國立東華大學應用數學系

碩士論文

基於結構化狀態空間擴散模型與自適應固定秩克利金法之未知區域時空預測

Spatiotemporal Prediction of Unknown Areas Based on Structured State Space Diffusion and Adaptive Fixed Rank Kriging

研究生：許堯智
指導教授：黃灝勻 博士

中華民國 115 年 6 月

國立東華大學應用數學系

碩士論文

基於結構化狀態空間擴散模型與自適應固定秩克利金法之未知區域時空預測

Spatiotemporal Prediction of Unknown Areas Based on Structured State Space Diffusion and Adaptive Fixed Rank Kriging

研究生：許堯智

指導教授：黃灝勻 博士

中華民國 115 年 6 月

學位考試委員會審定書

國立東華大學＿＿＿＿＿＿＿＿＿＿系所

研究生　＿＿＿＿＿＿＿＿君所提之論文

經本委員會審查並舉行口試，認為

符合碩士學位標準。

學位考試委員會召集人＿＿＿＿＿＿＿＿簽章

指導教授＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

委　　員＿＿＿＿＿＿＿＿簽章

系 主 任＿＿＿＿＿＿＿＿簽章
（所長）

中華民國　　年　　月　　日

國立東華大學
NATIONAL DONG HWA UNIVERSITY

學位論文原創性聲明書
DECLARATION OF THESIS/DISSERTATION ORIGINALITY

學位論文題目：*基於結構化狀態空間擴散模型與自適應固定秩克利金法之未知區域時空預測*

Thesis/Dissertation Title：*Spatiotemporal Prediction of Unknown Areas Based on Structured State Space Diffusion and Adaptive Fixed Rank Kriging*

本人在此聲明，所呈交的學位論文是在指導教授黃灝勻的指導下，由個人獨立研究所完成之最終版本。本人對論文內容負責，除了文中已經標註引用處的內容外，論文不包含任何其他人已經發表或撰寫過的研究成果。對本研究及學位論文做出重要貢獻的個人和組織，均已在文中以明確方式標明。

該論文內容如有違反學術道德或學術規範的行為，如造假、變造、抄襲、研究成果重複發表或未適當引註、以違法或不當手段影響論文審查、不當作者列名等，本人願意承擔由此而產生的法律責任和法律後果。

I declare that the thesis/dissertation herein is the final version of my work, which is composed and accomplished individually under the guidance of my supervisor, Prof. Hao-Yun Huang. I am responsible for the contents of this thesis/dissertation: It contains no research result that was previously published or written by another person. Information derived from published and unpublished work of others has been acknowledged in the text, and a list of references is given . Any contribution made by other individual or organization is explicitly acknowledged in the thesis/dissertation.

If any research misconduct, including fabrication, falsification, or plagiarism in proposing, performing, or reviewing research, or in reporting research results, is discovered in my thesis/dissertation, I am willing to bear corresponding legal responsibilities and all the results therefrom.

聲明人 Declarant：

日期 Date：_____(yyyy/mm/dd)

1140303修訂

摘要

隨著感測技術與資料蒐集系統之發展，時空資料於交通、氣象與環境監測等領域中大量累積，然而實務上仍普遍存在觀測缺漏與未觀測地點之問題，對後續分析與預測造成挑戰。爲兼顧時間序列之動態特性與空間資料之相關結構，本研究提出一種整合式時空預測框架 $SSSD^{S4+AFRK}$ ，結合結構化狀態空間擴散模型（Structured State Space Diffusion with S4 layers, $SSSD^{S4}$ ）與自適應固定秩克利金法（Adaptive Fixed Rank Kriging, AFRK），以提升缺失資料重建與未來預測之準確性。

在方法上，本研究首先利用 $SSSD^{S4}$ 擷取時間序列之長期依賴與非線性動態特徵，並於模型訓練過程中引入 AFRK 對擴散過程中的雜訊估計進行空間結構調整，以強化時空一致性；於推論階段，則結合時間預測結果與空間基底函數進行未觀測地點之空間插值與未來預測。

實驗採用 Weather2K、MERRA-2 與 2b-8 等多樣化時空資料集，並與多種基準模型（包含 TFT、VAR、SVGP 與 STDK）進行比較。結果顯示，所提出之 $SSSD^{S4+AFRK}$ 在多數情境下，特別是在未觀測地點之未來預測任務中，能顯著降低平均平方預測誤差（MSPE），並於高維度及高變異資料中展現穩定且優異之預測效能。相較之下，僅考慮時間或空間資訊之模型，其預測表現較易受到資料分布與變異性影響。

本研究證實透過深度時間序列模型與空間統計方法之整合，可有效提升時空資料在缺失與外插情境下之預測能力，並提供一具擴展性之建模框架，對於未來大規模時空資料分析具有實務應用潛力。

關鍵字：時空預測、結構化狀態空間模型、擴散模型、自適應固定秩克利金、空間插值、時間序列分析

Abstract

With the advancement of sensing technologies and data acquisition systems, large volumes of spatiotemporal data have been accumulated in domains such as transportation, meteorology, and environmental monitoring. However, in practice, missing observations and unobserved locations remain prevalent, posing significant challenges for subsequent analysis and forecasting. To address both the dynamic nature of temporal sequences and the spatial dependency structure of data, this study proposes an integrated spatiotemporal forecasting framework, $\text{SSSD}^{\text{S4+AFRK}}$, which combines Structured State Space Diffusion with S4 layers (SSSD^{S4}) and Adaptive Fixed Rank Kriging (AFRK), aiming to improve the accuracy of missing data reconstruction and future prediction.

Methodologically, the proposed approach first leverages SSSD^{S4} to capture long-range dependencies and nonlinear temporal dynamics in time series data. During training, AFRK is incorporated to adjust spatial structures in the noise estimation process of the diffusion model, thereby enhancing spatiotemporal consistency. During inference, the model further integrates temporal predictions with spatial basis functions to perform spatial interpolation and future forecasting at unobserved locations.

Experiments are conducted on diverse spatiotemporal datasets, including Weather2K, MERRA-2, and 2b-8, and the proposed method is compared against several baseline models, including TFT, VAR, SVGP, and STDK. Results demonstrate that $\text{SSSD}^{\text{S4+AFRK}}$ consistently reduces mean squared prediction error (MSPE) under most settings, particularly in forecasting tasks at unobserved locations. Moreover, it exhibits stable and superior performance in high-dimensional and highly variable data regimes. In contrast, models that consider only temporal or spatial information are more susceptible to data distribution shifts and variability.

Overall, the findings validate that integrating deep temporal sequence models with spatial statistical methods can effectively enhance forecasting performance under missing and extrapolation scenarios. Furthermore, the proposed framework provides a scalable modeling paradigm with strong potential for large-scale spatiotemporal data analysis in real-world applications.

Keywords: spatiotemporal forecasting, Structured State Space Models, Diffusion Models, Adaptive Fixed Rank Kriging, spatial interpolation, time series analysis

Contents

1 Introduction

With the rapid advancement of sensing technologies and digital infrastructure, researchers and the general public can now more easily access diverse types of data, such as traffic flow, water quality monitoring, and satellite remote sensing, through online platforms, real-time sensors, and various open-access databases.

However, during the data collection process, instruments may still experience failures, malfunctions, or maintenance downtime, resulting in missing observations and thus incomplete spatiotemporal datasets. When attempting to predict future trends at locations with missing values, an important challenge arises that how to simultaneously account for the characteristics of temporal dynamics and the structural dependencies inherent in spatial data (Decorte et al., 2024).

To address this issue, this study proposes a methodology that integrates time series forecasting with spatial statistical modeling. We employ a Structured State Space Diffusion model (SSSD) (Alcaraz and Strodthoff, 2023) to capture temporal dependencies in the data, while incorporating Adaptive Fixed Rank Kriging (AFRK) (Tzeng and Huang, 2018) to characterize spatial correlations. Through this joint spatiotemporal framework, we aim to enhance the reconstruction of missing data and improve the predictive accuracy of future trends.

2 Related Works

Spatiotemporal data analysis plays a crucial role in many fields, including transportation engineering, hydrological monitoring, and environmental science. With the advancement of observation technologies and the widespread deployment of sensing devices, large-scale and high-resolution spatiotemporal data can be continuously collected. However, during the data acquisition process, missing values and outliers may still arise due to sensor malfunctions, communication interruptions, or maintenance operations, which increase the difficulty of data analysis and predictive modeling (Decorte et al., 2024; Little and Rubin, 2002).

Previous studies have often focused separately on either temporal models or spatial models. Nevertheless, because spatiotemporal data simultaneously exhibit temporal dependencies and spatial correlation structures, recent research has increasingly developed spatiotemporal

modeling approaches that integrate both temporal and spatial information in order to improve the accuracy of prediction and estimation (N. Cressie and Wikle, 2011; Shi et al., 2015).

In terms of temporal modeling, traditional approaches, such as the Autoregressive Integrated Moving Average (ARIMA) model (Box and Jenkins, 1976), are effective in capturing univariate temporal dependencies. The Vector Autoregression (VAR) model (Sims, 1980) further extends these capabilities to multivariate systems, describing the dynamic interdependencies among variables. Nevertheless, these linear models exhibit inherent limitations when processing complex nonlinear structures (Primiceri, 2005).

Subsequently, the Long Short-Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) was developed to enhance the capture of nonlinear dynamics; however, it remains constrained by computational inefficiency and the loss of long-range information when dealing with extensive time-series data (Bengio, Simard, and Frasconi, 1994). In contrast, the Temporal Fusion Transformer (TFT) (Lim et al., 2020) introduces variable selection mechanisms and attention architectures to integrate diverse types of input variables. This improves the model's ability to identify complex spatio-temporal features and enhances predictive interpretability. Nonetheless, when confronted with extreme high-dimensional data or observation sequences characterized by substantial continuous missing values, the attention mechanism in TFT often struggles to maintain stable representation capabilities, leaving room for improvement in its robustness against latent stochastic perturbations (Wu et al., 2022).

Building upon these foundations, recently proposed State Space Model (SSM) (Kalman, 1960) and Structured State Space Diffusion (SSSD) model (Alcaraz and Strodthoff, 2023) demonstrate superior advantages in long-sequence modeling. By integrating S4 layers with a diffusion probabilistic framework, these approaches achieve high-quality data imputation and forecasting performance while maintaining effective control over computational costs.

On the spatial modeling side, the Kriging method in spatial statistics (N. A. C. Cressie, 1993) has been widely applied to spatial interpolation and missing value estimation. However, when dealing with large-scale datasets, the computational cost of Kriging can be substantial, limiting its feasibility for real-time applications. To address this issue, Fixed Rank Kriging (FRK) (N. Cressie and Johannesson, 2008) and its extension, Adaptive Fixed Rank Kriging (AFRK), have been proposed (N. Cressie and Johannesson, 2008; Tzeng and Huang, 2018). These methods effectively reduce computational complexity and improve computational efficiency.

Temporal models emphasize the dynamic evolution of data, whereas spatial models capture the dependency structures among neighboring locations. Recent studies have increasingly recognized the importance of integrating these two dimensions, with applications in areas such as transportation demand forecasting, meteorological simulation, and environmental monitoring. Motivated by this context, the present study proposes a spatiotemporal framework that combines SSSD and AFRK to enhance predictive accuracy and robustness under scenarios with missing data.

2.1 Time Series Models

2.1.1 Vector Autoregression Model

The Vector Autoregression (VAR) model, introduced by Sims (1980), serves as an extension of the univariate Autoregressive (AR) model, designed to capture the dynamic interdependencies among multiple time-varying variables. Distinguishing itself from univariate models, VAR treats all variables within the system as endogenous variables and describes their lagged influences through a system of simultaneous equations.

For a p-order vector autoregression model with k variables, denoted as VAR(p), the mathematical expression is defined as:

$$\boldsymbol{y}_t = \boldsymbol{c} + \sum_{i=1}^{p} \boldsymbol{\Phi}_i \boldsymbol{y}_{t-i} + \boldsymbol{\epsilon}_t, \tag{1}$$

where $\boldsymbol{y}_t \in \mathbb{R}^k$ represents the observation vector at time t, $\boldsymbol{c}$ is the vector of intercept terms, $\boldsymbol{\Phi}_i \in \mathbb{R}^{k \times k}$ denotes the lag operator matrix that quantifies the influence of variables at time $t - i$ on the current state, and $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \boldsymbol{\Omega})$ is the noise vector.

The primary advantage of the VAR model lies in its ability to analyze causal transmission mechanisms and dynamic interactions through Impulse Response Functions (IRF) and Variance Decomposition (Sims, 1980). However, as the number of variables k or the lag order p increases, the number of parameters grows quadratically, which often leads to overfitting issues. Furthermore, traditional linear VAR models struggle to capture complex nonlinear structures and the time-varying nature of parameters (Primiceri, 2005).

2.1.2 Temporal Fusion Transformers

Given the limitations of traditional linear multivariate models in capturing complex nonlinear dynamics and high-dimensional feature correlations, recent research has increasingly shifted toward deep learning architectures to enhance predictive performance. To overcome the challenges faced by conventional recurrent architectures in handling long-range dependencies and to effectively integrate diverse information sources, including static covariates, known future inputs, and historical observations, Lim et al. (2020) proposed the Temporal Fusion Transformer (TFT). The TFT is a deep learning architecture specifically designed for multi-horizon time-series forecasting. Its design emphasizes the allocation of weights to input variables of different natures through specialized network components, thereby enhancing the interpretability of prediction results and addressing the inadequacies of traditional statistical models when processing large-scale heterogeneous data.

The implementation of TFT relies on Gated Residual Networks (GRN), which regulate information flow via Gated Linear Units (GLU). This component enables the model to automatically adjust the depth of nonlinear transformations based on data characteristics. For an input vector $\boldsymbol{a}$ and an optional context vector $\boldsymbol{c}$, the operation is as follows:

$$\text{GRN}_{\omega}(\boldsymbol{a}, \boldsymbol{c}) = \text{LayerNorm}(\boldsymbol{a} + \text{GLU}_{\omega}(\boldsymbol{\eta}_1)), \tag{2}$$

where $\boldsymbol{\eta}_1$ is the feature vector transformed by a weight matrix. This mechanism ensures high flexibility when processing sequences of varying complexities, effectively preventing deep networks from overfitting on simpler datasets.

For spatio-temporal data containing numerous external factors, TFT introduces Variable Selection Networks (VSN) to identify key variables from a large pool of input features. By assigning a weight $\nu_t^{(i)}$ to each feature, the model can automatically ignore redundant information and focus on influential factors. The integrated feature vector is represented as:

$$\tilde{\boldsymbol{\xi}}_t = \sum_{i=1}^{m} \nu_t^{(i)} \tilde{\boldsymbol{\xi}}_t^{(i)}, \tag{3}$$

where $\tilde{\boldsymbol{\xi}}_t^{(i)}$ is the processed feature vector. This design significantly enhances the model's robustness when dealing with high-dimensional feature inputs and allows researchers to intuitively quantify the contribution of various variables to the prediction results.

Regarding the capture of temporal relationships, TFT utilizes a modified Temporal Self-Attention mechanism to handle long-term dependencies. Compared to the standard Transformer architecture, TFT incorporates gating layers for residual connections within the attention layer and integrates historical and future spatio-temporal context through a decoder to identify the most influential time steps for the current prediction. TFT not only demonstrates superior predictive accuracy but also grants deep learning models the ability to interpret the significance of specific time steps or features (Lim et al., 2020).

2.1.3 State Space Model

The State Space Model (SSM) is a class of mathematical models that describes dynamic systems or sequential data through a latent state vector. Initially proposed by Kalman (1960) within the fields of control theory and filtering, SSMs were designed to address optimal filtering and prediction problems for linear dynamic systems. Subsequently, researchers such as Gu, Goel, and Ré (2022) extended this concept to deep learning architectures for long-sequence time series modeling, demonstrating that SSMs outperform traditional RNNs and LSTMs in capturing long-range dependencies and maintaining stable gradients (Gu, Goel, and Ré, 2022).

Given a one-dimensional input signal sequence $\boldsymbol{u}(t)$ and a one-dimensional output signal sequence $\boldsymbol{y}(t)$, the basic formulation of an SSM is

$$\begin{aligned} \boldsymbol{x}'(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t); \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t), \end{aligned} \tag{4}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^N$ is an N-dimensional latent state that maps the input $\boldsymbol{u}(t)$, $\boldsymbol{x}'(t) = \frac{d}{dt}\boldsymbol{x}(t)$ denotes its time derivative, $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ is the state matrix, and $\boldsymbol{B} \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{C} \in \mathbb{R}^{1\times N}$ are the input and output matrices, respectively, characterizing how the input influences the state and how the state maps to the output. The term $\boldsymbol{D} \in \mathbb{R}$ is the feedthrough matrix, allowing the input to directly affect the output, and is typically set to zero. In deep learning contexts, these parameters are generally learned via gradient descent.

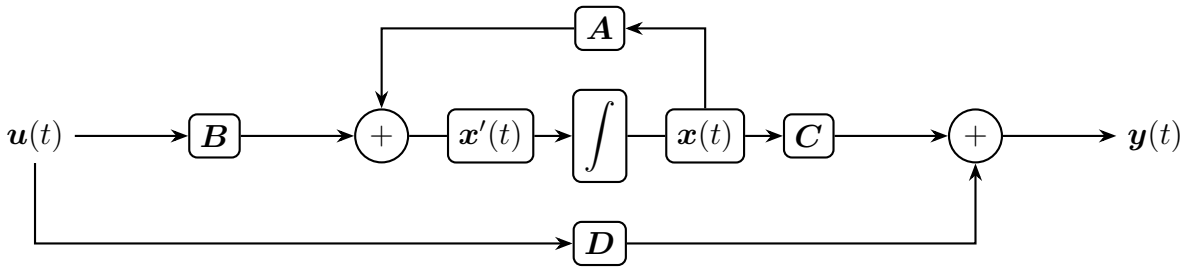

Figure 1: Typical State Space Model.

To address the practical issue in which the gradients of SSMs may increase or decrease exponentially with sequence length, Gu, Goel, and Ré (2022) introduced the HiPPO (High-order Polynomial Projection Operators) matrix (Gu, Dao, et al., 2020) to replace the original random matrix $\boldsymbol{A}$ in (4).

$$\boldsymbol{A}_{nk} = -\begin{cases} (2n+1)^{1/2}(2k+1)^{1/2}, & \text{if } n > k; \\ n+1, & \text{if } n = k; \\ 0, & \text{if } n < k. \end{cases} \tag{5}$$

As shown in (5), the HiPPO matrix dynamically evaluates the importance of each past time step as time evolves, enabling adaptive memory updates and retaining all historical information. By replacing the original random matrix $\boldsymbol{A}$ with the HiPPO matrix, the latent state $\boldsymbol{x}(t)$ can effectively store the historical information of the input sequence $\boldsymbol{u}(t)$ while avoiding gradient explosion or vanishing. Experimental results demonstrate that this design not only enhances computational stability but also significantly improves performance in long-sequence forecasting tasks (Gu, Goel, and Ré, 2022).

2.1.4 Structured State Space Model

Building upon the theory discussed in the previous section, Gu, Goel, and Ré (2022) proposed the Structured State Space Sequence Model (S4). This model aims to discretize the continuous-time State Space Model and embed it within deep learning frameworks for handling long sequential data.

S4 is grounded in the SSM formulation and discretizes (4) to enable its application to discrete input sequences. Let the step size be defined as Δ, then the discretized SSM can be written as follows:

$$\begin{aligned} \boldsymbol{x}_k &= \overline{\boldsymbol{A}}\boldsymbol{x}_{k-1} + \overline{\boldsymbol{B}}\boldsymbol{u}_k; \\ \boldsymbol{y}_k &= \overline{\boldsymbol{C}}\boldsymbol{x}_k, \end{aligned} \tag{6}$$

where $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}, \overline{\boldsymbol{C}}$ are the discrete approximations of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$, respectively.

$$\begin{aligned} \overline{\boldsymbol{A}} &= (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}(\boldsymbol{I} + \Delta/2 \cdot \boldsymbol{A}); \\ \overline{\boldsymbol{B}} &= (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}\Delta\boldsymbol{B}; \\ \overline{\boldsymbol{C}} &= \boldsymbol{C}. \end{aligned} \tag{7}$$

To reduce the computational cost of matrix operations, S4 diagonalizes the discretized matrices, expressing them in an equivalent form under a different basis:

$$(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}) \sim (\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}, \boldsymbol{V}^{-1}\boldsymbol{B}, \boldsymbol{C}\boldsymbol{V}), \tag{8}$$

where $\boldsymbol{V}$ is the basis transformation matrix. Moreover, the discretized SSM can be reformulated in a convolutional form to enhance parallel computation efficiency:

$$\boldsymbol{y} = \overline{\boldsymbol{K}} * \boldsymbol{u}; \tag{9}$$

$$\overline{\boldsymbol{K}} \in \mathbb{R}^L := (\overline{\boldsymbol{C}\boldsymbol{B}}, \overline{\boldsymbol{C}\boldsymbol{A}\boldsymbol{B}}, \cdots, \overline{\boldsymbol{C}\boldsymbol{A}}^{L-1}\overline{\boldsymbol{B}}), \tag{10}$$

where $\overline{\boldsymbol{K}}$ denotes the SSM convolution kernel and L represents the convolution length.

To further reduce computational complexity, S4 employs the Normal Plus Low-Rank (NPLR) parameterization to express the matrix $\overline{\boldsymbol{A}}$ as:

$$\boldsymbol{A} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^* - \boldsymbol{P}\boldsymbol{Q}^\top = \boldsymbol{V}(\boldsymbol{\Lambda} - (\boldsymbol{V}^*\boldsymbol{P})(\boldsymbol{V}^*\boldsymbol{Q})^*)\boldsymbol{V}^*, \tag{11}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix, $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{R}^{N\times r}$ are low-rank matrices, and $\boldsymbol{V} \in \mathbb{C}^{N\times N}$ is a unitary matrix.

Based on the above theoretical components, S4 integrates the HiPPO matrix, discretization, diagonalization, convolution, and NPLR parameterization, enabling efficient computa-

tion and strong performance on long-sequence tasks (Gu, Goel, and Ré, 2022).

2.1.5 Diffusion Model

Diffusion models are a class of generative models that learn data distributions through a dual process consisting of a *forward diffusion process* and a *reverse denoising process* (Sohl-Dickstein et al., 2015). In the forward process, Gaussian noise is gradually injected into the original data until it approaches a standard normal distribution. The model is then trained to learn the reverse mapping of this process in order to reconstruct the data distribution. In recent years, diffusion models have been applied to time series imputation, where diffusion and denoising are performed only on missing segments to recover complete sequences under conditional observations (Alcaraz and Strodthoff, 2023).

Let $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$ denote an original data sample. The forward process is defined as a fixed-parameter Gaussian Markov chain that simulates progressively perturbed data generation:

$$\begin{cases} q(\boldsymbol{x}_1|\boldsymbol{x}_0) = \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}); \\ q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}), \end{cases} \tag{12}$$

where β_t is the variance schedule controlling noise intensity, and $\mathcal{N}$ denotes the normal distribution.

To reconstruct the data, the model must learn the reverse mapping of this process. The reverse process is defined as:

$$\begin{cases} p_\theta(x_0) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t); \\ p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \end{cases} \tag{13}$$

where $p(\boldsymbol{x}_T) = \mathcal{N}(\boldsymbol{x}_T; 0, \boldsymbol{I})$ denotes a standard normal distribution, while μ_θ and Σ_θ represent the mean vector and covariance matrix parameterized by a neural network with parameters θ, respectively.

However, directly modeling the mean of the reverse process, μ_θ, is often difficult to optimize and may lead to unstable convergence in practice. To address this issue, Ho, Jain, and Abbeel (2020) proposed a parameterization known as the Denoising Diffusion Probabilistic

Model (DDPM), which reparameterizes $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ as

$$\mu_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\Big(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\,\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\Big), \tag{14}$$

$$\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2 \boldsymbol{I}, \qquad \sigma_t^2 = \beta_t \text{ or } \sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t, \tag{15}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$. Under this framework, $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ is used to estimate the random Gaussian noise added to $\boldsymbol{x}_t$ during the forward diffusion process, and $\boldsymbol{x}_{t-1}$ is reconstructed by removing the estimated noise component from $\boldsymbol{x}_t$.

This parameterization avoids the need to directly model complex high-dimensional data distributions, thereby substantially simplifying the training objective and improving numerical stability. Consequently, the sample at any diffusion step t can be expressed as a linear combination of the original data $\boldsymbol{x}_0$ and Gaussian noise:

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{16}$$

which allows the model to randomly sample the time step t and noise $\boldsymbol{\epsilon}$ during training without iteratively computing intermediate diffusion steps. Since this representation transforms the original problem of directly fitting the data distribution into the estimation of Gaussian noise, the training objective can be further simplified as

$$\min_\theta \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right)\right\|^2\right], \tag{17}$$

which corresponds to minimizing the Mean Squared Error (MSE) between the predicted noise and the injected noise.

For time series imputation, Alcaraz and Strodthoff (2023) further proposed a Conditional Diffusion Model that applies diffusion and denoising operations only to missing segments. During the training stage, the model receives partially observed sequences as conditional inputs to learn how to reconstruct the complete sequence. During the generation stage, the observed portions are fixed while the reverse denoising process is performed, allowing the missing segments to be reconstructed while preserving temporal consistency. By leveraging the stability and high-quality generation capability of diffusion models, this approach can effectively handle time series data with long-term dependencies or structurally missing segments.

2.1.6 Structured State Space Diffusion Model with S4 Layers

The Structured State Space Diffusion (SSSD) model, proposed by Alcaraz and Strodthoff (2023), is designed to integrate the generative stability of the DiffWave-based diffusion architecture (Kong et al., 2021) with the long-range dependency modeling capability of SSM. The model is formulated as a conditional diffusion framework for time series imputation, in which noise is applied exclusively to the missing segments while the observed portions remain unperturbed. This strategy effectively prevents information leakage and preserves conditional consistency. By learning the reverse denoising process at each diffusion step, SSSD progressively reconstructs the missing data under fixed observations, achieving high-quality imputation and generation performance.

Building upon this foundation, Alcaraz and Strodthoff (2023) further introduce the Structured State Space Diffusion model with S4 layers (SSSD^{S4}) as an enhanced variant tailored for time series tasks. SSSD^{S4} retains the conditional diffusion design of SSSD but replaces the Bidirectional Dilated Convolutional Layers in the original DiffWave architecture with Structured State Space (S4) layers, thereby improving the model's ability to capture long-range temporal dynamics. The overall architecture is illustrated in Figure 2. Experimental results demonstrate that this variant achieves more stable and accurate imputation across diverse missing patterns, outperforming diffusion models based on conventional convolutional or Transformer architectures (Alcaraz and Strodthoff, 2023).

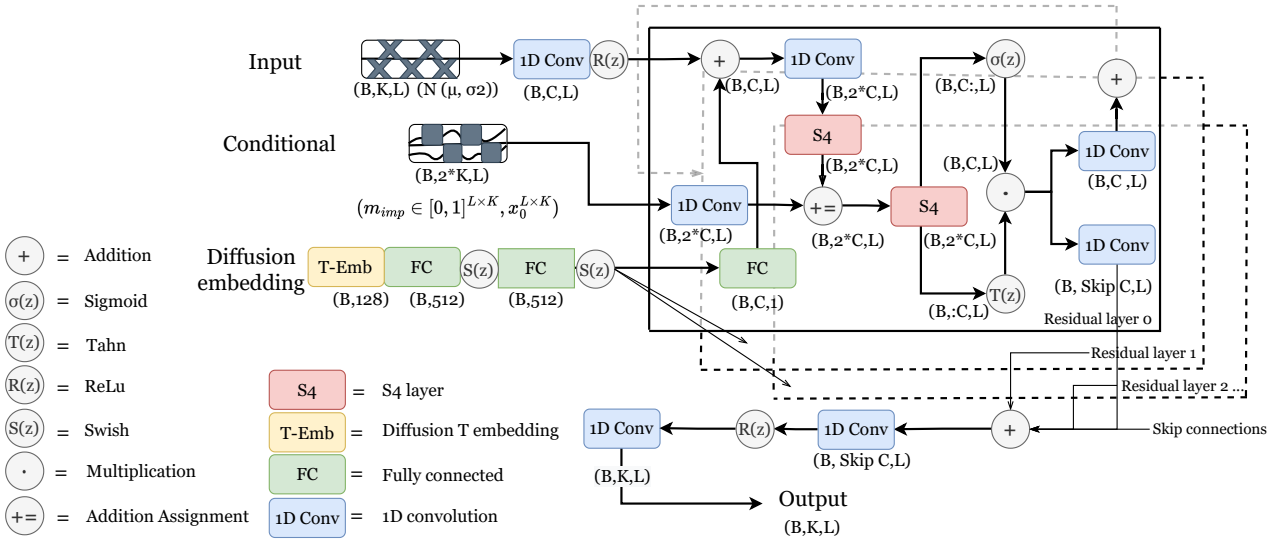

Figure 2: SSSD^{S4} model architecture.

2.2 Spatial Statistics

2.2.1 Kriging

Kriging originates from spatial statistics in geosciences and is a linear interpolation method used to estimate the values of a random field at unobserved spatial locations based on observed data (N. A. C. Cressie, 1993). The observed value $Z(\boldsymbol{s})$ at location $\boldsymbol{s}$ is modeled as:

$$Z(\boldsymbol{s}) = Y(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}), \qquad \boldsymbol{s} \in D \subset \mathbb{R}^D, \tag{18}$$

where $Y(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \xi(\boldsymbol{s})$ represents a linear mean structure varying over space, and $\varepsilon(\boldsymbol{s})$ is a zero-mean random noise term that is uncorrelated with $Y(\boldsymbol{s})$. The covariance function of the noise is given by $C(\boldsymbol{s}, \boldsymbol{s}') = \text{Cov}(\varepsilon(\boldsymbol{s}), \varepsilon(\boldsymbol{s}'))$, which may correspond to a non-stationary spatial covariance structure.

Traditional Kriging relies on the inversion of a full covariance matrix, leading to computational costs that scale cubically with the number of observations n, thereby creating a significant computational bottleneck as n increases (N. Cressie and Johannesson, 2008). Fixed Rank Kriging (FRK), along with its adaptive extension Adaptive Fixed Rank Kriging (AFRK), was introduced in this context to alleviate the computational burden associated with large-scale spatial data analysis.

2.2.2 Fixed Rank Kriging

To reduce the computational burden of Kriging, N. Cressie and Johannesson (2008) proposed Fixed Rank Kriging (FRK), which represents the random field using a finite set of basis functions, thereby approximating high-dimensional spatial random effects with low-dimensional random coefficients:

$$Y(\boldsymbol{s}) = \mu(\boldsymbol{s}) + \boldsymbol{f}(\boldsymbol{s})^\top \boldsymbol{w} + \xi(\boldsymbol{s}), \tag{19}$$

where $\boldsymbol{f}(\boldsymbol{s}) = (f_1(\boldsymbol{s}), \ldots, f_K(\boldsymbol{s}))^\top$ is a pre-specified K-dimensional vector of basis functions with $K \leq n$, $\boldsymbol{w} \sim N(\boldsymbol{0}, \boldsymbol{M})$ with $\boldsymbol{M}$ is an unknown nonnegative-definite matrix, and $\xi(\boldsymbol{s}) \sim \mathcal{N}(0, \sigma_\xi^2)$ represents fine-scale random noise. The corresponding covariance matrix can then

be expressed as:

$$\text{Cov}(Y(\boldsymbol{s}), Y(\boldsymbol{s}')) = \boldsymbol{f}(\boldsymbol{s})^\top \boldsymbol{M} \boldsymbol{f}(\boldsymbol{s}') + \sigma_\xi^2 \boldsymbol{I}(\boldsymbol{s} = \boldsymbol{s}'). \tag{20}$$

Since the rank of $\boldsymbol{f}\boldsymbol{M}\boldsymbol{f}^\top$ is typically much smaller than the number of observation points n, FRK can significantly reduce the computational cost of covariance matrix inversion, making it particularly suitable for large-scale remote sensing and environmental monitoring data.

2.2.3 Adaptive Fixed Rank Kriging

Building upon FRK, Tzeng and Huang (2018) further proposed Adaptive Fixed Rank Kriging (AFRK). The core idea is to allow the resolution of the basis functions to automatically adapt to the spatial distribution of the data, thereby providing greater flexibility in capturing spatial heterogeneity. This adaptive mechanism enables the model to allocate appropriate spatial resolution across different regions according to the underlying spatial variability.

The basis functions adopted in AFRK are multi-resolution spline basis functions (MRTS), which are constructed from thin-plate splines (TPS). TPS is a commonly used smoothing spline method that produces smooth functions by minimizing the sum of squared errors together with a smoothness penalty term (Wahba and Wendelberger, 1980; Green and Silverman, 1993). Based on TPS, Tzeng and Huang (2018) further constructed an ordered set of basis functions at multiple resolutions via eigen-decomposition, referred to as the MRTS.

To automatically adapt to the spatial distribution of the data and spatial heterogeneity, AFRK selects the number of basis functions according to the magnitude of the corresponding eigenvalues. Only the bases that explain most of the spatial variability are retained, allowing the model to capture the dominant spatial variations using a relatively small number of basis functions while improving computational efficiency.

In AFRK, the MRTS functions are defined as:

$$f_k(\boldsymbol{s}) = \begin{cases} 1, & \text{if } k = 1; \\ x_{k-1}, & \text{if } k = 2, \ldots, d+1; \\ \lambda_{k-d-1}^{-1} \times \left\{ \phi(\boldsymbol{s}) - \boldsymbol{\Phi}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{x} \right\}' \boldsymbol{v}_{k-d-1}, & \text{if } k = d+2, \ldots, n, \end{cases} \tag{21}$$

where f_k is the k-th basis in $\boldsymbol{f}$, $\boldsymbol{X} \in \mathbb{R}^{N\times(d+1)}$ is the design matrix, with each row corresponding to the intercept and coordinates of an observation location $\boldsymbol{s}_i$, and $\boldsymbol{x} = (1, \boldsymbol{s}')' = (1, x_1, \ldots, x_d)'$. The matrix $\boldsymbol{\Phi}$ is defined as

$$J(f) = \boldsymbol{\alpha}'\boldsymbol{\Phi}\boldsymbol{\alpha}, \tag{22}$$

an $n \times n$ matrix with entries $\phi_j(\boldsymbol{s}_i)$, where $\phi(\boldsymbol{s})$ is defined by:

$$\phi_i(\boldsymbol{s}) = \begin{cases} \frac{1}{12}\|\boldsymbol{s} - \boldsymbol{s}_i\|^3, & \text{if } d = 1; \\ \frac{1}{8\pi}\|\boldsymbol{s} - \boldsymbol{s}_i\|^2 \log(\|\boldsymbol{s} - \boldsymbol{s}_i\|), & \text{if } d = 2; \\ -\frac{1}{8}\|\boldsymbol{s} - \boldsymbol{s}_i\|, & \text{if } d = 3; \end{cases} \tag{23}$$

and $\boldsymbol{v}_k$ denotes the k-th row of matrix $\boldsymbol{V}$, where $\boldsymbol{V}$ diag$(\lambda_1, \ldots, \lambda_n)\boldsymbol{V}'$ is the eigendecomposition of $\boldsymbol{Q}\boldsymbol{\Phi}\boldsymbol{Q}$, with $\boldsymbol{Q} = \boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'$. This approach adjusts the basis resolution according to data density, producing higher-resolution bases in densely sampled regions while maintaining smoother structures in sparse regions.

By retaining the computational advantages of low-rank approximation while introducing data-driven multi-resolution bases, AFRK effectively handles irregular sampling and regions with significant local variability. Compared to conventional FRK, AFRK demonstrates superior predictive performance on non-uniform and non-stationary spatial data.

2.3 Spatiotemporal Models

2.3.1 Gaussian Processes

A Gaussian Process (GP) is a non-parametric Bayesian model used to define a distribution over functions that map inputs to outputs (Rasmussen and Williams, 2006). A GP is fully specified by a mean function and a covariance function (also known as a kernel function). Within the Bayesian framework, GP provides both predictive means and uncertainty quantification for predictions. However, exact posterior inference requires inversion or Cholesky decomposition of an $N \times N$ covariance matrix, resulting in a computational complexity of $\mathcal{O}(N^3)$. This makes standard GP models infeasible for large-scale spatio-temporal datasets.

To address this limitation, Hensman, Fusi, and Lawrence (2013) proposed the Stochastic

Variational Gaussian Process (SVGP). This approach introduces a set of $M \ll N$ inducing points to approximate the full posterior distribution. In the implementation framework of Gardner et al. (2021), the variational distribution is parameterized as a multivariate Gaussian with a full covariance matrix. To ensure positive definiteness during optimization, the covariance matrix is parameterized via its mean vector and a lower-triangular matrix, i.e., through Cholesky factorization (Hensman, Matthews, and Ghahramani, 2014).

Within the SVGP framework, the size of the variational distribution is determined by the number of inducing points. Specifically, the variational mean has dimension M, and the variational covariance matrix is of size $M \times M$. By approximating the full posterior using $M \ll N$ inducing points and optimizing the Evidence Lower Bound (ELBO), SVGP reduces the computational complexity to $\mathcal{O}(M^3)$ (Gardner et al., 2021).

2.3.2 Spatio-temporal DeepKriging

Spatio-temporal DeepKriging (STDK) is a recently proposed non-parametric approach that integrates deep learning with spatial statistics for large-scale spatio-temporal interpolation and probabilistic prediction (Nag, Sun, and Reich, 2023). In contrast to traditional Gaussian Processes, which require a pre-specified covariance function and suffer from $\mathcal{O}(N^3)$ computational complexity, STDK learns spatio-temporal dependencies directly from data in a data-driven manner, thereby improving scalability for large datasets.

STDK first embeds spatio-temporal coordinates $(\mathbf{s}, t)$ into a high-dimensional feature space. Let the basis function vector be defined as

$$\boldsymbol{\phi}(\mathbf{s}, t) = [\phi_1(\mathbf{s}, t), \ldots, \phi_K(\mathbf{s}, t)]^\top, \tag{24}$$

where $\{\phi_k(\cdot)\}_{k=1}^K$ are multi-resolution basis functions. Previous studies commonly adopt compactly supported Wendland functions or radial basis functions (RBFs) to capture spatial dependencies across multiple scales (Nag, Sun, and Reich, 2023).

The embedded features are then passed into a deep neural network to model nonlinear mappings, which can be expressed as

$$Z(\mathbf{s}, t) = f_{\boldsymbol{\theta}}\big(\boldsymbol{\phi}(\mathbf{s}, t)\big) + \epsilon, \tag{25}$$

where $f_{\boldsymbol{\theta}}(\cdot)$ denotes a deep neural network parameterized by $\boldsymbol{\theta}$, and ϵ represents a random noise term.

For probabilistic forecasting, STDK adopts a quantile loss function to estimate different quantiles of the conditional distribution, thereby constructing predictive intervals. Compared with mean squared error (MSE)-based approaches, this enables explicit uncertainty quantification.

By combining basis function embeddings with deep neural networks, STDK avoids explicit modeling and decomposition of covariance matrices, resulting in an approximate computational complexity of $\mathcal{O}(N)$. Therefore, it is often used as a strong baseline method for large-scale spatio-temporal data modeling.

3 Methodology

In this study, we propose a spatiotemporal forecasting framework that integrates deep time series modeling with spatial statistical modeling. The objective is to simultaneously exploit temporal and spatial information in order to improve reconstruction quality under missing data scenarios and enhance prediction accuracy for future time points.

The proposed framework utilizes the SSSD^{S4} model to capture the dynamic structure of the data along the temporal dimension, thereby extracting sequential temporal features. During model training, spatial dependence is incorporated through AFEK, allowing the model to account for spatial correlations among different locations.

In the inference stage, the temporal features predicted by SSSD^{S4} together with the spatial coordinates of unobserved locations are provided as inputs to AFRK. By integrating both temporal dynamics and spatial dependence, AFRK estimates the target values at unobserved locations.

The remainder of this section describes the problem formulation, the proposed spatiotemporal modeling framework, and the overall algorithmic procedure of the integrated method.

3.1 Problem Formulation

Consider a spatial domain consisting of a set of locations $\mathcal{S}$, which can be partitioned into the subset of observed locations $\mathcal{S}_{\text{observed}}$ with available measurements and the subset of unobserved locations $\mathcal{S}_{\text{unobserved}}$ with no recorded observations. For each observed location $\boldsymbol{s} \in \mathcal{S}_{\text{observed}}$, the target variable is fully observed over the temporal horizon $t \in \{1, \ldots, T\}$, and the corresponding observations are denoted as $y_t(\boldsymbol{s})$. In contrast, for each unobserved location $\boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$, no historical measurements exist within this time interval. The objective of this study is to construct a predictive model that leverages the historical spatiotemporal features from the observed locations, denoted by $\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}})$, in order to perform temporal extrapolation and spatial estimation of the target variable $\hat{y}_t(\boldsymbol{s}^*)$ at unobserved locations for future time points $t > T$.

The objective can be formulated as learning a mapping function

$$\hat{y}_t(\boldsymbol{s}^*) = f\left(\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}}), \boldsymbol{s}^*\right), \qquad \boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}},\ t > T, \tag{26}$$

where $\boldsymbol{Y}_{1:T}(\mathcal{S}_{\text{observed}})$ denotes the collection of observed sequences from all known locations up to time T, and $\hat{y}_t(\boldsymbol{s}^*)$ represents the estimated value at an unknown location $\boldsymbol{s}^*$ for a future time step t.

3.2 Spatiotemporal Modeling Approach

To achieve the aforementioned mapping objective, this study first employs the SSSD^{S4} model to capture the dynamic dependencies along the temporal dimension, and subsequently integrates AFRK to fuse the extracted temporal features with spatial coordinates, enabling spatial interpolation at unobserved locations as well as future forecasting.

3.2.1 Temporal Modeling Based on SSSD^{S4}

To effectively capture the temporal dependencies inherent in the data, the SSSD^{S4} model is adopted as the temporal feature extractor. Its objective is to learn latent temporal features that simultaneously encode long-term dependencies and local dynamics from incomplete or noisy time series, leveraging the combination of the diffusion model and the S4 layers.

Using the observations at the observed locations $\mathcal{S}_{\text{observed}}$ as the training basis, the data are first standardized as

$$\tilde{y}_t(\boldsymbol{s}) = \frac{y_t(\boldsymbol{s}) - \mu(\boldsymbol{s})}{\sigma(\boldsymbol{s})}, \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{27}$$

where $\mu(\boldsymbol{s})$ and $\sigma(\boldsymbol{s})$ denote the mean and standard deviation of the time series at location $\boldsymbol{s}$, defined as

$$\mu(\boldsymbol{s}) = \frac{1}{T}\sum_{t=1}^{T} y_t(\boldsymbol{s}), \tag{28}$$

$$\sigma(\boldsymbol{s}) = \sqrt{\frac{1}{T}\sum_{t=1}^{T} \left(y_t(\boldsymbol{s}) - \mu(\boldsymbol{s})\right)^2}. \tag{29}$$

The standardized input sequences $\tilde{y}_t(\boldsymbol{s})$ are then fed into the SSSD$^{\text{S4}}$ model to learn the temporal dependency structure.

According to Ho, Jain, and Abbeel (2020), during the parameter optimization phase, a diffusion model approximates the true Gaussian noise $\boldsymbol{\epsilon}$ added in the forward diffusion process by the noise prediction term $\boldsymbol{\epsilon}_\theta$ output by the neural network, and updates the model parameters by minimizing the objective

$$\mathcal{L} = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\|^2. \tag{30}$$

In this study, during the reverse diffusion process, $\boldsymbol{\epsilon}_\theta$ is further adjusted based on its spatial adjacency structure. Specifically, prior to the gradient update, AFRK is applied to the predictions of $\boldsymbol{\epsilon}_\theta$ at the same spatial locations to enforce spatial consistency and smoothness, thereby enhancing the ability of the reverse diffusion process to capture spatial structures accurately.

$$\hat{\boldsymbol{\epsilon}}_\theta = \text{AFRK}(\boldsymbol{\epsilon}_\theta) = \mu_{\boldsymbol{\epsilon}_\theta} + \boldsymbol{f}_{\boldsymbol{\epsilon}_\theta}^\top \hat{\boldsymbol{w}}_{\boldsymbol{\epsilon}_\theta} + \hat{\xi}_{\boldsymbol{\epsilon}_\theta}. \tag{31}$$

The predictions of the SSSD$^{\text{S4}}$ model adjusted via AFRK can be expressed as

$$\hat{\tilde{y}}_t(\boldsymbol{s}) = g_{\theta_{\text{S4+AFRK}}}\big(\tilde{y}_{1:t}(\boldsymbol{s})\big), \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{32}$$

where $g_{\theta_{\text{S4+AFRK}}}(\cdot)$ denotes the SSSD model employing the S4 architecture and adjusted via

AFRK, and $\hat{\tilde{y}}_t(\boldsymbol{s})$ represents the model predictions on the standardized scale. Since the model is trained and evaluated on standardized data, the final predictions need to be transformed back to the original data scale via the inverse standardization:

$$\hat{y}_t(\boldsymbol{s}) = \sigma(\boldsymbol{s})\hat{\tilde{y}}_t(\boldsymbol{s}) + \mu(\boldsymbol{s}), \tag{33}$$

where $\mu(\boldsymbol{s})$ and $\sigma(\boldsymbol{s})$ denote the mean and standard deviation of the time series at location $\boldsymbol{s}$, respectively. This transformation restores the model outputs to the original measurement scale, facilitating subsequent analyses and spatial interpolation applications.

The SSSD model is trained using the MSE as the loss function, which iteratively updates the parameters $\theta_{\text{S4+AFRK}}$ to ensure that the extracted temporal features are stable and predictive. For tasks such as missing value imputation and multi-step future forecasting, the temporal representations learned by SSSD$^{\text{S4}}$ provide discriminative sequential embeddings, while the spatial regularization imposed by AFRK during training further enhances the overall accuracy and reliability of spatiotemporal estimation.

3.2.2 Spatial Modeling Based on AFRK

Within the proposed integrated framework, AFRK treats the temporal predictions at observed locations generated by SSSD$^{\text{S4}}$, denoted as $\hat{y}_t(\boldsymbol{s})$ for $\boldsymbol{s} \in \mathcal{S}_{\text{observed}}$, as input information along the temporal dimension. To remove scale differences across locations, the predictions at a given time point t are first standardized along the spatial dimension:

$$\tilde{y}_t(\boldsymbol{s}) = \frac{\hat{y}_t(\boldsymbol{s}) - \mu_t}{\sigma_t}, \qquad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}, \tag{34}$$

where μ_t and σ_t denote the sample mean and sample standard deviation at time t over all observed locations $\mathcal{S}_{\text{observed}}$.

The standardized values $\tilde{y}_t(\boldsymbol{s})$ are then used as input to AFRK to perform conditional estimation at unobserved locations $\boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$. On the standardized scale, the predictions at unobserved locations can be expressed as

$$\hat{\tilde{y}}_t(\boldsymbol{s}^*) = \mu(\boldsymbol{s}^*) + \boldsymbol{f}(\boldsymbol{s}^*)^\top \hat{\boldsymbol{w}}_t + \hat{\xi}_t(\boldsymbol{s}^*), \tag{35}$$

where $\hat{\boldsymbol{w}}_t$ and $\hat{\xi}_t(\boldsymbol{s}^*)$ are the spatial conditional distribution parameters estimated by AFRK

based on the observed location information $\tilde{y}_t(\boldsymbol{s})$.

Finally, the spatial predictions are transformed back to the original measurement scale via inverse standardization:

$$\hat{y}_t(\boldsymbol{s}^*) = \sigma_t \hat{\tilde{y}}_t(\boldsymbol{s}^*) + \mu_t. \tag{36}$$

thereby completing the reconstruction and forecasting of the global spatiotemporal field.

3.3 Model Training and Inference Algorithms

This section summarizes the operational workflow of the aforementioned spatiotemporal integration framework. Algorithm 1 describes the training procedure of SSSD^{S4} combined with AFRK, while Algorithm 2 presents the complete steps for model inference.

Algorithm 1 SSSD-AFRK Training Stage.

Require: $y_t(\boldsymbol{s}), \boldsymbol{s} \in \mathcal{S}_{\text{observed}}$; Max steps T; Learning rate η

$\tilde{y}_0(\boldsymbol{s}) = \left(y_t(\boldsymbol{s}) - \mu(\boldsymbol{s})\right)/\sigma(\boldsymbol{s})$
repeat
  $t \sim \text{Uniform}(\{1, \ldots, T\}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
  $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\tilde{y}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$
  $\mathbf{h} = \text{S4-Layer}(\boldsymbol{x}_t, t)$
  $\boldsymbol{\epsilon}_\theta = \mathbf{W}_o\mathbf{h} + \mathbf{b}_o$
  $\hat{\boldsymbol{\epsilon}}_\theta = \text{AFRK}(\boldsymbol{\epsilon}_\theta) = \mu_{\boldsymbol{\epsilon}_\theta} + \boldsymbol{f}_{\boldsymbol{\epsilon}_\theta}^\top \hat{\boldsymbol{w}}_{\boldsymbol{\epsilon}_\theta} + \hat{\xi}_{\boldsymbol{\epsilon}_\theta}$
  $\mathcal{L} = \|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}_\theta\|^2$
  $\theta \leftarrow \theta - \eta\nabla_\theta\mathcal{L}$
until convergence

Here, $\boldsymbol{h}$ denotes the latent temporal features obtained after processing through the S4 layer, and $\mathbf{W}_o$ and $\mathbf{b}_o$ represent the learnable weight matrix and bias vector of the output projection layer, respectively.

Algorithm 2 SSSD-AFRK Inference Stage.

Require: Known samples $y_t(\boldsymbol{s})$; Unknown locations $\mathcal{S}_{\text{unobserved}} = \{\boldsymbol{s}_1^*, \ldots, \boldsymbol{s}_m^*\}$; Model θ

{Stage 1: Temporal Sequence Generation (at $\mathcal{S}_{\text{observed}}$)}

$\tilde{y}_0(\boldsymbol{s}) = (y_t(\boldsymbol{s}) - \mu(\boldsymbol{s}))/\sigma(\boldsymbol{s})$

$\boldsymbol{x}_T \sim \mathcal{N}(0, \boldsymbol{I})$

for $t = T, \ldots, 1$ **do**

  $\mathbf{h} = \text{S4-Layer}(\boldsymbol{x}_t, t)$

  $\boldsymbol{\epsilon}_\theta = \mathbf{W}_o\mathbf{h} + \mathbf{b}_o$

  $\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta\right) + \sigma_t\mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(0, \boldsymbol{I})$

end for

$\hat{y}_t(\boldsymbol{s}) = \sigma(\boldsymbol{s})\boldsymbol{x}_0 + \mu(\boldsymbol{s}), \quad \boldsymbol{s} \in \mathcal{S}_{\text{observed}}$

{Stage 2: Spatial Interpolation via AFRK (at $\mathcal{S}_{\text{unobserved}}$)}

$\tilde{y}_t(\boldsymbol{s}) = (\hat{y}_t(\boldsymbol{s}) - \mu_t)/\sigma_t$

$\hat{\tilde{y}}_t(\boldsymbol{s}^*) = \text{AFRK}(\tilde{y}_t(\boldsymbol{s}) \mid \boldsymbol{s}^*), \quad \forall \boldsymbol{s}^* \in \mathcal{S}_{\text{unobserved}}$

$\hat{y}_t(\boldsymbol{s}^*) = \sigma_t\hat{\tilde{y}}_t(\boldsymbol{s}^*) + \mu_t$

return Full field $\hat{y}_t(\mathcal{S}_{\text{observed}} \cup \mathcal{S}_{\text{unobserved}})$

4 Experiments

This chapter presents the experimental design and setup for the spatiotemporal integration framework, aiming to evaluate its effectiveness and feasibility in forecasting future spatiotemporal values at unobserved locations. The experiments are conducted using two primary datasets, with detailed descriptions of model configurations, training and inference procedures, as well as the characteristics and partitioning of the datasets, in order to demonstrate the framework's performance under different data conditions.

4.1 Datasets

This study adopts two representative meteorological datasets, which respectively cover in-situ ground station observations and global reanalysis data, as well as one synthetic dataset. These datasets are used to evaluate the model's applicability and generalization capability under different spatio-temporal dynamics.

4.1.1 Weather2K

Weather2K is a recently proposed multivariate ground-based observation benchmark dataset, comprising measurements from thousands of meteorological stations across China, with a temporal resolution of 3 hours and covering near-surface meteorological variables such as air temperature, air pressure, humidity, and wind speed (Zhu et al., 2023). The open-source version, Weather2K-R, contains 1,866 stations and 13,632 consecutive time steps, featuring complete and regularly sampled time series without missing values. The dataset also provides constant location information (latitude, longitude, and elevation), which facilitates spatiotemporal modeling.

The Weather2K dataset exhibits heterogeneous spatial distribution and uneven station density, encompassing various geographical environments and climatic conditions, including urban areas, plains, and mountainous regions, thus providing rich spatiotemporal variation signals. These characteristics make Weather2K suitable for spatiotemporal interpolation, short-term sequence forecasting, and evaluations of model generalization and robustness.

In this study, observations from Weather2K-R between 00:00 on July 1, 2021, and 21:00 on August 31, 2021, are selected. A total of 200 observation sites are used in the experiments, and Air Temperature is selected as the target variable. Detailed descriptions of the variables are provided in Appendix A.

4.1.2 MERRA-2

MERRA-2 (Modern-Era Retrospective Analysis for Research and Applications, Version 2) is a global atmospheric reanalysis dataset provided by the NASA Goddard Earth Sciences Data and Information Services Center (GES DISC). It reconstructs the state of the global atmosphere since 1980 through data assimilation techniques that integrate numerical weather prediction models with multi-source observations, primarily from satellites (GMAO, 2015). In this study, we use the hourly, single-level, instantaneous assimilation diagnostic product M2I1NXASM (Version 5.12.4) as the analysis dataset.

MERRA-2 is archived and managed by the Distributed Active Archive Center (DAAC) at NASA Goddard Space Flight Center, providing globally consistent reanalysis data. Its data assimilation process integrates satellite, ground-based, and remote sensing observations, with

significant improvements over its predecessor MERRA in representing physical processes such as aerosols, radiative balance, and the hydrological cycle. It has been widely applied in climate trend analysis, extreme event studies, energy balance diagnostics, and numerical model evaluation. The M2I1NXASM product offers hourly instantaneous data with high resolution and representative climate signals, making it suitable for short-term spatiotemporal forecasting and statistical feature analysis.

This study selects data from the M2I1NXASM (Version 5.12.4) product of the MERRA-2 dataset, spanning from 00:00 on December 11, 2023 to 23:00 on December 31, 2023. The Surface Skin Temperature variable is used as the primary data source. Detailed descriptions of the variables are provided in Appendix B. The experimental domain is defined as a rectangular spatial region bounded by $26.0°$ to $48.0°$N latitude and $70.0°$ to $123.0°$W longitude, within which 200 observation points are selected.

4.1.3 2b-8

The 2b-8 dataset originates from The Second Competition on Spatial Statistics, which aims to investigate issues related to prediction accuracy and computational efficiency in large-scale spatial and spatio-temporal data analysis (Abdulah, Alamri, Nag, et al., 2022). The dataset provides a series of carefully sampled and simulated spatial observations, specifically designed to evaluate the interpolation performance of complex spatio-temporal models, particularly under high-dimensional, large-scale settings with intricate spatial dependence structures.

In this study, 200 observation points are selected from the 2b-8 dataset (Abdulah, Alamri, Ltaief, et al., 2022) as the data source. A detailed description of the dataset is provided in Appendix C.

4.2 Environment and Computational Resources

To ensure the feasibility and reliability of training and inference experiments of the proposed spatiotemporal integration framework on large-scale datasets, a unified computational environment was established. Multiple existing software packages were integrated to support model development, training, and evaluation.

The SSSD model has been implemented in Python by its original authors (AI4HealthUOL, 2023), which is capable of effectively capturing long-range dependency structures in time-series data. AFRK was implemented in the R programming language by Wen-Ting Wang and released as the `autoFRK` package (Tzeng, Huang, Wang, Nychka, et al., 2021), providing stable and scalable spatial interpolation capabilities. To integrate these functionalities, this study reimplemented and encapsulated `autoFRK` as a Python package (Tzeng, Huang, Wang, and Hsu, 2025). The algorithm was further integrated with the original SSSD implementation, enabling a complete spatiotemporal modeling workflow within the PyTorch framework in Python.

All experiments, including model training, validation, and inference, were conducted on the Taiwania 2 high-performance computing platform (NCHC, 2018). Taiwania 2 provides high-performance GPU computing resources, along with large-capacity memory and high-speed storage systems. These resources enable efficient processing of high-resolution datasets and long time series while ensuring computational stability and reproducibility of the experimental results.

4.3 Experimental Design

To clearly present the configurations of SSSD^{S4} and `autoFRK`, the following sections summarize the hyperparameters for training and inference as shown in Table 1.

Table 1: Model Hyperparameter Settings.

Model	**Hyperparameter**	**Value**
Training Configuration		
	Batch size	40
	Learning rate	0.001
	Only generate missing	True
	Masking	Forecast
	Missing k	—
SSSD^{S4}		
WaveNet	Input channels	1

Model	Hyperparameter	Value
	Output channels	1
	Residual layers	32
	Residual channels	64
	Skip channels	64
Diffusion step embedding	Input dimension	64
	Hidden dimension	128
	Output dimension	128
S4	Max sequence length	—
	State dimension	128
	Dropout	0.2
	Bidirectional	True
	Layer normalization	True
Diffusion	Diffusion steps (T)	100
	β_0	0.0001
	β_T	0.05
autoFRK		
	Method	Fast
	Thin-plate spline method	Rectangular

Table 2 summarizes the experimental parameter settings and dataset configurations used to compare the performance of different models. The input sequence is denoted as $\boldsymbol{X} \in \mathbb{R}^{N \times T \times C}$, where N represents the number of spatial locations, T denotes the temporal sequence length, and C corresponds to the number of input variables or channels. In the dataset configuration, the temporal split between training and testing is set to 0.9 and 0.1, respectively, while the spatial split between observed and unobserved locations is 0.8 and 0.2.

The experimental design further investigates whether integrating AFRK during the training stage improves model performance. Specifically, we compare the baseline SSSD^{S4} model with the AFRK-enhanced variant SSSD$^{S4+AFRK}$ to assess whether AFRK strengthens spatial dependency modeling and improves the imputation of unobserved locations during inference.

In addition, this study includes several representative baselines, namely TFT, VAR, SVGP, and STDK, to provide a comprehensive comparison across different model architectures and learning paradigms for spatio-temporal forecasting tasks.

Table 2: Experimental parameter settings and dataset configurations (Weather2K / MERRA-2 / 2b-8).

Parameter	**Value**
Number of iterations	500
Observed locations	160
Unobserved locations	40
Sequence length	448 / 456 / 90
Missing sequence length k	48 / 48 / 10

The spatial distribution of observed and unobserved stations in the datasets is illustrated in Figure 3, Figure 4, and Figure 5. Among the sampled locations, blue circles represent observed stations, while green triangles denote unobserved stations.

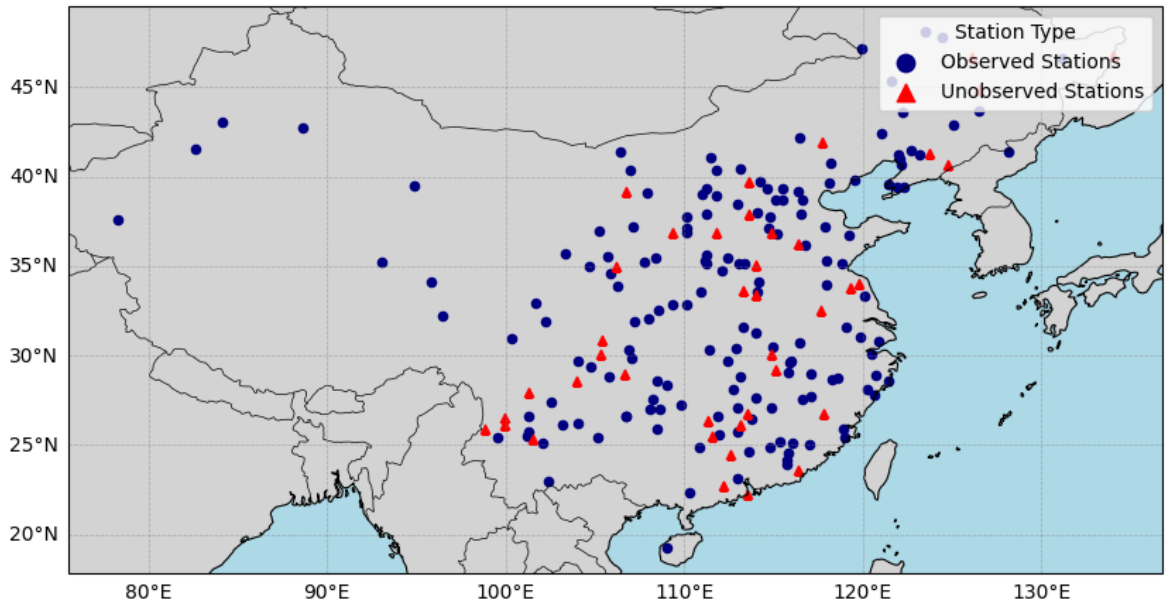

Figure 3: Spatial distribution of Weather2K 3-Hourly observation stations.

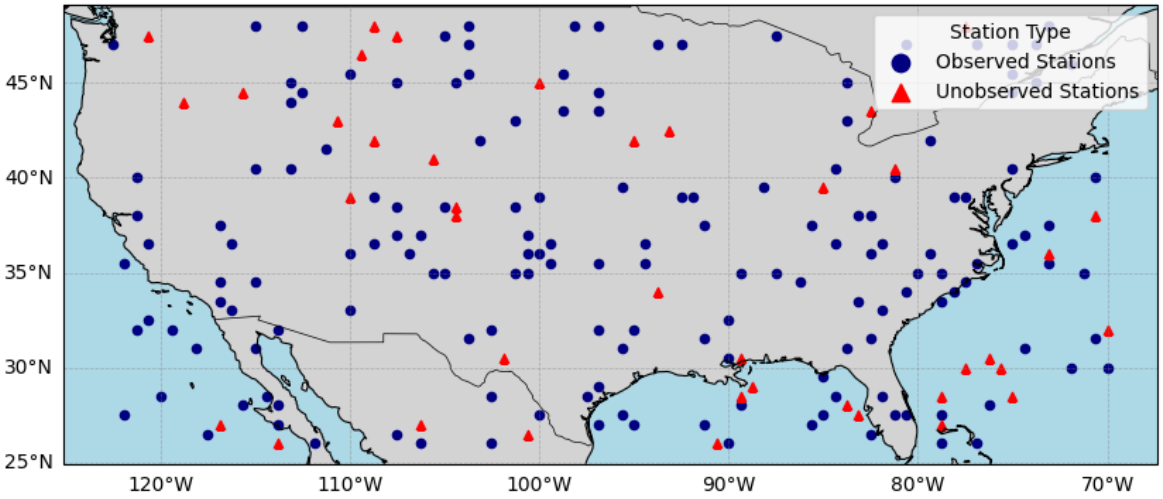

Figure 4: Spatial distribution of MERRA-2 Hourly observation stations.

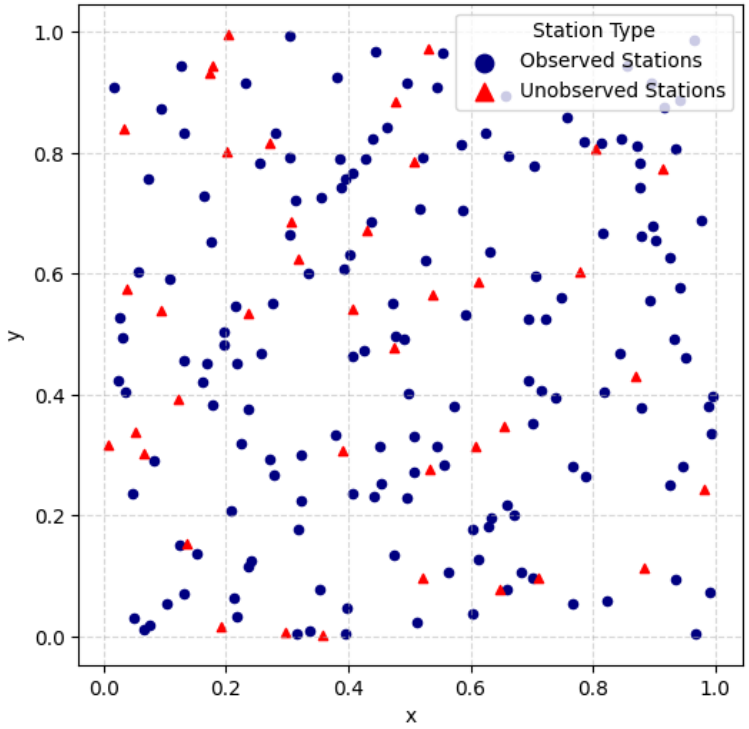

Figure 5: Spatial distribution of 2b-8 observation stations.

5 Experimental Results

In this study, the Mean Squared Prediction Error (MSPE) is adopted as the primary evaluation metric to quantify predictive accuracy at unobserved locations and future time steps. Table 3 reports the average MSPE over 30 independent runs for the integrated spatio-temporal framework $\text{SSSD}^{\text{S4+AFRK}}$, as well as for the baseline model SSSD^{S4} with AFRK applied only during the inference stage for spatial imputation. The table further presents the number of MRTS basis functions selected by AFRK during inference.

In addition, several baseline models are included for comparison. Since some of these models are designed solely for temporal forecasting, AFRK is additionally employed to perform imputation at unobserved spatial locations, enabling a consistent evaluation across different model architectures in spatio-temporal prediction tasks.

Table 3: MSPE and number of MRTS basis functions selected during inference across different models, datasets and prediction settings.

Setting	$\text{SSSD}^{\text{S4+AFRK}}$	SSSD^{S4}	TFT	VAR	SVGP	STDK
Weather2K						
Unobserved & Future	**19.1587**	19.2287	22.2616	30.1737	31.1692	30.5259
Unobserved & Past	4.0865	**4.0800**	4.0913	4.0967	20.5138	23.0728
Unobserved & Future	**15.0609**	15.0880	19.1520	23.4325	29.7829	38.8186
# of MRTS Basis	123	123	123	123	—	—
MERRA-2						
Unobserved & Future	**10.0391**	10.2388	12.8345	235.2112	29663.98	178.5403
Unobserved & Past	6.7505	**6.7384**	6.7403	6.7460	18013.08	174.8610
Unobserved & Future	**7.9678**	8.1248	13.8354	331.8714	27667.48	112.6103
# of MRTS Basis	123	123	123	123	—	—
2b-8						
Unobserved & Future	0.8761	0.8344	0.9071	1.0477	0.9153	**0.8253**
Unobserved & Past	0.7835	**0.7834**	0.7843	0.7846	0.8601	0.8809
Unobserved & Future	1.1884	1.0674	1.6772	2.4342	0.9362	**0.9093**
# of MRTS Basis	23	23	23	23	—	—

Table 3 shows that the two configurations, SSSD$^{S4+AFRK}$ and SSSD^{S4}, exhibit consistent performance differences across datasets. In particular, the model incorporating AFRK achieves lower MSPE in most future prediction tasks at unobserved locations, indicating that AFRK provides effective compensation for spatial structure and enhances predictive performance under spatial extrapolation settings.

A dataset-wise analysis further reveals that, in the Weather2K dataset, SSSD$^{S4+AFRK}$ achieves the most pronounced improvement in the future prediction task at unobserved locations. This suggests that AFRK is particularly effective in enhancing predictive accuracy under conditions of high spatio-temporal variability.

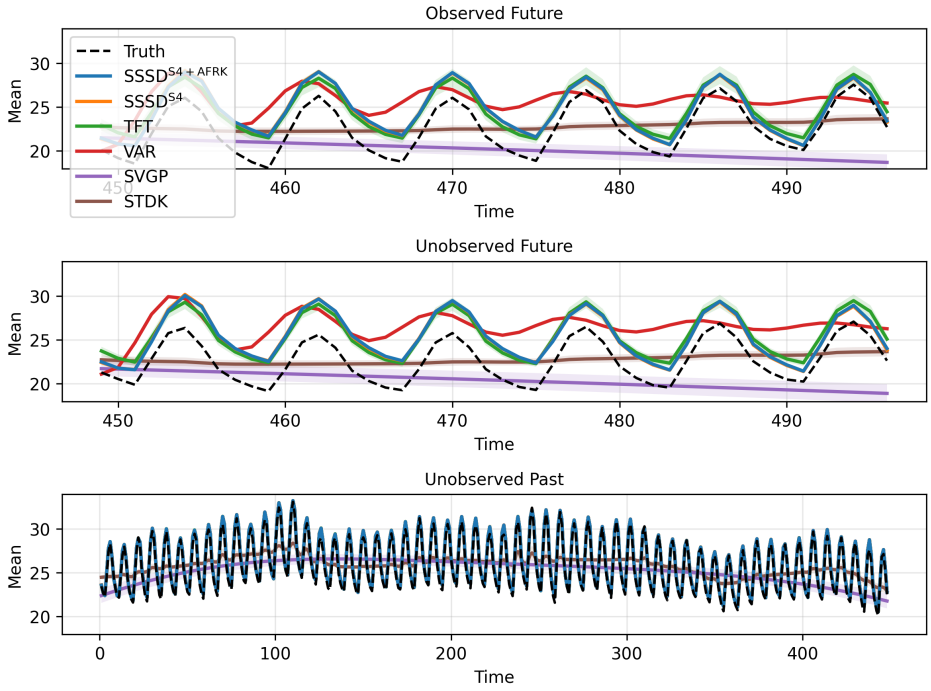

Figure 6: Comparison of predicted mean values under different forecasting settings in the Weather2K dataset.

Figure 6 and Figure 7 present a comparison of predicted mean trajectories and MSPE across three forecasting settings: future prediction at observed locations, future prediction at unobserved locations, and past reconstruction at unobserved locations. The figures also include the corresponding 95% confidence intervals for each model.

The results indicate that, in both future and past prediction tasks at unobserved locations, SSSD$^{S4+AFRK}$ consistently achieves predictions that are closer to the ground truth and yields

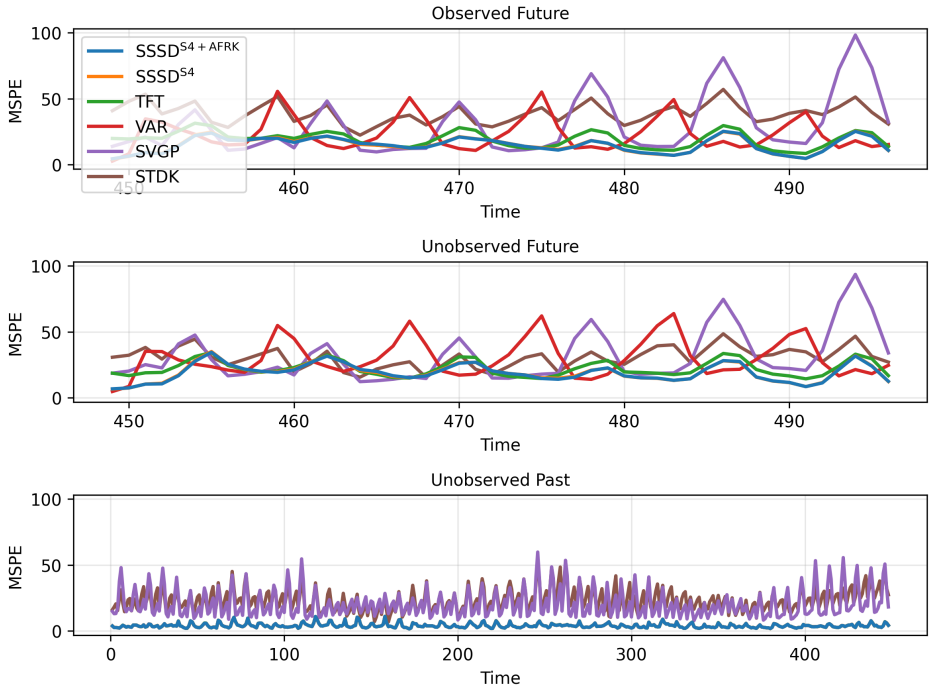

Figure 7: Comparison of MSPE under different forecasting settings in the Weather2K dataset.

lower MSPE compared with baseline models such as TFT, VAR, SVGP, and STDK. This demonstrates its superior capability in capturing complex spatio-temporal dynamics. In contrast, for prediction tasks at observed locations, SSSD^{S4} and SSSD$^{S4+AFRK}$ exhibit highly similar performance, suggesting that the primary contribution of AFRK lies in enhancing model expressiveness for unobserved regions through improved spatial generalization and compensation.

In the MERRA-2 dataset, SSSD$^{S4+AFRK}$ also achieves lower MSPE in future prediction at unobserved locations. Moreover, the magnitude of improvement is more stable compared to that observed in Weather2K, indicating that AFRK maintains consistent spatial modeling capability even under large-scale and high-dimensional climate data. In addition, for observed-location prediction tasks, SSSD$^{S4+AFRK}$ also shows a reduction in error relative to SSSD^{S4}, suggesting that AFRK not only improves inference in unobserved regions but also contributes to performance gains in observed regions by enhancing the learned spatial structure.

As shown in Figure 8 and Figure 9, the SVGP model, which exhibits relatively weaker performance in Table 3, is excluded for clearer comparison. It can be observed from Figure 8 that SSSD^{S4}, SSSD$^{S4+AFRK}$, and TFT all produce predictions that closely follow the ground

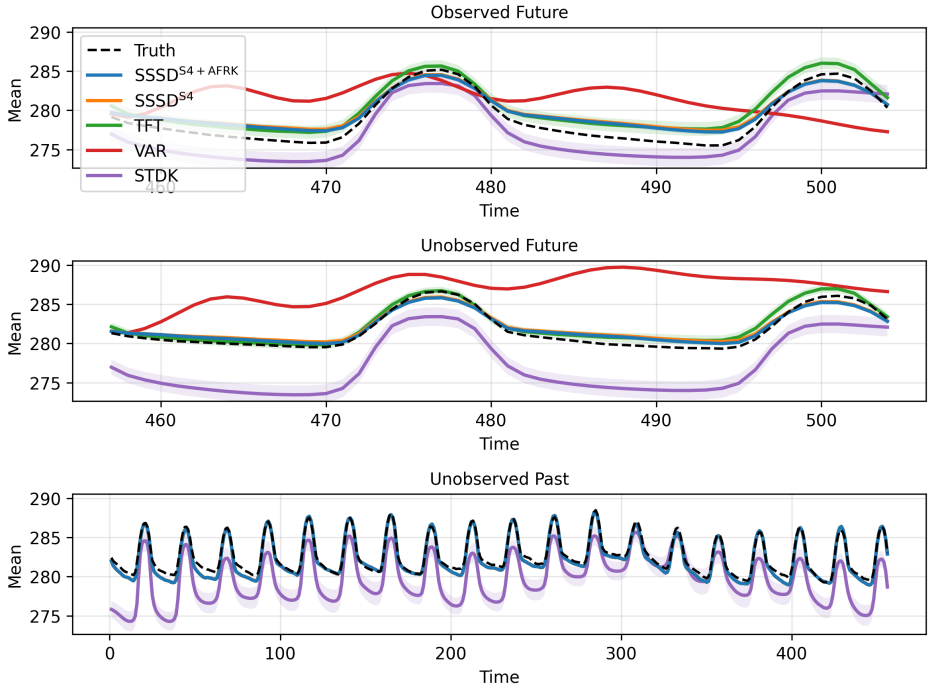

Figure 8: Comparison of predicted mean values under different forecasting settings in the MERRA-2 dataset.

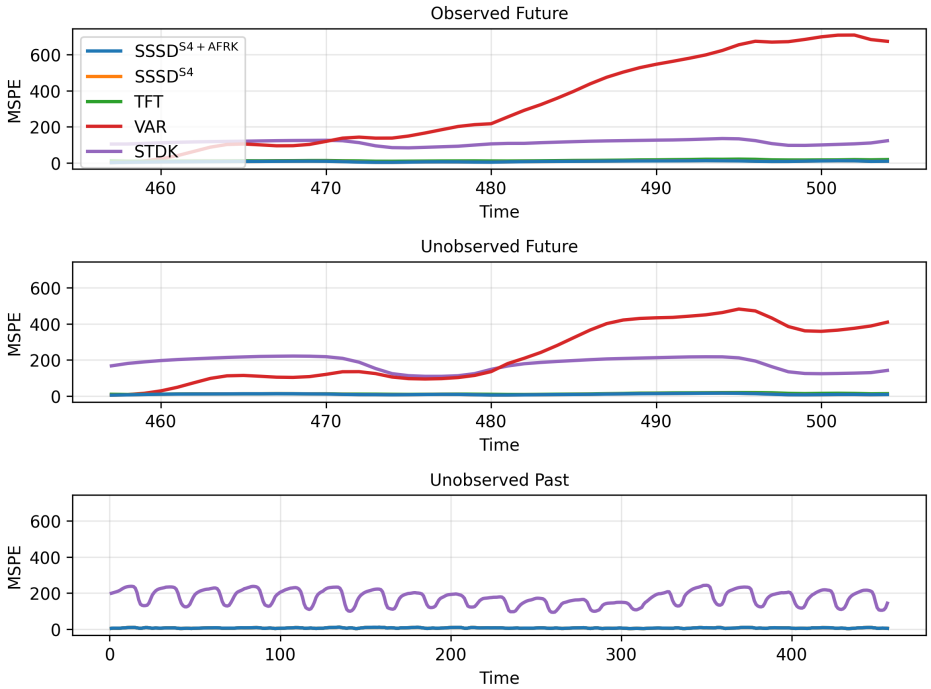

Figure 9: Comparison of MSPE under different forecasting settings in the MERRA-2 dataset.

truth after spatial interpolation via AFRK across different forecasting scenarios. Furthermore, in the latter portion of the future prediction horizon, $\text{SSSD}^{\text{S4+AFRK}}$ yields predicted mean values that are closer to the ground truth than those of TFT, indicating that $\text{SSSD}^{\text{S4+AFRK}}$ has superior capability in capturing long-range spatio-temporal dependencies.

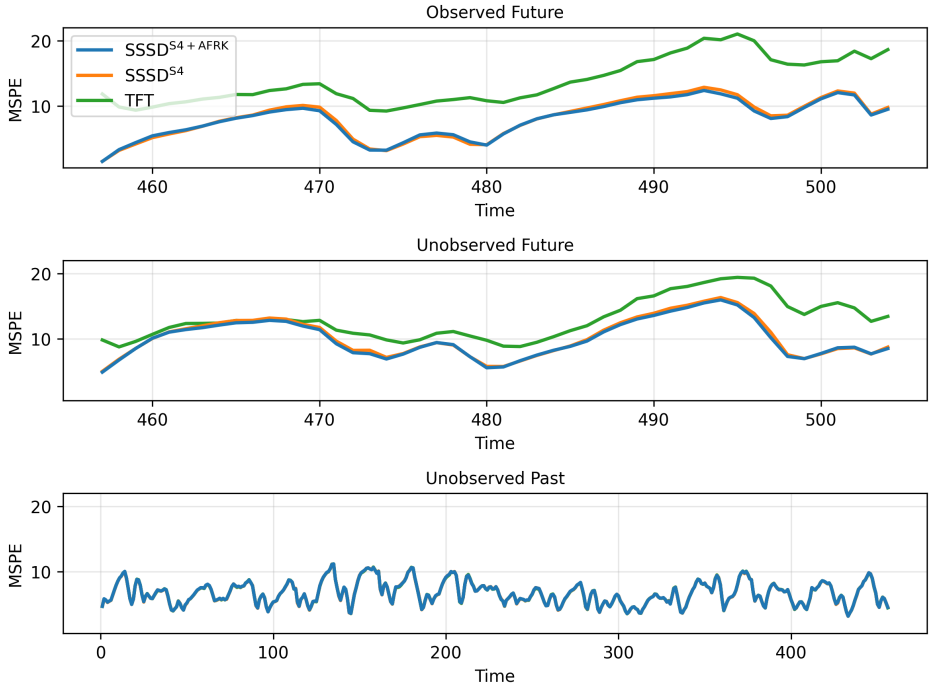

Figure 10: Comparison of MSPE among SSSD^{S4}, $\text{SSSD}^{\text{S4+AFRK}}$, and TFT under different forecasting settings in the MERRA-2 dataset.

As shown in Figure 10, $\text{SSSD}^{\text{S4+AFRK}}$ achieves lower MSPE in the future prediction task at unobserved locations. In addition, its error trajectory is generally more stable and consistently lower than those of both SSSD^{S4} and TFT. This indicates that the multi-scale spatial basis constructed by AFRK is effective in assisting the main model in capturing long-range spatio-temporal dependencies, particularly in high-dimensional climate datasets.

For the 2b-8 synthetic dataset, due to its relatively smooth spatial variability, the overall performance across different models is comparable. In the future prediction task at unobserved locations, STDK achieves the best MSPE, suggesting that carefully designed deep spatio-temporal convolutional architectures can generalize effectively under highly regular and low-variability conditions. In contrast, the performance gap between $\text{SSSD}^{\text{S4+AFRK}}$ and SSSD^{S4} is marginal, indicating that the contribution of AFRK is limited in this dataset.

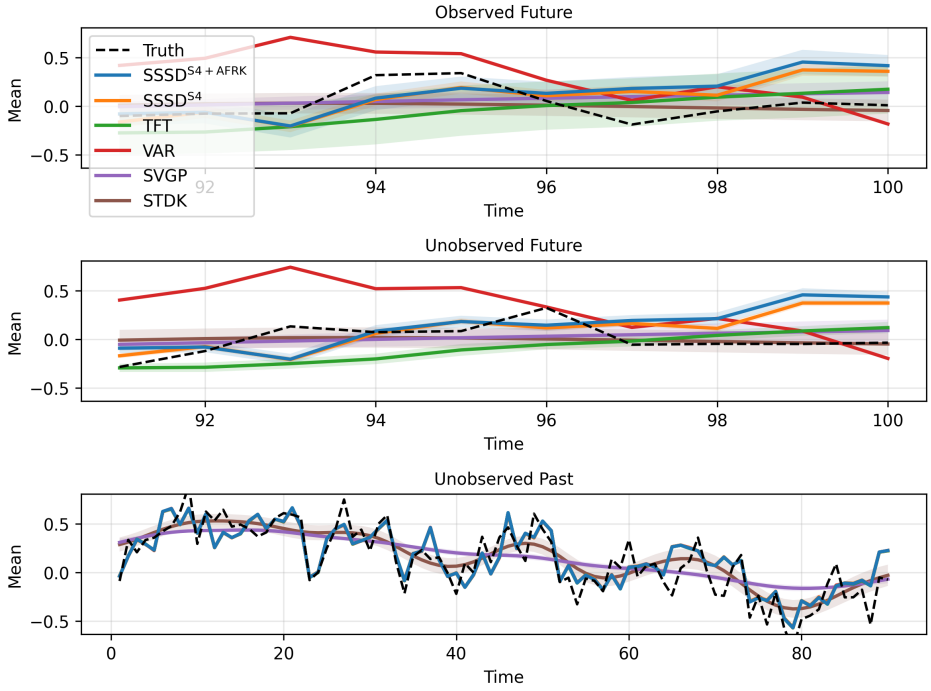

Figure 11: Comparison of predicted mean values under different forecasting settings in the 2b-8 dataset.

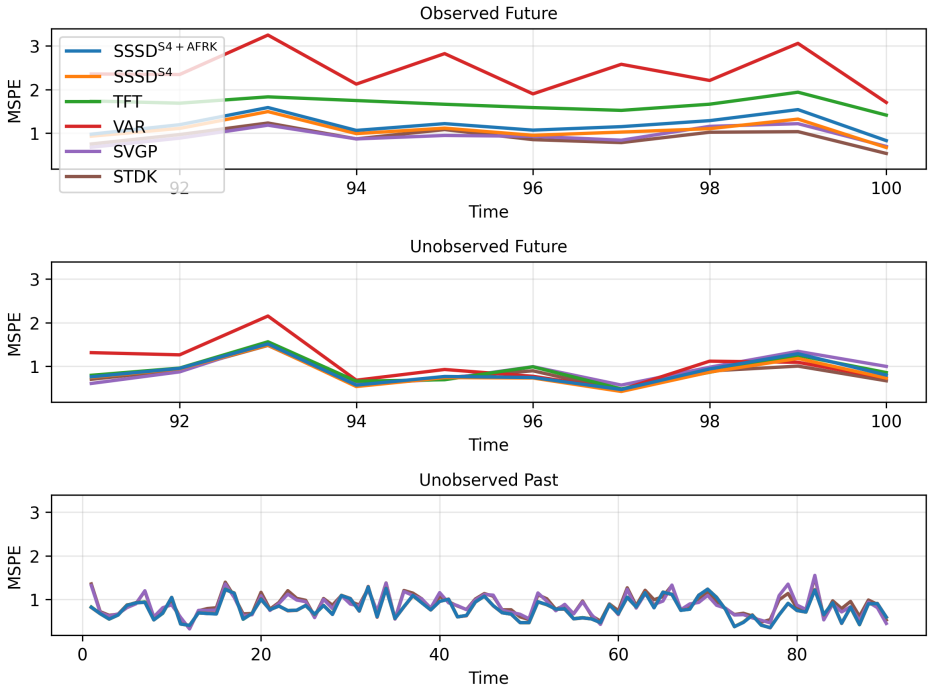

Figure 12: Comparison of MSPE under different forecasting settings in the 2b-8 dataset.

Under synthetic data with relatively simple and regular spatial structures, AFRK does not exhibit a significant advantage; however, it still achieves performance comparable to other spatial modeling approaches in terms of predictive accuracy.

Among the baseline models, TFT demonstrates competitive performance in purely temporal forecasting tasks and exhibits relatively stable error trajectories. The VAR model provides reasonable predictions on certain datasets (e.g., Weather2K), but its performance degrades substantially under highly spatio-temporally varying conditions such as MERRA-2, where the prediction error increases significantly. Although STDK and SVGP are theoretically capable of modeling spatial dependencies, they show limited stability in high-dimensional spatio-temporal settings. In particular, SVGP produces overly smooth predictive curves and substantially inflated errors in both Weather2K and MERRA-2, indicating that its kernel-based approximation struggles to converge effectively under large-scale data regimes.

Overall, these results demonstrate that the proposed $\text{SSSD}^{\text{S4+AFRK}}$ spatio-temporal framework more accurately captures complex spatial dependencies in real-world data compared to models that consider only temporal or spatial structures independently. Moreover, it maintains consistently low prediction error across datasets with varying distributions and degrees of variability.

6 Conclusion

This study proposes an integrated spatio-temporal forecasting framework, $\text{SSSD}^{\text{S4+AFRK}}$, for time series data with geospatial characteristics. By combining a deep state-space model with spatial basis functions, the proposed method effectively improves prediction performance at unobserved locations. Experimental results demonstrate that incorporating AFRK significantly reduces the MSPE in spatial extrapolation tasks, particularly for future prediction at unobserved locations.

In addition, consistent improvements are also observed in future prediction at observed locations and past reconstruction at unobserved locations, indicating that AFRK not only enhances the modeling of missing spatial information but also strengthens the overall predictive capability of the main model. Compared with baseline methods, the proposed framework exhibits superior and more stable performance under high-dimensional and highly variable datasets, such as Weather2K and MERRA-2, highlighting the necessity of joint spatio-temporal modeling.

Future work may further extend the design of AFRK or incorporate additional spatio-temporal features to further improve predictive performance in complex environments.

References

Abdulah, Sameh, Faten Alamri, Hatem Ltaief, et al. (2022). *Data for "The Second Competition on Spatial Statistics for Large Datasets"*. KAUST Research Repository. DOI: 10.25781/KAUST-4ADYZ. URL: https://doi.org/10.25781/KAUST-4ADYZ.

Abdulah, Sameh, Faten Alamri, Pratik Nag, et al. (2022). *The Second Competition on Spatial Statistics for Large Datasets*. arXiv: 2211.03119 [stat.OT]. URL: https://arxiv.org/abs/2211.03119.

AI4HealthUOL (2023). *SSSD: Official Implementation of Diffusion-based Time Series Imputation and Forecasting with Structured State Space Models*. https://github.com/AI4HealthUOL/SSSD. Accessed: 2025-11-28.

Alcaraz, Juan Miguel Lopez and Nils Strodthoff (2023). *Diffusion-based Time Series Imputation and Forecasting with Structured State Space Models*. arXiv: 2208.09399 [cs.LG]. URL: https://arxiv.org/abs/2208.09399.

Bengio, Y., P. Simard, and P. Frasconi (1994). "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2, pp. 157–166. DOI: 10.1109/72.279181.

Box, George E. P. and Gwilym M. Jenkins (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day series in time series analysis and digital processing. Holden-Day. ISBN: 9780816211043. URL: https://books.google.com.tw/books?id=1WVHAAAAMAAJ.

Cressie, Noel and Gardar Johannesson (2008). "Fixed rank kriging for very large spatial data sets". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70.1, pp. 209–226. DOI: https://doi.org/10.1111/j.1467-9868.2007.00633.x. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2007.00633.x. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00633.x.

Cressie, Noel and Christopher K. Wikle (2011). *Statistics for Spatio-Temporal Data*. CourseSmart Series. Wiley. ISBN: 9780471692744. URL: https://books.google.com.tw/books?id=-kOC6D0DiNYC.

Cressie, Noel A. C. (1993). *Statistics for Spatial Data*. John Wiley & Sons, Inc. DOI: 10.1002/9781119115151. URL: https://doi.org/10.1002/9781119115151.

Decorte, Thomas et al. (2024). “Missing Value Imputation of Wireless Sensor Data for Environmental Monitoring”. In: *Sensors* 24.8. ISSN: 1424-8220. DOI: 10.3390/s24082416. URL: https://www.mdpi.com/1424-8220/24/8/2416.

Gardner, Jacob R. et al. (2021). *GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration*. arXiv: 1809.11165 [cs.LG]. URL: https://arxiv.org/abs/1809.11165.

Global Modeling and Assimilation Office (GMAO) (2015). *MERRA-2 inst1_2d_asm_Nx: 2d, 1-Hourly, Instantaneous, Single-Level, Assimilation, Single-Level Diagnostics V5.12.4*. Accessed: 2025-11-01. Greenbelt, MD, USA. DOI: 10.5067/3Z173KIE2TPD.

Gneiting, Tilmann (2002). “Nonseparable, Stationary Covariance Functions for Space-Time Data”. In: *Journal of the American Statistical Association* 97.458, pp. 590–600. ISSN: 01621459. URL: http://www.jstor.org/stable/3085674 (visited on 04/17/2026).

Green, P. J. and B. W. Silverman (1993). *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. 1st ed. Chapman and Hall/CRC. DOI: 10.1201/b15710.

Gu, Albert, Tri Dao, et al. (2020). *HiPPO: Recurrent Memory with Optimal Polynomial Projections*. arXiv: 2008.07669 [cs.LG]. URL: https://arxiv.org/abs/2008.07669.

Gu, Albert, Karan Goel, and Christopher Ré (2022). *Efficiently Modeling Long Sequences with Structured State Spaces*. arXiv: 2111.00396 [cs.LG]. URL: https://arxiv.org/abs/2111.00396.

Hensman, James, Nicolo Fusi, and Neil D. Lawrence (2013). *Gaussian Processes for Big Data*. arXiv: 1309.6835 [cs.LG]. URL: https://arxiv.org/abs/1309.6835.

Hensman, James, Alex Matthews, and Zoubin Ghahramani (2014). *Scalable Variational Gaussian Process Classification*. arXiv: 1411.2005 [stat.ML]. URL: https://arxiv.org/abs/1411.2005.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel (2020). *Denoising Diffusion Probabilistic Models*. arXiv: 2006.11239 [cs.LG]. URL: https://arxiv.org/abs/2006.11239.

Hochreiter, Sepp and Jürgen Schmidhuber (Nov. 1997). “Long Short-Term Memory”. In: *Neural Computation* 9.8, pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. eprint: https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

Kalman, R. E. (Mar. 1960). “A New Approach to Linear Filtering and Prediction Problems”. In: *Journal of Basic Engineering* 82.1, pp. 35–45. ISSN: 0021-9223. DOI: 10.1115/1.

3662552. eprint: https://asmedigitalcollection.asme.org/fluidsengineering/article-pdf/82/1/35/5518977/35_1.pdf. URL: https://doi.org/10.1115/1.3662552.

Kong, Zhifeng et al. (2021). *DiffWave: A Versatile Diffusion Model for Audio Synthesis*. arXiv: 2009.09761 [eess.AS]. URL: https://arxiv.org/abs/2009.09761.

Lim, Bryan et al. (2020). *Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting*. arXiv: 1912.09363 [stat.ML]. URL: https://arxiv.org/abs/1912.09363.

Little, Roderick J. A. and Donald B. Rubin (2002). *Statistical Analysis with Missing Data*. John Wiley & Sons, Inc. DOI: 10.1002/9781119013563. URL: https://doi.org/10.1002/9781119013563.

Nag, Pratik, Ying Sun, and Brian J Reich (2023). *Spatio-temporal DeepKriging for Interpolation and Probabilistic Forecasting*. arXiv: 2306.11472 [stat.ML]. URL: https://arxiv.org/abs/2306.11472.

National Center for High-Performance Computing (NCHC) (2018). *Taiwania 2 / Taiwan Computing Cloud (TWCC) High-Performance AI Cloud Platform*. National Center for High-Performance Computing, Taiwan. Supercomputer: Taiwania 2; 9 PFLOPS, 2,016 NVIDIA Tesla V100 GPUs; Operated via TWCC.

Primiceri, Giorgio E. (July 2005). "Time Varying Structural Vector Autoregressions and Monetary Policy". In: *The Review of Economic Studies* 72.3, pp. 821–852. ISSN: 0034-6527. DOI: 10.1111/j.1467-937X.2005.00353.x. eprint: https://academic.oup.com/restud/article-pdf/72/3/821/18344172/72-3-821.pdf. URL: https://doi.org/10.1111/j.1467-937X.2005.00353.x.

Rasmussen, Carl Edward and Christopher K. I. Williams (2006). *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press. URL: https://gaussianprocess.org/gpml/.

Shi, Xingjian et al. (2015). "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2015/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf.

Sims, Christopher A. (1980). "Macroeconomics and Reality". In: *Econometrica* 48.1, pp. 1–48. DOI: 10.2307/1912017.

Sohl-Dickstein, Jascha et al. (2015). *Deep Unsupervised Learning using Nonequilibrium Thermodynamics*. arXiv: 1503.03585 [cs.LG]. URL: https://arxiv.org/abs/1503.03585.

Tzeng, ShengLi and Hsin-Cheng Huang (2018). "Resolution Adaptive Fixed Rank Kriging". In: *Technometrics* 60.2, pp. 198–208. DOI: 10.1080/00401706.2017.1345701. eprint: https://doi.org/10.1080/00401706.2017.1345701. URL: https://doi.org/10.1080/00401706.2017.1345701.

Tzeng, ShengLi, Hsin-Cheng Huang, Wen-Ting Wang, and Yao-Chih Hsu (2025). *autoFRK-python: Automatic Fixed Rank Kriging. The Python version with PyTorch*. Python package version 1.2.3. URL: https://pypi.org/project/autoFRK/.

Tzeng, ShengLi, Hsin-Cheng Huang, Wen-Ting Wang, Douglas Nychka, et al. (2021). *autoFRK: Automatic Fixed Rank Kriging*. R package version 1.4.3. URL: https://CRAN.R-project.org/package=autoFRK.

Wahba, Grace and James Wendelberger (1980). "Some New Mathematical Methods for Variational Objective Analysis Using Splines and Cross Validation". In: *Monthly Weather Review* 108.8, pp. 1122–1143. DOI: 10.1175/1520-0493(1980)108<1122:SNMMFV>2.0.CO;2. URL: https://journals.ametsoc.org/view/journals/mwre/108/8/1520-0493_1980_108_1122_snmmfv_2_0_co_2.xml.

Wu, Haixu et al. (2022). *Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting*. arXiv: 2106.13008 [cs.LG]. URL: https://arxiv.org/abs/2106.13008.

Zhu, Xun et al. (2023). *Weather2K: A Multivariate Spatio-Temporal Benchmark Dataset for Meteorological Forecasting Based on Real-Time Observation Data from Ground Weather Stations*. arXiv: 2302.10493 [cs.LG]. URL: https://arxiv.org/abs/2302.10493.

A Weather2K Dataset

The Weather2K-R dataset includes a set of meteorological variables. In this study, Air Temperature is selected as the primary variable for analysis.

Table 4: Variable List of Weather2K. Description of variables, their abbreviations, and measurement units.

Variable	**Short Name**	**Unit**
Latitude	lat	degrees east
Longitude	lon	degrees north
Altitude	alt	m
Air Pressure	ap	hPa
Air Temperature	t	°C
Maximum Temperature	mxt	°C
Minimum Temperature	mnt	°C
Relative Humidity	rh	%
Precipitation in 3h	p3	mm
Wind Direction	wd	degrees
Wind Speed	ws	m s^{-1}
Maximum Wind Direction	mwd	degrees
Maximum Wind Speed	mws	m s^{-1}

B MERRA-2 Dataset

The MERRA-2 dataset includes a set of atmospheric variables. In this study, Surface Skin Temperature is selected as the primary variable for analysis.

Table 5: Variable List of MERRA-2. Description of variables, their abbreviations, and measurement units.

Variable	**Short Name**	**Unit**
Longitude	lon	degrees east
Latitude	lat	degrees north
Time	time	minutes since 2024-06-01 00:00:00
2-Meter Air Temperature	t2m	K
Total Precipitable Liquid Water	tql	kg m^{-2}
Total Column Odd Oxygen	tox	kg m^{-2}
2-Meter Eastward Wind	u2m	m s^{-1}
Surface Pressure	ps	Pa
Tropopause Temperature Using Blended TROPP Estimate	tropt	K
Northward Wind at 50 Meters	v50m	m s^{-1}
Zero Plane Displacement Height	disph	m
Total Column Ozone	to3	Dobsons
Surface Skin Temperature	ts	K
10-Meter Air Temperature	t10m	K
Tropopause Pressure Based on Thermal Estimate	troppt	Pa

Variable	Short Name	Unit
Total Precipitable Ice Water	tqi	kg m^{-2}
Sea Level Pressure	slp	Pa
Tropopause Pressure Based on Blended Estimate	troppb	Pa
Total Precipitable Water Vapor	tqv	kg m^{-2}
2-Meter Northward Wind	v2m	m s^{-1}
Tropopause Specific Humidity Using Blended TROPP Estimate	tropq	kg kg^{-1}
10-Meter Northward Wind	v10m	m s^{-1}
Eastward Wind at 50 Meters	u50m	m s^{-1}
10-Meter Eastward Wind	u10m	m s^{-1}
2-Meter Specific Humidity	qv2m	kg kg^{-1}
Tropopause Pressure Based on EPV Estimate	troppv	Pa
10-Meter Specific Humidity	qv10m	kg kg^{-1}

C 2b-8 Dataset

The 2b-8 dataset originates from the *KAUST Spatial Statistics Competition (2022)*, which provides large-scale simulated datasets designed to evaluate the performance of spatial statistical methods under a unified experimental framework. The dataset is generated using the ExaGeoStat high-performance statistical computing framework and is based on reproducible Gaussian process (GP) simulations, offering standardized and comparable benchmarking conditions (Abdulah, Alamri, Nag, et al., 2022; Abdulah, Alamri, Ltaief, et al., 2022).

In Sub-competition 2b, a non-separable and stationary Gaussian process model is adopted,

where the covariance structure follows the formulation proposed in Gneiting (2002). For any two spatial locations $s \in [0,1]^2$ and temporal lag $t \in \mathbb{R}$, the covariance function is defined as:

$$C(\boldsymbol{h}, u; \boldsymbol{\theta}) = \frac{\sigma^2}{a_t|u|^{2\alpha}+1} M_\nu\left(\frac{\|\boldsymbol{h}\|/a_s}{(a_t|u|^{2\alpha}+1)^{\beta/2}}\right), \tag{37}$$

where $\boldsymbol{h}$ denotes the spatial lag and u denotes the temporal lag. The parameter $\sigma^2 > 0$ represents the variance, $\nu > 0$ and $\alpha \in [0,1]$ control smoothness, $a_s, a_t > 0$ are spatial and temporal scaling parameters, and $\beta \in (0,1]$ governs the interaction strength between space and time. $M_\nu(\cdot)$ denotes the Matérn correlation function.

According to the competition protocol, Sub-competitions 2a and 2b generate a total of 18 datasets, covering multiple configurations of spatial scales (weak, moderate, and strong), spatial resolutions (1K and 10K locations), and temporal lengths (100 and 1000 time steps). Three types of missingness mechanisms are considered for prediction tasks: random spatial removal (RS), random spatio-temporal removal (RST), and systematic removal of the last 10 time steps (T10). All configurations are summarized in Table 1 of Abdulah, Alamri, Nag, et al. (2022). The 2b-8 dataset corresponds to a specific configuration within this benchmark, defined by a particular combination of spatial resolution, temporal length, and model parameters.