA, which solves the problem of data distribution extrapolation by mapping the nonlinear meteorological evolution to the time-continuous streaming space, and realizes the extrapolation of continuous meteorological states at any moment (including unseen moments) within the time interval. Then the MSF-TCMA solved the problems of spatiotemporal dependent characterization learning and the lack of multi-source data fusion by multi-scale depth-wavelet feature extraction branch and the cross-modal spatial-temporal information fusion branch and achieved the accurate downscaling performance. Furthermore, multi-moment weighting loss, state estimation loss and energy change deduction loss are introduced to further optimize the algorithm performance.

To fully verify the effectiveness of MSF-TCMA in industrial applications, downscaling tasks in the Northern Indian Ocean and Southwestern China demonstrate that the single-time downscaling performance of the algorithm is improved by 3.99–99.64% compared to other methods. Moreover, two engineering application experiments confirm that the proposed method can provide reliable meteorological state evolution data with high accuracy for multi-moment (including unseen moment) downscaling tasks and future meteorological state downscaling tasks.

Through MSF-TCMA, a reliable and fine-temporal-grained meteorological state evolution model is provided to meet the demand for high-temporal-resolution meteorological data, supporting applications in aerospace, construction, virtual experiments, and other fields. However, several issues remain to be addressed in future research. First, the acquisition of high-temporal-resolution validation data remains challenging, as this study is constrained by the resolution of available datasets. We evaluated the model's performance for unseen moment estimation by training on temporally downsampled datasets and testing on the original-resolution data. Future research should explore more accessible high-temporal-resolution datasets to validate meteorological evolution at finer temporal scales. In addition, the accuracy of temporal downscaling for future meteorological evolution still needs improvement. This requires the development of more precise forecasting models and the accumulation of more meteorological data.

**Table 11**

MAE results of the future multi-moment downscaling task based on the GFS product.

Variable	Methods	Moment				
		1h	2h	3h	4h	5h
t2m	SGM-VFI	0.7233	0.6764	0.6337	0.5957	0.5739
	M2M-VFI	0.6140	0.5652	0.5356	0.5174	0.5230
	Linear	0.6699	0.6427	0.6215	0.6092	0.6164
	Ours	<b>0.4667</b>	<b>0.4562</b>	<b>0.4515</b>	<b>0.4548</b>	<b>0.4702</b>
sp	SGM-VFI	438.3244	373.6342	340.8405	342.0079	377.5028
	M2M-VFI	411.1263	344.0945	310.3100	311.7499	351.0490
	Linear	418.4802	355.4056	324.3217	327.1814	364.4002
	Ours	<b>47.3769</b>	<b>55.2112</b>	<b>58.1009</b>	<b>52.4727</b>	<b>70.0452</b>
u10	SGM-VFI	1.3756	1.3762	1.3791	1.3891	<b>1.4031</b>
	M2M-VFI	2.4786	2.5353	2.5682	2.5622	2.5363
	Linear	2.4684	2.4744	2.4789	2.4887	2.5016
	Ours	<b>1.3287</b>	<b>1.3042</b>	<b>1.3043</b>	<b>1.3422</b>	1.4428
v10	SGM-VFI	1.4485	1.4481	1.4322	1.4176	1.4096
	M2M-VFI	1.6353	1.6558	1.6730	1.6694	1.6482
	Linear	1.6711	1.6714	1.6585	1.6464	1.6405
	Ours	<b>1.2824</b>	<b>1.2493</b>	<b>1.2467</b>	<b>1.2741</b>	<b>1.3407</b>

Note: The best results are highlighted in **bold**.

Therefore, future studies will focus on the accurate prediction of future meteorological states. Moreover, spatial downscaling is also a key objective of our research. Through these efforts, we aim to provide high-spatiotemporal resolution and accurate meteorological data to comprehensively support the meteorological demands of engineering applications.

#### CRediT authorship contribution statement

**Sheng Gao:** Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Data curation, Conceptualization. **Lianlei Lin:** Supervision, Methodology, Data curation, Conceptualization. **Zongwei Zhang:** Writing – original draft, Visualization, Software, Methodology. **Jiawei Wang:** Writing – original draft, Data curation, Conceptualization.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### References

- Al Kajbaf, A., Bensi, M., Brubaker, K.L., 2022. Temporal downscaling of precipitation from climate model projections using machine learning. Stoch. Environ. Res. Risk Assess. 36 (8), 2173–2194. <http://dx.doi.org/10.1007/s00477-022-02259-2>, URL: <Go to ISI>/WOS:000825008700001, Sp. Iss. SI 411sv Times Cited:10 Cited References Count:50.
- Balog, I., Caputo, G., Iatauro, D., Signoretti, P., Spinelli, F., 2023. Downscaling of hourly climate data for the assessment of building energy performance. Sustain. 15 (3), 2762.
- Buster, G., Rossol, M., Maclaurin, G., Xie, Y., Sengupta, M., 2021. A physical downscaling algorithm for the generation of high-resolution spatiotemporal solar irradiance data. Sol. Energy 216, 508–517.
- Chen, H.R., Qin, H.W., Dai, Y.W., 2022. Fc-zsm: Spatiotemporal downscaling of rain radar data using a feature constrained zooming slow-mo network. Front. Earth Sci. 10, <http://dx.doi.org/10.3389/feart.2022.887842>, URL: <Go to ISI>/WOS:000811309800001, 2d1io Times Cited:4 Cited References Count:39.
- Choi, M., Kim, H., Han, B., Xu, N., Lee, K.M., 2020. Channel attention is all you need for video frame interpolation. In: Thirty-Fourth Aaai Conference on Artificial Intelligence, the Thirty-Second Innovative Applications of Artificial Intelligence Conference and the Tenth Aaai Symposium on Educational Advances in Artificial Intelligence. Vol. 34, pp. 10663–10671, URL: <Go to ISI>/WOS:000668126803014, Br7jr Times Cited:197 Cited References Count:49 AAI Conference on Artificial Intelligence.
- contributor, 2015. Ncep gfs 0.25 degree global forecast grids historical archive. URL: <https://rda.ucar.edu/datasets/dsd084001/>.
- Dong, J., Ota, K., Dong, M., 2023. Video frame interpolation: A comprehensive survey. ACM Trans. Multimed. Comput. Commun. Appl. 19 (2, S), <http://dx.doi.org/10.1145/3556544>.
- Faticchi, S., Ivanov, V.Y., Caporali, E., 2013. Assessment of a stochastic downscaling methodology in generating an ensemble of hourly future climate time series. Clim. Dyn. 40 (7–8), 1841–1861. <http://dx.doi.org/10.1007/s00382-012-1627-2>, URL: <Go to ISI>/WOS:000317075700018, 119cv Times Cited:80 Cited References Count:74.
- Gao, S.H., Cheng, M.M., Zhao, K., Zhang, X.Y., Yang, M.H., Torr, P., 2021. Res2Net: A new multi-scale backbone architecture. IEEE Trans. Pattern Anal. Mach. Intell. 43 (2), 652–662. <http://dx.doi.org/10.1109/TPAMI.2019.2938758>, URL: <Go to ISI>/WOS:000607383300018, Pr6zz Times Cited:1969 Cited References Count:68.
- Gao, S., Lin, L., Zhang, Z., Wang, J., Zhao, H., Yu, H., 2025. Local spatial self-attention based deep network for meteorological data downscaling. Neurocomputing 649, 130653. <http://dx.doi.org/10.1016/j.neucom.2025.130653>, URL: <https://www.sciencedirect.com/science/article/pii/S0925231225013256>.
- Gao, S., Meng, G., Lin, L.L., Zhang, Z.W., Wang, J.K., Zhao, H.Q., 2024. Spatiotemporal multiwavenet for efficiently generating environmental spatiotemporal series. Ieee Trans. Geosci. Remote. Sens. 62, <http://dx.doi.org/10.1109/Tgrs.2024.3424241>, URL: <Go to ISI>/WOS:001289662900033, C5h0z Times Cited:0 Cited References Count:36.
- Glawion, L., Polz, J., Kunstmann, H., Fersch, B., Chwala, C., 2025. Global spatio-temporal ERA5 precipitation downscaling to km and sub-hourly scale using generative AI. Npj Clim. Atmospheric Sci. 8 (1), 219.
- Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., et al., 2023. ERA5 hourly data on single levels from 1940 to present, copernicus climate change service (C3S) climate data store (CDS)[data set].
- Hou, Y.K., Chen, H., Xu, C.Y., Chen, J., Guo, S.L., 2017. Coupling a Markov chain and support vector machine for at-site downscaling of daily precipitation. J. Hydrometeorol. 18 (9), 2385–2406. <http://dx.doi.org/10.1175/Jhm-D-16-0130.1>, URL: <Go to ISI>/WOS:000417351800004, Fp1ds Times Cited:12 Cited References Count:75.
- Hu, P., Niklaus, S., Sclaroff, S., Saenko, K., 2022. Many-to-many splatting for efficient video frame interpolation. In: 2022 Ieee/Cvf Conference on Computer Vision and Pattern Recognition. Cvpr 2022, pp. 3543–3552. <http://dx.doi.org/10.1109/Cvpr52688.2022.00354>, URL: <Go to ISI>/WOS:000867754203077, Bu0eq Times Cited:28 Cited References Count:51 IEEE Conference on Computer Vision and Pattern Recognition.
- Huang, Z., Zhang, T., Heng, W., Shi, B., Zhou, S., 2022. Real-time intermediate flow estimation for video frame interpolation. In: European Conference on Computer Vision. pp. 624–642.
- Jennings, S.A., Lambert, M.F., Kuczera, G., 2010. Generating synthetic high resolution rainfall time series at sites with only daily rainfall using a master-target scaling approach. J. Hydrol. 393 (3–4), 163–173. <http://dx.doi.org/10.1016/j.jhydrol.2010.08.013>, URL: <Go to ISI>/WOS:000284343400003, 681tu Times Cited:13 Cited References Count:21.
- Lee, H., Kim, T., Chung, T.Y., Pak, D., Ban, Y., Lee, S., 2020. AdaCoF: Adaptive collaboration of flows for video frame interpolation. In: 2020 Ieee/Cvf Conference on Computer Vision and Pattern Recognition. Cvpr, pp. 5315–5324. <http://dx.doi.org/10.1109/Cvpr42600.2020.00536>, URL: <Go to ISI>/WOS:000620679505059, Bq8ld Times Cited:163 Cited References Count:51 IEEE Conference on Computer Vision and Pattern Recognition.
- Li, L.F., 2019. Geographically weighted machine learning and downscaling for high-resolution spatiotemporal estimations of wind speed. Remote. Sens. 11 (11), <http://dx.doi.org/10.3390/rs11111378>, URL: <Go to ISI>/WOS:000472648000121, Ie8uc Times Cited:46 Cited References Count:51.
- Liang, J.Y., Cao, J.Z., Sun, G.L., Zhang, K., Van Gool, L., Timofte, R., 2021. SwinIR: Image restoration using swin transformer. In: 2021 Ieee/Cvf International Conference on Computer Vision Workshops. Iccvw 2021, pp. 1833–1844. <http://dx.doi.org/10.1109/Iccvw54120.2021.00210>, URL: <Go to ISI>/WOS:000739651101103, Bs5qb Times Cited:2055 Cited References Count:99 IEEE International Conference on Computer Vision Workshops.
- Liu, C., Zhang, G., Zhao, R., Wang, L., 2024. Sparse global matching for video frame interpolation with large motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19125–19134.
- McDowell, T., Kummert, M., 2016. Estimating sub-hourly solar radiation and effective sky temperature from hourly weather data. In: ASHRAE & IBPSA-USA SimBuild Conference 2016. Vol. 7, ASHRAE/IBPSA-USA, pp. 423–430.
- McDowell, T.P., Letellier-Duchesne, S., Kummert, M., 2018. A new method for determining sub-hourly solar radiation from hourly data. In: 2018 Building Performance Analysis Conference and Simbuild Co-Organized By ASHRAE and IBPSA-USA Chicago, IL, September. pp. 26–28.
- Min, M., Wu, C.Q., Li, C., Liu, H., Xu, N., Wu, X., Chen, L., Wang, F., Sun, F.L., Qin, D.Y., Wang, X., Li, B., Zheng, Z.J., Cao, G.Z., Dong, L.X., 2017. Developing the science product algorithm testbed for Chinese next-generation geostationary meteorological satellites: Fengyun-4 series. J. Meteorol. Res. 31 (4), 708–719. <http://dx.doi.org/10.1007/s13351-017-6161-z>, URL: <Go to ISI>/WOS:000408337200006, Fe6rs Times Cited:146 Cited References Count:32.

- Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., Boussetta, S., Choulga, M., Harrigan, S., Hersbach, H., et al., 2021. ERA5-land: A state-of-the-art global reanalysis dataset for land applications. *Earth Syst. Sci. Data* 13 (9), 4349–4383.
- Nguyen, D.T., Chen, S.T., 2022. Generating continuous rainfall time series with high temporal resolution by using a stochastic rainfall generator with a copula and modified huff rainfall curves. *Water* 14 (13), <http://dx.doi.org/10.3390/w14132123>, URL: <Go to ISI>://WOS:000825499700001, 2x9ei Times Cited:2 Cited References Count:64.
- Omoyele, O., Hoffmann, M., Koivisto, M., Larraneta, M., Weinand, J.M., Linßen, J., Stolten, D., 2024. Increasing the resolution of solar and wind time series for energy system modeling: A review. *Renew. Sustain. Energy Rev.* 189, 113792.
- Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention.
- Ruiz-Arias, J.A., 2022. Mean-preserving interpolation with splines for solar radiation modeling. *Sol. Energy* 248, 121–127.
- Serifi, A., Günther, T., Ban, N., 2021. Spatio-temporal downscaling of climate data using convolutional and error-predicting neural networks. *Front. Clim.* 3, <http://dx.doi.org/10.3389/fclim.2021.656479>, URL: <Go to ISI>://WOS:001023296700001, L4wy1 Times Cited:29 Cited References Count:47.
- Sun, Y., Deng, K., Ren, K., Liu, J., Deng, C., Jin, Y., 2024b. Deep learning in statistical downscaling for deriving high spatial resolution gridded meteorological data: A systematic review. *ISPRS J. Photogramm. Remote Sens.* 208, 14–38. <http://dx.doi.org/10.1016/j.isprsjprs.2023.12.011>, URL: <https://www.sciencedirect.com/science/article/pii/S0924271623003489>.
- Sun, F., Li, L., Bian, D., Bian, W., Wang, Q., Wang, S., 2025. Photovoltaic power prediction based on multi-scale photovoltaic power fluctuation characteristics and multi-channel LSTM prediction models. *Renew. Energy* 246, <http://dx.doi.org/10.1016/j.renene.2025.122866>.
- Sun, L., Zhang, X., Wang, H., Xiao, P., Wang, Y., 2024a. Estimating daily snow density through a spatiotemporal random forest model. *Water Resour. Res.* 60 (7), <http://dx.doi.org/10.1029/2023WR036942>.
- Tatsubori, M., Moriyama, T., Ishikawa, T., Fraccaro, P., Jones, A., Edwards, B., Kuehnert, J., Remy, S.L., 2022. Deep temporal interpolation of radar-based precipitation. In: 2022 Ieee International Conference on Acoustics, Speech and Signal Processing. Icassp, pp. 1685–1689. <http://dx.doi.org/10.1109/Icassp43922.2022.9747829>, URL: <Go to ISI>://WOS:000864187901192, Bt9ub Times Cited:0 Cited References Count:17 International Conference on Acoustics Speech and Signal Processing ICASSP.
- Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* 13 (4), 600–612.
- Wang, L., Li, Q., Lv, Q., Peng, X., You, W., 2025. TemDeep: a self-supervised framework for temporal downscaling of atmospheric fields at arbitrary time resolutions. *Geosci. Model. Dev.* 18 (8), 2427–2442. <http://dx.doi.org/10.5194/gmd-18-2427-2025>, URL: <https://gmd.copernicus.org/articles/18/2427/2025/>.
- Wang, L.W., Li, Q., Peng, X., Lv, Q., 2024. A temporal downscaling model for gridded geophysical data with enhanced residual U-Net. *Remote. Sens.* 16 (3), <http://dx.doi.org/10.3390/rs16030442>, URL: <Go to ISI>://WOS:001160089700001, Hn1k3 Times Cited:2 Cited References Count:54.
- Wang, Y., Liang, D.Q., Wang, B.S., 2006. Daubechies wavelet construction using continuation method. In: Wavelet Active Media Technology and Information Processing. Vol. 1 and 2, pp. 813–+, URL: <Go to ISI>://WOS:000240252100122, Bez27 Times Cited:0 Cited References Count:8.
- Wang, F., Tian, D., Lowe, L., Kalin, L., Lehrter, J., 2021. Deep learning for daily precipitation and temperature downscaling. *Water Resour. Res.* 57 (4), <http://dx.doi.org/10.1029/2020WR029308>, URL: <Go to ISI>://WOS:000644063800057, Rs9an Times Cited:96 Cited References Count:56.
- Xiang, L., Guan, J.P., Xiang, J., Zhang, L.F., Zhang, F.H., 2022. Spatiotemporal model based on transformer for bias correction and temporal downscaling of forecasts. *Front. Environ. Sci.* 10, <http://dx.doi.org/10.3389/fenvs.2022.1039764>, URL: <Go to ISI>://WOS:000895527600001, 6w1xk Times Cited:5 Cited References Count:46.
- Xiang, X.Y., Tian, Y.P., Zhang, Y.L., Fu, Y., Allebach, J.P., Xu, C.L., 2020. Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In: 2020 Ieee/Cvf Conference on Computer Vision and Pattern Recognition. Cvpr, pp. 3367–3376. <http://dx.doi.org/10.1109/Cvpr42600.2020.00343>, URL: <Go to ISI>://WOS:000620679503062, Bq8ld Times Cited:135 Cited References Count:42 IEEE Conference on Computer Vision and Pattern Recognition.
- Xu, H.F., Zhang, J., Cai, J.F., Rezatofighi, H., Tao, D.C., 2022. Gmflow: Learning optical flow via global matching. In: 2022 Ieee/Cvf Conference on Computer Vision and Pattern Recognition. Cvpr, pp. 8111–8120. <http://dx.doi.org/10.1109/Cvpr52688.2022.00795>, URL: <Go to ISI>://WOS:000870759101017, Bu0l Times Cited:126 Cited References Count:52 IEEE Conference on Computer Vision and Pattern Recognition.
- Yi, Z., Wang, Y., Zeng, Z., Li, W., Che, H., Zhang, X., 2025. Decadal trends and drivers of dust emissions in east Asia: Integrating statistical and shap-based interpretability approaches. *Remote. Sens.* 17 (7), 1313.
- Zhang, Z.W., Lin, L.L., Gao, S., Wang, J.K., Zhao, H.Q., 2024. Wind speed prediction in China with fully-convolutional deep neural network. *Renew. Sustain. Energy Rev.* 201, <http://dx.doi.org/10.1016/j.rser.2024.114623>, URL: <Go to ISI>://WOS:001256427100001, Wq8k1 Times Cited:5 Cited References Count:48.
- Zhang, Z., Lin, L., Gao, S., Wang, J., Zhao, H., Yu, H., 2025. A machine learning model for hub-height short-term wind speed prediction. *Nat. Commun.* 16 (1), 3195.
- Zhang, G.Z., Zhu, Y.H., Wang, H.N., Chen, Y.X., Wu, G.S., Wang, L.M., 2023. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In: 2023 Ieee/Cvf Conference on Computer Vision and Pattern Recognition, Cvpr, pp. 5682–5692. <http://dx.doi.org/10.1109/Cvpr52729.2023.00550>, URL: <Go to ISI>://WOS:001058542606003, Bv6j2 Times Cited:48 Cited References Count:59 IEEE Conference on Computer Vision and Pattern Recognition.