CrossFM: Cross-City Fine-Grained Urban Flow Inference with Incomplete Data

Wenchao Wu[1,2] and Yuanbo Xu[1,2(✉)]

[1] MIC Lab, College of Computer Science and Technology, Jilin University, Changchun, China
[2] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, China
wuwc23@mails.jlu.edu.cn, yuanbox@jlu.edu.cn

Abstract. Fine-grained urban flow inference provides important insights for smart city applications such as urban planning and traffic management, but its accuracy is often hindered by incomplete observations due to sparse sensor deployment. While existing methods can handle minor data gaps, their performance degrades significantly under high missing rates, particularly in newly developed urban areas. Along these lines, we propose a novel cross-city super-resolution data map inference framework (CrossFM), designed to transform incomplete coarse-grained urban flows into accurate fine-grained data maps by harnessing cross-city spatio-temporal dynamics. Specifically, we first perform temporal alignment between the source city and the target city data using timestamps. Then, guided by Point-of-Interest (POI) similarity to identify similar regions, we impute missing values in the target city's coarse-grained flow maps. This completion adaptively leverages information from both the source and target city data, resulting in an enhanced coarse-grained representation. Finally, a super-resolution module processes the spatial patterns within the completed coarse data to generate the high-resolution urban flow maps. The framework components are trained jointly end-to-end within a multi-task setup. We conduct extensive experiments on two real-world datasets and demonstrate that CrossFM significantly outperforms the state-of-the-art methods, especially under severe data scarcity.

Keywords: Transfer Learning · Super-Resolution · Fine-Grained Inference · Spatio-Temporal Data.

1 Introduction

Fine-grained urban flows, like taxi and bike flows, depict human mobility patterns crucial for smart city applications such as urban planning, traffic management [7, 22]. Acquiring such data requires dense sensor networks, incurring substantial operational and maintenance costs. However, sensor deployment is often sparse and uneven due to budget and logistical constraints, resulting in coarse-grained and incomplete observations[19]. This necessitates methods for

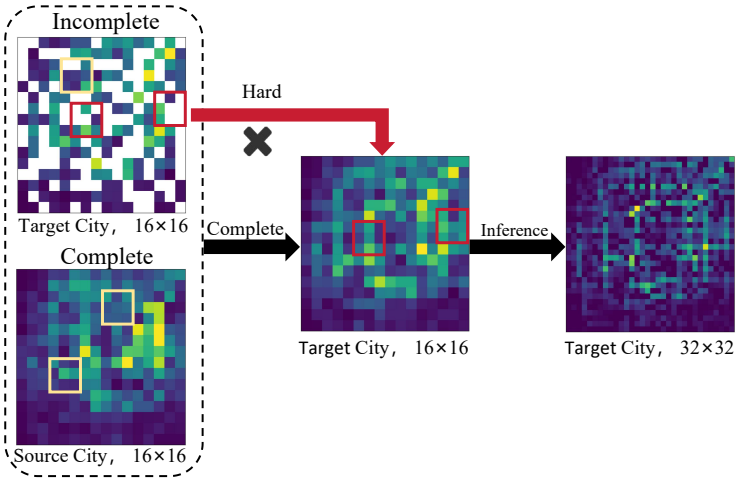


Fig. 1: Fine-grained urban flow inference with incomplete data. The white cell regions denote that the urban flows are unavailable. With high data scarcity, it is challenging to complete the flow map using only the target city's data.

inferring high-resolution urban flows from limited sensor data, a research problem that has garnered significant attention[20].

Traditional urban flow inference relied on statistical methods like interpolation and tensor factorization [9]. However, these approaches often struggle with urban data's scale and complexity and typically ignore external factors (e.g., weather, holidays), limiting accuracy. More recently, inspired by computer vision super-resolution (SR) [21], researchers framed urban flow inference as a spatio-temporal SR task [20], treating flow snapshots as images. Building on early image SR methods (e.g., SRCNN [3], VDSR [10]), UrbanFM first adapted SR to urban flows, incorporating external factors [12]. UrbanPy introduced a pyramidal approach for higher upscaling rates [16]. Addressing the common issue of incomplete coarse data, MT-CSR proposed a multi-task framework for joint data completion and super-resolution [11].

However, a critical limitation remains: existing methods addressing data incompleteness heavily rely on intra-city context. Consequently, their performance degrades significantly with high missing rates, common in newly developed areas or regions with exceptionally sparse sensor coverage. Accurately inferring fine-grained flows under such severe data scarcity remains a significant challenge [4, 5].

The challenge arises primarily from: (1) Failure of Intra-City Completion under High Scarcity: When the available data within the target city is extremely limited, completion methods relying solely on local neighborhood correlations or intra-city semantic similarities (like POI) become unreliable and insufficient for accurate completion. (2) Effectively Leveraging External Data Sources: While using external data holds promise, naively incorporating it is problematic. Uti-

lizing data from a different city, for instance, requires robust mechanisms to ensure relevance. This involves precise temporal alignment to compare corresponding time periods and effective spatial correspondence methods to identify functionally similar, though geographically distinct, regions across different urban environments.

To tackle these challenges, we propose CrossFM, a novel Cross-City Fine-Grained Urban Flow Inference framework. CrossFM introduces a strategy to leverage data from an auxiliary source city to aid inference in the target. Specifically, CrossFM operates sequentially by first performing temporal alignment between the auxiliary source city and the target city using timestamps to ensure that comparisons and data borrowing respect temporal dynamics. Guided by Point-of-Interest (POI) similarity, which helps identify functionally similar urban regions across the two cities, our Cross-city Completion module (CrossCMP) completes missing values in the target city's coarse-grained flow map. CrossCMP adaptively leverages information from both the temporally-aligned source and the available target city data, creating an enhanced, more complete coarse-grained representation. Finally, this enhanced representation is fed into a super-resolution module that processes the spatial patterns to generate the final high-resolution, fine-grained urban flow maps for the target city. The entire CrossFM framework is designed to be trained end-to-end.

The contributions of this paper are summarized as follows:

- To the best of our knowledge, we are the first to address the problem of fine-grained urban flow inference through cross-city transfer learning under conditions of high data scarcity.
- We propose CrossFM, a novel cross-city transfer learning framework that leverages a data-rich source city to enhance fine-grained urban flow inference for a data-scarce target city.
- We design the CrossCMP module to perform completion by considering cross-city spatio-temporal dependencies and global POI similarity between the source and target city.
- We conduct extensive experiments on two real-world datasets, demonstrating that CrossFM significantly outperforms state-of-the-art methods.

2 Related Work

2.1 Fine-grained Urban Flow Inference

Inferring fine-grained urban flows often employs super-resolution (SR) techniques adapted from computer vision, with recent advancements leveraging diffusion models and other deep learning approaches [23]. Urban-specific SR models, such as UrbanFM and UrbanPy [12, 16], were developed to incorporate domain knowledge like external factors or handle high upscaling rates. Recognizing data incompleteness, MT-CSR proposed joint intra-city data completion and SR [11]. However, these methods primarily rely on information within the target city and struggle significantly when coarse data suffers from high missing rates. CrossFM

is specifically designed to address such high scarcity scenarios where intra-city information is insufficient.

2.2 Spatio-Temporal Data Completion

Spatio-temporal data completion aims to complete missing values in sparse datasets like urban flows. Early approaches included statistical algorithms, exemplar-based inpainting seeking similar patches [2], efficient patch-matching techniques [1], and offset fusion methods [6], though these often struggled with complex spatio-temporal correlations. More recently, deep learning (DL) has gained prominence, frequently treating gridded spatio-temporal data as images to leverage image completion advances [8]. Notable DL examples include Context Encoders using encoder-decoder structures [17], Partial Convolutions designed to handle missing data explicitly [13], and edge-focused models like EdgeConnect aimed at reducing blurriness [15]. Despite these methodological advances, a key limitation persists across many approaches, from traditional to deep learning. They fundamentally rely on sufficient surrounding context within the same domain [14]. This dependence causes performance to degrade significantly under high global scarcity or when extensive regions are missing – precisely the challenging conditions CrossFM targets. Consequently, CrossFM introduces a novel completion methodology specifically designed to overcome this reliance on intra-city data by leveraging external information from an auxiliary cross-city source, enabling more robust completion even under severe data scarcity.

3 Nations and Problem Definition

We will first give some definitions to help state the studied problem, and then present a formal problem definition.

Definition 1. (Region). We divide a city into a grid map based on latitude and longitude, consisting of $I \times J$ cell regions. The set of all regions is denoted as $\mathcal{R} = \{r_{i,j} | 1 \leq i \leq I, 1 \leq j \leq J\}$, where $r_{i,j}$ represents the cell region at the i-th row and j-th column.

Definition 2. (Urban Flow Map). For a given time interval t, the urban flow map captures the movement intensity. For each region $r_{i,j}$, we define inflow $X^t_{in,i,j}$ and outflow $X^t_{out,i,j}$ based on trajectories $\mathcal{T}$:

$$X^t_{in,i,j} = \sum_{f \in \mathcal{T}} \{(f_{t-1} \notin r_{i,j} \wedge f_t \in r_{i,j})\} \tag{1}$$

$$X^t_{out,i,j} = \sum_{f \in \mathcal{T}} \{f_t \in r_{i,j} \wedge f_{t+1} \notin r_{i,j}\} \tag{2}$$

where f_t is the location of urban flow trajectory f at time t. We represent the inflow and outflow for all regions at time t as an urban flow map tensor $X_t \in \mathbb{R}^{2 \times I \times J}$.

Definition 3. (Coarse- and fine-grained Flow Maps). A coarse-grained urban flow map represents the observed urban flows derived from the flow sensors. It is generated by integrating neighboring grids within an $N \times N$ range from a fine-grained urban flow map, where N is the upscaling factor. We denote the coarse-grained and fine-grained urban flow maps at time t as $X_{cg}^t \in \mathbb{R}^{2\times I\times J}$ and $X_{fg}^t \in \mathbb{R}^{2\times NI\times NJ}$, respectively. In practice, the observed coarse-grained map $\hat{X}_{cg}^t \in \mathbb{R}^{2\times I\times J}$ is often incomplete.

Definition 4. (Point-of-Interest Features). To capture functional characteristics essential for our cross-city approach, we use Point-of-Interest (POI) features. These are represented as a tensor $P \in \mathbb{R}^{K\times I\times J}$ for a given city, where K is the number of POI categories (e.g., commercial, residential) aggregated within each coarse-grained region $r_{i,j}$. Both the target city D and source city D' have associated POI features, denoted as P^D and $P^{D'}$, respectively.

Problem Statement. Given the upscaling factor N, the sequence of observed incomplete coarse-grained flow maps $\{\hat{X}_{cg,D}^t\}_{t\in T_D}$ from the data-scarce target city D, the sequence of relatively complete coarse-grained flow maps $\{X_{cg,D'}^{t'}\}_{t'\in T_{D'}}$ from the data-rich source city D', and the POI features P^D and $P^{D'}$ for both cities, our goal is to infer the complete fine-grained urban flow map $X_{fg,D}^t \in \mathbb{R}^{2\times NI\times NJ}$ for the target city D at time t.

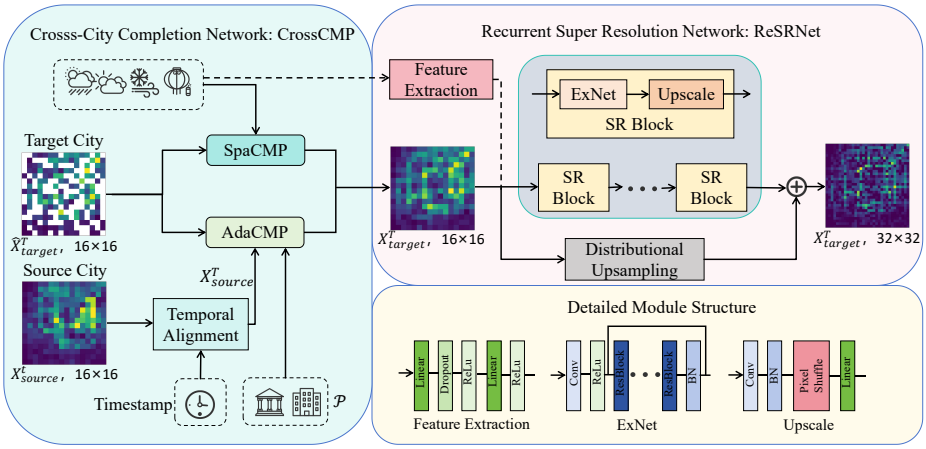


Fig. 2: Framework of the proposed CrossMF model.

4 Methodology

4.1 Cross-City Completion Network

The CrossCMP network completes missing values in coarse-grained urban flow maps of a target city by leveraging data from potentially richer source city and

local spatio-temporal dynamics from the limited data in the target city, as shown in Fig. 2. Let the input be a time series $\hat{X}_{target} \in \mathbb{R}^{T\times C\times I\times J}$. To conduct data completion over the regions where the data are unavailable, we first define the mask operation as follows:

$$\mathbf{M}_{cg}(r_{i,j}) = \begin{cases} 1 & \text{if } r_{i,j} > 0 \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where $M_{cg}()$ is a mask function,where marks the regions without observations as 0 and the regions with data as 1. CrossCMP produces a completed coarse-grained map $X_{cmp} \in \mathbb{R}^{C\times I\times J}$ through two complementary branches: Local Spatio-Temporal Completion (SpaCMP) and Adaptive Cross-City Completion (AdaCMP).

Local Spation-Temporal Completion This module aims to capture local spatial correlations and their temporal evolution inherent in urban flow data. By modeling how flow patterns in nearby regions influence each other over time, SpaCMP can effectively complete missing values based on observed spatio-temporal context within the target city. This is achieved using a recurrent architecture employing convolutional layers to process the spatial map at each time step while propagating temporal information via a hidden state.

Let $\hat{X}_{cg}^{t}$ be the input map at time t and H^{t-1} be the hidden state from the previous step. The hidden state H^t is computed as follows:

$$H^t = \sigma(\mathcal{F}(\hat{X}_{cg}^{t}, W_x, b_x) + \mathcal{F}(H^{t-1}, W_h, b_h)). \tag{4}$$

Here, σ represents a non-linear activation function, and $\mathcal{F}$ denotes the standard 2D convolution operation with learnable weight kernels (W_x, W_h) and bias terms (b_x, b_h). The application of $\mathcal{F}$ captures local spatial dependencies. By recurrently applying Eq 4 for $t = 1, ..., T$, the network integrates information across both space and time. The final hidden state after processing all time steps, $X_{spa} = H^T$, encapsulates the learned local spatio-temporal representation and serves as the output of this branch.

Adaptive Cross-City Completion However, SpaCMP's reliance on local spatio-temporal patterns fails under extensive target city data scarcity, potentially leaving large gaps. To ensure robust completion even with severe scarcity, we introduce the Adaptive Cross-City Completion (AdaCMP) module. AdaCMP complements SpaCMP by using Point-of-Interest (POI) guided spatial similarity and adaptively leveraging a data-rich source city when target information is insufficient. Meaningful regional similarity is needed to leverage spatial relationships with missing data; we use POI distributions for this. Since regional POI distributions reflect urban function (e.g., commercial, residential), which shapes mobility and flow patterns (volume, direction, timing), comparing POI profiles identifies functionally similar regions expected to exhibit similar flows. This provides a reliable basis for knowledge transfer, even across different cities. We first

define a function $\mathrm{Sim}(r_a, r_b)$ to compute the similarity between the POI feature vectors P_{r_a} and P_{r_b} of any two regions r_a, r_b. Cosine similarity is employed:

$$\mathrm{Sim}(r_a, r_b) = \frac{P_{r_a} \cdot P_{r_b}}{||P_{r_a}||_2 ||P_{r_b}||_2}. \tag{5}$$

Adaptive Completion Logic: For each target region r_k where the data is missing ($M(r_k) = 0$), AdaCMP adaptively selects the best available information source based on POI similarity using a prioritized strategy:

Intra-City Similarity: It identifies the region r^*_{target} within the target city's observed regions $\mathcal{R}_{target} = \{r_j | M(r_j) = 1\}$ that is most similar to r_k based on POI data:

$$r^*_{target} = \underset{r_j \in \mathcal{R}_{target}}{\arg\max}\ \mathrm{Sim}(r_k, r_j). \tag{6}$$

If the similarity $\mathrm{Sim}(r_k, r^*_{target})$ meets or exceeds a predefined threshold θ, the flow value from this most similar observed region within the target city is used for completion:

$$X^T_{fill}(r_k) = X^T_{target}(r^*_{target}). \tag{7}$$

Cross-City Similarity: If the intra-city similarity is below the threshold θ, indicating insufficient guidance from within the target city, the module attempts to leverage information from the source city. It identifies the region r^*_{source} in the source city $\mathcal{R}_{source}$ most similar to r_k:

$$r^*_{source} = \underset{r_l \in \mathcal{R}_{source}}{\arg\max}\ \mathrm{Sim}(r_k, r_l). \tag{8}$$

If the source data $X_{source}(r^*_{source})$ at the corresponding location is considered valid, its value is used:

$$X^T_{fill}(r_k) = X^T_{source}(r^*_{source}). \tag{9}$$

Let $X_{fill} \in \mathbb{R}^{C \times H \times W}$ be the map containing the completed values $X_{fill}(r_k)$ for all originally missing locations. The final output of the AdaCMP branch, X_{ada}, is constructed by combining the original observed values with the newly completed values:

$$X_{ada} = M \odot \hat{X}^t_{cg} + (1 - M) \odot X_{fill}, \tag{10}$$

where $\odot$ denotes element-wise multiplication. This adaptive cross-city strategy significantly enhances completion capabilities, especially in data-scarce target regions, by intelligently borrowing information from a data-richer source city based on functional similarity.

Module Combination The CrossCMP network integrates the complementary information captured by two branches. The output from the local spatio-temporal branch X_{spa} and the adaptive cross-city completion branch X_{ada} are fused using a learnable weight ω:

$$X_{cmp} = \omega \cdot X_{spa} + (1 - \omega) \cdot X_{ada}. \tag{11}$$

This learnable weight ω allows the model to dynamically balance the contributions from local spatio-temporal patterns and POI-based spatial similarities during end-to-end training, yielding the final conpleted coarse-grained map X_{cmp}.

4.2 Recurrent Super-Resolution Network

Feature Extraction The ReSRNet first processes the input completed coarse map X_{cmp} to extract an initial set of features suitable for the subsequent enhancement and upscaling tasks. If optional external features E_{ext} (e.g., weather data, holidays) are provided, they are processed by a separate feature extraction sub-network, and the resulting embeddings are typically fused with the features derived from X_{cmp}. Let the output feature map after the extraction and fusion be denoted as F:

$$F^0 = \mathcal{F}_{extract}(X_{cmp}, E_{ext}), \tag{12}$$

where $\mathcal{F}_{extract}$ represents the combined operations of convolution and external feature integration.

Recurrent Super-Resolution Blocks The core of the ReSRNet comprises a sequence of L stacked Recurrent Super-Resolution blocks. These blocks are designed to progressively refine the feature representation while simultaneously increasing the spatial resolution. Each block takes the feature map F^{t-1} from the preceding block and produces a higher-resolution feature map F^t. The total upscaling N is achieved across these stages.

Within each block, a recurrent mechanism iteratively refines the features over S steps ($s = 1, ..., S$). Let H_0^t be the initial state derived from the block input F^{t-1}. The refinement process uses a shared-parameter module $\mathcal{F}_{refine}$ based on residual connections:

$$H_s^t = \mathcal{F}_{refine}(H_{s-1}^t). \tag{13}$$

The refined features H_s^t from each internal step are then upscaled and potentially post-processed by $Upscale$. The final output F^t for stage t aggregates the information from all refinement steps via a weighted summation, allowing contributions from features at different refinement levels:

$$F^t = \sum_{s=1}^{S} \alpha_s \cdot \text{Upscale}(H_s^t). \tag{14}$$

After the final stage (t=L), the resulting feature map F^L captures the high-frequency details learned through the staged, recurrent process. The final high-resolution ouput, X_{fg}, is then obtained via a global residual connection as follows:

$$X_{fg} = \mathcal{F}_{idisup}(X_{cmp}, N) + F^L, \tag{15}$$

where $\mathcal{F}_{disup(X_{cmp},N)}$ represents the completed coarse-grained map X_{cmp} after being upsampled by a factor of N using a distributional upsampling function. The residual designed facilitates the learning of these fine details, learning to the final high-resolution output X_{fg}.

4.3 Training Strategy

The proposed CrossFM framework, comprising the Cross-City Completion Network (CrossCMP) and the Recurrent Super-Resolution Network (ReSRNet), is trained in an end-to-end manner. The objective combines two pixel-wise loss terms. The completion loss $\mathcal{L}_{cmp}$ quantifies the difference between the completed coarse map output X_{cmp} and its corresponding ground truth X'_{cg}. The super-resolution loss $\mathcal{L}_{sr}$ measures the difference between the final fine-grained map X_{fg} and its ground truth X'_{fg}:

$$\mathcal{L}_{cmp} = ||X_{cmp} - X_{cg}||_F^2, \tag{16}$$

$$\mathcal{L}_{sr} = ||X_{fg} - X'_{fg}||_F^2. \tag{17}$$

The total loss $\mathcal{L}$ minimized during end-to-end training is a weighted combination of these two components:

$$\mathcal{L} = \lambda\mathcal{L}_{cmp} + \mu\mathcal{L}_{sr}, \tag{18}$$

where λ and μ are hyper-parameters balancing the contribution of each task.

5 Experiment

Table 1: Dataset Description

Dataset	BJTaxi	NYCTaxi
Longitude	(115.42, 117.51)	(-74.25, -73.70)
Latitude	(39.44, 41.06)	(40.50, 41.08)
Time Span	3/1/2015-6/30/2015	1/1/2016-6/30/2015
Time Interval	30 minutes	30 minutes
Coarse-grained Shape	$32 \times 32/64 \times 64$	$32 \times 32/64 \times 64$
Fine-grained Shape	128×128	128×128
Upscaling Factor	2/4	2/4
#POI	79063	177824

5.1 Datasets

We evaluate CrossFM using two real-world datasets: BJTaxi, the target city for completion and super-resolution, and NYCTaxi, the source city for cross-city information. Table 1 summarizes their specifications after preprocessing.

- **BJTaxi**: This dataset comprises taxi trip records from Beijing, China (March 1 - June 30, 2015), processed into coarse-grained urban flow maps. The data is split into training (70%), validation (20%), and test (10%) sets.
- **NYCTaxi**: This dataset contains taxi trip records from New York City, USA (January 1 - June 30, 2016). It serves as an external data source for cross-city insights and is not split for training/validation/testing.

5.2 Experimental settings

Evaluation Metrics. We use Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) to evaluate the inference performance. .

Baselines. We compare CrossFM against several baseline methods. These include straightforward statistical approaches like Mean partition (**Mean**), which evenly distributes coarse-grained flow, and Historical Average (**HA**), which applies this distribution based on historical averages. We also consider advanced deep learning models for Fine-grained Urban Flow Inference (FUFI). Specifically, **UrbanFM** [12] utilizes distributional upsampling and fuses external factors. **UrbanPy** [16] employs a pyramid architecture with multiple components for upsampling and refinement. **UrbanSTC** [18] applies contrastive self-supervised learning for efficient pre-training, especially in low-resource scenarios. The most recent state-of-the-art model, **MT-CSR** [11], tackles FUFI with potentially incomplete coarse data through multi-task learning for simultaneous completion and super-resolution.

Table 2: Model Performance on BJTaxi Dataset (Best results bold, second best underlined)

Model				Mean	HA	UrbanFM	UrbanPy	UrbanSTC	MT-CSR	CrossFM	Improve
BJTaxi	2	40%	MAE	9.50	9.30	7.50	7.23	**5.99**	6.10	6.05	-1.0%
			RMSE	26.60	26.04	21.02	20.24	**17.10**	17.40	17.25	-0.9%
		60%	MAE	10.51	10.30	8.22	8.00	6.90	6.85	**6.45**	5.8%
			RMSE	29.43	28.84	22.96	22.42	19.60	19.45	**19.01**	2.3%
		80%	MAE	11.80	10.67	9.30	9.12	8.05	8.01	**7.12**	11.1%
			RMSE	33.04	28.81	26.04	25.48	22.70	22.59	**20.32**	10.1%
	4	40%	MAE	10.88	10.67	8.50	8.27	7.02	7.10	**7.01**	0.1%
			RMSE	29.38	28.81	22.86	20.99	20.56	19.21	**19.19**	0.1%
		60%	MAE	12.31	12.07	9.50	9.85	8.05	7.99	**7.56**	5.4%
			RMSE	33.24	32.59	25.45	23.73	21.95	21.83	**19.95**	8.6%
		80%	MAE	13.92	13.66	10.55	11.41	9.35	9.29	**7.85**	15.5%
			RMSE	37.58	36.88	28.14	27.19	25.00	24.81	**20.33**	18.1%

5.3 Performance Comparison

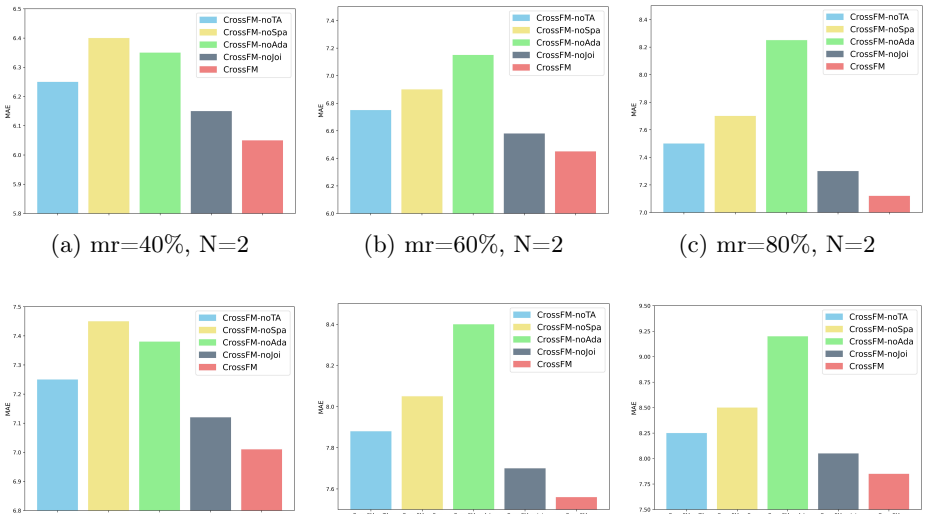
As Table 2 shows, we can observe that most methods degrade as missing data increases. Simple baselines (Mean, HA) perform poorly, unable to model complex spatio-temporal dynamics. Sophisticated methods like UrbanFM and UrbanPy, while better, still falter, potentially lacking inherent mechanisms for large missing blocks without specific completion modules or retraining. Models considering spatio-temporal correlations or advanced architectures (e.g., UrbanSTC, MT-CSR) handle missing data better than simpler methods. MT-CSR, designed for the same joint task, is a strong competitor but consistently outperformed by CrossFM, especially as scarcity increases. This suggests CrossFM's adaptive cross-city completion strategy offers benefits over MT-CSR's auxiliary completion under severe scarcity. UrbanSTC is also competitive but surpassed by

CrossFM, indicating CrossFM's targeted cross-city knowledge transfer provides an edge beyond capturing local spatio-temporal patterns alone.

Amidst these trends, CrossFM consistently achieves the best performance across all evaluated scenarios. More importantly, while baseline accuracy declines sharply with increasing data scarcity, CrossFM demonstrates significantly greater robustness via a much more gradual performance decrease. For instance, comparing results at 80% versus 40% missing data, the increase for CrossFM is considerably less pronounced than for most baseline methods.

CrossFM's robustness stems from its unique design. The key is the AdaCMP module, which mitigates data scarcity by adaptively borrowing information from a source city based on POI similarity. Furthermore, the joint end-to-end training of the completion (CrossCMP) and super-resolution (ReSRNet) modules creates a powerful synergy, enhancing overall accuracy and outperforming less integrated approaches.

5.4 Ablation Study



(a) mr=40%, N=2 (b) mr=60%, N=2 (c) mr=80%, N=2

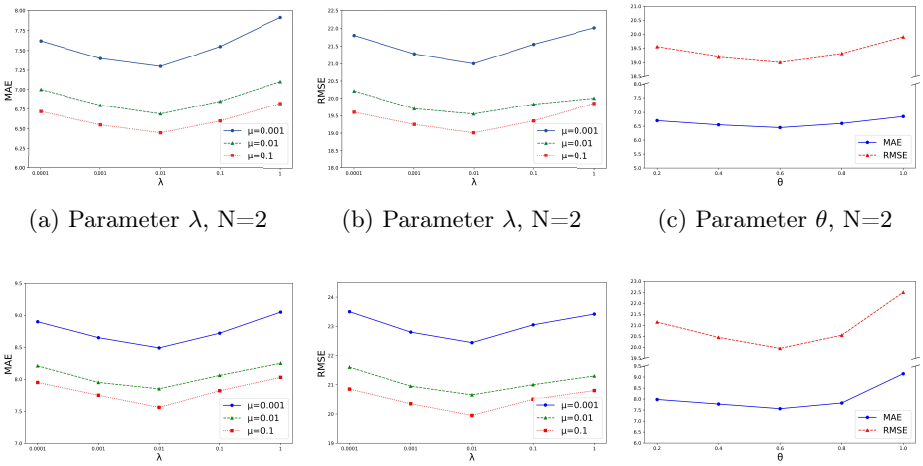
(d) mr=40%, N=4 (e) mr=60%, N=4 (f) mr=80%, N=4

Fig. 3: Ablation study of CrossFM on BJTaxi. 'mr' means 'missing rate'

We compare the full CrossFM model against variants where specific components are ablated: **CrossFM-noSpa**. This variant removes the local Spatio-temporal completion module. **CrossFM-noAda**. Removing the Adaptive Cross-City completion. **CrossFM-noTA** This variant removes Time Alignment. **CrossFM-noJoi**. This variant uses separate training instead of joint optimization.

The performance under different missing rates (mr) and scale factor (N) is illustrated in Fig. 3. The results consistently demonstrate that CrossFM achieves the best performance across all variants, indicating that all ablated components contribute positively to the final performance. Among the variants, CrossFM-noAda results in the most significant performance degradation, yielding the highest MAE in all depicted cases. Conversely, CrossFM-noJoi shows the smallest performance drop compared to the full model, suggesting that while joint end-to-end optimization provides benefits, the core components function effectively even when trained sequentially. The CrossFM-noTA and CrossFM-noSpa also led to performance decreases, further confirming their positive contributions to the framework. These findings collectively underscore the effectiveness of each component.

5.5 Parameter Analysis



(a) Parameter λ, N=2 (b) Parameter λ, N=2 (c) Parameter θ, N=2

(d) Parameter λ, N=4 (e) Parameter λ, N=4 (f) Parameter θ, N=4

Fig. 4: Parameter study of CrossFM on BJTaxi. Missing rate = 60%

We analyze the impact of the completion loss weight λ in the final objective function (Equation 18) and the POI similarity threshold θ used within the AdaCMP module (described in Section 4). Experiments were conducted under a 60% missing rate setting, and the results are summarized in Fig. 4. For the loss weights, we varied λ within the range 0.0001, 0.001, 0.01, 0.1 and μ in the range 0.001,0.01,0.1. As observed, performance initially improves as λ increases from very low values, but degrades slightly when λ is set equal to μ (0.1), suggesting a slight emphasis on the super-resolution task yields better overall results. For the POI similarity threshold θ, we tested values in 0.2, 0.4, 0.6, 0.8, 1.0.

The results indicate that performance suffers when the threshold is too low (potentially accepting less relevant intra-city matches) or too high (under-utilizing intra-city information or forcing reliance on cross-city data, with $\theta = 1.0$ performing worst). Based on these empirical results, we determined the optimal settings for our experiments to be $\lambda = 0.01$ alongside $\mu = 0.1$ and $\theta = 0.6$, which achieved the best performance.

5.6 Visualization

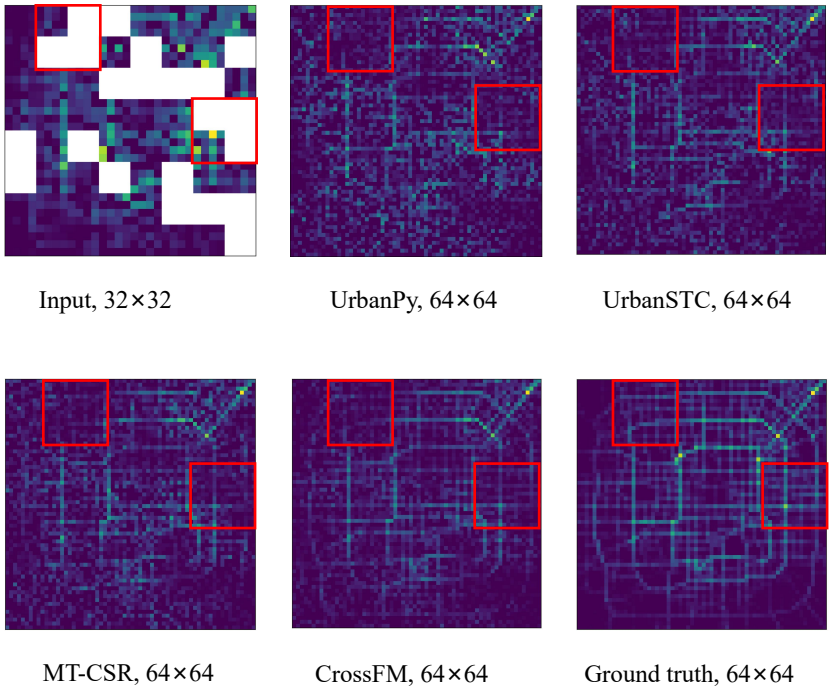


Fig. 5: Visualization of the urban flows inference with different methods on the BJTaxi. The cell regions in white color denote that the urban flow data are unavailable

To further intuitively demonstrate the model performance, we visualize the fine-grained urban flow inference results generated by different methods alongside the ground truth. Fig. 5 compares visualized heat maps (BJTaxi dataset, 32x32 input to 64x64 output, 40% missing rate) from CrossFM against baselines UrbanPy, UrbanSTC, and MT-CSR, alongside the ground truth. Visual inspection reveals that while baselines capture general patterns, UrbanPy lacks sharpness, UrbanSTC misses fine-grained hotspots, and MT-CSR shows minor inaccuracies, particularly in completed regions. In contrast, CrossFM generates

maps visually closest to the ground truth, accurately reconstructing both sharp details and high-flow hotspots. This visual superiority confirms CrossFM's effectiveness in generating high-fidelity fine-grained urban flow maps.

6 Conclusion

We proposed CrossFM, a framework leveraging cross-city dynamics to infer fine-grained urban flow from sparse observations. CrossFM integrates a POI-guided cross-city completion network (CrossCMP) and a recurrent super-resolution network (ReSRNet) via end-to-end training. Experiments show CrossFM outperforms state-of-the-art methods. While our current temporal alignment is simple, future work will focus on developing more sophisticated techniques to further improve inference accuracy in complex urban environments.

Acknowledgments

This work is supported by the Natural Science Foundation of China No. 62472196, Jilin Science and Technology Research Project 202301 01067JC.

References

1. Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans. Graph. **28**(3), 24 (2009)
2. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. IEEE Transactions on image processing **13**(9), 1200–1212 (2004)
3. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2015)
4. Gao, H., Su, Y., Yang, F., Yang, Y.: Fine-grained data inference via incomplete multi-granularity data. In: Proceedings of the ACM on Web Conference 2025. pp. 3377–3388 (2025)
5. Guo, W., Zhuang, F., Zhang, X., Tong, Y., Dong, J.: A comprehensive survey of federated transfer learning: challenges, methods and applications. Frontiers of Computer Science **18**(6), 186356 (2024)
6. He, K., Sun, J.: Image completion approaches using the statistics of similar patches. IEEE transactions on pattern analysis and machine intelligence **36**(12), 2423–2435 (2014)
7. Hong, Y., Chen, L., Wang, L., Xie, X., Luo, G., Wang, C., Chen, L.: Stkopt: Automated spatio-temporal knowledge optimization for traffic prediction. In: Proceedings of the ACM on Web Conference 2025. pp. 2238–2249 (2025)
8. Ji, J., Wang, J., Huang, C., Wu, J., Xu, B., Wu, Z., Zhang, J., Zheng, Y.: Spatio-temporal self-supervised learning for traffic flow prediction. In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 4356–4364 (2023)

9. Jiang, Y., Yang, Y., Xu, Y., Wang, E.: Spatial-temporal interval aware individual future trajectory prediction. IEEE Transactions on Knowledge and Data Engineering **36**(10), 5374–5387 (2024). https://doi.org/10.1109/TKDE.2023.3332929
10. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1646–1654 (2016)
11. Li, J., Wang, S., Zhang, J., Miao, H., Zhang, J., Yu, P.S.: Fine-grained urban flow inference with incomplete data. IEEE Transactions on Knowledge and Data Engineering **35**(6), 5851–5864 (2023)
12. Liang, Y., Ouyang, K., Jing, L., Ruan, S., Liu, Y., Zhang, J., Rosenblum, D.S., Zheng, Y.: Urbanfm: Inferring fine-grained urban flows. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 3132–3142 (2019)
13. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: Proceedings of the European conference on computer vision (ECCV). pp. 85–100 (2018)
14. Luo, H., Zhuang, F., Xie, R., Zhu, H., Wang, D., An, Z., Xu, Y.: A survey on causal inference for recommendation. The Innovation **5**(2) (2024)
15. Nazeri, K., Ng, E., Joseph, T., Qureshi, F., Ebrahimi, M.: Edgeconnect: Structure guided image inpainting using edge prediction. In: Proceedings of the IEEE/CVF international conference on computer vision workshops. pp. 0–0 (2019)
16. Ouyang, K., Liang, Y., Liu, Y., Tong, Z., Ruan, S., Zheng, Y., Rosenblum, D.S.: Fine-grained urban flow inference. IEEE transactions on knowledge and data engineering **34**(6), 2755–2770 (2020)
17. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2536–2544 (2016)
18. Qu, H., Gong, Y., Chen, M., Zhang, J., Zheng, Y., Yin, Y.: Forecasting fine-grained urban flows via spatio-temporal contrastive self-supervision. IEEE Transactions on Knowledge and Data Engineering **35**(8), 8008–8023 (2022)
19. Wang, A., Ye, Y., Song, X., Zhang, S., Yu, J.J.: Traffic prediction with missing data: A multi-task learning approach. IEEE Transactions on Intelligent Transportation Systems **24**(4), 4189–4202 (2023)
20. Wang, S., Cao, J., Philip, S.Y.: Deep learning for spatio-temporal data mining: A survey. IEEE transactions on knowledge and data engineering **34**(8), 3681–3700 (2020)
21. Wang, Z., Chen, J., Hoi, S.C.: Deep learning for image super-resolution: A survey. IEEE transactions on pattern analysis and machine intelligence **43**(10), 3365–3387 (2020)
22. Xu, Y., Wang, E., Yang, Y., Chang, Y.: A unified collaborative representation learning for neural-network based recommender systems. IEEE Transactions on Knowledge and Data Engineering **34**(11), 5126–5139 (2022). https://doi.org/10.1109/TKDE.2021.3054782
23. Zheng, Y., Zhong, L., Wang, S., Yang, Y., Gu, W., Zhang, J., Wang, J.: Diffuflow: Robust fine-grained urban flow inference with denoising diffusion model. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 3505–3513 (2023)