DIFFUSION PROBABILISTIC MODELING FOR FINE-GRAINED URBAN TRAFFIC FLOW INFERENCE WITH RELAXED STRUCTURAL CONSTRAINT

Xovee Xu, Yutao Wei, Pengyu Wang, Xucheng Luo, Fan Zhou, Goce Trajcevski[⋆]

University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China
[⋆]Iowa State University, Ames, IA 50011, USA

ABSTRACT

Inferring the citywide urban traffic flows is critical for numerous smart city applications such as urban planning, traffic control, and transportation management. Urban traffic flow inference problem aims to generate fine-grained flow maps from the coarse-grained ones. It is still challenging due to the lack of handling uncertainties of flow distributions and complex external factors that affect the inference performance. In this work, we propose a diffusion probabilistic augmentation-based network for considering the uncertainties of urban flows with a relaxed structural constraint and a disentangled scheme for flow map and external factor learning. Experiments are conducted on four large-scale urban flow datasets, and the results show that our method achieves significant performance improvements over strong baselines.

***Index Terms*—** Fine-grained urban flow inference, urban computing, diffusion probabilistic modeling, mobile sensing

1. INTRODUCTION

Inferring and predicting fine-grained citywide urban traffic flows benefit numerous transportation fields such as city traffic monitoring, urban planning, and intelligent traffic management. For example, spatial-temporal traffic flow patterns are studied in [1] to forecast future traffic flows accurately, and geo-contextual multi-task embedding learners are utilized in [2] for transportation scenario planning.

Typical transportation systems often require a large number of monitoring sensors to obtain a fine-grained view of the traffic conditions (e.g., taxi, bike, and crowd flows), which needs expensive maintenance, deployment, and electricity costs every year [3, 4, 5, 6, 7, 8]. As an important task in urban computing, fine-grained urban traffic flow inference (FUFI) aims to infer the fine-grained urban flows from the coarse-grained ones. FUFI can effectively reduce the number of deployed devices while keeping the original data granularity unchanged and the inference accuracy satisfactory.

Existing FUFI methods generally followed the learning scheme of single image super-resolution (SISR) and additionally considered the rich external factors (e.g., date, weather, and holidays that may greatly influence the volumes of urban flows) and the structural constraint imposed by FUFI. For example, UrbanFM [9] – which first formulates the FUFI problem – devised a convolutional network-based feature extractor to handle the spatial flow correlations and an N^2-Normalization layer to distribute inferred urban flows. FODE [10] proposes an ordinary differential equation-based model to overcome the unstable gradient computation problem in FUFI and balances inference accuracy and computational efficiency. UrbanPy [11] extends UrbanFM by employing a pyramid architecture with geographic embeddings, non-shared convolution, and distributional loss.

Limitation. Notwithstanding the improvements achieved on FUFI, prior methods confront several key limitations: (*i*) The urban traffic flows are inherently dynamic under significant uncertainty, they are influenced by many factors, such as emergency events, weather conditions, and traffic jams. Although certain factors are considered by existing methods, many other factors, whether endogenous or exogenous, explicit or implicit, greatly affect the flow distribution. Thus, there is a need for deterministic methods for handling of flow uncertainties. (*ii*) Existing models usually adopt the distributional upsampling [9] to obey the structural constraint of FUFI and speed the model's convergence process. However, given that information loss will inevitably lead to inference errors when upsampling the coarse-grained flow map, the too strict structural constraint will force the model to allocate more (or less) flow volumes to certain regions and, as a result, lowers the inference performance. (*iii*) The learning of flow map features and the learning of external factors are often performed jointly in existing methods. Although this scheme is efficient, the learned tangled feature maps sacrifice interpretability and may further lead to overfitting.

Present work. To address the limitations mentioned above, we propose a Diffusion Probabilistic modeling-based network for fine-grained Traffic Flow Inference (DP-TFI) – a novel framework integrating urban flow uncertainties and relaxed

This work was supported in part by the NSFC under Grant Nos. 62062077 and 62176043, NSF of Sichuan Province, China, under Grant No. 2022NSFSC0505, Sichuan Science and Technology Program under Grant No. 2020YFG0053, and NSF under Grant SWIFT No. 2030249.

Corresponding author: Xucheng Luo, xucheng@uestc.edu.cn.

This is the author version of the paper presented at ICASSP '23.
doi: 10.1109/ICASSP49357.2023.10096169

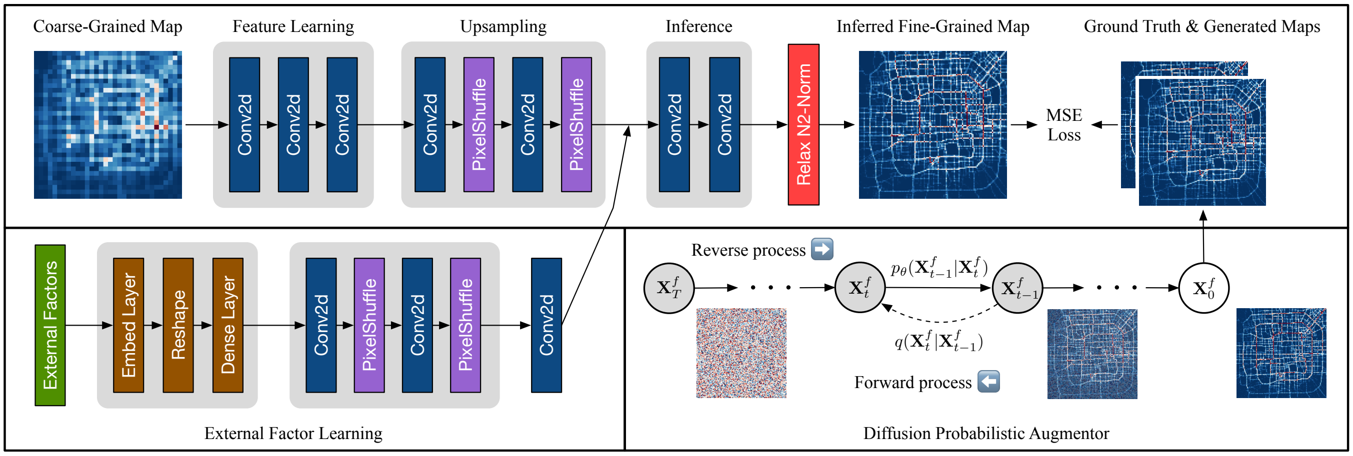

Fig. 1. The framework of DP-TFI. It takes the coarse-grained flow map and external factors as two streams of inputs and uses diffusion probabilistic augmentations and relaxed distributional upsampling to infer the fine-grained traffic flow maps.

structural constraint in a disentangled manner. Specifically, (i) we design a probabilistic urban flow augmentor that progressively generates new fine-grained flow maps from a simple distribution (e.g., an isotropic Gaussian) into a data distribution. The generated flow maps can be seen as different instances with uncertainty linked back to the same coarse-grained flow map. (ii) we relax FUFI's structural constraint by a new distributional upsampling layer that is both flexible and effective. (iii) we employ a disentangled learning scheme for urban flow dynamics and external factors. Extensive experiments conducted on four large-scale citywide urban flow datasets demonstrate the effectiveness of our proposed DP-TFI model over state-of-the-art FUFI baselines.

2. PROBLEM DEFINITION

In this section, we first give the necessary background information, and then formally define the problem of FUFI.

Definition 1 (Grid Flow Map)**.** *Given a city flow map* $\mathbf{X} \in \mathbb{R}_+^{I\times J}$ *partitioned into* $I \times J$ *grid, each entry* x_{ij} *of the map denotes the traffic flow volume during a period of time.*

Definition 2 (Superregion and Subregion)**.** *The map cells in the coarse-grained flow map are superregions and the map cells in the fine-grained flow map are subregions. Each of the superregions in the coarse-grained flow map corresponds to* $N \times N$ *subregions in the fine-grained flow map. Here* N *is the scaling factor.*

Definition 3 (Structural Constraint)**.** *Given a superregion and its corresponding* N^2 *subregions, the flow volume* x^c_{ij} *in the superregion is equal to the sum of volumes in the subregions:*

$$x^c_{ij} = \sum_{i',j'} x^f_{i'j'}, \quad s.t. \lfloor \frac{i'}{N} \rfloor = i, \lfloor \frac{j'}{N} \rfloor = j, \tag{1}$$

where $i = 1, 2, \ldots, I$ *and* $j = 1, 2, \ldots, J$.

FUFI problem: given a coarse-grained flow map $\mathbf{X}^c \in \mathbb{R}_+^{I\times J}$ and its corresponding external factors $\mathbf{f} \in \mathbf{R}^{d_f}$, we aim to infer the fine-grained flow map $\mathbf{X}^f \in \mathbb{R}^{NI\times NJ}$.

3. METHOD

The overview of the proposed DP-TFI is depicted in Fig. 1.

Disentangled Flow Feature Learning. For a coarse-grained flow map $\mathbf{X}^c$ and its corresponding external factors $\mathbf{f}$, we use two streams of convolutional neural networks to learn their representations in different latent spaces. Specifically, we use three convolutional layers – each has 3x3 filter size, F filters, and followed by a ReLU activation function – to learn low-level hidden feature maps $\mathbf{H}^c \in \mathbb{R}^{I\times J\times F}$. Then we upscale the coarse-grained feature map to fine-grained feature maps $\mathbf{H}^f \in \mathbb{R}^{N\times J\times F}$ via stacked PixelShuffle and convolutional layers. For external factor learning, we first transform $\mathbf{f}$ into representation via multiple embedding layers and dense layers, then we transform the representation to an external factor feature map $\mathbf{H}^{ef} \in \mathbb{R}^{NI\times NJ\times 1}$ via stacked PixelShuffle and convolutional layers. In doing so, the learning of external factor feature maps $\mathbf{H}^f$ are decoupled with the learning of flow feature maps $\mathbf{H}^{ef}$, which encourages the inference network to learn urban flow dynamics solely based on the internal characteristics of the flows, e.g., regional differences and spatial dependencies.

Relaxed Structural Constraint. Traditional FUFI methods often impose a too strict structural constraint during urban flow map inference. For example, the N^2-Normalization layer ensures that the sum of inferred flow volumes in the subregions equals the corresponding flow volume in the superregion. We propose a Relaxed Distributional Upsampling (RDU) module that allows the model to allocate additional or reduced flow volumes to subregions. Given the fact that information loss and inference errors are inevitable, RDU can help the model to infer more accurate flow maps for certain important regions without worrying too much about the flow budget. Specifically, given the last flow feature map $\mathbf{H}^{fl} = \mathbf{H}^{f} \oplus \mathbf{H}^{ef}$, we infer the final fine-grained flow map by a relaxed N^2-Normalization layer. We generate fine-grained flow distribution by:

$$\tilde{\mathbf{H}}^{fl}_{i'j'} = \mathbf{H}^{fl}_{i'j'} \quad / \sum_{\substack{i'=(\lfloor i/N \rfloor -1)N+1 \\ j'=(\lfloor j/N \rfloor -1)N+1}}^{\substack{i'=(\lfloor i/N \rfloor)N \\ j'=(\lfloor j/N \rfloor)N}} \mathbf{H}^{fl}_{i'j'}, \tag{2}$$

where $\tilde{\mathbf{H}}^{fl}_{i'j'} \in [0, 1]$. Then we use nearest neighbor upsampling with a relax coefficient μ and the scaling factor N to obtain the upscaled flow map $\mathbf{X}^{c}_{\text{relax}} \in \mathbb{R}^{NI \times NJ}_{+}$:

$$\mathbf{X}^{c}_{\text{relax}} = \text{Upsampling}(\mathbf{X}^{c}_{\text{up}}) \otimes \mathbf{R}^{f}, \tag{3}$$

$$\mathbf{R}^{f} = 2\mu \text{Sigmoid}(\text{Conv2d}(\mathbf{H}^{f})) - (\mu + 1)\mathbf{J}, \tag{4}$$

where $\mathbf{J}$ is the all-ones matrix and $\mathbf{R}^{f} \in \mathbb{R}^{NI \times NJ}$ is the relaxed flow matrix. At last, we have inferred fine-grained flow map as: $\hat{\mathbf{X}}^{f} = \mathbf{X}^{c}_{\text{relax}} \otimes \tilde{\mathbf{H}}^{fl}$. The optimization of the model is guided by mean squared error (MSE) between the inferred map and the ground truth:

$$\mathcal{L}_{\text{loss}} = ||\hat{\mathbf{X}}^{f} - \mathbf{X}^{f}||^2.$$

Diffusion Probabilistic Augmentor. To handle the uncertainties of the urban traffic flow inference process, we introduce a diffusion probabilistic augmentor (DPA) [12, 13, 14] to progressively generate new fine-grained urban flow maps from a simple distribution into a more complex data distribution. As shown in Figure 1, DPA takes a source fine-grained urban flow map $\mathbf{X}^{f}$ as input, then it uses a forward Markovian diffusion process q to add Gaussian noise to $\mathbf{X}^{f}_{0} = \mathbf{X}^{f}$ through T iterations gradually:

$$q(\mathbf{X}^{f}_{1:T}|\mathbf{X}^{f}_{0}) = \prod_{t=1}^{T} q(\mathbf{X}^{f}_{t}|\mathbf{X}^{f}_{t-1}), \tag{5}$$

$$q(\mathbf{X}^{f}_{t}|\mathbf{X}^{f}_{t-1}) = \mathcal{N}\left(\mathbf{X}^{f}_{t}|\sqrt{\alpha_t}\mathbf{X}^{f}_{t-1}, (1-\alpha_t)\mathbf{I}\right), \tag{6}$$

where $\alpha_{1:T}$ are hyper-parameters. We then define a reverse Markovian diffusion process p_θ which is the opposite of the forward process:

$$p_\theta(\mathbf{X}^{f}_{0:T}|\mathbf{X}^{f}) = p(\mathbf{X}^{f}_{T}) \prod_{t=1}^{T} p_\theta(\mathbf{X}^{f}_{t-1}|\mathbf{X}^{f}_{t}, \mathbf{X}^{f}), \tag{7}$$

$$p(\mathbf{X}^{f}_{T}) = \mathcal{N}\left(\mathbf{X}^{f}_{T}|\mathbf{0}, \mathbf{I}\right), \tag{8}$$

$$p_\theta(\mathbf{X}^{f}_{t-1}|\mathbf{X}^{f}_{t}, \mathbf{X}^{f}) = \mathcal{N}\left(\mathbf{X}^{f}_{t-1}|\sigma_\theta(\mathbf{X}^{f}, \mathbf{X}^{f}_{t}, \gamma_t), \sigma^2_t\mathbf{I}\right) \tag{9}$$

We then pass the generated new traffic flow map $\mathbf{X}^{f}_{0}$ through the N^2-Normalization layer to obey the structural constraint and ensure $\mathbf{X}^{f}_{0}$ is not too different from the original map. We use the DPA to help the inference model learn better flow map features with uncertainty and make the model more robust.

Table 1. Data Statistics of TaxiBJ

Data	**TaxiBJ**
time span	P1: Jul 1, 2013 - Dec 31, 2013
	P2: Feb 1, 2014 - Jun 30, 2014
	P3: Mar 1, 2015 - Jun 30, 2015
	P4: Nov 1, 2015 - Mar 31, 2016
time interval	30 minutes
coarse-grained map size	32x32
fine-grained map size	128x128
upscaling factor N	4
latitude range	39°49'12N - 39°59'24N
longitude range	116°15'36E - 116°29'24E
temperature (°C)	[-24.6, 41.0]
wind speed (mph)	[0, 48.6]
weather condition	16 types

4. EXPERIMENTS

Data. The traffic flow data are collected from the citywide taxi flows in Beijing city from 2013 to 2015 [9] and are divided into four time spans (P1-P4). Detailed statistics of the data as well as external factors are shown in Table 1.

Baselines. To verify the effectiveness of our proposed DP-TFI method, we compare it with ten baselines including six SISR methods and four FUFI methods. (1) *SRCNN* [15]: is a convolutional SISR neural network. (2) *ESPCN* [16]: is a real time SISR model that proposes an efficient sub-pixel layer for feature map aggregation. (3) *VDSR* [17]: uses a deep learning-based residual architecture. (4) *DeepSD* [18]: is a statistical upscaling method for meteorological data which stacks multiple SRCNNs for SISR. (5) *SRResNet* [19]: stacks many residual blocks and employs perceptual loss for SISR. (6) *LapSRN* [20]: does not require bicubic interpolation and progressively reconstructs sub-brand residuals of fine-resolution images. (7) *UrbanFM* [9]: is the first work studies FUFI problem, which designs a residual-based inference network and an N^2-Normalization layer. (8) *FODE* [10]: introduces ordinary differential equations (ODEs) to balance the flow inference accuracy and computational efficiency. (9) *UrbanODE* [7]: incorporates two neural ODE blocks with attention mechanism. (10) *UrbanPy* [11]: extends UrbanFM by a cascading model for progressive urban flow inference and is the state-of-the-art for FUFI.

Experimental Settings. We use Adam optimizer with an initial learning rate of $1e^{-3}$. For a fair comparison, for all baselines and our model, the number of base channels F and batch size are set to 128 and 16, respectively. We implement our model with Torch library on GeForce RTX 3090 GPU. To evaluate the model performance from multiple facets, following previous FUFI methods [9, 10], we use three common metrics: root mean square error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE).

Table 2. Performance comparison between proposed DP-TFI and baselines.

Data	TaxiBJ P1			TaxiBJ P2			TaxiBJ P3			TaxiBJ P4		
Metrics	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE
SRCNN [15]	4.297	2.491	0.714	4.612	2.681	0.689	4.815	2.829	0.727	3.838	2.289	0.665
ESPCN [16]	4.206	2.497	0.732	4.569	2.727	0.732	4.744	2.862	0.773	3.728	2.228	0.711
VDSR [17]	4.159	2.213	0.467	4.586	2.498	0.486	4.730	2.548	0.461	3.654	1.978	0.411
DeepSD [18]	4.156	2.368	0.614	4.554	2.612	0.621	4.692	2.739	0.682	3.877	2.297	0.652
SRResNet [19]	4.164	2.457	0.713	4.524	2.660	0.688	4.690	2.775	0.717	3.667	2.189	0.637
LapSRN [20]	3.997	2.040	0.339	4.353	2.235	0.324	4.539	2.343	0.330	0.351	1.841	0.315
UrbanFM [9]	3.949	1.997	0.330	4.359	2.227	0.323	4.519	2.319	0.326	3.514	1.821	0.314
FODE [10]	4.058	2.142	0.403	4.432	2.377	0.395	4.628	2.490	0.417	3.583	1.947	0.396
UrbanODE [7]	4.042	2.135	0.406	4.432	2.357	0.394	4.601	2.460	0.408	3.559	1.929	0.391
UrbanPy [11]	3.944	1.998	0.333	4.315	2.210	0.323	4.436	2.272	0.318	3.470	1.801	0.313
DP-TFI	**3.853**	**1.926**	**0.298**	**4.240**	**2.144**	**0.292**	**4.404**	**2.227**	**0.287**	**3.429**	**1.759**	**0.292**

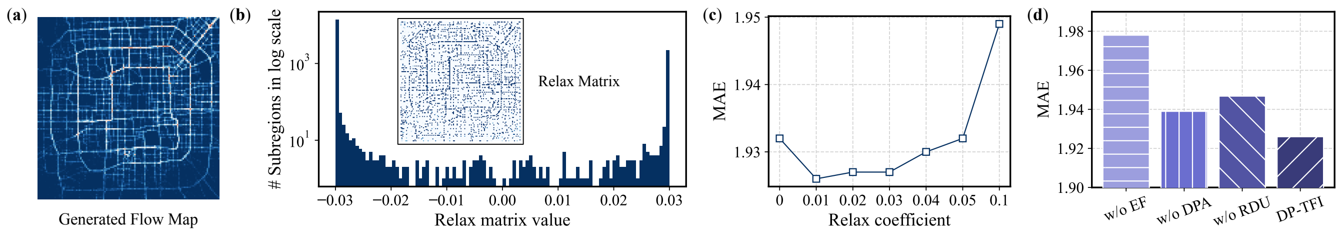

Fig. 2. Experimental analysis on TaxiBJ P1 dataset. (**a**): Case study of generated flow map; (**b**) the relax matrix entry value distribution and the relax matrix (inset); (**c**) parameter sensitivity of relax coefficient; (**d**) ablation study on three variants.

Experimental Results. The FUFI inference performance between our proposed DP-TFI and baselines on TaxiBJ P1-P4 datasets are shown in Table 2. We can see that DP-TFI has the best performance on all four datasets in terms of both RMSE, MAE, and MAPE. SISR-based methods generally perform worse than FUFI-based methods, which is because they do not take external factors into account and neglect the structural constraint imposed by FUFI. FIFU-based methods such as UrbanODE and UrbanPy achieve better performance due to their capability to fuse external factors and more powerful feature learning architecture. However, they cannot handle urban traffic flow uncertainties, and the external factor fusion is tangled with flow maps which may reduce the inference errors. Moreover, the too strict structural constraint limits the model on allocating accurate flows to subregions.

Analysis. In Fig. 2, we conducted several analyses of DP-TFI's model behaviors. First, we showed a case flow map generated by the diffusion probabilistic augmentor, which largely preserves the flow characteristics of the source flow map but also introduces uncertainty. The relax matrix value distribution indicates that the model prefers to break the structural constraint for most of the subregions (relax coefficient is set to 3%). Specifically, those regions with heavy traffic flows will have more allocated flows and vice versa. We also analyzed the influence of relax coefficient μ, which determines how flexible we allow the model to allocate flows. We varied the coefficient from 0 to 0.1, the result showed that μ around 0.02 leads to better performance. At last, we ablated the three proposed modules by removing any one of them in DP-TFI. The three variants are denoted as w/o EF (without external factor), w/o DPA (without diffusion probabilistic augmentor), and w/o RDU (without relaxed distributional upsampling). We can obverse that external factor fusion contributes the most to the improvement, which suggests external factors are essential for inferring an accurate traffic flow map. Not surprisingly, combining all modules achieves the best performance, which again verifies our motivation that modeling the uncertainties and (slightly) relaxing the structural constraints are beneficial for FUFI.

5. CONCLUSION

In this paper, a novel probabilistic model DP-TFI has been proposed to tackle the task of fine-grained urban traffic flow inference problem. It uses a diffusion probabilistic augmentor to handle the urban flow uncertainties and relax the structural constraint in a disentangled manner. Future work can further take the external factors and coarse-grained flow maps as conditions for probabilistic flow map generation and use loss regularization techniques to optimize the inference process.

6. REFERENCES

[1] Xiyue Zhang, Chao Huang, Yong Xu, Lianghao Xia, Peng Dai, Liefeng Bo, Junbo Zhang, and Yu Zheng, "Traffic flow forecasting with spatial-temporal graph diffusion network," in *AAAI*, 2021, pp. 15008–15015.

[2] Ana Alice Peregrino, Soham Pradhan, Zhicheng Liu, Nivan Ferreira, and Fabio Miranda, "Transportation scenario planning with graph neural networks," *arXiv:2110.13202*, 2021.

[3] Sang Keon Lee, Heeseo Rain Kwon, HeeAh Cho, Jongbok Kim, and Donju Lee, "International case studies of smart cities: Anyang, Republic of Korea," *IADB: Inter-American Development Bank*, 2016.

[4] Yuxuan Liang, Kun Ouyang, Yiwei Wang, Ye Liu, Junbo Zhang, Yu Zheng, and David S. Rosenblum, "Revisiting convolutional neural networks for citywide crowd flow analytics," in *ECML-PKDD*, 2020.

[5] Zhangtao Cheng, Joojo Walker, Ting Zhong, and Fan Zhou, "Modeling multi-view interactions with contrastive graph learning for collaborative filtering," in *IJCNN*, 2022.

[6] Fan Zhou, Xovee Xu, Goce Trajcevski, and Kunpeng Zhang, "A survey of information cascade analysis: Models, predictions, and recent advances," *ACM Computing Surveys*, vol. 54, no. 2, 2021.

[7] Fan Zhou, Xin Jing, Liang Li, and Ting Zhong, "Inferring high-resolutional urban flow with Internet of mobile things," in *ICASSP*, 2021, pp. 7948–7952.

[8] Haoyang Yu, Xovee Xu, Ting Zhong, and Fan Zhou, "Overcoming forgetting in fine-grained urban flow inference via adaptive knowledge replay," in *AAAI*, 2023.

[9] Yuxuan Liang, Kun Ouyang, Lin Jing, Sijie Ruan, Ye Liu, Junbo Zhang, David S Rosenblum, and Yu Zheng, "UrbanFM: Inferring fine-grained urban flows," in *SIGKDD*, 2019, pp. 3132–3142.

[10] Fan Zhou, Liang Li, Ting Zhong, Goce Trajcevski, Kunpeng Zhang, and Jiahao Wang, "Enhancing urban flow maps via neural ODEs," in *IJCAI*, 2020.

[11] Kun Ouyang, Yuxuan Liang, Ye Liu, Zekun Tong, Sijie Ruan, Yu Zheng, and David S. Rosenblum, "Fine-grained urban flow inference," *IEEE TKDE*, vol. 34, no. 6, pp. 2755–2770, 2022.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel, "Denoising diffusion probabilistic models," *NeurIPS*, vol. 33, pp. 6840–6851, 2020.

[13] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *ICML*, 2015, pp. 2256–2265.

[14] Haomin Wen, Youfang Lin, Yutong Xia, Huaiyu Wan, Roger Zimmermann, and Yuxuan Liang, "DiffSTG: Probabilistic spatio-temporal graph forecasting with denoising diffusion models," *arXiv:2301.13629*, 2023.

[15] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Image super-resolution using deep convolutional networks," *IEEE TPAMI*, vol. 38, no. 2, pp. 295–307, 2015.

[16] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *CVPR*, 2016, pp. 1874–1883.

[17] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *CVPR*, 2016, pp. 1646–1654.

[18] Thomas Vandal, Evan Kodra, Sangram Ganguly, Andrew Michaelis, Ramakrishna Nemani, and Auroop R Ganguly, "DeepSD: Generating high resolution climate change projections through single image super-resolution," in *SIGKDD*, 2017, pp. 1663–1672.

[19] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR*, 2017, pp. 4681–4690.

[20] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *CVPR*, 2017.