ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230519

GridFormer: Spatial-Temporal Transformer Network for Citywide Crowd Flow Prediction

Chaoqun Su[a], Chenwang Wu[a] and Defu Lian[a;*]

[a]University of Science and Technology of China

Abstract. Crowd flow prediction plays a vital role in various fields such as traffic management, public safety, and urban planning. The main challenge in crowd flow prediction lies in effectively modeling the periodic temporal dependency and long-range spatial dependency. In the temporal domain, crowd flow shows a strong periodicity which is exploited by existing works to build multi-time-scale spatial-temporal features. However, these works hardly consider the disturbance of periods, that is, the crowd flow is not strictly periodic. In the spatial domain, existing works mainly utilize CNN to capture spatial dependency, but the small receptive field of the convolution operator limits the ability to capture the long-range dependency between crowd flows in different regions. In this paper, we propose GridFormer, a Transformer network, in which a periodically shifted sampling method and attention mechanism are employed to handle the temporal shifting in the daily and weekly periodicity, and a pyramid 3D Swin Transformers network is designed to capture long-range spatial dependency in a hierarchical manner. Meanwhile, the pyramid 3D Swin Transformers network jointly models spatial-temporal features to enable better interaction between the spatial and temporal domains. Experimental results on three crowd flow datasets demonstrate that our GridFormer outperforms the state-of-the-art crowd flow prediction methods.

1 Introduction

Spatial-temporal prediction plays a crucial role in urban development by providing insights into future trends based on historical spatial and temporal dynamics. Among various spatial-temporal prediction tasks, crowd flow prediction holds significant importance in diverse application scenarios, ranging from emergency management [27, 8] and traffic control [26, 25] to urban planning [12]. Government agencies rely on crowd flow prediction to devise control measures and prevent potential stampedes during festival celebrations. Ride-sharing companies like Uber utilize crowd flow prediction to optimize taxi dispatch and meet the travel demands of city residents efficiently. In this paper, we specifically focus on grid-based crowd flow prediction, which involves forecasting the inflow and outflow of each region in a city. Here, each grid represents a region within the city, while inflow and outflow refer to the total traffic of individuals entering and leaving a particular region during a specific time interval. The task entails using historical observations of crowd flow as input and generating predictions for the subsequent time step.

The main challenge of citywide crowd flow prediction lies in how to model the periodic temporal dependency and long-range spatial dependency. Firstly, crowd flow data exhibits daily and weekly periodic patterns that are crucial for accurate predictions. However, it should be noted that these patterns are not strictly consistent [25]. For example, peak hours on weekdays may vary between 7:30 am and 10:00 am. Therefore, it is essential to account for temporal shifting within the periodicity to effectively utilize the periodic information for precise predictions. Secondly, with the rapid development of urban transportation, people can easily travel across the city in a short period using various modes of transport, such as taxis or subways. Consequently, the long-range spatial dependency between different regions significantly influences crowd movements. Therefore, capturing the complex long-range spatial dependency is significant for accurate citywide crowd flow prediction.

For the task of crowd flow prediction, several studies have been proposed based on deep learning techniques. However, existing approaches have limitations in addressing all of the aforementioned challenges. The first category of studies, including Deep-ST [28], ST-ResNet [27], and DeepSTN+ [12] aim to tackle these challenges by converting the 4D input tensor $(Timestep, Height, Width, Channel)$ into a 3D tensor $(Height, Width, Timestep \times Channel)$ through concatenating the channels at each timestep. Subsequently, CNNs are employed to capture the spatial dependency. *Although these methods demonstrate promising results, they have certain limitations. By simply concatenating the channels, the temporal dynamics are not fully captured, potentially leading to a loss of temporal information.* The second class of studies including DMVST-Net [26], STDN [25] and LMST3D [4] adopt a different approach by using local CNNs to capture spatial dependency among neighboring grids. STDN [25] also takes into account the temporal shifting in the periodicity, but only within the daily period range. *Notably, these methods can only effectively capture spatial dependency when the crowd flow map has a small mesh-grid number. The reliance on local CNNs imposes restrictions on the ability to model long-range dependencies, limiting their effectiveness in capturing comprehensive spatial information.* The third category of studies, including PCRN [30], Multitask-DF [29] and DeepCrowd [8] employ ConvGRU [1] and ConvLSTM [20] as basic units jointly model spatial-temporal features. *While these methods consider temporal and spatial dependency simultaneously, they still rely on CNN which does not have the capability of long-range modeling to capture spatial dependency and overlook the temporal shifting that occurs within the periodicity.*

To address the aforementioned challenges, we propose GridFormer, a novel Transformer network for crowd flow prediction. First, we propose a periodically shifted sampling method to take out

* Corresponding Author. Email: liandefu@ustc.edu.cn

relevant samples corresponding to the temporal shifted intervals of the daily and weekly periodicity and build a set of unique spatial-temporal features. The attention mechanism is utilized to handle the temporal shifting in the daily and weekly periodicity by learning to assign different weights to the information of each timestep. Inspired by Transformer models' excellent capability of capturing long-range dependency and the 3D Swin Transformer's excellent performance on video modeling (analogously to spatial-temporal modeling) tasks [15], we design a pyramid 3D Swin Transformers network to capture long-range spatial dependency. Meanwhile, benefiting from spatial-temporal joint modeling, the pyramid 3D Swin Transformers network enables better interaction between the spatial and temporal domains. Finally, a parametric-matrix-based fusion method is employed to fuse the spatial-temporal features of different temporal properties (i.e., hourly trend, daily trend, weekly trend). The proposed GridFormer is carefully optimized for addressing the long-range spatial dependency and periodic shifting in crowd flow prediction and accordingly proposes a novel pyramid 3D Swin Transformer and periodic shifting sampling. Our contributions can be summarized as follows:

- To address the issue of periodic shifting, we introduce a periodically shifted sampling method that enables the construction of a distinct set of multi-time-scale spatial-temporal features. Additionally, we leverage the attention mechanism to effectively capture the temporal shifting present in the daily and weekly periodic patterns. The multi-time-scale spatial-temporal features encompass three distinct temporal properties: hourly trend, daily trend, and weekly trend.
- We propose a pyramid 3D Swin Transformers network to effectively capture the intricate long-range spatial dependency and enable efficient joint modeling in the spatial-temporal domain. The pyramid 3D Swin Transformers network adopts a hierarchical approach to simultaneously capture local and global information. Specifically, we employ three independent pyramid 3D Swin Transformers networks to handle the spatial-temporal features associated with the three aforementioned temporal properties.
- To the best of our knowledge, we are the first to comprehensively address both of these pivotal issues simultaneously. Our proposed method effectively tackles the periodic shifting problem and captures long-range spatial dependency in the crowd flow prediction task. We conducted extensive experiments on three real-life crowd flow datasets with varying mesh-grid numbers to evaluate the performance of our approach. The results clearly demonstrate the superiority of our method over existing approaches.

2 Preliminaries

2.1 Problem Formulation

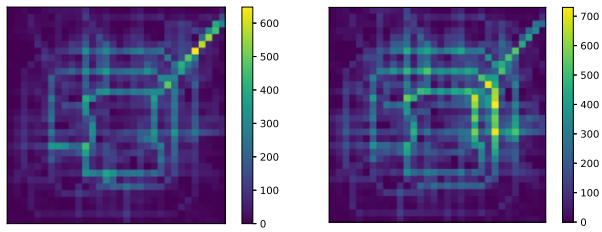
Definition 1 (**Region** [28]) To indicate the regions within the city, we partition a city into $H \times W$ grids based on the longitude and latitude, where each grid represents a region with all grids of equal size.

Definition 2 (**Inflow/outflow** [28]) Let $\mathbb{P}$ be a collection of trajectories at the t^{th} time interval. To express the crowd flows in the city, we define inflow and outflow for the region (h, w) at the t^{th} time interval as follows:

$$x_t^{h,w,in} = \sum_{T_r \in \mathbb{P}} |\{j > 1 | g_{j-1} \notin (h,w) \wedge g_j \in (h,w)\}|$$

$$x_t^{h,w,out} = \sum_{T_r \in \mathbb{P}} |\{j \geqslant 1 | g_{j-1} \in (h,w) \wedge g_j \notin (h,w)\}|$$

where $T_r : g1 \rightarrow g2 \rightarrow \cdots \rightarrow g_{|T_r|}$ is a trajectory in $\mathbb{P}$ and g_j is the geospatial coordinate; $g_j \in (h, w)$ means the point g_j lies within the grid (h,w), and vice versa; $|\cdot|$ denotes the cardinality of a set.



(a) (b)

Figure 1: Examples of 32×32 partitioned grid maps. Left to right: inflows and outflows in every region of Beijing

According to the above definitions, for a spatial region represented by a $H \times W$ grid map, there are 2 types of flows in each grid over time. At the t^{th} time interval, the crowd flow in all $H \times W$ grids can be denoted as a tensor $\mathbf{X}_t \in \mathbb{R}^{H \times W \times 2}$ where $(\mathbf{X}_t)_{h,w,0} = x_t^{h,w,in}$ and $(\mathbf{X}_t)_{h,w,1} = x_t^{h,w,out}$. The inflow and outflow matrix is shown in Figure 1.

Crowd Flow Prediction: Given the historical observations $\{\mathbf{X}_i | i = 1, 2, \cdots, n-1\}$, predict $\mathbf{X}_n$.

2.2 3D Swin Transformer

The Transformer [22] has been specifically designed for sequence modeling and transduction tasks, utilizing self-attention mechanisms to effectively capture long-range dependencies within the data. In the field of computer vision, Transformer models have demonstrated remarkable ability in capturing long-range dependency [2, 6, 24]. Moreover, Liu et al. [14] have further introduced the principles of locality, hierarchy, and translation invariance, endowing the Transformer with the capability to serve as a versatile backbone for various image recognition tasks. Swin Transformer disperses more attention to the connection of each patch by the application of shifted window and patch merging. By extending Swin Transformer from 2D to 3D space, Liu et al. [15] proposed 3D Swin Transformer. A 3D Transformer block consists of a 3D shifted window-based MSA module followed by a feed-forward network, specifically a 2-layer MLP, with GELU non-linearity in between.

3D Swin Transformer has demonstrated exceptional performance in various video modeling tasks due to its superior ability to simultaneously capture spatial and temporal dependency. Its hierarchical structure and multi-head self-attention with shifted window enable efficient modeling of long-range spatial dependency. Motivated by the remarkable capability of Transformer models in capturing long-range information and the outstanding performance of the 3D Swin Transformer in video modeling, we adopt the 3D Swin Transformer as the fundamental component for handling our high-dimensional sequential data, which shares similarities with video data in terms of spatial-temporal modeling tasks.

3 GridFormer

In this section, we introduce GridFormer and its components:(1) Periodically shifted sampling, (2) Pyramid 3D Swin Transformers network, and (3) Attention mechanism and fusion process in detail.

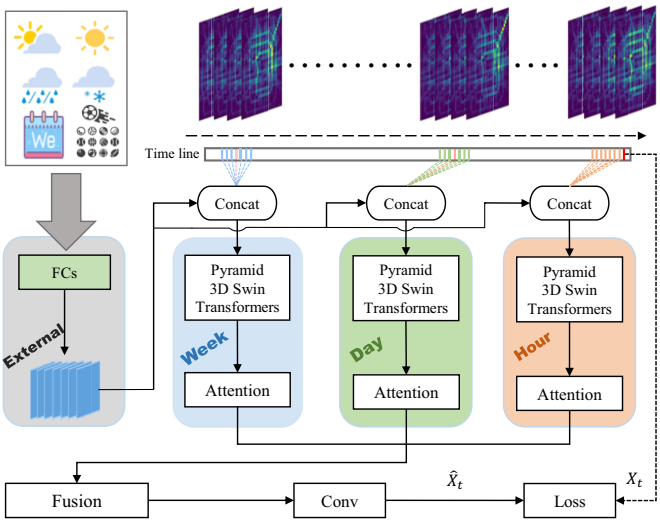


Figure 2: The architecture of GridFormer. The spatial-temporal features of three temporal properties are built from the sequence of crowd flow maps and external information.

Figure 2 illustrates the architecture of GridFormer, which comprises four main branches: hourly trend, daily trend, weekly trend, and external information. In this architecture, the inflow and outflow in each region are computed at half-hour intervals to construct a sequence of crowd flow maps. Using the periodically shifted sampling method, we construct a set of distinct spatial-temporal features: $\mathbf{X}^{Hour}$, $\mathbf{X}^{Day}$, and $\mathbf{X}^{Week}$, which correspond to recent time intervals (i.e., hourly trend), the previous day (i.e., daily trend), and the previous week (i.e., weekly trend), respectively. Each feature is represented as a 4D tensor $(Timestep, Height, Width, Channel = 2)$. External-info vector is sent to two fully connected layers and then reshaped to a 4D tensor of the same shape as the features tensor. The concatenated spatial-temporal features are fed into the pyramid 3D Swin Transformers network. Further, the attention mechanism is employed to capture the temporal shifting in the periodicity by learning to assign different weights to the information of each timestep. Finally, the fused features are obtained by aggregating the three branches using a parametric-matrix-based fusion. After fusion, a convolutional layer is employed to get the final prediction result. The details of GridFormer's components will be introduced below.

3.1 Periodically Shifted Sampling

Crowd flow exhibits periodic patterns across various time scales, such as daily or weekly cycles. For instance, the flow of individuals through a specific region tends to increase during rush hours and decrease during nighttime, repeating this pattern on a daily basis. On a weekly scale, the crowd flow pattern on weekdays is similar, while it differs from the pattern observed on weekends. These periodic patterns can be visualized in Figure 3.

However, the crowd flow is not strictly periodic [25]. For instance, peak hours on weekdays may vary between 8:00 am and 10:30 am. This temporal shifting of periodicity is illustrated in Figure 3. *In other words, there is a consistent temporal shift of several time intervals between periods*. We propose a periodically shifted sampling method to take out relevant samples corresponding to the temporal shifted intervals of periodicity. The set of unique spatial-temporal

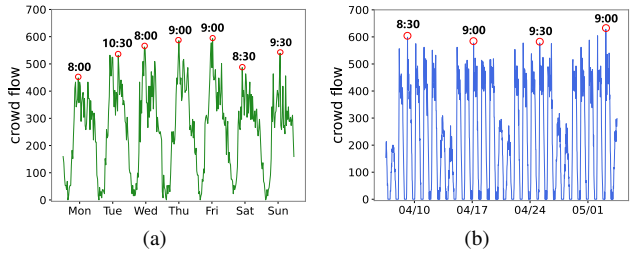


Figure 3: The periodic patterns in the TaxiBJ data. (a) Temporal shifting between days. (b) Temporal shifting between weeks. Each time represents a time interval (e.g., 9:30 am means 9:00-9:30 am).

features is built as follows:

$$\mathbf{X}_t^{Hour} = [\mathbf{X}_{t-h-L}, \mathbf{X}_{t-h-(L-1)}, \cdots, \mathbf{X}_{t-h-1}]$$
$$\mathbf{X}_t^{Day} = [\mathbf{X}_{t-d-\lfloor \frac{L}{2} \rfloor}, \mathbf{X}_{t-d-(\lfloor \frac{L}{2} \rfloor-1)}, \cdots, \mathbf{X}_{t-d+\lfloor \frac{L}{2} \rfloor}]$$
$$\mathbf{X}_t^{Week} = [\mathbf{X}_{t-w-\lfloor \frac{L}{2} \rfloor}, \mathbf{X}_{t-w-(\lfloor \frac{L}{2} \rfloor-1)}, \cdots, \mathbf{X}_{t-w+\lfloor \frac{L}{2} \rfloor}]$$

where L is the length of the sampling sequence. The time interval for all datasets is 30 minutes, which means that h is set to 0 to capture the hourly trend, d is set to 48 to represent the daily periodicity (with 48 half-hour intervals in a day), and w is set to 7×48 to account for the weekly periodicity (with 7 days in a week). The external information vector $\mathbf{V}^E$ is processed through two FC layers and reshaped into a 4D tensor $\mathbf{X}^E$ with the same shape as the feature tensors. $\mathbf{X}^E$ will be concatenated with feature tensors $\mathbf{X}^{Hour}$, $\mathbf{X}^{Day}$, $\mathbf{X}^{Week}$ as $\mathbf{X}^H$, $\mathbf{X}^D$, $\mathbf{X}^W$ and then be fed into the network.

3.2 Pyramid 3D Swin Transformers

To overcome the limitation of existing methods in capturing long-range dependency, we propose a novel approach called the pyramid 3D Swin Transformers network which constructs hierarchical features and fuses features of different levels step by step to capture spatial dependency with varying ranges, especially long-range spatial dependency. Meanwhile, benefiting from spatial-temporal joint modeling, the pyramid 3D Swin Transformers network enables better interaction between the spatial and temporal factors, leading to a better understanding of crowd flow dynamics.

Specifically, a ConvLSTM is employed to model the spatial-temporal features at the bottom of the pyramid architecture in order to preserve high-resolution spatial features well and achieve higher modeling efficiency. We make use of patch merging blocks to achieve downsampling that is, the stride of sampling is equal to 2. Further, the higher-level 3D Swin Transformer blocks are employed to handle hierarchical spatial-temporal features and model long-range spatial dependency. Hierarchically, high-level spatial-temporal features are upsampled and then aggregated with lower-level spatial-temporal features which skip connects to the next 3D Swin Transformer block by a FC layer. The hierarchies generated in the network are able to generate finer details in the output. Based on the above design, the network can jointly capture local and global information in a hierarchical manner. Figure 4(a) presents the architecture of the pyramid 3D Swin Transformers network. Two consecutive 3D Swin Transformer blocks are computed as:

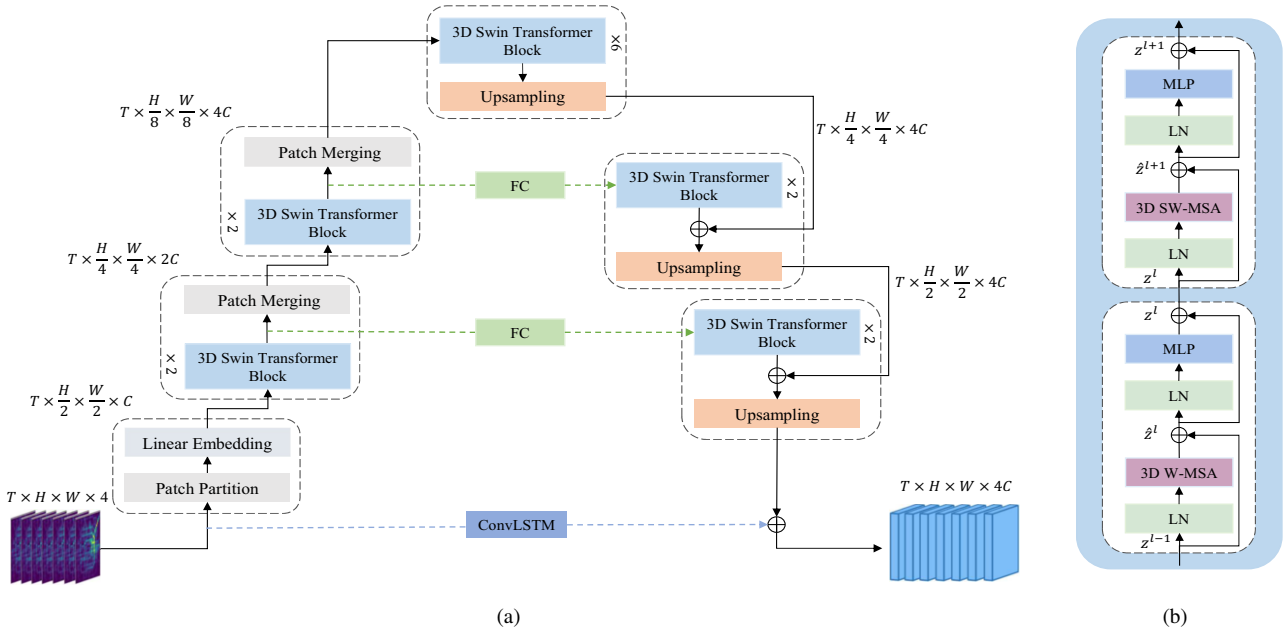


Figure 4: (a) The architecture of Pyramid 3D Swin Transformers. (b) An illustration of two successive 3D Swin Transformer blocks.

$$
\begin{aligned}
\hat{\mathbf{z}}^{l} &= \text{3DW-MSA}(\text{LN}(\mathbf{z}^{l-1})) + \mathbf{z}^{l-1}, \\
\mathbf{z}^{l} &= \text{FFN}(\text{LN}(\hat{\mathbf{z}}^{l})) + \hat{\mathbf{z}}^{l}, \\
\hat{\mathbf{z}}^{l+1} &= \text{3DSW-MSA}(\text{LN}(\mathbf{z}^{l})) + \mathbf{z}^{l}, \\
\mathbf{z}^{l+1} &= \text{FFN}(\text{LN}(\hat{\mathbf{z}}^{l+1})) + \hat{\mathbf{z}}^{l+1},
\end{aligned} \tag{1}
$$

where $\hat{\mathbf{z}}^{l}$ and $\mathbf{z}^{l}$ denote the output features of the 3D(S)W-MSA module and the FFN module for block l, respectively; 3DW-MSA and 3DSW-MSA denote 3D window based multi-head self-attention using regular and shifted window partitioning configurations, respectively. An illustration of two successive 3D Swin Transformer blocks is shown in Figure 4(b). ConvLSTM [20] is an extension of the fully connected LSTM that incorporates convolutional structures. It allows for the iterative calculation of the hidden state h_t from $t = 1$ to T for an input sequence $X = [x_1, x_2, \cdots, x_T]$. The In the pyramid network, the patch partition block accepts a tensor of shape $T \times H \times W \times 4$ and splits it into non-overlapping patches in the dimensions of height and width. The patch partition is followed by a linear embedding to map the split results to a tensor of shape $T \times \frac{H}{2} \times \frac{W}{2} \times C$ which is fed into a 3D Swin Transformer block. The network for each $\mathbf{X}$ in $[\mathbf{X}^H, \mathbf{X}^D, \mathbf{X}^W]$ are the same, so we list the formulas for one $\mathbf{X} = [x_1, x_2, \cdots, x_T]$ as follows:

$$
\begin{aligned}
[x'_1, x'_2, \cdots, x'_T] &= \text{LE}(\text{PP}([x_1, x_2, \cdots, x_T])), \\
[z^1_1, z^1_2, \cdots, z^1_T] &= f^1_{3DST}([x'_1, x'_2, \cdots, x'_T]), \\
[z^2_1, z^2_2, \cdots, z^2_T] &= f^2_{3DST}(\text{PM}([z^1_1, z^1_2, \cdots, z^1_T])), \\
&\vdots \\
[z^l_1, z^l_2, \cdots, z^l_T] &= f^l_{3DST}(\text{PM}([z^{l-1}_1, z^{l-1}_2, \cdots, z^{l-1}_T])),
\end{aligned} \tag{2}
$$

where PP, LE, and PM denote patch partition, linear embedding, and patch merging, respectively. The function $f^l_{3DST}(\cdot)$ refers to the set of operations in Eq.1, l for the l^{th} level, and $\mathbf{z}$ are the output of the 3D Swin Transformer. High-level spatial-temporal features are upsampled and then aggregated with lower-level spatial-temporal features which skip connects to the next 3D Swin Transformer block by a FC layer. After aggregating spatial-temporal features from ConvLSTM, we get the final output. The process is as follows:

$$
\begin{aligned}
[p^l_1, p^l_2, \cdots, p^l_T] &= f^l_{Up}([z^l_1, z^l_2, \cdots, z^l_T]), \\
[q^l_1, q^l_2, \cdots, q^l_T] &= f^l_{3DST}(f_{FC}([z^{l-1}_1, z^{l-1}_2, \cdots, z^{l-1}_T])), \\
[s^l_1, s^l_2, \cdots, s^l_T] &= [p^l_1, p^l_2, \cdots, p^l_T] \oplus [q^l_1, q^l_2, \cdots, q^l_T], \\
&\vdots \\
[p^1_1, p^1_2, \cdots, p^1_T] &= f^1_{Up}([s^2_1, s^2_2, \cdots, s^2_T]), \\
[q^1_1, q^1_2, \cdots, q^1_T] &= f_{CL}([x_1, x_2, \cdots, x_T]), \\
[s_1, s_2, \cdots, s_T] &= [p^1_1, p^1_2, \cdots, p^1_T] \oplus [q^1_1, q^1_2, \cdots, q^1_T],
\end{aligned} \tag{3}
$$

where the function $f_{CL}(\cdot)$ and $f_{FC}(\cdot)$ refer to the ConvLSTM operations and a FC layer, respectively. And $f_{Up}(\cdot)$ denotes an upsampling operation. Here, p_i, q_i, and $s_i (i = 1, \cdots, T)$ are intermediate features and the final output, respectively. In summary, the pyramid 3D Swin Transformers network is abstracted as follows:

$$
[s_1, s_2, \cdots, s_T] = f_{Py3DSTs}[x_1, x_2, \cdots, x_T] \tag{4}
$$

3.3 Attention Mechanism and Fusion Process

Attention mechanism. We employ the attention mechanism to effectively capture the temporal shifting within the periodicity. In our approach, we extend the original attention mechanism to handle a 4D tensor $(Timestep, Height, Width, Filter)$ as input and generate a 3D attention tensor $(Height, Width, Filter)$ as output. The implementation of the attention mechanism is as follows:

$$
\begin{aligned}
z_i &= \tanh(\mathbf{W}_{att} \cdot s_i + \mathbf{b}_{att}), \\
\alpha_i &= \frac{\exp(z_i)}{\sum_j \exp(z_j)}, \\
\mathbf{X}_{att} &= \sum_{i=1}^{T} \alpha_i \cdot s_i,
\end{aligned} \tag{5}
$$

Here, $\mathbf{W}_{att}$ is weight and $\mathbf{b}_{att}$ is bias and s_i is the i-th result in $[s_1, s_2, \cdots, s_T]$ outputted by the network. Note that s_i is a 3D tensor $(Height, Width, Filter = 4C)$, and $\mathbf{W}_{att}$ has $(Height \times Width \times Filter)$ learnable parameters. For each $\mathbf{S}$ in $[\mathbf{S}^H, \mathbf{S}^D, \mathbf{S}^W]$, we utilize an independent attention block to handle. Let $f_{att}(\cdot)$ denote the operation set in Eq.5:

$$\begin{aligned} \mathbf{X}_{att}^H &= f_{att}^H([s_1^H, s_2^H, \cdots, s_T^H]), \\ \mathbf{X}_{att}^D &= f_{att}^D([s_1^D, s_2^D, \cdots, s_T^D]), \\ \mathbf{X}_{att}^W &= f_{att}^W([s_1^W, s_2^W, \cdots, s_T^W]). \end{aligned} \tag{6}$$

Parametric-matrix-based fusion. Rather than relying solely on linear combinations, it is important to consider the existence of more complex interactions within the three branches. To effectively aggregate the spatial-temporal features associated with the three temporal properties, we adopt the following approach:

$$\mathbf{X}_{fusion} = \mathbf{W}_h \odot \mathbf{X}_{att}^H + \mathbf{W}_d \odot \mathbf{X}_{att}^D + \mathbf{W}_w \odot \mathbf{X}_{att}^W \tag{7}$$

where $\odot$ is Hadamard product, $\mathbf{W}_h$, $\mathbf{W}_d$, $\mathbf{W}_w$ are the learnable parameters that adjust the degrees affected by the three temporal properties, respectively. After end fusion, we utilize a convolution layer to get the final prediction result $\widehat{\mathbf{X}}_t$.

3.4 Differentiation from Existing Work

Periodic temporal dependency. Previous works, with the exception of STDN [25], have not adequately addressed the issue of period shift. STDN [25] seeks to rectify periodic shifting based on the daily periodicity of a specific grid (a pixel). In contrast, our method first considers the interdependence of the crowd flow map (all pixels) and models them simultaneously. Second, we revealed that weekly periodicity also exhibits shifting challenges due to climate, events, etc. Therefore, we introduce the weekly period to enrich the temporal dependency to alleviate periodic shifting further.

Long-range spatial dependency. Previous work such as DeepCrowd [8] has attempted to utilize ConvLSTM to capture these dependencies, but its limited receptive field renders it inadequate for capturing and difficult to optimize [16]. Besides, the features far away from a specific location have to pass through a large number of layers before affecting the location for both forward propagation and backward propagation, which would add optimization difficulties during the training [3]. Therefore, ConvLSTM tends to suffer from the limited ability to capture long-range spatial dependency. In contrast, the proposed GridFormer effectively aggregates longer spatial information through the transformer's capability of capturing long-range dependency and concise pyramid-based downsampling.

In conclusion, the proposed GridFormer is carefully optimized for addressing the long-range spatial dependency and periodic shifting in crowd flow prediction, and we accordingly propose a novel pyramid 3D Swin Transformer and periodic shifting sampling. To the best of our knowledge, we are the first to summarize and address these two pivotal issues simultaneously.

4 Experiments

4.1 Experimental Settings

Dataset. Three crowd flow datasets are adopted for our experiments, which were used by existing works [27, 23, 4, 8].

- **TaxiBJ [27, 23, 4]:** The crowd flow dataset used in this study is derived from taxicab GPS data, encompassing the inflow and outflow of taxis across various regions in Beijing. The dataset covers four distinct time periods: 7/1/2013 to 10/30/2013, 3/1/2014 to 6/30/2014, 3/1/2015 to 6/30/2015, and 11/1/2015 to 4/10/2016. We divide Beijing city into a grid system consisting of 32×32 grids. Additionally, we set the time interval for the dataset as 30 minutes, allowing for a granular temporal resolution.
- **BousaiOSA [8]:** The crowd flow dataset used in this study is sourced from GPS trajectory data recorded by a smartphone application utilized by users. The data spans a continuous period from 4/1/2017 to 7/9/2017 and comprises information regarding the inflow and outflow of individuals in various regions of Osaka. Osaka is divided into a grid system consisting of 60×60 grids. The time interval between consecutive data points is set to 30 minutes.
- **BousaiTYO [8]:** This crowd flow dataset is derived from GPS trajectory data recorded by a smartphone application used by users. It covers the same time period as the BousaiOSA dataset. BousaiTYO focuses on collecting information regarding the inflow and outflow of crowds in different regions of Tokyo, which is partitioned into a grid system consisting of 80×80 grids. Similar to BousaiOSA, the time interval for BousaiTYO is set to 30 minutes.

Evaluation Metric. Following previous works [27, 12, 8], we use RMSE and MAE as metrics:

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left\|\mathbf{X}_t - \widehat{\mathbf{X}}_t\right\|_2^2}$$

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|\mathbf{X}_t - \widehat{\mathbf{X}}_t|$$

Where $\mathbf{X}_t$ represents the ground-truth value at the t^{th} time interval, while $\widehat{\mathbf{X}}_t$ corresponds to the prediction. The variable T indicates the total number of samples in the testing data.

Implementation Details. For preprocessing, we employ Min-Max normalization to scale the crowd flow values within the range of [0, 1]. After making predictions, we denormalize the predicted values and utilize them for model evaluation. Following established practices in both industry and academia, we calculate the evaluation metrics for crowd flow values that are larger than 10. This approach is commonly adopted since low-flow instances have minimal significance in real-world applications [26]. Regarding external factors, we utilize one-hot encoding to transform metadata such as HourOfDay, DayOfWeek, and Weekend/Weekday into binary vectors. For the division of data, we allocate 80% for training purposes and the remaining 20% for testing. Within the training set, we further allocate 80% for model learning and the remaining 20% for validation purposes. To generate samples, we utilize a sliding window technique for both the training and testing datasets.

For training, we configure the number of heads in multi-head self-attentions as 8, which allows for the effective capturing of diverse attention patterns. The pyramid 3D Swin Transformers generate spatial-temporal features with a channel number of 192 (i.e., C=48), providing a comprehensive representation of the data. To implement our proposed model, we utilize the PyTorch framework and train it end-to-end using the Adam optimization algorithm [9] with a learning rate of 1×10^{-4}. The training process is carried out over 100 epochs, with a batch size of 4 samples. During training, the model is monitored, and the parameters are saved when the model achieves the lowest root mean squared error (RMSE) on the validation dataset, ensuring optimal performance.

Table 1: Performance comparison with SOTA methods. $^{***}(^{**})$ means the result is significant according to Student's T-test at level 0.01 (0.05) compared to the best baseline. ΔRMSE indicates the reduction of RMSE compared with ST-RestNet.

Method	TaxiBJ			BousaiOSA			BousaiTYO		
	RMSE	MAE	Δ RMSE	RMSE	MAE	Δ RMSE	RMSE	MAE	Δ RMSE
Historical Average	37.01	28.70	89.07%	12.73	11.32	77.38%	15.43	16.04	86.30%
CopyLastFrame	26.38	21.69	27.86%	17.91	15.26	149.51%	23.53	20.27	184.09%
CNN	27.88	19.21	35.12%	12.15	9.71	69.33%	13.07	12.14	57.71%
ConvLSTM	22.42	13.77	8.67%	9.75	7.28	35.80%	10.19	9.84	23.05%
ST-ResNet	20.53	12.89	0.00%	7.18	5.84	0.00%	8.28	6.25	0.00%
DMVST-Net	22.54	14.04	9.22%	8.00	6.37	11.52%	9.01	7.13	8.78%
PCRN	21.87	13.75	8.10%	7.56	6.18	5.41%	8.44	6.46	6.70%
STDN	19.51	12.71	-4.99%	6.96	5.73	-3.07%	8.07	6.18	-2.57%
DeepSTN+	19.63	12.82	-4.40%	6.95	5.70	-3.11%	8.03	6.17	-3.03%
DeepCrowd	19.27	12.97	-6.15%	6.86	5.43	-4.36%	7.95	5.98	-4.01%
LMST3D	19.33	12.48	-5.86%	6.75	5.33	-5.94%	7.83	5.81	-5.88%
GridFormer	**18.02*****	**12.15*****	**-10.94%*****	**6.59*****	**5.18*****	**-8.10%*****	**7.58*****	**5.72****	**-8.47%*****

4.2 Comparisons with State-of-the-Art Methods

We compare our proposed method with 12 baselines, including (1) Historical Average (2) CopyLastFrame, i.e., the crowd flow map corresponding to the previous time interval of the predicting target (3) CNN (4) ConvLSTM [20] (5) ST-ResNet [27] (6) PCRN [30] (7) DMVST-Net [26] (8) DeepSTN+ [12] (9) STDN [25] (10) DeepCrowd [8] (11)LMST3D [4]. All baselines' hyperparameter settings and training procedures are implemented completely in accordance with their original papers or source code. It should be noted that to test the pure ability of grid-based modeling on the spatial-temporal data, we only use metadata for external info, and extra data such as Point-Of-Interest(POI) data and meteorology data are excluded from our model and all the baselines. We run each baseline 10 times and report the mean of the experimental results. Besides, we also conduct student t-test.

Performance Evaluation. Table 1 shows the performance comparison of our GridFormer with other competing methods on TaxiBJ, BousaiOSA, and BousaiTYO datasets. Our GridFormer achieves the lowest RMSE and MAE on three datasets with varying mesh-grid numbers and significantly outperforms other competing methods by a large margin. When compared with the most classic ST-ResNet, GridFormer reduces RMSE by 8.10% to 10.94%. Specifically, Historical Average and CopyLastFrame don't perform well, because they only rely on historical data of predicted value and overlook spatial and context features. Also, our GridFormer outperforms ST-ResNet and DeepSTN+, because they only use CNN to capture spatial dependency, but overlooks the temporal sequential dependency. DMVST-Net and STDN handle spatial-temporal features by local CNN and LSTM. LMST3D models the spatial-temporal correlation across multiple local regions using local 3D CNN. However, all of them are only suitable for datasets with a small mesh-grid number like 10×10 and cannot capture long-range spatial dependency. PCRN and DeepCrowd jointly model the spatial-temporal features based on ConvGRU or ConvLSTM, but they overlook the temporal shifting in the periodicity. The better performance of GridFormer demonstrates the effectiveness of the periodically shifted sampling method combined with the attention mechanism to handle periodic temporal shifting and the pyramid 3D Swin Transformer network to model long-range spatial dependency.

Qualitative evaluation w.r.t. Long-range Spatial Dependency. To better demonstrate the model's effectiveness in capturing long-range spatial dependency, we add standard normal random noise to the outermost grid of the crowd flow map and measure the performance fluctuation at the central grid. If the model can better capture long-range spatial dependency, long-range noise will have a greater impact on performance. Here we compare the proposed algorithm with DeepCrowd, which also considers long-range spatial dependency, and the results are shown in Table 2. We can find that the noise added in the long-range has a greater impact on the performance of GridFormer, which means that GridFormer captures long-range spatial dependency more effectively.

Table 2: Effectiveness verification of long-range capturing.

Model (TaxiBJ)	RMSE (w/o noise)	RMSE((w/ noise))	Δ RMSE
DeepCrowd	21.07	22.61	7.31%
GridFormer	19.92	23.16	16.26%

Efficiency Evaluation. We conducted an analysis to evaluate the efficiency of the main models in terms of model complexity and model training time, as presented in Table 3. The complexity of GridFormer is found to be comparable to that of existing methods, indicating that the significant improvements achieved by GridFormer do not come at the expense of increased complexity.

Table 3: Efficiency evaluation.

Main Model	Training time (Hour)	Params
ST-ResNet	0.81	484,384
DeepSTN+	7.60	33,607,666
DMVST-Net	28.50	1,578,253
DeepCrowd	11.78	5,852,721
STDN	77.35	6,349,922
LMST3D	25.65	9,267,786
GridFormer	13.60	17,348,533

4.3 Ablation Study

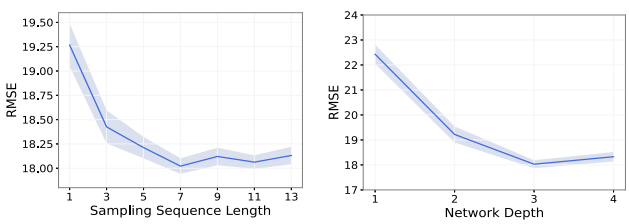
Effectiveness of GridFormer components. We conduct a comparative analysis between our GridFormer model and six different model variants on TaxiBJ dataset to examine the individual components of GridFormer. "Original 3D Swin Transformer" refers to our substitution of the proposed pyramid structure with the conventional 3D Swin Transformer. It is evident that the absence of an effective fusion of multi-level spatial-temporal features through the pyramid structure leads to a significant decline in performance. "NoExternal" denotes

the utilization of GridFormer without external info. The results obtained from this variant highlight the significance of external info in order to achieve enhanced performance. Specifically, the periodically shifted sampling method, attention mechanism, and end-fusion contribute to enhancements of 5.04%, 3.48%, and 2.16%, respectively, compared to "NoShiftedSampling," "NoAttention," and "NoEndFusion." The results serve as evidence of the effectiveness of each individual component within the GridFormer. Furthermore, the experimental results provide confirmation that the integration of ConvLSTM at the lower level of the pyramid 3D Swin Transformers network helps in preserving high-resolution spatial information. This finding adds further validation to the overall efficacy and reliability of each component incorporated in our GridFormer approach.

Table 4: Ablation study on different components of GridFormer.

Variant	RMSE	MAE	Δ RMSE
Original Swin Transformer	19.21	12.69	+7.71%
NoExternal	19.49	12.88	+8.16%
NoShiftedSampling	18.93	12.67	+5.04%
NoConvLSTM	19.13	12.73	+6.16%
NoAttention	18.67	12.47	+3.48%
NoEndFusion	18.41	12.28	+2.16%
GridFormer	**18.02**	**12.15**	**0.00%**

Effectiveness of the sampling sequence length. Fig 5(a) illustrates the effectiveness of the sampling sequence length on TaxiBJ dataset. It can be observed that as the sampling length increases, the RMSE decreases, indicating that longer input sequences generally yield better results due to the increased information about the crowd flow dynamics provided to the model. When the length of the sampling sequence is greater than 7, the RMSE no longer decreases, indicating that the period shift of TaxiBJ data is around 3 time intervals, which is in line with the actual situation.



(a) (b)

Figure 5: Effectiveness analysis. (a) Effectiveness of the sampling sequence length. (b)Effectiveness of the network depth.

Effectiveness of the depth of pyramid 3D Swin Transformers network. Fig 5(b) illustrates the impact of the depth of pyramid 3D Swin Transformers network on TaxiBJ dataset. The network depth corresponds to the height of the pyramid 3D Swin Transformers network. When the network depth is set to 1, only ConvLSTM is utilized to handle the spatial-temporal features. As the network depth increases, the RMSE decreases significantly, indicating that the ability of the pyramid 3D Swin Transformers network to capture long-range dependency greatly enhances the prediction accuracy. However, when the network depth reaches 4, the model's prediction accuracy slightly declines. This is mainly because as the network gets deeper, the magnification of downsampling also increases, which will cause the window of self-attention computation to be smaller and make the interactions between patches less efficient.

5 Related Work

In this section, we will discuss the related works pertaining to crowd flow prediction tasks. Traditional time-series forecasting models such as ARIMA and Kalman filtering have been extensively utilized for crowd flow prediction problems [19, 11, 17, 13]. Some studies further consider spatial relations [7, 5] and utilize external context info (e.g., adding features of venue information, weather condition, and events) [18, 21]. While these studies have demonstrated that considering spatial relations and external information can improve the accuracy of crowd flow prediction, they have encountered challenges in capturing the complex non-linear spatial-temporal dependencies inherent in the data.

Deep learning has achieved great success in computer vision and natural language processing [10] and provides a promising way to capture non-linear spatial-temporal relations. A series of studies were proposed for crowd flow prediction. Deep-ST [28] is the first model to use CNN to capture the spatial relations and ST-ResNet [27] employs residual learning to capture citywide spatial dependency. Further, DeepSTN+ [12] enhances the ST-ResNet by designing a unique ConvPlus block to attempt to capture long-range spatial dependency. These methods don't model the temporal sequence dependency explicitly. On dataset with a small mesh-grid number like TaxiNYC which only has 20×10 grids, DMVST-Net [26] and STDN [25] are designed to use a local CNN to capture spatial dependency among nearby grids and employ LSTM to capture temporal dependency. LMST3D [4] models the spatial-temporal correlation across multiple local regions using local 3D CNN. While these studies explicitly model temporal dependency and spatial dependency, none of them consider long-range spatial dependency. With the emergence of convolutional recurrent networks (CRNs) such as ConvGRU [1] and ConvLSTM [20], PCRN [30], Multitask-DF [29] and DeepCrowd [8] employ the CRNs model as the basic unit to handle spatial and temporal dependency for crowd flow prediction. However, these studies overlook the temporal shifting of periodicity. PRNet [23] applies residual connections on different time segments of the periods to improve the model, but this work does not consider the long-range spatial dependency and the temporal shifting of periodicity.

In summary, we propose a new sampling method and then combine the attention mechanism to deal with the periodic shifting problem and design a pyramid 3D Swin Transformers network to capture long-range spatial dependency.

6 Conclusion

In this paper, we proposed a novel Transformer network for citywide crowd flow prediction. A periodically shifted sampling method and attention mechanism are employed to handle the temporal shifting in the daily and weekly periodicity, and a pyramid 3D Swin Transformers network is designed to capture long-range dependency between regions in a hierarchical manner. Meanwhile, the network jointly models spatial-temporal features to enable better interaction between the spatial and temporal domains. The experiment results on three real-life crowd flow datasets varying mesh-grid numbers show that our proposed GridFormer outperforms the state-of-the-art crowd flow prediction methods.

Acknowledgements

This research is supported by Science and Technology on Underwater Information and Control Laboratory.

References

[1] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville, 'Delving deeper into convolutional networks for learning video representations', in *4th International Conference on Learning Representations (ICLR)*, (2016).

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, and et al., 'End-to-end object detection with transformers', in *European conference on computer vision (ECCV)*, pp. 213–229, (2020).

[3] Kalantidis Chen, Yunpeng and et al., 'Aˆ 2-nets: Double attention networks', *Advances in neural information processing systems(NeurIPS)*, (2018).

[4] Yibi Chen, Xiaofeng Zou, Kenli Li, and et al., 'Multiple local 3d cnns for region-based prediction in smart cities', *Information Sciences*, 476–491, (2021).

[5] Dingxiong Deng, Cyrus Shahabi, Ugur Demiryurek, Linhong Zhu, and et al., 'Latent space model for road networks to predict time-varying traffic', in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1525–1534, (2016).

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, and et al., 'An image is worth 16x16 words: Transformers for image recognition at scale', in *9th International Conference on Learning Representations, (ICLR)*, (2021).

[7] Tsuyoshi Idé and Masashi Sugiyama, 'Trajectory regression on road networks', in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 203–208, (2011).

[8] Renhe Jiang, Zekun Cai, Zhaonan Wang, Chuang Yang, Zipei Fan, and et al., 'Deepcrowd: A deep model for large-scale citywide crowd density and flow prediction', *IEEE Transactions on Knowledge and Data Engineering*, (2021).

[9] Diederik P. Kingma and Jimmy Ba, 'Adam: A method for stochastic optimization', in *3rd International Conference on Learning Representations (ICLR)*, (2014).

[10] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, 'Deep learning', *Nature*, 436–444, (2015).

[11] Xiaolong Li, Gang Pan, Zhaohui Wu, Guande Qi, and et al., 'Prediction of urban human mobility using large-scale taxi traces and its applications', *Frontiers of Computer Science*, 111–121, (2012).

[12] Ziqian Lin, Jie Feng, Ziyang Lu, Yong Li, and Depeng Jin, 'Deepstn+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis', in *Proceedings of the AAAI conference on artificial intelligence*, pp. 1020–1027, (2019).

[13] Marco Lippi, Matteo Bertini, and Paolo Frasconi, 'Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning', *IEEE Transactions on Intelligent Transportation Systems*, 871–882, (2013).

[14] Ze Liu, Yutong Lin, Yue Cao, Han Hu, and et al., 'Swin transformer: Hierarchical vision transformer using shifted windows', in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10012–10022, (2021).

[15] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, and et al., 'Video swin transformer', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3202–3211, (2022).

[16] Wenjie Luo, Yujia Li, and et al., 'Understanding the effective receptive field in deep convolutional neural networks', *Advances in neural information processing systems(NeurIPS)*, (2016).

[17] Moreira-Matias, Luis, Gama, Joao, and et al., 'Predicting taxi–passenger demand using streaming data', *IEEE Transactions on Intelligent Transportation Systems*, 1393–1402, (2013).

[18] Bei Pan, Ugur Demiryurek, and Cyrus Shahabi, 'Utilizing real-world transportation data for accurate traffic prediction', in *IEEE International Conference on Data Mining (ICDM)*, pp. 595–604, (2012).

[19] Shashank Shekhar and Billy M Williams, 'Adaptive seasonal time series models for forecasting short-term traffic flow', *Transportation Research Record*, 116–125, (2007).

[20] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo, 'Convolutional lstm network: A machine learning approach for precipitation nowcasting', in *Advances in neural information processing systems (NeurIPS)*, pp. 802–810, (2015).

[21] Yongxin Tong, Yuqiang Chen, Zimu Zhou, Lei Chen, and et al., 'The simpler the better: a unified approach to predicting original taxi demands based on large-scale online platforms', in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1653–1662, (2017).

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, and et al., 'Attention is all you need', in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, (2017).

[23] Chengxin Wang, Yuxuan Liang, and Gary Tan, 'Periodic residual learning for crowd flow forecasting', in *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, pp. 1–10, (2022).

[24] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, and et al., 'Segformer: Simple and efficient design for semantic segmentation with transformers', *Advances in Neural Information Processing Systems (NeurIPS)*, 12077–12090, (2021).

[25] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng, and Zhenhui Li, 'Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction', in *Proceedings of the AAAI conference on artificial intelligence*, pp. 5668–5675, (2019).

[26] Huaxiu Yao, Fei Wu, Jintao Ke, Xianfeng Tang, and et al., 'Deep multi-view spatial-temporal network for taxi demand prediction', in *Proceedings of the AAAI conference on artificial intelligence*, pp. 2588–2595, (2018).

[27] Junbo Zhang, Yu Zheng, and Dekang Qi, 'Deep spatio-temporal residual networks for citywide crowd flows prediction', in *Proceedings of the AAAI conference on artificial intelligence*, pp. 1655–1661, (2017).

[28] Junbo Zhang, Yu Zheng, Dekang Qi, Ruiyuan Li, and et al., 'Dnn-based prediction model for spatio-temporal data', in *Proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems*, pp. 1–4, (2016).

[29] Junbo Zhang, Yu Zheng, Junkai Sun, and Dekang Qi, 'Flow prediction in spatio-temporal networks based on multitask deep learning', *IEEE Transactions on Knowledge and Data Engineering*, 468–478, (2019).

[30] Ali Zonoozi, Jung-jae Kim, Xiao-Li Li, and Gao Cong, 'Periodic-crn: A convolutional recurrent model for crowd density prediction with recurring periodic patterns.', in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 3732–3738, (2018).