Expert Systems With Applications 220 (2023) 119702

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An improved convolutional network capturing spatial heterogeneity and correlation for crowd flow prediction

Hengyu Zhang, Yuewen Liu *, Yuquan Xu, Min Liu, Ping An

Department of Information Management and Commerce Intelligence, School of Management, Xi'an Jiaotong University, No. 28, West Xianning Road, Xi'an, Shaanxi Province 710049, PR China

ARTICLE INFO

Keywords:
Crowd flow prediction
Convolutional network
Spatial heterogeneity
Spatial correlation
Spatiotemporal data

ABSTRACT

Crowd flow prediction plays an important role in urban management and public safety. However, the existing prediction models still have some shortcomings in capturing spatial heterogeneity and multi-scale spatial correlation. To fulfill the research gaps, this paper proposes an improved convolutional network (SHC-Net). The proposed SHC-Net model improves the existing models by capturing the spatial heterogeneity of the temporal patterns of crowd flow, considering both global and local spatial correlations simultaneously, and combining external factors and spatiotemporal features to consider the heterogeneous impact of external factors on crowd flows. We conduct experiments on two real large-scale datasets, and the results show that our model consistently outperforms the state-of-the-art baselines.

1. Introduction

Crowd flow prediction is helpful for urban management and public safety (Zheng et al., 2014). Accurate prediction of crowd flows can help the city authorities to rationalize the allocation of geographic resources. The neglect of crowd flows may cause economic losses and even serious trample tragedies. For instance, on December 31, 2014, people in Shanghai gathered in the historic bund district to celebrate the New Year which led to a serious stampede (BBC, 2015). In the early hours of April 30, 2021, thousands of people gathered to celebrate the Jewish holiday at Mount Melon in the north of Israel, resulting in a catastrophic trample tragedy (BBC, 2021). If the crowd flows of each region in a city can be predicted in advance, the authorities may take measures to control the crowd flows and prevent accidents.

Due to the importance of crowd flow prediction, plenty of researchers investigate this prediction problem based on kinds of diverse spatiotemporal data, such as AFC (Automatic Fare Collection) data, CDR (Call Detail Record) data, and traffic trajectory data (Atluri et al., 2018; Chai et al., 2018; Hoang et al., 2016; Luo et al., 2020; Wang et al., 2020; Zhao et al., 2019; Zheng, 2015). Early studies use traditional statistical methods to deal with crowd flow prediction problems. For instance, Chen et al. (2011) design a model which combines ARIMA (Autoregressive Integrated Moving Average) with GARCH (Generalized AutoRegressive Conditional Heteroskedasticity) to predict traffic flows. Recent studies apply deep learning models in crowd flow prediction because deep learning models have excellent capability to capture nonlinear features. For example, some studies use RNN (Recurrent Neural Network) to forecast traffic flows (Fu et al., 2016), while some others utilize CNN (Convolutional Neural Network) to extract spatiotemporal features to predict crowd flows (He et al., 2020; Zhang, Yu et al., 2019b). However, the existing models have some shortcomings in the following two perspectives:

Spatial heterogeneity. The spatial heterogeneity reflects that each city region has heterogeneous crowd flow patterns (Guo et al., 2019). On the one hand, the crow flow pattern of each city region is heterogeneous. For example, for the office area and commercial area, the office area will have morning and evening work peaks on weekdays, while for commercial areas such as restaurants, there will be a peak at mealtimes, especially on weekends. The existing method to learn the diverse crowd flow patterns is to use the region's POI (Point-of-Interest, such as commercial areas, parks, and suburbs) information (e.g., Feng et al., 2022; Lin et al., 2019; Shao et al., 2021). However, even regions with the same functional type may have diverse crowd flow patterns due to geographical differences. Take commercial area as an example, crowd flow patterns in central business districts may be different from those in relatively remote commercial districts, and the existing POI method fails

* Corresponding author.
E-mail addresses: zhy1999@stu.xjtu.edu.cn (H. Zhang), liuyuewen@mail.xjtu.edu.cn (Y. Liu), xuyuquan@stu.xjtu.edu.cn (Y. Xu), liumin0@stu.xjtu.edu.cn (M. Liu), anping@stu.xjtu.edu.cn (P. An).

https://doi.org/10.1016/j.eswa.2023.119702
Received 9 October 2022; Received in revised form 21 December 2022; Accepted 10 February 2023
Available online 14 February 2023

to capture these differences. On the other hand, the impact of external factors on crowd flows may also have spatial heterogeneity. For instance, a sudden rain or holidays will influence the crowd flows in scenic spots more than in office areas. In summary, there is still a research gap in the literature to model the spatial heterogeneity of crowd flow.

Multi-scale spatial correlation. The crowd flows of a region could be affected by both the adjacent regions (local spatial correlation) and some spatially non-adjacent regions (global spatial correlation, such as the correlation between two far-apart scenic spots). The existing studies commonly use CNN models, which are effective in modeling local spatial correlations but ineffective in modeling global spatial correlations (Guo et al., 2019; Xu, Wang et al., 2018; Yao et al., 2019; Yao et al., 2018). To simultaneously capture the local and global spatial correlations: (1) some studies stack deep convolution networks to increase the receptive field (e.g., Luo et al., 2020; Zhang et al., 2017), but these models increase the difficulty of training and application in reality due to the high latency (Goyal et al., 2021); (2) some studies utilize a double-branch network with max-pooling layers, which may lead to the loss of crowd flow information (Yuan et al., 2020); and (3) some other studies use GCN (Graph Convolutional Network) models, which cannot stack as deep as CNN models for the over-smoothing problems (Huang et al., 2021; Li et al., 2018; Ye et al., 2020; Zheng et al., 2022). Therefore, the models which simultaneously capture the local and global spatial correlations should also be further improved.

To fulfill the research gaps, we propose an improved convolutional network to capture spatial heterogeneity and correlation (SHC-Net) for crowd flow prediction. To address the spatial heterogeneity problem, we propose a temporal block that considers the various flow patterns of different regions; we also design an external block to consider the various impact of external factors on the crowd flow patterns. To address the multi-scale spatial correlation problem, we propose a spatial block with traditional convolution and dilated convolution to simultaneously capture each region's local and global spatial dependency. We also design a fusion method based on the spatial attention mechanism to learn informative positions from inter-spatial relationship and fuse the spatial, temporal, and external features. We conduct experiments on two real-world datasets to evaluate the performance of our SHC-Net model. The experimental results show that our SHC-Net outperforms the state-of-the-art methods consistently.

In summary, our main contributions are as follows:

(1) Our SHC-Net model improves the previous models by solving both spatial heterogeneity and multi-scale spatial correlation problems. In terms of the spatial heterogeneity problem, our SHC-Net model considers each region's heterogeneous crowd flow pattern and the heterogeneous impact of external factors on crowd flow patterns. In terms of the multi-scale spatial correlation problem, our SHC-Net model simultaneously captures each region's local and global spatial correlation. Compared to the previous models, our SHC-Net model is closer to the reality of crowd flow.
(2) A purely CNN-based convolutional network SHC-Net is proposed. To better capture the spatial heterogeneity, region coding is added to learn differences in temporal flow patterns and external influence between regions. To better learn multi-scale spatial correlation, we propose a parallel spatial block to simultaneously capture both local and global spatial correlations. Besides, we design a novel spatial attention-based fusion method to learn informative positions from inter-spatial relationship and fuse diverse features.
(3) Based on the experiment on two real-world datasets, our SHC-Net model consistently outperforms the state-of-the-art crowd flow prediction models. Our SHC-Net model provides the authorities with a new measure to predict crowd flows, allocate geographic resources, and prevent accidents.

The organization of this paper is as follows. Section 2 reviews the related previous works. Section 3 formulates the crowd flow prediction problem in detail. Section 4 introduces the proposed SHC-Net model, then Section 5 evaluates the model performance compared with several state-of-the-art models. Finally, Section 6 summarizes our work.

2. Related works

We review some previous studies on crowd flow prediction and traffic flow prediction. There are three categories of existing flow prediction models: models based on traditional statistical methods, machine learning, and deep learning, respectively.

Traditional statistical methods predict future flows through past flow data and treat the prediction problem as a linear problem, such as HA (History Average), ARIMA, and ARIMA variants (Van Der Voort et al., 1996; Williams & Hoel, 2003). For example, Dong et al. (2009) adopt ARIMA and train it through diverse training data to predict road traffic flow. Later, some researchers argue that the flow prediction problem is complex and non-linear, so machine learning methods with non-linear kernels appear to solve the flow prediction problem. For instance, Tong et al. (2017) design an XGBoost (Extreme Gradient Boosting) model with feature engineering to predict taxi demands. Castro-Neto et al. (2009) propose an online-SVR method to forecast traffic flow. KNN (K-nearest neighbor) model and BN (Bayesian network) model are also widely used in traffic flow prediction problems (Xie et al., 2020). However, the performance of machine learning models depends on the quality of feature engineering, and large-scale feature engineering is a hard problem for humans.

Recently, with the development of deep learning and neural networks, models based on deep learning show their advantages for flow prediction, which can effectively extract information from complex data and achieve higher prediction accuracy. RNN (Recurrent Neural Network), such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit), is particularly popular in predicting crowd flows (Li et al., 2021). In the work of Xu, Ji, Liu (2018), an LSTM-based approach is proposed to predict the citywide dynamic bike demand for a station-free bike sharing system.

However, these models are weak in capturing the spatial dependency on flow prediction problems. Therefore, several studies utilize CNN to capture the spatial correlation. For instance, Xu, Wang et al. (2018) propose a CNN-based convolutional prediction model, which utilizes cascade multiplicative units (CMU) to learn the sequential correlations between the adjacent frames. Zhang et al. (2017) propose a spatiotemporal model ST-ResNet based on CNN and Resnet to extract spatiotemporal features and forecast crowd inflows and outflows in each region of the city. Zhang, Zheng et al. (2019) predict the flow at both nodes and edges through residual and fully-connected networks. Du et al. (2019) design a novel model DST-ICRL with irregular CNN and LSTM to predict traffic flows. In Li et al. (2021), a deep learning framework combing CNN with LSTM is designed, which captures both the spatiotemporal and geographical information with flow data and POI data. Yao et al. (2019) model the spatiotemporal correlation by combing CNN with LSTM, where they use CNN to learn the spatial features and utilize LSTM to learn the long-term temporal dependency. Also with the architecture of CNN and LSTM, Jin et al. (2018) emphasize temporal dependency and design a model called STRCNs to forecast short-term citywide crowd flows. Yao et al. (2018) propose a multi-view spatiotemporal network, which captures the local correlations between regions through a local CNN and learns the latent semantic information by a region graph embedding. Fiorini et al. (2020) propose a spatiotemporal model which jointly employs 3D convolution and LSTM networks to predict mobility dynamics.

Although CNN can derive some spatial correlations, it depends on Euclidean space, like regular grid-based maps or images. Recently, GCN has been proposed to tackle the limitation of CNN, which solves problems of non-Euclidean space (Kipf & Welling, 2016; Scarselli et al.,

2008). GCN can use a graph to capture the irregular topologic information and learn the spatial correlation of crowd flows. Regarding flow prediction through GCN, Zhao et al. (2019) propose a model, T-GCN, which uses GCN to learn the topologic road network information, and extracts the temporal features through GRU. Specifically, each road is treated as a node, and the connection of roads is the edge, then GCN can learn the features of the target node and its adjacent nodes simultaneously. Zheng et al. (2022) propose a novel deep learning model, named SAGCN-SST, which constructs a k-hop adjacent matrix and uses a self-attention mechanism to learn the various influence of different neighbor nodes on the targeted node. Sun et al. (2022) design a variant of GCN and a comprehensive framework, where the metadata, external data, and flow data can be fused effectively. However, GCN models cannot stack as deep as CNN models for the over-smoothing problems (Li et al., 2018).

More recently, several models utilize a graph construction or applied external data like POI data to improve the prediction accuracy (Jiang et al., 2021). For example, Chai et al. (2018) establish a multi-graph convolution model with distance, interaction, and correlation graphs to predict urban bike flows. Lv et al. (2020) propose an improved T-MGCN model, which not only considers the topological graph but also emphasizes semantic graphs, namely traffic pattern graph and functionality graph, to improve the prediction accuracy of traffic flows in the city. Moreover, some other studies adopt multi-graph GCN methods for traffic speed forecasting (Lee & Rhee, 2022) and metro passenger plow prediction (Yang et al., 2021). Ye et al. (2020) define a novel spatial matrix based on the shortest path algorithm instead of the Laplacian matrix in GCN to learn different kinds of spatial correlations and utilize LSTM to learn the temporal correlation to predict subway passenger flows. Huang et al. (2021) propose a hybrid deep learning method with a TGCN model and a graph fusion module to predict citywide human OD flows. They extract spatiotemporal features through TGCN and employ the graph fusion module to capture the semantic information.

Based on the previous models in the literature, we propose our model SHC-Net by considering spatial heterogeneity and multi-scale spatial correlations of crowd flow prediction.

3. Problem formulation

Definition 1. Region. We use longitude and latitude to divide the city into an I^*J grid map. Each grid represents a region of the city, and a region (i, j) represents the grid in the i^{th} row and j^{th} column of the city.

Definition 2. Crowd Inflows/Outflows (Zhang et al., 2018). Given a collection of trajectories at the t^{th} timestep $\mathbb{P}$, the crowd inflows, outflows are defined respectively as:

$$f_t^{in,i,j} = \sum_{Tr_u \in \mathbb{P}} |\{u | g_{k-1} \notin (i,j) \wedge g_k \in (i,j), k > 1\}|, \tag{1}$$

$$f_t^{out,i,j} = \sum_{Tr_u \in \mathbb{P}} |\{u | g_k \in (i,j) \wedge g_{k+1} \notin (i,j), k \geq 1\}|, \tag{2}$$

where $Tr_u : g_1 \rightarrow g_2 \rightarrow \cdots g_{|Tr_u|}$ is a trajectory of a user u in $\mathbb{P}$, and g_k is the geographic coordinate.

Definition 3. Crowd Density. Given a collection of users' records of the base stations that they communicate with at the t^{th} timestep $\mathbb{Q}$, the crowd density is defined as:

$$f_t^{den,i,j} = \sum_{Re_u \in \mathbb{Q}} |\{u | set(Re_u) \cap B_{i,j} \neq \varnothing\}| \tag{3}$$

where $Re_u : b_1 \rightarrow b_2 \rightarrow \cdots b_{|Re_u|}$ is a record of the base stations a user u communicates with at the t^{th} time interval, $set(Re_u)$ is the collection of base stations in Re_u and $B_{i,j}$ represents the set of base stations in the region (i,j).

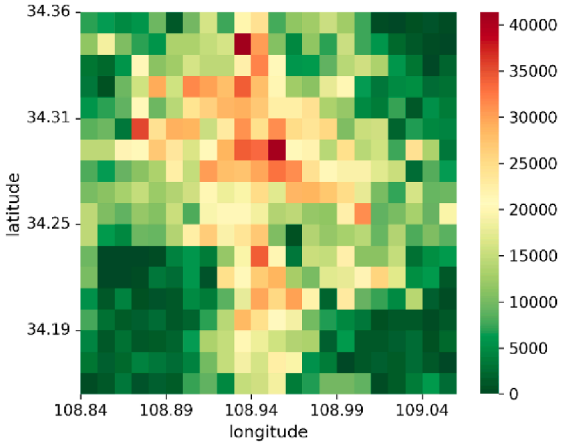

Fig. 1. Crowd flow image.

Then we can get the crowd flow heatmap as shown in Fig. 1.

With the definitions above, we can define the crowd flow prediction problem as follows:

Problem 1. Crowd flow prediction problem. Given a collection of past crowd flow observations $\{F_t | t = 0,1,2 \ldots n\text{-}1\}$, *predict* F_n.

4. Model framework

In this section, we will introduce our SHC-Net model detailedly. Fig. 2 shows the model framework. It contains four blocks to model the temporal correlation, spatial heterogeneity, multi-scale spatial correlation (local and global), and external factors.

(a) In the input block, similar to Zhang et al. (2018), we divide the historical crowd flows into three time slices: closeness, period, and trend. Then we parallelly put the three time slices into the temporal block and the spatial block.
(b) The temporal block is proposed to learn the dynamic temporal correlation of each region. Besides, each region will be encoded differently to be distinguished from each other for better learning the spatial heterogeneity of time series patterns.
(c) The spatial block consists of a local spatial block, a global spatial block, and a spatial attention-based fusion block. The local spatial block aims to learn the local spatial correlation of a region, which includes the nearby regions' influence on the target region. In the global spatial block, we use dilated convolution proposed by Wang et al. (2018) to expand the receptive field range of the convolution kernel to learn the global spatial correlation of large-scale crowd flows. Also, inspired by the spatial attention module in Woo et al. (2018), we design a spatial attention-based fusion method to learn informative positions from inter-spatial relationship and fuse the features.
(d) In the external block, we collect some external factors (i.e., temperature, the day of the week), extract external features, and fuse them with the spatiotemporal features to make a final prediction.

4.1. Input block

The input data is the past crowd flow data $F = \{F_0, F_1, \cdots, F_{n-1}\}$; $F \in \mathbb{R}^{T^*V^*I^*J}$, where T denotes the length of the historical crowd flow data and V represents the number of crowd flow features. Considering three kinds of temporal correlations, we follow the work of Zhang et al. (2017) to divide F into three time fragments: F_C, F_P, F_T, representing closeness, period and trend respectively, as shown in Fig. 3 and Eqs. (4)–(6).

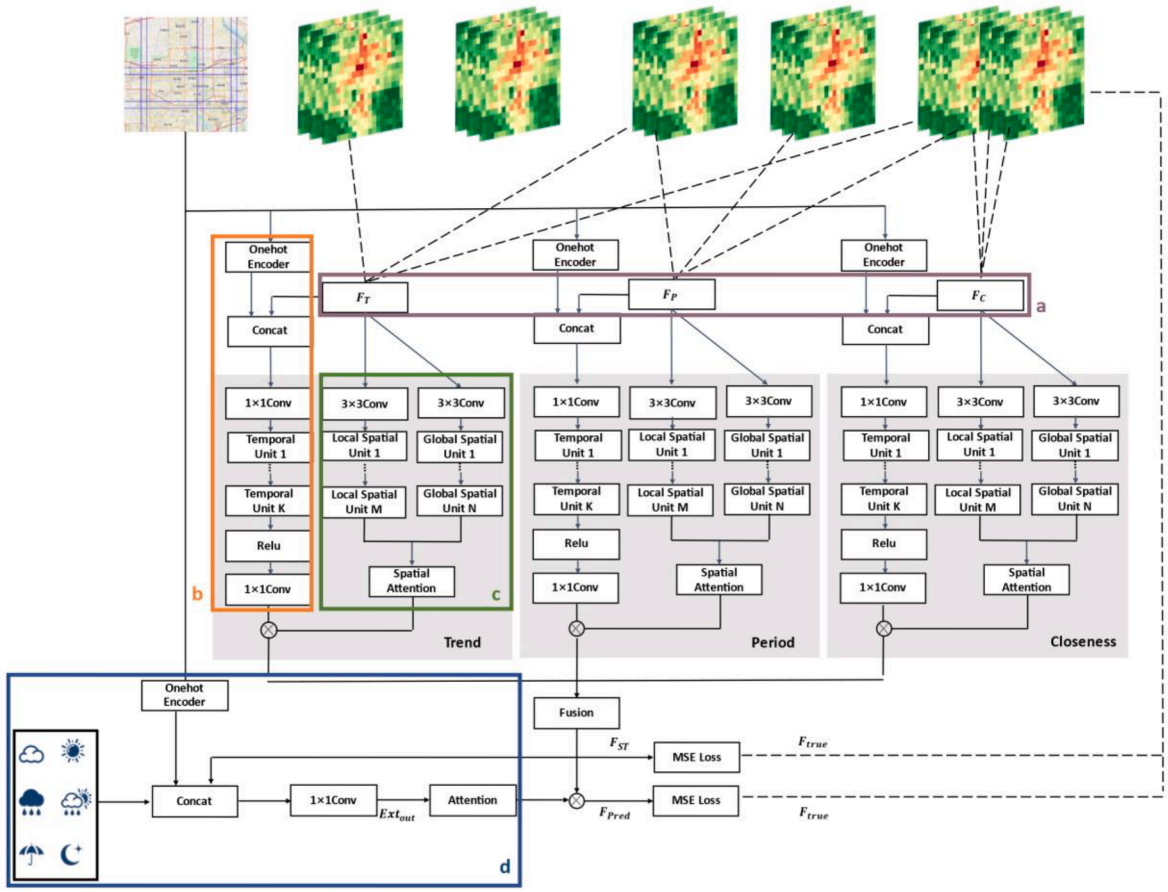

Fig. 2. The model framework. (a) Input block. The past crowd data is divided into three time fragments: closeness, period and trend. (b) Temporal block. The temporal block employs 1 × 1 convolution and residual units, which takes the time fragments from input block and concatenates them with region coding to learn spatial heterogeneity of time series patterns. (c) Spatial block. The spatial block consists of a local spatial block (left), a global spatial block (right), and a spatial attention-based fusion block (below). Local spatial block employs local CNNs and residual units to capture local spatial correlation while global spatial block uses dilated convolutions with residual units to learn global spatial correlation, which both take the time fragments from input block. The spatial attention-based fusion method is used at the end to learn the informative positions, which takes concatenated spatial features from local and global spatial block. Then the spatial attention map is multiplied with the temporal features from temporal block to fuse spatial and temporal features. (d) External block. The external block utilizes 1 × 1 convolution to extract external features, which takes the external data with region coding to learn spatial heterogeneity of external influence. Finally, the spatial attention-based fusion method is used again to fuse external features with spatiotemporal features for prediction.

$$F_C = \left\{F_{n-l_{clo}*c}, F_{n-(l_{clo}-1)*c}, \cdots, F_{n-c}\right\},\ F_C \in \mathbb{R}^{l_{clo}*K*I*J}, \tag{4}$$

$$F_P = \left\{F_{n-l_{per}*p}, F_{n-\left(l_{per}-1\right)*p}, \cdots, F_{n-p}\right\},\ F_P \in \mathbb{R}^{l_{per}*K*I*J}, \tag{5}$$

$$F_T = \left\{F_{n-l_{tre}*t}, F_{n-(l_{tre}-1)*t}, \cdots, F_{n-t}\right\},\ F_T \in \mathbb{R}^{l_{tre}*K*I*J} \tag{6}$$

where c is the closeness span, p is the period, and t is the trend span. The l_{clo}, l_{per}, l_{tre} are the length of closeness, period, and trend dependency sequence, respectively.

Then we put the time fragments F_C, F_P,F_T into the temporal block and spatial block (local and global), respectively.

4.2. Temporal block

With the proposal of TCN (Bai et al., 2018), the traditional convolutional network has been extended to forecast time series and found effective. Inspired by TCN, in the temporal block, we utilize 1 × 1 convolutional kernel to fit the historical crowd flow data and capture the temporal correlation of each region. Compared with using LSTM for time series prediction (Hochreiter & Schmidhuber, 1997), our model is an entirely CNN-based convolutional network, which is convenient for training.

At the same time, considering that deep networks may cause difficulties for training, we use the structure of convolutional units and residual units proposed in Zhang et al. (2017) to improve the network. The temporal block shared by three-time fragments is shown in Fig. 4 and Fig. 5.

For each time fragment F_C, F_P, F_T, take F_C as an example, we first reshape the $F_C \in R^{l_{clo}*V*I*J}$ feature representation into a Vl_{clo}*I*J feature representation. Next, considering that each region has heterogeneous attributes and various flow patterns (spatial heterogeneity of time series patterns), we add the one-hot encoding vector $label \in R^{IJ*I*J}$ for each region as new features and concatenate these features with the time fragment F_C along with the channel axis to get the input feature $F_C^{(0)} \in R^{(l_{clo}V+IJ)*I*J}$, enabling the convolutional network to adjust extracted features according to different location information.

The input feature $F_C^{(0)}$ is operated by a convolution as follows:

$$F_{C,T}^{(1)} = W_{C,T}^{(1)} \odot F_C^{(0)} + b_{C,T}^{(1)}. \tag{7}$$

Then we use K emporal(residual) units as follows:

$$F_{C,T}^{(l+1)} = F_{C,T}^{(l)} + \mathscr{F}\left(F_{C,T}^{(l)}; \theta_{C,T}^{(l)}\right), l = 1, 2, \cdots K, \tag{8}$$

where $\odot$ denotes the convolution operation; $\mathscr{F}$ is the residual function shown in Fig. 5; $W_{C,T}^{(1)}$, $b_{C,T}^{(1)}$, $\theta_{C,T}^{(l)}$ are the learnable parameters.

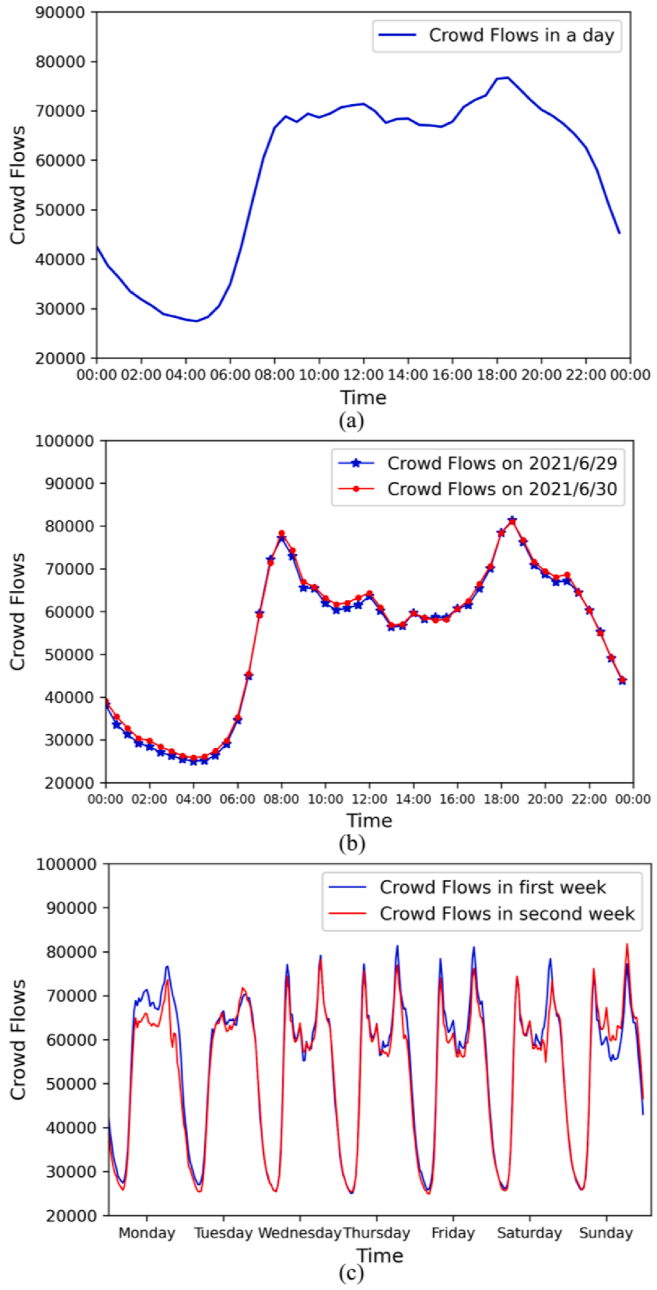

Fig. 3. (a) Closeness. Crowd flows of a region are impacted by those of recent time steps. (b) Period. Crowd flows may be similar on consecutive days. (c) Trend. Crowd flows have a tendency change within one week.

4.3. Spatial block

4.3.1. Local spatial block

CNN has been widely applied in different fields such as Nature Language Processing and Computer Vision, and it can extract effective information from an image (Pouyanfar et al., 2018). The previous work has proved its effectiveness in learning the local spatial correlations (Yao et al., 2019; Yao et al., 2018). Given a grid-based city map, the crowd flow matrix from the map can be seen as an image, and the flow values are treated as pixels.

Therefore, in the local spatial block, following the work in Yao et al. (2018), we use the combination of batch normalization, activation, and convolution twice with residual connection to build a local spatial unit. The whole local spatial block can be seen in Fig. 6 and Fig. 7.

For each time fragment, taking F_C as an example, the local spatial features are obtained by the convolution operations as follows:

$$F_C^{(0)} = F_C, \tag{9}$$

$$F_{C,LS}^{(1)} = W_{C,LS}^{(1)} \odot F_C^{(0)} + b_{C,LS}^{(1)}, \tag{10}$$

$$F_{C,LS}^{(l+1)} = F_{C,LS}^{(l)} + \mathscr{G}\left(F_{C,LS}^{(l)}; \theta_{C,LS}^{(l)}\right), \ l = 1, 2, \cdots, M, \tag{11}$$

where $\odot$ denotes the convolution operation; $\mathscr{G}$ is a combined operation shown in Fig. 7; $W_{C,LS}^{(1)}$, $b_{C,LS}^{(1)}$, $\theta_{C,LS}^{(l)}$ are the learnable parameters.

4.3.2. Global spatial block

In addition to the local flows between adjacent regions, the crowd flows of the city also include distant flows (i.e., crowd flows from a scenic spot to another scenic spot), namely the global spatial dependency. However, it is hard to capture such a global crowd flow pattern only using traditional convolutional layers. Therefore, in the global spatial block, we use dilated convolutions proposed in Wang et al. (2018) instead of traditional convolutional layers to extract global spatial features of crowd flows, for it can effectively expand the receptive field of networks without changing the output size and the number of parameters, as shown in Fig. 8.

The whole structure of the global spatial block is similar to the local spatial block. The difference between these two blocks is that the global spatial unit uses the combination of batch normalization, activation, and convolution three times and employs dilated convolution instead of traditional convolution, as shown in Fig. 9 and Fig. 10.

Specifically, in the global spatial unit, we use 3 dilated convolutional layers, and the dilation rates are 1, 2, and 5, respectively. The operations in the global spatial block are shown as:

$$F_C^{(0)} = F_C, \tag{12}$$

$$F_{C,GS}^{(1)} = W_{C,GS}^{(1)} \odot F_C^{(0)} + b_{C,GS}^{(1)}, \tag{13}$$

$$F_{C,GS}^{(l+1)} = F_{C,GS}^{(l)} + \mathscr{Z}\left(F_{C,GS}^{(l)}; \theta_{C,GS}^{(l)}\right), l = 1, 2 \cdots N, \tag{14}$$

where $\odot$ denotes the convolution operation; $\mathscr{Z}$ is a combined operation shown in Fig. 10; $W_{C,GS}^{(1)}$, $b_{C,GS}^{(1)}$, $\theta_{C,GS}^{(l)}$ are the learnable parameters.

4.3.3. Spatial attention-based fusion block

After obtaining local and global spatial features, we concatenate them. Taking F_C as an example, we concatenate local spatial features $F_{C,LS}$ and global spatial features $F_{C,GS}$ to get the spatial features $F_{C,S}$. Then we further learn informative positions from inter-spatial relationship and use it to modify temporal prediction to fuse spatial and temporal features.

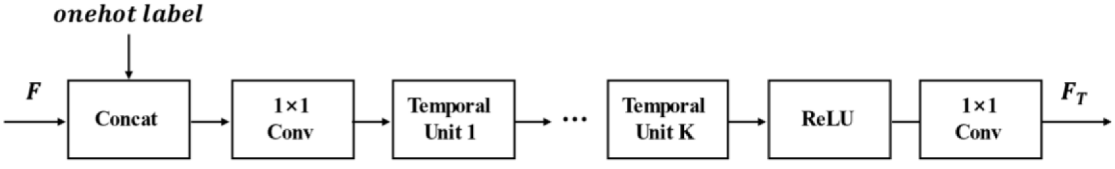

Fig. 4. The structure of the temporal block.

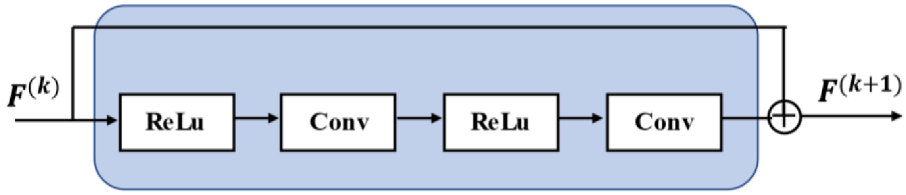

Fig. 5. The details of the temporal unit.

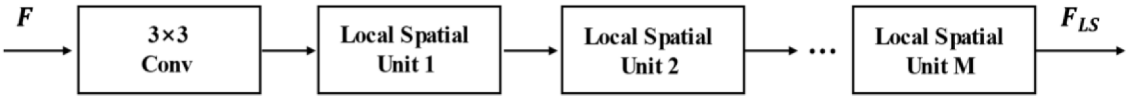

Fig. 6. The structure of the local spatial block.

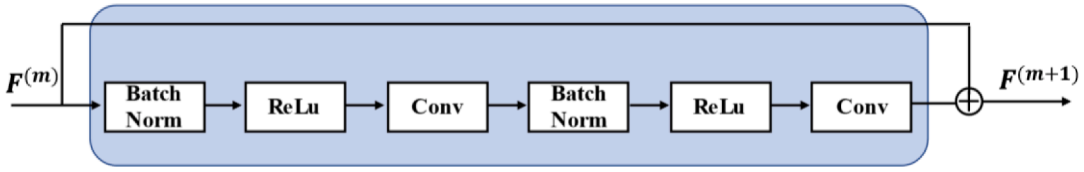

Fig. 7. The details of the local spatial unit.

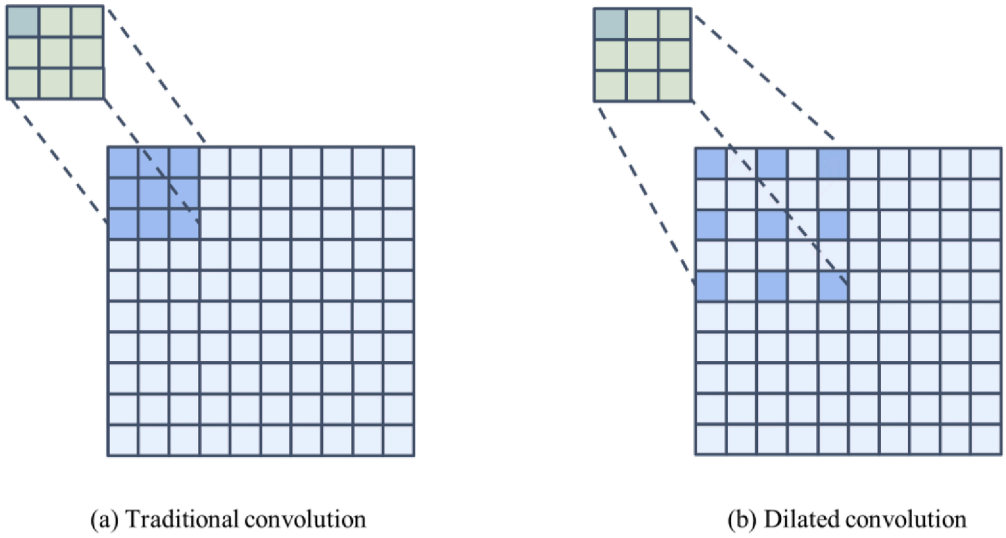

Fig. 8. The difference between traditional convolution (a) and dilated convolution (b).

Fig. 9. The structure of the global spatial block.

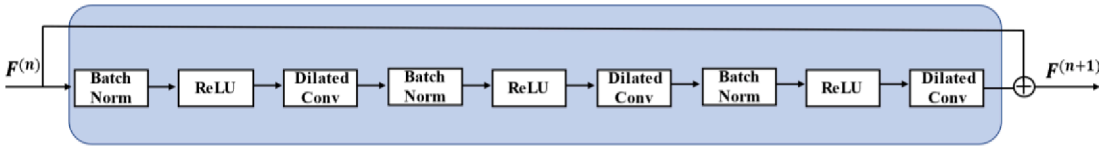

Fig. 10. The details of the global spatial unit.

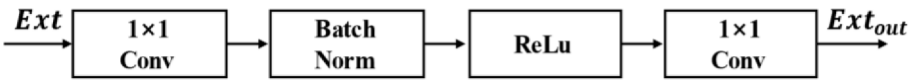

Fig. 11. The structure of the external block.

In detail, inspired by the spatial attention module in Woo et al. (2018), we design a spatial attention-based fusion module to learn informative positions and fuse different features. For the first step, we use $F_{C,S}$ to obtain a spatial attention map. We use the average pooling, max pooling and min pooling layers (Kang et al., 2014) along the channel axis (Komodakis & Zagoruyko, 2017) to process the concatenated spatial features $F_{C,S}$ and get the average-pooled features $F_{C,S}^{avg}$, max-pooled features $F_{C,S}^{max}$ and min-pooled features $F_{C,S}^{min}$. Then we concatenate them and utilize a convolution layer and sigmoid active function. To get a spatial attention feature map with both positive and negative values to indicate the positions to stress or suppress, we subtract 0.5 after the sigmoid active function. The specific calculation of the spatial attention map is as follows:

$$M_S(F_{C,S}) = \sigma\left(f^{7\times7}\left(\left[F_{C,S}^{avg}; F_{C,S}^{max}; F_{C,S}^{min}\right]\right)\right) - 0.5, \tag{15}$$

where σ is the sigmoid active function and $f^{7\times7}$ denotes a convolution operation with the 7×7 kernel size.

With the spatial attention feature map, we multiply it with the output of the temporal block to fuse the spatial features and temporal features and get the spatiotemporal features as follows:

$$F_{C,ST} = F_{C,T} + k_1 * M_S(F_{C,S}) * F_{C,T}, \tag{16}$$

where * is the element-wise product, k_1 is a learnable parameter.

Likewise, we get the $F_{P,ST}$ and $F_{T,ST}$ for period and trend. Next, we merge three kinds of spatiotemporal features as follows:

$$F_{ST} = W_C^{'} * F_{C,ST} + W_P^{'} * F_{P,ST} + W_T^{'} * F_{T,ST}, \tag{17}$$

where $W_C^{'}, W_P^{'}$, and $W_T^{'}$ are learnable parameters.

4.4. External block

Besides the temporal dependency and the spatial dependency of crowd flows, crowd flows are also impacted by some external factors (i.e., the weather temperature and the time of the day), so the external data can help improve the prediction accuracy.

Previous models only use external information alone to extract the external features and describe the impact of external factors on crowd flows, which may weaken the diverse external impact on crowd flows, for the same external conditions may have different effects under different flow conditions. For example, a sudden rain may affect a region with large flows more than that with small flows. Therefore, it is more accurate to extract the external features from the combination of external information and flow information.

Specifically, for discrete data, namely the time of the day and the day of the week, we use one-hot encoding to transform it into a vector $ext_{dis} \in R^D$, where D is the length of the vector. As for the map $I*J$ at the same time, each region on the map shares the same ext_{dis}. We can get a matrix $Ext_{dis} \in R^{D*I*J}$. Similarly, for continuous data temperature and wind power, the max–min normalization method is used to normalize the data between 0 and 1 and get the matrix $Ext_{con} \in R^{C*I*J}$, where C is the number of continuous external factors.

Moreover, considering the spatial heterogeneity of external influence, we also add a one-hot encoding vector $Ext_{place} \in R^{IJ*I*J}$ to distinguish different regions.

Next, we concatenate Ext_{dis}, Ext_{con}, Ext_{place} and the spatiotemporal features $F_{ST} \in R^{V*I*J}$ along the channel axis to get the final external vector $Ext \in R^{(D+C+IJ+V)*I*J}$.

The external vector will be input into the external block to extract the external features, as shown in Fig. 11, and the specific operation is defined in Eq. (18).

$$Ext_{out} = W_{ext}^2 \odot (Relu(Bn(W_{ext}^1 \odot Ext + b_{ext}^1))) + b_{ext}^2, \tag{18}$$

where Bn is batch normalization operation, $Relu$ is an active function, and $W_{ext}^1, b_{ext}^1, W_{ext}^2$, and b_{ext}^2 are learnable parameters.

Then we fuse the external features with the spatiotemporal features. Similar to the fusion of spatial and temporal features, we get the external attention feature map and multiply it with F_{ST} to get the final results. The fusion is calculated as follows:

$$M_S(Ext_{out}) = \sigma(Ext_{out}) - 0.5, \tag{19}$$

$$F_{Pred} = tanh(F_{ST} + k_2 * M_S(Ext_{out}) * F_{ST}), \tag{20}$$

where k_2 is a learnable parameter and $tanh$ is an active function.

4.5. Loss function

The loss of the model consists of two parts. In addition to the prediction loss defined in Eq. (21), an auxiliary loss defined in Eq. (22) is also used.

Since the flow data strongly correlates with the external information, if we only use the end-to-end prediction loss in Eq. (21), the model may focus more on external information and ignore some vital spatiotemporal information, resulting in the underfitting of the model. So we use the auxiliary loss in Eq. (22) to ensure the dominance of spatiotemporal information in the prediction. The final loss is defined in Eq. (23), as shown below:

$$Loss_{pred} = ||F_{Pred} - F_{true}||^2, \tag{21}$$

$$Loss_{aux} = ||F_{ST} - F_{true}||^2, \tag{22}$$

$$Loss = (1-\lambda) * Loss_{pred} + \lambda * Loss_{aux}, \tag{23}$$

where λ is a proportional coefficient that controls the proportion of the two types of losses.

4.6. Training algorithm

Algorithm 1 further introduces the outline of our SHC-Net model. We construct the training data from the historical crowds data, and our SHC-Net is trained through backpropagation and Adam.

Algorithm 1. SHC-Net Training Algorithm

Input: Historical crowd flows/density observations $\{F_n | n = 0,1,2...N\text{-}1\}$; external features $\{ext_n | n = 0,1,2...N\text{-}1\}$; closeness, period, trend c,p,t; length of closeness, period and trend $l_{clo}, l_{per}, l_{tre}$; image size: $I*J$; **Output:** Learned SHC-Net model

D ← ∅
for *all available time intervals* n $(0 \le n \le N-1)$ **do**
  $F_C = \{F_{n-l_{clo}*c}, F_{n-(l_{clo}-1)*c}, \cdots, F_{n-c}\}$
  $F_P = \{F_{n-l_{per}*p}, F_{n-(l_{per}-1)*p}, \cdots, F_{n-p}\}$
  $F_T = \{F_{n-l_{tre}*t}, F_{n-(l_{tre}-1)*t}, \cdots, F_{n-t}\}$
  //F_n is the target at time n
  $(\{F_C, F_P, F_T, ext_n\}, F_n) \rightarrow$ D
initialize all the learnable parameters θ
repeat
  randomly select a batch of instances $\mathscr{D}_b$ from D
  find θ by minimizing the object (23) with $\mathscr{D}_b$
until *stopping criteria* is met

Table 1
Dataset summary.

Dataset	BikeNYC	XACDR
Location	New York	Xi'an
Feature Type	crowd inflows/outflows	crowd density
Time Span	4/1/2014–9/30/2014	6/26/2021–10/29/2021
Timestep	1 h	30 min
Image Size	(16,8)	(22,18)
Available Timesteps	4392	6048
Range	[0,267]	[0,138835]
Temperature/°C	[0,33]	[6,39]
Wind Power/grade	/	1–4

Table 2
Environment configuration.

Item	Parameter
Operating system	Ubuntu 20.04.4 LTS
Memory	512G
CPU	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30 GHz
GPU	NVIDIA Tesla A100
Language	Python 3.9

5. Performance evaluation

5.1. Datasets

We use two real-world datasets from New York and Xi'an, as described in Table 1. The details of these two datasets are the following:

- BikeNYC: It contains the trajectory data from April 1, 2014 to September 30, 2014. Each trajectory record includes: start time and end time, starting ID and ending ID, and the trip duration.
- XACDR: It consists of the anonymized Call Detail Records in Xi'an, Shaanxi, which contains millions of users' mobile phone location records in Xi'an from June 26, 2021 to October 29, 2021. The specific record includes SIM_Card_ID, Base_ID, Base location based on longitude and latitude, and the timestamp.

5.2. Settings

5.2.1. Evaluation metrics

We use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to evaluate our model, which are both common performance metrics and computed as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i}(F_i - \widehat{F}_i)^2}, \tag{24}$$

Table 4
Difference between baselines and our model SHC-Net.

Model		Temporal	Local spatial	Global spatial	Spatial heterogeneity
Traditional	HA	✓			
	ARIMA	✓			
LSTM	GRU	✓			
CNN+ LSTM	ConvLSTM	✓	✓		
	PCRN	✓	✓		
	DMVST-Net	✓✓	✓		
	STDN	✓	✓		
	3D-CLoST	✓	✓		
CNN	CNN		✓		
	ST-ResNet	✓	✓		
	ST-3DNet	✓	✓		✓
	DeepSTN+	✓	✓	✓	
	SHC-Net	✓	✓	✓	✓
GCN	MVGCN	✓	✓	✓	
VT	MSP-STTN	✓	✓	✓	

$$MAE = \frac{1}{n}\sum_{i}(|F_i - \widehat{F}_i|), \tag{25}$$

where F_i and $\widehat{F}_i$ are the ground truth of the crowd flows and the corresponding prediction values, respectively; n is the total number of the samples.

5.2.2. Preprocessing

For each dataset, we first fill in the missing values. Then we normalize the flow data to [−1,1] through the max–min normalization method. As for the external data, we also utilize the max–min normalization method to normalize the continuous data (i.e., temperature, wind power) into the range [0,1] and process the discrete data (i.e., the day of the week, the time of the day) by one-hot coding. The crowd flow prediction will be renormalized back to the normal values for comparison.

5.2.3. Hyperparameters

The model we proposed is implemented by Pytorch and detailed server configuration is shown in Table 2. For BikeNYC, we segment the city into a 16 * 8 grid-based map, and the time interval is 1 h. As for CDR data in Xi'an, considering the massive data in the whole city, without loss of generality, we select partial data in the main urban area through latitude and longitude screening. Then we partition the area into a 22 * 18 grid-based map, and the time interval is 30 min.

On BikeNYC, we follow Zhang et al. (2017) to choose the data of the

Table 3
Detailed hyperparameters of SHC-Net on BikeNYC.

Module	Layer	Type	Filer	Dilation rate	Channel in	Channel out	Activation
Temporal block	T_in	Conv	1 × 1	/	6	64	/
	TU1	Conv、Res	1 × 1	/	64	64	ReLu
	TU2	Conv、Res	1 × 1	/	64	64	ReLu
	TU3	Conv、Res	1 × 1	/	64	64	ReLu
	TU4	Conv、Res	1 × 1	/	64	64	ReLu
	T_out	Conv	1 × 1	/	64	2	/
Local spatial block	LS_in	Conv	3 × 3	/	2	32	/
	LSU1	Conv、Res	3 × 3	/	32	32	ReLu
Global spatial block	GS_in	Conv	3 × 3	/	2	32	/
	GSU1	Conv、Res	3 × 3	{1,2,5}	32	32	ReLu
Spatial attention-based fusion block	/	Conv	7 × 7	/	64	2	Sigmoid
External block	/	Conv	1 × 1	/	162	2	Sigmoid
Output	/	/	/	/	/	/	Tanh

last ten days for testing and the rest for training. On XACDR, we use the data of the last two weeks (about 10%) for testing and the left for training. The batch size is 32, the learning rate is set in {0.0001, 0.0002, 0.0003, 0.0005, 0.001} and the optimizer is the Adam. The closeness c is a timestep, period p is a day, and the trend t is a week. We set l_{clo} in {3,4,5,6}; l_{per} in {1,2,3,4}; l_{tre} in {1,2,3,4}. More details of the layers in our model SHC-Net are shown in Table 3.

5.3. Baselines

Our model SHC-Net is compared with several well-known methods. These state-of-the-art methods can be divided into three categories, which consider (1) temporal, (2) spatial, and (3) spatiotemporal correlations, respectively, and the detailed difference is shown in Table 4. The specific baselines are as follows:

- **HA**: HA model utilizes the average of past crowd flows during the same periods to predict the future crowd flows.
- **ARIMA**: It is a widely-used model for time series forecasting.
- **GRU**: GRU is the most commonly used time series prediction model based on deep learning, which can effectively learn the temporal dependency. Here, we stack 3 GRU layers to predict future crowd flows.
- **CNN**: CNN can effectively learn spatial correlation. We use 5 convolution layers with 3×3 kernel size to predict the future crowd flows.
- **ConvLSTM** (Shi et al., 2015): It is a variant of the LSTM model mainly used for predicting time series data. Compared with LSTM, ConvLSTM can take images as the input of the network and extract features for time series prediction. Here, we stack 3 ConvLSTM layers to predict the future crowd flows.
- **ST-ResNet** (Zhang et al., 2017): It is a spatiotemporal crowd flow prediction model, which captures both temporal and spatial features. It stacks residual units to capture spatiotemporal features from past crowd flow data.
- **PCRN** (Zonoozi et al., 2018): It is a novel model for multi-step crowd density prediction, which expands the CRN (convolutional recurrent network) to learn the spatiotemporal dependency.
- **DMVST-Net** (Yao et al., 2018): It is a novel multi-view spatiotemporal model with temporal view, spatial view and semantic view to predict taxi demands.
- **STDN** (Yao et al., 2019): It is a deep learning based traffic flow prediction model with local CNN and LSTM to learn spatial and temporal dependency respectively, which designs a novel periodically shifted attention mechanism to capture diverse long-term temporal correlations.
- **ST-3DNet** (Guo et al., 2019): It is an end-to-end deep learning model for traffic data forecasting, which automatically employs 3D convolutions to learn spatial and temporal correlations of traffic data.
- **DeepSTN+** (Lin et al., 2019): It is a context-aware model for citywide crowd flow prediction, which utilizes the ConvPlus structure to learn the global spatial correlations of crowd flows.
- **3D-CLoST** (Fiorini et al., 2020): It is a CNN-LSTM based spatiotemporal flow prediction model. It jointly combines 3D convolution and LSTM networks to predict mobility dynamics.
- **MVGCN** (Sun et al., 2022): It is a GCN-based citywide crowd flow prediction model with multi-view graphs for irregular regions, which designs a multi-view fusion module to fuse diverse features from temporal view, spatial view and external view.
- **MSP-STTN** (Xie et al., 2022): It is a novel Transformer-based prediction model for short and long-term crowd flow prediction, which can capture cross-space–time correlation through the multisize patched Transformer.

Table 5
Comparisons with different baselines.

Models	BikeNYC		XACDR	
	RMSE	MAE	RMSE	MAE
HA	12.39	6.76	4454.73	2860.18
ARIMA	11.77	6.49	2055.34	1191.25
GRU	9.03	5.33	1458.44	908.79
CNN	6.80	4.51	1108.63	766.59
ConvLSTM	6.02	3.74	897.53	601.30
ST-ResNet	6.30	3.85	1156.07	785.94
PCRN	6.34	3.76	908.09	639.03
DMVST-Net	5.79	3.78	868.22	594.78
STDN	5.79	3.70	1531.49	982.73
ST-3DNet	5.74	3.53	835.50	570.89
DeepSTN+	5.71	3.54	826.51	577.65
3D- CLoST	6.13	3.69	1697.65	1115.99
MVGCN	5.90	3.54	1237.27	833.60
MSP-STTN	5.70	3.48	1117.57	729.84
SHC-Net	**5.68**	**3.41**	**772.71**	**511.46**

5.4. Performance comparison

From Table 5, we can see that our SHC-Net model performs best with the lowest RMSE and MAE on both datasets among all baselines. To be more specific, traditional temporal models HA and ARIMA perform worst among all models, for these two traditional models treat the change of crowd flows as a simple linear relationship, and they cannot capture the complex spatiotemporal dependency of crowd flows. GRU works better than HA and ARIMA due to its ability to extract non-linear temporal features. As a spatial model, CNN can effectively extract spatial features and has obtained the RMSE of 6.80 and 1108.63 on BikeNYC and XACDR, respectively. Spatiotemporal models (including ConvLSTM, ST-ResNet, PRCN, DMVST-Net, STDN, ST-3DNet, DeepSTN+, 3D-CLoST, MVGCN and MSP-STTN) mostly perform better on both datasets, for they consider both spatial and temporal correlations of crowd flows. For example, ConvLSTM and ST-3DNet get 897.53 and 835.50 RMSE on XACDR, respectively.

Note that the performance of the spatiotemporal model 3D-CLoST on XACDR is not good. Compared to ST-3DNet, which also employs 3D convolutions, 3D-CLoST uses a single 3D convolution-based framework instead of multiple branches to model the different temporal correlations. However, on XACDR, the different temporal correlations of the data are more obvious. Thus simply using a branch to model three kinds of temporal correlations may not be effective for the network in extracting useful features. At the same time, we can also find that the performance of ST-ResNet on XACDR is not very good, and we consider that this may be because ST-ResNet uses a spatiotemporally-coupled network to extract temporal and spatial information. A spatiotemporally-coupled network may lead to the underfitting or overfitting of learning space (time) information when appropriately learning time (space) information. Our model can effectively avoid this problem by separately modeling time and space. As for the performance of MST-STTN, one of possible reasons is that the use of complex vision transformer may lead to difficulty in training and overfitting of the model due to the regular change tendency on XACDR.

Compared with all baselines, taking the RMSE of the crowd density of XACDR as an example, our model SHC-Net is approximately 83% better than HA, 62% better than ARIMA, 47% better than GRU, 30% better than CNN, 14% better than ConvLSTM, 33% better than ST-ResNet, 15% better than PRCN, 11% better than DMVST-Net, 50% better than STDN, 8% better than ST-3DNet, 7% better than DeepSTN+, 54% better than 3D-CLoST, 38% better than MVGCN, and 31% better than MSP-STTN.

We also visualize the real and predicted crowd flow images at a random timestamp on both BikeNYC and XACDR, as shown in Fig. 12 and Fig. 13. From the two figures, we can see that the predicted crowd

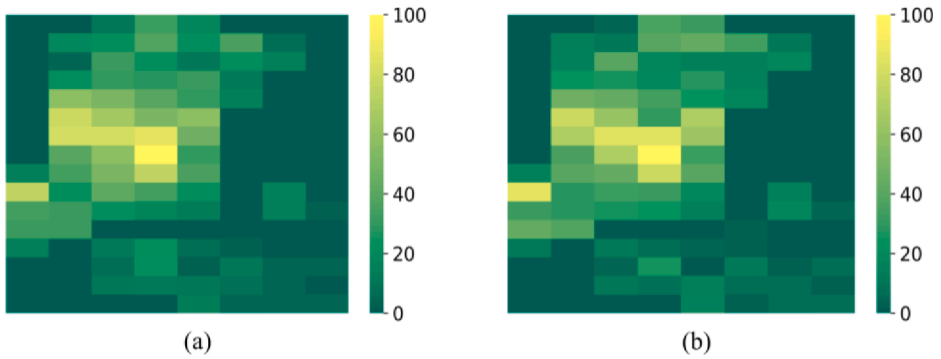

Fig. 12. Actual (a) and predicted (b) crowd inflow image at a random timestamp on BikeNYC.

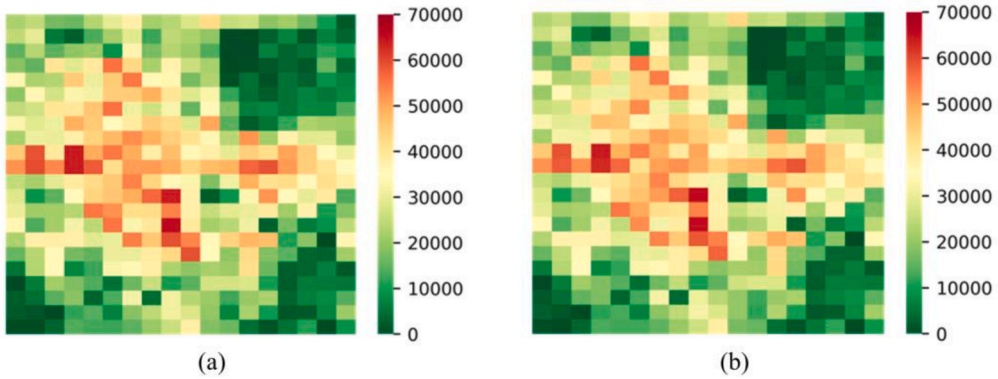

Fig. 13. Actual (a) and predicted (b) crowd density image at a random timestamp on XACDR.

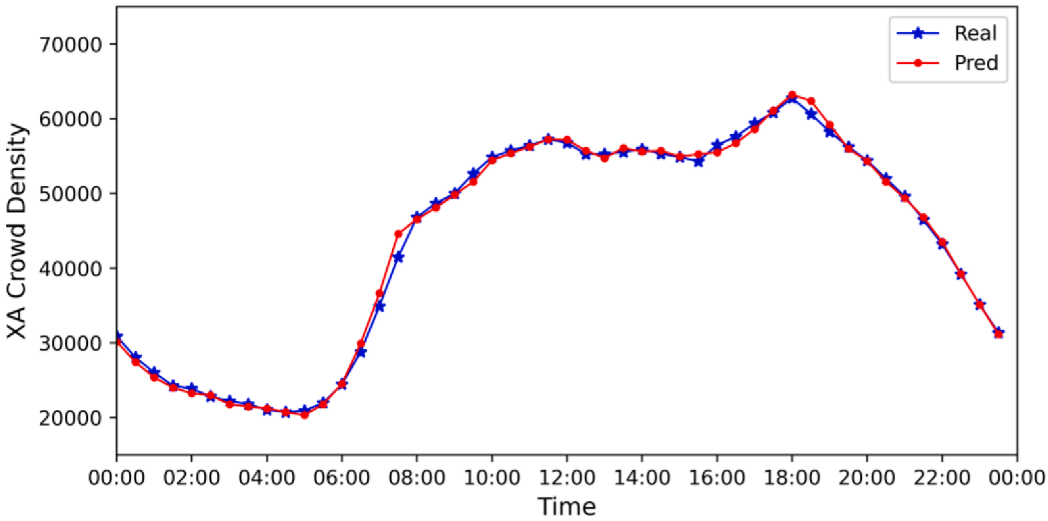

Fig. 14. Actual and predicted crowd density of a random region for one test day on XACDR.

flow images are pretty similar to the actual crowd flow images, further verifying the effectiveness of our proposed model SHC-Net.

Besides, we randomly select a region to visualize the crowd density for one day on XACDR. Fig. 14 shows the results of actual crowd density and predicted crowd density in *region*$(5,7)$ of Xi'an. From the figure, we can find that our model can accurately predict crowd density peaks and valleys, indicating our model not only performs best among all baselines but also gets an accurate crowd flow prediction.

5.5. Ablation experiments

5.5.1. Effects of different blocks

In addition, we also investigate the effects of different blocks in our model and propose seven variants of SHC-Net as follows:

- **SHC-Net-T-nc**: It is a variant of SHC-Net, where we only use the temporal block **without region coding** to predict the crowd flows.
- **SHC-Net-T**: It is a variant of SHC-Net, where we only use the temporal block (with region coding) to predict the crowd flows.

Table 6
Effects of different blocks.

Models	BikeNYC		XACDR	
	RMSE	MAE	RMSE	MAE
SHC-Net-T-nc	6.88	4.04	1152.44	710.97
SHC-Net-T	6.59	3.95	1041.45	668.99
SHC-Net-TL	5.84	3.54	809.41	529.16
SHC-Net-TG	5.83	3.55	810.12	534.45
SHC-Net-TLG-na	5.89	3.63	805.00	531.66
SHC-Net-TLG	5.80	3.53	797.88	527.21
SHC-Net-TLGE-nc	5.76	3.49	790.73	520.94
SHC-Net	**5.68**	**3.41**	**772.71**	**511.46**

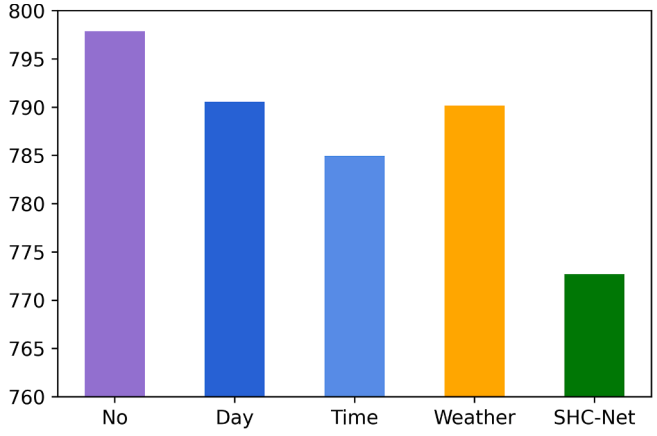

Fig. 15. Effects of different external factors.

- **SHC-Net-TL**: It is a variant of SHC-Net, where we only use temporal block and local spatial block to predict the crowd flows.
- **SHC-Net-TG**: It is a variant of SHC-Net, where we only use temporal block and global spatial block to predict the crowd flows.
- **SHC-Net-TLG-na**: It is a variant of SHC-Net, where we use temporal block, local and global spatial block to predict the crowd flows, but utilize the 1×1 convolution layer to replace the spatial attention-based fusion.
- **SHC-Net-TLG**: It is a variant of SHC-Net, where we only exclude the external block.
- **SHC-Net-TLGE-nc**: It is a variant of SHC-Net, where we use all blocks but the external block is **without region coding** to predict the crowd flows.

Table 6 shows the results of SHC-Net and its variants. Specifically, from the temporal view, we can see that SHC-Net-T performs better than SHC-Net-T-nc, which verifies the spatial heterogeneity of time series patterns and the effectiveness of encoding. From the spatial view, SHC-Net-TL, and SHC-Net-TG both have a good performance on crowd flow prediction, showing the importance of spatial information. Moreover, SHC-Net-TLG outperforms the two models with a single kind of spatial information (local or global), which proves that the spatial impact on crowd flows is multi-scale and our model can effectively capture multi-scale spatial correlations. Also, SHC-Net-TLG has higher prediction accuracy than SHC-Net-TLG-na, indicating the effectiveness of our designed fusion method. Besides, SHC-Net performs better than SHC-Net-TLGE-nc, proving the spatial heterogeneity of external influence on crowd flows and the effectiveness of the region coding again. In general, SHC-Net consistently outperforms SHC-Net-T-nc, SHC-Net-T, SHC-Net-TL, SHC-Net-TG, SHC-Net-TLG-na, SHC-Net-TLG, and SHC-Net-TLGE-nc, which not only denotes the effectiveness of the fusion of each block in our model, but also verifies that our model can effectively capture both the spatial heterogeneity and multi-scale spatial correlations of the crowd flow prediction problem.

5.5.2. *Effects of diverse external factors*

For external influence, we consider three kinds of external factors: the day of the week, the time of the day, and weather (temperature and wind power), respectively. We investigate the effects of different external factors on crowd flow prediction. As shown in Fig. 15, the X-axis represents models with different external factors, and Y-axis represents the evaluation metric. It can be observed that the model without any external factor has the worst prediction accuracy, which proves the vital role of external information in crowd flow prediction. Besides, we can see the model with the "Time (of the day)" information gets the best performance among three kinds of external factors, showing the crowd flows patterns are closely related to the time of the day. On the whole, we can find that our model SHC-Net has the lowest RMSE among all models, indicating the effectiveness of the fusion of different external factors.

5.5.3. *Impact of model hyperparameters*

It can be seen from Fig. 16 that at the beginning, RMSE decreases for the stack of temporal, local spatial, or global spatial units can better extract the crowd flows features. However, as the number of temporal, local spatial, or global spatial units increases, the loss value gradually increases, which indicates that a deeper network leads to a more difficult training process and more complexity.

6. Conclusions

In this paper, we propose an improved convolutional network capturing both spatial heterogeneity and correlation (SHC-Net) for predicting crowd flows. Considering the spatial heterogeneity of crowd flow patterns, we design the temporal block and external block with region coding to capture the diverse flow patterns. Meanwhile, in the spatial block, we simultaneously model the local and global spatial dependency of crowd flows by employing traditional convolution and dilated convolution correspondingly. Besides, we also design a spatial

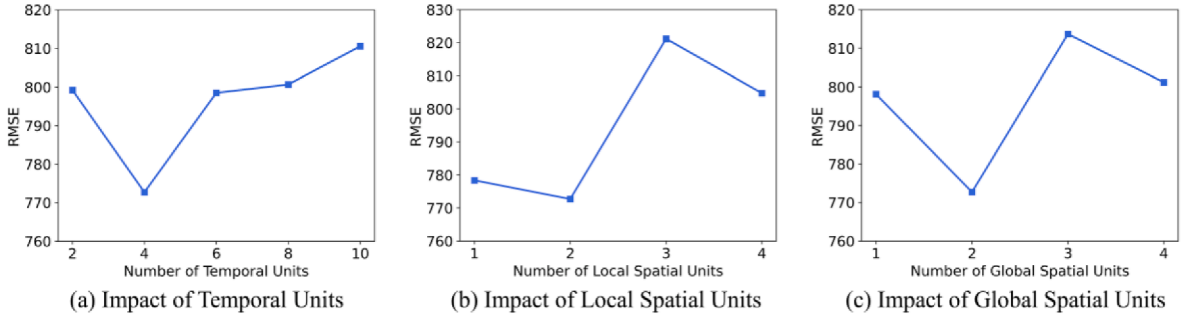

Fig. 16. Impact of model hyperparameters.

attention-based fusion method to make our model perform better. The evaluation of two real-world datasets indicates that our proposed model SHC-Net outperforms all the baselines.

Crowd flows could change suddenly when influenced by abnormal events (i.e., the celebration activities). In the future, our model could also be extended to consider the correlation between abnormal events and crowd flows.

CRediT authorship contribution statement

Hengyu Zhang: Conceptualization, Methodology, Software, Writing – original draft, Visualization. **Yuewen Liu:** Conceptualization, Methodology, Writing – review & editing, Supervision. **Yuquan Xu:** Conceptualization, Writing – original draft, Writing – review & editing. **Min Liu:** Data curation, Resources, Methodology, Validation. **Ping An:** Data curation, Resources, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study is supported by the National Natural Science Foundation of China (Project No. 71871179, 62141223, and 62041206), the National Key Research and Development Program of China (Project No. 2022YFC3320800), and the Yunnan Key Research and Development Program (Project No. 202203ZC100001).

References

Atluri, G., Karpatne, A., & Kumar, V. (2018). Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys (CSUR), 51*(4), 1–41.

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv: 1803.01271*.

BBC. (2015). Shanghai new year crush kills 36. Retrieved from https://www.bbc.com/news/world-asia-china-30646918.

BBC. (2021). Israel crush: Dozens killed at Lag B'Omer religious festival. Retrieved from https://www.bbc.com/news/world-middle-east-56938657.

Castro-Neto, M., Jeong, Y.-S., Jeong, M.-K., & Han, L. D. (2009). Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert Systems with Applications, 36*(3), 6164–6173.

Chai, D., Wang, L., & Yang, Q. (2018). Bike flow prediction with multi-graph convolutional networks. In *The proceedings of the 26th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 397–400).

Chen, C., Hu, J., Meng, Q., & Zhang, Y. (2011). Short-time traffic flow prediction with ARIMA-GARCH model. In *The 2011 IEEE intelligent vehicles symposium (IV)* (pp. 607–612).

Dong, H., Jia, L., Sun, X., Li, C., & Qin, Y. (2009). Road traffic flow prediction with a time-oriented ARIMA model. In *The 2009 fifth international joint conference on INC, IMS and IDC* (pp. 1649–1652).

Du, B., Peng, H., Wang, S., Bhuiyan, M. Z. A., Wang, L., Gong, Q., Liu, L., & Li, J. (2019). Deep irregular convolutional residual LSTM for urban traffic passenger flows prediction. *IEEE Transactions on Intelligent Transportation Systems, 21*(3), 972–985.

Feng, Y., Zhu, X., Tang, X., & Hu, Z. (2022). Multi-step point-of-interest-level crowd flow prediction based on meta learning. In *The 2022 14th international conference on machine learning and computing (ICMLC)* (pp. 229–236).

Fiorini, S., Pilotti, G., Ciavotta, M., & Maurino, A. (2020). 3d-clost: A CNN-LSTM approach for mobility dynamics prediction in smart cities. In *The 2020 IEEE international conference on big data (Big Data)* (pp. 3180–3189).

Fu, R., Zhang, Z., & Li, L. (2016). Using LSTM and GRU neural network methods for traffic flow prediction. In *The 2016 31st youth academic annual conference of Chinese association of automation (YAC)* (pp. 324–328).

Goyal, A., Bochkovskiy, A., Deng, J., & Koltun, V. (2021). Non-deep networks. *arXiv preprint arXiv:2110.07641*.

Guo, S., Lin, Y., Li, S., Chen, Z., & Wan, H. (2019). Deep spatial–temporal 3D convolutional neural networks for traffic data forecasting. *IEEE Transactions on Intelligent Transportation Systems, 20*(10), 3913–3926.

He, Z., Chow, C.-Y., & Zhang, J.-D. (2020). STNN: A spatio-temporal neural network for traffic predictions. *IEEE Transactions on Intelligent Transportation Systems, 22*(12), 7642–7651.

Hoang, M. X., Zheng, Y., & Singh, A. K. (2016). FCCF: Forecasting citywide crowd flows based on big data. In *The proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 1–10).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Huang, Q., Yang, Y., Xu, Y., Wang, E., & Zhu, K. (2021). Human origin-destination flow prediction based on large scale mobile signal data. *Wireless Communications & Mobile Computing (Online), 2021*.

Jiang, R., Yin, D., Wang, Z., Wang, Y., Deng, J., Liu, H., Cai, Z., Deng, J., Song, X., & Shibasaki, R. (2021). Dl-traff: Survey and benchmark of deep learning models for urban traffic prediction. In *The proceedings of the 30th ACM international conference on information & knowledge management* (pp. 4515–4525).

Jin, W., Lin, Y., Wu, Z., & Wan, H. (2018). Spatio-temporal recurrent convolutional networks for citywide short-term crowd flows prediction. In *The proceedings of the 2nd international conference on compute and data analysis* (pp. 28–35).

Kang, L., Ye, P., Li, Y., & Doermann, D. (2014). Convolutional neural networks for no-reference image quality assessment. In *The Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1733–1740).

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Komodakis, N., & Zagoruyko, S. (2017). Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *ICLR*.

Lee, K., & Rhee, W. (2022). DDP-GCN: Multi-graph convolutional network for spatiotemporal traffic forecasting. *Transportation Research Part C: Emerging Technologies, 134*, Article 103466.

Li, J., Guo, F., Sivakumar, A., Dong, Y., & Krishnan, R. (2021). Transferability improvement in short-term traffic prediction using stacked LSTM network. *Transportation Research Part C: Emerging Technologies, 124*, Article 102977.

Li, Q., Han, Z., & Wu, X.-M. (2018). Deeper insights into graph convolutional networks for semi-supervised learning. *The thirty-second AAAI conference on artificial intelligence*.

Lin, Z., Feng, J., Lu, Z., Li, Y., & Jin, D. (2019). Deepstn+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis. In *The proceedings of the AAAI conference on artificial intelligence* (pp. 1020–1027).

Luo, D., Zhao, D., Ke, Q., You, X., Liu, L., Zhang, D., Ma, H., & Zuo, X. (2020). Fine-grained service-level passenger flow prediction for bus transit systems based on multitask deep learning. *IEEE Transactions on Intelligent Transportation Systems, 22*(11), 7184–7199.

Lv, M., Hong, Z., Chen, L., Chen, T., Zhu, T., & Ji, S. (2020). Temporal multi-graph convolutional network for traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems, 22*(6), 3337–3348.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C., & Iyengar, S. S. (2018). A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR), 51*(5), 1–36.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The graph neural network model. *IEEE Transactions on Neural Networks, 20*(1), 61–80.

Shao, E., Wang, H., Feng, J., Xia, T., Yang, H., Geng, L., Jin, D., & Li, Y. (2021). DeepFlowGen: Intention-aware fine grained crowd flow generation via deep neural networks. *IEEE Transactions on Knowledge and Data Engineering*.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., & Woo, W.-C. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems* (p. 28).

Sun, J., Zhang, J., Li, Q., Yi, X., Liang, Y., & Zheng, Y. (2022). Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks. *IEEE Transactions on Knowledge and Data Engineering, 34*(5), 2348–2359.

Tong, Y., Chen, Y., Zhou, Z., Chen, L., Wang, J., Yang, Q., Ye, J., & Lv, W. (2017). The simpler the better: A unified approach to predicting original taxi demands based on large-scale online platforms. In *The proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1653–1662).

Van Der Voort, M., Dougherty, M., & Watson, S. (1996). Combining Kohonen maps with ARIMA time series models to forecast traffic flow. *Transportation Research Part C: Emerging Technologies, 4*(5), 307–318.

Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., & Cottrell, G. (2018). Understanding convolution for semantic segmentation. In *The 2018 IEEE winter conference on applications of computer vision (WACV)* (pp. 1451–1460).

Wang, X., Ma, Y., Wang, Y., Jin, W., Wang, X., Tang, J., Jia, C., & Yu, J. (2020). Traffic flow prediction via spatial temporal graph neural network. In *The proceedings of the web conference 2020* (pp. 1082–1092).

Williams, B. M., & Hoel, L. A. (2003). Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of Transportation Engineering, 129*(6), 664–672.

Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). Cbam: Convolutional block attention module. In *The proceedings of the European conference on computer vision (ECCV)* (pp. 3–19).

Xie, P., Li, T., Liu, J., Du, S., Yang, X., & Zhang, J. (2020). Urban flow prediction from spatiotemporal data using machine learning: A survey. *Information Fusion, 59*, 1–12.

Xie, Y., Niu, J., Zhang, Y., & Ren, F. (2022). Multisize patched spatial-temporal transformer network for short-and long-term crowd flow prediction. *IEEE Transactions on Intelligent Transportation Systems, 23*(11), 21548–21568.

Xu, C., Ji, J., & Liu, P. (2018). The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets. *Transportation Research Part C: Emerging Technologies, 95*, 47–60.

Xu, Z., Wang, Y., Long, M., Wang, J., & KLiss, M. (2018). PredCNN: Predictive learning with cascade convolutions. In *IJCAI* (pp. 2940–2947).

Yang, J., Liu, T., Li, C., Tong, W., Zhu, Y., & Ai, W. (2021). MGSTCN: A multi-graph spatio-temporal convolutional network for metro passenger flow prediction. In *The*

2021 7th international conference on big data computing and communications (BigCom) (pp. 164–171).

Yao, H., Tang, X., Wei, H., Zheng, G., & Li, Z. (2019). Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *The proceedings of the AAAI conference on artificial intelligence* (pp. 5668–5675).

Yao, H., Wu, F., Ke, J., Tang, X., Jia, Y., Lu, S., Gong, P., Ye, J., & Li, Z. (2018). Deep multi-view spatial-temporal network for taxi demand prediction. *The Proceedings of the AAAI conference on artificial intelligence*.

Ye, J., Zhao, J., Ye, K., & Xu, C. (2020). Multi-stgcnet: A graph convolution based spatial-temporal framework for subway passenger flow forecasting. In *The 2020 international joint conference on neural networks (IJCNN)* (pp. 1–8).

Yuan, H., Zhu, X., Hu, Z., & Zhang, C. (2020). Deep multi-view residual attention network for crowd flows prediction. *Neurocomputing, 404*, 198–212.

Zhang, J., Zheng, Y., & Qi, D. (2017). Deep spatio-temporal residual networks for citywide crowd flows prediction. *The thirty-first AAAI conference on artificial intelligence*.

Zhang, J., Zheng, Y., Qi, D., Li, R., Yi, X., & Li, T. (2018). Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artificial Intelligence, 259*, 147–166.

Zhang, J., Zheng, Y., Sun, J., & Qi, D. (2019). Flow prediction in spatio-temporal networks based on multitask deep learning. *IEEE Transactions on Knowledge and Data Engineering, 32*(3), 468–478.

Zhang, W., Yu, Y., Qi, Y., Shu, F., & Wang, Y. (2019). Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning. *Transportmetrica A: Transport Science, 15*(2), 1688–1711.

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., & Li, H. (2019). T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems, 21*(9), 3848–3858.

Zheng, G., Chai, W. K., & Katos, V. (2022). A dynamic spatial–temporal deep learning framework for traffic speed prediction on large-scale road networks. *Expert Systems with Applications, 195*, Article 116585.

Zheng, Y. (2015). Trajectory data mining: An overview. *ACM Transactions on Intelligent Systems and Technology, 6*(3).

Zheng, Y., Capra, L., Wolfson, O., & Yang, H. (2014). Urban computing: Concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST), 5*(3), 1–55.

Zonoozi, A., Kim, J.-J., Li, X.-L., & Cong, G. (2018). Periodic-CRN: A convolutional recurrent model for crowd density prediction with recurring periodic patterns. In *IJCAI* (pp. 3732–3738).