LATENT NEURAL FIELD DIFFUSION FOR SPATIO-TEMPORAL FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatio-temporal foundation models (STFMs) seek to learn transferable patterns from heterogeneous data for reuse across domains. Unlike time-series foundation models, however, STFMs must additionally handle the spatial dimension, where real-world observations exhibit diverse spatial structures, resolutions, and scales. We formulate spatio-temporal foundation modeling from a field perspective, treating discrete observations as samples from continuous fields and enabling prediction at arbitrary spatio-temporal coordinates. This formulation introduces two key challenges: learning a shared field representation across heterogeneous spatial geometries and modeling the uncertainty of diverse tasks under partial observations. We propose FieldDiff, a two-stage latent neural field diffusion framework. FieldDiff learns a unified latent field by jointly encoding observed values and physical coordinates with globally shared and geometry-adaptive tokens, together with variational regularization. It further unifies forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation as conditional field generation, using residual latent diffusion to model uncertainty unresolved by the observations. Experiments across diverse spatio-temporal datasets demonstrate strong multi-task performance and robust zero-shot and few-shot generalization across domains and spatial scales.a

1 INTRODUCTION

Spatio-temporal data encompass observations that evolve jointly over spatial and temporal dimensions. Such data arise in an extremely diverse range of domains, including urban systems (Liu & Zhang, 2026; Li et al., 2025a), transportation (Wang et al., 2024; He et al., 2025), environmental monitoring (Chen et al., 2024; Xin et al., 2026), and scientific applications (Wu et al., 2024; Park et al., 2025). Although these domains differ in physical semantics, spatial organization, and temporal resolution, they often exhibit recurring spatial dependency and temporal dynamics. Such cross-domain commonality naturally motivates the development of spatio-temporal foundation models (STFMs). By pretraining on heterogeneous spatio-temporal data, STFMs learn transferable patterns that can be reused across downstream domains and tasks.

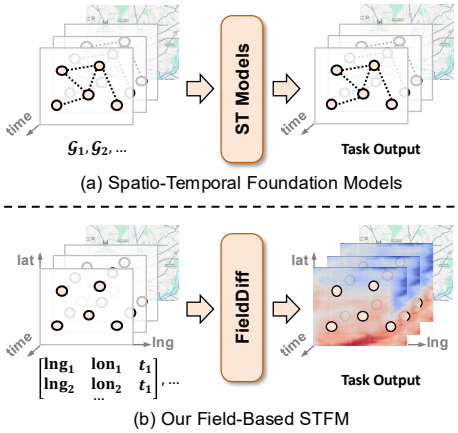

Figure 1: Paradigms for STFMs.

Most existing STFMs (Li et al., 2024; Yuan et al., 2024; Zhong et al., 2025) mainly focus on temporal forecasting following time-series foundation models (Wang et al., 2025; Shi et al., 2025; Lyu et al., 2026). Only a few recent works (Yuan et al., 2025) further consider multiple tasks including spatial extrapolation. However, the spatial dimension makes spatio-temporal data fundamentally different from time series. Furthermore, real-world observations are often distributed over irregular, non-grid spatial structures with complex geometry and non-uniform density, and such spatial irregularity varies across domains. This poses a key challenge for models built on fixed and fully observed spatial supports to learn shared representations across domains.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To address this issue, we reformulate spatio-temporal foundation modeling from a field perspective (Sitzmann et al., 2020). We regard discrete spatio-temporal data as samples from an underlying continuous spatio-temporal field. This view removes the dependence on a particular graph or grid discretization. It also enables different downstream tasks to be formulated as querying field values at arbitrary spatio-temporal coordinates. Recent work such as OmniField (Valencia et al., 2026) has demonstrated the potential of neural fields for unified modeling multimodal spatio-temporal data within a region. However, pretraining a generalizable field-based STFM across domains and regions remains largely unexplored.

Extending neural fields to STFMs raises two fundamental challenges. The first is *scalability across heterogeneous domains*. Different domains span drastically different spatial extents, resolutions, and sampling densities, ranging from localized urban flows to large-scale meteorological fields. These scale variations induce distinct spatial correlation structures, making a fixed field representation difficult to generalize across domains. A scalable STFM therefore requires a canonical latent space with a shared topology but domain-adaptive spatial representations. The second is *multi-task modeling under heterogeneous patterns*. As the observed support becomes sparse or structurally incomplete, recovering the unobserved field in different spatio-temporal tasks becomes inherently underdetermined. Existing conditioned neural fields typically formulate this process as deterministic reconstruction and thus collapse such conditional uncertainty into a single estimate. A general STFM should instead unify these tasks as partial-observation modeling and learn the conditional distribution of the unobserved field.

To this end, we propose **FieldDiff**, a two-stage latent neural field diffusion framework for spatio-temporal foundation modeling. In the first stage, FieldDiff maps heterogeneous observations into a canonical latent field by jointly encoding values and physical spatio-temporal coordinates, and compressing them with globally shared and geometry-adaptive latent tokens. A variational constraint further regularizes the learned field representation for generative modeling. In the second stage, we unify multi-tasks as conditional generation from partial observations. A condition encoder predicts a condition-dependent latent reference, while residual latent diffusion models the uncertainty unresolved by the observations. The recovered neural field can be queried at arbitrary spatio-temporal coordinates, enabling unified modeling across heterogeneous domains, spatial scales, and tasks. Our main contributions are summarized as follows:

- We formulate spatio-temporal foundation modeling from a field perspective, treating heterogeneous discrete observations as samples from continuous spatio-temporal fields to unify spatial structures, resolutions, and arbitrary query coordinates.
- We develop a two-stage latent generative framework that learns a geometry-adaptive variational field representation and performs conditional residual diffusion, enabling unified partial-observation generation across diverse spatio-temporal tasks.
- Extensive experiments demonstrate that FieldDiff achieves strong performance across multiple spatio-temporal datasets and tasks. It also exhibits robust zero-shot and few-shot generalization across domains and spatial scales.

2 Related Work

2.1 Spatio-Temporal Foundation Models

The remarkable success of large neural networks in NLP (Brown et al., 2020; Nie et al., 2025) and CV (Kirillov et al., 2023; Yang et al., 2025) has recently inspired increasing efforts to developing foundation models for spatio-temporal data mining. STFMs are pretrained on large collections of heterogeneous spatio-temporal data, with the goal of generalizing across multiple tasks. Representative STFMs can be broadly categorized into LLM-based and pretraining from scratch. LLM-based methods, such as UrbanGPT (Li et al., 2024) and ST-LLM (Liu et al., 2025), adapt existing LLMs to spatio-temporal data through task-specific fine-tuning. However, the modality gap between natural language and structured spatio-temporal observations limits the effectiveness of directly transferring language priors. This has motivated a second line of work that pretrains STFMs from scratch. UniST (Yuan et al., 2024) and UrbanDiT (Yuan et al., 2025) adopt prompt-based mechanisms to unify heterogeneous datasets and downstream tasks within a shared pre-training framework. More recent

studies further focus on spatial heterogeneity across domains. FactoST (Zhong et al., 2025) decouples learning into temporal backbone pretraining and spatio-temporal adaptation. HURST (Zhou et al., 2026) introduces mixture-of-experts to capture heterogeneous spatial patterns across different domains. Most existing STFMs primarily focus on temporal forecasting, while only a few efforts (Yuan et al., 2025; Zhou et al., 2026) extend to spatially incomplete settings. Additionally, these methods remain tied to domain-specific spatial discretizations, whose geometry, density, resolution, and scale may vary substantially across datasets.

2.2 Spatio-Temporal Fields

Neural fields model continuous coordinate-based functions that map coordinates to signal values (Sitzmann et al., 2020). By representing signals continuously, they support compact representation, high-fidelity reconstruction, and resolution-free querying, which has led to broad adoption in visual reconstruction and generation (Lan et al., 2026; Wang et al., 2026). In the spatio-temporal domain, STFNN (Feng et al., 2024) made an early attempt to bridge field-based and graph-based modeling for air-quality inference, formulating the task as continuous field reconstruction from discrete observations. SCENT (Park et al., 2025) further extended neural fields to scientific spatio-temporal data and unified multiple downstream tasks, including interpolation, reconstruction, and forecasting. Building on SCENT, OmniField (Valencia et al., 2026) further incorporates cross-modal interaction for multimodal spatio-temporal modeling within a shared region. In contrast, our approach targets cross-domain and cross-region foundation modeling by learning a unified field latent space over heterogeneous spatial supports.

3 Preliminaries

Spatio-Temporal Data. We consider a spatio-temporal system consisting of N spatial entities observed over T time steps. Its observations are represented as $\boldsymbol{X} = [X_{n,t}] \in \mathbb{R}^{N\times T}$, where $X_{n,t}$ denotes the observation of the n-th spatial entity at time step t. For graph-based data, N corresponds to the number of graph nodes, while for grid-based data with spatial resolution $H \times W$, we flatten the spatial dimensions and set $N = H \times W$. Therefore, heterogeneous spatio-temporal data with different spatial organizations can be uniformly represented by an $N \times T$ observation matrix.

Spatio-Temporal Downstream Tasks. A general-purpose spatio-temporal foundation model is expected to support different downstream tasks under a unified formulation. We characterize each task by the relationship between the observed and target spatio-temporal coordinates. Let $\boldsymbol{M} \in \{0,1\}^{N\times T}$ denote an observation mask, where $M_{n,t} = 1$ indicates that $X_{n,t}$ is observed and $M_{n,t} = 0$ indicates a target value to be inferred. Forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation correspond to different structured forms of $\boldsymbol{M}$. Detailed definitions are deferred to Appendix A. We further denote the observed and unobserved spatio-temporal coordinate sets as $\mathbb{O} = \{(n,t) \mid M_{n,t} = 1\}$ and $\mathbb{U} = \{(n,t) \mid M_{n,t} = 0\}$. Under this formulation, all downstream tasks can be viewed as recovering the unobserved portion of a spatio-temporal system conditioned on its partial observations as $p(\boldsymbol{X}_{\mathbb{U}} \mid \boldsymbol{X}_{\mathbb{O}}, \boldsymbol{M})$. This provides a unified formulation for the conditional generative modeling in FieldDiff.

4 Methodology

4.1 Overview

As shown in Figure 2, in the first stage, we construct a field-based variational autoencoder to map heterogeneous spatio-temporal observations into a structured latent field. Given an observation $\boldsymbol{X} \in \mathbb{R}^{N\times T}$ together with its spatial coordinates $\boldsymbol{S}$ and temporal coordinates $\boldsymbol{T}$, the encoder produces $\mathbf{Z} = \mathcal{E}_\phi(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{T}) \in \mathbb{R}^{T\times K\times D}$, where K denotes the number of latent spatial tokens and D is the latent dimension. Thus, heterogeneous spatial structures are unified into a common latent topology. A coordinate-conditioned decoder subsequently reconstructs field values at arbitrary spatio-temporal coordinates. In the second stage, FieldDiff models the conditional distribution of the complete latent field from partial observations. A condition encoder aligned with field encoder maps the observed context into the same latent topology, while a conditional prediction head provides a coarse latent

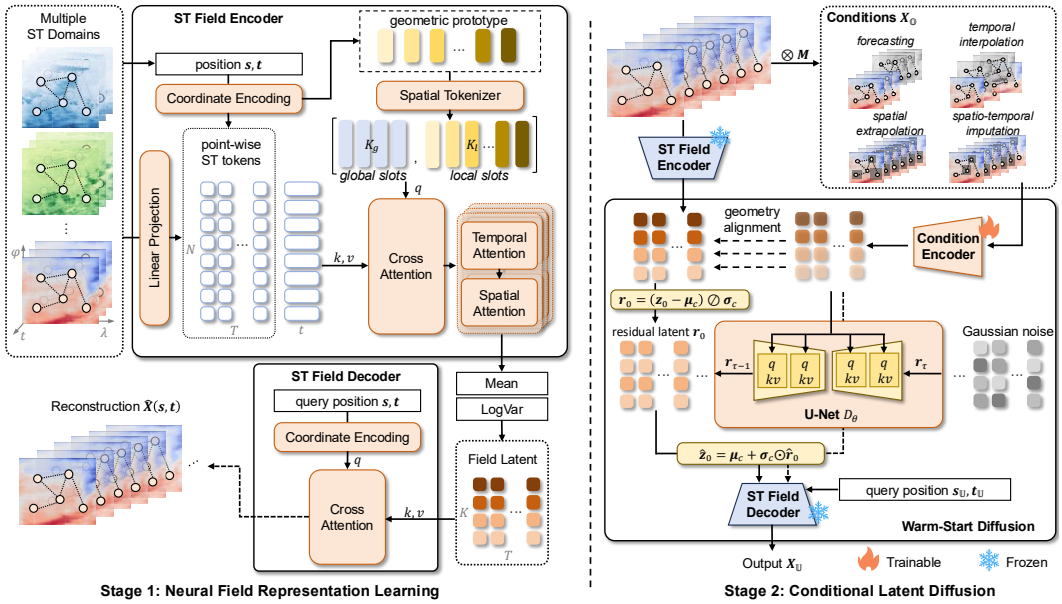

Figure 2: The overall architecture of the proposed spatio-temporal foundation model FieldDiff.

estimate. The diffusion model then refines the remaining uncertainty in residual latent space and reconstructs the complete field for downstream querying.

4.2 Neural Field Representation Learning

Spatio-Temporal Coordinate Encoding. FieldDiff directly encodes physical spatial and temporal coordinates instead of relying on graph indices or grid positions, enabling a unified representation across heterogeneous spatial resolutions. For node n, let $\boldsymbol{s}_n = (\lambda_n, \varphi_n)$ denote its longitude and latitude. We perform a *global-local* spatial coordinate encoding by first mapping $\boldsymbol{s}_n$ into the unit earth sphere

$$\boldsymbol{g}_n = [\cos\varphi_n \cos\lambda_n,\ \cos\varphi_n \sin\lambda_n,\ \sin\varphi_n]^\top . \tag{1}$$

Then we define the domain-relative coordinate $\boldsymbol{l}_n^{(d)} = (\boldsymbol{g}_n - \boldsymbol{c}_d)/r_d$, where $\boldsymbol{c}_d$ and r_d denote the spatial center and scale of domain d. Thus, $\boldsymbol{g}_n$ preserves absolute geographic information, while $\boldsymbol{l}_n^{(d)}$ captures normalized local geometry. Both coordinates are encoded using fixed Gaussian Fourier features and projected into the spatial embedding

$$\boldsymbol{e}_n^s = h_s\left(\left[\gamma(\boldsymbol{g}_n; \boldsymbol{B}_g); \gamma(\boldsymbol{l}_n^{(d)}; \boldsymbol{B}_l); \boldsymbol{g}_n; \boldsymbol{l}_n^{(d)}\right]\right), \tag{2}$$

where $\gamma(\boldsymbol{v}; \boldsymbol{B}) = [\sin(\boldsymbol{B}^\top \boldsymbol{v}); \cos(\boldsymbol{B}^\top \boldsymbol{v})]$ denotes the Gaussian Fourier mapping, $\boldsymbol{B}$ is a fixed random projection matrix sampled from a zero-mean Gaussian distribution, $\boldsymbol{B}_g$ and $\boldsymbol{B}_l$ are the corresponding projection matrices for the global and local coordinates, and $h_s(\cdot)$ is a learnable projection network. For the temporal coordinate $\boldsymbol{t}_t$, we encode periodic calendar information, relative time, and sampling interval through a deterministic transformation $\rho(\boldsymbol{t}_t)$ as $\boldsymbol{e}_t^t = h_t(\gamma(\rho(\boldsymbol{t}_t); \boldsymbol{B}_t))$. Details of temporal encoding are provided in Appendix B.1. We embed the observed value $X_{n,t}$ using a learnable value projection $h_x(\cdot)$ and fuse it with the corresponding spatial and temporal embeddings. The resulting point-wise spatio-temporal tokens $\boldsymbol{h}_{n,t}$ jointly encodes the observed value and its continuous spatial and temporal context, and serves as the input to the subsequent field tokenization.

Field Tokenization. To remove the dependence on domain-specific spatial discretizations, we compress the N_d observations of domain d into a fixed number of latent spatial tokens. We use $K = K_g + K_l$ latent tokens, where K_g globally shared tokens capture transferable field patterns and K_l local tokens are derived from the physical geometry of each domain. Detailed geometry-based local tokens assignment is provided in Appendix B.2. Let $\boldsymbol{H}_t^{(d)} \in \mathbb{R}^{N_d \times D}$ denote the point-wise

representations of the N_d observations in domain d at time step t. We construct the latent query matrix $\boldsymbol{Q}^{(d)} = [\boldsymbol{Q}_g; \boldsymbol{Q}_l^{(d)}] \in \mathbb{R}^{K \times D}$, where $\boldsymbol{Q}_g \in \mathbb{R}^{K_g \times D}$ and $\boldsymbol{Q}_l^{(d)} \in \mathbb{R}^{K_l \times D}$ denote the globally shared and domain-adaptive local queries, respectively. For each time step, the latent field is then obtained through cross-attention,

$$\boldsymbol{Z}_t^{(d)} = \mathrm{Attn}\left(\boldsymbol{Q}^{(d)}, \boldsymbol{H}_t^{(d)}, \boldsymbol{H}_t^{(d)}\right), \qquad \boldsymbol{Z}_t^{(d)} \in \mathbb{R}^{K \times D}. \tag{3}$$

where $\mathrm{Attn}(\cdot)$ denotes multi-head cross-attention. Stacking the latent tokens over all time steps gives $\mathbf{Z}^{(d)} \in \mathbb{R}^{T \times K \times D}$. This maps arbitrary spatial supports N_d into a shared latent topology while preserving domain-specific geometry. Temporal and spatial axial attention are then applied to refine dependencies within the latent field.

Latent Space Geometry. We further introduce a lightweight variational constraint to regularize the latent distribution. The detailed posterior parameterization and training objective are provided in Appendix B.3. Raw spatio-temporal observations from heterogeneous domains can exhibit anisotropic local densities, which leads to heterogeneous score-field curvature and unstable denoising trajectories. By mapping observations into a shared field representation and weakly regularizing its distribution toward an isotropic prior, FieldDiff encourages a better-conditioned latent geometry. Formally, let

$$\kappa_X^{(d)}(\boldsymbol{x}) = \left\| \nabla_{\boldsymbol{x}}^2 \log p_X^{(d)}(\boldsymbol{x}) \right\|_{\mathrm{op}}, \qquad \kappa_Z^{(d)}(\boldsymbol{z}) = \left\| \nabla_{\boldsymbol{z}}^2 \log p_Z^{(d)}(\boldsymbol{z}) \right\|_{\mathrm{op}} \tag{4}$$

denote the local score curvatures in the observation and latent spaces, respectively. Under a bi-Lipschitz twice-differentiable encoder, the latent curvature can be bounded as

$$\kappa_Z^{(d)}(\boldsymbol{z}) \le C_{1,d}\, \kappa_X^{(d)}(\boldsymbol{x}) + C_{2,d}, \qquad \boldsymbol{x} = E_\phi^{(d),-1}(\boldsymbol{z}) \tag{5}$$

where $C_{1,d}$ and $C_{2,d}$ depend on the local smoothness and conditioning of the encoder. This suggests that learning a smooth and weakly regularized field representation can reduce the geometric difficulty encountered by the subsequent diffusion model. The complete assumptions and derivation are provided in Appendix C.

4.3 Conditional Latent Diffusion

Unified Partial Observation Modeling. FieldDiff formulates diverse spatio-temporal tasks as a unified partial-observation recovery problem. Given an observation mask $\boldsymbol{M}$, the observed and target sets are denoted by $\mathbb{O}$ and $\mathbb{U}$, respectively. Forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation differ only in the structure of $\boldsymbol{M}$ and share the same objective $p_\theta(\mathbf{X}_\mathbb{U} \mid \mathbf{X}_\mathbb{O}, \boldsymbol{M})$. To preserve the representation learned in Stage I, the condition encoder inherits the same value representation, coordinate encoding, and spatial tokenizer. Observed values are encoded, whereas unobserved positions are represented by a learnable unknown embedding while retaining their coordinates. The resulting condition latent $\mathbf{C} \in \mathbb{R}^{T \times K \times D}$ therefore shares the same structured topology as the complete latent field.

Warm-Start Latent Diffusion. Starting reverse diffusion from an unconditional Gaussian ignores the information already revealed by the partial observations and requires the denoiser to recover both the coarse field structure and its unresolved uncertainty. We instead use the condition latent $\mathbf{C}$ to predict the condition-dependent latent reference $(\boldsymbol{\mu}_c, \boldsymbol{\sigma}_c) = h_c(\mathbf{C})$, where $h_c(\cdot)$ is a prediction head and $\boldsymbol{\mu}_c$ and $\boldsymbol{\sigma}_c$ denote the latent center and scale, respectively. Let $\boldsymbol{z}_0 = \mathrm{vec}(\mathbf{Z})$ denote the complete Stage I latent. We then construct the standardized residual $\boldsymbol{r}_0 = (\boldsymbol{z}_0 - \boldsymbol{\mu}_c) \oslash \boldsymbol{\sigma}_c$, which represents the latent variation that remains unresolved by the partial observations. The forward diffusion process is applied to the residual as

$$\boldsymbol{r}_\tau = \alpha_\tau \boldsymbol{r}_0 + \sigma_\tau \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \tag{6}$$

where τ denotes the diffusion step, and α_τ and σ_τ are the corresponding signal and noise coefficients. Equivalently, in the original latent space,

$$\boldsymbol{z}_\tau = \alpha_\tau \boldsymbol{z}_0 + (1 - \alpha_\tau)\boldsymbol{\mu}_c + \sigma_\tau \boldsymbol{\epsilon}. \tag{7}$$

The condition latent $\mathbf{C}$ is further injected into the denoising network $D_\theta(\cdot)$ through cross-attention, enabling the reverse process to focus on uncertainty unresolved by the partial observations. Thus, as the signal component vanishes at large noise levels, the diffusion trajectory remains centered around the condition-derived latent estimate rather than an unconditional origin.

Training and Inference. FieldDiff is trained in two stages. In Stage I, the neural field VAE is optimized with $\mathcal{L}_{\mathrm{I}} = \mathcal{L}_{\mathrm{rec}} + \beta_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}$, where $\mathcal{L}_{\mathrm{rec}}$ denotes the field reconstruction loss, $\mathcal{L}_{\mathrm{KL}}$ regularizes the posterior toward the isotropic Gaussian prior, and β_{KL} controls the regularization strength. After convergence, the Stage I encoder and decoder are frozen, and the posterior mean is used as the complete latent target for Stage II. During Stage II training, a task-specific observation mask $\boldsymbol{M}$ is sampled to construct partial observations, which are encoded into the condition latent $\mathbf{C}$. The standardized residual is perturbed according to Eq. (6), and the conditional denoiser D_θ is optimized by

$$\mathcal{L}_{\mathrm{diff}} = \mathbb{E}_{\sigma,\boldsymbol{\epsilon}}\left[w(\sigma)\left\|D_\theta(\boldsymbol{r}_\sigma, \sigma, \mathbf{C}) - \boldsymbol{r}_0\right\|_2^2\right], \tag{8}$$

where $w(\sigma)$ denotes the noise-dependent weighting function. At inference, the partial observations are first encoded into $\mathbf{C}$. The residual latent is initialized as $\boldsymbol{r}_{\sigma_{\mathrm{start}}} \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathrm{start}}^2\boldsymbol{I})$, where σ_{start} denotes the initial noise level of the reverse diffusion process. It is then progressively denoised following a decreasing noise schedule to obtain $\hat{\boldsymbol{r}}_0$. The complete latent is recovered by

$$\hat{\boldsymbol{z}}_0 = \boldsymbol{\mu}_c + \boldsymbol{\sigma}_c \odot \hat{\boldsymbol{r}}_0, \tag{9}$$

which is reshaped into $\hat{\mathbf{Z}}$ and decoded at the target coordinates through the neural field decoder. The complete training and inference procedures are summarized in Appendix D.

5 Experiments

Datasets. We pretrain FieldDiff on 20 spatio-temporal datasets spanning urban mobility, traffic, energy, environment, and weather, comprising 445.3M observations with 60-5,016 spatial nodes and sampling intervals from 5 minutes to 1 hour. We splits training and testing sets from the pretraining datasets for in-domain evaluation. To evaluate generalization beyond the pretraining distribution, we further use 8 datasets covering cross-region, cross-domain, and cross-scale generalization settings. These evaluation datasets contain 81-2,048 spatial nodes with sampling intervals ranging from 5 minutes to 1 month. Details of the datasets are provided in Appendix E.1.

Baselines. To comprehensively evaluate the performance of our proposed FieldDiff, we select state-of-art models spanning two distinct categories: (1) **Single-Task Models**: These models are trained for specific spatio-temporal tasks, including (a) *prediction models*: Graph WaveNet (Wu et al., 2019), PatchTST (Nie et al., 2023), PromptST (Zhang et al., 2023), STID (Shao et al., 2022), and iTransformer (Liu et al., 2024). (b) *interpolation models*: BRITS (Cao et al., 2018), CSDI (Tashiro et al., 2021), GRIN (Cini et al., 2022), and ImputeFormer (Nie et al., 2024). (c) *extrapolation models*: BKMF (Lei et al., 2022), IGNNK (Wu et al., 2021), INCREASE (Zheng et al., 2023), and KITS (Xu et al., 2025). (2) **Pretrained Models**: These models are trained across multiple datasets, including ClimaX (Nguyen et al., 2023), OpenCity (Li et al., 2025b), UrbanGPT (Li et al., 2024), UniST (Yuan et al., 2024), OmniField (Valencia et al., 2026), and UrbanDiT (Yuan et al., 2025). Details of baselines are provided in Appendix E.4.

5.1 One-for-All Results

Spatio-Temporal Prediction. For one-for-all prediction task, we set both the historical input window and prediction horizon to 12 time steps. All pretrained models are jointly trained on the same pretraining datasets, whereas task-specific baselines are trained independently on each dataset. As shown in Table 1, pretrained models generally exhibit stronger and more consistent performance than task-specific baselines across heterogeneous datasets. Single-task methods are more sensitive to dataset-specific spatial and temporal characteristics, and are outperformed by pretrained models on most benchmarks, highlighting the benefit of transferring shared spatio-temporal patterns across domains. Among pretrained baselines, LLM-based methods such as UrbanGPT show limited competitiveness, suggesting that the modality gap between language representations and structured spatio-temporal signals can hinder effective knowledge transfer. OmniField performs well on several regular urban datasets but degrades substantially on large-scale highway sensor networks, indicating that a shared continuous-field representation alone may struggle with highly sparse and irregular spatial supports. UrbanDiT is the strongest competing foundation model overall, yet FieldDiff achieves lower MAE on 10 of 11 datasets and lower RMSE on 10 of 11 datasets. Compared with UrbanDiT, FieldDiff reduces MAE and RMSE by an average of 7.62% and 9.61%, respectively,

Table 1: One-for-all spatio-temporal prediction performance comparison. Best results are in **bold** and second-best are underlined.

Method	Metric	*NYCbike*	*NYCtaxi*	*TRTbike*	*METRLA*	*PeMS4flow*	*PeMS8flow*	*GBA*	*GLA*	*PVUS*	*AQIpm25*	*ERA5temp*
GraphWaveNet	MAE	17.41	18.82	1.27	6.15	34.23	26.95	44.26	44.16	3.95	26.70	4.77
	RMSE	31.34	30.54	4.66	13.63	48.54	39.77	62.19	66.59	10.38	43.77	5.85
PatchTST	MAE	18.91	20.81	1.70	6.00	31.65	26.45	53.97	62.20	4.17	26.55	4.02
	RMSE	32.92	33.37	4.71	13.61	49.42	39.70	75.63	88.76	10.91	43.26	4.75
PromptST	MAE	20.56	24.44	1.68	5.55	30.98	27.88	36.23	37.97	3.71	28.09	4.12
	RMSE	38.70	36.28	5.50	12.71	48.07	40.87	52.19	53.98	9.76	44.65	4.98
STID	MAE	16.04	16.14	1.62	5.97	29.26	22.63	33.74	42.07	4.07	25.96	4.15
	RMSE	30.19	25.89	4.81	13.55	45.98	33.95	52.45	60.70	10.47	42.60	4.95
iTransformer	MAE	17.35	21.51	1.69	5.47	28.81	24.74	61.77	54.02	5.10	30.10	4.99
	RMSE	30.31	35.87	4.66	**12.39**	44.47	38.45	84.09	83.57	13.01	48.47	5.74
ClimaX	MAE	13.70	13.54	1.29	5.69	29.76	22.69	34.27	38.63	3.70	25.89	3.18
	RMSE	24.20	22.99	4.47	13.49	45.63	33.76	50.86	57.09	9.89	43.04	4.41
OpenCity	MAE	16.45	18.37	1.36	6.15	30.41	22.78	40.91	41.23	4.29	26.13	3.28
	RMSE	28.17	31.43	4.34	13.88	48.50	35.26	62.31	64.12	9.55	46.36	4.53
UrbanGPT	MAE	17.64	17.99	1.68	7.01	44.99	39.68	85.10	85.39	4.32	29.29	3.77
	RMSE	30.21	30.95	4.76	13.03	62.47	53.70	104.53	107.43	9.89	45.36	4.98
UniST	MAE	11.16	10.74	1.37	5.71	27.36	22.92	37.67	40.84	4.20	29.65	3.94
	RMSE	18.89	17.12	4.36	13.35	45.11	34.69	60.14	64.53	9.78	50.17	4.73
OmniField	MAE	18.77	10.28	1.38	6.47	27.34	25.35	62.17	139.37	5.61	28.39	14.82
	RMSE	25.19	17.20	4.42	12.89	43.55	40.75	88.65	172.94	11.99	43.67	18.26
UrbanDiT	MAE	15.31	10.52	1.19	5.85	27.38	22.12	32.65	36.34	3.63	29.10	3.16
	RMSE	27.07	**16.84**	4.16	12.97	43.45	34.07	51.23	57.28	9.47	48.60	5.53
FieldDiff	MAE	**9.15**	**10.07**	**1.17**	**5.41**	**27.04**	**21.90**	**30.19**	**36.00**	**3.37**	**25.82**	**3.13**
	RMSE	**15.45**	17.04	**3.83**	12.77	**42.16**	**33.22**	**50.04**	**53.20**	**9.19**	**42.51**	**4.20**

Table 2: One-for-all spatial extrapolation performance comparison. Best results are in **bold** and second-best are underlined.

Method	Metric	*NYCbike*	*NYCtaxi*	*TRTbike*	*METRLA*	*PeMS4flow*	*PeMS8flow*	*GBA*	*GLA*	*PVUS*	*AQIpm25*	*ERA5temp*
BKMF	MAE	11.86	12.01	1.74	8.01	75.56	69.19	55.64	65.86	3.86	20.39	3.73
	RMSE	22.23	19.91	4.66	12.87	104.70	96.48	78.97	94.65	10.13	33.36	5.60
IGNNK	MAE	15.17	15.99	3.55	9.93	85.46	81.34	60.25	73.19	4.80	22.00	4.82
	RMSE	26.65	26.76	8.13	14.61	116.01	110.57	83.57	101.47	10.88	36.49	6.81
INCREASE	MAE	14.38	16.40	2.75	8.00	83.59	80.03	58.47	65.42	3.63	19.33	3.17
	RMSE	28.22	29.79	8.17	14.71	116.91	111.69	82.82	101.16	11.22	33.05	4.95
KITS	MAE	14.63	16.04	3.16	8.36	82.48	81.01	50.65	59.12	3.80	18.81	3.99
	RMSE	29.86	28.84	9.09	14.19	115.00	112.44	74.83	89.66	11.64	33.21	5.73
OmniField	MAE	6.17	6.62	1.23	6.17	29.62	28.55	59.81	114.42	5.15	23.01	14.95
	RMSE	10.06	11.07	3.06	11.41	46.34	46.06	86.99	151.40	11.77	36.27	18.34
UrbanDiT	MAE	7.72	8.22	1.15	8.35	47.16	39.85	64.09	72.94	5.73	22.25	5.84
	RMSE	12.91	13.36	2.99	15.43	68.31	58.92	94.74	110.59	15.07	37.81	9.18
FieldDiff	MAE	**5.83**	**6.31**	**0.96**	**5.39**	**27.93**	**26.30**	**38.62**	**44.23**	**3.19**	**18.57**	**3.11**
	RMSE	**9.62**	**10.91**	**2.34**	**11.21**	**45.61**	**45.45**	**58.96**	**69.33**	**9.07**	**31.22**	**4.16**

across all benchmarks. The gains are particularly pronounced on *NYCbike*, where MAE and RMSE are reduced by 40.24% and 42.93%, respectively. The consistent advantage over UrbanDiT further suggests that diffusion over a canonical latent field is better suited to heterogeneous spatio-temporal data than directly modeling noisy observations in the original data space.

Temporal Interpolation. We set the missing ratio to 0.5, where observations at even-numbered time steps are retained and those at odd-numbered time steps are treated as targets. As shown in Table 7, FieldDiff achieves the best performance across most datasets, demonstrating its ability to recover temporally missing observations under a unified pretrained model.

Spatial Extrapolation. We mask 50% of spatial locations over the entire temporal sequence and evaluate the ability to infer signals at completely unseen nodes. As shown in Table 2, FieldDiff consistently outperforms all baselines, reducing MAE and RMSE by an average of 10.65% and 9.82%, respectively, compared with the best baseline on each dataset. These gains are particularly notable because several competing methods are specifically designed for spatial kriging, highlighting the advantage of FieldDiff in learning transferable spatial representations beyond fixed graph structures.

Spatio-Temporal Imputation. We randomly mask 50% of spatio-temporal observations to simulate irregular missingness across both space and time. Table 8 shows that FieldDiff achieves the best

Table 3: Zero-shot and few-shot spatio-temporal prediction performance comparison of pretrained models. Best results are in **bold** and second-best are underlined.

Method	Metric	Zero-shot				5% Few-shot				10% Few-shot			
		DCbike	*PeMSBAY*	*ERA5prec*	*CMIP6clt*	*DCbike*	*PeMSBAY*	*ERA5prec*	*CMIP6clt*	*DCbike*	*PeMSBAY*	*ERA5prec*	*CMIP6clt*
ClimaX	MAE	3.69	2.29	1.10	15.76	2.44	2.24	1.06	13.04	2.27	2.27	1.07	12.95
	RMSE	11.25	5.73	4.67	19.15	8.05	5.65	4.57	16.94	7.87	5.66	4.56	16.84
OpenCity	MAE	3.18	2.25	1.11	17.66	2.75	2.14	1.06	13.90	2.83	**2.11**	1.05	13.83
	RMSE	10.58	5.22	4.15	23.06	8.51	5.09	4.15	17.39	8.44	5.08	4.12	17.29
UrbanGPT	MAE	4.30	3.35	1.58	15.48	2.33	3.05	1.07	13.52	2.36	3.00	1.04	13.32
	RMSE	11.11	6.06	4.88	19.94	8.80	5.80	4.18	16.19	8.24	5.87	4.13	17.05
UniST	MAE	3.39	2.22	1.10	15.37	**2.23**	2.22	1.07	13.12	2.38	2.15	1.09	13.05
	RMSE	11.03	5.52	4.82	19.93	8.10	5.52	4.18	16.54	8.07	5.42	4.10	16.50
OmniField	MAE	3.03	5.55	1.66	15.50	2.60	4.00	1.48	12.99	2.48	3.96	1.48	13.04
	RMSE	11.07	9.53	4.25	19.42	7.87	7.10	4.17	16.37	7.55	7.08	4.17	16.45
UrbanDiT	MAE	3.33	2.62	1.48	15.48	2.79	2.52	1.37	13.29	2.82	2.52	1.38	13.46
	RMSE	10.39	5.91	4.26	19.75	**7.36**	5.70	4.35	16.74	7.51	5.69	4.35	16.97
FieldDiff	MAE	**2.76**	**2.14**	**1.08**	**14.63**	2.25	**2.12**	**1.01**	**12.68**	**2.23**	2.12	**1.01**	**12.69**
	RMSE	**10.32**	**4.88**	**4.12**	**18.26**	7.55	**4.80**	**4.07**	**16.00**	**7.43**	**4.89**	**4.05**	**15.97**

overall performance across the evaluated datasets, further demonstrating the robustness of unified partial-observation modeling under irregular missing patterns.

5.2 Few-shot and Zero-shot Results

We further evaluate whether the pretrained models can generalize to unseen datasets. As shown in Table 3, we consider both zero-shot transfer and few-shot adaptation with 5% and 10% labeled training data. The results validate three complementary forms of generalization. First, *DCbike* and *PeMSBAY* evaluate cross-region transfer within the urban mobility and traffic domains, respectively, and demonstrate whether the pretrained models can generalize to new geographic locations under similar data semantics. Second, *ERA5prec* evaluates cross-domain transfer at the global scale, where the model must generalize to an unseen meteorological variable. Third, *CMIP6clt* evaluates cross-scale transfer from globally pretrained weather fields to a regional field over North America. Overall, FieldDiff achieves the best zero-shot performance on all four datasets and remains consistently competitive under both 5% and 10% few-shot adaptation. These results show that the proposed canonical latent field transfers effectively across geographic regions, physical variables, and spatial scales. In particular, under zero-shot evaluation, FieldDiff reduces MAE by 4.96% and RMSE by 3.14% on average compared with the best competing baseline across the four datasets. Full results are provided in Appendix G. We further visualize the node-wise mean heatmaps of the generated data. As shown in Figure 3, FieldDiff produces spatial patterns that are more faithful to the ground truth, especially at previously unseen nodes, indicating stronger spatial reasoning and extrapolation ability. This observation is consistent with the quantitative results and further supports the advantage of our field-based spatial modeling.

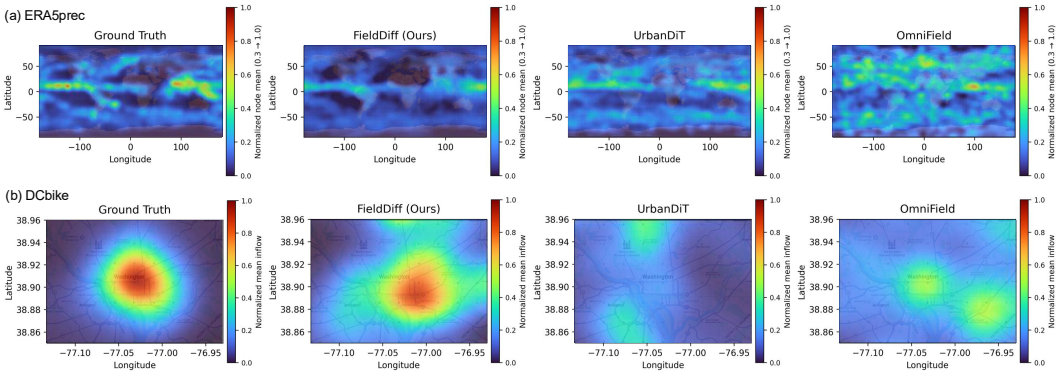

Figure 3: Qualitative comparisons on zero-shot spatial extrapolation performance.

5.3 Model Analysis

Ablation Studies. We assess the contribution of each major component of FieldDiff on *NYCbike* in Figure 4. Removing the global–local coordinate encoding causes the largest degradation (↑21.44%

MAE, ↑33.81% RMSE), underscoring the importance of domain-relative geometry. Disabling adaptive tokenization or the variational bottleneck also leads to consistent performance drops, while removing condition alignment moderately degrades both metrics. Warm-start diffusion is another critical component. Replacing the condition-derived initialization with pure noise increases MAE by 17.33% and RMSE by 13.37%.

Scaling Analysis. Figure 5 evaluates FieldDiff on *NYCbike* under different pretraining data and model sizes. Increasing data from 5% to the full dataset reduces MAE from 15.87 to 9.15 and RMSE from 28.83 to 15.45, demonstrating a clear data scaling effect. Increasing the model size from 13.45M to 20.66M parameters similarly improves MAE from 10.45 to 9.15 and RMSE from 20.03 to 15.45. This consistent improvement is particularly desirable for real-world STFMs, where increasing data availability can continuously enhance model performance.

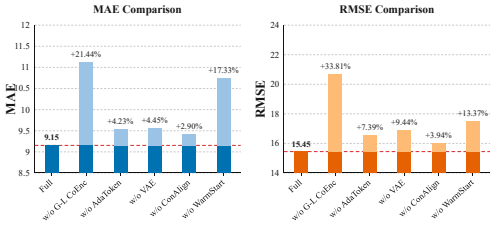

Figure 4: Abaltion studies of each component.

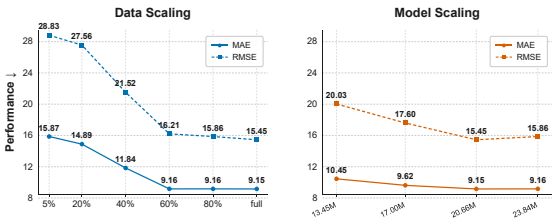

Figure 5: Data and model scaling analysis.

Efficiency Analysis. Figure 6 compares model efficiency on *NYCbike*. FieldDiff contains 20.6M parameters, fewer than OmniField (31.9M), while achieving substantially better prediction accuracy. Compared with UrbanDiT, FieldDiff reduces inference time from 50.68 s to 6.57 s, benefiting from performing diffusion in the compact latent field rather than directly in the observation space.

Geometry Analysis. We evaluate local denoiser smoothness at a representative diffusion noise level of $\sigma = 0.35$ using the finite-difference estimate $\|D_\theta(\boldsymbol{r}_\sigma + \boldsymbol{\delta}, \sigma, \mathbf{C}) - D_\theta(\boldsymbol{r}_\sigma, \sigma, \mathbf{C})\|_2 / \|\boldsymbol{\delta}\|_2$, where $\boldsymbol{\delta}$ denotes a small random perturbation applied to the noisy residual latent. As shown in Figure 7, the latent-space denoiser exhibits lower and more concentrated sensitivity than the data-space denoiser. This indicates smoother local denoising behavior in the learned latent space.

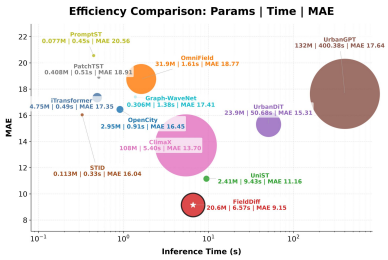

Figure 6: Efficiency comparison.

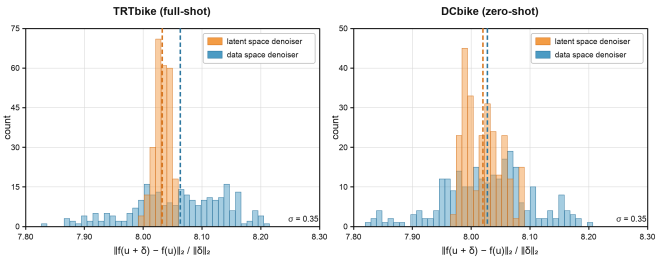

Figure 7: Finite-difference Lipschitz estimates.

6 Conclusion

In this paper, we present FieldDiff, a latent neural field diffusion framework for spatio-temporal foundation modeling. FieldDiff formulates heterogeneous spatio-temporal observations as continuous fields, enabling unified modeling across spatial structures, scales, and downstream tasks with arbitrary-coordinate querying. Our extensive experiments demonstrate strong multi-task performance and robust zero-shot and few-shot generalization across heterogeneous domains and spatial scales. We hope this work advances field-based spatio-temporal foundation modeling and inspires future research toward more generalizable models for complex spatio-temporal systems.

Reproducibility Statement.

All architectural designs, training objectives, and key hyperparameter settings are described in the Implementation Details section, with additional configuration, optimization, and experimental protocol details provided in the appendix to facilitate reproducibility. The project page and source code are available at `https://anonymous.4open.science/r/FieldDiff-C3C3`.

AI use statement

In this work, we used generative AI tools to assist with language editing and writing refinement. The primary use was to improve grammar, sentence structure, readability, and consistency of terminology throughout the manuscript. We did not use generative AI tools to generate the core research ideas, methodology, experimental design, or scientific conclusions of this work. All AI-assisted text was carefully reviewed and revised by the authors to ensure technical correctness, originality, and consistency with the actual methods and results. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020.

Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Wei Chen, Xixuan Hao, Yuankai Wu, and Yuxuan Liang. Terra: A multimodal spatio-temporal dataset spanning the earth. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 66329–66356. Curran Associates, Inc., 2024. doi: 10.52202/079017-2121.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the g_ap_s: Multivariate time series imputation by graph neural networks. In *International Conference on Learning Representations*, 2022.

Yutong Feng, Qiongyan Wang, Yutong Xia, Junlin Huang, Siru Zhong, and Yuxuan Liang. Spatio-temporal field neural networks for air quality inference. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/803.

Sicheng He, Junzhong Ji, and Minglong Lei. Decomposed spatio-temporal mamba for long-term traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11772–11780, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3992–4003, 2023. doi: 10.1109/ICCV51070.2023.00371.

Yushi Lan, Fangzhou Hong, Shangchen Zhou, Shuai Yang, Xuyi Meng, Yongwei Chen, Zhaoyang Lyu, Bo Dai, Xingang Pan, and Chen Change Loy. Ln3diff++: Scalable latent neural fields diffusion for speedy 3d generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3):3002–3017, 2026. doi: 10.1109/TPAMI.2025.3633073.

Mengying Lei, Aurelie Labbe, Yuankai Wu, and Lijun Sun. Bayesian kernelized matrix factorization for spatiotemporal traffic data imputation and kriging. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):18962–18974, 2022. doi: 10.1109/TITS.2022.3161792.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

Yudong Li, Jingyuan Wang, Xie Yu, Peiyu Wang, and Qian Huang. Cross city traffic flow generation via retrieval augmented diffusion model. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 70025–70055. Curran Associates, Inc., 2025a. doi: 10.52202/085713-2354.

Zhonghang Li, Lianghao Xia, Jiabin Tang, Yong Xu, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. Urbangpt: Spatio-temporal large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 5351–5362, 2024.

Zhonghang Li, Long Xia, Lei Shi, Yong Xu, Dawei Yin, and Chao Huang. Open spatio-temporal foundation models for traffic prediction. *ACM Transactions on Intelligent Systems and Technology*, 2025b.

Aoyu Liu and Yaying Zhang. A general spatio-temporal backbone with scalable contextual pattern bank for urban continual forecasting. In *International Conference on Learning Representations*, volume 2026, pp. 72080–72103, 2026.

Chenxi Liu, Kethmi Hirushini Hettige, Qianxiong Xu, Cheng Long, Shili Xiang, Gao Cong, Ziyue Li, and Rui Zhao. St-llm+: Graph enhanced spatio-temporal large language models for traffic prediction. *IEEE Transactions on Knowledge and Data Engineering*, 37(8):4846–4859, 2025. doi: 10.1109/TKDE.2025.3570705.

Xu Liu, Yutong Xia, Yuxuan Liang, Junfeng Hu, Yiwei Wang, LEI BAI, Chao Huang, Zhenguang Liu, Bryan Hooi, and Roger Zimmermann. Largest: A benchmark dataset for large-scale traffic forecasting. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 75354–75371. Curran Associates, Inc., 2023. doi: 10.52202/075280-3293.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 11116–11140, 2024.

Sisuo Lyu, Siru Zhong, Tiegang Chen, Weilin Ruan, Qingxiang Liu, Taiqiang Lv, Qingsong Wen, Raymond Chi-Wing Wong, and Yuxuan Liang. Ts-memory: Plug-and-play memory for time series foundation models. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 3562–3572, 2026.

Tung Nguyen, Johannes Brandstetter, Ashish Kapoor, Jayesh K Gupta, and Aditya Grover. ClimaX: A foundation model for weather and climate. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25904–25938. PMLR, 23–29 Jul 2023.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan LI. Large language diffusion models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 50608–50646. Curran Associates, Inc., 2025. doi: 10.52202/085713-1689.

Tong Nie, Guoyang Qin, Wei Ma, Yuewen Mei, and Jian Sun. Imputeformer: Low rankness-induced transformers for generalizable spatiotemporal imputation. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 2260–2271, 2024.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023.

David Keetae Park, Xihaier Luo, Guang Zhao, Seungjun Lee, Miruna Oprescu, and Shinjae Yoo. SCENT: Robust spatiotemporal learning for continuous scientific data via scalable conditioned neural fields. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48128–48152. PMLR, 13–19 Jul 2025.

Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, pp. 4454–4458, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392365. doi: 10.1145/3511808.3557702.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-moe: Billion-scale time series foundation models with mixture of experts. In *International conference on learning representations*, volume 2025, pp. 34635–34667, 2025.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 7462–7473. Curran Associates, Inc., 2020.

Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 914–921, 2020.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. Csdi: Conditional score-based diffusion models for probabilistic time series imputation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 24804–24816. Curran Associates, Inc., 2021.

Kevin Valencia, Thilina Balasooriya, Xihaier Luo, Shinjae Yoo, and David Park. Omnifield: Conditioned neural fields for robust multimodal spatiotemporal learning. In *International Conference on Learning Representations*, volume 2026, pp. 93840–93866, 2026.

Binwu Wang, Jiaming Ma, Pengkun Wang, Xu Wang, Yudong Zhang, Zhengyang Zhou, and Yang Wang. Stone: A spatio-temporal ood learning framework kills both spatial and temporal shifts. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2948–2959, 2024.

Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12694–12702, 2025.

Shuai Wang, Ziteng Gao, Chenhui Zhu, Weilin Huang, and Limin Wang. Pixnerd: Pixel neural field diffusion. In *International Conference on Learning Representations*, volume 2026, pp. 43559–43580, 2026.

Hao Wu, Yuxuan Liang, Wei Xiong, Zhengyang Zhou, Wei Huang, Shilong Wang, and Kun Wang. Earthfarsser: Versatile spatio-temporal dynamical systems modeling in one model. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 15906–15914, 2024.

Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. Inductive graph neural networks for spatiotemporal kriging. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 4478–4485, 2021.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pp. 1907–1913. AAAI Press, 2019. ISBN 9780999241141.

Weilin Xin, Chenyu Huang, Peilin Li, Jing Zhong, and Jiawei Yao. Urbangraph: Physics-informed spatio-temporal dynamic heterogeneous graphs for urban microclimate prediction. In *International Conference on Learning Representations*, volume 2026, pp. 58856–58889, 2026.

Qianxiong Xu, Cheng Long, Ziyue Li, Sijie Ruan, Rui Zhao, and Zhishuai Li. Kits: Inductive spatio-temporal kriging with increment training strategy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12945–12953, 2025.

Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. Mmada: Multimodal large diffusion language models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 138867–138907. Curran Associates, Inc., 2025. doi: 10.52202/085713-4636.

Yuan Yuan, Jingtao Ding, Jie Feng, Depeng Jin, and Yong Li. Unist: A prompt-empowered universal model for urban spatio-temporal prediction. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 4095–4106, 2024.

Yuan Yuan, Chonghua Han, Jingtao Ding, Guozhen Zhang, Depeng Jin, and Yong Li. Diffusion transformers as open-world spatiotemporal foundation models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 30518–30545. Curran Associates, Inc., 2025. doi: 10.52202/085713-1023.

Zijian Zhang, Xiangyu Zhao, Qidong Liu, Chunxu Zhang, Qian Ma, Wanyu Wang, Hongwei Zhao, Yiqi Wang, and Zitao Liu. Promptst: Prompt-enhanced spatio-temporal multi-attribute prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23, pp. 3195–3205, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701245. doi: 10.1145/3583780.3615016.

Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, Jianzhong Qi, Chaochao Chen, and Longbiao Chen. Increase: Inductive graph representation learning for spatio-temporal kriging. In *Proceedings of the ACM Web Conference 2023*, WWW '23, pp. 673–683, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394161. doi: 10.1145/3543507.3583525.

Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 2267–2276, 2015.

Siru Zhong, Junjie Qiu, Yangyu Wu, Xingchen Zou, Zhongwen Rao, Bin Yang, Chenjuan Guo, Hao Xu, and Yuxuan Liang. Learning to factorize spatio-temporal foundation models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 88232–88260. Curran Associates, Inc., 2025. doi: 10.52202/085713-2949.

Zirui Zhou, Xun Zhou, Kanyu Bao, Shengxin Liu, Kehai Chen, and Min Zhang. Hurst: Heterogeneity-adaptive urban foundation models for spatiotemporal prediction via self-partitional mixture-of-spatial-experts. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 6923–6934, 2026.

A Definitions of Spatio-Temporal Downstream Tasks

Spatio-Temporal Prediction. Given observations of all spatial entities over a historical time interval $\mathcal{T}_{\text{his}} = \{1, \dots, T_h\}$, forecasting aims to predict their observations over the future interval $\mathcal{T}_{\text{fut}} = \{T_h + 1, \dots, T\}$:

$$\mathbf{X}_{:,\mathcal{T}_{\text{his}}} \longrightarrow \mathbf{X}_{:,\mathcal{T}_{\text{fut}}}. \tag{10}$$

Accordingly, all spatial entities are observed in the historical window, while the future time steps are entirely unobserved:

$$M_{n,t} = \begin{cases} 1, & t \in \mathcal{T}_{\text{his}}, \\ 0, & t \in \mathcal{T}_{\text{fut}}. \end{cases} \tag{11}$$

Temporal Interpolation. Temporal interpolation aims to recover observations at unobserved time steps within an observed temporal range. Let $\mathcal{T}_o \subset \mathcal{T}$ and $\mathcal{T}_u = \mathcal{T} \setminus \mathcal{T}_o$ denote the observed and unobserved time sets, respectively. Given observations of all spatial entities at $\mathcal{T}_o$, the task is

$$\mathbf{X}_{:,\mathcal{T}_o} \longrightarrow \mathbf{X}_{:,\mathcal{T}_u}, \tag{12}$$

where

$$M_{n,t} = \begin{cases} 1, & t \in \mathcal{T}_o, \\ 0, & t \in \mathcal{T}_u. \end{cases} \tag{13}$$

In contrast to forecasting, the unobserved time steps lie within the temporal range covered by the observations.

Spatial Extrapolation. Spatial extrapolation infers the temporal dynamics of spatial entities that are entirely unobserved over a given time interval. We partition the spatial entities into an observed set $\mathcal{V}_o$ and an unobserved set $\mathcal{V}_u$, where $\mathcal{V}_o \cap \mathcal{V}_u = \varnothing$ and $\mathcal{V}_o \cup \mathcal{V}_u = \mathcal{V}$. The task is formulated as

$$\mathbf{X}_{\mathcal{V}_o,:} \longrightarrow \mathbf{X}_{\mathcal{V}_u,:}, \tag{14}$$

with the observation mask

$$M_{n,t} = \begin{cases} 1, & n \in \mathcal{V}_o, \\ 0, & n \in \mathcal{V}_u. \end{cases} \tag{15}$$

Unlike temporal interpolation, the missingness is structured along the spatial dimension, requiring the model to generalize the learned spatial field to previously unobserved locations.

Spatio-Temporal Imputation. Spatio-temporal imputation considers irregularly missing observations jointly distributed across the spatial and temporal dimensions. Let

$$\Omega_o = \{(n, t) \mid M_{n,t} = 1\} \tag{16}$$

denote the set of observed spatio-temporal coordinates and $\Omega_u = (\mathcal{V} \times \mathcal{T}) \setminus \Omega_o$ the missing coordinates. The task is to infer

$$\mathbf{X}_{\Omega_o} \longrightarrow \mathbf{X}_{\Omega_u}. \tag{17}$$

Different from the structured missing patterns in forecasting, temporal interpolation, and spatial extrapolation, spatio-temporal imputation allows missing observations to occur irregularly at arbitrary spatial entities and time steps.

B Methodology Details

B.1 Temporal Coordinate Encoding

For each time step t, the temporal coordinate is represented by

$$\boldsymbol{t}_t = [y_t, m_t, d_t, u_t]^\top, \tag{18}$$

where y_t, m_t, d_t, and u_t denote the year, month, day, and hour, respectively. To provide a unified temporal representation across datasets with different calendar periods and sampling intervals, we

apply a deterministic calendar transformation $\rho(\boldsymbol{t}_t) = \boldsymbol{p}_t \in \mathbb{R}^{11}$. For a scalar calendar variable u with period P, we define its periodic encoding as

$$\eta(u; P) = \left[\sin\left(\frac{2\pi u}{P}\right), \cos\left(\frac{2\pi u}{P}\right)\right]^\top. \tag{19}$$

The canonical temporal feature is then constructed as

$$\boldsymbol{p}_t = [\tilde{y}_t; \eta(m_t; P_m); \eta(d_t; P_d); \eta(u_t; P_h); \eta(a_t; P_a); e_t; \log(1+\delta)] \in \mathbb{R}^{11}, \tag{20}$$

where $\tilde{y}_t$ denotes the normalized year, P_m, P_d, P_h, and P_a denote the periods associated with the monthly, daily, hourly, and annual cycles, respectively, a_t denotes the day-of-year index computed while accounting for leap years, e_t denotes the elapsed time within the current input window, and δ denotes the sampling interval. Specifically, let τ_t denote the absolute time associated with $\boldsymbol{t}_t$ expressed in hours. The relative elapsed time is defined in days as

$$e_t = \frac{\tau_t - \tau_1}{24}, \tag{21}$$

while the sampling interval is estimated in hours by

$$\delta = \mathrm{median}_{j=2,\ldots,T} \left|\tau_j - \tau_{j-1}\right|. \tag{22}$$

Finally, the canonical temporal feature is encoded using fixed Gaussian Fourier features and a learnable projection network,

$$\boldsymbol{e}_t^t = h_t\left(\gamma(\boldsymbol{p}_t; \boldsymbol{B}_t)\right), \tag{23}$$

where $\boldsymbol{B}_t$ is a fixed Gaussian Fourier projection matrix, $\gamma(\cdot; \boldsymbol{B}_t)$ denotes the Fourier mapping defined in the main text, and $h_t(\cdot)$ is a learnable multilayer perceptron that maps the encoded features into the temporal embedding space.

B.2 Geometry-Grounded Spatial Tokenization

For domain d, the point-wise spatio-temporal representations introduced in the main text are collected at each time step as $\boldsymbol{H}_t^{(d)} = [\boldsymbol{h}_{1,t}, \ldots, \boldsymbol{h}_{N_d,t}]^\top \in \mathbb{R}^{N_d \times D}$. Rather than constructing local spatial supports directly from raw coordinates, we perform geometric assignment in the encoded spatial representation space. Let $\boldsymbol{E}_s^{(d)} = \left[\boldsymbol{e}_1^s, \ldots, \boldsymbol{e}_{N_d}^s\right]^\top \in \mathbb{R}^{N_d \times D_s}$ denote the spatial embeddings produced by the global–local coordinate encoder, where D_s is the spatial embedding dimension. We introduce K_l learnable geometric prototypes $\{\boldsymbol{p}_k^l\}_{k=1}^{K_l}$ in the same embedding space. The affinity between local prototype k and node n is computed as

$$s_{k,n}^{(d)} = \frac{\left\langle \bar{\boldsymbol{p}}_k^l, \bar{\boldsymbol{e}}_n^s \right\rangle}{\tau}, \tag{24}$$

where $\bar{\boldsymbol{p}}_k^l = \boldsymbol{p}_k^l / \|\boldsymbol{p}_k^l\|_2$ and $\bar{\boldsymbol{e}}_n^s = \boldsymbol{e}_n^s / \|\boldsymbol{e}_n^s\|_2$ denote the normalized prototype and spatial embedding, respectively, and $\tau > 0$ controls the assignment sharpness. The assignment affinity is

$$\tilde{a}_{k,n}^{(d)} = \exp\left(s_{k,n}^{(d)}\right). \tag{25}$$

To avoid degenerate assignments, we apply balanced Sinkhorn normalization to the affinity matrix, yielding $\boldsymbol{A}^{(d)} \in \mathbb{R}^{K_l \times N_d}$. The resulting assignment matrix defines geometry-adaptive spatial supports for the local latent tokens. For each local token, we summarize its assigned spatial support as

$$\boldsymbol{e}_k^{l,(d)} = \frac{\sum_{n=1}^{N_d} A_{k,n}^{(d)} \boldsymbol{e}_n^s}{\sum_{n=1}^{N_d} A_{k,n}^{(d)}}, \tag{26}$$

and construct the corresponding geometry-adaptive query by

$$\boldsymbol{q}_k^{l,(d)} = h_q\left(\boldsymbol{e}_k^{l,(d)}\right), \tag{27}$$

where $h_q(\cdot)$ is a learnable projection network. Collecting all local queries gives $\boldsymbol{Q}_l^{(d)} \in \mathbb{R}^{K_l \times D}$. The complete latent query matrix is $\boldsymbol{Q}^{(d)} = [\boldsymbol{Q}_g; \boldsymbol{Q}_l^{(d)}] \in \mathbb{R}^{K \times D}$.

B.3 Variational Encoding and Value-Anchored Reconstruction

Given the structured field representation, the encoder parameterizes a diagonal Gaussian posterior

$$q_\phi\left(\mathbf{z} \mid \boldsymbol{X}, \boldsymbol{S}, \boldsymbol{T}\right) = \mathcal{N}\left(\mathbf{z}; \boldsymbol{\mu}_\phi, \operatorname{diag}\left(\boldsymbol{\sigma}_\phi^2\right)\right), \tag{28}$$

where $\boldsymbol{\mu}_\phi$ and $\boldsymbol{\sigma}_\phi$ denote the posterior mean and standard deviation, respectively. A latent sample is obtained using the reparameterization trick,

$$\boldsymbol{z} = \boldsymbol{\mu}_\phi + \boldsymbol{\sigma}_\phi \odot \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \tag{29}$$

and is reshaped into the structured latent field $\mathbf{Z} \in \mathbb{R}^{T \times K \times D}$. The posterior is regularized toward an isotropic Gaussian prior through

$$\mathcal{L}_{\mathrm{KL}} = D_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \boldsymbol{X}, \boldsymbol{S}, \boldsymbol{T}) \,\|\, \mathcal{N}(\mathbf{0}, \boldsymbol{I})\right). \tag{30}$$

Together with the field reconstruction objective, the Stage-I loss is

$$\mathcal{L}_{\mathrm{I}} = \mathcal{L}_{\mathrm{rec}} + \beta_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}}, \tag{31}$$

where β_{KL} controls the strength of the variational regularization. We use a small KL weight so that the latent distribution is regularized without substantially degrading reconstruction fidelity.

To improve transfer across domains with different value ranges, each latent token additionally preserves four transferable value anchors: the local mean, log standard deviation, mean absolute deviation, and temporal mean difference. These statistics retain local magnitude information that may otherwise be difficult to transfer across heterogeneous domains. Given an arbitrary spatio-temporal query $(\boldsymbol{s}, \boldsymbol{t})$, the decoder predicts a dimensionless residual and reconstructs the corresponding field value as

$$\hat{x}(\boldsymbol{s}, \boldsymbol{t}) = \mu_a(\boldsymbol{s}, \boldsymbol{t}) + \eta_a(\boldsymbol{s}, \boldsymbol{t})\, r_\psi(\mathbf{Z}, \boldsymbol{s}, \boldsymbol{t}), \tag{32}$$

where $\mu_a(\boldsymbol{s}, \boldsymbol{t})$ and $\eta_a(\boldsymbol{s}, \boldsymbol{t})$ denote the interpolated local value center and scale, respectively, and $r_\psi(\mathbf{Z}, \boldsymbol{s}, \boldsymbol{t})$ is the dimensionless residual predicted by the coordinate-conditioned decoder. The local center and scale are obtained by interpolating the value anchors of neighboring latent tokens according to their spatial proximity. This anchor-affine formulation separates domain-dependent value magnitude from the transferable latent representation, allowing the decoder to reconstruct fields with substantially different value scales using a shared set of parameters.

C Theoretical Analysis of Latent Field Space for Diffusion

We provide a geometric interpretation of why FieldDiff performs diffusion in the learned latent field space rather than directly in the heterogeneous observation space.

Consider a spatio-temporal domain d with $\boldsymbol{X}^{(d)} \in \mathbb{R}^{N_d \times T}$. After vectorization, let

$$\boldsymbol{x}^{(d)} = \operatorname{vec}\left(\boldsymbol{X}^{(d)}\right) \in \mathbb{R}^{N_d T}. \tag{33}$$

Since the number of spatial observations N_d varies across domains, different datasets naturally reside in observation spaces with different ambient dimensions. FieldDiff instead maps every domain into a structured latent field $\mathbf{Z}^{(d)} \in \mathbb{R}^{T \times K \times D}$. Let

$$\boldsymbol{z}^{(d)} = \operatorname{vec}\left(\mathbf{Z}^{(d)}\right) \in \mathbb{R}^{D_z}, \qquad D_z = TKD, \tag{34}$$

where the latent dimension D_z is shared across domains.

For the variational encoder, we use the posterior mean as the deterministic representation analyzed below,

$$\mathcal{E}_\phi^{(d)}\left(\boldsymbol{x}^{(d)}\right) = \mathbb{E}\left[\mathbf{z} \mid \boldsymbol{x}^{(d)}\right] = \boldsymbol{\mu}_\phi\left(\boldsymbol{x}^{(d)}\right). \tag{35}$$

We assume that observations from domain d concentrate around a smooth data manifold $\mathbb{M}_X^{(d)} \subset \mathbb{R}^{N_d T}$, and that $\mathcal{E}_\phi^{(d)}$ is twice continuously differentiable and bi-Lipschitz on this manifold. Specifically, for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{M}_X^{(d)}$,

$$m_d \left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|_2 \le \left\|\mathcal{E}_\phi^{(d)}(\boldsymbol{x}_1) - \mathcal{E}_\phi^{(d)}(\boldsymbol{x}_2)\right\|_2 \le L_d \left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|_2, \tag{36}$$

for constants $0 < m_d \leq L_d < \infty$. This assumption requires the encoder to preserve informative local geometry without collapsing distinct field states or excessively amplifying small perturbations.

Let $p_X^{(d)}$ and $p_Z^{(d)}$ denote the probability densities induced by domain d in the observation and latent spaces, respectively. We characterize their local geometric complexity using the curvature of the corresponding log densities,

$$\kappa_X^{(d)}(\boldsymbol{x}) = \left\| \nabla_{\boldsymbol{x}}^2 \log p_X^{(d)}(\boldsymbol{x}) \right\|_{\mathrm{op}}, \qquad \kappa_Z^{(d)}(\boldsymbol{z}) = \left\| \nabla_{\boldsymbol{z}}^2 \log p_Z^{(d)}(\boldsymbol{z}) \right\|_{\mathrm{op}}, \tag{37}$$

where $\|\cdot\|_{\mathrm{op}}$ denotes the matrix operator norm. A larger value indicates more rapid local variation of the score field and, consequently, more complex local denoising dynamics.

The variational objective further regularizes the latent distribution toward the isotropic Gaussian prior $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. For an isotropic Gaussian distribution, the score is linear and the log-density Hessian is constant and isotropic. The KL regularization therefore encourages a latent density with more regular local geometry. Importantly, this regularization does not by itself guarantee lower score curvature; the geometry of the learned representation map also plays an essential role.

Proposition A.1 (Latent Score Curvature Bound). Suppose the above regularity assumptions hold and the encoder $\mathcal{E}_\phi^{(d)}$ and its inverse on the encoded data manifold have uniformly bounded derivatives up to second order. Assume further that

$$\left\| \nabla_{\boldsymbol{x}} \log p_X^{(d)}(\boldsymbol{x}) \right\|_2 \leq B_d \tag{38}$$

for all $\boldsymbol{x} \in \mathbb{M}_X^{(d)}$. Then there exist finite constants $C_{1,d} > 0$ and $C_{2,d} \geq 0$ such that

$$\kappa_Z^{(d)}(\boldsymbol{z}) \leq C_{1,d} \kappa_X^{(d)}(\boldsymbol{x}) + C_{2,d}, \qquad \boldsymbol{x} = \mathcal{E}_\phi^{(d),-1}(\boldsymbol{z}). \tag{39}$$

Proof. Let $J_{\mathcal{E}}(\boldsymbol{x})$ denote the local volume change induced by $\mathcal{E}_\phi^{(d)}$ on the data manifold. Under the smooth embedding assumption, the change-of-variables relation on the manifold gives

$$p_Z^{(d)}(\boldsymbol{z}) = \frac{p_X^{(d)}(\boldsymbol{x})}{J_{\mathcal{E}}(\boldsymbol{x})}, \qquad \boldsymbol{x} = \mathcal{E}_\phi^{(d),-1}(\boldsymbol{z}). \tag{40}$$

Equivalently,

$$\log p_Z^{(d)}(\boldsymbol{z}) = \log p_X^{(d)}(\boldsymbol{x}) + \log J_{\mathcal{E}^{-1}}(\boldsymbol{z}), \tag{41}$$

where $J_{\mathcal{E}^{-1}}(\boldsymbol{z})$ denotes the corresponding inverse volume correction.

Differentiating with respect to $\boldsymbol{z}$ yields

$$\nabla_{\boldsymbol{z}} \log p_Z^{(d)}(\boldsymbol{z}) = \boldsymbol{J}_{\mathcal{E}^{-1}}(\boldsymbol{z})^\top \nabla_{\boldsymbol{x}} \log p_X^{(d)}(\boldsymbol{x}) + \nabla_{\boldsymbol{z}} \log J_{\mathcal{E}^{-1}}(\boldsymbol{z}), \tag{42}$$

where $\boldsymbol{J}_{\mathcal{E}^{-1}}(\boldsymbol{z})$ is the Jacobian matrix of the inverse representation map.

Differentiating once more, applying the chain rule, and taking operator norms gives

$$\kappa_Z^{(d)}(\boldsymbol{z}) \leq \|\boldsymbol{J}_{\mathcal{E}^{-1}}(\boldsymbol{z})\|_{\mathrm{op}}^2 \, \kappa_X^{(d)}(\boldsymbol{x}) + C_{A,d} \left\| \nabla_{\boldsymbol{x}} \log p_X^{(d)}(\boldsymbol{x}) \right\|_2 + C_{b,d}, \tag{43}$$

where $C_{A,d}$ and $C_{b,d}$ collect bounded second-order terms arising from the inverse representation map and the local volume correction.

By the bi-Lipschitz assumption,

$$\|\boldsymbol{J}_{\mathcal{E}^{-1}}(\boldsymbol{z})\|_{\mathrm{op}} \leq \frac{1}{m_d}. \tag{44}$$

Together with $\|\nabla_{\boldsymbol{x}} \log p_X^{(d)}(\boldsymbol{x})\|_2 \leq B_d$, we obtain

$$\kappa_Z^{(d)}(\boldsymbol{z}) \leq m_d^{-2} \kappa_X^{(d)}(\boldsymbol{x}) + C_{A,d} B_d + C_{b,d}. \tag{45}$$

Therefore, Eq. (39) follows with $C_{1,d} = m_d^{-2}$ and $C_{2,d} = C_{A,d} B_d + C_{b,d}$. □

The proposition does not imply that latent-space curvature is universally smaller than observation-space curvature. Rather, it shows that under a sufficiently regular representation map, the complexity of the latent score field remains controlled by that of the original data distribution together with bounded distortion terms. Combined with the weak variational regularization toward an isotropic prior, this provides a geometric motivation for performing diffusion in the learned latent field space.

D ALGORITHM

Algorithm 1 Two-Stage Training of FieldDiff

Require: Multi-domain datasets $\{\boldsymbol{X}^{(d)}, \boldsymbol{S}^{(d)}, \boldsymbol{T}^{(d)}\}_{d=1}^{D_{\mathrm{dom}}}$, Stage I encoder $\mathcal{E}_\phi$ and decoder $\mathcal{D}_\psi$, condition encoder $\mathcal{E}_c$, prediction head h_c, and latent denoiser D_θ.

Stage I: Neural Field Representation Learning

for each training iteration **do**
  Sample a domain d and a complete field $(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{T})$.
  Encode physical coordinates using the global–local coordinate system.
  Construct K_g global tokens and K_l geometry-adaptive local tokens.
  Aggregate spatial observations by cross-attention and refine the latent field with axial attention.
  Obtain the variational latent field following Eq. (29).
  Reconstruct observations with the value-anchored coordinate decoder in Eq. (32).
  Update Stage I parameters using $\mathcal{L}_{\mathrm{I}}$.
end for
Freeze the Stage I encoder and decoder.

Stage II: Conditional Latent Diffusion

for each training iteration **do**
  Sample a complete field $(\boldsymbol{X}, \boldsymbol{S}, \boldsymbol{T})$ and obtain the posterior-mean latent target $\boldsymbol{z}_0$ from the frozen Stage I encoder.
  Sample a task-specific observation mask $\boldsymbol{M}$ and construct the partial observations.
  Encode the partial observations into the condition latent $\mathbf{C}$.
  Predict the condition-dependent latent center and scale $(\boldsymbol{\mu}_c, \boldsymbol{\sigma}_c) = h_c(\mathbf{C})$.
  Construct the standardized residual $\boldsymbol{r}_0 = (\boldsymbol{z}_0 - \boldsymbol{\mu}_c) \oslash \boldsymbol{\sigma}_c$.
  Sample a noise level σ and perturb the residual according to Eq. (6).
  Predict the clean residual with D_θ conditioned on $\mathbf{C}$.
  Update the Stage II parameters using $\mathcal{L}_{\mathrm{diff}}$.
end for
return trained FieldDiff.

Algorithm 2 Unified Inference of FieldDiff

Require: Partial observations $\boldsymbol{X}_{\mathrm{obs}}$, observation mask $\boldsymbol{M}$, spatial coordinates $\boldsymbol{S}$, temporal coordinates $\boldsymbol{T}$, trained $\mathcal{E}_c$, h_c, D_θ, and frozen decoder $\mathcal{D}_\psi$.

Ensure: Predictions $\hat{\boldsymbol{X}}_{\mathbb{U}}$ at unobserved coordinates.

Encode the partial observations into the condition latent $\mathbf{C}$.
Predict the condition-dependent latent center and scale $(\boldsymbol{\mu}_c, \boldsymbol{\sigma}_c) = h_c(\mathbf{C})$.
Initialize $\boldsymbol{r}_{\sigma_{\mathrm{start}}} \sim \mathcal{N}(\mathbf{0}, \sigma_{\mathrm{start}}^2 \boldsymbol{I})$.
for each reverse diffusion step **do**
  Predict the denoised residual with D_θ conditioned on $\mathbf{C}$ and update the residual state using the reverse solver.
end for
Recover the complete latent following Eq. (9).
Decode the recovered latent at target coordinates $(\boldsymbol{s}_n, \boldsymbol{t}_t)$, $(n, t) \in \mathbb{U}$.
return $\hat{\boldsymbol{X}}_{\mathbb{U}}$.

E EXPERIMENTAL SETTINGS

E.1 DATASETS

Pretraining Datasets. We construct the pretraining spatio-temporal datasets spanning urban mobility, traffic sensing, renewable energy, air quality, and weather. The urban mobility datasets are collected from public bike-sharing and taxi trip records in New York City[1], Chicago[2], and Toronto[3]. The traffic datasets are derived from highway sensor networks in California, including METRLA (Li et al., 2018), PeMS (Song et al., 2020), and LargeST Liu et al. (2023). PVUS[4] contains photovoltaic power measurements collected from geographically distributed solar stations across the United States. AirQuality datasets (Zheng et al., 2015) contain pollutant measurements from monitoring stations distributed across major cities in China. For weather data, we use WeatherBench[5] variables derived from global reanalysis data. Together, these datasets provide substantial diversity in spatial support, physical scale, sampling frequency, and underlying dynamics for multi-domain pretraining. Detailed statistics are summarized in Table 4.

Table 4: Statistics of the datasets used for pretraining.

Dataset	Domain	Nodes	Sample Rate	Time Span	Time Points	Location
NYCbike	Mobility	65	30 min	2024/01/01 - 2024/12/31	17,568	New York City, USA
NYCtaxi	Mobility	65	30 min	2024/01/01 - 2024/12/31	17,568	New York City, USA
CHIbike	Mobility	77	1 h	2024/01/01 - 2024/12/31	8,760	Chicago, USA
CHItaxi	Mobility	77	30 min	2024/01/01 - 2024/12/31	17,568	Chicago, USA
TRTbike	Mobility	60	1 h	2020/01/01 - 2020/12/31	8,760	Toronto, Canada
METRLA	Traffic	207	5 min	2012/03/01 - 2012/06/27	34,272	Los Angeles, USA
PEMS4flow	Traffic	3,562	5 min	2018/01/01 - 2018/02/28	16,992	California District 4, USA
PEMS4occupancy	Traffic	3,562	5 min	2018/01/01 - 2018/02/28	16,992	California District 4, USA
PEMS4speed	Traffic	2,414	5 min	2018/01/01 - 2018/02/28	16,992	California District 4, USA
PEMS8flow	Traffic	1,920	5 min	2016/07/01 - 2016/08/31	17,856	California District 8, USA
PEMS8occupancy	Traffic	1,920	5 min	2016/07/01 - 2016/08/31	17,856	California District 8, USA
PEMS8speed	Traffic	1,920	5 min	2016/07/01 - 2016/08/31	17,856	California District 8, USA
GLA	Traffic	3,834	15 min	2021/10/01 - 2021/12/31	8,832	Greater Los Angeles, USA
GBA	Traffic	2,352	15 min	2021/10/01 - 2021/12/31	8,832	Greater Bay Area, USA
PVUS	Energy	5,016	10 min	2006/10/01 - 2006/12/31	13,248	United States
AQIpm25	Environment	437	1 h	2014/05/01 - 2015/04/30	8,760	43 cities, China
AQIpm10	Environment	437	1 h	2014/05/01 - 2015/04/30	8,760	43 cities, China
AQIno2	Environment	437	1 h	2014/05/01 - 2015/04/30	8,760	43 cities, China
ERA5temp	Weather	2,048	1 h	2018/01/01 - 2018/12/31	8,760	Global
ERA5humid	Weather	2,048	1 h	2018/01/01 - 2018/12/31	8,760	Global

Generalization Evaluation Datasets. We further construct a fully held-out evaluation suite to assess whether FieldDiff can transfer beyond the domains observed during pre-training. We consider three complementary forms of generalization. For *cross-region generalization*, bike flow in Washington, D.C. (Li et al., 2025a), PeMSBAY (Li et al., 2018), and SD in LargeST (Liu et al., 2023) evaluate transfer to unseen cities or regions while retaining data semantics similar to those observed during pretraining. For *cross-domain generalization*, O3 and CO in AirQuality (Zheng et al., 2015), and precipitation in WeatherBench introduce unseen physical variables over spatial range related to the pretraining domains. Finally, for *cross-scale generalization*, cloud cover percentage and sea level pressure in CMIP6 (Nguyen et al., 2023) evaluate whether representations learned from global meteorological fields can transfer to regional fields over North America. Detailed statistics are summarized in Table 5.

E.2 IMPLEMENTATION DETAILS

All models are implemented in PyTorch and trained on a single NVIDIA A800 GPU. FieldDiff is trained in two stages. Stage I learns the neural-field representation, after which the encoder and decoder are frozen and Stage II optimizes the conditional latent diffusion model. For each source

[1] https://www.nyc.gov

[2] https://www.chicago.gov

[3] https://ckan0.cf.opendata.inter.prod-toronto.ca

[4] https://www.nrel.gov/grid/solar-power-data.html

[5] https://github.com/pangeo-data/WeatherBench

Table 5: Statistics of the datasets used for generalization evaluation.

Dataset	Domain	Nodes	Sample Rate	Time Span	Time Points	Location
DCbike	Mobility	81	1 h	2023/01/01 - 2023/12/31	8,760	Washington, D.C., USA
PeMSBAY	Traffic	325	5 min	2017/01/01 - 2017/06/30	52,128	San Francisco Bay Area, USA
SD	Traffic	716	15 min	2021/10/01 - 2021/12/31	8,832	San Diego, USA
AQIo3	Environment	437	1 h	2014/05/01 - 2015/04/30	8,760	43 cities, China
AQIco	Environment	437	1 h	2014/05/01 - 2015/04/30	8,760	43 cities, China
ERA5prec	Weather	2,048	1 h	2018/01/01 - 2018/12/31	8,760	Global
CMIP6clt	Climate	1,161	1 month	1850/01/16 - 2014/12/16	1,980	North America
CMIP6psl	Climate	1,161	1 month	1850/01/16 - 2014/12/16	1,980	North America

Table 6: Key hyperparameter settings of FieldDiff.

Hyperparameter	Value
Input window length	24
Latent field size	$24 \times 48 \times 128$
Global / local tokens	16 / 32
Stage-I hidden dimension	128
Stage-I axial layers / heads	4 / 4
Spatial tokenizer layers	2
Sinkhorn temperature / iterations	0.25 / 4
Global / local Fourier scales	2 / 6
KL weight β_{KL}	1×10^{-4}
Stage-II hidden dimension	256
Stage-II denoising blocks / heads	8 / 8
σ_{data}	0.5
Training noise $P_{\mathrm{mean}}/P_{\mathrm{std}}$	-1.2 / 1.2
Sampling start σ_{start}	2.0
Sampling minimum σ_{min}	0.002
Sampling steps	12
Karras ρ	7
Sampling solver	Heun
Optimizer	AdamW
Learning rate	3×10^{-4}
Weight decay	1×10^{-4}
Gradient clipping	1.0
Condition dropout	0.1

dataset, the first temporal 70% of the observations are used for training and validation, while the remaining 30% are reserved for testing. The model is jointly trained for forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation under the unified partial-observation formulation. At inference, latent sampling is initialized from the condition-derived estimate and performed with a Karras noise schedule and a second-order Heun solver. For few-shot adaptation, Stage I remains frozen and only the Stage II diffusion module is updated. Detailed architectural and optimization hyperparameters are summarized in Table 6.

E.3 Evaluation Metrics

We adopt two commonly used error metrics, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), to measure the field reconstruction accuracy across all spatio-temporal tasks. Let $\mathbb{U}$ denote the set of target spatio-temporal coordinates, with ground-truth values $\{x_{n,t}\}_{(n,t)\in\mathbb{U}}$ and model outputs $\{\hat{x}_{n,t}\}_{(n,t)\in\mathbb{U}}$. The metrics are defined as

$$\mathrm{MAE} = \frac{1}{|\mathbb{U}|} \sum_{(n,t)\in\mathbb{U}} |x_{n,t} - \hat{x}_{n,t}|, \qquad \mathrm{RMSE} = \sqrt{\frac{1}{|\mathbb{U}|} \sum_{(n,t)\in\mathbb{U}} (x_{n,t} - \hat{x}_{n,t})^2}. \tag{46}$$

Lower values indicate smaller reconstruction errors. These metrics are consistently applied to the target spatio-temporal coordinates of forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation.

E.4 BASELINES

All baseline methods are evaluated under a unified experimental setting. We use the official implementations from the original papers or repositories. Detailed descriptions for each baseline category are provided below.

E.4.1 SPATIO-TEMPORAL SINGLE-TASK MODELS

Single-task baselines are trained independently for each dataset and each downstream task. For each setting, the model is optimized only on the corresponding training set and evaluated on the testing set of the same dataset.

Prediction Models. We include five representative models for spatio-temporal prediction. For all prediction baselines, the historical observation window and prediction horizon are kept identical to those used by FieldDiff, and the models are trained separately on each dataset.

- **Graph WaveNet** (Wu et al., 2019) combines adaptive graph structure learning with stacked dilated temporal convolutions. The learned dependency matrix captures latent spatial relationships, while the dilated convolutional backbone models long-range temporal dependencies.
- **PatchTST** (Nie et al., 2023) segments each time series into temporal patches and processes them with a channel-independent Transformer. This design preserves local temporal patterns while enabling efficient modeling of long historical contexts.
- **PromptST** (Zhang et al., 2023) employs a shared spatio-temporal Transformer to capture common patterns across multiple attributes, together with prompt tuning for lightweight attribute-specific adaptation.
- **STID** (Shao et al., 2022) augments a lightweight MLP architecture with explicit spatial and temporal identity embeddings, improving the distinguishability of observations across locations and time steps.
- **iTransformer** (Liu et al., 2024) treats individual variables as tokens and their temporal trajectories as token features. Self-attention captures inter-variable dependencies, while feed-forward layers learn nonlinear representations for each variable.

Interpolation Models. For temporal interpolation and spatio-temporal imputation, we adopt the same set of imputation baselines, as both tasks recover missing observations and mainly differ in the structure of their missing patterns: temporal interpolation follows structured temporal masks, whereas ST imputation considers general spatio-temporal missingness.

- **BRITS** (Cao et al., 2018) performs missing-value imputation with a bidirectional recurrent dynamical system. It directly treats missing values as variables within the recurrent computation, enabling information propagation from both temporal directions.
- **CSDI** (Tashiro et al., 2021) formulates time-series imputation as conditional score-based generation. It trains a diffusion model conditioned on observed values to recover missing entries while exploiting correlations among observations.
- **GRIN** (Cini et al., 2022) incorporates graph neural networks into multivariate time-series imputation to explicitly model relational structure among spatial variables. Missing values are reconstructed through spatio-temporal message passing over the underlying graph.
- **ImputeFormer** (Nie et al., 2024) combines Transformer-based representation learning with a low-rank inductive bias for spatio-temporal imputation. The low-rank structure captures global regularities while the Transformer models complex spatio-temporal dependencies.

Extrapolation Models. Unlike temporal interpolation and spatio-temporal imputation, spatial extrapolation requires inferring signals at previously unseen spatial nodes rather than recovering missing values on an observed spatial support.

- **BKMF** (Lei et al., 2022) formulates spatio-temporal kriging through Bayesian kernelized matrix factorization. Gaussian-process priors regularize the latent factors, while a fully Bayesian inference scheme jointly estimates latent representations and kernel hyperparameters.

- **IGNNK** (Wu et al., 2021) introduces an inductive graph neural network for recovering signals at unsampled sensors. By training on randomly generated subgraphs, it learns transferable spatial message-passing patterns that can be applied to unseen nodes.
- **INCREASE** (Zheng et al., 2023) models multiple spatial relations between observed and unobserved locations, including proximity, functional similarity, and transition patterns. Relation-aware recurrent modeling and attention are then used to aggregate spatio-temporal information for inductive kriging.
- **KITS** (Xu et al., 2025) reduces the train–test graph discrepancy by introducing virtual nodes during training. It transfers information from similar observed nodes and constructs pseudo labels for the virtual nodes, enabling the learned patterns to generalize to truly unseen locations.

E.4.2 SPATIO-TEMPORAL PRETRAINED MODELS

For a fair comparison, all pretrained baselines are trained and evaluated on the same pretraining and evaluation datasets used by FieldDiff, whenever their architectures and task formulations permit. Among these methods, OmniField and UrbanDiT support multiple spatio-temporal tasks, while the remaining baselines are evaluated on forecasting only.

- **ClimaX** (Nguyen et al., 2023) is a Transformer-based foundation model for weather and climate modeling. It introduces variable encoding and aggregation modules to accommodate heterogeneous meteorological variables and spatio-temporal coverage.
- **OpenCity** (Li et al., 2025b) combines Transformer and graph neural network architectures to capture transferable spatio-temporal dependencies across heterogeneous urban traffic datasets. Its large-scale pretraining enables zero-shot generalization to unseen urban forecasting scenarios.
- **UrbanGPT** (Li et al., 2024) integrates a spatio-temporal dependency encoder with instruction-tuned large language models, enabling LLMs to incorporate structured spatial and temporal information for urban forecasting.
- **UniST** (Yuan et al., 2024) pre-trains a universal urban spatio-temporal prediction model on diverse scenarios and introduces knowledge-guided prompts to improve transferability across datasets and urban environments.
- **OmniField** (Valencia et al., 2026) learns a continuous neural field conditioned on multimodal scientific observations and uses iterative cross-modal refinement to align heterogeneous signals. It supports multiple spatio-temporal tasks, including reconstruction, interpolation, and prediction.
- **UrbanDiT** (Yuan et al., 2025) scales diffusion Transformers to heterogeneous urban spatio-temporal data and introduces data- and task-specific prompts for unified multi-dataset and multi-task learning. It supports forecasting, temporal interpolation, spatial extrapolation, and spatio-temporal imputation within a single pretrained model.

F ADDITIONAL ONE-FOR-ALL RESULTS

Table 7: One-for-all temporal interpolation performance comparison. Best results are in **bold** and second-best are underlined.

Method	Metric	*NYCbike*	*NYCtaxi*	*TRTbike*	*METRLA*	*PeMS04flow*	*PeMS08flow*	*GBA*	*GLA*	*PVUS*	*AQIpm25*	*ERA5temp*
BRITS	MAE	17.34	23.68	4.33	10.50	116.35	111.28	76.68	91.71	4.50	34.44	7.49
	RMSE	33.32	37.13	11.11	16.58	154.10	147.45	103.83	122.57	11.76	53.23	10.91
CSDI	MAE	7.14	6.41	0.97	3.39	23.73	16.38	21.86	22.04	1.42	13.61	0.77
	RMSE	13.50	10.99	3.07	7.93	34.83	24.86	32.49	35.10	3.86	28.13	1.25
GRIN	MAE	6.88	6.90	0.71	4.00	18.31	16.09	12.31	**14.16**	0.96	13.26	0.75
	RMSE	12.65	10.86	2.07	7.52	**29.10**	27.51	21.52	26.00	3.60	26.75	1.22
ImputeFormer	MAE	7.56	8.19	0.72	3.33	18.92	15.50	12.78	16.18	0.97	12.66	0.73
	RMSE	13.05	12.41	2.00	7.90	30.09	23.62	21.43	27.99	3.23	27.13	1.27
OmniField	MAE	6.21	6.22	1.14	5.70	26.85	24.24	59.48	115.21	5.13	23.22	14.96
	RMSE	10.35	10.28	2.93	10.94	43.20	39.48	86.62	151.98	11.74	36.50	18.36
UrbanDiT	MAE	6.63	6.20	0.75	3.26	20.73	15.53	13.38	14.86	0.92	15.30	0.73
	RMSE	11.31	9.87	2.12	**7.39**	33.44	24.13	21.94	25.56	2.81	30.53	1.26
FieldDiff	MAE	**5.50**	**5.63**	**0.66**	**3.25**	**18.07**	**15.20**	**12.30**	14.84	**0.86**	**12.51**	**0.71**
	RMSE	**9.02**	**9.32**	**1.88**	7.43	30.22	**23.44**	**20.84**	**25.01**	**2.78**	**26.36**	**1.16**

Table 8: One-for-all spatio-temporal imputation performance comparison. Best results are in **bold** and second-best are underlined.

Method	Metric	*NYCbike*	*NYCtaxi*	*TRTbike*	*METRLA*	*PeMS04flow*	*PeMS08flow*	*GBA*	*GLA*	*PVUS*	*AQIpm25*	*ERA5temp*
BRITS	MAE	15.56	22.92	3.69	10.18	114.80	106.01	65.68	83.35	4.87	29.05	6.60
	RMSE	30.62	36.08	10.02	16.05	152.12	141.18	89.51	113.49	11.62	45.01	9.31
CSDI	MAE	6.44	6.16	0.83	3.63	21.17	16.49	21.43	22.99	1.14	13.87	1.66
	RMSE	12.33	10.81	2.80	6.28	35.13	26.57	32.82	33.36	3.26	28.76	3.17
GRIN	MAE	8.90	7.90	1.21	3.69	20.32	17.02	12.53	14.40	1.15	13.60	1.32
	RMSE	17.54	12.54	3.39	6.08	35.88	27.60	20.99	26.90	3.58	32.28	2.61
ImputeFormer	MAE	9.65	12.39	2.01	3.58	20.27	16.98	13.12	14.84	1.02	12.82	1.52
	RMSE	20.43	19.65	6.88	6.19	35.72	27.33	20.65	26.19	3.11	26.59	2.92
OmniField	MAE	5.71	6.00	1.04	5.50	26.83	24.18	59.52	114.86	5.14	22.67	14.94
	RMSE	9.23	10.00	2.53	10.26	43.09	39.40	86.80	151.79	11.76	35.56	18.35
UrbanDiT	MAE	6.33	6.37	0.76	3.28	22.44	16.86	14.79	15.93	1.19	14.75	1.30
	RMSE	10.35	10.13	1.82	6.65	36.22	26.73	25.75	28.44	3.66	27.69	2.82
FieldDiff	MAE	**5.33**	**5.53**	**0.63**	**3.12**	**19.86**	**15.01**	**12.48**	**14.09**	**0.97**	**12.80**	**1.08**
	RMSE	**8.48**	**9.12**	**1.73**	**5.65**	**34.05**	**23.50**	**20.04**	**25.25**	**2.99**	**26.26**	**2.13**

G ADDITIONAL FEW-SHOT AND ZERO-SHOT RESULTS

Table 9: Zero-shot and few-shot prediction performance comparison on all transfer datasets. Best results are in **bold** and second-best are underlined.

Setting	Method	Metric	*AQIco*	*AQIo3*	*DCbike*	*CMIP6psl*	*CMIP6clt*	*PeMSBAY*	*ERA5prec*	*SD*
Zero-shot	ClimaX	MAE	0.46	16.47	3.69	318.57	15.76	2.29	1.10	39.10
		RMSE	0.83	22.98	11.25	428.38	19.15	5.73	4.67	57.53
	OpenCity	MAE	0.46	17.44	3.18	466.10	17.66	2.25	1.11	40.27
		RMSE	0.88	26.11	10.58	628.36	23.06	5.22	4.15	62.29
	UrbanGPT	MAE	0.54	24.40	4.30	321.66	15.48	3.35	1.58	94.06
		RMSE	0.90	30.36	11.11	427.66	19.94	6.06	4.88	117.25
	UniST	MAE	0.47	16.08	3.39	324.58	15.37	2.22	1.10	42.13
		RMSE	0.89	21.99	11.03	434.10	19.93	5.52	4.82	67.74
	OmniField	MAE	0.60	30.79	3.03	385.30	15.50	5.55	1.66	124.49
		RMSE	0.93	38.81	11.07	489.08	19.42	9.53	4.25	159.98
	UrbanDiT	MAE	0.53	20.33	3.33	390.38	15.48	2.62	1.48	38.03
		RMSE	0.95	29.88	10.39	509.53	19.75	5.91	4.26	59.84
	FieldDiff	MAE	**0.45**	**14.44**	**2.76**	**311.63**	**14.63**	**2.14**	**1.08**	**36.17**
		RMSE	**0.81**	**21.69**	**10.32**	**415.82**	**18.26**	**4.88**	**4.12**	**56.08**
5% Few-shot	ClimaX	MAE	0.47	17.65	2.44	287.88	13.04	2.24	1.06	34.01
		RMSE	0.85	23.36	8.05	389.08	16.94	5.65	4.57	51.77
	OpenCity	MAE	0.45	15.35	2.75	245.64	13.90	2.14	1.06	34.88
		RMSE	0.84	22.72	8.51	342.89	17.39	5.09	4.15	56.99
	UrbanGPT	MAE	0.59	25.43	2.33	342.21	13.52	3.05	1.07	97.52
		RMSE	0.97	30.76	8.80	445.98	16.19	5.80	4.18	133.42
	UniST	MAE	0.47	15.11	**2.23**	245.19	13.12	2.22	1.07	32.13
		RMSE	0.89	22.40	8.10	339.59	16.54	5.52	4.18	50.74
	OmniField	MAE	0.55	19.46	2.60	240.53	12.99	4.00	1.48	49.47
		RMSE	0.88	26.33	7.87	336.04	16.37	7.10	4.17	72.10
	UrbanDiT	MAE	0.53	17.54	2.79	327.57	13.29	2.52	1.37	33.00
		RMSE	0.95	25.77	**7.36**	436.64	16.74	5.70	4.35	52.04
	FieldDiff	MAE	**0.44**	**14.29**	2.25	**224.51**	**12.68**	**2.12**	**1.01**	**31.38**
		RMSE	**0.81**	**21.99**	7.55	**329.94**	**16.00**	**4.80**	**4.07**	**49.70**
10% Few-shot	ClimaX	MAE	0.47	17.96	2.27	292.77	12.95	2.27	1.07	34.69
		RMSE	0.85	23.59	7.87	393.72	16.84	5.66	4.56	52.55
	OpenCity	MAE	0.46	15.94	2.83	252.94	13.83	**2.11**	1.05	34.64
		RMSE	0.84	23.59	8.44	353.82	17.29	5.08	4.12	58.25
	UrbanGPT	MAE	0.60	23.20	2.36	338.98	13.32	3.00	1.04	93.93
		RMSE	1.00	28.39	8.24	439.10	17.05	5.87	4.13	123.74
	UniST	MAE	0.47	15.09	2.38	246.12	13.05	2.15	1.09	32.68
		RMSE	0.82	21.68	8.07	339.87	16.50	5.42	4.10	50.93
	OmniField	MAE	0.55	18.66	2.48	241.45	13.04	3.96	1.48	51.70
		RMSE	0.88	25.34	7.55	327.47	16.45	7.08	4.17	74.31
	UrbanDiT	MAE	0.53	17.37	2.82	328.30	13.46	2.52	1.38	32.90
		RMSE	0.95	25.49	7.51	437.28	16.97	5.69	4.35	51.91
	FieldDiff	MAE	**0.44**	**14.21**	**2.23**	**222.42**	**12.69**	2.12	**1.01**	**31.52**
		RMSE	**0.80**	**21.10**	**7.43**	**326.43**	**15.97**	**4.89**	**4.05**	**49.93**

Table 10: Zero-shot and few-shot temporal interpolation performance comparison on all transfer datasets. Best results are in **bold** and second-best are underlined.

Setting	Method	Metric	*AQIco*	*AQIo3*	*DCbike*	*CMIP6psl*	*CMIP6clt*	*PeMSBAY*	*ERA5prec*	*SD*
Zero-shot	OmniField	MAE	0.50	24.25	4.97	393.07	15.62	5.42	1.78	104.82
		RMSE	0.80	32.51	12.03	496.58	19.78	9.15	4.33	142.57
	UrbanDiT	MAE	0.48	21.35	2.26	315.85	14.82	2.14	0.65	34.53
		RMSE	0.79	28.76	4.41	430.77	18.27	4.23	2.68	53.83
	FieldDiff	MAE	**0.44**	**18.99**	**1.73**	**312.16**	**13.90**	**1.99**	**0.60**	**33.43**
		RMSE	**0.74**	**26.76**	**3.88**	**418.18**	**17.37**	**2.84**	**2.07**	**51.43**
5% Few-shot	OmniField	MAE	0.46	18.27	1.56	255.29	14.28	3.82	1.50	43.75
		RMSE	0.75	25.28	4.36	346.81	18.80	6.83	4.23	65.27
	UrbanDiT	MAE	0.47	20.01	1.01	298.20	13.80	2.10	0.63	34.11
		RMSE	0.70	26.67	3.40	410.57	17.90	4.14	2.62	53.11
	FieldDiff	MAE	**0.43**	**17.49**	**0.96**	**221.18**	**12.88**	**1.71**	**0.55**	**29.82**
		RMSE	**0.63**	**24.77**	**3.29**	**327.13**	**16.19**	**2.85**	**2.11**	**47.14**
10% Few-shot	OmniField	MAE	0.46	18.16	1.54	255.06	14.51	3.83	1.51	43.73
		RMSE	0.74	25.16	4.22	346.71	18.12	6.84	4.23	65.08
	UrbanDiT	MAE	0.47	19.99	1.01	296.69	13.80	2.10	0.63	34.07
		RMSE	0.70	26.67	3.39	408.73	17.90	4.14	2.62	53.15
	FieldDiff	MAE	**0.43**	**17.48**	**0.96**	**218.63**	**12.87**	**1.71**	**0.57**	**30.31**
		RMSE	**0.63**	**24.77**	**3.34**	**324.15**	**16.17**	**2.88**	**2.09**	**47.80**

Table 11: Zero-shot and few-shot spatial extrapolation performance comparison on all transfer datasets. Best results are in **bold** and second-best are underlined.

Setting	Method	Metric	*AQIco*	*AQIo3*	*DCbike*	*CMIP6psl*	*CMIP6clt*	*PeMSBAY*	*ERA5prec*	*SD*
Zero-shot	OmniField	MAE	0.51	23.92	5.13	378.43	15.40	5.56	1.81	104.47
		RMSE	0.80	32.22	14.11	482.70	19.44	9.33	4.37	142.30
	UrbanDiT	MAE	0.55	20.42	5.69	249.81	9.47	6.33	1.73	90.49
		RMSE	0.90	29.61	15.70	358.67	12.83	11.36	4.59	134.89
	FieldDiff	MAE	**0.44**	**18.38**	**5.01**	**215.51**	**9.76**	**5.04**	**1.19**	**83.42**
		RMSE	**0.73**	**25.87**	**13.98**	**316.10**	**12.26**	**8.89**	**4.15**	**121.55**
5% Few-shot	OmniField	MAE	0.47	17.87	4.80	231.90	9.09	5.10	1.50	94.55
		RMSE	0.75	24.77	14.98	316.08	11.51	9.31	4.24	136.22
	UrbanDiT	MAE	0.52	18.23	4.75	197.12	6.48	5.81	1.63	85.34
		RMSE	0.87	26.46	14.73	308.24	8.94	10.74	4.76	129.89
	FieldDiff	MAE	**0.43**	**17.45**	**3.82**	**187.84**	**6.42**	**4.75**	**1.17**	**80.37**
		RMSE	**0.72**	**24.65**	**11.02**	**283.58**	**8.75**	**8.88**	**4.19**	**108.15**
10% Few-shot	OmniField	MAE	0.46	17.80	4.79	231.45	9.07	5.09	1.51	94.55
		RMSE	0.74	24.80	14.88	315.63	11.48	9.26	4.24	136.01
	UrbanDiT	MAE	0.52	18.24	4.73	198.14	6.49	5.83	1.62	86.42
		RMSE	0.87	26.47	14.53	309.95	8.93	10.71	4.74	121.53
	FieldDiff	MAE	**0.43**	**17.49**	**3.79**	**189.60**	**6.37**	**4.76**	**1.15**	**80.71**
		RMSE	**0.72**	**24.73**	**11.14**	**284.95**	**8.68**	**8.90**	**4.17**	**108.60**

Table 12: Zero-shot and few-shot spatio-temporal imputation performance comparison on all transfer datasets. Best results are in **bold** and second-best are underlined.

Setting	Method	Metric	*AQIco*	*AQIo3*	*DCbike*	*CMIP6psl*	*CMIP6clt*	*PeMSBAY*	*ERA5prec*	*SD*
Zero-shot	OmniField	MAE	0.50	23.60	5.00	384.29	15.14	5.42	1.79	104.05
		RMSE	0.79	31.75	11.95	488.84	19.10	9.17	4.36	141.64
	UrbanDiT	MAE	0.49	21.07	2.73	335.24	14.64	2.47	0.85	36.76
		RMSE	0.79	27.77	4.25	434.93	18.77	4.02	3.08	60.08
	FieldDiff	MAE	**0.43**	**17.91**	**1.73**	**312.41**	**13.36**	**1.99**	**0.81**	**32.77**
		RMSE	**0.72**	**25.51**	**3.62**	**410.05**	**16.76**	**2.85**	**3.01**	**50.58**
5% Few-shot	OmniField	MAE	0.46	17.52	1.42	230.37	9.06	3.83	1.50	43.18
		RMSE	0.73	24.41	3.60	314.34	11.47	6.87	4.25	64.46
	UrbanDiT	MAE	0.49	20.26	1.28	220.50	6.95	2.40	0.82	35.19
		RMSE	0.79	26.53	3.74	307.23	9.28	3.88	3.03	56.80
	FieldDiff	MAE	**0.42**	**17.03**	**0.97**	**197.02**	**6.34**	**1.73**	**0.77**	**31.16**
		RMSE	**0.71**	**24.34**	**3.10**	**281.17**	**8.63**	**2.82**	**2.83**	**50.41**
10% Few-shot	OmniField	MAE	0.45	17.45	1.40	229.92	9.05	3.84	1.51	43.21
		RMSE	0.73	24.45	3.45	313.70	11.47	6.87	4.26	64.31
	UrbanDiT	MAE	0.49	20.25	1.28	241.28	6.94	2.41	0.81	35.13
		RMSE	0.79	26.54	3.69	308.62	9.27	3.89	3.03	56.84
	FieldDiff	MAE	**0.42**	**17.04**	**0.98**	**198.59**	**6.30**	**1.73**	**0.76**	**29.80**
		RMSE	**0.71**	**24.37**	**3.24**	**282.60**	**8.59**	**2.85**	**2.81**	**50.24**

H ADDITIONAL ABLATION STUDY RESULTS

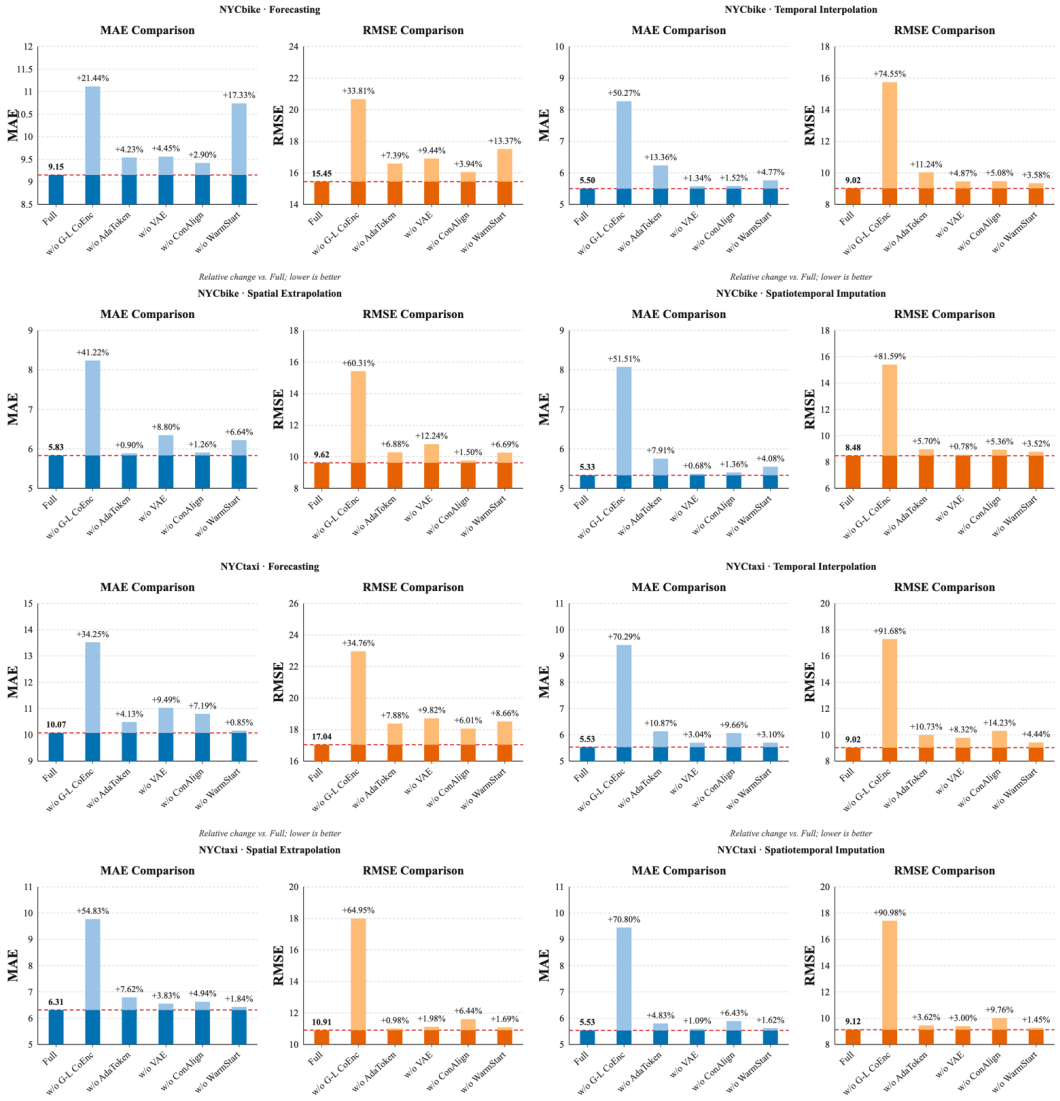

Figure 8: Ablation studies of each component on multi-tasks.

I ADDITIONAL SCALING RESULTS

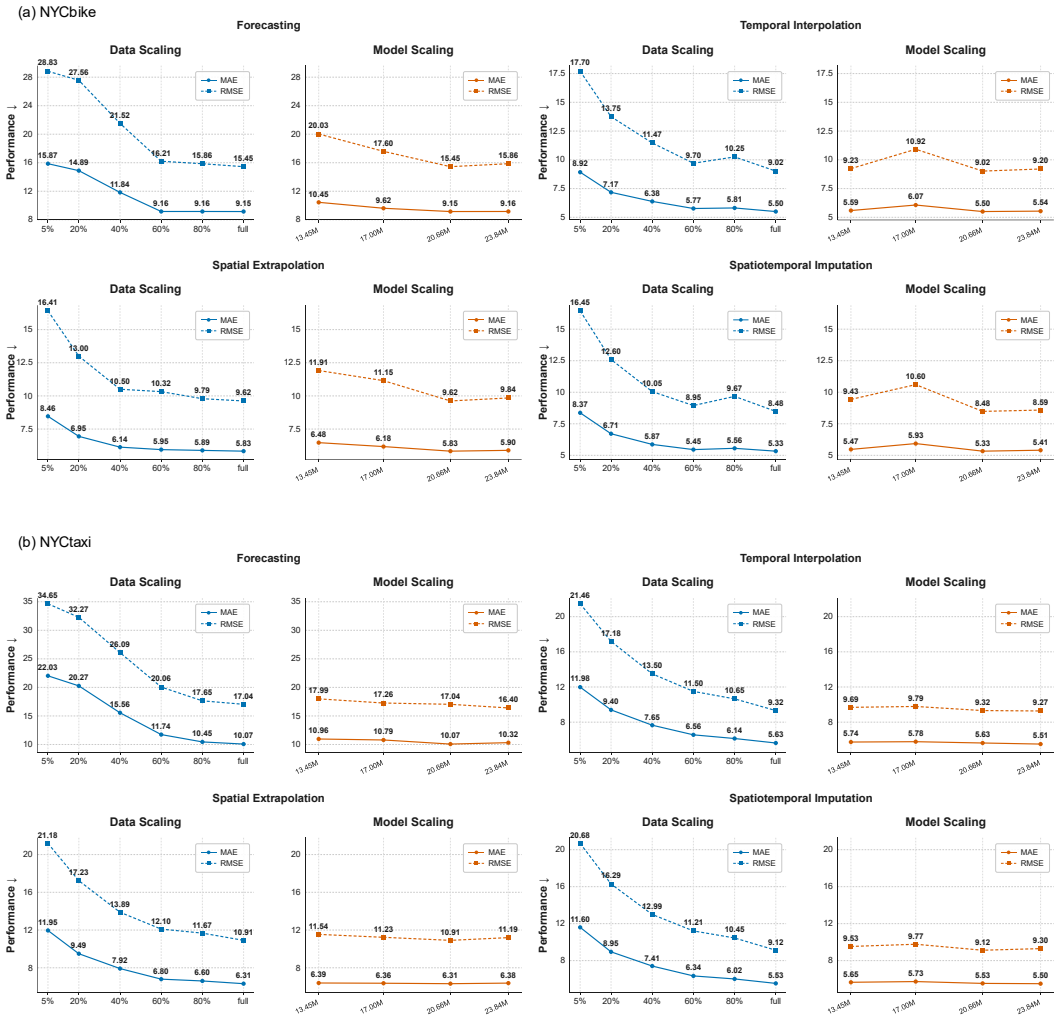

Figure 9: Pretraining data scaling and model size scaling analysis on multi-tasks.