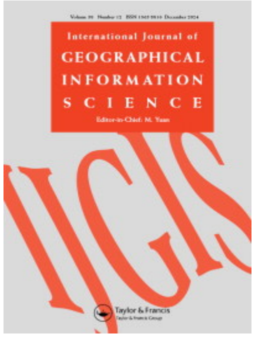

International Journal of Geographical Information Science

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tgis20

A tensor decomposition method based on embedded geographic meta-knowledge for urban traffic flow imputation

Xiaoyue Luo, Shifen Cheng, Lizeng Wang, Yuxuan Liang & Feng Lu

INTERNATIONAL JOURNAL OF GEOGRAPHICAL INFORMATION SCIENCE
https://doi.org/10.1080/13658816.2024.2434665

RESEARCH ARTICLE

A tensor decomposition method based on embedded geographic meta-knowledge for urban traffic flow imputation

Xiaoyue Luo[a,b], Shifen Cheng[a,b], Lizeng Wang[a,b], Yuxuan Liang[c] and Feng Lu[a,b,d,e]

[a]State Key Laboratory of Resources and Environmental Information Systems, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing, China; [b]University of Chinese Academy of Sciences, Beijing, China; [c]Intelligent Transportation Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China; [d]The Academy of Digital China, Fuzhou University, Fuzhou, China; [e]Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing, China

ABSTRACT

Accurate and reliable traffic flow data are essential for intelligent transportation systems; however, limitations arising from hardware and communication costs often lead to missing data. Tensor decomposition is widely used to address these issues. However, existing imputation methods employ a fixed geographic feature similarity matrix to constrain the tensor decomposition process, which fails to accurately capture the spatial heterogeneity of traffic flows, thus limiting the imputation accuracy and robustness. This study proposes a tensor decomposition method embedded with geographic meta-knowledge (Meta-TD) to accurately determine the spatial heterogeneity of traffic flows. The key innovation is establishing a dynamic relationship between the geographic meta-knowledge and spatial heterogeneity of traffic flows, and then using the spatial heterogeneity of the traffic flows to constrain the tensor decomposition process. Experimental results based on real urban traffic flows demonstrated the superiority of Meta-TD over fifteen baseline models under random, block, and long time-series missing patterns, achieving reductions in MAE, RMSE, and MAPE of 6.97–97.05%, 3.33–94.68%, and 0.72–90.89%, respectively. Notably, Meta-TD maintained high accuracy for sudden changes in traffic flow states, evidencing its robustness to varying missing data rates and distribution patterns. This adaptability makes it highly suitable for complex and dynamic urban traffic environments.

ARTICLE HISTORY
Received 15 June 2024
Accepted 20 November 2024

KEYWORDS
Traffic flow imputation; geographic meta-knowledge; spatial weight matrix; tensor decomposition; spatial heterogeneity

1. Introduction

Accurate and reliable traffic flow data are fundamental for traffic engineering research and application of intelligent transportation systems. The acquisition of traffic flow data primarily relies on specialized fixed detection devices such as cameras and radar/laser speed detectors. However, the communication and data storage capabilities of

CONTACT Shifen Cheng chengsf@lreis.ac.cn

these devices are susceptible to extreme weather conditions, power supply issues, and equipment performance limitations, inevitably leading to missing data. This limitation hinders the accurate and reliable acquisition of traffic flow data within urban road networks (Jain and Oh 2014, Cheng *et al*. 2018).

Traffic flow imputation involves the application of interpolation methods to estimate small-scale occasional or large-scale periodic missing traffic data, and is mainly classified into two categories: statistical-based and machine learning-based methods (Balažević *et al*. 2019, Zhang *et al*. 2023, Xie *et al*. 2024). Statistical-based methods employ mathematical and statistical techniques to impute missing values using a straightforward and comprehensible structure. The algorithms commonly include probabilistic principal component models, Bayesian models, and kernel probability models (Li *et al*. 2013, Li *et al*. 2014, Benahmed & Houichi 2018, Thomas & Rajabi 2021, Nguyen *et al*. 2023). However, these methods often neglect the multidimensional spatiotemporal features inherent in traffic flows, making them suitable only for imputing small-scale episodic missing data. On the other hand, machine learning-based methods train models on large amounts of sample data, enabling them to capture complex correlations and nonlinear relationships among the data (You *et al*. 2024). These methods are suitable for imputing large-scale periodic missing traffic data, and have emerged as the mainstream approach (Liang *et al*. 2019, Sivakani & Ansari 2020, Yang *et al*. 2021). Matrix/tensor decomposition is an important branch of machine learning, particularly in the field of imputing traffic flow data. These methods utilize the low-rank attributes of the observed sample data to impute missing data by representing the data as a linear combination of low-rank approximations (Bansal *et al*. 2021, Ongie *et al*. 2021, Ganji *et al*. 2022, Chen *et al*. 2022). Previous studies have established similarity relationships among various road segments based on geographic features, such as the types and numbers of points of interest (POIs) (Meng *et al*. 2017). These relationships were used to construct a geographic feature similarity matrix to constrain the tensor decomposition process, thereby utilizing traffic flow data from other road segments to impute missing values (Wang *et al*. 2020, Zhang *et al*. 2023). However, the reliance on a fixed similarity matrix in these methods limits their ability to accurately capture the spatial heterogeneity of traffic flows, thus limiting the accuracy and robustness of traffic flow imputation.

In general, the similarity relationships of traffic flows can be represented as a graph structure, where the node attributes of the graph include the surrounding environmental features of the monitoring points, such as the types and numbers of POIs and road levels, and the edge attributes of the graph represent the relationships between the monitoring points, such as road connectivity and distance (Pan *et al*. 2019, Wang *et al*. 2023). Therefore, the spatial heterogeneity of traffic flows is influenced by the geographic features of nodes and edges. Existing tensor decomposition methods typically consider only the geographic features of nodes, while overlooking those of the edges. This limitation hinders the accurate modeling of spatial heterogeneity in traffic flows, thereby affecting the effectiveness of traffic flow imputation (Narita *et al*. 2012, Goulart *et al*. 2017). For example, these methods construct a similarity matrix based solely on the similarity of POIs around road segments without considering the actual connectivity or distance between these roads (Zhang *et al*. 2024). However, even if certain road segments have similar geographic features, their traffic flow patterns may differ due to the road connectivity and distance. More importantly, the spatial heterogeneity of traffic flows is dynamic and varies with changes in the traffic states of road segments. For example, during peak and off-peak periods, the traffic flow on a road segment can exhibit entirely different patterns which may not be reflected in the static geographic feature similarity matrix. Therefore, considering both the geographic features of nodes and edges as well as the dynamic traffic states during the tensor decomposition process is crucial for enhancing the accuracy and robustness of missing data imputation.

Therefore, we propose a tensor decomposition method embedded with geographic meta-knowledge (Meta-TD). Meta-knowledge refers to high-level semantic feature mappings derived from geographic attributes that influence the heterogeneity of traffic flows. It can be further divided into two categories, node and edge meta-knowledge, when expressed as a graph structure. Our main idea is to establish a dynamic relationship between geographic meta-knowledge and spatial heterogeneity of traffic flows, and then use the spatial heterogeneity to constrain the tensor decomposition process, thereby achieving accurate imputation of missing traffic flow data. The main contributions of this study are as follows:

- We designed a spatial weight matrix calculation method to account for the spatial heterogeneity of traffic flows. This method uses meta-learning to extract meta-knowledge from road configurations and then calculates the spatial weight matrix by integrating the meta-knowledge and traffic flow features through dynamic graph generation, thereby modeling the dynamic relationship between the meta-knowledge and spatial heterogeneity.
- We proposed a tensor decomposition method embedded with a spatial weight matrix. This method optimizes the tensor-solving process using a spatial weight matrix and enhances the imputation performance of missing traffic flow data in urban scenarios by collaboratively optimizing the spatial weight calculation and traffic flow imputation results.
- Extensive experiments on real urban traffic flow datasets have shown that the proposed method surpassed state-of-the-art baseline models in terms of imputing random missing, block missing, and long time-series missing traffic flow data. Notably, the proposed method maintained high accuracy even for sudden changes in traffic flow states, demonstrating its robustness to varying missing data rates and distribution patterns. This adaptability renders it suitable for use in complex and dynamic urban traffic environments.

2. Related work

2.1. Statistical learning-based methods

Common statistical learning models include historical averaging (HA) (Campbell & Thompson 2008), simple exponential smoothing (SES) (Gardner 2006), principal component analysis (PCA), probabilistic PCA (PPCA), kernel probabilistic PCA (KPPCA), and Bayesian PCA (BPCA) (Benahmed & Houichi 2018, Ke *et al.* 2019, Joelianto *et al.* 2022).

These methods typically consider traffic data as a matrix and impute missing data by constructing suitable statistical models. For example, Qu *et al*. (2009) proposed PPCA for missing data imputation, improving the accuracy by 25% compared to traditional PCA methods. Li *et al*. (2013) used PPCA to extract statistical features from known traffic data and constructed a sliding regression model to impute missing values. Li *et al*. (2014) used the PPCA method to construct a probability distribution model to capture the linear features in traffic flows, and demonstrated good performance in both random and mixed missing patterns of a single monitoring point. Additionally, methods such as KPPCA and BPCA have been proposed to more effectively capture the nonlinear features of traffic flows thereby improving the imputation accuracy (Shao & Chen 2018, Smith and Climer 2024, Xie *et al*. 2024). Although statistical-based methods have achieved good performance, they convert traffic data into matrices, ignoring the inherent multidimensional spatiotemporal features. Therefore, these methods are more suitable for the imputation of small-scale missing traffic flow data.

2.2. Machine learning-based methods

Machine learning-based methods can effectively model complex correlations and nonlinear relationships in extensive sample data through training, making them more suitable for imputing large amounts of missing data. In recent years, tensor decomposition methods, as an important branch of unsupervised learning, have garnered significant attention in the field of traffic flow imputation. According to the principles of tensor decomposition, current methods can be divided into CANDECOMP/PARAFAC (CP) decomposition (Wu *et al*. 2017, Battaglino *et al*. 2018, Zhu *et al*. 2022), Tucker decomposition (Li *et al*. 2014, Goulart *et al*. 2017), and low-rank decomposition based on nuclear norms (Tang *et al*. 2020, Nie *et al*. 2022). For example, Sure *et al*. (2022) introduced low-rank attributes into the tensor decomposition process to capture the spatiotemporal correlation of traffic flow data, thereby enhancing tensor decomposition accuracy. Xu *et al*. (2023) decomposed historical tensor data into factor matrices and a core tensor using Tucker decomposition, and then applied time series and Laplace regularization to constrain the tensor decomposition process from temporal and spatial perspectives, respectively, achieving traffic flow imputation under different missing data patterns. These studies typically assume that monitoring points within a specific area exhibit similar traffic flow trends and can rely on flow information from upstream and downstream monitoring points to impute missing values. However, maintaining high completeness and consistency in imputation results is challenging when observations at previous times or during the same historical period are missing at the monitoring points (Wu *et al*. 2022). Therefore, some studies constrain the tensor decomposition process using a geographic feature similarity matrix and employ traffic flow data from non-adjacent sections to impute missing values (Jia *et al*. 2021, Suleiman *et al*. 2022). For example, Said & Erradi (2022) used POI data to construct a geographic feature similarity matrix and embed it into CP decomposition. Huang *et al*. (2022) integrated POI data and traffic congestion state data to construct a geographic similarity matrix that constrained the tensor decomposition process, thereby improving the accuracy of missing data imputation. However, the fixed geographic feature similarity matrix used in the current study fails to reflect dynamic changes in traffic flows, limiting the imputation accuracy and robustness.

Furthermore, with the rapid development of Graph Neural Networks (GNNs), new methods have emerged for capturing spatial heterogeneity in traffic flow data (Khaled *et al.* 2022). For example, Wang *et al.* (2022) proposed a multi-view bidirectional spatiotemporal graph network (Multi BiSTGN) that captures dynamic changes in traffic flows from various temporally correlated perspectives. Li *et al.* (2023) developed a hierarchical spatio-temporal graph convolutional neural network (HSTGCN) to analyze dynamic relationships among traffic flows across various road levels, and then impute the missing values. Zong *et al.* (2024) proposed the Dynamic Attention Generating Adversarial Network (DATGAN), designed to enhance imputation accuracy by capturing spatial heterogeneity within the data. These methods primarily capture spatial heterogeneity of traffic flows from a data-driven perspective. However, when traffic flow data is incomplete, the accuracy of spatial heterogeneity extraction is compromised, subsequently affecting the overall performance of data imputation. Therefore, it is essential to effectively integrate geographic features with traffic flow features to accurately represent the spatial heterogeneity of traffic flows, thereby enhancing the accuracy and robustness of data imputation.

3. Preliminary

This section introduces the fundamental concepts of graph and traffic flows, and the imputation problems that need to be addressed. Additionally, it adopts a consistent naming convention for tensor-related notation, using handwritten $\mathcal{X}$ for the tensor, boldface capital letters for the matrix (second-order tensor), $\mathcal{X}_{ijk}$ for the elements of the third-order tensor, $||\mathrm{X}||$ for the sum of the squares of all elements in the tensor, and $\mathcal{X} \times_n \boldsymbol{Q}$ for the product of the n-module tensor $\mathcal{X}$ and matrix $\boldsymbol{Q}$.

Definition 1 (Traffic volume). The traffic volume refers to the number of vehicles passing through a road section at monitoring points at specific time intervals. The traffic flow data for N monitoring points over D days and T time intervals can be represented as a tensor $\mathcal{X} \in \mathbb{R}^{N \times D \times T}$.

Definition 2 (Graph). A graph $G = (V, E)$ represents the traffic monitoring points and their relationships, where $V = \{v_1, \ldots, v_N\}$ and $E = \{e_{ij} | i \le N, j \le N\}$ are the sets of nodes and edges in the graph, respectively. The node attributes are determined by the surrounding environment of the monitoring point and are expressed as a feature vector $v_i = \{v_1^i, ., v_f^i | i < N; f < c\}$, where v_f^i denotes the number of various types of POIs around node i, the road width, the number of intersections, and other attributes, and NF denotes the number of attributes. Edge attributes are determined by the relationship between the monitoring points and are expressed as a feature vector $e_{ij} = \left\{\left(e_1^{ij}, ., e_k^{ij}\right) | k < c\right\}$, where e_k^{ij} denotes features such as the distance and association strength between nodes i and j, and c denotes the number of features.

Problem Statement. Given the traffic flow data $\mathcal{X} \in \mathbb{R}^{N \times D \times T}$ with missing values and the graph $G = (V, E)$, the goal of this research is to construct a model to impute the missing values in $\mathcal{X}$.

4. Methodology

The proposed method is divided into three parts (Figure 1). First, meta-learning is used to extract geographic meta-knowledge from the node and edge attributes that affect traffic flow features. Second, a spatial weight matrix calculation method is designed to account for spatial heterogeneity. This involves integrating meta-knowledge with traffic flow features to generate dynamic graphs for constructing a spatial weight matrix between the monitoring points. Finally, a tensor decomposition method embedded with a spatial weight matrix is developed. This method utilizes spatial weight matrices to constrain the tensor decomposition process and continuously optimizes the spatial weight calculation and traffic flow imputation results through collaborative learning.

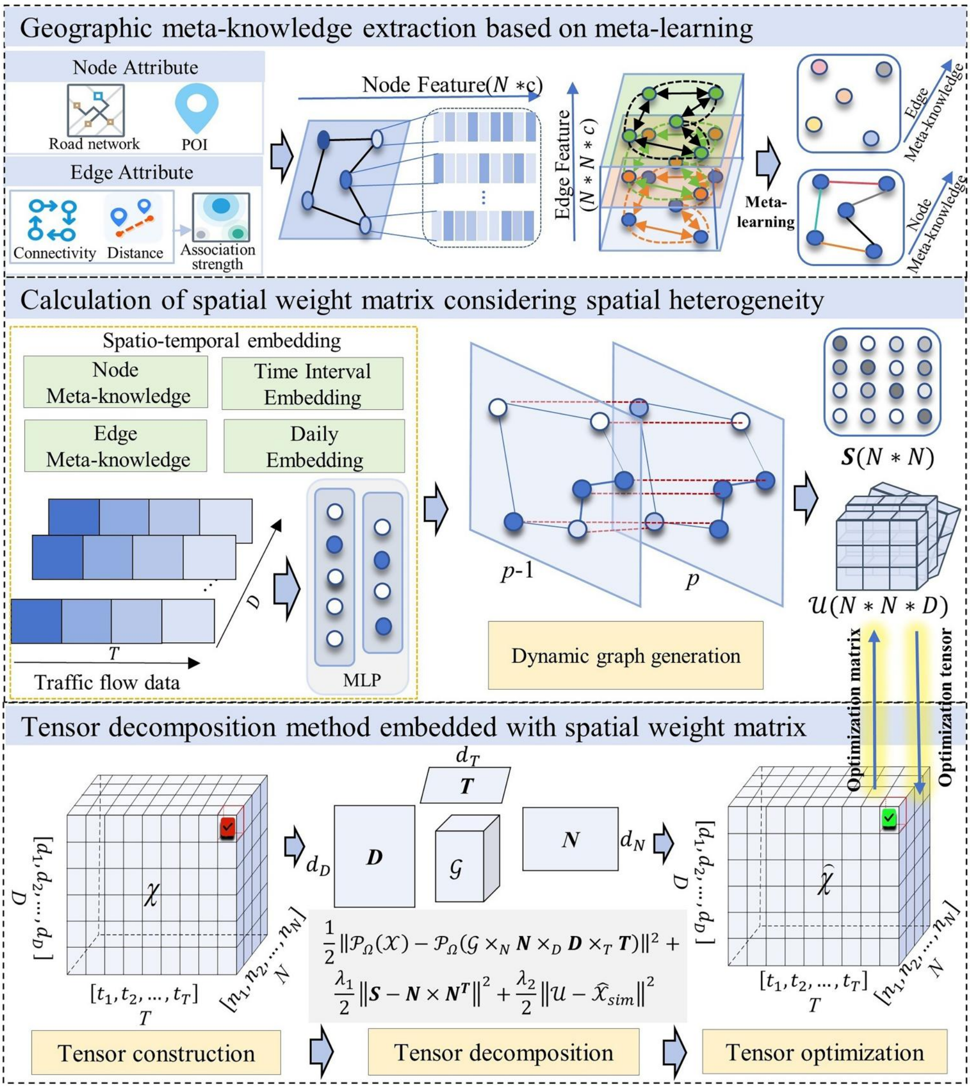

Figure 1. Framework of the proposed method.

4.1. Geographic meta-knowledge extraction based on meta-learning

In a complex urban traffic system, the similarity of traffic flow between monitoring points can be represented by a graph structure that is influenced by both node and edge attributes (Jiang *et al*. 2023). However, directly using the attributes of the nodes and edges as input features may lead to redundant information and noise, complicating the task of capturing the complex relationships and patterns inherent in the graph structure. Therefore, we designed a geographic meta-knowledge extraction module based on meta-learning to extract node and edge meta-knowledge from their respective attributes, enabling the mapping of high-level semantic features. For each node i in the graph, we constructed a 100-meter radius buffer centered on its spatial coordinates to obtain the number of POIs and each type around that node (Pan *et al*. 2019), including companies, living services, transportation facilities, etc. Then, we calculated the Euclidean distance between node i and each type of POIs, summed these distances, and normalized the result to serve as the features of node i. For each edge e_{ij} in the graph, the attributes comprised association strength features, such as the distance and connectivity between nodes. The distance feature is quantified using the Euclidean distance between monitoring points, representing their spatial proximity. The connectivity feature is quantified using an adjacency matrix, where elements corresponding to connected nodes are assigned a value of 1, and those for disconnected nodes are assigned a value of 0.

Based on this, we employed the Multilayer Perceptron (MLP) as a meta-learner to extract meta-knowledge from node and edge attributes. The model achieved this through the following formulation:

$$h_v = MLP(v_i) \tag{1}$$

$$h_e = MLP(e_{ij}) \tag{2}$$

where $h_v \in \mathbb{R}^{N\times c}$ and $h_e \in \mathbb{R}^{N\times N\times c}$ represent the node meta-knowledge and edge meta-knowledge, respectively. Here, c denotes the embedding dimension of the features.

4.2. Calculation of spatial weight matrix considering spatial heterogeneity

The spatial heterogeneity relationship of traffic flows is influenced by both node and edge meta-knowledge and varies with changing traffic states of road segments. The traffic flow state between monitoring points can be characterized at both the global and local scales. At the local scale, traffic flows exhibit periodic seasonal patterns. For instance, on weekdays, traffic flows typically increase during peak hours in the morning and afternoon, whereas on weekends, peak periods tend to be more stable. At the local scale, traffic flows may experience fluctuations or disturbances within a specific time range due to events or special circumstances (Bhanu *et al*. 2021). To accurately capture the spatial heterogeneity of traffic flows across different timescales using meta-knowledge and traffic flow features, we designed a spatial weight matrix calculation module. This module takes into account spatial heterogeneity and employs dynamic graph generation to effectively aggregate node and edge meta-knowledge.

Specifically, to capture the dynamic spatial heterogeneity of traffic flows over time, we introduced the learnable embedding $F_p^D \in \mathbb{R}^{D\times c}$ and $F_p^T \in \mathbb{R}^{T\times c}$, which represented the daily features and time interval features of traffic flows. We then expanded the dimensions of h_v to $N \times 1 \times 1 \times c$, F_p^D to $1 \times D \times 1 \times c$, and F_p^T to $1 \times 1 \times T \times c$. Since these features shared, the same last dimension c after dimension expansion, they can be multiplied element-wise to obtain a spatiotemporal embedding $E_p^{st} \in \mathbb{R}^{N\times D\times T\times c}$. The motivation behind this fusion method was to weigh the meta-knowledge derived from node attributes, aligning it with the varying characteristics of traffic flows to establish a dynamic relationship between geographic meta-knowledge and the spatial heterogeneity of traffic flows. The spatiotemporal embedding $E_p^{st} \in \mathbb{R}^{N\times D\times T\times c}$ can be formulated as follows:

$$E_p^{st} = h_v \odot F_p^D \odot F_p^T \tag{3}$$

where F_p^D and F_p^T denote the daily embedding and time interval embedding, respectively, which are determined by the daily index p. Besides, a dimension expansion strategy is employed before the element-wise product operation to maintain consistency in dimension.

Traffic flows $\mathcal{X}$ were input into the MLP layer to accurately characterize the dynamic changes in traffic flow features. The formula used is as follows:

$$I_p = MLP(\mathcal{X}_p) \tag{4}$$

where I_p represents the output features from the MLP layer that encapsulates the dynamic features of the traffic flows at daily index p. By performing an element-wise multiplication of I_p with E_p^{st}, we generated the dynamic graph embedding $E_p^h \in \mathbb{R}^{N\times c}$. The formula used is as follows:

$$E_p^h = \sum\nolimits_T I_p \odot E_p^{st} \tag{5}$$

Subsequently, the dynamic graph embeddings E_p^h were multiplied by their transpose $E_p^{h^T}$ and integrated with edge meta-knowledge h_e to infer heterogeneous relationships between nodes. This methodology facilitated the construction of a dynamic graph that encapsulated both meta-knowledge and spatial heterogeneity. Through this process, dynamic and static features were effectively combined to provide a comprehensive representation of the heterogeneous relationships inherent in traffic flows. To facilitate the calculations for subsequent tensor decomposition, we normalized the features of the dynamic graph to generate the spatial weight matrix $A_p^h \in \mathbb{R}^{N\times N}$.

$$A_p^h = ReLU\left(E_p^h E_p^{h^T} \odot h_e\right) \tag{6}$$

where p incrementally increased from 1 to D, the resulting spatial weight matrices collectively formed a tensor $\mathcal{U} \in \mathbb{R}^{N\times N\times D}$, encapsulating the spatial weight information across different locations and days. Additionally, $\boldsymbol{S} = \sum_p A_p^h / p$ denotes the spatial weight matrix for different locations of the N monitoring points.

4.3. Tensor decomposition method embedded with spatial weight matrix

The tensor decomposition model decomposes a high-order tensor into the product of a core tensor and multiple-factor matrices. This model can handle high-dimensional sparse data and extract hidden features, making it suitable for addressing the problem of traffic flow imputation (Kolda & Bader 2009). Traditional tensor decomposition methods typically utilize a static geographic feature similarity matrix for constraints, overlooking the geographic features of nodes and edges as well as the dynamic traffic state, which affects the accuracy and robustness of missing data imputation. Therefore, we proposed a tensor decomposition method embedded with spatial weight matrix, which utilizes the Tucker model to decompose $\mathcal{X}$ into a core tensor $\mathcal{G} \in \mathbb{R}^{d_N \times d_D \times d_T}$ and three factor matrices $\boldsymbol{N} \in \mathbb{R}^{N \times d_N}$, $\boldsymbol{D} \in \mathbb{R}^{D \times d_D}$, and $\boldsymbol{T} \in \mathbb{R}^{T \times d_T}$ (Malik and Becker 2018). Here, $\boldsymbol{N} \in \mathbb{R}^{T \times d_N}$ is the spatial factor matrix of N monitoring points, $\boldsymbol{D} \in \mathbb{R}^{D \times d_D}$ and $\boldsymbol{T} \in \mathbb{R}^{T \times d_T}$ are temporal factor matrices, $\boldsymbol{D} \in \mathbb{R}^{D \times d_D}$ can reflect the flow features between different days, $\boldsymbol{T} \in \mathbb{R}^{T \times d_T}$ can reflect the flow features within T time intervals of a day, and the core tensor $\mathcal{G} \in \mathbb{R}^{d_D \times d_N \times d_T}$ represents the variation of flow in different spaces and times. Based on this, we established the index set Ω of known traffic flows. Then, the set of known traffic flows of $\mathcal{X}$ can be represented as $\mathcal{P}_\Omega(\mathcal{X})$, as shown in the following formula:

$$[\mathcal{P}_\Omega(\mathcal{X})]_{ijk} = \begin{cases} \mathcal{X}_{ijk}, & if(i,j,k) \in \Omega \\ 0, & otherwise \end{cases} \tag{7}$$

The objective function controlling the decomposition error of the core tensor and factor matrices is shown in Equation (8).

$$\begin{aligned} \mathcal{L}(\mathcal{G}, \boldsymbol{N}, \boldsymbol{D}, \boldsymbol{T}) = & \frac{1}{2} \|\mathcal{P}_\Omega(\mathcal{X}) - \mathcal{P}_\Omega(\mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T})\|^2 \\ & + \frac{\lambda}{2}\left(\|\mathcal{G}\|^2 + \|\boldsymbol{N}\|^2 + \|\boldsymbol{D}\|^2 + \|\boldsymbol{T}\|^2\right) \end{aligned} \tag{8}$$

where $\|.\|^2$ denotes the l_2 paradigm; $\lambda/2(\|\mathcal{G}\|^2 + \|\boldsymbol{D}\|^2 + \|\boldsymbol{N}\|^2 + \|\boldsymbol{T}\|^2)$ is the regularization penalty term, which controls the decomposition error; and λ is a parameter controlling the degree of the penalty of the regularization term. $\mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T}$ represents the reconstructed flow value, while $\mathcal{P}_\Omega(\mathcal{X}) - \mathcal{P}_\Omega(\mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T})$ denotes the difference between the non-missing position and the corresponding position after reconstruction. By minimizing the objective function, the optimized core tensor $\mathcal{G} \in \mathbb{R}^{d_N \times d_D \times d_T}$ and factor matrices $\boldsymbol{D} \in \mathbb{R}^{D \times d_D}$, $\boldsymbol{N} \in \mathbb{R}^{N \times d_N}$ and $\boldsymbol{T} \in \mathbb{R}^{T \times d_T}$ can be obtained, and the missing values in $\mathcal{X}$ can be filled using Equation (9); the process is shown in Figure 2. First, multiply the factor matrix $\boldsymbol{N}$ with the matrix $\mathcal{G}(1)$ of the core tensor $\mathcal{G}$ expanded along mode-1 to obtain $\mathcal{G}' \in \mathbb{R}^{N \times d_D \times d_T}$. Based on this, multiply the factor matrix $\boldsymbol{D}$ with the matrix $\mathcal{G}(2)$ of $\mathcal{G}'$ expanded along mode-2 to obtain $\mathcal{G}'' \in \mathbb{R}^{N \times D \times d_T}$. Lastly, multiply the factor matrix $\boldsymbol{T}$ with the matrix $\mathcal{G}(3)$ of $\mathcal{G}''$ expanded along mode-3 to obtain the complemented tensor $\widehat{\mathcal{X}} \in \mathbb{R}^{N \times D \times T}$.

$$\mathcal{P}_\Omega(\widehat{\mathcal{X}}) \approx \mathcal{P}_\Omega(\mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T}) \tag{9}$$

where $\times$ represents matrix multiplication and $\times_D$ represents the product of the tensor and matrix.

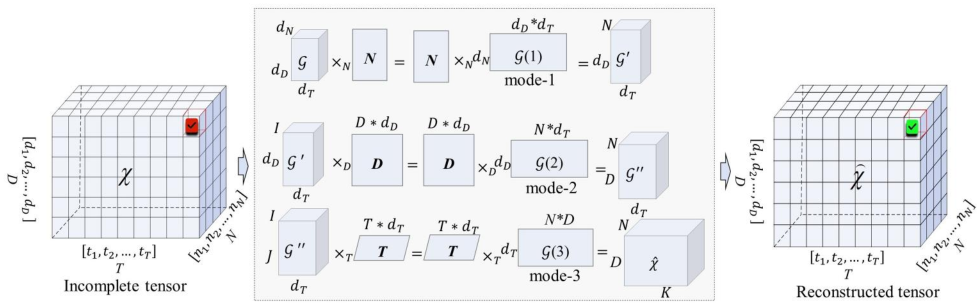

Figure 2. Overview of the tensor decomposition and reconstruction process: The incomplete tensor $\mathcal{X}$ is decomposed into a core tensor $\mathcal{G}$ and three factor matrices $\boldsymbol{N}$, $\boldsymbol{D}$, and $\boldsymbol{T}$, which are then used to reconstruct the tensor to obtain $\widehat{\mathcal{X}}$.

Additionally, $\boldsymbol{S} \in \mathbb{R}^{N \times N}$ and $\mathcal{U} \in \mathbb{R}^{N \times N \times D}$ obtained in Section 4.2 were used to constrain the process of tensor decomposition, where $\boldsymbol{S} \in \mathbb{R}^{N \times N}$ represents the spatial weight matrices of different monitoring points, $\boldsymbol{S}$ can be converted to $\boldsymbol{N}^{N \times d_N} \times (\boldsymbol{N}^{N \times d_N})^T$, and $\mathcal{U} \in \mathbb{R}^{N \times N \times D}$ contains the spatial weight matrices in different time intervals. The spatial weights $\widehat{\mathcal{X}}_{sim}$ of $\widehat{\mathcal{X}}$ at different time intervals were calculated using Equation (10). Therefore, the objective function for the imputation of $\mathcal{X}$ can be transformed into Equation (11):

$$\widehat{\mathcal{X}}_{sim} = \left(\frac{\hat{\mathcal{X}}}{\sqrt{\sum_{j=1}^{J} \widehat{\mathcal{X}}_{(:,j,:)}}} \times \left(\frac{\widehat{\mathcal{X}}}{\sqrt{\sum_{j=1}^{J} \widehat{\mathcal{X}}_{(:,j,:)}}} \right)' \right)_{iij} \tag{10}$$

$$\begin{aligned}\mathcal{L}(\mathcal{G}, \boldsymbol{N}, \boldsymbol{D}, \boldsymbol{T}, S, \mathcal{U}) = {} & \frac{1}{2} \| \mathcal{P}_{\Omega}(\mathcal{X}) - \mathcal{P}_{\Omega}(\mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T}) \|^2 + \frac{\lambda_1}{2} \| S - \boldsymbol{N} \times \boldsymbol{N}^T \|^2 \\ & + \frac{\lambda_2}{2} \| \mathcal{U} - \widehat{\mathcal{X}}_{sim} \|^2 + \frac{\lambda_3}{2} \left(\|\mathcal{G}\|^2 + \|\boldsymbol{N}\|^2 + \|\boldsymbol{D}\|^2 + \|\boldsymbol{T}\|^2 \right)\end{aligned} \tag{11}$$

Here, $\| \mathcal{P}_{\Omega}(\mathcal{X}) - \mathcal{P}_{\Omega}(\mathcal{G} \times {}_N \boldsymbol{N} \times {}_D \boldsymbol{D} \times {}_T \boldsymbol{T}) \|^2$ controls the decomposition error of $\mathcal{X}$, $\| \boldsymbol{S} - \boldsymbol{N} \times \boldsymbol{N}^T \|^2$ controls the decomposition error of the factor S, which was used to constrain the relationship of the flow at different locations; $\| \mathcal{U} - \widehat{\mathcal{X}}_{sim} \|$ controls the decomposition error of the factor $\mathcal{U}$, which was used to constrain the relationship of the flow at different times; $\|\mathcal{G}\|^2 + \|\boldsymbol{N}\|^2 + \|\boldsymbol{D}\|^2 + \|\boldsymbol{T}\|^2$ is a regularization term, which was used to prevent the objective function from being overfitted; and λ_1, λ_2, and λ_3 are the weights for controlling the regularization term.

To obtain the updated core tensor and factor matrices, we used the conjugate gradient descent method to solve the objective function (Golub and Van Loan 2013). This allowed separately obtaining the least squares estimation approximate solutions for the partial derivatives of $\mathcal{G}$, $\boldsymbol{N}, \boldsymbol{D}, \boldsymbol{T}, \boldsymbol{S}$, and $\mathcal{U}$. The calculations were as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathcal{G}} = {} & -\mathcal{P}_{\Omega}(\mathcal{X} - \mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T}) \times_N \boldsymbol{N}^T \times_D {}^T \times_T \boldsymbol{T}^T \\ & -\lambda_2 (\mathcal{U} - \boldsymbol{N} \mathcal{G} (\boldsymbol{D} \otimes \boldsymbol{T})^T (\boldsymbol{D} \otimes \boldsymbol{T}) \mathcal{G}^T)(\boldsymbol{D} \otimes \boldsymbol{T})^T \boldsymbol{N}^T + \lambda_3 \mathcal{G}\end{aligned} \tag{12}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial N} = {} & -\mathcal{P}_{\Omega}\left(\mathcal{X}_{(1)} - \boldsymbol{N} \mathcal{G}_{(1)} (\boldsymbol{D} \otimes \boldsymbol{T})^T \right) (\boldsymbol{D} \otimes \boldsymbol{T}) \mathcal{G}_{(1)}{}^T + \lambda_1 (\boldsymbol{N}\boldsymbol{N}^{\boldsymbol{T}} - \boldsymbol{S}) \boldsymbol{N} \\ & -\lambda_2 (\mathcal{U} - \boldsymbol{N} \mathcal{G}_{(1)} (\boldsymbol{D} \otimes \boldsymbol{T})^T (\boldsymbol{D} \otimes \boldsymbol{T}) \mathcal{G}_{(1)}{}^T) \mathcal{G}_{(1)} (\boldsymbol{D} \otimes \boldsymbol{T})^T (\boldsymbol{D} \otimes \boldsymbol{T}) \mathcal{G}_{(1)}{}^T + \lambda_3 \boldsymbol{N}\end{aligned} \tag{13}$$

$$\begin{gathered}\frac{\partial \mathcal{L}}{\partial D} = -\mathcal{P}_\Omega\left(\mathcal{X}_{(2)} - \boldsymbol{D}\mathcal{G}_{(2)}(\boldsymbol{N} \otimes \boldsymbol{T})^T\right)(\boldsymbol{N} \otimes \boldsymbol{T})\mathcal{G}_{(2)}{}^T \\ -\lambda_2\left(\mathcal{U} - \boldsymbol{N}\mathcal{G}_{(2)}(\boldsymbol{D} \otimes \boldsymbol{T})^T(\boldsymbol{D} \otimes \boldsymbol{T})\mathcal{G}_{(2)}{}^T\right)\left(\boldsymbol{N}\mathcal{G}_{(2)}\right)^T(\mathcal{G}_{(2)} \otimes \boldsymbol{T}^T) + \lambda_3\boldsymbol{D}\end{gathered} \tag{14}$$

$$\begin{gathered}\frac{\partial \mathcal{L}}{\partial T} = -\mathcal{P}_\Omega\left(\mathcal{X}_{(3)} - \boldsymbol{T}\mathcal{G}_{(3)}(\boldsymbol{N} \otimes \boldsymbol{D})^T\right)(\boldsymbol{N} \otimes \boldsymbol{D})\mathcal{G}_{(3)}{}^T \\ -\lambda_2\left(\mathcal{U} - \boldsymbol{N}\mathcal{G}_{(3)}(\boldsymbol{D} \otimes \boldsymbol{T})^T(\boldsymbol{D} \otimes \boldsymbol{T})\mathcal{G}_{(3)}{}^T\right)\boldsymbol{D}\mathcal{G}_{(3)}(\boldsymbol{D} \otimes \boldsymbol{T}) + \lambda_3\boldsymbol{T}\end{gathered} \tag{15}$$

$$\frac{\partial \mathcal{L}}{\partial S} = \lambda_1(\boldsymbol{S} - \boldsymbol{N}\boldsymbol{N}^{\boldsymbol{T}}) \tag{16}$$

$$\frac{\partial \mathcal{L}}{\partial \mathcal{U}} = \lambda_2\left(\mathcal{U} - \boldsymbol{N}\mathcal{G}(\boldsymbol{D} \otimes \boldsymbol{T})^T(\boldsymbol{D} \otimes \boldsymbol{T})\mathcal{G}^T\right) \tag{17}$$

where $\otimes$ denotes the Kronecker product of matrices, $\mathcal{X}_{(1)} \approx \boldsymbol{N}\mathcal{G}_{(1)}(\boldsymbol{D} \otimes \boldsymbol{T})^T$ is the matrix of the tensor expanded along the $\boldsymbol{N}$ dimension, $\mathcal{X}_{(2)} \approx \boldsymbol{D}\mathcal{G}_{(2)}(\boldsymbol{N} \otimes \boldsymbol{T})^T$ is the matrix of the tensor expanded along the $\boldsymbol{D}$ dimension, $\mathcal{X}_{(3)} \approx \boldsymbol{T}\mathcal{G}_{(3)}(\boldsymbol{N} \otimes \boldsymbol{D})^T$ is the matrix of the tensor expanded along the $\boldsymbol{T}$ dimension, and $\mathcal{G}_{(1)}$, $\mathcal{G}_{(2)}$ and $\mathcal{G}_{(3)}$ stand for the matrices of the core tensor expanded along the $\boldsymbol{N}$, $\boldsymbol{D}$ and $\boldsymbol{T}$ dimensions, respectively.

4.4. Algorithm and training

As shown in Algorithm 1, the core tensor $\mathcal{G}$ and the three factor matrices $\boldsymbol{N}$, $\boldsymbol{D}$ and $\boldsymbol{T}$ were initialized. The meta-knowledge of the nodes and edges extracted from the meta-learning module was used as the input for the spatial weight matrix imputation module to obtain the initial values of the spatial weight matrices $\boldsymbol{S}$ and $\mathcal{U}$ (lines 1–2). Based on this initialization, the core tensor $\mathcal{G}$ and three factor matrices $\boldsymbol{N}$, $\boldsymbol{D}$ and $\boldsymbol{T}$ were updated by the tensor decomposition module, embedding the spatial weight matrices $\boldsymbol{S}$ and $\mathcal{U}$ as constraints to control the decomposition error (line 3–11). The iterative process calculated the loss value $Loss_{epoch}$ between the imputation result $\widehat{\mathcal{X}}$ and original tensor $\mathcal{X}$. The loss value was back-propagated to each layer of the meta-learning module, optimizing the calculation of the spatial weight matrices, and dynamically updating the spatial weight matrices $\boldsymbol{S}$ and $\mathcal{U}$ (lines 12–16). To summarize, the time complexity and space complexity of the Meta-TD algorithm primarily focused on three core modules: geographic meta-knowledge extraction, calculation of spatial weight matrix, and tensor decomposition. Specifically, the geographic meta-knowledge extraction module exhibited a time complexity of $O(N^2 \cdot c)$ and a space complexity of $O(N \cdot c + N^2 \cdot c)$, where N is the number of nodes and c is the dimensions of the embedded features. The calculation of the spatial weight matrix module had a time complexity of $O(N \cdot D \cdot T \cdot c)$ and a space complexity of $O(N^2 \cdot c)$. The tensor decomposition module demonstrated a time complexity of $O(k(N \cdot D \cdot T \cdot d_D \cdot d_N \cdot d_T))$ and a space complexity of $O(N \cdot D \cdot T + d_D \cdot d_N \cdot d_T)$, where d_D, d_N and d_T corresponded to the three dimensions of the core tensor, and k signified the number of iterations required to solve the objective function. Consequently, the overall time complexity of the Meta-TD algorithm could be expressed as $O(N \cdot (D \cdot T \cdot c + k \cdot D \cdot T \cdot d_D \cdot d_N \cdot d_T))$, while the space complexity was represented as $O(N^2 \cdot c + N \cdot D \cdot T + d_D \cdot d_N \cdot d_T)$.

Algorithm 1: Tensor decomposition module considering spatial heterogeneity

Input: Tensor $\mathcal{X}$, spatial weight matrix $\boldsymbol{S}$, $\mathcal{U}$, a loss threshold ε, regularized term weights λ_1, λ_2 and λ_3, learning rate η, an epoch threshold *num*

Output: $\mathcal{G}$, $\boldsymbol{N}$, $\boldsymbol{D}$, $\boldsymbol{T}$, $\boldsymbol{S}$, $\mathcal{U}$, $\widehat{\mathcal{X}}$

1. **Initialize** $\mathcal{G} \in \mathbb{R}^{d_N \times d_D \times d_T}$ 、$\boldsymbol{D} \in \mathbb{R}^{D \times d_D}$ 、$\boldsymbol{N} \in \mathbb{R}^{N \times d_N}$ 、$\boldsymbol{T} \in \mathbb{R}^{T \times d_T}$
2. Calculate $\boldsymbol{S}$,$\mathcal{U}$ by equation (6)
3. $epoch = 1$
4. **While** $|Loss_{epoch} - Loss_{epoch+1}| > \varepsilon$ and $epoch < num$
5. Solve equation (11) using the conjugate gradient method
6. Update $\boldsymbol{N}$ by equation (12)
7. Update $\boldsymbol{D}$ by equation (13)
8. Update $\boldsymbol{T}$ by equation (14)
9. Update $\mathcal{G}$ by equation (15)
10. Update $\boldsymbol{S}$ by equation (16)
11. Update $\mathcal{U}$ by equation (17)
12. $\widehat{\mathcal{X}} = \mathcal{G} \times_N \boldsymbol{N} \times_D \boldsymbol{D} \times_T \boldsymbol{T}$
13. $Loss_{epoch} = \mathcal{X} - \widehat{\mathcal{X}}$
14. **Updates** $\boldsymbol{S}$,$\mathcal{U}$ by equation (6)
15. $epoch = epoch + 1$
16. **Return** $\widehat{\mathcal{X}}$, $\mathcal{G}$, $\boldsymbol{N}$, $\boldsymbol{D}$, $\boldsymbol{T}$, $\boldsymbol{S}$,$\mathcal{U}$

5. Experiments

In this section, we evaluate the performance of the Meta-TD based on real traffic datasets to answer the following research questions (RQs):

RQ1. How does the imputation performance of Meta-TD vary under different missing patterns?

RQ2. Is Meta-TD sensitive to parameter selection?

RQ3. Are the designs of each component in Meta-TD effective?

RQ4. Is the geographic meta-knowledge in Meta-TD effective?

RQ5. How does the robustness of the Meta-TD method manifest?

5.1. Data description

5.1.1. Data source

The performance of Meta-TD was evaluated based on real urban traffic flow data collected in Wuhan, China. Data were collected using automatic license plate recognition technology with 67 surveillance cameras, with each camera treated as a traffic monitoring point with a unique identifier. As shown in Figure 3, the monitoring points

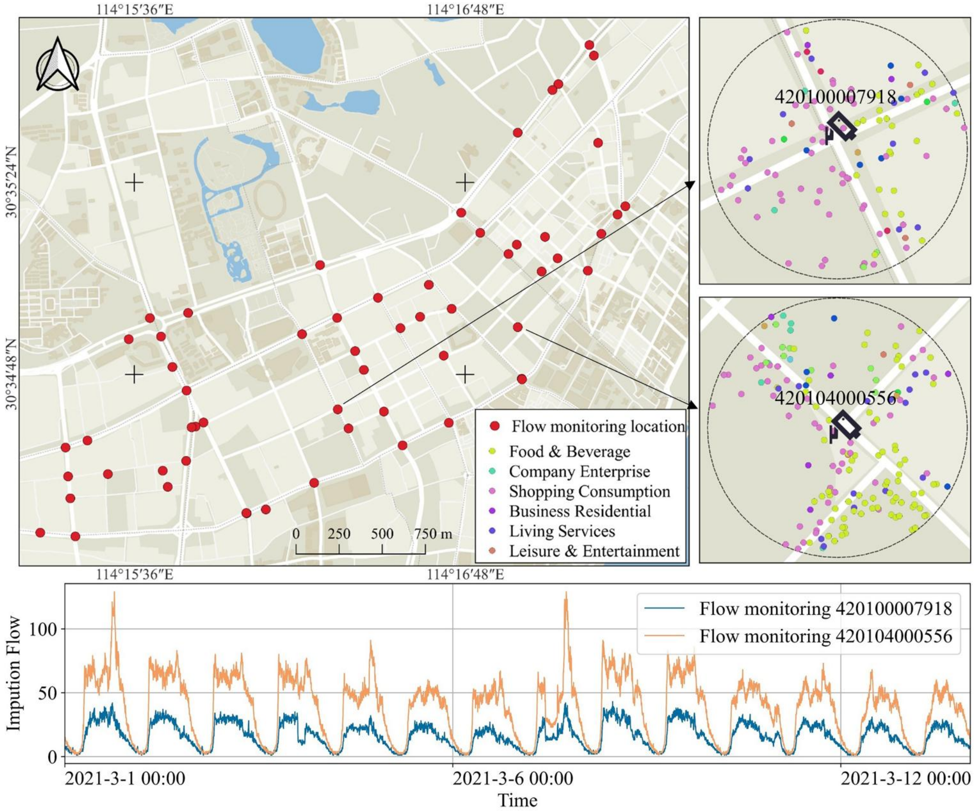

Figure 3. Distribution of traffic monitoring points and points of interest.

Table 1. Example of data from a single traffic monitoring point.

Monitoring point ID	Time	Longitude	Latitude	Traffic flow
4201****	2021-3-6 00:00-00:05	114.1**	30.6**	30
4201****	2021-3-6 00:05-00:10	114.1**	30.6**	23
… ….	… ….	… ….	… ….	… ….
4201****	2021-3-6 23:55-00:00	114.1**	30.6**	10

were located within the second ring of Wuhan, with each point surrounded by multiple commercial and residential areas. The time span of the traffic flow data was from March 01, 2021, to March 28, 2021, and the time window size was 5 min (Wang *et al.* 2023). Table 1 presents the data from a single traffic monitoring point, with each record including the monitoring point ID, time interval, longitude, latitude, and traffic flows. In addition, the geographic meta-knowledge required for Meta-TD was extracted from POIs and road networks, where POIs include 14 categories such as restaurants and food, shopping and consumption, living services, transportation facilities, and leisure and entertainment; and road networks include five types such as urban arterials, secondary arterials, and intersections.

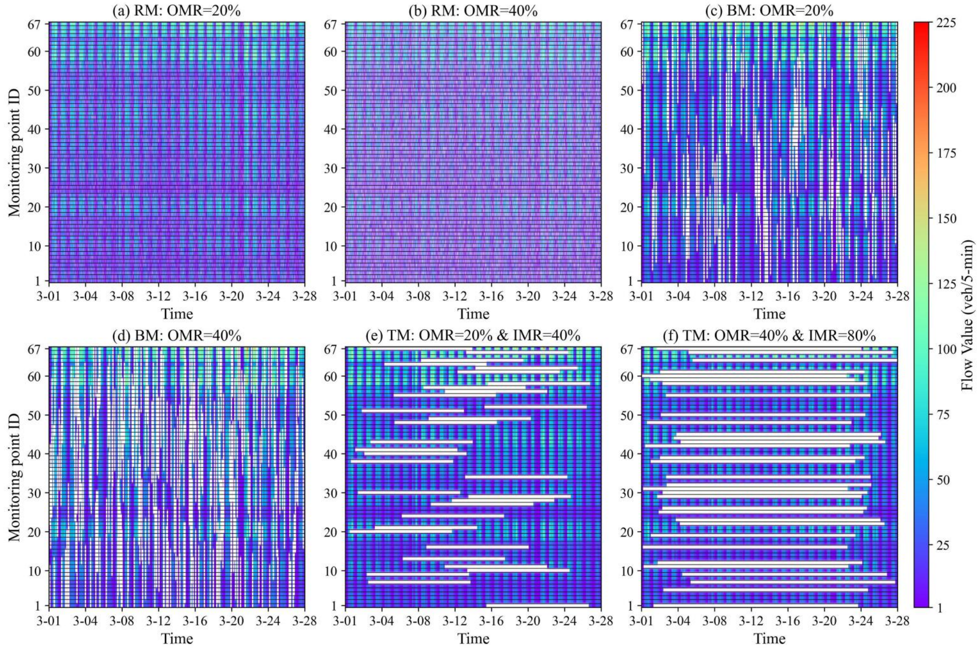

Figure 4. Traffic flow data under different missing patterns.

5.1.2. Data preprocessing

To evaluate the imputation performance of Meta-TD for traffic flow data under various missing patterns, three specific patterns were defined: random missing (RM), block missing (BM), and long time-series missing (TM). Each pattern included two overall missing rates (OMR) of 20% and 40%, as shown in Figure 4. In addition, to better assess the performance of the algorithm under the TM pattern, we imposed constraints on the missing rate at individual traffic monitoring points (IMRs): with an OMR of 20%, the IMR was set to 40% and with an OMR of 40%, the IMR was set to 80%.

5.2. Evaluation metrics

The accuracy of the traffic flow imputation results was evaluated using widely used quantitative metrics, including mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE). These metrics are defined as follows.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\mathcal{X}_i - \widehat{\mathcal{X}}_i\right| \tag{18}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|\mathcal{X}_i - \widehat{\mathcal{X}}_i\right|^2} \tag{19}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\mathcal{X}_i - \widehat{\mathcal{X}}_i}{\mathcal{X}_i}\right| \tag{20}$$

where N denotes the number of missing instances, $\widehat{\mathcal{X}}_i$ is the imputation result, and $\mathcal{X}_i$ is the ground truth.

The performance of Meta-TD was evaluated against fifteen baseline methods, which can be broadly classified into two main groups. The first group consists of statistical learning methods, including Historical average (HA) (Campbell & Thompson 2008), Harmonic mean clustering (KHM) (Anwar *et al*. 2019), and Simple exponential smoothing (SES)(Gardner 2006). The second group comprises machine learning methods, such as K-Nearest Neighbors (KNN) (Pujianto *et al*. 2019), Bidirectional Recurrent Imputation for Time Series (BRITS) (Cao *et al*. 2018), temporal convolutional networks (TCN) (Bai *et al*. 2018), semi-supervised generative adversarial network model (SSGAN) (Miao *et al*. 2021), temporal modeling network (TimeNet) (Wu *et al*. 2023), and three classes of tensor decomposition methods. The first class comprises improved models based on CP decomposition, including tensor factorization with alternating least squares (TF-ALS) (Jain and Oh 2014) and Bayesian Gaussian CANDECOMP/PARAFAC (BGCP) (Chen *et al*. 2019); the second class comprised improved models based on Bayesian decomposition, such as Bayesian temporal tensor factorization (BTTF) (Chen *et al*. 2019), Bayesian augmented tensor factorization (BATF) (Chen *et al*. 2019), and Bayesian temporal matrix factorization (BTMF) (Chen *et al*. 2022); the third class includes models improved upon low-rank tensor imputation with the nuclear norm, such as low-rank tensor completion with truncated nuclear norm (LRTC-TNN) (Chen *et al*. 2022) and low-rank autoregressive matrix completion with truncated nuclear norm (LAMC-TNN) (Chen *et al*. 2020). The tensor factorization methods in the second and third classes are sourced from the Transdim Library (https://transdim.github.io/).

- HA: HA imputes the flow values at the missing locations by calculating the average of the observations from historical data.
- KHM: KHM employs a clustered distribution of observations for the imputation of missing values.
- SES: SES imputes missing values using a weighted average of historical time observations.
- KNN: KNN employs the observations of the nearest K neighbors to address missing data and ensure completeness.
- BRITS: BRITS employs a bidirectional recurrent neural network (RNN) to effectively address the complementation of multiple relevant missing values in time series data.
- TCN: TCN examines the relationship between sequence modeling and recurrent neural networks in the context of time series data analysis.
- SSGAN: SSGAN generates various dynamic weight matrices during the semi-supervised learning phase to impute missing values in multivariate time series data.
- TimeNet: TimeNet transforms time series data from a one-dimensional format to a two-dimensional tensor, facilitating the extraction of multi-scale cyclical change information for imputing missing data.
- TF-ALS: TF-ALS interpolates data using the least squares method based on CP decomposition.

- BTTF: BTTF integrates low-rank matrix/tensor decomposition and vector autoregression processes into a single probabilistic graphical model to impute missing traffic values.
- BTMF: BTMF is a variant of the TRMF model that incorporates the Bayesian theory into the solution of the TRMF model to impute missing traffic values.
- BATF: BATF uses variational Bayes to automatically learn the model parameters and discover interpretable patterns from global parameters, biases, and latent factors to impute missing values.
- BGCP: The BGCP is a special tensor decomposition method that integrates Bayesian theory to impute missing traffic states based on traditional tensor decomposition.
- LRTC-TNN: LRTC-TNN imputes the missing traffic states by decomposing the spatiotemporal tensor of the traffic values.
- LAMC-TNN: LAMC-TNN introduces alternate minimization schemes to impute missing traffic values.

5.3. Baseline comparison (RQ1)

Table 2 presents the comparison results of various methods for the RM and BM patterns. For the RM pattern with an OMR of 20%, Meta-TD outperformed the baseline models by 12.15–95.57% in MAE, 3.33–91.8% in RMSE, and 22.79–88.65% in MAPE. With an OMR of 40%, Meta-TD demonstrated superiority over the baseline models by 6.97–88.96% in MAE, 10.22–90.94% in RMSE, and 13.21–84.29% in MAPE. Notably, the accuracy of statistical learning-based methods is lower than that of machine learning-based methods. This discrepancy arises primarily because statistical learning-based methods depend on the proximity relationship between observations and missing values. In the RM pattern, it is often observed that the values adjacent to the missing data points are also absent, which significantly undermines the efficacy of imputation techniques. In contrast, machine learning-based methods can maintain greater adaptability in handling RM pattern by mining the spatiotemporal patterns from the known data.

Under the BM pattern with an OMR of 20%, Meta-TD exhibited superior performance over the baseline models by 36.04–97.05%in MAE, 18.14–93.76% in RMSE, and 35.15–92.50% in MAPE. Similarly, with an OMR of 40%, Meta-TD outperformed the baseline models by 51.33–94.95% in MAE, 25.59–94.07% in RMSE, and 31.96–90.89% in MAPE. The imputation performance of the low-rank tensor imputation models, LRTC-TNN and LAMC-TNN, is significantly compromised under the BM pattern because it disrupts the low-rank property of the traffic data, thereby affecting the imputation results. Overall, Meta-TD performed well for both RM and BM imputation, with a higher accuracy in the BM pattern than that for the baseline models. This performance can be attributed to Meta-TD's utilization of the kernel tensor and factor matrix decomposition, as well as the incorporation of a spatial weight matrix to improve the imputation results. Furthermore, it is noteworthy that in cases of BM pattern, the imputation performance of deep learning models like SSGAN and TimeNet significantly declines, with this downward trend becoming more pronounced as the missing rate increases.

Table 2. Comparison of Meta-TD and baselines under random missing (RM) and block missing (BM) patterns.

		RM		BM	
		MAE/RMSE/MAPE (%) ↓			
Model	Missing type	OMR: 20%	OMR: 40%	OMR: 20%	OMR: 40%
Statistical learning method	HA	10.765/17.104/47.172	10.878/17.229/47.205	12.239/18.229/61.447	12.457/18.808/58.109
	KHM	10.551/18.035/34.775	10.614/18.121/34.839	10.743/17.154/42.955	11.091/17.967/40.777
	SES	6.652/11.601/28.889	7.515/12.835/33.155	11.302/16.810/67.070	12.799/18.663/78.250
Machine learning method	KNN	2.815/4.952/13.664	3.256/5.898/15.121	4.921/9.205/22.218	6.045/11.381/25.392
	TCN	1.724/1.980/10.391	1.910/2.051/11.906	1.581/1.819/10.180	1.687/2.839/10.743
	BRITS	0.640/1.647/9.492	1.765/2.871/10.676	2.009/3.666/16.324	2.276/4.425/18.651
	SSGAN	2.625/4.989/10.492	2.762/5.019/11.608	2.928/5.332/23.713	3.274/5.665/21.008
	TimeNet	0.543/1.788/6.933	1.294/1.893/9.164	0.713/1.940/7.756	2.072/2.747/10.469
	LAMC-TNN	1.009/1.549/8.439	1.501/2.975/11.879	15.469/23.86/28.25	16.25/34.44/28.96
	LRTC-TNN	1.101/1.530/8.552	1.291/1.829/9.203	3.157/4.357/20.940	3.660/5.024/22.42
	TF-ALS	2.772/4.019/18.724	2.779/4.021/18.432	2.966/4.394/18.117	2.967/4.474/18.66
	BTTF	1.689/2.393/10.059	1.706/2.419/10.083	1.980/2.986/10.826	2.074/3.171/11.32
	BTMF	1.212/1.843/7.790	1.432/1.993/8.544	2.819/4.376/17.320	3.091/4.879/19.321
	BATF	1.403/1.942/8.921	1.416/1.97/9.27	1.934/3.025/10.717	2.146/3.408/12.067
	BGCP	1.367/1.897/8.613	1.407/1.970/8.709	1.966/3.206/10.653	2.036/3.042/11.321
	Meta-TD	**0.477/1.479/5.353**	**1.201/1.642/7.415**	**0.456/1.489/5.030**	**0.821/2.044/7.123**

Note: The optimal values are bolded.

Table 3. Comparison of Meta-TD and baselines under the long time-series missing (TM) pattern.

		TM					
		OMR = 20% & IMR = 40%			OMR = 40% & IMR = 80%		
Model	Missing type	MAE ↓	RMSE ↓	MAPE (%) ↓	MAE ↓	RMSE ↓	MAPE (%) ↓
Statistical learning-based method	HA	11.083	18.773	42.062	12.794	21.070	44.977
	KHM	12.043	20.850	34.947	13.657	22.816	38.445
	SES	8.221	13.732	37.113	8.093	14.564	31.139
Machine learning-based method	KNN	3.264	5.811	13.833	5.158	10.251	18.553
	TCN	1.963	2.456	13.879	3.209	4.474	22.821
	BRITS	2.517	3.191	13.340	2.625	3.694	19.093
	SSGAN	4.325	5.049	24.947	4.853	5.515	29.531
	TimeNet	1.879	2.271	7.116	1.963	3.357	9.586
	LAMC-TNN	15.235	15.140	19.500	32.688	43.587	45.850
	LRTC-TNN	4.939	6.255	19.400	13.713	16.715	23.510
	TF-ALS	3.057	4.454	18.290	5.182	9.477	25.390
	BTTF	2.017	3.392	10.060	2.023	4.029	10.570
	BTMF	1.575	2.413	8.250	1.601	2.880	8.070
	BATF	1.684	2.522	9.270	1.877	2.977	10.660
	BGCP	1.637	2.410	8.800	1.838	3.261	9.370
	Meta-TD	**0.497**	**1.571**	**5.070**	**1.102**	**2.318**	**8.012**

Note: The optimal values are bolded.

To evaluate the performance of Meta-TD in imputing missing long time-series traffic flow data, we conduct a comparative experiment using the TM pattern. The results indicate that Meta-TD achieved the highest imputation accuracy relative to the baseline models (Table 3). When the OMR was 20% and IMR was 40%, Meta-TD outperformed the baseline models by 68.44–96.74% in MAE, 28.75–87.96% in RMSE, and 28.75–87.94% in MAPE. Notably, the imputation performance of TCN and SSGAN decreases significantly as the IMR increases. When the missing rate of traffic flows at a single monitoring point exceeds 80%, the model struggles to effectively learn the underlying traffic patterns. Similarly, when the OMR was 40% and the IMR was 80%, Meta-TD surpassed the baseline models by 31.16–96.62% in MAE, 19.51–94.68% in RMSE, and 0.79–82.53% in MAPE. The imputation results reveal a significant difference in performance between Meta-TD and the baseline model in the TM pattern. This disparity primarily stems from the challenges associated with relying solely on the hidden spatiotemporal features of observed data to impute missing values in long time series. On the one hand, missing data can significantly impede the analysis of spatiotemporal distribution patterns, a problem that worsens with a higher missing rate. On the other hand, when a single monitoring point experiences substantial data loss, accurately determining the correlation between its flow and that of other monitoring points becomes challenging, complicating the imputation process. Therefore, introducing additional geographic features to establish flow relationships between monitoring points is essential.

While improved models of tensor decomposition attempt to improve the imputation performance for the TM pattern by introducing a fixed geographic similarity matrix, these methods often fail to accurately capture the spatial heterogeneity in the traffic flows. Although both SSGAN and TimeNet utilize dynamic weight matrices to reflect the changing relationships in traffic flows, their effectiveness diminishes with increased missing data, limiting their ability to fully learn the underlying traffic patterns. In contrast, Meta-TD effectively leverages additional geographic features to address these challenges through a dynamic spatial weight matrix to capture the

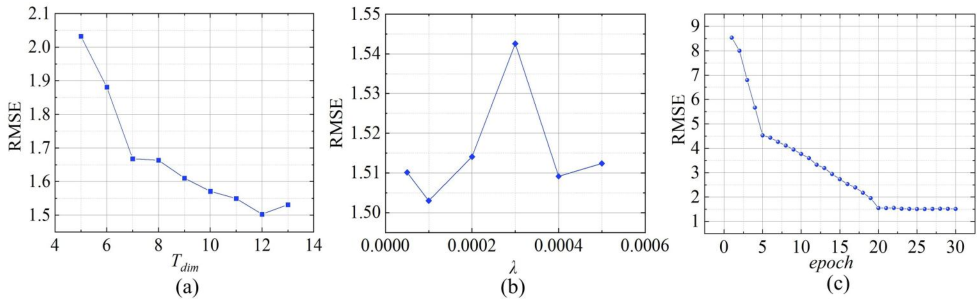

Figure 5. Hyper-parameter sensitivity analysis: (a) T_{dim}, (b) λ, and (c) *epoch*.

spatial heterogeneity relationships between monitoring points. This strategy enables it to achieve the highest imputation accuracy in the TM pattern compared to the baseline models.

5.4. Hyper-parameter sensitivity analysis (RQ2)

We conducted a sensitivity analysis on three hyper-parameters that affected the performance of Meta-TD: the dimension of the core tensor T_{dim}, regularization coefficient λ, and the number of iterations for the tensor solution *epoch*. To assess the impact of each parameter on imputation performance, the controlled variable method was employed by varying one parameter while keeping the others fixed for comparative analysis. The range of T_{dim} was set to [5–13], λ ranged from [0.00005–0.0005], and *epoch* ranged from [1–30]. As shown in Figure 5, T_{dim} determines the degree of interaction between factor matrices; an excessively small T_{dim} cannot effectively capture the key information in the original tensor, resulting in the RMSE gradually decreasing with increasing T_{dim} until reaching the optimal performance when T_{dim} was 12 (Figure 5(a)); λ dictates the extent to which the spatial heterogeneity matrix and Laplacian regularization constrain the tensor decomposition. The highest model accuracy was achieved when λ was set to 0.0001 (Figure 5(b)); *epoch* represents the number of iterations in model training. When *epoch* was set to 20, the RMSE tended to stabilize, indicating that the model gradually converged (Figure 5(c)).

5.5. Ablation study (RQ3)

This section aims to validate the effectiveness of each key component of Meta-TD. The definitions of the different model variants are as follows:

Meta-TD/NM: Node meta-knowledge was used to calculate the spatial weight matrix, neglecting the impact of edge meta-knowledge on the spatial heterogeneity of traffic flows.

Meta-TD/EM: Edge meta-knowledge was used to calculate the spatial weight matrix, neglecting the impact of node meta-knowledge on the spatial heterogeneity of traffic flows.

Meta-TD/NEM: Utilized both node and edge meta-knowledge to calculate the spatial weight matrix while neglecting the impact of traffic flow features on spatial heterogeneity.

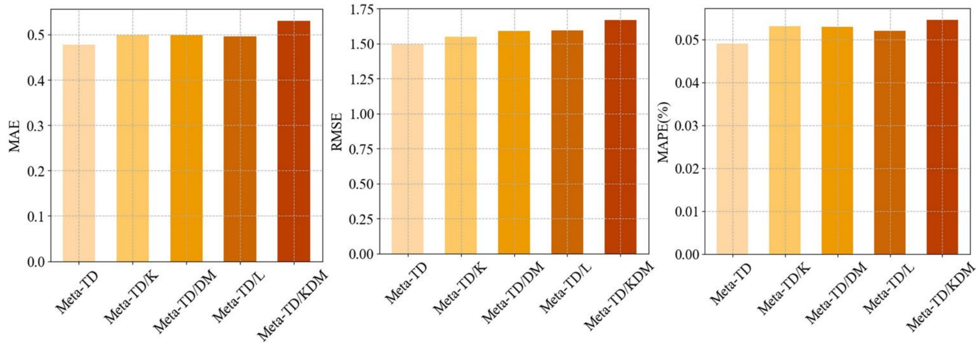

Figure 6. Effect of different components on imputation results.

Meta-TD/ALL: Omitted both meta-knowledge and traffic flow features in the tensor decomposition process, aiming to validate the impact of the spatial heterogeneity learning module on traffic flow imputation.

As shown in Figure 6, the absence of any component adversely affects performance, with all variants exhibiting increases in RMSE, MSE, and MAPE of 3.77–9.87%, 3.67–10.42%, and 5.90–10.05%, respectively. These results indicate that each component proposed in this study played a positive role in the overall performance of Meta-TD. Specifically, the performance gap between Meta-TD/ALL and Meta-TD was the most pronounced, with differences of 9.87% in RMSE, 10.42% in MSE, and 10.05% in MAPE, highlighting the importance of modeling spatial heterogeneity relationships between nodes for the traffic flow imputation process. Moreover, the effect of changes in traffic flow features on spatial heterogeneity should not be neglected, as evidenced by the 6.21% increase in RMSE for Meta-TD/NEM relative to Meta-TD.

5.6. Effectiveness of geographic meta-knowledge (RQ4)

Influenced by node and edge meta-knowledge, the closer the distance and the more similar the geographic features between two monitoring points, the larger the values in their spatial weight matrix, resulting in more similar traffic flow variation trends, and vice versa. To verify this phenomenon, we randomly selected nine traffic monitoring points numbered N1–N9 and used pie charts to display the top five POI types for N1, N4, and N6 (Figure 7(a)). Additionally, we selected two time intervals, T1 and T2, spanning March 1 to March 7, 2021, and the calculation results of the spatial weight matrix and the traffic flow imputation results were visualized.

Statistical analysis showed that the POIs around N4 and N6 primarily comprised business residential areas, accounting for 30.6% and 31.6%, respectively. N1 was surrounded mainly by company enterprises, accounting for 32.6% of its POIs (Figure 7(b)). It can be observed that N4 and N6 had similar functional zones and were closer to each other compared to the distance between N1 and N6. Therefore, the values in the spatial weight matrix were consistently larger between N4 and N6 than between N1 and N6 across different time intervals, leading to similar traffic flow variation trends between N4 and N6 compared to that between N1 and N6(Figures 7(c–e)). This phenomenon demonstrates that spatial weight matrices can be effectively learned

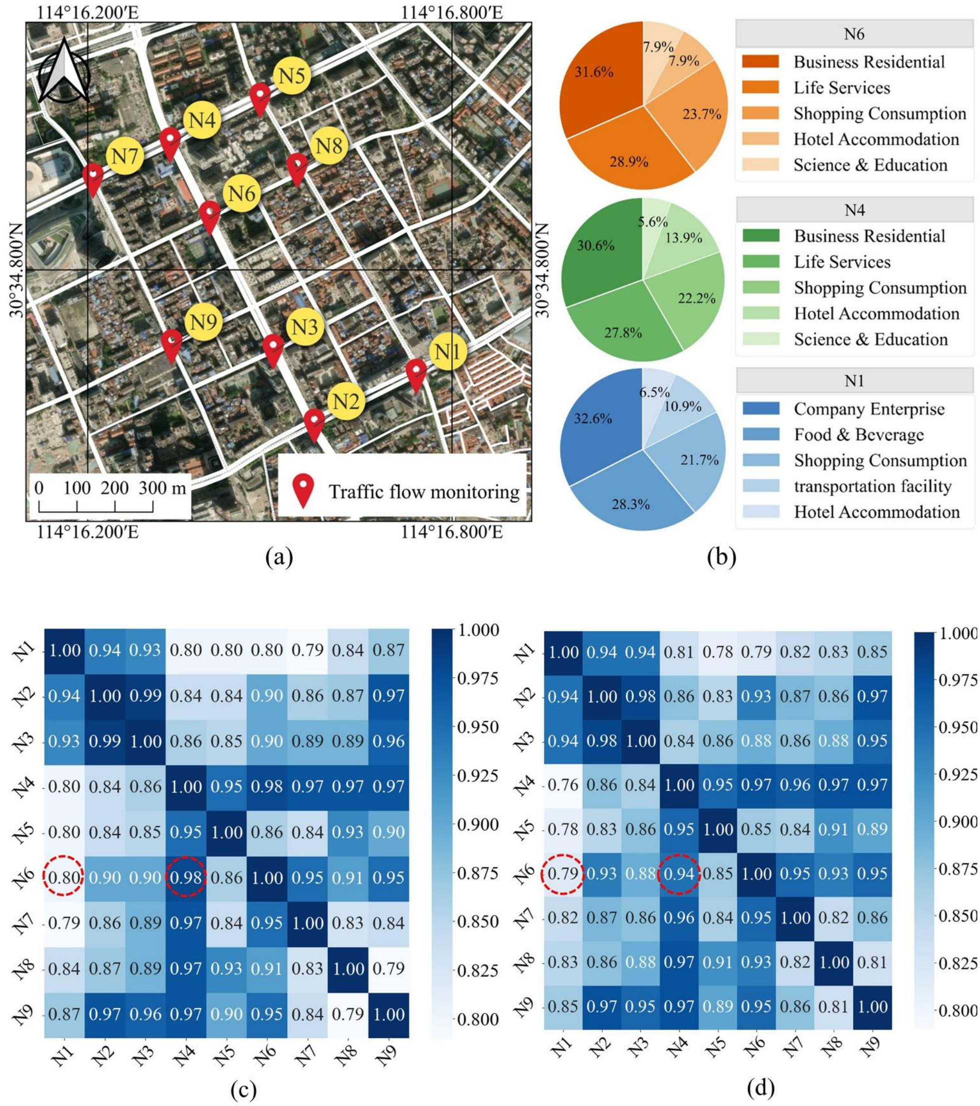

Figure 7. Analysis of the effectiveness of the spatial weight matrix. (a) Distribution of monitoring point locations; (b) distribution of POI types around monitoring points; (c) spatial weight matrix at time interval T1; (d) spatial weight matrix at time interval T2; (e) traffic flow trends at time interval T1 and T2.

through geographic meta-knowledge, thereby reflecting the spatial heterogeneity of the traffic flows between monitoring points.

5.7. Robustness of the proposed method (RQ5)

To verify the robustness of the proposed method, this analysis employed the Meta-TD and BGCP methods, which showed better performance, as described in Section 5.3, in imputing missing data for traffic flows under both normal and mutation conditions. As shown in Figure 8(a), under normal traffic conditions, the imputation results of the

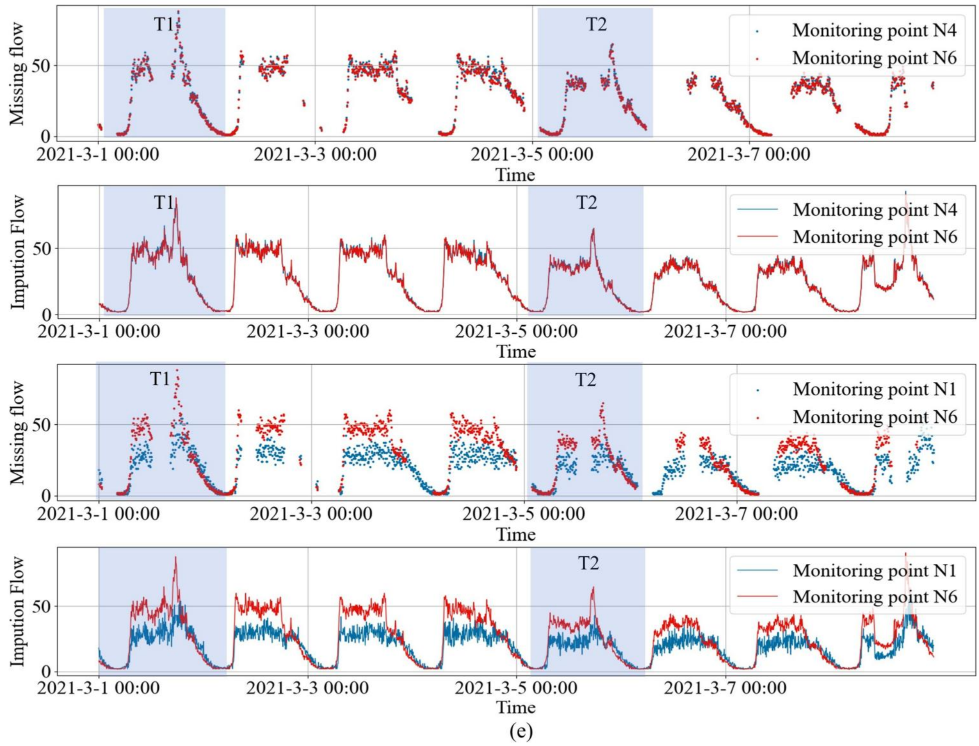

Figure 7. Continued.

Meta-TD method were closer to the true values than those of the BGCP method, although the difference was small (Figure 8(c)). However, under mutation traffic conditions, the Meta-TD method maintained a high imputation performance, whereas the BGCP method showed an obvious deviation from the true values (Figure 8(b)). This is because the BGCP method adopts a fixed geographic feature similarity matrix to constrain the tensor decomposition process, which ignores the dynamic impact of traffic states on the spatial correlation of traffic flows. In contrast, the Meta-TD method considers geographic meta-knowledge and traffic flow features by embedding a dynamic spatial weight matrix in the tensor-solving process. This allows it to capture changes in traffic flows effectively, thereby maintaining high accuracy in the imputation of missing traffic flow data.

6. Discussion

6.1. Innovation of Meta-TD

Tensor decomposition is a mainstream method for traffic flow imputation, and its performance enhancement critically depends on the reasonable embedding of contextual information during the decomposition process (Zheng *et al.* 2014, Zhang *et al.* 2023). However, existing methods generally construct a fixed geographic feature similarity matrix as contextual information to constrain the tensor decomposition process. This

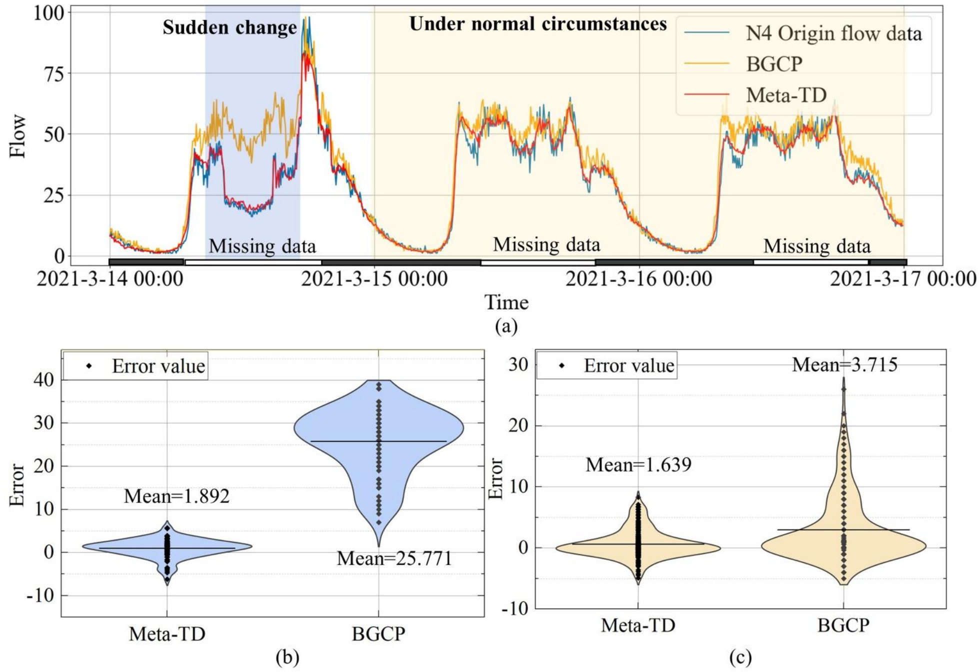

Figure 8. Imputation performance of different methods in normal and mutation cases. (a) The imputation result curves of different methods; (b) comparison of mean error box plots under normal case; (c) comparison of mean error box plots under mutation case.

approach often fails to capture the spatial heterogeneity of traffic flows accurately, thereby limiting the accuracy and robustness of traffic flow imputation (Said & Erradi 2022, Huang *et al*. 2022, Zhao *et al*. 2023). In fact, the spatial heterogeneity of traffic flows is determined by a complex interplay between the static geographic attributes and the dynamic traffic flow states. To effectively capture this heterogeneity, this study introduces "geographic meta-knowledge", which creates learnable feature embeddings through high-level semantic mapping. This approach provides a representation framework for the interactions between geographic features and traffic flow conditions, thereby enabling a more precise and adaptive characterization of spatial heterogeneity in traffic flows. Building on this concept, a tensor decomposition method embedded with geographic meta-knowledge was proposed. This method calculates spatial weight matrices by integrating geographic meta-knowledge with traffic flow features to constrain the tensor-solving process, thereby enabling the model to perceive and understand the spatial heterogeneity of traffic flows more accurately.

6.2. Scalability of Meta-TD

We performed comparative experiments using fifteen baseline models on real urban traffic flow data. The results demonstrate that the Meta-TD method maintains high robustness under varying missing rates and distribution patterns. To evaluate the scalability of the proposed method on large-scale datasets, we analyzed its time and space complexity. Similar to the widely used CP and Tucker decomposition methods

(Battaglino *et al*. 2018; Li *et al*. 2014), the time and space complexity of the Meta-TD method is primarily determined by the dimensionality of the decomposed core tensor, establishing a linear relationship with data volume. Moreover, the method facilitates batch training by partitioning the data along the time dimension, thereby reducing the data load per iteration and enhancing its applicability to large-scale datasets.

6.3. Hyper-parameter selection

The selection of hyper-parameters is crucial when applying an algorithm to different datasets. For the Meta-TD method, three key hyper-parameters need to be considered: *epoch*, T_{dim} and λ. During the model training process, we introduce a tolerance parameter for the number of iterations (*epochs*) in tensor solving, allowing for automatic termination when the improvement in the loss function falls below a specified threshold. This mechanism enables the model to adaptively adjust the number of iterations based on the characteristics of various datasets. For the dimensionality (T_{dim}) and regularization coefficient (λ) of the core tensor, we use the classical grid search method for hyper-parameter selection (Syarif *et al*. 2016). By performing a comprehensive search within a predefined range of parameters, the method can automatically select the optimal configurations across different datasets. Notably, the Meta-TD method exhibits robustness to the regularization factor λ, indicating that variations in λ within a certain range do not significantly affect model performance. This characteristic greatly reduces the search space, ensuring that Meta-TD maintains stable performance across diverse datasets.

6.4. Potential applications

The Meta-TD method exhibited high robustness across various missing rates and distribution patterns, making it particularly applicable to complex and dynamic urban traffic scenarios. This reliability enables traffic regulatory authorities to utilize Meta-TD for intelligent transportation applications, such as dynamic route planning, traffic resource allocation, and traffic system management. By integrating geographic meta-knowledge with tensor decomposition, Meta-TD provides a novel approach for exploring multidimensional spatiotemporal correlations and uncovering complex patterns in the data. Furthermore, the Meta-TD method can be applied to other spatiotemporal datasets, including meteorological and air quality datasets that may also contain missing values. These datasets can be structured as three-dimensional tensors (stations, days, and time intervals), facilitating effective imputation. Moreover, if the values to be predicted in spatiotemporal datasets are treated as missing values for imputation purposes, the Meta-TD method can also be utilized for spatiotemporal prediction. This application holds significant potential across diverse domains, such as traffic management and planning, environmental monitoring, and social network analysis.

6.5. Limitations and future work

Although this study validates the superiority of Meta-TD over multiple baseline models on a real dataset, certain limitations remain. For example, the algorithm relies on geographic meta-knowledge to compute spatial weight matrices, which led to comparative experiments being conducted on a limited range of traffic datasets with distinct geographic attributes. The reliance on geographic meta-knowledge raises a critical question: could spatiotemporal heterogeneity be extracted directly from the data itself, rather than depending on geographic meta-knowledge? While this alternative approach could enhance flexibility and adaptability, it is challenged by the issue of missing data. Specifically, the presence of missing values can significantly compromise the accuracy of spatiotemporal heterogeneity extraction (Dong *et al*. 2024). Thus, extracting robust spatiotemporal heterogeneity under conditions of missing data remains a complex task. To address this challenge, future work may incorporate contrastive and self-supervised learning methods to directly capture latent spatiotemporal heterogeneity from the data itself. These approaches could strengthen model robustness by structurally representing unlabeled data, thereby generating dynamic spatial weight matrices without relying on external meta-knowledge.

7. Conclusions

Accurate and reliable urban traffic flow data can provide essential information for various applications including urban planning, route selection, and traffic control. However, fixed detection devices such as cameras and radars/laser detectors are not effective, resulting in missing traffic flow data. This poses a challenge for smart city applications. Therefore, this study proposes a geographic meta-knowledge embedded tensor decomposition (Meta-TD) method for urban traffic flow imputation, which enhances the accuracy and robustness of imputing missing urban traffic flow data.

The method fully considers the geographic features of nodes and edges, as well as dynamic traffic states, by designing a geographic meta-knowledge extraction module based on meta-learning. This module extracts node meta-knowledge and edge meta-knowledge from the node and edge attributes. Additionally, a spatial weight matrix calculation module that considers spatial heterogeneity was designed to model the dynamic relationship between geographic meta-knowledge and spatial heterogeneity. Finally, a tensor decomposition method for spatial weight matrix embedding was proposed, with continuous optimization of the spatial weight matrix computation and flow imputation results through collaborative learning to obtain the final imputation results.

Experiments conducted on real urban traffic flow data demonstrate that the proposed method outperforms seven baseline models in the RM, BM, and TM patterns, with the MAE, RMSE, and MAPE reduced by 6.97–97.05%, 3.33–94.68%, and 1.97–82.53%, respectively. These metrics consistently remained within the ranges of 0.48–1.20, 1.48–2.32, and 5.03–8.01%. Furthermore, the spatial weight matrix learned by the Meta-TD method reflected the spatial heterogeneity of the traffic flows between monitoring points. In the event of a sudden change in traffic flow data within a day, the spatial weight matrix adapts dynamically to ensure that the imputation results of the Meta-TD method remain accurate and robust across different missing patterns. This approach effectively addresses the imputation requirements for missing traffic flow data patterns and provides timely and dependable data support for intelligent decision-making and traffic control applications.

Acknowledgments

The authors would also like to thank the anonymous referees for their helpful comments and suggestions.

Disclosure statement

The authors report there are no competing interests to declare.

Funding

This work was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences [grant numbers XDB0740100-02], the National Natural Science Foundation of China [grant number 42371469, 42101423] and Innovation Project of LREIS [grant number KPI003 and YPI002].

Notes on contributors

Xiaoyue Luo is a Ph.D. candidate in the Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences. Her research interests include spatiotemporal data imputation, urban computing, and intelligent transportation. She contributed to the conceptualization, methodology, formal analysis, investigation, visualization, writing – original draft and writing – review & editing.

Shifen Cheng is an Associate Professor with the Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences. His research interests include spatiotemporal data mining, urban computing, and intelligent transportation. He contributed to the conceptualization, methodology, formal analysis, investigation, visualization, and writing – review & editing.

Lizeng Wang is a Ph.D. candidate in the Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences. His research interests include spatiotemporal data mining, spatiotemporal prediction, and urban computing. He contributed to the writing – review & editing.

Yuxuan Liang is an Assistant Professor at the Intelligent Transportation Thrust, Hong Kong University of Science and Technology (Guangzhou). His research interests include spatiotemporal data mining and Artificial Intelligence. He contributed to the writing – review & editing.

Feng Lu is a Professor at the Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences. His research interests cover trajectory data mining, computational transportation science, and location-based services. He contributed to writing – review & editing, supervision, project administration, and funding acquisition.

Data and codes availability statement

The data and codes that support the findings of this study are available in “figshare.com” with the identifier at the private link https://doi.org/10.6084/m9.figshare.25997134.

References

Anwar, T., *et al.*, 2019. A study on missing values imputation using K-Harmonic means algorithm: mixed datasets. *AIP conference proceedings*. AIP Publishing.

Bai, S., Kolter, J.Z., and Koltun, V., 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv Preprint arXiv, 1803.01271.

Balažević, I., Allen, C., and Hospedales, T.M., 2019. Tucker: tensor factorization for knowledge graph imputation. arXiv Preprint arXiv, 1901.09590.

Bansal, P., Deshpande, P., and Sarawagi, S., 2021. Missing value imputation on multidimensional time series. arXiv Preprint arXiv, 2103.01600.

Battaglino, C., Ballard, G., and Kolda, T.G., 2018. A practical randomized CP tensor decomposition. *SIAM Journal on Matrix Analysis and Applications*, 39 (2), 876–901.

Benahmed, L., and Houichi, L., 2018. The effect of simple imputations based on four variants of PCA methods on the quantiles of annual rainfall data. *Environmental Monitoring and Assessment*, 190 (10), 569.

Bhanu, M., Mendes-Moreira, J., and Chandra, J., 2021. Embedding traffic network characteristics using tensor for improved traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 22 (6), 3359–3371.

Campbell, J.Y., and Thompson, S.B., 2008. Predicting excess stock returns out of sample: can anything beat the historical average?. *Review of Financial Studies*, 21 (4), 1509–1531.

Cao, W., *et al.*, 2018. Brits: bidirectional recurrent imputation for time series. *Advances in Neural Information Processing Systems*, 31 (7), 6775–6785.

Chen, X., *et al.*, 2019. Missing traffic data imputation and pattern discovery with a Bayesian augmented tensor factorization model. *Transportation Research Part C: Emerging Technologies*, 104, 66–77.

Chen, X., *et al.*, 2022. Low-rank autoregressive tensor imputation for spatiotemporal traffic data imputation. *IEEE Transactions on Intelligent Transportation Systems*, 23 (8), 12301–12310.

Chen, X., He, Z., and Sun, L., 2019. A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*, 98, 73–84.

Chen, X., Yang, J., and Sun, L., 2020. A nonconvex low-rank tensor imputation model for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*, 117, 102673.

Cheng, S., *et al.*, 2018. Short-term traffic forecasting: An adaptive ST-KNN model that considers spatial heterogeneity. *Computers, Environment and Urban Systems*, 71, 186–198.

Dong, Z., *et al.*, 2024. Heterogeneity-informed meta-parameter learning for spatiotemporal time series forecasting. *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, 631–641.

Ganji, A., Zhang, M., and Hatzopoulou, M., 2022. Traffic volume prediction using aerial imagery and sparse data from road counts. *Transportation Research Part C: Emerging Technologies*, 141, 103739.

Gardner, E.S., 2006. Exponential smoothing: the state of the art—Part II. *International Journal of Forecasting*, 22 (4), 637–666.

Golub, G.H., and Van Loan, C.F., 2013. *Matrix computations.* Baltimore: Johns Hopkins University Press.

Goulart, J.H.M., Kibangou, A.Y., and Favier, G., 2017. Traffic data imputation via tensor completion based on soft thresholding of Tucker core. *Transportation Research Part C: Emerging Technologies*, 85, 348–362.

Huang, L., *et al.*, 2022. Context-aware road travel time estimation by coupled tensor decomposition based on trajectory data. *Knowledge-Based Systems*, 245, 108596.

Jain, P., and Oh, S., 2014. Provable tensor factorization with missing data. *Advances in Neural Information Processing Systems*, 27 (2), 1431–1439.

Jia, X., *et al.*, 2021. Missing data imputation for traffic congestion data based on joint matrix factorization. *Knowledge-Based Systems*, 225, 107114.

Jiang, R., *et al.*, 2023. Spatio-temporal meta-graph learning for traffic forecasting. *Proceedings of the AAAI conference on artificial intelligence*, 8078–8086.

Joelianto, E., *et al.*, 2022. Analysis of spatiotemporal data imputation methods for traffic flow data in urban networks. *ISPRS International Journal of Geo-Information*, 11 (5), 310.

Ke, J., *et al.*, 2019. PCA-based missing information imputation for real-time crash likelihood prediction under imbalanced data. *Transportmetrica A: Transport Science*, 15 (2), 872–895.

Khaled, A., Elsir, A.M.T., and Shen, Y., 2022. TFGAN: traffic forecasting using generative adversarial network with multi-graph convolutional network. *Knowledge-Based Systems*, 249, 108990.

Kolda, T.G., and Bader, B.W., 2009. Tensor decompositions and applications. *SIAM Review*, 51 (3), 455–500.

Li, J., Li, R., and Xu, L., 2023. Multi-stage deep residual collaboration learning framework for complex spatial-temporal traffic data imputation. *Applied Soft Computing*, 147, 110814.

Li, Y., *et al.*, 2013. Comparison on PPCA, KPPCA and MPPCA based missing data imputing for traffic flows. *Improving multimodal transportation systems-information, safety, and integration*, 1151–1156.

Li, Y., Li, Z., and Li, L., 2014. Missing traffic data: comparison of imputation methods. *IET Intelligent Transport Systems*, 8 (1), 51–57.

Liang, Y., *et al.*, 2019. Urbanfm: inferring fine-grained urban flows. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 3132–3142.

Malik, O.A., and Becker, S., 2018. Low-rank tucker decomposition of large tensors using tensor sketch. *Advances in Neural Information Processing Systems*, 31 (14), 10096–10106.

Meng, C., *et al.*, 2017. City-wide traffic volume inference with loop detector data and taxi trajectories. *Proceedings of the 25th ACM SIGSPATIAL international conference on advances in geographic information systems*, 1–10.

Miao, X., *et al.*, 2021. Generative semi-supervised learning for multivariate time series imputation. *Proceedings of the AAAI conference on artificial intelligence*, 8983–8991.

Narita, A., *et al.*, 2012. Tensor factorization using auxiliary information. *Data Mining and Knowledge Discovery*, 25 (2), 298–324.

Nguyen, T., *et al.*, 2023. Principal components analysis based frameworks for efficient missing data imputation algorithms. *Asian conference on intelligent information and database systems*. Cham: Springer Nature Switzerland, 254–266.

Nie, T., Qin, G., and Sun, J., 2022. Truncated tensor schatten p-norm based approach for spatiotemporal traffic data imputation with complicated missing patterns. *Transportation Research Part C: Emerging Technologies*, 141, 103737.

Ongie, G., *et al.*, 2021. Tensor methods for nonlinear matrix completion. *SIAM Journal on Mathematics of Data Science*, 3 (1), 253–279.

Pan, Z., *et al.*, 2019. Urban traffic prediction from spatio-temporal data using deep meta learning. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 1720–1730.

Pujianto, U., Wibawa, A.P., and Akbar, M.I., 2019. K-nearest neighbor (k-NN) based missing data imputation. *2019 5th international conference on science in information technology (ICSITech). IEEE*, 83–88.

Qu, L., *et al.*, 2009. PPCA-based missing data imputation for traffic flows volume: A systematically approach. *IEEE Transactions on Intelligent Transportation Systems*, 10 (3), 512–522.

Said, A.B., and Erradi, A., 2022. Spatiotemporal tensor completion for improved urban traffic imputation. *IEEE Transactions on Intelligent Transportation Systems*, 23 (7), 6836–6849.

Shao, W., and Chen, L., 2018. License plate recognition data-based traffic volume estimation using collaborative tensor decomposition. *IEEE Transactions on Intelligent Transportation Systems*, 19 (11), 3439–3448.

Sivakani, R., and Ansari, G.A., 2020. Imputation using machine learning techniques. *2020 4th International conference on computer, communication and signal processing (ICCCSP)*. IEEE, 1–6.

Smith, K., and Climer, S., 2024. Improving data cleaning using discrete optimization. arXiv Preprint arXiv, 2405.00764.

Suleiman, B., *et al.*, 2022. Multi-contextual recommender using 3D latent factor models and online tensor decomposition. *International conference on computational science. Cham: Springer International Publishing*, 276–290.

Sure, P., Srinivasan, C.P., and Babu, C.N., 2022. Spatio-temporal constraint-based low rank matrix completion approaches for road traffic networks. *IEEE Transactions on Intelligent Transportation Systems*, 23 (8), 13452–13462.

Swersky, K., Snoek, J., and Adams, R.P., 2013. Multi-task bayesian optimization. *Advances in Neural Information Processing Systems*, 26 (3), 2004–2012.

Syarif, I., Prugel-Bennett, A., and Wills, G., 2016. SVM parameter optimization using grid search and genetic algorithm to improve classification performance. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 14 (4), 1502–1509.

Tang, K., *et al.*, 2020. A tensor decomposition method for cycle-based traffic volume estimation using sampled vehicle trajectories. *Transportation Research Part C: Emerging Technologies*, 118, 102739.

Thomas, T., and Rajabi, E., 2021. A systematic review of machine learning-based missing value imputation techniques. *Data Technologies and Applications*, 55 (4), 558–585.

Wang, J., *et al.*, 2020. Understanding urban dynamics via context-aware tensor factorization with neighboring regularization. *IEEE Transactions on Knowledge and Data Engineering*, 32 (11), 2269–2283.

Wang, P., *et al.*, 2022. A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation. *International Journal of Geographical Information Science*, 36 (6), 1231–1257.

Wang, P., *et al.*, 2023. Urban traffic flow prediction: A dynamic temporal graph network considering missing values. *International Journal of Geographical Information Science*, 37 (4), 885–912.

Wu, H., *et al.*, 2023. Timesnet: Temporal 2d-variation modeling for general time series analysis. arXiv Preprint, arXiv2022. 221002186.

Wu, X., *et al.*, 2022. A multi-attention tensor completion network for spatiotemporal traffic data imputation. *IEEE Internet of Things Journal*, 9 (20), 20203–20213.

Wu, Y., *et al.*, 2017. Robust tensor decomposition based on Cauchy distribution and its applications. *Neurocomputing*, 223, 107–117.

Xie, H., Gong, Y., and Dong, X., 2024. Spatial-temporal regularized tensor decomposition method for traffic speed data imputation. *International Journal of Data Science and Analytics*, 17 (2), 203–223.

Xu, M., *et al.*, 2023. AGNP: Network-wide short-term probabilistic traffic speed prediction and imputation. *Communications in Transportation Research*, 3, 100099.

Yang, J.M., Peng, Z.R., and Lin, L., 2021. Real-time spatiotemporal prediction and imputation of traffic status based on LSTM and graph Laplacian regularized matrix factorization. *Transportation Research Part C: Emerging Technologies*, 129, 103228.

You, L., *et al.*, 2024. FMGCN: Federated meta learning-augmented graph convolutional network for EV charging demand forecasting. *IEEE Internet of Things Journal*, 11 (14), 24452–24466.

Zhang, K., *et al.*, 2024. Semantic understanding and prompt engineering for large-scale traffic data imputation. *Information Fusion*, 102, 102038.

Zhang, Y., *et al.*, 2023. Incorporating multimodal context information into traffic speed forecasting through graph deep learning. *International Journal of Geographical Information Science*, 37 (9), 1909–1935.

Zhao, Z., *et al.*, 2023. Toward urban traffic scenarios and more: a spatio-temporal analysis empowered low-rank tensor imputation method for data imputation. *International Journal of Geographical Information Science*, 37 (9), 1936–1969.

Zheng, Y., *et al.*, 2014. Diagnosing New York city's noises with ubiquitous data. *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*, 715–725.

Zhu, Y., *et al.*, 2022. A Bayesian robust CP decomposition approach for missing traffic data imputation. *Multimedia Tools and Applications*, 81 (23), 33171–33184.

Zong, X., *et al.*, 2024. An intelligent deep learning framework for traffic flow imputation and short-term prediction based on dynamic features. *Knowledge-Based Systems*, 300, 112178.