Physics-Inspired Decomposition for Weak Low-Rank Spatiotemporal Completion in Sparse Crowdsensing

Mijia Zhang[1], En Wang[1,*], Wenbin Liu[1,*], Junwei Zhao[2], Yang Xu[3], Bo Yang[1], Jie Wu[2,4]
[1]*College of Computer Science and Technology*, *Jilin University*, China
[2]*China Telecom Cloud Computing Research Institute*, China
[3]*College of Computer Science and Electronic Engineering*, *Hunan University*, China
[4]*Temple University*, United States
Email: {zhangmijia, wangen, liuwenbin}@jlu.edu.cn, jwz@pku.edu.cn, xuyangcs@hnu.edu.cn, ybo@jlu.edu.cn, jiewu@temple.edu

***Abstract*—Sparse crowdsensing is a cost-effective paradigm for large-scale monitoring, where data is collected sparsely from distributed participants with Internet of Things (IoT) devices but suffers from missing entries due to uneven coverage. Existing completion methods, particularly matrix factorization and deep learning models, assume strong low-rank structures, which may fail in heterogeneous or weakly structured real-world data. To address this limitation, we propose an interpretable spatiotemporal completion framework named Matrix Completion by Wave and Diffusion Decomposition (MCWDD), which leverages partial differential equations (PDEs) to decouple spatiotemporal matrices into three components: wave, diffusion, and residual. Each component is analytically regularized to exhibit low-rank properties, enabling robust completion even for complex data structures. Our approach is designed as a general-purpose completion framework, capable of adapting to diverse data patterns beyond conventional low-rank assumptions. Its spatiotemporal pattern not only adheres to underlying physical constraints but also manifests strong low-rank properties. PDE-based models act as priors, avoiding overfitting risks in data-driven methods under limited training conditions. MCWDD retains some data-driven structures to compensate for modeling gaps in simplified partial differential equation formulas, outperforming conventional methods in handling heterogeneous crowdsensing data. While demonstrated for crowdsensing, the framework generalizes to other spatiotemporal completion tasks. Extensive experiments on real-world datasets demonstrate that MCWDD achieves over 5% error reduction compared to state-of-the-art baselines while offering enhanced robustness and physical interpretability.**

***Index Terms*—Internet of Things, sparse crowdsensing, spatiotemporal completion, partial differential equation**

I. Introduction

The synergy between 5G mobile networks [1] and pervasive computing devices [2] is disrupting and redefining distributed sensing paradigms. Mobile crowdsensing (MCS) leverages widespread mobile devices to collect and share data from urban environments [3]–[7]. Common techniques include two main approaches: participatory sensing (user-initiated tasks [8]) and opportunistic sensing (automatic data gathering via sensors like GPS [9] or cameras [10]). MCS enables applications such as traffic monitoring [11], pollution mapping [12], and noise-level detection [13] in smart city management [14]. However, the technique often struggles with incomplete spatial coverage due to uneven user participation, leaving data gaps in less-visited areas. Sparse crowdsensing addresses this limitation by intelligently selecting minimal yet strategic sampling points or participants, achieving high accuracy while reducing sensing costs and resource consumption [15], [16].

This work is supported in part by the National Natural Science Foundation of China under Grant Nos. 92567204, 62272193, and 62472194, in part by the Fundamental Research Funds for the Central Universities, in part by the Jilin Science and Technology Research Project under Grant No. 20260101016JJ, and in part by the Postdoctoral Fellowship Program of CPSF under Grant No. GZC20251096.

* En Wang and Wenbin Liu are the corresponding authors.

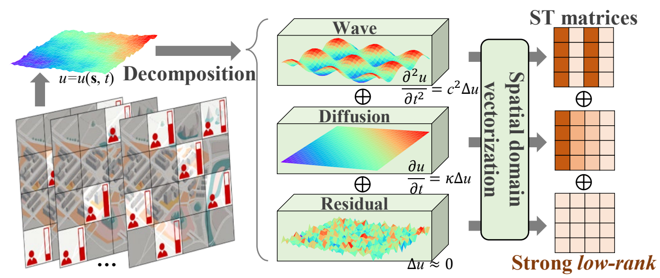


Fig. 1: Exemplary scenarios illustrating physics-inspired decomposition for interpretable spatiotemporal completion ideas and challenges in sparse crowdsensing.

In sparse crowdsensing, uneven participant distribution and sporadic data submissions lead to fragmented spatiotemporal observations, creating significant spatial gaps (locations with no data) and temporal gaps (periods with missing samples), as shown in Fig. 1 (left part). The core difficulty is inferring accurate, dense data values across unobserved locations and timestamps from these sparse, irregular samples. This requires sophisticated computational methods (e.g., tensor completion [17], graph neural networks [18], or spatiotemporal Kriging [19]) that exploit intrinsic structural patterns in the data (e.g., spatial autocorrelation [20], temporal periodicity [21], and environmental coherence [22]) to reconstruct a complete and consistent data field while mitigating noise and resource constraints inherent to sparse sensing deployments. Existing methods fall into two main categories: compressed sensing

(CS) and deep learning (DL). CS-based methods critically require the spatiotemporal matrix (ST matrix) to be low-rank, an assumption often invalid in practice. DL-based methods, while powerful, need large training datasets (often unavailable in sparse crowdsensing) and suffer from poor interpretability, hindering trust in critical applications. To overcome these limitations, we propose Matrix Completion by Wave and Diffusion Decomposition (MCWDD), a physics-inspired approach that integrates partial differential equations (PDEs) as physical priors into the completion process, eliminating the stringent low-rank requirement while enhancing both the accuracy and interpretability of the spatiotemporal completion. Beyond crowdsensing, MCWDD applies to any spatiotemporal data with wave and diffusion characteristics, such as traffic flow and meteorological fields.

Fig. 1 visually outlines the core idea of addressing the physics-inspired spatiotemporal completion. It decomposes complex original data into three PDE-constrained components: wave component, diffusion component, and residual component. Unlike seasonal and trend decomposition using locally estimated scatterplot smoothing (STL) for time series, this explicitly embeds physics, which relies on PDE constraints, and represents data as a sum of physically meaningful components with strong low-rank. Through spatial domain vectorization, these exhibit strong low-rank properties due to wave periodicity, diffusion linearity, and residual negligibility. However, designing a physics-inspired spatiotemporal completion framework presents the following three distinct challenges: (i) **the effective component decomposition method**, (ii) **the subsequent processing after decomposition**, and (iii) **the spatiotemporal completion accuracy achieved by the physics-inspired method**.

Designing an effective method to decompose the spatiotemporal data into wave, diffusion, and residual components is challenging. It must inherently enforce the distinct governing PDE constraints (like wave equations or diffusion equations) on each separated component during the decomposition process itself. Therefore, how to **design the data decomposition method** is the first challenge.

Determining the optimal model structure and processing steps after decomposition is crucial. The framework must intelligently handle each physically meaningful component (wave, diffusion, residual) according to its specific characteristics and exploit their strong low-rank properties revealed via spatial vectorization for effective completion. Therefore, how to **process the decomposed components after decomposing** is the second challenge.

Ensuring the physics-inspired approach achieves competitive data completion accuracy compared to purely data-driven methods is challenging. The inherently physics-constrained nature of the components might limit flexibility compared to unconstrained deep learning models, a limitation that can become pronounced when modeling complex patterns, necessitating strategies to bridge this potential accuracy gap while retaining physical interpretability. Therefore, how to **deal with the limitations of the physical model** is the third challenge.

To address these challenges, we propose MCWDD, which achieves component decomposition using PDE constraints with iterative smoothing for the wave and diffusion components. After decomposition, we perform completion on each of the three components separately. Crucially, for the completion of each individual component, we employ a hybrid approach combining physical models with data-driven methods. This strategy effectively mitigates the potential negative impact on data completion accuracy that could arise from solely relying on physical models. The final completed result is obtained by summing the completed versions of these three components. Core innovations and technical contributions of this work are summarized as follows:

- We propose MCWDD, a novel spatiotemporal completion framework. It uses PDEs to decompose data matrices into three physically interpretable components: wave, diffusion, and residual components.
- MCWDD employs analytical regularization to enforce strong low-rank properties in each component. This enables robust completion even for complex or non-low-rank data where traditional factorization fails.
- MCWDD integrates PDE-based physical priors with data-driven deep matrix factorization: the former enforces constraint adherence and reduces overfitting under scarce data, while the latter captures unmodeled dynamics to compensate for PDE simplifications. This combination enforces strong low-rank properties in each component, enabling robust completion for weak low-rank data.
- Extensive experiments on synthetic and real-world data show MCWDD achieves higher completion accuracy (over 5% compared to conventional methods) and enhanced interpretability than conventional methods, demonstrating superior capability in handling crowdsensing data heterogeneity.

II. Related Work

A. Sparse Crowdsensing

Sparse crowdsensing leverages ubiquitous mobile devices to collect urban-scale data (e.g., traffic flow [11], noise pollution [23], etc.) at low cost. Unlike traditional dense deployments, spatial sparsity arises from uneven user participation and device heterogeneity, while temporal sparsity stems from irregular sampling patterns. Prior work addresses these challenges through participant selection frameworks that optimize coverage-energy trade-offs [24] and incentive mechanisms combining game theory with deep reinforcement learning [25]. Furthermore, Feng *et al.* [26] investigated the task allocation problem in crowdsensing based on vehicular networks, combining clustering with the CMAB model to enhance efficiency.

For details, Xue *et al.* [11] proposed a novel spatiotemporal model combining graph neural networks and transformers to address the instability and degraded accuracy of traffic state estimation caused by data sparsity. Rimediotti *et al.* [23] proposed a data-driven crowdsensing system for noise pollution monitoring that leverages historical records and spatial correlations between road segments to predict noise levels in urban areas despite data sparsity. Zhang *et al.* [24] proposed a conceptual framework using a four-stage lifecycle model and the 4W1H taxonomy to systematically characterize crowdsensing research domains and identify future research directions. Wang *et al.* [25] proposed a deep reinforcement learning framework using Q-networks to optimize cell selection to reduce sensed cells while maintaining data inference quality. However, unrestricted participant recruitment is often constrained by limited sensing budgets [27]. By leveraging spatiotemporal correlations extracted from historical data, unsensed areas can be inferred through statistical completion frameworks, as demonstrated in the work of Xie *et al.* [28]. On the other hand, in terms of system deployment, Liu *et al.* [29] studied the revenue-maximizing deployment mechanism for online service function chains in edge-cloud environments, and their optimization approach provides valuable insights for constructing efficient crowd-sensing systems.

B. Spatiotemporal Completion

In the field of data recovery, besides traditional matrix and tensor methods, Liu *et al.* [30] proposed new data recovery techniques for heterogeneous graphs containing multi-source incremental data, providing a fresh perspective for processing complex correlated perceptual data. Despite progress, these methods overlook dynamic spatiotemporal dependencies and physical correlations. Spatiotemporal completion in sparse crowdsensing has been addressed through diverse methodologies spanning statistical models, tensor decomposition, and deep learning architectures. Early approaches employed Gaussian Processes, Pacheco [31] establishes theoretically sound and computationally feasible Gaussian process frameworks for spatiotemporal modeling. Sparse crowdsensing fundamentally represents a spatiotemporal matrix completion problem from a mathematical modeling standpoint. These matrices, arising in real-world scenarios, inherently possess low-rank properties stemming from spatiotemporal correlations across neighboring locations and time intervals. Although matrix factorization (MF) [32] demonstrates effectiveness for linearly correlated data inference, it faces limitations in capturing complex nonlinear spatiotemporal relationships. To bridge this gap, Fan and Cheng [17] develop Deep Matrix Factorization (DMF), integrating neural networks to model nonlinear patterns. Xie *et al.* [33] propose a two-phase framework leveraging environmental features for fractured spatiotemporal data recovery. Critically, existing approaches still lack a robust methodology that simultaneously ensures physical interpretability and reduces dependency on strict low-rank assumptions.

III. System Model and Problem Formulation

A. System Model

As illustrated in Fig. 1, our research operates within the established paradigm of sparse crowdsensing – a cost-effective approach for large-scale monitoring. This framework strategically recruits participatory mobile device users to collect spatiotemporally constrained measurements across metropolitan areas. The critical challenge lies in reconstructing the complete spatiotemporal data field from these severely undersampled and irregularly distributed observations. To systematically address this inverse problem, this subsection establishes the rigorous mathematical formulation of our proposed physics-informed decomposition framework, formalizing its operational mechanics and optimization objectives.

1) Sensing Model & Observed Data: Given a whole urban sensing map that the observed data is represented as a spatiotemporal matrix $\mathbf{Y} \in \mathbb{R}^{N_s \times N_t}$, where N_s denotes the number of discrete spatial locations and N_t denotes the number of discrete time steps, the set $\Omega \subset \{1, \ldots, N_s\} \times \{1, \ldots, N_t\}$ indexes the observed entries such that $|\Omega| \ll N_s N_t$. The incomplete observations are given by the projection $\mathcal{P}_\Omega(\mathbf{Y})$. Therefore, the core objective is to construct a data completion function $f(\cdot)$ capable of systematically inferring all unsensed entries from the sparse matrix $\mathcal{P}_\Omega(\mathbf{Y})$. The estimated complete matrix $\hat{\mathbf{Y}}$ is derived as:

$$\hat{\mathbf{Y}} = f(\mathcal{P}_\Omega(\mathbf{Y})). \tag{1}$$

2) Target Matrix & Physical Dynamics: The underlying complete spatiotemporal field is denoted as $u(\mathbf{s}, t) \in \mathbb{R}$, where $\mathbf{s}$ denotes the spatial coordinate of a discrete spatial location and where t denotes the temporal coordinate of a discrete time step. For any spatiotemporal index $\omega \in \{1, \ldots, N_\mathrm{s}\} \times \{1, \ldots, N_\mathrm{t}\}$, there is a spatiotemporal coordinate $(\mathbf{s}, t)$ corresponding to it. There is the following mapping relationship of these two values:

$$\omega \xrightarrow{\mathcal{T}} (\mathbf{s}, t), \tag{2}$$

which can also be represented by $(\mathbf{s}, t) = \mathcal{T}(\omega)$ or $\omega = \mathcal{T}^{-1}(\mathbf{s}, t)$. The dynamics of $u(\mathbf{s}, t)$ are governed by PDEs. As shown in Fig. 1, $u(\mathbf{s}, t)$ is decomposed as follows:

$$u(\mathbf{s}, t) = u_\mathrm{W}(\mathbf{s}, t) + u_\mathrm{D}(\mathbf{s}, t) + u_\mathrm{R}(\mathbf{s}, t), \tag{3}$$

where $u_\mathrm{W}(\mathbf{s}, t), u_\mathrm{D}(\mathbf{s}, t), u_\mathrm{R}(\mathbf{s}, t) \in \mathbb{R}$ denotes the wave component, the diffusion component, and the residual component, respectively. The corresponding ST matrix equation expression is denoted as follows:

$$\mathbf{Y} = \mathbf{W} + \mathbf{D} + \mathbf{R}, \tag{4}$$

where

- $\mathbf{W}$ - the ST matrix expression of the wave component;
- $\mathbf{D}$ - the ST matrix expression of the diffusion component;
- $\mathbf{R}$ - the ST matrix expression of the residual component.

In Fig. 1, the primary types (ignoring partial differential components above the third order) of physics are considered:

- Wave propagation: $\frac{\partial^2 u}{\partial t^2} = c^2 \Delta u$,
- Diffusion: $\frac{\partial u}{\partial t} = \kappa \Delta u$,
- Stable field: $\Delta u = 0$,

where $c > 0$ is the wave speed, $\kappa > 0$ is the diffusion coefficient, and Δ is the Laplacian operator.

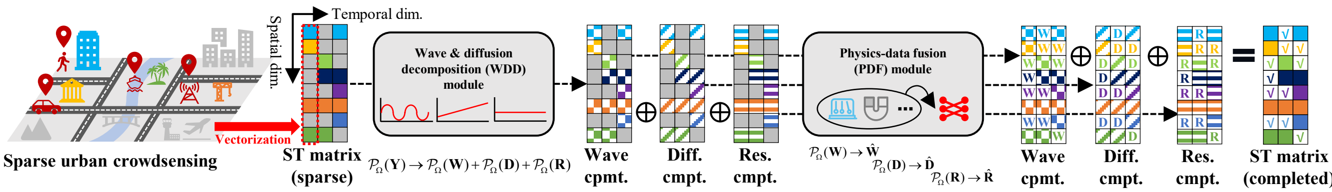


Fig. 2: The framework employs physics-inspired decomposition for interpretable spatiotemporal completion. Gray icons or grids indicate missing data. The raw sparse sensing data undergoes decomposition by the WDD module and completion via the PDF module, yielding the final complete ST matrix.

B. Problem Formulation

Problem [Physics-Inspired spatiotemporal completion via Sparse Crowdsensing]: Within a crowdsensing framework where distributed users collect data, the spatiotemporal observations can be structured into an $N_s \times N_t$ matrix (N_s locations across spatial dimensions and N_t discrete time steps). This matrix is inherently sparse due to uneven sensing coverage. Our core methodology simultaneously addresses two objectives: (i) completing this sparse matrix and (ii) constructing a physics-inspired data completion model. The completion error is thereby jointly minimized under governing physical equations through this integrated approach. The mathematical expression for our goal is denoted as follows:

$$\min_{f,\mathbf{W},\mathbf{D},\mathbf{R}} \ ||\mathbf{W}+\mathbf{D}+\mathbf{R}-f(\mathcal{P}_\Omega(\mathbf{Y}))||, \tag{5}$$

$$\text{s.t.} \ \ \mathcal{P}_\Omega(\mathbf{W}+\mathbf{D}+\mathbf{R}) = \mathcal{P}_\Omega(\mathbf{Y}). \tag{6}$$

To provide a specific example demonstrating our model's operation within sparse crowdsensing, consider an urban environmental monitoring task requiring the reconstruction of complete spatiotemporal maps. To reduce sensing costs while enhancing robustness, only partial observations are collected per time step. As Fig. 2 illustrates, we first decompose the sparse observations through Wave-Diffusion Decomposition (WDD), representing them as the sum of three sparse matrices. Subsequently, the Physics-Data Fusion (PDF) module performs physics-informed data completion by synergistically integrating physical constraints with data-driven priors. Finally, the completed components are aggregated to reconstruct the full spatiotemporal field.

IV. Wave and Diffusion Decomposition Module

Aiming at the wave-diffusion decomposition problem of sparse spatiotemporal matrices, our ideas are shown as follows: As shown in Fig. 3, our approach iteratively decomposes sparse spatiotemporal data by first initializing an arbitrary diffusion component, then cyclically subtracting this diffusion estimate from the input to perform physics-constrained wave smoothing on the residual, subsequently subtracting the updated wave component to execute diffusion-constrained smoothing, computing the remaining residual, and repeating this alternating extraction process until changes in both wave and diffusion components diminish below negligible thresholds, achieving self-consistent physical decomposition. We also provide a pseudocode flow as shown in **Algorithm 1**.

Algorithm 1 Wave and Diffusion Decomposition

Input: the sparse ST matrix $\mathcal{P}_\Omega(\mathbf{Y})$
Output: the sparse wave component matrix $\mathcal{P}_\Omega(\mathbf{W})$, the sparse diffusion component matrix $\mathcal{P}_\Omega(\mathbf{D})$, the sparse residual component matrix $\mathcal{P}_\Omega(\mathbf{R})$

```
Rand. Init. the wave component matrix P_Ω(W), the diffusion component matrix P_Ω(D);
count ← 0;
while not convergent and count < MAX_ITER:
    P_Ω(W) ← P_Ω(Y) − P_Ω(D);
    Smooth W to make L_W(W) = ∂²W/∂t² − c²ΔW reduce;
    P_Ω(D) ← P_Ω(Y) − P_Ω(W);
    Smooth D to make L_D(D) = ∂D/∂t − κΔD reduce;
    Calculate P_Ω(R) ← P_Ω(Y) − P_Ω(W) − P_Ω(D);
    count ← count + 1;
return P_Ω(W), P_Ω(D), and P_Ω(R).
```

WDD is a sophisticated and robust filtering procedure for decomposing sparse spatiotemporal data into its three components: Wave ($\mathbf{W}$), Diffusion ($\mathbf{D}$), and Residual ($\mathbf{R}$) components. It uses iterative applications of PDE smoothing. The PDE-based smoothing benefits data with wave-like periodicity or diffusion-driven smoothness (e.g., environmental monitoring, traffic flow); the residual component captures deviations from these assumptions. The core decomposition formula is as follows:

$$\mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{W}) + \mathcal{P}_\Omega(\mathbf{D}) + \mathcal{P}_\Omega(\mathbf{R}), \tag{7}$$

where $\mathcal{P}_\Omega(\mathbf{Y})$ denotes the observed sparse spatiotemporal value, $\mathcal{P}_\Omega(\mathbf{W})$ denotes the estimated wave component, which represents repeating patterns over a fixed, known spatiotemporal period. $\mathcal{P}_\Omega(\mathbf{D})$ denotes the estimated diffusion component. This captures large-scale growth or decline. $\mathcal{P}_\Omega(\mathbf{R})$ denotes the residual component (or remainder) that remains after the wave and diffusion components have been removed ($\mathcal{P}_\Omega(\mathbf{R}) = \mathcal{P}_\Omega(\mathbf{Y}) - \mathcal{P}_\Omega(\mathbf{W}) - \mathcal{P}_\Omega(\mathbf{D})$), which contains noise, outliers, and any tiny-scale fluctuations not explained by the other components.

Algorithm 1 is designed to decompose a sparse, incomplete spatiotemporal data matrix $\mathcal{P}_\Omega(\mathbf{Y})$ into three distinct underlying components: a wave-like component $\mathcal{P}_\Omega(\mathbf{W})$, a diffusion-like component $\mathcal{P}_\Omega(\mathbf{D})$, and a residual component $\mathcal{P}_\Omega(\mathbf{R})$. The operator $\mathcal{P}_\Omega(\cdot)$ signifies projection onto the set

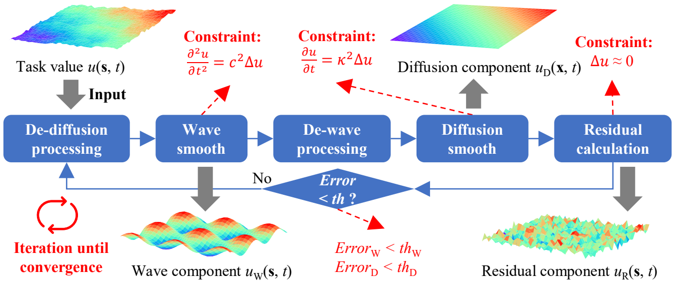


Fig. 3: Wave and Diffusion Decomposition Module.

of observed entries Ω, meaning the algorithm is specifically formulated to handle data that is missing a large portion of its values (sparse sampling). The goal is to recover these components only at the observed locations within Ω, producing the sparse output matrices $\mathcal{P}_\Omega(\mathbf{W})$, $\mathcal{P}_\Omega(\mathbf{D})$, and $\mathcal{P}_\Omega(\mathbf{R})$.

The algorithm begins by randomly initializing the two main component matrices specifically at the observed locations: $\mathcal{P}_\Omega(\mathbf{W})$ (representing the initial sparse wave component) and $\mathcal{P}_\Omega(\mathbf{D})$ (representing the initial sparse diffusion component). An iteration counter, count, is set to zero to keep track of the progress through the iterative loop. This randomization introduces the initial separation guess that will be refined.

The core of the algorithm is an iterative process that runs until either a convergence criterion is met (denoted as "not convergent" - the specific convergence measure is implied but not detailed in the figure) or a predefined maximum iteration limit (`MAX_ITER`) is reached.

Then we will update the sparse wave component ($\mathcal{P}_\Omega(\mathbf{W})$). The current estimate of the sparse wave component is updated based on the observed data $\mathcal{P}_\Omega(\mathbf{Y})$ minus the current estimate of the sparse diffusion component:

$$\mathcal{P}_\Omega(\mathbf{W}) \leftarrow \mathcal{P}_\Omega(\mathbf{Y}) - \mathcal{P}_\Omega(\mathbf{D}). \tag{8}$$

Following this update, a critical physics-based smoothing operation is applied to the completed wave component matrix $\mathbf{W}$. This step aims to enforce the characteristic behavior of wave phenomena, defined by the wave equation operator $\mathcal{L}_{\mathrm{W}}$:

$$\mathcal{L}_{\mathrm{W}}(\mathbf{W}) = \frac{\partial^2 \mathbf{W}}{\partial t^2} - c^2 \Delta \mathbf{W}, \tag{9}$$

where $\frac{\partial^2 \mathbf{W}}{\partial t^2}$ represents the second-order partial derivative in time (acceleration), Δ represents the spatial Laplacian operator (describing spatial curvature), and c is the wave speed constant. The smoothing step minimizes this operator, effectively making the updated $\mathbf{W}$ satisfy the wave equation more closely. The result is a new, smoother full $\mathbf{W}$ matrix; the relevant sparse part for the next steps remains $\mathcal{P}_\Omega(\mathbf{W})$.

Update Sparse Diffusion Component ($\mathcal{P}_\Omega(\mathbf{D})$): Using the newly updated sparse wave component, the algorithm now updates the sparse diffusion component by subtracting it from the input data:

$$\mathcal{P}_\Omega(\mathbf{D}) \leftarrow \mathcal{P}_\Omega(\mathbf{Y}) - \mathcal{P}_\Omega(\mathbf{W}). \tag{10}$$

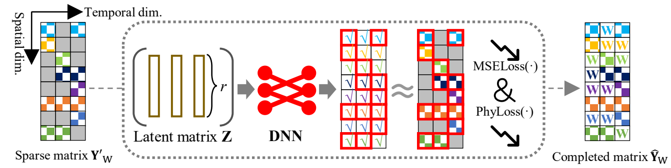


Fig. 4: Physics-Data Fusion Module.

Similar to the wave component, the entire (dense) diffusion component matrix $\mathbf{D}$ undergoes a physics-based smoothing step tailored to diffusion processes. This step minimizes the diffusion equation operator $\mathcal{L}_{\mathrm{D}}$:

$$\mathcal{L}_{\mathrm{D}}(\mathbf{D}) = \frac{\partial \mathbf{D}}{\partial t} - \kappa \Delta \mathbf{D}. \tag{11}$$

In this operator, $\frac{\partial \mathbf{D}}{\partial t}$ is the first-order partial derivative in time (rate of change), Δ again represents the spatial Laplacian, and κ is the diffusion coefficient constant. Minimizing this operator ensures the updated $\mathbf{D}$ adheres more closely to diffusion-like behavior (changes proportional to spatial gradients). This produces a smoother full $\mathbf{D}$ matrix, with $\mathcal{P}_\Omega(\mathbf{D})$ now reflecting this smoothed estimate at observed points. Calculate Sparse Residual Term ($\mathcal{P}_\Omega(\mathbf{R})$): After updating both $\mathcal{P}_\Omega(\mathbf{W})$ and $\mathcal{P}_\Omega(\mathbf{D})$ within the iteration loop, the algorithm computes the sparse residual term. This represents the part of the observed data not explained by the current matrices of wave component and diffusion component:

$$\mathcal{P}_\Omega(\mathbf{R}) \leftarrow \mathcal{P}_\Omega(\mathbf{Y}) - \mathcal{P}_\Omega(\mathbf{W}) - \mathcal{P}_\Omega(\mathbf{D}). \tag{12}$$

This residual captures noise, model inaccuracies, or other unmodeled phenomena present in the sparse data.

Once the loop terminates (either due to detected convergence or reaching `MAX_ITER`), the algorithm returns the estimated full component matrices $\mathbf{W}$, $\mathbf{D}$, and $\mathbf{R}$. Although the primary computations focus on the sparse entries (via $\mathcal{P}_\Omega(\cdot)$), the smoothing steps implicitly define the components over the entire spatiotemporal domain. The output thus provides the complete decomposed matrices of wave component, diffusion component, and residual component.

V. Physics-Data Fusion Module

Aiming at the physics-data fusion problem of sparse spatiotemporal matrices, our ideas are shown in Fig. 4. Specifically, we first leverage the structure of DMF to complete sparse matrices. Simultaneously, we incorporate constraints derived from physical models into the loss function of the neural network. This dual-driven approach, integrating both physics and data, achieves explainable matrix completion while simultaneously enhancing the accuracy of the completion results to a certain extent. In this section, we will take the physics-inspired completion of the wave component as an example, and the completion methods of the diffusion component and the residual component are exactly the same.

DMF serves as the fundamental computational framework enabling this physics-data fusion strategy. While standard

Algorithm 2 Physics-Data Fusion (take wave component as an example)

Input: the sparse ST matrix $\mathcal{P}_\Omega(\mathbf{W})$, the Physics model $\mathcal{L}_{\mathrm{W}}(\mathbf{W})$

Output: the completed matrix $\hat{\mathbf{W}}$

Rand. Init. $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_{N_t}]$;
Rand. Init. $\Theta = \{\mathbf{\Theta}^{(1)}, \mathbf{\Theta}^{(2)}, \cdots, \mathbf{\Theta}^{(L)}\}$;
Rand. Init. $b = \{\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \cdots, \mathbf{b}^{(L)}\}$;
$count \leftarrow 0$;
Build the Neural Networks by using $\mathbf{Q}$, Θ, and b;
while not convergent **and** $count <$ MAX_ITER:
    Fix $\mathbf{Q}$, Θ, and b to reduce $\mathcal{L}$ by (17);
    $count \leftarrow count + 1$;
for j **from** 1 **to** N_{t}:
    Calculate $\hat{\mathbf{w}}_j$ by (16);
$\hat{\mathbf{W}} \leftarrow [\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \cdots, \hat{\mathbf{w}}_{N_t}]$;
$\hat{\mathbf{W}} \leftarrow \mathcal{P}_\Omega(\mathbf{W}) + \hat{\mathbf{W}} - \mathcal{P}_\Omega(\hat{\mathbf{W}})$;
return $\hat{\mathbf{W}}$.

Matrix Factorization (MF) offers a principled starting point for representing complex interactions within incomplete matrices, its reliance on simple linear combinations through latent factors may not fully capture the intricate dynamics inherent in physical systems. DMF builds upon this classical method, enhancing it with deeper architectures and crucially, the capacity to seamlessly incorporate physics-derived constraints into its learning process.

DMF builds upon the foundation of traditional MF used in collaborative filtering (e.g., recommender systems, data completion). Standard MF models the spatiotemporal interaction matrix $\mathbf{W} \in \mathbb{R}^{N_{\mathrm{s}} \times N_{\mathrm{t}}}$ (where each entry w_{ij} represents the interaction between the spatial location i and the time step j, such as temperature) by decomposing it into two lower-dimensional latent factor matrices: $\mathbf{W} = \mathbf{PQ}$, where $\mathbf{P} \in \mathbb{R}^{N_{\mathrm{s}} \times r}$ is with the column rank of r, and $\mathbf{Q} \in \mathbb{R}^{r \times N_{\mathrm{t}}}$ is with the row rank of r. r is the latent dimensionality ($r << \min(N_{\mathrm{s}}, N_{\mathrm{t}})$). The predicted interaction $w_{ij} = \mathbf{P}_i \mathbf{Q}_j$, where $\mathbf{P}_i$ is the i^{th} row of the matrix $\mathbf{P}$ (latent vector for the spatial location i) and $\mathbf{Q}_j$ is the j^{th} row of the matrix $\mathbf{Q}$ (latent vector for the time step j). While powerful, standard MF primarily captures linear relationships between the spatial locations and the time steps through the dot product.

DMF addresses the limitation of linearity by incorporating deep neural networks (specifically Multi-Layer Perceptrons - MLPs) into the factorization process. Instead of directly using the dot product between the spatial location and the time step vectors, DMF first processes these vectors separately or jointly through one or more nonlinear neural network layers. This allows the model to learn complex, nonlinear interactions between the spatial location and the time step features that are beyond the scope of simple dot products. The key idea is to learn a joint deep representation of the spatiotemporal pair that captures intricate patterns.

Therefore, the PDF architecture we designed consists of the following key components:

- **Input Layers**: Separate input layers for spatial locations and time steps. Each user i is represented by a one-hot encoded vector of length N_{s}. Similarly, each item j is represented by a one-hot encoded vector of length N_{t}.
- **Embedding Layers**: These one-hot vectors are fed into corresponding embedding layers (functionally equivalent to the $\mathbf{P}$ and $\mathbf{Q}$ matrices in MF). The spatial location embedding layer projects the one-hot input into a dense latent vector $\mathbf{p}_i \in \mathbb{R}^r$ (where r is the embedding dimension). The time step embedding layer outputs the dense latent vector $\mathbf{q}_j \in \mathbb{R}^r$.
- **Deep Interaction Layer**: The vector $\mathbf{q}_j$ is regarded as the latent expression of the sparse vector $\mathcal{P}_\Omega(\mathbf{w}_j)$. This vector is then passed through one or more fully connected (dense) neural network layers with nonlinear activation functions (e.g., $\mathrm{ReLU}(\cdot)$, $\tanh(\cdot)$). The final layer typically outputs a scalar prediction. The mathematical expression for the above process is as follows:

$$\mathbf{h}_j^{(0)} = \sigma_0(\mathbf{\Theta}^{(0)} \mathbf{q}_j), \tag{13}$$

$$\mathbf{h}_j^{(1)} = \sigma_1(\mathbf{\Theta}^{(1)} \mathbf{h}_j^{(0)} + \mathbf{b}^{(1)}), \tag{14}$$

$$\cdots$$

$$\mathbf{h}_j^{(L)} = \sigma_L(\mathbf{\Theta}^{(L)} \mathbf{h}_j^{(L-1)} + \mathbf{b}^{(L)}), \tag{15}$$

$$\hat{\mathbf{w}}_j = \sigma_{L+1}(\mathbf{\Theta}^{(L+1)} \mathbf{h}_j^{(L)} + \mathbf{b}^{(L+1)}), \tag{16}$$

  where $\sigma_0(\cdot)$ is the initial interaction function, $\sigma_1(\cdot)$, $\cdots$, $\sigma_L(\cdot)$ denote the activation function ($\mathrm{ReLU}(\cdot)$ is common for hidden layers), $\mathbf{\Theta}^{(l)}$, $\mathbf{b}^{(l)}$ are the weights and biases of layer l, L is the number of hidden layers, and $\sigma_{L+1}(\cdot)$ is the output activation function (e.g., sigmoid for probabilities, linear for ratings). The final prediction $\hat{\mathbf{w}}_j$ corresponds to the predicted interaction score $\mathcal{P}_\Omega(\mathbf{w}_j)$.
- **Training Objective**: The model is trained to minimize a loss function between the predicted scores $\hat{\mathbf{w}}_j$ and the actual observed interactions $\mathcal{P}_\Omega(\mathbf{w}_j)$ (or labels like click/no-click). Common loss functions include Mean Squared Error (MSE) for explicit feedback (ratings), i.e.:

$$\mathcal{L} = \sum_{j=1}^{N_{\mathrm{t}}} ||\mathcal{P}_\Omega(\mathbf{w}_j - \hat{\mathbf{w}}_j)||^2 + \lambda_1 \sum_{j=1}^{N_{\mathrm{t}}} ||\mathbf{q}_j||^2 + \lambda_2 \sum_{l=0}^{L+1} ||\mathbf{\Theta}^{(l)}||^2 + \lambda_3 \sum_{l=1}^{L+1} ||\mathbf{b}^{(l)}||^2 + \lambda_4 ||\mathcal{L}_{\mathrm{W}}(\mathbf{W})||, \tag{17}$$

  where λ_1, λ_2, λ_3, and λ_4 represent the weight parameters of the regularized penalty items.

In summary, the result of physics-data fusion is as follows:

$$\hat{\mathbf{W}} = [\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \cdots, \hat{\mathbf{w}}_{N_{\mathrm{t}}}]. \tag{18}$$

Considering that parts of the data (the index value $\omega \in \Omega$) are correctly sensed, the final result can be expressed as:

$$\hat{\mathbf{W}} \leftarrow \mathcal{P}_\Omega(\mathbf{W}) + \hat{\mathbf{W}} - \mathcal{P}_\Omega(\hat{\mathbf{W}}). \tag{19}$$

The pseudocode for the complete process of the above steps is shown in **Algorithm 2**.

TABLE I: Statistics of the **U-Air** dataset.

Country/Region - City	Chinese mainland - Beijing	
# Location	36 (each with 1 km $\times$ 1 km)	
# Time step	264 (each with an hour)	
Data [Unit]	$\mathrm{PM}_{2.5}$ [$\mu\mathrm{g/m}^3$]	PM_{10} [$\mu\mathrm{g/m}^3$]
Mean $\pm$ Std.	79.11 ± 81.21	63.12 ± 48.56

TABLE II: Statistics of the **Sensor-Scope** dataset.

Country/Region - City	Switzerland - Lausanne	
# Location	57 (each with 30 m $\times$ 50 m)	
# Time step	336 (each with half an hour)	
Data [Unit]	Temperature [°C]	Humidity [%]
Mean $\pm$ Std.	6.04 ± 1.87	84.52 ± 6.32

VI. Performance Evaluation

This section first details the experimental datasets and state-of-the-art baselines for sparse data completion. We subsequently evaluate our method's performance on each dataset, framed around four primary research questions that explicitly justify our experimental design objectives:

- **RQ1**: How do the hyper-parameters affect the performance of MCWDD?
- **RQ2**: Whether the performance of MCWDD in terms of data completion accuracy exceeds the baseline method.
- **RQ3**: How much impact does each component of WDD have on data completion results?
- **RQ4**: How does the removal or alteration of specific components affect the overall performance of MCWDD?
- **RQ5**: What is the training cost of MCWDD?

A. Datasets

Our evaluation framework employs two established urban sensing datasets — **U-Air** [34] and **Sensor-Scope** [35] — to benchmark MCWDD's physics-inspired data completion capabilities. Selected for their dual correlated sensing tasks, these datasets enable rigorous cross-task generalization assessment. Comprehensive statistical characterizations are detailed in TABLEs I and II. Both datasets, which widely adopted in the research community, feature extended high-precision collection periods, providing ideal testbeds for comparing our approach against baseline methods.

The **U-Air** [34] dataset provides critical air quality measurements across Beijing, China, with key indicators including PM_{10} and $\mathrm{PM}_{2.5}$ concentrations. For our data completion evaluation, we focus on both values due to their significance as a primary air pollution metric. We analyze 36 spatially distributed monitoring locations (each covering $1\mathrm{km} \times 1\mathrm{km}$), selected based on spatial relevance to urban sensing patterns and alignment with our research scope. This dataset enables a comprehensive assessment of spatiotemporal data completion methodologies in urban air quality contexts, building upon foundational urban sensing research. Our work aims to advance precision in urban air pollution monitoring systems through robust data recovery techniques.

TABLE III: Comparison of different data completion methods.

Method	Non-linear	Data-driven	Physics-inspired
KNN [20]	✗	✓	✗
ST-Kriging [19]	✗	✗	✗
GPR [36], [37]	✓	✓	✗
KDE [38]	✓	✓	✗
DMF [17]	✓	✓	✗
IGMC [18]	✓	✓	✗
ImputeFormer [39]	✓	✓	✗
PINO [40]	✓	✓	✓
MCWDD *	✓	✓	✓

* It is proposed in Sections III, IV and V of this paper.

The **Sensor-Scope** [35] comprises multi-modal environmental measurements from static sensors across École Polytechnique Fédérale de Lausanne (EPFL). While other parameters are available, this study exclusively analyzes temperature and humidity data due to their significance in urban microclimate studies. We partition the campus into a spatially gridded structure of 57 discrete locations (each $50\mathrm{m} \times 30\mathrm{m}$), enabling statistical analysis of microenvironmental variations. Following rigorous data preprocessing, substantial volumes of sensing data are aggregated across these locations. Our data completion framework aims to extract actionable insights about urban conditions' influence on environmental parameters through effective spatiotemporal pattern mining.

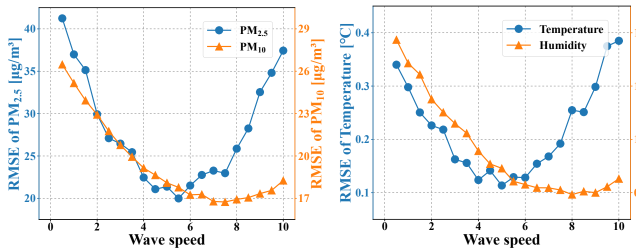
B. Baselines & Measures

1) Baselines: To enable spatiotemporal completion from sparse sensed data, we propose the MCWDD algorithm. Its performance is evaluated against six representative baselines:

- **KNN** [20]: Computes averages of the K-nearest temporal neighbors, with spatial interpolation capabilities.
- **ST-Kriging** [19]: Advanced geostatistical interpolation extending traditional Kriging for spatiotemporal data.
- **GPR** [36], [37]: Models data distributions via Gaussian processes to infer missing values.
- **KDE** [38]: Non-parametric probability density estimation for continuous distribution approximation.
- **IGMC** [18]: GNN-based completion framework originally designed for recommendation systems.
- **DMF** [17]: Neural-enhanced matrix factorization for high-dimensional data imputation.
- **ImputeFormer** [39]: Transformer-based architecture utilizing spatiotemporal attention and generative modeling for long-range, multivariate time series imputation.
- **PINO** [40]: Neural operator-based model that incorporates physical governing equations into the training loss for learning PDE solutions.

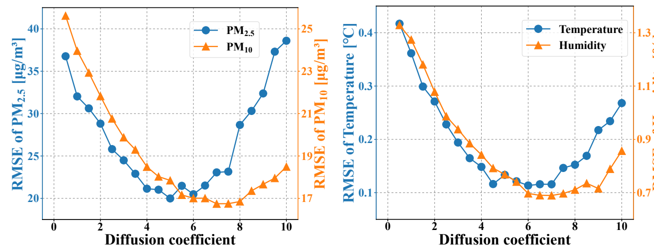
Given the methodological diversity spanning linear/non-linear, data-driven, and physics-inspired, we systematically compare their characteristics in TABLE III.

2) Measures: This paper addresses the problem of sparse crowdsensing spatiotemporal data imputation. To rigorously evaluate the completion accuracy, we employ the Root Mean



(a) **U-Air** (b) **Sensor-Scope**

Fig. 5: Completion error vs. wave speed.



(a) **U-Air** (b) **Sensor-Scope**

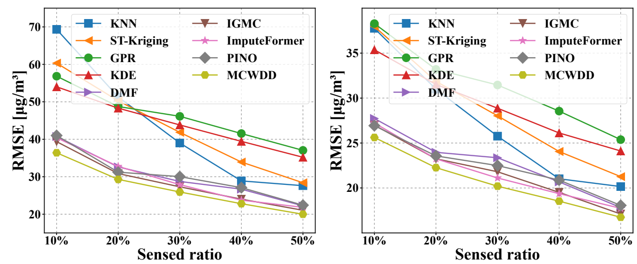
Fig. 6: Completion error vs. diffusion coefficient.

Square Error (RMSE) as the primary metric. Unless otherwise specified, the term "error" used throughout this paper consistently denotes RMSE.

C. *Impact of Hyper-Parameters* ***(RQ1)***

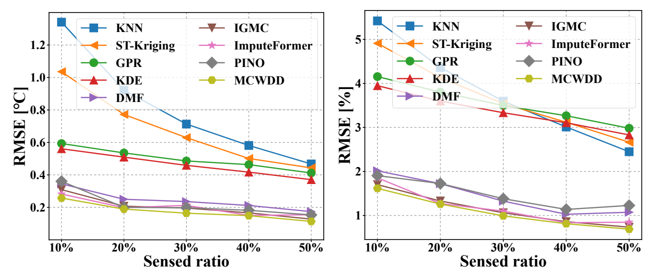
1) Wave Speed c: In the experimental evaluation of our MCWDD model, designed for sparse spatiotemporal completion, we first conducted a hyperparameter sensitivity analysis, focusing on the wave speed c, a critical parameter within the governing wave equation. To determine its optimal value, we systematically evaluated the model's data completion error across a range of c values. The experimental results, depicted in Fig. 5(a), illustrate the error profiles for the temperature and humidity data within the **U-Air** dataset. Corresponding results for the **Sensor-Scope** dataset are presented in Fig. 5(b). Analysis of these figures reveals that the completion error exhibits a clear sensitivity to the chosen c value for both sensing tasks. Based on the error minimization observed in our experiments, for the **U-Air** dataset we ultimately configured the wave speed $c = 5.5$ for the $PM_{2.5}$ data completion task and $c = 7$ for the PM_{10} data completion task across the evaluated datasets. For the **Sensor-Scope** dataset we ultimately configured the wave speed $c = 5$ for the temperature data completion task and $c = 8$ for the humidity data completion task across the evaluated datasets.

2) Diffusion Coefficient κ: Next, we evaluated the impact of the diffusion coefficient κ, a pivotal parameter within the diffusion equation framework of our MCWDD model. To identify its optimal configuration for sparse spatiotemporal completion, we systematically assessed the completion error across multiple values of κ. The performance profiles, detailing the model's accuracy in reconstructing missing data points under varying κ values, are presented for the **U-Air** dataset in Fig. 6(a) and for the **Sensor-Scope** dataset in Fig. 6(b). Analysis of these results demonstrates a clear dependency of the reconstruction fidelity on the chosen κ value for both sensing modalities. We observed distinct minima in the error curves corresponding to the optimal κ values for temperature and humidity completion. Consequently, based on the empirical error minimization for each specific sensing task, we selected a diffusion coefficient $\kappa = 5$ for the $PM_{2.5}$ task, $\kappa = 7$ for the PM_{10} task, $\kappa = 6$ for the temperature task, and $\kappa = 7$ for the humidity task.



(a) $PM_{2.5}$ (b) PM_{10}

Fig. 7: Completion error vs. sensed ratios (**U-Air**).

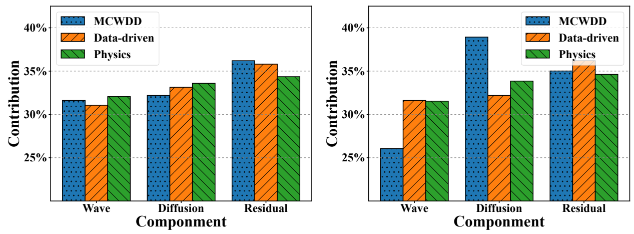


(a) Temperature (b) Humidity

Fig. 8: Completion error vs. sensed ratio (**Sensor-Scope**).

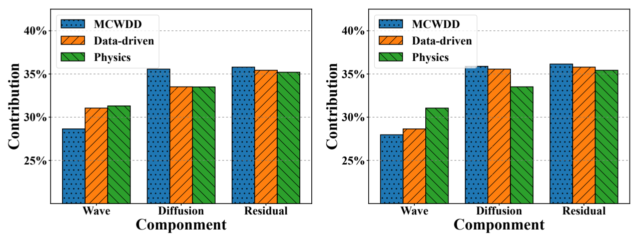
D. *Robustness Analysis* ***(RQ2)***

To assess the robustness of our MCWDD approach under varying levels of data sparsity, we conducted a series of experiments by progressively increasing the sensed ratio from a highly sparse 10% to a more informative 50%. This test rigorously evaluates the performance stability when the density of available observations fluctuates, a common challenge in real-world sparse crowdsensing deployments. Alongside our method, we benchmarked against a representative selection of baselines, including KNN, ST-Kriging, GPR, KDE, DMF, and IGMC. Additionally, we have introduced a Transformer-based method ImputeFormer, along with a physics-inspired method PINO for data completion. The data completion error, as defined previously, was measured and compared across all methods at each sensed ratio. Comparative results illustrating the performance evolution with increasing data density are presented in Figs. 7 and 8.



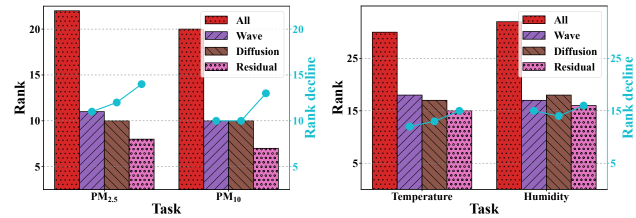
(a) $PM_{2.5}$ (b) PM_{10}

Fig. 9: The impact of data completion contributions from different components over **U-Air**.



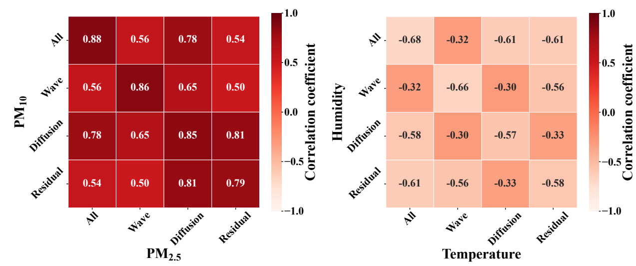
(a) Temperature (b) Humidity

Fig. 10: The impact of data completion contributions from different components over **Sensor-Scope**.



(a) **U-Air** (b) **Sensor-Scope**

Fig. 11: The impact of rank decline by different components over **U-Air** and **Sensor-Scope**.



(a) **U-Air** (b) **Sensor-Scope**

Fig. 12: The correlation coefficient of different components over **U-Air** and **Sensor-Scope**.

E. Impact of Individual Components ***(RQ3)***

A core innovation of our MCWDD is its novel fusion of physics-informed modeling with deep learning and matrix factorization. Specifically, MCWDD decomposes the complex spatiotemporal field into interpretable physical components governed by distinct dynamics. To evaluate the contribution of each component to the overall imputation performance, we conducted a component ablation study across diverse tasks within our datasets. In Figs. 9 and 10, this experiment systematically isolates the output of each term, allowing us to measure its individual impact on the final data completion error and gain deeper insights into how the physical priors enhance imputation accuracy in different sensing scenarios.

Another theoretical advantage of MCWDD is its ability to decompose a matrix exhibiting weak or non-low-rank properties into a sum of components, each inherently possessing strong low-rank structures due to the governing physics (wave/diffusion equations) and the targeted regularization of the residual term. To empirically validate this effect, Fig. 11 presents bar charts comparing the matrix rank before decomposition against the individual ranks of these components. This analysis was rigorously conducted for four distinct sensing tasks across both datasets. The results consistently and convincingly demonstrate that the rank of each decomposed component is significantly lower than that of the original. This critical rank reduction, achieved through MCWDD's physics-informed decomposition, ultimately facilitates the enhanced performance of its data-driven spatiotemporal completion module when handling complex real-world scenarios.

MCWDD addresses the inherent sparsity of urban crowd sensing data by decomposing the original sparse matrix into three distinct components: wave, diffusion, and residual terms. To rigorously evaluate the effectiveness of this decomposition strategy in capturing these distinct phenomena, we specifically examined the correlation structures within a unified dataset containing two data streams known to exhibit strong interdependencies. Specifically, we computed the correlation coefficients between these streams both before decomposition and within each resulting component after decomposition. This quantifies how the decomposition affects the observed correlation structure. The results are shown in Fig. 12.

F. Ablation Experiments ***(RQ4)***

To rigorously quantify the individual contributions and synergistic benefits of the dual modules in our model, we conducted a comprehensive ablation study. We systematically removed each core component: first, the physics-informed modeling module, followed by the deep learning/matrix factorization module, while keeping all other architectural elements unchanged. Performance was evaluated under the same set of varying sensed ratios (from 10% to 50%) employed in the robustness analysis phase. The detailed results, presented in TABLEs IV to VII, clearly illustrate two key findings: (i) Both modules provide significant and distinct contributions to the overall accuracy of MCWDD. The physics-informed component integrates domain knowledge and physical constraints, enhancing the model's extrapolation reliability and generalizability under data-scarce conditions. Meanwhile, the data-driven component effectively captures complex spatiotemporal patterns and nonlinear relationships from high-dimensional observations. Their functionalities are complementary, together forming a cohesive and robust modeling framework.

TABLE IV: Completion error of $PM_{2.5}$ vs. sensed ratios over **U-Air** [$\mu g/m^3$].

Method	Sensed ratio				
	10%	20%	30%	40%	50%
w/o data-driven	39.977	31.037	29.366	27.946	25.705
w/o physics	39.369	30.845	27.311	24.010	21.050
MCWDD	36.399	29.295	25.931	22.807	19.991
Improvement	2.970	1.550	1.380	1.203	1.059

TABLE V: Completion error of PM_{10} vs. sensed ratios over **U-Air** [$\mu g/m^3$].

Method	Sensed ratio				
	10%	20%	30%	40%	50%
w/o data-driven	28.469	23.278	22.690	20.589	17.660
w/o physics	26.985	23.246	21.804	19.512	17.105
MCWDD	25.622	22.260	20.199	18.533	16.745
Improvement	1.363	0.986	1.605	0.979	0.360

TABLE VI: Completion error of Temperature vs. sensed ratios over **Sensor-Scope** [°C].

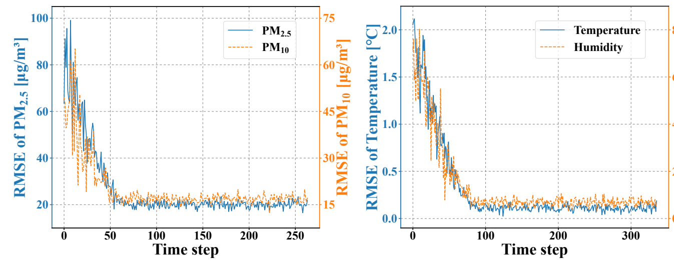
Method	Sensed ratio				
	10%	20%	30%	40%	50%
w/o data-driven	0.3813	0.2327	0.2113	0.1692	0.1302
w/o physics	0.3089	0.2072	0.1958	0.1661	0.1275
MCWDD	0.2573	0.1893	0.1640	0.1489	0.1135
Improvement	0.0516	0.0179	0.0318	0.0172	0.0140

TABLE VII: Completion error of Humidity vs. sensed ratios over **Sensor-Scope** [%].

Method	Sensed ratio				
	10%	20%	30%	40%	50%
w/o data-driven	1.8225	1.3484	1.0753	0.9321	0.7709
w/o physics	1.7014	1.3283	1.0613	0.8592	0.7346
MCWDD	1.6161	1.2592	0.9907	0.8133	0.6895
Improvement	0.0853	0.0691	0.0706	0.0459	0.0451

(ii) Removing either module leads to substantial performance degradation, which is particularly pronounced at lower sensed ratios (e.g., 10%–30%). This outcome underscores the critical synergy between data-driven learning and physical priors in handling sparse and intricate spatiotemporal imputation tasks. Physical constraints provide interpretable structural guidance, reducing arbitrariness in scenarios with extreme data sparsity.

Through this quantitative ablation analysis, we have not only verified the necessity of each module but also highlighted the methodological value of integrated cross-domain modeling in addressing practical spatiotemporal data challenges. Relying solely on one approach often falls short of balancing interpretability and flexibility, whereas a thoughtfully designed fusion of both can achieve a harmonious improvement in both accuracy and reliability.



(a) **U-Air** (b) **Sensor-Scope**

Fig. 13: Inference accuracy under different input time steps over **U-Air** and **Sensor-Scope**.

TABLE VIII: Training Time over **U-Air** and **Sensor-Scope** [s].

Dataset / Task	U-Air		Sensor-Scope	
	$PM_{2.5}$	PM_{10}	Temperature	Humidity
Training Time	107.254	99.369	127.147	117.950

*G. Training Cost (**RQ5**)*

1) Number of Required Time Steps: To evaluate the minimum data requirements for effective model training, we systematically increased the number of available time steps while maintaining a fixed spatial resolution. Starting from a minimal configuration, we measured the completion accuracy of MCWDD as more temporal data became available. The experimental results, as illustrated in Fig. 13, demonstrate that the model's performance improves significantly as the number of time steps increases from 10 to approximately 100. For the $PM_{2.5}$ monitoring task in the **U-Air** dataset, stable performance is achieved with around 70 time steps, while the temperature monitoring task in the **Sensor-Scope** dataset requires approximately 100 time steps for optimal performance. This indicates that MCWDD can achieve robust completion accuracy without requiring extensive historical data, making it particularly suitable for sparse crowdsensing applications where data collection is limited.

2) Model Training Time: The computational efficiency of MCWDD was evaluated by measuring the training time required for different sensing tasks. As shown in TABLE VIII, our experiments were conducted on a standard computing platform with Intel(R) Core(TM) i5-14600KF (3.50 GHz) and 32 GB RAM, using Python 3 with PyTorch implementation. According to the results summarized in TABLE VIII, the training time for MCWDD varies by dataset and task complexity: 107.254s for $PM_{2.5}$ and 99.396s for PM_{10} completion in the **U-Air** dataset, while the **Sensor-Scope** dataset required 127.147s for temperature and 117.950s for humidity monitoring. These results demonstrate that despite the sophisticated physics-informed decomposition approach, MCWDD maintains reasonable computational requirements, completing model training within approximately 2min for typical urban sensing scenarios with lots of locations and time steps.

VII. Conclusion

To overcome limitations of traditional low-rank methods in sparse crowdsensing data completion, this work proposes MCWDD, a novel physics-informed framework. MCWDD leverages PDEs to decompose raw data into three regularized components, imposing physically meaningful structure that ensures each component exhibits strong low-rank properties for robust completion while adhering to core physical constraints. By fusing PDE priors with data-driven residuals, MCWDD offers a general solution beyond global low-rank assumptions.

References

[1] Y. Zhang and C. L. P. Chen, "Secure heterogeneous data deduplication via FoReferencesg-assisted mobile crowdsensing in 5G-enabled IIoT," *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2849–2857, Apr. 2022.

[2] X. Kong, J. Cao, H. Wu, and C. R. Hsu, "Mobile crowdsourcing and pervasive computing for smart cities," *Pervasive Mob. Comput.*, vol. 61, 2020, Art. no. 101114.

[3] A. Abdeddine, Y. Iraqi, and L. Mekouar, "Mobile crowdsensing model: A survey," *J. Syst. Archit.*, vol. 162, 2025, Art. no. 103384.

[4] J. Wu and E. Wang, *Mobile Crowdsourcing From Theory to Practice*. Berlin, Germany: Springer, 2023.

[5] S. Zhao, G. Qi, T. He, J. Chen, Z. Liu, and K. Wei, "A survey of sparse mobile crowdsensing: Developments and opportunities," *IEEE Open J. Comput. Soc.*, vol. 3, pp. 73–85, 2022.

[6] R. She, "Survey on incentive strategies for mobile crowdsensing system," in *Proc. IEEE 11th Int. Conf. Softw. Eng. Service Sci.*, 2020, pp. 511–514.

[7] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: Current state and future challenges," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32–39, Nov. 2011.

[8] Z. Li, J. Wu, S. Long, Z. Zheng, C. Li, and M. Dong, "User-driven privacy-preserving data streams release for multi-task assignment in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 24, no. 5, pp. 3719–3734, May 2025.

[9] R. Mandal, N. Agarwal, S. K. Dey, S. Saha, S. Nandi, and S. Chakraborty, "GPS crowdsensing for public stoppage planning of city buses: A perspective of developing economies," in *Proc. 14th Int. Conf. Commun. Syst. Netw.*, 2022, pp. 716–724.

[10] Z. Pan, H. Yu, C. Miao, and C. Leung, "Crowdsensing air quality with camera-enabled mobile devices," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 4728–4733.

[11] J. Xue, Y. Xu, W. Wu, T. Zhang, Q. Shen, H. Zhou, and W. Zhuang, "Sparse mobile crowdsensing for cost-effective traffic state estimation with spatio-temporal transformer graph neural network," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 16 227–16 242, May 2024.

[12] J. Rubio-Aparicio and J. Santa, "An embedded crowdsensing unit for mobile urban pollution monitoring," *IEEE Commun. Mag.*, vol. 61, no. 1, pp. 90–96, Jan. 2023.

[13] P. Huang, X. Zhang, L. Guo, and M. Li, "Incentivizing crowdsensing-based noise monitoring with differentially-private locations," *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 519–532, Feb. 2021.

[14] Y. Zhao, Z. Guan, and J. Zhang, "Return freight strategies and selling formats in e-commerce supply chain: The perspective of consumer fairness concerns and online shopping returns," *J. Retail. Consum. Serv.*, vol. 75, 2023, Art. no. 103493.

[15] Y. Wang, Q. Zhou, Y. Zhang, G. Deng, L. Liang, and X. Zhang, "Demand-driven sparse mobile crowdsensing with neighborhood-aware reconstruction," *IEEE Internet Things J.*, vol. 12, no. 14, pp. 28 287–28 302, Jul. 2025.

[16] H. Zeng, Y. Xiong, J. She, and A. Yu, "Task assignment scheme designed for online urban sensing based on sparse mobile crowdsensing," *IEEE Internet Things J.*, vol. 12, no. 11, pp. 17 791–17 806, Jun. 2025.

[17] J. Fan and J. Cheng, "Matrix completion by deep matrix factorization," *Neural Netw.*, vol. 98, pp. 34–41, 2018.

[18] M. Zhang and Y. Chen, "Inductive matrix completion based on graph neural networks," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 11 738–11 751.

[19] Y. Wu, D. Zhuang, A. Labbe, and L. Sun, "Inductive graph neural networks for spatiotemporal kriging," in *Proc. Workshops 35th AAAI Conf. Artif. Intell.*, 2021, pp. 4478–4485.

[20] L. Wang, D. Zhang, A. Pathak, C. Chen, H. Xiong, D. Yang, and Y. Wang, "CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, 2015, pp. 683–694.

[21] H. Shu, H. Wang, J. Peng, and D. Meng, "Low-rank tensor completion with 3-D spatiotemporal transform for traffic data imputation," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 11, pp. 18 673–18 687, Nov. 2024.

[22] X. Guo, H. Zhang, W. Liu, R. Zhang, and P. Wang, "A joint trajectory completion method combined spatiotemporal motion for long-span bridge surveillance system with dead zone," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–12, 2025, Art. no. 5009012.

[23] J. Rimediotti, F. Montori, L. Sciullo, and L. Bononi, "Inferring the urban noise pollution with sparse data through crowdsensing," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops Affiliated Events*, 2024, pp. 643–648.

[24] D. Zhang, L. Wang, H. Xiong, and B. Guo, "4W1H in mobile crowd sensing," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 42–48, Aug. 2014.

[25] L. Wang, W. Liu, D. Zhang, Y. Wang, E. Wang, and Y. Yang, "Cell selection with deep reinforcement learning in sparse mobile crowdsensing," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst.*, 2018, pp. 1543–1546.

[26] X. Feng, D. Guo, Z. Liu, and H. Zheng, "Task allocation in IoV-based crowdsensing combing clustering and CMAB," *Chinese J. Internet Things*, vol. 5, no. 3, pp. 86–96, Sep. 2021.

[27] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 161–167, Jul. 2016.

[28] K. Xie, X. Li, X. Wang, G. Xie, J. Wen, and D. Zhang, "Active sparse mobile crowd sensing based on matrix completion," in *Proc. Int. Conf. Manag. Data*, 2019, pp. 195–210.

[29] G. Liu, S. Chen, and Z. Pang, "Deployment mechanism for maximizing revenue from online service function chains in edge cloud environments," *J. Tsinghua Univ. (Sci. Technol.)*, vol. 65, no. 8, pp. 1516–1529, Aug. 2025.

[30] Y. Liu, X. Jiang, Y. Cui, Y. Wang, and W. Hu, "Missing data recovery for heterogeneous graphs with incremental multi-source data fusion," *Front. Comput. Sci.*, vol. 19, no. 12, 2025, Art. no. 1912614.

[31] R. A. Pacheco, "Gaussian processes for spatiotemporal modelling," Ph.D. dissertation, University of Sheffield, UK, 2015.

[32] A. Falini, G. Castellano, C. Tamborrino, F. Mazzia, R. M. Mininni, A. Appice, and D. Malerba, "Saliency detection for hyperspectral images via sparse-non negative-matrix-factorization and novel distance measures*," in *Proc. IEEE Conf. Evolving Adapt. Intell. Syst.*, 2020, pp. 1–8.

[33] K. Xie, X. Ning, X. Wang, D. Xie, J. Cao, G. Xie, and J. Wen, "Recover corrupted data in sensor networks: A matrix completion solution," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1434–1448, May 2017.

[34] Y. Zheng, F. Liu, and H. Hsieh, "U-Air: When urban air quality inference meets Big Data," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2013, pp. 1436–1444.

[35] F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Application-specific sensor network for environmental monitoring," *ACM Trans. Sens. Netw.*, vol. 6, no. 2, pp. 17:1–17:32, 2010.

[36] F. Yin and F. Gunnarsson, "Distributed recursive Gaussian processes for RSS map applied to target tracking," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 3, pp. 492–503, Apr. 2017.

[37] H. C. Yen and C. C. Wang, "Cross-device Wi-Fi map fusion with Gaussian processes," *IEEE Trans. Mobile Comput.*, vol. 16, no. 1, pp. 44–57, Jan. 2017.

[38] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.

[39] T. Nie, G. Qin, W. Ma, Y. Mei, and J. Sun, "ImputeFormer: Low rankness-induced transformers for generalizable spatiotemporal imputation," in *Proc. 30th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2024, pp. 2260–2271.

[40] Z. Li, H. Zheng, N. Kovachki, D. Jin, H. Chen, B. Liu, K. Azizzadenesheli, and A. Anandkumar, "Physics-informed neural operator for learning partial differential equations," *ACM/IMS J. Data Sci.*, vol. 1, no. 3, pp. 1–27, May 2024.