Information Fusion 130 (2026) 104084

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

ST-Imputer: Multivariate dependency-aware diffusion network with physics guidance for spatiotemporal imputation

Xingyu Zhao [a], Jianpeng Qi [a], Bin Lu [b], Lei Zhou [c], Lei Cao [d], Junyu Dong [a], Yanwei Yu [a,*]

[a] Faculty of Information Science and Engineering, Ocean University of China, Qingdao, 266100, Shandong, China
[b] Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China
[c] School of Oceanography, Shanghai Jiao Tong University, Shanghai, 200240, China
[d] Department of Computer Science, The University of Arizona, Tucson, 85721, USA

ARTICLE INFO

Keywords:
Spatiotemporal data imputation
Diffusion probabilistic model
Multi-source spatiotemporal data
Graph convolutional network

ABSTRACT

Data preparation is crucial for achieving optimal results in deep learning. Unfortunately, missing values are common when preparing large-scale spatiotemporal databases. Most existing imputation methods primarily focus on exploring the spatiotemporal correlations of single-source data; however, high missing rates in single-source data result in sparse distributions. Furthermore, existing methods typically focus on shallow correlations at a single scale, limiting the ability of imputation models to effectively leverage multi-scale spatial features. To tackle these challenges, we propose a multivariate dependency-aware spatiotemporal imputation model, named ST-Imputer. Specifically, we introduce multi-source context data to provide sufficient correlation features for target data (*i.e.*, data that needs imputation), alleviating the issue of insufficient available features caused by high missing rates in single-source data. By applying a multi-variate spatiotemporal dependency extraction module, ST-Imputer captures potential associations between different spatial scales. Subsequently, the noise prediction module utilizes the learned dual-view features to formulate the spatiotemporal transmission module, thereby reducing weight errors caused by excessive noise. Finally, physical constraints are applied to prevent unrealistic predictions. Extensive experiments on three large-scale datasets demonstrate the significant superiority of ST-Imputer, achieving up to a 13.07 % improvement in RMSE. The code of our model is available at https://github.com/Lion1a/ST-Imputer.

1. Introduction

Data preparation is a vital step in the process of large-scale data analysis and modeling [1], and one of the key challenges in this process is dealing with missing values. Missing values are commonly encountered in large-scale raw data due to environmental complexity, sensor malfunctions, or other external disturbances [2]. These missing data not only compromise the accuracy of database analysis and decision-making , but also hinder the integrity and usability of the database [3,4]. Recent research has focused on mining spatiotemporal dependencies in existing data to address the issue of missing values in large-scale spatiotemporal datasets [5], with significant results.

Building on the issue of missing values in spatiotemporal data, early research on spatiotemporal imputation focuses on statistical methods, such as mean imputation [6], linear interpolation [7], and K-nearest neighbors (KNN) [8]. With the advancement of computational power, spatiotemporal imputation methods based on deep learning have increasingly gained attention. The temporal regularized matrix factorization method (TRMF) [9] and the Bayesian augmented tensor factorization method (BATF) [10] decompose the single imputation target into several low-dimensional matrices to impute the missing values. In recent years, researchers have begun applying diffusion probability models to the task of spatiotemporal imputation, such as PriSTI [11]. Different from the diffusion probability models, methods such as OxyGenerator [12] and HSTGCN [5] model spatiotemporal data as graphs, and use graph structures to infer the features at the missing spatial nodes. In summary, existing methods leverage available data to capture temporal dependencies and spatial correlations, facilitating the imputation process of large-scale databases, and laying a solid foundation for subsequent data integration and analysis.

Nevertheless, three significant challenges remain in spatiotemporal data imputation. ***First, single-source data, with high missing rates, cannot provide sufficient spatiotemporal correlations, which limits the performance of imputation methods***. Existing spatiotemporal

* Corresponding author.
E-mail address: yuyanwei@ouc.edu.cn (Y. Yu).

https://doi.org/10.1016/j.inffus.2025.104084
Received 2 September 2025; Received in revised form 18 November 2025; Accepted 18 December 2025
Available online 20 December 2025
1566-2535/© 2025 Elsevier B.V. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

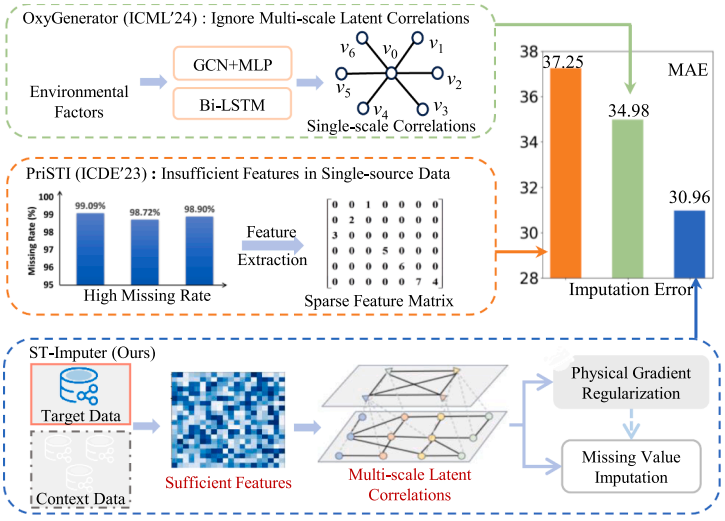

Fig. 1. The difference and performance improvement (measured by MAE) of the proposed model over SOTA models on the North Atlantic Ocean dissolved oxygen dataset.

imputation methods often rely solely on single-source data as the primary input for the model, but the high missing rate of some data severely limits the quality of the imputation results. For example, from the Global Ocean Data Analysis Project (GLODAP) [13], it is found that over 99 % of the dissolved oxygen observations in certain regions have been missing over the past 100 years. Although existing spatiotemporal imputation models leverage known information to model spatiotemporal correlations [11,14], a high missing rate implies that there is insufficient available correlation information in single-source data. Furthermore, the high sparsity of known data can also introduce more noise, leading to error accumulation effects during the modeling process.

Second, existing imputation methods primarily focus on single-scale spatial features, neglecting potential correlations between different spatial scales. Currently, most approaches employ graph convolutional networks (GCN) to aggregate correlated features or leverage a convolutional kernel for feature extraction [15]. These methods emphasize capturing spatial features from a fixed feature extraction dimension, neglecting the learning process of correlations between different spatial scales. For instance, real-world sea surface temperature (SST) exhibits strong local correlations, yet certain oceanic currents or eddies can dynamically alter these correlations across larger spatial scales. Consequently, capturing spatiotemporal dependencies across varying spatial scales is critical for the imputation task.

Third, existing data imputation models fail to incorporate the constraints imposed by physical mechanisms in real-world systems. Existing spatiotemporal imputation models primarily rely on data-driven methods [14]. But, these methods merely focus on temporal and spatial correlations without considering any physical constraints. Indeed, the real world is governed by physical laws, where the relationships among multiple variables can be represented by explicit equations, and these variables exhibit stable interdependencies under the determined equation coefficients. Ignoring such stable variable relationships may lead the model to produce unrealistic predictions.

To address these challenges, we propose a multivariate dependency-aware spatiotemporal imputation network, named **ST-Imputer**. Specifically, we incorporate multiple context data to provide sufficient spatiotemporal correlations for the imputation target, alleviating the problem of insufficient spatiotemporal features caused by a high missing rate in single-source data. Additionally, we use the multi-scale correlation learning module to capture spatial correlations across regions at different scales, addressing existing studies' limitations that ignore potential correlations across regions at varying spatial scales. Moreover, a physical gradient variance loss is introduced to incorporate physical consistency guidance directly into the model training process, enhancing the alignment with real-world physical relationships. Experiments on three large-scale datasets show the significant superiority of our proposed model over SOTA baselines. As illustrated in Fig. 1, ST-Imputer achieves +11.48 % performance improvement in MAE compared to the SOTA baseline OxyGenerator [12] on the North Atlantic Ocean dissolved oxygen dataset.

We summarize the key contributions of this work as follows:

- We propose a multivariate dependency-aware spatiotemporal imputation model, named ST-Imputer. ST-Imputer incorporates multiple context data relevant to the target data, and adopts a multi-variate correlation learning module to capture rich spatiotemporal correlations from multi-source data.
- We apply the multi-scale correlation learning module to extract spatial correlations across regions at different spatial scales. Furthermore, we utilize a physical gradient variance regularization to calibrate the uncertainty in missing value imputation, enhancing the intrinsic correlation between the imputation results and physical variables.
- We conduct extensive experiments on three real-world datasets to verify the superiority of our ST-Imputer compared with SOTA baselines, and prove the effectiveness of each sub-module in our model through ablation study.

2. Related work

2.1. Traditional imputation models

Lin-ITP [7] and KNN [8] are based on the distribution of individual missing datasets to perform imputation, relying primarily on the inherent statistical characteristics of the data. Methods like Kalman filtering (KF) [16], and vector auto-regression (VAR) [17] employ linear prediction at each time step to estimate missing values, capturing linear changes over time. *However, all of these models are limited by their reliance on basic statistical trends or linear predictions, which makes it difficult to capture nonlinear variations and complex spatiotemporal dependencies effectively.*

2.2. Recurrent neural network based imputation models

Recurrent neural network (RNN) based methods are first proposed to address the spatiotemporal imputation problem. Among them, GP-

VAE [18] combines variational autoencoder (VAE) with the Gaussian process, enabling it to handle high-dimensional time series data. BRITS [19] captures the correlations between features through a bidirectional recurrent network, dynamically updating weights based on gradient descent and filling in missing values in the hidden states. Then, LGnet [20] expands into the spatiotemporal aspects of the data, using long short-term memory (LSTM) to capture the local relationships of spatiotemporal features. *However, these RNN-based methods are inevitably affected by the problem of error accumulation, resulting in the current missing values being sequentially imputed based on inaccurate historical estimates.*

2.3. Graph neural network based imputation models

Graph neural network (GNN) based methods are also proposed to address the spatiotemporal imputation problem. HSPGNN [21] integrates spatiotemporal features into the graph neural network, employing adaptive optimization to adapt the interrelations among different features. GRIN [22], ImputeFormer [23], and OxyGenerator [12] rely on a graph-based message-passing mechanism. By dynamically adjusting parameters across different regions, it effectively captures feature correlations across temporal and spatial in sparse data. Then, extending these approaches to global network structures, DSTGCN [24] employs a global partition graph to capture regional and proximal correlations within road networks, address the spatiotemporal correlation issues in adjacent areas. *However, due to the sparse spatial distribution of single-source data and the lack of density in the graph structure, information transfer between adjacent nodes is limited when modeling GCN.*

2.4. Generative based imputation models

In recent years, generative models have been widely used for spatiotemporal imputation tasks. CSDI [25], and PriSTI [11] rely on constructing conditional features to provide an additional feature embedding matrix for the diffusion probabilistic model, simplifying the process of learning spatiotemporal dependencies.CoFILL [26] proposes temporal convolutional and graph convolutional networks to model temporal and spatial dependencies, providing conditional information for the denoising process of the diffusion model. Compared to directly using the diffusion model for imputation in CoFILL, RDPI [27] divides the imputation process into two distinct steps: it first uses GRIN to generate an initial estimate, and then employs a conditional diffusion model to optimize the residuals. This design reduces the complexity of handling all issues in a single stage, thereby mitigating the impact of overfitting and noise on the imputation results. Similarly, MLGAN [28] is based on generative adversarial networks (GAN), utilizing GCN and one-dimensional dilated convolutional neural networks to extract spatiotemporal correlations between features. MTSCI [29] and Score-CDM [15] employ the attention mechanism to balance the importance of local and global features, but essentially fail to learn the multi-scale local correlations of the spatial data. *These methods rely solely on shallow graph structures or fixed convolutional kernels to capture spatial features, which causes the generative model to overlook the potential dependencies between different spatial scales, leading to significant errors in the data generation process.*

Different from existing spatiotemporal imputation methods, our model introduces multi-variate context data to provide sufficient correlation features for target data. Then, the designed multi-variate spatiotemporal dependency extraction module provides multi-scale spatial associations for the imputation process. Finally, physical constraints of gradient variations ensure the generated data closely aligns with real-world patterns.

3. Preliminaries

In this section, we first introduce the preliminary concepts, and then formally define the problem of spatiotemporal missing values imputation.

Table 1
Key notations and descriptions.

Notations	Descriptions
$\mathbf{X}$	Multi-variate spatiotemporal data
$\tilde{\mathbf{X}}$	Masked multi-variate spatiotemporal data
N	The number of spatial points in $\tilde{\mathbf{X}}$
T	The number of time steps in $\tilde{\mathbf{X}}$
S	Diffusion steps
L	The number of layers in MVC module
L'	The number of layers in MSC module
$\mathbf{A}^{(l)}$	Adjacency matrix in l-th layer of MVC module
$\mathbf{E}^{(l)}$	Output in l-th layer of MVC module
$\mathbf{Z}^{(l')}$	Output in l'-th layer of MSC module
$\mathbf{M}_r^{(l')}$	Mapping matrix in l'-th layer of MSC module
$\mathbf{Y}$	Output of one-dimensional convolution
$\mathbf{H}$	Dual-view feature matrix
$\mathbf{H}^{loc}$	Local embedding matrix
$\mathbf{H}^{glo}$	Global embedding matrix
$\mathbf{H}^{Ini}$	Initial input matrix
$\varphi^{(i)}, \psi^{(i)}$	Attention weights
$\mathbf{H}^{spa}, \mathbf{H}^{tem}$	Output of spatial and temporal attention
$\mathbf{Q}_\tau, \mathbf{K}_\tau, \mathbf{V}_\tau$	Attention matrix
ϵ_θ	Noise prediction module

Definition 1 (Target data). *Given a target spatiotemporal data, which includes N different spatial points, each spatial point has an observation value at each time step, the target data is formally defined as $\mathbf{X}^{Target} \in \mathbb{R}^{N\times T}$, where T is the number of time steps.*

Definition 2 (Context data). *Given a target data, context data refers to other observed spatiotemporal data that have a strong spatiotemporal correlation with the target data. After alignment along the spatial and temporal dimensions, it is formally defined as $\mathbf{X}^{Cont} \in \mathbb{R}^{N\times T}$.*

Definition 3 (Multi-variate Spatiotemporal Data). *Given a target data $\mathbf{X}^{Target}$ and its multiple context data $\mathbf{X}^{Cont}$, they are concatenated together to form a multi-variate spatiotemporal data, which is formally denoted as $\mathbf{X} = \{\mathbf{X}^{Target}, \mathbf{X}^{Cont_1}, \mathbf{X}^{Cont_2}, \ldots, \mathbf{X}^{Cont_{D-1}}\} \in \mathbb{R}^{D\times N\times T}$.*

For simplicity, we use one target data and $D-1$ context data to form D-dimensional variables in $\mathbf{X}$.

Definition 4 (Observation mask). *We use a binary mask matrix $\mathbf{M} \in \{0,1\}^{N\times T}$ to represent the observation mask, where $m^{i,j} = 1$ indicates that the observed value is available, and $m^{i,j} = 0$ denotes that the observed value is missing. Applying the Hadamard product, the mask matrix is element-wise multiplied with the target data $\mathbf{X}^{Target}$ to simulate different masking strategies:*

$$\tilde{\mathbf{X}}^{Target} = \mathbf{X}^{Target} \odot \mathbf{M}. \tag{1}$$

Different masking strategies are used to simulate various missing scenarios in the real world, specific details will be introduced in Section 5.

Problem (Context-aware spatiotemporal imputation). *Given the multi-variate spatiotemporal data $\tilde{\mathbf{X}}$ which contains $\tilde{\mathbf{X}}^{Target}$ and its context data, the task of spatiotemporal imputation aims to fill in the missing values in $\tilde{\mathbf{X}}^{Target}$ by capturing the spatiotemporal correlations among the $\tilde{\mathbf{X}}^{Target}$ and its context data. This problem is formulated as a mapping:*

$$\hat{\mathbf{X}}^{Target} = f(\tilde{\mathbf{X}}), \tag{2}$$

where $f(\cdot)$ represents a deep learning model that performs missing value imputation by leveraging multi-variate spatiotemporal correlations, and $\hat{\mathbf{X}}^{Target}$ is the imputation result of the target data.

In this work, we leverage the diffusion probabilistic model to impute missing values in target data.

Diffusion probabilistic model. The diffusion probabilistic model is a generative model that has gained significant attention in the field of deep learning in recent years [30]. Compared to other generative models like GANs or VAEs, which directly produce generated results, the

diffusion probabilistic model's step-by-step noise recovery process more naturally captures the diversity of the data, making the final imputed results better aligned with the true underlying distribution, which is suitable for high missing rate imputation tasks. In particular, the diffusion probabilistic model can be formalized as two Markov chains, which refer to the diffusion process and the reverse process [31]. The diffusion process is designed to corrupt the original data, making it progressively more indistinguishable from pure noise. Conversely, the reverse process starts from the noise distribution and iteratively denoises it, reconstructing the data to resemble its original true distribution closely.

(1) Diffusion process: Given an initial masked multi-variate spatiotemporal data $\tilde{\mathbf{X}}^{(0)}$, the diffusion process gradually adds noise to the data to generate a series of intermediate variables $\tilde{\mathbf{X}}^{(1:S)}$, until obtaining $\tilde{\mathbf{X}}^{(S)}$, where S is the diffusion step, and $\mathcal{N}$ is Gaussian distribution.

(2) Reverse process: After the diffusion process, by applying the reverse process to gradually denoise from $\tilde{\mathbf{X}}^{(S)}$, a new $\tilde{\mathbf{X}}^{(0)} \sim p_{\text{org}}$ is eventually generated, where p_{org} represents the original data distribution.

More details on applying the diffusion probabilistic model for imputation will be introduced in Sections 4.2 and 4.3. The key notations are summarized in Table 1.

4. Method

The details of our proposed ST-Imputer framework are illustrated in Fig. 2. Specifically, ST-Imputer consists of four key components: (1) *Multi-variate Spatiotemporal Dependency Extraction Module*, (2) *Gaussian Noise Diffusion Module*, (3) *Dual-view Feature Based Noise Prediction Module*, and (4) *Model Learning with Physics Guidance*. *First*, the multi-variate spatiotemporal dependency extraction module captures dual-view correlated features of multiple variables from local and global perspectives. *Second*, the Gaussian noise diffusion module adds Gaussian noise to the missing values, simulating the forward diffusion process in the diffusion probabilistic model. *Third*, the dual-view feature based noise prediction module utilizes the learned dual-view correlated features to assist the reverse denoising process. *Fourth*, the model learning process leverages the variance of physical gradients to guide training, ensuring that the imputed results conform to the physical equations.

4.1. Multi-variate spatiotemporal dependency extraction

Existing studies demonstrate that providing additional auxiliary information helps improve the model's performance [11]. However, most methods extract only single-scale spatial features from single-source, high missing rates in single-source data cannot provide sufficient spatiotemporal correlations.

To address this problem, context data is introduced to supply additional spatiotemporal associations, and stack multi-variate correlation learning (MVC) and multi-scale correlation learning (MSC) modules to form local dependency perception, extracting multi-scale spatial features. Simultaneously, we use temporal and spatial attention to capture correlations from a global perspective. Finally, dual-view features are formed from both perspectives, providing spatial correlation information at different granularities for the denoising process.

4.1.1. Multi-variate correlation learning

Performing graph convolution based on multi-variate data enables the feature propagation between different variables. This allows GCN to model the mutual influence between features by aggregating similar features from both the target and context data, thereby compensating for the feature insufficiency caused by the high missing rate in a single source of data.

To do that, we use a Gaussian kernel to estimate the pairwise similarity between nodes, which is formalized as follows:

$$\mathbf{Sim}_{ij}^{(l-1)} = \exp\left(-\frac{\|d_{ij}^{(l-1)}\|^2}{\eta^2}\right), \tag{3}$$

where $d_{ij}^{(l-1)}$ represents the Euclidean distance of i-th and j-th nodes in $(l-1)$-th layer, respectively. η is the scale parameter of the Gaussian kernel. Notice that d includes the Euclidean distance between different nodes of the target data and the context data. Then, we utilize a sparsification operation to obtain the corresponding adjacency matrix $\mathbf{A}^{(l-1)}$.

$$\mathbf{A}_{ij}^{(l-1)} = \begin{cases} \mathbf{Sim}_{ij}^{(l-1)}, & \text{if } \mathbf{Sim}_{ij}^{(l-1)} \geq c \\ 0, & \text{otherwise} \end{cases}, \tag{4}$$

where c is the predefined similarity threshold parameter.

Subsequently, we apply a multi-variate enhanced GCN to learn the correlations between the target data and its context data, proactively aggregating massive similar features from multiple data sources to alleviate the issue of insufficient features caused by the sparsity of a single target data.

$$\mathbf{E}^{(l)} = \sigma\left(\mathbf{A}^{(l-1)}\mathbf{E}^{(l-1)}\mathbf{W}^{(l-1)}\right), \tag{5}$$

where $\mathbf{E}^{(l)} \in \mathbb{R}^{n \times d_l}$ is the embedding of the l-th layer, n is the number of all nodes, d_l is embedding dimension in l-th layer, σ is ReLU activation function, and $\mathbf{W}^{(l-1)} \in \mathbb{R}^{d_{l-1} \times d_l}$ is the learnable weight matrix. Initially, the masked multivariate spatiotemporal data $\tilde{\mathbf{X}}$ first undergoes a linear interpolation operation, subsequently being used as $\mathbf{E}^{(0)}$ for input.

As shown in MVC module in Fig. 3, C_i^j is i-th grid node in j-th context data, T_i is i-th grid node in target data. Given initial input, feature aggregation between different nodes from target data and its context data can be achieved through multi-variate enhanced GCN, establishing feature correlations between C_i^j and T_i.

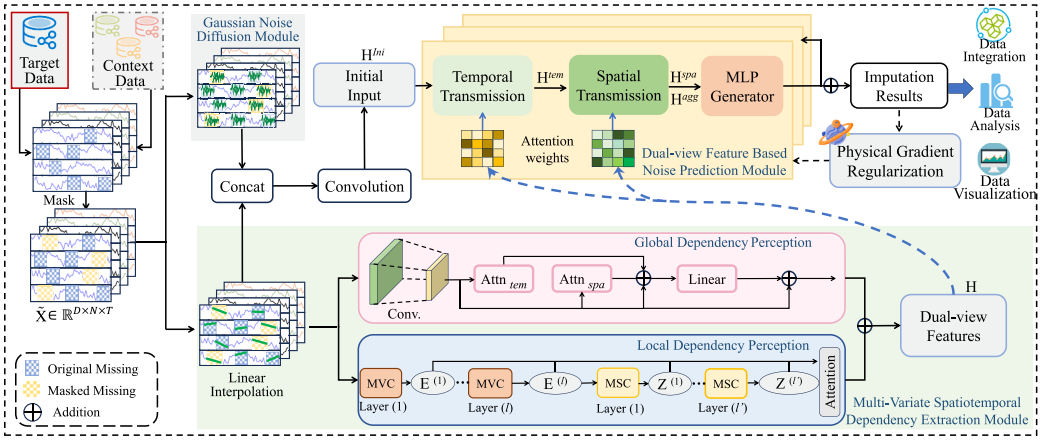

Fig. 2. Overall framework of the proposed ST-Imputer.

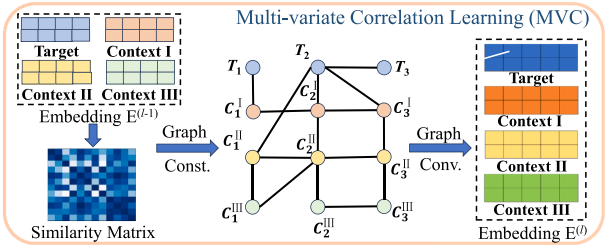

Fig. 3. The architecture of the multi-variate correlation learning module. *Abbr.*: Const.: Construction, Conv.: Convolution.

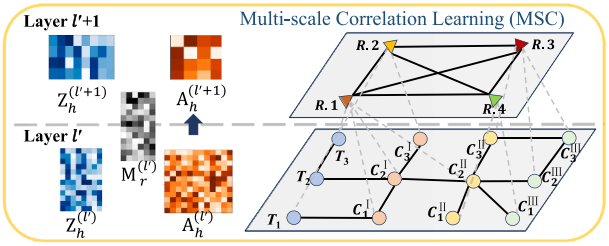

Fig. 4. The architecture of the multi-scale correlation learning module. *Abbr.*: R.: Region.

4.1.2. Multi-scale correlation learning

In real-world scenarios, there exist various spatial dependency associations between different regions. For example, in some coastal areas, ocean currents or wind speed significantly impact local spatiotemporal dependencies. These effects can extend to broader spatial scales, influencing climate patterns and ecological conditions in larger regions. Therefore, after obtaining the node-level embeddings among multivariate spatiotemporal data, it is necessary to consider correlations between different regional ranges, *i.e.*, capturing multi-scale spatial correlations.

To establish spatial correlations across different scales, we aggregate different regions in a hierarchical manner. Specifically, we first generate a region mapping matrix $\mathbf{M}_r$ based on the region embeddings. This matrix is used to transform the embeddings from the previous layer, thereby producing larger-scale embeddings in the next layer. These new embeddings effectively combine multi-scale associations, representing broader correlations across regions and providing a more comprehensive view of spatial dependencies. The specific process is as follows:

$$\mathbf{M}_r^{(l')} = \sigma\left(\mathbf{L}_h^{(l')}\mathbf{Z}_h^{(l')}\Theta_m + b_m\right), \tag{6}$$

where $\mathbf{L}_h^{(l')} \in \mathbb{R}^{n_{l'} \times n_{l'}}$ is normalized laplace matrix of the adjacency matrix $\mathbf{A}_h^{(l')}$ in l'-th layer of MSC module, $n_{l'}$ is the number of nodes in l'-th layer, $\mathbf{Z}_h^{(l')}$ is region embeddings in l'-th layer, Θ_m is a weight matrix, and b_m is bias term. $\mathbf{M}_r^{(l')}$ represents the region mapping matrix in l'-th layer. Assuming that $n_{l'+1}$ nodes are obtained after the mapping in $(l'+1)$-th layer, $\mathbf{M}_r^{(l'+1)} \in \mathbb{R}^{n_{l'} \times n_{l'+1}}$. Initially, the output $\mathbf{E}^{(L)}$ in MVC is used as $\mathbf{Z}_h^{(1)}$, and $\mathbf{A}^{(L)}$ is used as $\mathbf{A}_h^{(1)}$.

Then, the region embeddings from the previous layer are further propagated to the next layer through the region mapping matrix, generating new and broader region-level embeddings and adjacency matrix:

$$\begin{aligned} \mathbf{Z}_h^{(l'+1)} &= \mathbf{M}_r^{(l')\mathrm{T}}\mathbf{Z}_h^{(l')}, \\ \mathbf{A}_h^{(l'+1)} &= \mathbf{M}_r^{(l')\mathrm{T}}\mathbf{A}_h^{(l')}\mathbf{M}_r^{(l')}, \end{aligned} \tag{7}$$

where $\mathbf{Z}_h^{(l')} \in \mathbb{R}^{n_{l'} \times d_{l'}}$, $n_{l'}$ and $d_{l'}$ denote the number of nodes and the embedding dimension in l'-th layer, respectively. After the mapping, we generate $\mathbf{Z}_h^{(l'+1)} \in \mathbb{R}^{n_{l'+1} \times d_{l'}}$ and $\mathbf{A}_h^{(l'+1)} \in \mathbb{R}^{n_{l'+1} \times n_{l'+1}}$ in $(l'+1)$-th layer. Initially, the embeddings represent a smaller node-level scale. Then, these regions are propagated through the region mapping matrix $\mathbf{M}_r^{(l')}$, they are mapped into broader region embeddings in the $(l'+1)$-th layer. As shown in MSC module in Fig. 4, the $(l'+1)$-th layer new embeddings are influenced by the previous layer l', as the number of nodes decreases, the embeddings in $(l'+1)$-th layer can aggregate embeddings from multiple regions in a broader perspective, representing features at a larger spatial scale.

By multiplying the multi-scale spatial embeddings aggregated by GCN with $\mathbf{M}_r^{(l')}$, the dimensions of the aggregated matrix can be kept consistent with the previous layer. Additionally, $\mathbf{M}_r^{(l')}$ helps selectively aggregate information from the previous layer, retaining key structural information while reducing dimensions, thereby efficiently propagating embeddings throughout the network. The multi-scale graph convolution operation is formalized as:

$$\mathbf{Z}^{(l'+1)} = \mathbf{M}_r^{(l')}\left(\sigma\left(\mathbf{L}_h^{(l'+1)}\mathbf{Z}_h^{(l'+1)}\Theta_h + b_h\right)\right), \tag{8}$$

where $\mathbf{Z}^{(l'+1)}$ denotes the output of the multi-scale graph convolution at $(l'+1)$-th layer.

In MSC module, the multi-variate correlations are further refined through multi-scale spatial correlation mapping. *From the perspective of variable correlations*, MSC extracts broader-scale information from different regions. Through multi-layer information propagation, it refines the expression of highly correlated embeddings between regions at different scales. *From the perspective of perception range*, MSC progressively aggregates the region embeddings, as the number of layers increases, a broader range of spatial correlation information can be dynamically integrated, without the need to design larger convolution kernels and fixed local weighting processes to learn multi-scale spatial correlations.

4.1.3. Local dependency perception

After multi-layer MVC and MSC, we obtain both multi-variate and multi-scale correlations. To effectively integrate these correlations and establish local dependencies, we utilize an attention mechanism to fuse these associations:

$$\mathbf{H}^{loc} = \sum_{i=1}^{L} \varphi^{(i)}\mathbf{E}^{(i)} + \sum_{i=1}^{L'} \psi^{(i)}\mathbf{Z}^{(i)}. \tag{9}$$

where $\varphi^{(i)}$ and $\psi^{(i)}$ represent the attention weights in different layers of MVC and MSC modules, respectively, and L and L' denote the number

of layers in MVC and MSC modules. Through Eq. (10), the attention weights for each layer's embeddings can be calculated.

$$\varphi^{(i)} = \frac{\exp(\mathbf{E}^{(i)}\mathbf{W} + b)}{\sum_{j=1}^{L}\exp(\mathbf{E}^{(j)}\mathbf{W} + b) + \sum_{j=1}^{L'}\exp(\mathbf{Z}^{(j)}\mathbf{W} + b)},$$
$$\psi^{(i)} = \frac{\exp(\mathbf{Z}^{(i)}\mathbf{W} + b)}{\sum_{j=1}^{L}\exp(\mathbf{E}^{(j)}\mathbf{W} + b) + \sum_{j=1}^{L'}\exp(\mathbf{Z}^{(j)}\mathbf{W} + b)}, \tag{10}$$

where $\mathbf{W}$ is a shared weight matrix, and b is a bias term that provides a learnable offset to prevent extreme attention weights caused by variations in feature.

By assigning an attention score to each layer, the model captures the complex correlations among multi-variate embeddings and multi-scale embeddings. This design enables the model to integrate node-level embeddings and region-level embeddings, and weigh the importance of different layers, effectively providing rich multi-scale local correlation dependencies for the target data.

4.1.4. Global dependency perception

To further capture the correlation between target data and its context data across the entire spatial domain, we apply one-dimensional convolution to the multi-variate spatiotemporal data after linear interpolation, using a sliding window to capture the spatiotemporal correlations across all time steps.

$$\mathbf{Y} = \mathrm{Conv}(\mathrm{Interp}(\tilde{\mathbf{X}})), \tag{11}$$

where $\mathrm{Conv}(\cdot)$ denotes the convolution operation, $\mathrm{Interp}(\cdot)$ represents linear interpolation, and $\mathbf{Y}$ is global spatiotemporal features extracted by the convolution. Applying linear interpolation to $\tilde{\mathbf{X}}$, initially constructs a rough yet effective spatiotemporal representation. Intuitively, this interpolation preserves a certain level of spatiotemporal consistency without introducing randomness into the time series.

Then, temporal and spatial attention are used separately to enhance the focus on important features in the global temporal and spatial dimensions, improving the model's ability to capture the correlation between target data and its context data from a global perspective.

$$\mathbf{H}^{ta} = \mathrm{Norm}(\mathrm{Att}_{tem}(\mathbf{Y}) + \mathbf{Y}),$$
$$\mathbf{H}^{sa} = \mathrm{Norm}(\mathrm{Att}_{spa}(\mathbf{Y}) + \mathbf{Y}), \tag{12}$$

where $\mathrm{Norm}(\cdot)$ denote a layer normalization operation, $\mathrm{Att}_{spa}(\cdot)$ and $\mathrm{Att}_{tem}(\cdot)$ represent spatial attention and temporal attention respectively, which can be expressed as:

$$\mathrm{Att}_{tem}(\mathbf{Y}) = \mathrm{softmax}\left(\frac{\mathbf{Y}_{tem}\mathbf{W}_Q(\mathbf{Y}_{tem}\mathbf{W}_K)^{\mathrm{T}}}{\sqrt{d}}\right)\mathbf{Y}_{tem}\mathbf{W}_V,$$
$$\mathrm{Att}_{spa}(\mathbf{Y}) = \mathrm{softmax}\left(\frac{\mathbf{Y}_{spa}\mathbf{W}'_Q(\mathbf{Y}_{spa}\mathbf{W}'_K)^{\mathrm{T}}}{\sqrt{d}}\right)\mathbf{Y}_{spa}\mathbf{W}'_V, \tag{13}$$

where $\mathbf{Y}_{tem}$ and $\mathbf{Y}_{spa}$ are the results of flattening $\mathbf{Y}$ along the temporal and spatial dimensions, $\mathbf{W}$ and $\mathbf{W}'$ are weight matrices, d is the dimension of the embedding. Q, K, and V represent query, key, and value in the attention mechanism.

Through the linear layer, the temporal and spatial attention features are weighted and re-integrated, enabling the model to learn joint representations of the temporal and spatial features:

$$\mathbf{H}^{glo} = \mathrm{Linear}(\mathbf{H}^{ta} + \mathbf{H}^{sa} + \mathbf{Y}), \tag{14}$$

where $\mathbf{H}^{glo}$ represents the captured multi-variate global spatiotemporal correlation features.

4.2. Gaussian noise diffusion module

As mentioned in Section 3, the diffusion probabilistic model for imputation mainly consists of the diffusion and reverse processes, which are used for generating noise and denoising to recover the original data, respectively.

For convenience, in this paper, we use $\tilde{\mathbf{X}}^{\mathrm{diff}} = \mathrm{diff}(\tilde{\mathbf{X}})$ to represent the diffusion process of adding Gaussian noise, where $\tilde{\mathbf{X}}^{\mathrm{diff}}$ denotes the multi-variate data with Gaussian noise, and $\mathrm{diff}(\cdot)$ is the diffusion operation. Specifically, the diffusion process can be recursively formalized as:

$$q\left(\tilde{\mathbf{X}}^{(1:S)} \mid \tilde{\mathbf{X}}^{(0)}\right) = \prod_{t=1}^{S} q\left(\tilde{\mathbf{X}}^{(t)} \mid \tilde{\mathbf{X}}^{(t-1)}\right),$$
$$q\left(\tilde{\mathbf{X}}^{(t)} | \tilde{\mathbf{X}}^{(t-1)}\right) = \mathcal{N}\left(\tilde{\mathbf{X}}^{(t)}; \sqrt{1-\beta^{(t)}}\tilde{\mathbf{X}}^{(t-1)}, \beta^{(t)}\mathbf{I}\right), \tag{15}$$

where $q(\tilde{\mathbf{X}}^{(1:S)} \mid \tilde{\mathbf{X}}^{(0)})$ is the joint probability distribution from the initial state $\tilde{\mathbf{X}}^{(0)}$ to the S-th step, $\tilde{\mathbf{X}}^{(1:S)}$ represents all states from step 1 to step S. Using the chain rule, the joint probability from the initial state to step S is decomposed into the product of conditional probabilities at each step. $\mathcal{N}\left(\tilde{\mathbf{X}}^{(t)}; \sqrt{1-\beta^{(t)}}\tilde{\mathbf{X}}^{(t-1)}, \beta^{(t)}\mathbf{I}\right)$ denotes a Gaussian distribution with a variance of $\beta^{(t)}\mathbf{I}$. The scaling factor $\sqrt{1-\beta^{(t)}}$ controls the amount of noise added. $\mathbf{I}$ is the identity matrix, indicating that the independent and identically distributed noise is added across all feature dimensions.

By recursively adding noise through Eq. (15), the final state $\tilde{\mathbf{X}}^{\mathrm{diff}}$ approximates a Gaussian distribution. The ultimate result of this process is to transform the original data into Gaussian noise, enabling the reverse diffusion process to restore the data.

4.3. Dual-view feature based noise prediction module

After the diffusion process, the reverse process gradually removes noise. The reverse process is formulated as follows:

$$p_\theta\left(\tilde{\mathbf{X}}^{(0:S-1)} \mid \tilde{\mathbf{X}}^{(S)}\right) = \prod_{t=1}^{S} p_\theta\left(\tilde{\mathbf{X}}^{(t-1)} \mid \tilde{\mathbf{X}}^{(t)}\right),$$
$$p_\theta\left(\tilde{\mathbf{X}}^{(t-1)} \mid \tilde{\mathbf{X}}^{(t)}\right) = \mathcal{N}\left(\tilde{\mathbf{X}}^{(t-1)}; \mu_\theta(\tilde{\mathbf{X}}^{(t)}, t), \sigma_t^2\mathbf{I}\right), \tag{16}$$

where $p_\theta(\tilde{\mathbf{X}}^{(0:S-1)} \mid \tilde{\mathbf{X}}^{(S)})$ is the conditional probability of the sequence $\tilde{\mathbf{X}}^{(0:S-1)}$, given the final state $\tilde{\mathbf{X}}^{(S)}$. Based on previous work [32], $\mu_\theta(\tilde{\mathbf{X}}^{(t)}, t)$ and σ_t^2 can be formulated as:

$$\mu_\theta(\tilde{\mathbf{X}}^{(t)}, t) = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\tilde{\mathbf{X}}^{(t)} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\tilde{\mathbf{X}}^{(t)}, t)\right),$$
$$\sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t, \tag{17}$$

where $\mu_\theta(\cdot,\cdot)$ is the mean function, σ_t^2 denotes the variance on diffusion step, $\bar{\alpha}_t$ and $\bar{\alpha}_{t-1}$ are the scaling factors, representing the degree of preservation of the original data, and ϵ_θ represents a noise prediction module parameterized by θ. In this work, we use $\mathrm{reverse}(\cdot)$ to briefly represent the reverse denoising process.

To further leverage the fused features from dual-view to provide enriched context information for noise prediction, we combine $\mathbf{H}^{loc}$ and $\mathbf{H}^{glo}$ from Eqs. (9) and (14) into a new matrix $\mathbf{H}$, representing the dual-view features that integrate both local and global spatiotemporal dependencies.

$$\mathbf{H} = \mathbf{H}^{glo} + \mathbf{H}^{loc}. \tag{18}$$

Subsequently, $\mathbf{H}$ is used to formulate temporal attention and spatial attention for spatiotemporal transmission in the noise prediction module. Taking temporal attention as an example:

$$\mathbf{H}^{tem} = \mathbf{A}_\tau \cdot \mathbf{V}_\tau, \tag{19}$$

$$\mathbf{A}_\tau = \mathrm{softmax}\left(\frac{\mathbf{Q}_\tau\mathbf{K}_\tau^{\mathrm{T}}}{\sqrt{d_\tau}}\right), \tag{20}$$

where $\mathbf{H}^{tem}$ is the output of the temporal attention, $\mathbf{A}_\tau$ is the attention weight, $\mathbf{Q}_\tau, \mathbf{K}_\tau, \mathbf{V}_\tau$ can be further represented as:

$$\mathbf{Q}_\tau = \mathbf{H}\cdot\mathbf{W}_\tau^Q, \mathbf{K}_\tau = \mathbf{H}\cdot\mathbf{W}_\tau^K, \mathbf{V}_\tau = \mathbf{H}^{Ini}\cdot\mathbf{W}_\tau^V. \tag{21}$$

$\mathbf{W}_{\tau}^{Q}, \mathbf{W}_{\tau}^{K}, \mathbf{W}_{\tau}^{V}$ are learnable parameters, $\mathbf{H}^{Ini}$ represents the initial input, further formalized as:

$$\mathbf{H}^{Ini} = \text{Conv}(\text{Concat}(\tilde{\mathbf{X}}^{\text{diff}}, \text{Interp}(\tilde{\mathbf{X}}))). \tag{22}$$

Then, the spatial attention is formulated through Eq. (23), and the output of spatial attention is $\mathbf{H}^{spa}$.

$$\mathbf{H}^{spa} = \mathbf{A}_s \cdot \mathbf{V}_s, \tag{23}$$

where $\mathbf{A}_s$ is the spatial attention weight, and its calculation is similar to Eq. (20). $\mathbf{V}_s$ is the spatial attention value, which can be further expressed as:

$$\mathbf{V}_s = \mathbf{H}^{tem} \cdot \mathbf{W}_s^{V}. \tag{24}$$

The temporal and spatial attention mechanisms calculate the weights using dual-view features $\mathbf{H}$, it avoids the weight errors learned from $\mathbf{H}^{Ini}$. Then, $\mathbf{H}^{tem}$ is fed into a GCN layer for feature aggregation, which further enhances the representation of spatial dependencies among neighboring nodes.

$$\mathbf{H}^{agg} = \text{GCN}(\mathbf{H}^{tem}, \mathbf{A}_{dis}), \tag{25}$$

where $\mathbf{A}_{dis}$ represents the adjacency matrix constructed based on geographical distances. Finally, the output $\mathbf{H}^{spa}$ and $\mathbf{H}^{agg}$ from spatial transmission module are passed through an MLP:

$$\mathbf{F}^{(1)} = \text{GAU}(\text{MLP}(\mathbf{H}^{spa} + \mathbf{H}^{agg})), \tag{26}$$

where $\text{GAU}(\cdot)$ denotes gated activation unit, and $\mathbf{F}^{(1)}$ is the output from the first-layer of the noise prediction module. By stacking multiple noise prediction modules, we can obtain the noise prediction result of each layer:

$$\mathbf{F}^{(l+1)} = \text{Predict}(\mathbf{F}^{(l)}), \quad l = 0, \ldots, L_{pre} - 1, \tag{27}$$

where $\text{Predict}(\cdot)$ denotes the noise prediction module, L_{pre} is the number of layers in noise prediction module. Subsequently, we sum the outputs from each noise prediction module as ϵ_θ, together with $\tilde{\mathbf{X}}^{\text{diff}}$ to feed into the reverse denoising process for gradual noise removal:

$$\hat{\mathbf{X}}^{\text{Target}} = \text{reverse}(\sum_{l=1}^{L_{pre}} \mathbf{F}^{(l)}, \tilde{\mathbf{X}}^{\text{diff}}), \tag{28}$$

where $\text{reverse}(\cdot)$ represents the reverse denoising process described in Eq. (16), $\tilde{\mathbf{X}}^{\text{diff}}$ is the output of the noise diffusion process, and $\hat{\mathbf{X}}^{\text{Target}}$ is the final imputation result.

4.4. Model learning with physics guidance

During the reverse denoising process, we impose a denoising loss $\mathcal{L}_\theta$ to constrain the model's error in recovering data:

$$\mathcal{L}_\theta = \mathbb{E}_{\tilde{\mathbf{X}}^{(0)} \sim q(\tilde{\mathbf{X}}^{(0)}), \epsilon \sim \mathcal{N}(0,I)} \left\| \epsilon - \epsilon_\theta(\tilde{\mathbf{X}}^{(t)}, t) \right\|^2, \tag{29}$$

where $\mathbb{E}_{\tilde{\mathbf{X}}^{(0)} \sim q(\tilde{\mathbf{X}}^{(0)}), \epsilon \sim \mathcal{N}(0,I)}$ is the expectation for variables, $\tilde{\mathbf{X}}^{(0)} \sim q(\tilde{\mathbf{X}}^{(0)})$ represents the sample $\tilde{\mathbf{X}}^{(0)}$ extracted from the real data distribution q, and $\epsilon \sim \mathcal{N}(0, I)$ represents the noise extracted from the Gaussian distribution. ϵ is the noise added to the data, and $\epsilon_\theta(\cdot, \cdot)$ is a noise prediction module parameterized by θ. We apply $L2$ to minimize the difference between the predicted noise and the actual noise, allowing the model to effectively denoise.

Since $\mathbf{E}^{(l)}$ in Eq. (5) is learnable and dynamically updated during the training process, to ensure that each embedding represents a different correlation pattern, and maximize the differences between embeddings, we impose a Hilbert-Schmidt Independence Criterion (HSIC) to calculate the dependency relationship between the different embeddings in $\mathbf{E}^{(l)}$:

$$\mathcal{L}_{\text{HSIC}} = \sum_{i=1}^{n} \sum_{j=1}^{n} HSIC\left(\mathbf{E}_i^{(l)}, \mathbf{E}_j^{(l)}\right), \tag{30}$$

where $\mathbf{E}_i^{(l)}$ and $\mathbf{E}_j^{(l)}$ is the i-th and j-th embedding in $\mathbf{E}^{(l)}$, and n is the number of all nodes.

Additionally, in physical systems, the relationships among multiple variables can be expressed by explicit equations, and these variables exhibit stable interdependencies under the determined equation coefficients [33,34]. If their variance of gradient ratios is excessively large, it indicates abnormal fluctuations occurring in inter-variable relationships, which are inconsistent with the stable physical relationships in the equations. To ensure that the model's imputation results maintain stable correlations with the context data, the gradient variance should be minimized as much as possible.

$$\mathcal{L}_{\text{phy}} = \left\| \text{Var}\left(\left\{ \frac{\partial \hat{\mathbf{X}}^{\text{Target}}}{\partial \mathbf{X}^{\text{Cont}}} \right\} \right) \right\|_2^2, \tag{31}$$

where $\text{Var}(\cdot)$ denotes a function that computes variance, $\mathbf{X}^{\text{Cont}}$ refers to context data, and $\hat{\mathbf{X}}^{\text{Target}}$ is the imputation results. In the experiments, we apply gradient variance regularization to constrain the variable relationships encountered in different explicit equations.

Finally, we integrate the losses in Eqs. (29)–(31) as the total loss:

$$\mathcal{L} = \mathcal{L}_\theta + \lambda_1 \mathcal{L}_{\text{HSIC}} + \lambda_2 \mathcal{L}_{\text{phy}}, \tag{32}$$

where λ_1 and λ_2 are the weight coefficients.

5. Experiments

In this section, we introduce the datasets, baselines, experimental settings, and evaluation metrics. Then, we evaluate our model with extensive experiments to answer the following research questions:

- **RQ1:** Can ST-Imputer provides superior imputation performance in various missing strategies compared to several state-of-the-art baselines?
- **RQ2:** How does each module of ST-Imputer contribute to its overall performance?
- **RQ3:** What is the impact of the major hyperparameters of ST-Imputer on the imputation performance?
- **RQ4:** What are the differences in the error distribution of imputation results between ST-Imputer and state-of-the-art baselines?
- **RQ5:** How does ST-Imputer perform in terms of efficiency?

5.1. Datasets

We use the air quality index (AQI) data from 36 monitoring stations covering the entire Beijing [35] and dissolved oxygen (DO) data from the Global Ocean Data Analysis Project version 2 (GLODAPv2)[1] [13] as the imputation targets, respectively. Based on previous research [33, 34], the association between AQI, DO, and other variables is represented as Eq. (33). According to these relationships, we select temperature (T), salinity (S), wind speed (WS), and nitrate (Nit) from the European Centre for Medium-Range Weather Forecasts (ECMWF)[2], Copernicus Marine Environment Monitoring Service (CMEMS)[3], and GLODAPv2 as context data to provide spatiotemporal correlation for imputation target.

$$\begin{aligned} \underline{\text{DO}} &= \text{DO}_{\text{sat}} \cdot (1 - \gamma_0(\underline{\text{T}} - \text{T}_{\text{ref}})) - \gamma_1 \cdot \underline{\text{S}} + \gamma_2 \cdot \underline{\text{Nit}}, \\ \underline{\text{AQI}} &= \rho_0 + \rho_1 \cdot \underline{\text{T}} + \rho_2 \cdot \underline{\text{WS}}, \end{aligned} \tag{33}$$

where DO_{sat} is the saturated concentration of dissolved oxygen, T_{ref} is reference temperature under standard conditions, γ and ρ are weight coefficient based on empirical knowledge.

Table 2 shows detailed information on target data and context data. Particularly, for dissolved oxygen, we select the Northeastern Sea of Japan (NSJ) (30°-35°N, 140°-145°E), and the North Atlantic Ocean

[1] https://glodap.info/
[2] https://cds.climate.copernicus.eu/
[3] https://marine.copernicus.eu/

Table 2
Details of target and context data.

Target Data	Missing Rate	Context Data	Dataset	Temporal Range	Number of Records
Dissolved Oxygen	96 %-99 %	Nitrate Temperature Salinity	GLODAPv2	2000 - 2021	8.63×10^4
		Temperature Salinity	CMEMS	1993 - 2023	1.05×10^8
AQI-36	13.25 %	Wind Speed Temperature	ECMWF	May 2014 - April 2015	3.15×10^5

Table 3
Performance comparison of the proposed model and different baselines. Marker * indicates the results are statistically significant (*t*-test with p-value < 0.01).

Method	Dissolved Oxygen						AQI-36					
	Northeast Sea of Japan			North Atlantic Ocean			Point-Missing			Block-Missing		
	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)
KNN	124.271	123.658	81.097	133.809	129.065	54.439	30.974	23.549	23.498	40.598	35.444	32.993
Lin-ITP	62.988	55.417	49.500	69.782	60.704	44.227	20.498	12.309	15.862	22.389	12.699	16.989
MICE	100.505	94.382	41.010	106.524	98.544	37.829	29.194	23.207	23.993	32.728	25.461	24.560
VAR	91.256	86.799	40.210	89.577	79.652	35.009	20.501	12.980	17.021	21.429	13.510	17.422
TRMF	85.600	79.901	39.790	89.542	80.010	34.566	19.427	12.696	17.996	20.877	12.942	17.323
BATF	84.519	79.842	39.667	79.596	76.330	30.522	19.250	12.337	17.382	20.391	12.507	16.669
CSDI	46.896	38.672	25.151	47.059	38.466	29.538	13.456	7.201	11.898	16.648	7.992	13.569
GRIN	50.381	42.377	26.905	49.625	40.203	29.036	18.498	9.370	14.085	19.980	10.329	15.997
STD-GAE	51.256	44.709	27.575	50.726	41.503	30.120	17.448	8.770	14.965	20.598	14.051	16.005
DAMR	48.026	39.551	25.405	48.116	38.597	29.307	16.541	8.120	13.412	19.670	9.478	15.775
ImputeFormer	46.192	38.495	24.014	46.579	37.855	28.631	18.327	8.066	13.521	19.011	9.209	15.279
PoGeVon	46.532	38.704	24.451	47.220	37.997	29.012	16.011	7.381	11.779	16.530	7.875	12.990
ReCTSi	45.401	38.155	24.316	46.570	37.447	28.552	12.152	7.164	11.369	15.530	7.801	12.086
PriSTI	47.592	39.063	25.770	45.657	37.255	28.332	11.672	6.570	11.057	15.609	7.821	11.411
CIB	44.337	37.590	24.043	42.581	35.647	25.339	11.597	6.532	10.987	15.496	7.743	11.409
OxyGenerator	42.238	35.017	21.371	40.371	34.981	23.220	12.778	7.046	11.499	16.935	8.805	13.121
ST-Imputer	**36.710***	**31.376***	**20.040***	**35.472***	**30.964***	**21.246***	**11.128***	**6.088***	**10.439***	**14.525***	**7.388***	**11.120***
Improvement	13.07 %	10.39 %	6.22 %	12.13 %	11.48 %	8.52 %	4.04 %	6.79 %	4.98 %	6.26 %	4.58 %	2.53 %

(NAO) (50°-55°N, 15°-20°W), which have ocean events related to dissolved oxygen, such as fishing grounds and ocean currents, as the research areas. Based on the previous research strategy [36], we further resample these regions into grids with a spatial resolution of 0.25°×0.25°, and each grid is extended to 12 underwater depth intervals, reaching up to 1500m.

5.2. Baselines

To evaluate the superiority of our model, we compare the results with 16 imputation models, covering classical statistical models, machine learning models, and SOTA deep learning models.

Classical statistical models: Lin-ITP [7], KNN [8]. *Machine learning models*: MICE [37], VAR [17], TRMF [9], BATF [10]. *Deep learning models*: CSDI [25], GRIN [22], STD-GAE [38], DAMR [39], PoGeVon [40], ReCTSi [41], ImputeFormer [23], PriSTI [11], CIB [14], OxyGenerator [12].

5.3. Experimental settings

5.3.1. Implementation details

We implement our ST-Imputer with PyTorch 1.13.1, and run experiments on a machine with Ubuntu 20.04 and an NVIDIA GeForce 4090 GPU. For dissolved oxygen, we select the data from each year separately as the test set, and the remaining data as the training and validation sets to iteratively train the model. For AQI-36, we divide the training/validation/test sets following the settings of previous work [22],[35]: select March, June, September, and December as the test set, the last 10 % of the data from February, May, August, and November as the validation set, and the remaining data as the training set.

Table 4
Comparison of hybrid-missing results on AQI-36.

Method	RMSE	MAE	MAPE (%)
ImputeFormer	20.679	11.591	15.174
ReCTSi	19.339	9.784	14.101
PriSTI	18.711	9.555	12.859
OxyGenerator	19.289	9.577	13.254
CIB	18.650	9.503	12.570
ST-Imputer	**17.630***	**9.102***	**11.648***
Improvement	5.46 %	4.21 %	7.33 %

5.3.2. Hyper-parameter setting

We set the batch size to 16, the learning rate to 0.001, and the number of attention heads to 6 or 8 on different datasets. The minimum noise level β_1 [25] is set to 0.0001, and the maximum noise level β_T [25] is set to 0.2.

5.3.3. Imputation evaluation

For dissolved oxygen, the Northeast Sea of Japan and the North Atlantic Ocean have data missing rates of 97.97 % and 97.87 % in the original data, respectively. We randomly mask up to 99 % of the observed values for model training and validation. Since the missing rate of observed values in the original data is relatively high, we evaluate all available values in the test set to avoid the contingency of the evaluation metrics. For AQI-36, we follow the imputation criteria from previous research [11], based on the original missing rate of 13.25 %, simulating point missing, block missing, and hybrid-missing rates to 25 %, 24.64 %, and 24.58 %, respectively. The different missing strategies applied to the AQI-36 dataset are as follows:

Point missing: Based on the original missing rate of the observed data, we randomly mask the observed values to 25 %.

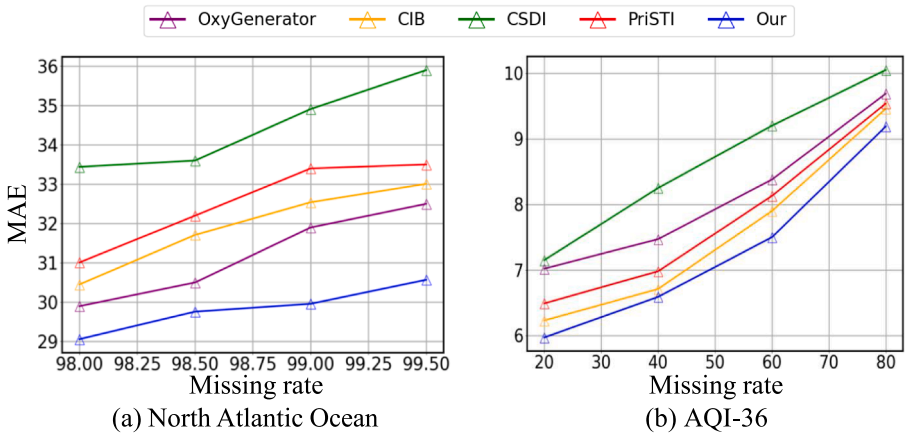

Fig. 5. The imputation results of different missing rates.

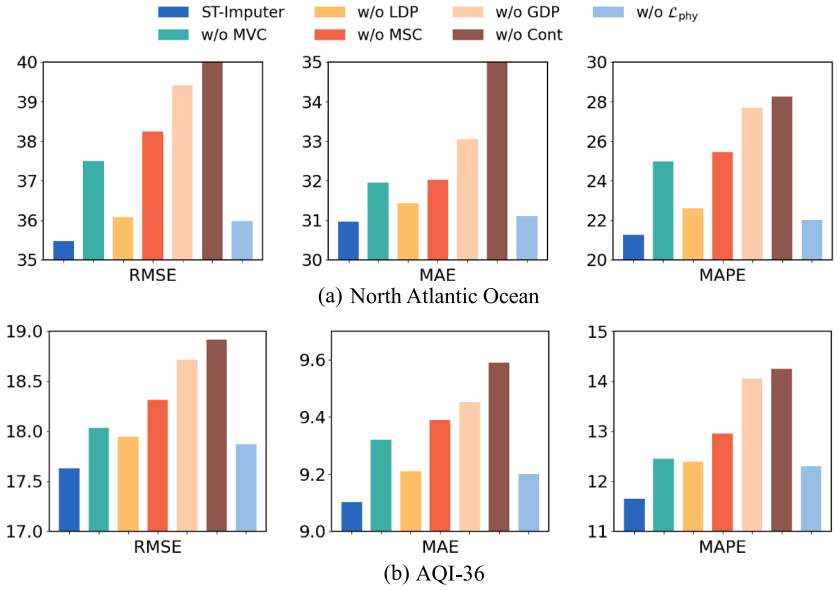

Fig. 6. Ablation study by removing different components.

Block missing: Based on masking 5 % of the observational data, we randomly mask 1 to 12 hours for each sensor with a probability of 0.15 %.

Hybrid missing: We identify all missing data for each month in the dataset, then remove the corresponding data from the same date in the next month [35].

Note that, except for Section 5.4, all other experimental results on AQI-36 dataset are based on the implementation of the hybrid missing strategy.

5.3.4. Evaluation metrics

To evaluate the accuracy of the imputation results, we use root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) to measure the results. Additionally, we use the coefficient of determination (R^2) to evaluate the difference between the imputation results and the observed data.

5.4. Performance comparison (RQ1)

Tables 3 and 4 show a comprehensive performance comparison among different imputation methods across three datasets. Our ST-Imputer achieves the best performance, the RMSE on dissolved oxygen and AQI-36 datasets improved by 13.07 %, 12.13 %, 4.04 %, 6.26 %, and 5.46 %, respectively. Proving its superiority in spatiotemporal imputation task.

First, previous studies such as CIB, PriSTI, CSDI, and ImputeFormer focus solely on modeling a single target data, but the high missing rates in the original spatiotemporal data have limited these models to extract sufficient spatiotemporal features from single-source data. On dissolved oxygen dataset with a high missing rate, ST-Imputer outperformed theirs, achieving an improvement of up to 17.20 % in RMSE compared to CIB. On AQI-36 dataset with a small missing rate of block missing strategy, ST-Imputer also achieved a performance improvement of up to 14.52 %. This demonstrates that our model alleviates the feature sparsity problem in single-source data by introducing multi-source contextual information, and uses the global dependency perception module and the local dependency perception module to capture spatiotemporal correlations between multi-variate context data and the target data.

Second, methods like DAMR, GRIN, and OxyGenerator model spatiotemporal correlations using graph structures, but they only consider spatiotemporal expressions within adjacent regions, overlooking potential feature correlations across multiple regions. Compared to

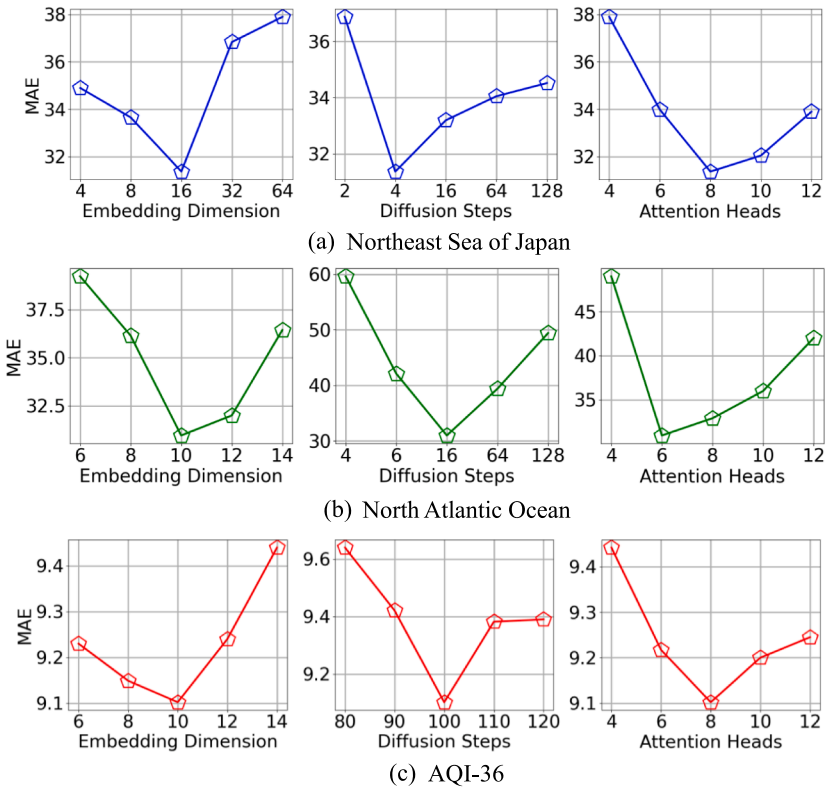

Fig. 7. The impact of embedding dimension d, diffusion steps S, and attention heads on ST-Imputer.

Table 5
The prediction results after imputation.

Dataset	Metric	Ori.	CIB	OxyGenerator	Our
AQI-36	RMSE	63.85	45.54	47.98	42.35
	MAE	44.87	30.28	36.26	28.21
NAO	RMSE	-	70.26	67.36	65.14
	MAE	-	64.70	62.51	60.23

OxyGenerator, ST-Imputer achieved a RMSE performance improvement of up to 13.07 % on the dissolved oxygen dataset and 14.23 % on the AQI-36 dataset. This demonstrates that ST-Imputer effectively leverages the MVC and MSC modules to learn multi-variate and multi-scale correlation. By utilizing multi-scale latent features, further enhancing the model's capacity to capture potential correlations at multiple scales.

We further conduct experiments with other different missing rates, and the imputation results are shown in Fig. 5. The performance of all methods declines as the missing rate increases, but ST-Imputer consistently maintains the best performance. Notably, on the North Atlantic Ocean dataset with a missing rate of 99.5 %, ST-Imputer outperforms OxyGenerator by 5.94 %, this indicates that under high missing rates, our approach still achieves better imputation performance than other baselines.

To validate the facilitative effect of data imputation on downstream tasks, we select the top three methods with the best performance in Table 3 to impute data, and apply the missing value compatible prediction model XGBoost, using the past 12 time steps to predict the next 12 time steps [11]. The prediction results are shown in Table 5. Ori. represents the raw data without imputation. Due to the original missing rate of the North Atlantic Ocean being too high (up to 99.5 %) for training, only the imputed data is used for prediction. The prediction results show that the performance is influenced by data completeness and the quality of imputation. Our model achieves the best prediction results and can provide support for downstream tasks.

5.5. Ablation study (RQ2)

To evaluate the effectiveness of the key components, we conduct a detailed ablation experiment on ST-Imputer by removing different components to get different variants.

- **w/o LDP**: remove local dependency perception module.
- **w/o MVC**: remove multi-variate correlation learning module.
- **w/o MSC**: remove multi-scale correlation learning module.
- **w/o GDP**: remove global dependency perception module.
- **w/o Cont**: remove imputation context data and retain only the imputation target as the input to the model.
- **w/o** $\mathcal{L}_{\text{phy}}$: remove physical gradient loss represented by Eq. (31).

The results of the ablation study are shown in Fig. 6. The performance of the model significantly declined after removing each module, demonstrating the effectiveness of each module.

First, **w/o Cont** shows the greatest performance decline, especially on the North Atlantic dataset with a high original missing rate, where performance decreased by 11.31 %. This indicates that when a single dataset has a high missing rate, providing auxiliary spatiotemporal associations through context data is extremely important, as relying solely on a single target dataset cannot capture sufficient spatiotemporal features. Second, **w/o GDP** also shows a decrease in RMSE of 5.78 % and 9.98 % on two datasets compared to ST-Imputer, which demonstrates that learning global multi-variate correlation can capture the interactions between context data and target data across the entire region from a broader perspective. Furthermore, **w/o LDP** shows performance declines of 3.72 % and 7.26 %, respectively, demonstrating that learn multi-variate correlation can effectively express the associations between context data and target data in highly similar regions

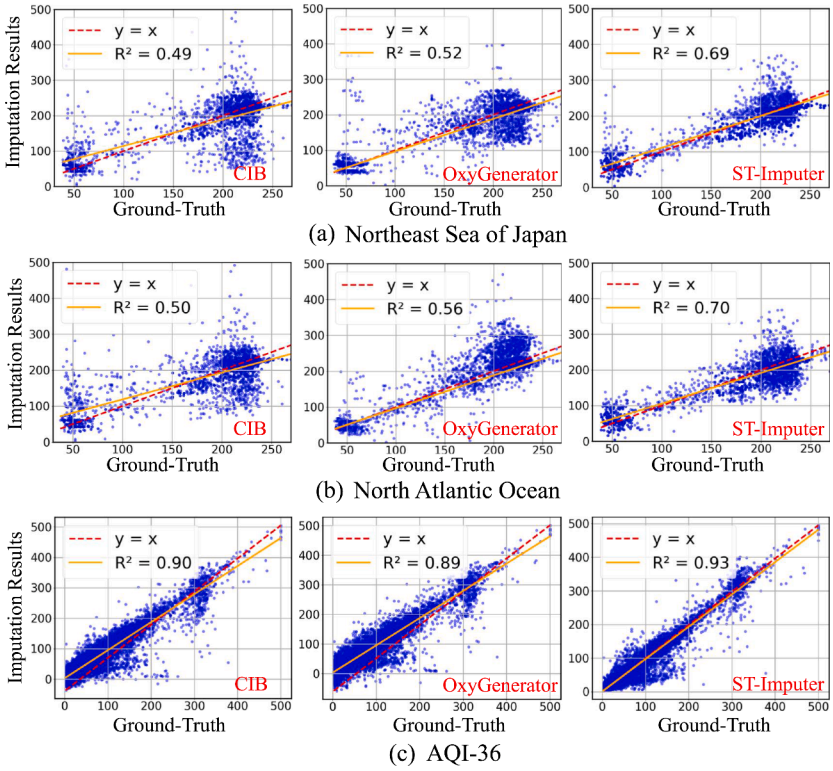

Fig. 8. Visualization of the imputation results, the yellow line represents the data trend.

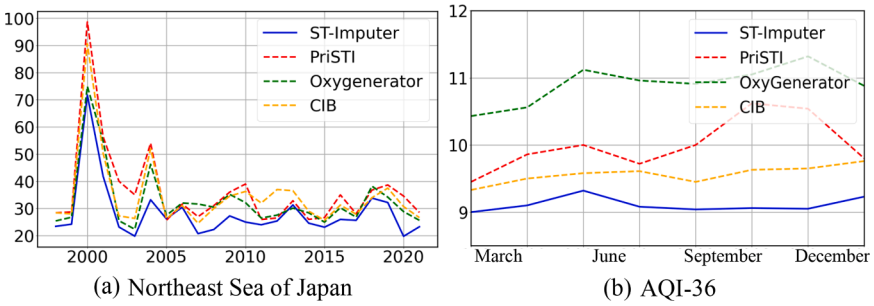

Fig. 9. Performance comparison of MAE over time.

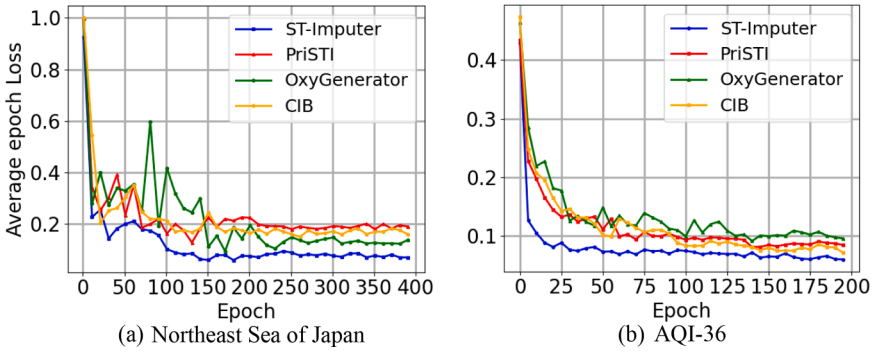

Fig. 10. Comparison of average loss with SOTA methods.

from a local perspective, providing the target data with rich spatiotemporal representations. MVC and MSC are submodules of LDP, compared to ST-Imputer, **w/o MVC** and **w/o MSC** decrease by 2.22 % and 1.74 %, demonstrating that multi-variate correlation learning and multi-scale correlation learning can progressively extract high-order potential information from multi-variate associations. Additionally, compared to **w/o MSC**, the performance decline of **w/o MVC** is more significant, further demonstrating the effectiveness of capturing correlations among multiple variables. **w/o** $\mathcal{L}_{\text{phy}}$ also shows performance declines, indicating that constrain the gradient variations between context data and the target to align with physical trends, the model can focus more on the impact of other physical features on the target data during noise prediction.

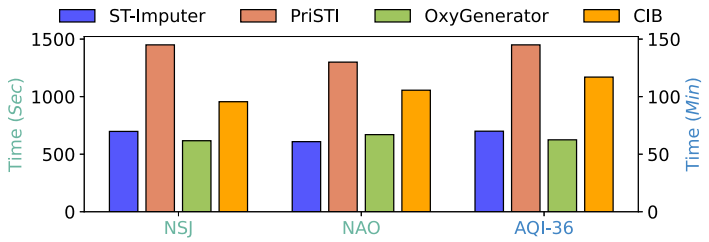

Fig. 11. Comparison of time consumption. NSJ and NAO refer to the left y-axis, AQI-36 refers to the right y-axis.

5.6. Impact of hyper-parameters (RQ3)

As shown in Fig. 7, we further conduct the sensitivity analysis for critical hyper-parameters. The results of different embedding dimensions show that as the embedding dimension increases, model performance improves significantly in the early stages. However, as the embedding dimension continues to grow, the increased sparsity of the embedding space introduces additional noise into the correlation between variables, reducing the effectiveness of m ulti-variable embeddings.

As the diffusion steps increase, the model can progressively reconstruct high-quality data during denoising, enhancing the diversity of generated samples. However, a larger step leads to faster noise accumulation in datasets with high missing rates (*e.g.*, dissolved oxygen). Under high missing rates, the model becomes more sensitive to noise and tends to generate unrealistic values. Therefore, for the dissolved oxygen dataset with limited available data, a smaller step can help reduce the negative impact of noise.

A small number of attention heads limits the model's ability to focus on different positions simultaneously. As the number of heads increases, the model can effectively learn complex multiple dependencies. However, when the number of attention heads is too high, the feature becomes dispersed across multiple heads, with each head capturing less useful information, making it difficult to capture important features.

5.7. Error distribution of imputation results (RQ4)

We use scatter plots to visualize the imputation results in AQI-36 and the dissolved oxygen datasets. From Fig. 8, it can be observed that ST-Imputer has a higher R^2 compared to the SOTA model, indicating that the imputation results generated by ST-Imputer have a strong correlation with the observed data. Additionally, in the Northeast Sea of Japan and North Atlantic Ocean datasets, CIB and OxyGenerator show varying degrees of deviation from the $y{=}x$, the imputation results of ST-Imputer are relatively more concentrated around the $y{=}x$, with the improvement of 32.69 % and 25 % in R^2.

We also analyze the variation of imputation error over time, as shown in Fig. 9. On the North Atlantic Ocean and AQI-36 datasets, ST-Imputer consistently demonstrates optimal imputation performance across different times. Notably, there is a peak error in dissolved oxygen data in 2000. However, ST-Imputer shows the smallest error compared to other SOTA models, indicating its ability to fully capture latent correlations between variables in extreme cases, mitigating the imputation instability caused by sparse and shallow features.

5.8. Efficiency comparison (RQ5)

We perform standardization on the data, retain only the mean squared error loss, and compare the model's convergence times. From Fig. 10, it can be observed that, compared to the SOTA models, ST-Imputer converges at the earliest epoch and achieves the lowest loss value, demonstrating that ST-Imputer design is highly compatible with the spatiotemporal imputation task. As shown in Fig. 11, we further compare the time consumed for model convergence. On three datasets, CIB has the highest time consumption, while OxyGenerator, due to its simple structure of MLP and graph message passing, has a relatively low time consumption. Our model outperforms CIB and PriSTI in efficiency by an average of 50.33 % and 36.49 %. Moreover, we outperform OxyGenerator by 9.1 % on the NAO dataset, and achieve comparable efficiency to OxyGenerator on the NSJ and AQI-36 datasets, proving that our model not only achieves optimal imputation performance but also exhibits excellent convergence efficiency.

6. Conclusion

We present a novel diffusion-based model for spatiotemporal missing values imputation, which incorporates multiple context data to alleviate the issue of insufficient features caused by excessive missing rate in target data. Notably, our model can effectively capture multi-scale correlation features among context data and target data, and the dual-view features based noise prediction module accurately imputes missing values with the assistance of global and local features. Experiments on two marine observation datasets and an urban environment dataset demonstrate the superiority of the proposed ST-Imputer in the spatiotemporal missing imputation task. In future work, we will investigate domain physics-embedded neural network architectures for spatiotemporal imputation to boost the model's interpretability and generalizability.

CRediT authorship contribution statement

Xingyu Zhao: Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis; **Jianpeng Qi:** Writing – review & editing, Supervision, Conceptualization; **Bin Lu:** Writing – review & editing, Investigation, Data curation; **Lei Zhou:** Writing – review & editing, Methodology, Investigation; **Lei Cao:** Writing – review & editing, Validation, Formal analysis; **Junyu Dong:** Writing – review & editing, Supervision, Conceptualization; **Yanwei Yu:** Writing – review & editing, Resources, Conceptualization.

Data availability

Data will be made available on request.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of China under Grant No. 62176243, the Fundamental Research Funds for the Central Universities under Grant No 202442005, and the Natural Science Foundation of Qingdao under Grant No.24-8-4-zrjj-3-jch.

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

[1] K. Zhang, F. Zhou, L. Wu, N. Xie, Z. He, Semantic understanding and prompt engineering for large-scale traffic data imputation, Inf. Fusion 102 (2024) 102038.

[2] A. Zhang, S. Song, Y. Sun, J. Wang, Learning individual models for imputation, in: 2019 IEEE 35th International Conference on Data Engineering (ICDE), IEEE, 2019, pp. 160–171.

[3] C. Liu, J. Wen, Z. Wu, X. Luo, C. Huang, Y. Xu, Information recovery-driven deep incomplete multiview clustering network, IEEE Trans. Neural Netw. Learn. Syst. 35 (11) (2023) 15442–15452.

[4] J. Wen, G. Xu, Z. Tang, W. Wang, L. Fei, Y. Xu, Graph regularized and feature aware matrix factorization for robust incomplete multi-view clustering, IEEE Trans. Circuits Syst. Video Technol. 34 (5) (2023) 3728–3741.

[5] D. Xu, H. Peng, Y. Tang, H. Guo, Hierarchical spatio-temporal graph convolutional neural networks for traffic data imputation, Inf. Fusion 106 (2024) 102292.

[6] L.R. Landerman, K.C. Land, C.F. Pieper, An empirical evaluation of the predictive mean matching method for imputing missing values, Sociol. Methods Res. 26 (1) (1997) 3–33.

[7] M. Perini, M. Nikolic, In-database data imputation, Proc. ACM Manag. Data 2 (1) (2024) 1–27.

[8] L. Beretta, A. Santaniello, Nearest neighbor imputation algorithms: a critical evaluation, BMC Med. Inform. Decis. Mak. 16 (2016) 197–208.

[9] H.-F. Yu, N. Rao, I.S. Dhillon, Temporal regularized matrix factorization for high-dimensional time series prediction, Adv. Neural Inf. Process. Syst. 29 (2016) 21.

[10] X. Chen, Z. He, Y. Chen, Y. Lu, J. Wang, Missing traffic data imputation and pattern discovery with a bayesian augmented tensor factorization model, Transp. Res. Part C Emerg. Technol. 104 (2019) 66–77.

[11] M. Liu, H. Huang, H. Feng, L. Sun, B. Du, Y. Fu, Pristi: a conditional diffusion framework for spatiotemporal imputation, in: 2023 IEEE 39th International Conference on Data Engineering (ICDE), IEEE, 2023, pp. 1927–1939.

[12] B. Lu, Z. Zhao, L. Han, X. Gan, Y. Zhou, L. Zhou, L. Fu, X. Wang, C. Zhou, J. Zhang, Oxygenerator: reconstructing global ocean deoxygenation over a century with deep learning, in: Proceedings of the 41st International Conference on Machine Learning, ICML, Vienna, Austria, 2024.

[13] S.K. Lauvset, N. Lange, T. Tanhua, H.C. Bittig, A. Olsen, A. Kozyr, M. Álvarez, K. Azetsu-Scott, P.J. Brown, B.R. Carter, L. Cotrim da Cunha, M. Hoppema, M.P. Humphreys, M. Ishii, E. Jeansson, A. Murata, J.D. Müller, F.F. Pérez, C. Schirnick, R. Steinfeldt, T. Suzuki, A. Ulfsbo, A. Velo, R.J. Woosley, R.M. Key, The annual update GLODAPv2.2023: the global interior ocean biogeochemical data product, Earth Syst. Sci. Data 16 (4) (2024) 2047–2072. https://doi.org/10.5194/essd-16-2047-2024

[14] M. Choi, C. Lee, Conditional information bottleneck approach for time series imputation, in: The Twelfth International Conference on Learning Representations, OpenReview.net, 2024.

[15] S. Zhang, S. Wang, H. Miao, H. Chen, C. Fan, J. Zhang, Score-CDM: score-weighted convolutional diffusion model for multivariate time series imputation, in: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, 2024.

[16] Q. Li, R. Li, K. Ji, W. Dai, Kalman filter and its application, in: 2015 8th International Conference on Intelligent Networks and Intelligent Systems, IEEE, ICINIS, 2015, pp. 74–77.

[17] B. Pfaff, VAR, SVAR And SVEC models: implementation within r package vars, J. Stat. Softw. 27 (2008) 1–32.

[18] V. Fortuin, D. Baranchuk, G. Rätsch, S. Mandt, Gp-vae: deep probabilistic time series imputation, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 1651–1661.

[19] W. Cao, D. Wang, J. Li, H. Zhou, L. Li, Y. Li, Brits: bidirectional recurrent imputation for time series, Adv. Neural Inf. Process. Syst. 31 (2018) 6776–6786.

[20] X. Tang, H. Yao, Y. Sun, C. Aggarwal, P. Mitra, S. Wang, Joint modeling of local and global temporal dynamics for multivariate time series forecasting with missing values, in: Proceedings of the AAAI Conference on Artificial Intelligence, AAAI, 2020, pp. 5956–5963.

[21] G. Liang, P. Tiwari, S. Nowaczyk, S. Byttner, Higher-order spatio-temporal physics-incorporated graph neural network for multivariate time series imputation, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, 2024, pp. 1356–1366.

[22] A. Cini, I. Marisca, C. Alippi, Filling the g_ap_s: multivariate time series imputation by graph neural networks, in: International Conference on Learning Representations, 2022.

[23] T. Nie, G. Qin, W. Ma, Y. Mei, J. Sun, Imputeformer: low rankness-induced transformers for generalizable spatiotemporal imputation, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 2260–2271.

[24] Y. Liang, Z. Zhao, L. Sun, Dynamic spatiotemporal graph convolutional neural networks for traffic data imputation with complex missing patterns, arXiv:2109.08357 24 (2021) 54.

[25] Y. Tashiro, J. Song, Y. Song, S. Ermon, Csdi: conditional score-based diffusion models for probabilistic time series imputation, Adv. Neural Inf. Process. Syst. 34 (2021) 24804–24816.

[26] W. He, J. Huang, J. Gu, J. Zhang, Y. Bai, Filling the Missings: Spatiotemporal Data Imputation by Conditional Diffusion, arXiv:2506.07099 (2025).

[27] Z. Liu, X. Zhao, Y. Song, RDPI: a refine diffusion probability generation method for spatiotemporal data imputation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 39, 2025, pp. 12255–12263.

[28] B. Zhang, R. Miao, Z. Chen, Spatial-temporal traffic data imputation based on dynamic multi-level generative adversarial networks for urban governance, Appl. Soft. Comput. 151 (2024) 111128.

[29] J. Zhou, J. Li, G. Zheng, X. Wang, C. Zhou, MTSCI: a conditional diffusion model for multivariate time series consistent imputation, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, 2024, pp. 3474–3483.

[30] H. Cao, C. Tan, Z. Gao, Y. Xu, G. Chen, P.-A. Heng, S.Z. Li, A survey on generative diffusion models, IEEE Trans. Knowl. Data Eng. 36 IEEE(7)(2024)2814–2830.

[31] Y. Wang, X. Wang, A.-D. Dinh, B. Du, C. Xu, Learning to schedule in diffusion probabilistic models, in: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2023, pp. 2478–2488.

[32] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, in: Advances in Neural Information Processing Systems, 33, Neurips, 2020, pp. 6840–6851.

[33] N. Gruber, J.N. Galloway, An earth-system perspective of the global nitrogen cycle, Nature 451 (7176) (2008) 293–296.

[34] K. Kamboj, A.K. Mathur, Assessment of relationship between meteorological parameters and air quality index of Kota, Rajasthan, India, Curr. World Environ. 18 (1) (2023) 265.

[35] X. Yi, Y. Zheng, J. Zhang, T. Li, ST-MVL: filling missing values in geo-sensory time series data, in: Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI, 2016.

[36] S. Zhang, Y. Yang, K. Xie, J. Gao, Z. Zhang, Q. Niu, G. Wang, Z. Che, L. Mu, S. Jia, Spatial-temporal siamese convolutional neural network for subsurface temperature reconstruction, IEEE Trans. Geosci. Remote Sens. 62 (2024) 1–16. https://doi.org/10.1109/TGRS.2023.3348653

[37] I.R. White, P. Royston, A.M. Wood, Multiple imputation using chained equations: issues and guidance for practice, Stat. Med. 30 (4) (2011) 377–399.

[38] Y. Fan, X. Yu, R. Wieser, D. Meakin, A. Shaton, J.-N. Jaubert, R. Flottemesch, M. Howell, J. Braid, L. Bruckman, et al., Spatio-temporal denoising graph autoencoders with data augmentation for photovoltaic data imputation, Proc. ACM Manag. Data 1 (1) (2023) 1–19.

[39] X. Ren, K. Zhao, P.J. Riddle, K. Taskova, Q. Pan, L. Li, DAMR: Dynamic adjacency matrix representation learning for multivariate time series imputation, Proc. ACM Manag. Data 1 (2) (2023) 1–25.

[40] D. Wang, Y. Yan, R. Qiu, Y. Zhu, K. Guan, A. Margenot, H. Tong, Networked time series imputation via position-aware graph enhanced variational autoencoders, in: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2023, pp. 2256–2268.

[41] Z. Lai, D. Zhang, H. Li, D. Zhang, H. Lu, C.S. Jensen, ReCTSi: resource-efficient correlated time series imputation via decoupled pattern learning and completeness-aware attentions, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 1474–1483.