electronics

MDPI

Article

Deep Convolutional Neural Network for Indoor Regional Crowd Flow Prediction

Qiaoshuang Teng [1], Shangyu Sun [1,2,*], Weidong Song [1], Jinzhong Bei [1,3] and Chongchang Wang [1]

[1] School of Geomatics, Liaoning Technical University, Fuxin 123000, China; souleleven@163.com (Q.T.)
[2] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China
[3] Chinese Academy of Surveying and Mapping, Beijing 100036, China
* Correspondence: shangyu_sun@126.com

Abstract: Crowd flow prediction plays a vital role in modern city management and public safety prewarning. However, the existing approaches related to this topic mostly focus on single sites or road segments, and indoor regional crowd flow prediction has yet to receive sufficient academic attention. Therefore, this paper proposes a novel prediction model, named the spatial–temporal attention-based crowd flow prediction network (STA-CFPNet), to forecast the indoor regional crowd flow volume. The model has four branches of temporal closeness, periodicity, tendency and external factors. Each branch of this model takes a convolutional neural network (CNN) as its principal component, which computes spatial correlations from near to distant areas by stacking multiple CNN layers. By incorporating the output of the four branches into the model's fusion layer, it is possible to utilize ensemble learning to mine the temporal dependence implicit within the data. In order to improve both the convergence speed and prediction performance of the model, a building block based on spatial–temporal attention mechanisms was designed. Furthermore, a fully convolutional structure was applied to the external factors branch to provide globally shared external factors contexts for the research area. The empirical study demonstrates that STA-CFPNet outperforms other well-known crowd flow prediction methods in processing the experimental datasets.

Keywords: indoor regional crowd flow prediction; crowd flow trajectories; deep learning; spatial-temporal attention mechanism; feature fusion method

Citation: Teng, Q.; Sun, S.; Song, W.; Bei, J.; Wang, C. Deep Convolutional Neural Network for Indoor Regional Crowd Flow Prediction. *Electronics* **2024**, *13*, 172. https://doi.org/10.3390/electronics13010172

Academic Editors: Phivos Mylonas, Katia Lida Kermanidis and Manolis Maragoudakis

Received: 15 November 2023
Revised: 25 December 2023
Accepted: 29 December 2023
Published: 30 December 2023

1. Introduction

With the advancement of urbanization, the population of regional central cities swells, and "urban diseases", such as traffic congestion, fuel consumption and environmental pollution, gradually emerge, bringing enormous pressure to urban management and posing severe challenges to urban sustainable development. To solve the existing problems in urban development, China has constructed smart cities on a wide scale. This study aimed to estimate and predict crowd flow volume, which has great theoretical and practical significance for improving the ability of urban management departments to deal with emergencies.

Therefore, relevant scholars in this field have carried a large quantity of studies for the purpose of predicting the crowd flow volume using several types of methods, such as statistics-based methods, traditional machine learning methods, deep learning methods and reinforcement learning methods. However, the majority of existing studies focus on the prediction of outdoor crowd flow volume, emphasizing population mobility between different functional urban areas in the city [1], and few academics pay attention to indoor regional crowd flow prediction.

The maturity of indoor positioning technology advances the development of indoor location-based services (LBSs) and promotes the cultivation of typical application scenarios such as personalized route recommendation, indoor service resource allocation and the

formulation of emergency response plans in large and complex indoor scenes. As such, advances in this research area could open new application fields for surveying and mapping geographic information technology. More than 80% of daily life is spent indoors [2], and various application scenarios of indoor LBSs are also structured around indoor activity. Therefore, the estimation and prediction of indoor regional crowd flow volume has strong practical application value for improving the application system of indoor LBSs and strengthening indoor LBS provisions. For example, museums can predict the future crowd flow volume according to historical data and adjust the distribution location and opening time of each exhibition hall in advance, allowing institutions to mostly avoid unnecessary safety accidents caused by excessive congestion in exhibition halls as a result of a large flow of crowd. In another example, an indoor LBS system can dynamically adjust the operation state of the sensors, as well as the temporal and spatial distribution of the crowd flow volume, to assist a sensor network for the purpose of realizing the dynamic management of power consumption. That is, when the crowd flow volume of the local area is large, the sensor power consumption can be increased or reduced.

The principal challenges of indoor regional crowd flow prediction can be summarized as follows:

1. Since it is impossible to deploy many sensors indoors to effectively monitor the crowd flow volume, there is an urgent need to establish a method that can accurately express the indoor regional crowd flow volume.
2. The crowd flow volume of a certain indoor region is influenced by adjacent regions. Thus, indoor regional crowd flow prediction cannot ignore the spatial correlation between different indoor regions.
3. Indoor regional crowd flow prediction is a typical time series prediction problem, and its calculation result is affected by time dependence. For instance, indoor regional crowd flow volumes during adjacent periods display little difference, and the indoor regional crowd flow volumes during the same period every day may be similar. Therefore, it is necessary to capture closeness and periodicity, as well as tendency between indoor regional crowd flow volumes during different periods.
4. External factors, such as weather conditions, holiday arrangements, activities, etc., may change the indoor regional crowd flow volume. Thus, in the process of indoor regional crowd flow prediction, external factors must be considered.

In view of the above challenges, this paper proposes a model named the spatial–temporal attention-based crowd flow prediction network (STA-CFPNet) to predict indoor regional crowd flow volume. Our contributions can be summarized as follows:

1. We propose a deep-learning-based crowd flow prediction model for indoor regions, known as STA-CFPNet. STA-CFPNet feeds on indoor trajectory data and can capture spatial correlation and time dependence by stacking multiple convolutional neural network (CNN) blocks. In addition, the external factor learning branch is proposed in order to add external factors to STA-CFPNet to improve the prediction accuracy.
2. We introduce a modeling and expression method for application to indoor regional crowd flow volume. This method converts the indoor space into spatial latticed grids and produces crowd flow matrices that represent the cumulative number of trajectory segments passing through the grid per unit time. This micro-granularity can truly reflect the indoor regional crowd flow situation.
3. We design a spatial–temporal attention block (STATT) with the residual structure and add it to the temporal closeness, periodicity and tendency branches of STA-CFPNet, enabling the model to learn spatial correlations and time dependence more efficiently. Additionally, this reduces the difficulty of model fitting and accelerates the convergence speed of the model.
4. We propose an encoding method that can encode external factors affecting the indoor regional crowd flow volumes as vectors. By inputting these vectors into the external factor learning branch of STA-CFPNet composed of multiple convolutional layers, it

is possible to obtain the randomness features implicit in the indoor regional crowd flow volumes so that the prediction accuracy of STA-CFPNet can be enhanced.

2. Related Work

With the development of positioning technology, we can obtain greater amounts of data containing location information. The question of how to apply these data has attracted widespread attention from scholars. Crowd flow prediction based on positioning data has been widely studied, serving as a research focus in recent years. In this section, we discuss two mainstream crowd flow prediction methods: statistics-based methods and deep-learning-based methods.

Traditional statistics-based methods, such as autoregressive integrated move average (ARIMA), seasonal ARIMA (SARIMA) [3], KARIMA [4] and ARIMAX [5], were proposed in order to predict traffic flow by considering both spatial and temporal features. Although statistics-based methods display simple model structures and are easily explained, they are incapable of considering aspects such as individual randomness and nonlinearity, as well as being inapplicable to large-scale scenarios [6]. These concerns limit the application of statistics-based methods in the field of crowd flow prediction.

In recent years, with the rapid development of deep learning, deep-learning-based methods have been widely employed in crowd flow prediction due to their ability to describe complicated, nonlinear data correlations. These deep-learning-based methods can be divided into two categories: outdoor crowd flow and indoor crowd flow prediction models. Most studies focus on outdoor crowd flow prediction based on regular grid-based regions, for example, ref. [7] proposed ST-ResNet to forecast crowd flow volume in each and every region of a city, which can comprehensively consider multiple complex factors. Other research [8] proposed a deep-learning-based multi-branch model named traffic flow forecasting network (TFFNet). This not only captures spatial correlation and temporal dependence, but also considers external factors in order to predict city-wide traffic flow. One article [9] proposed a deep attentive adaptation network model named ST-DAAN to transfer spatial–temporal knowledge for urban crowd flow prediction. In addition, a global spatial attention mechanism was designed to capture spatial dependencies, which is useful in efforts to improve prediction accuracy. Another study [10] proposed an adversarial learning framework for multi-step urban crowd flow prediction. This can not only capture the spatial–temporal correlation of the crowd flow data sequence, but is able to learn the external context features in a fine-grained manner. In addition, some outdoor crowd flow prediction models were proposed to solve the prediction task for irregular regions. For example, ref. [11] proposed a crowd flow prediction model for irregular regions. This method not only extracts hierarchical spatial–temporal correlation, but also captures dynamic and semantic information among the regions. A number of authors [12] built a multi-view graph convolutional network (MVGCN) for the crowd flow forecasting problems in irregular regions. MVGCN not only captures spatial correlations and many types of temporal properties, but it also assesses external factors and meta features. Furthermore, some research has been conducted for indoor crowd flow prediction. For example, ref. [13] proposed a Wi-Fi-positioning-based multi-grained spatiotemporal crowd flow prediction framework named CrowdTelescope. This framework adopted spatiotemporal graph neural networks (GNNs) to predict crowd flow using Wi-Fi connection records. One study [14] presented a transformer-based multi-scale indoor mobility model named WiFiMod that is capable of capturing the mobility periodicity and correlation across various scales, as well as long-term mobility dependencies, in order to obtain robust accuracy in indoor mobility prediction. One author [15] proposed a sequence-to-sequence crowd flow prediction deep learning network named DeepIndoorCrowd. This can not only capture historical temporal and future temporal, but can also consider the semantic features of indoor stores in the predictions. Based on the complex architecture, outdoor crowd flow prediction models utilizing deep learning can obtain abundant information on different scales, achieving better results than statistics-based methods. However, due to differences between the

spatial scales of outdoor and indoor spaces, outdoor crowd flow prediction models cannot be directly applied to the task of indoor regional crowd flow prediction. Additionally, the existing indoor crowd flow prediction models lack comprehensive consideration of spatial–temporal information and external factors that affect indoor crowd flow volume.

Therefore, inspired by previous research, this paper proposes a deep-learning-based prediction model for indoor regions. This model not only takes spatial correlation, time dependence and external factors into account, but it also designs the STATT block to enhance the data features that are beneficial for the model training. The model will be described in detail below.

3. Materials and Methods

3.1. Modeling and Expression of Indoor Regional Crowd Flow Volume

As a deep learning model, CNNs display efficient data feature extraction, strong nonlinear expression and robust generalization ability. They have been widely deployed in fields such as speech analysis, image recognition, object detection, etc. In this study, we constructed STA-CFPNet based on CNNs. Since the input of CNNs is usually in vector, matrix, or tensor form, we must divide the indoor space into regular grids and transform it into a crowd flow matrix. At the same time, it is necessary to split the trajectory data into equal time intervals, at which point it is possible to obtain the crowd flow volume per time interval of each grid unit. In this study, we selected 2 m as the side length of the grid unit and set the division granularity of the time axis as 15 min [16].

As it is impossible to deploy a variety of sensors indoors to effectively monitor the crowd flow volume, we utilized an indoor positioning dataset collected from a four-story shopping mall in Beijing with the support of the National Natural Science Foundation of China (grant number 42071343). This dataset records the Wi-Fi positioning data of customers' mobile phones in the shopping mall from 1 May 2019 to 31 May 2019. The example positioning data are shown in Table 1. Connecting the positioning data of the same user ID within the specified time, it is possible to obtain the indoor trajectory data. This study deployed indoor trajectory data to calculate the indoor regional crowd flow volume. The computational process unfolds as follows:

1. Based on the relationship between trajectory data and grid units, we calculate the cumulative number of trajectory segments passing through the grid unit in unit time as the indoor crowd flow volume. As shown in Figure 1a, two trajectory data intersect area R_{ij} to form three trajectory segments; thus, the indoor crowd flow volume of area R_{ij} is recorded as 3.
2. It should be noted that, due to the long sampling interval of positioning data, the indoor trajectories cannot demonstrate the actual movement of customers well. To improve the accuracy of indoor crowd flow volume, we carry out Hermite interpolation on positioning data in advance. As shown in Figure 1b, when using the Hermite method, three trajectory data intersect area R_{ij} to form four trajectory segments; thus, the indoor crowd flow volume of area R_{ij} is recorded as 4.
3. On this basis, we acquire the crowd flow volume of the whole indoor space, as shown in Figure 1c.

Table 1. Example of indoor positioning data.

Time Stamp	Floor ID	User ID	X	Y
1 May 2019 09:30:20	F1	789433A1***	13***91.7	4***74.3
1 May 2019 09:33:54	F2	509122U0***	13***55.0	4***77.4
...	...	...	...	...
1 May 2019 18:42:35	F4	289738G6***	13***47.9	4***06.3

*** indicates omissions.

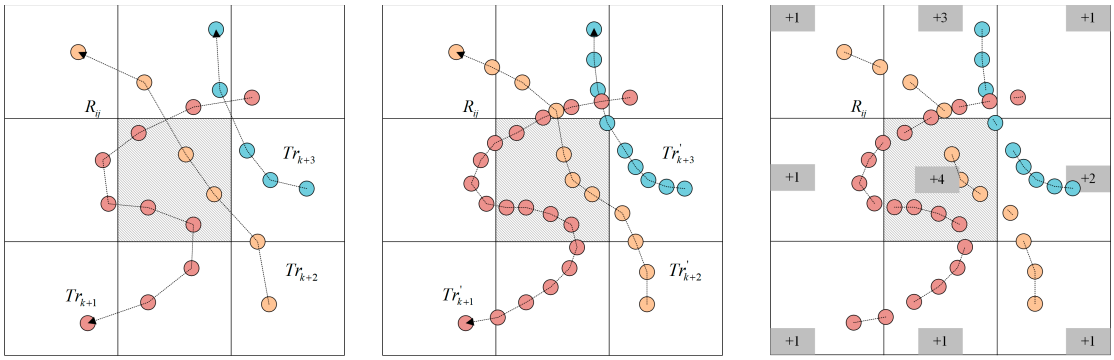

Figure 1. Calculation process for indoor regional crowd flow volume. (**a**) Spatial connection operation; (**b**) Hermite interpolation; (**c**) calculation results of indoor regional crowd flow volume.

For a grid unit R_{ij}, the "area" that lies at the i^{th} row and the j^{th} column of the indoor space, the cumulative crowd flow volume x_t^{ij} at time interval t^{th} is defined as follows:

$$x_t^{ij} = \sum_{Tr_k \in S} \left| \{ k \geq 1 | Tr_k \cap R_{ij} \neq \varnothing \} \right|, \quad (1)$$

where S is a group of trajectories, Tr_k is a trajectory in S and $Tr_k \cap R_{ij} \neq \varnothing$ is a set of trajectory segments obtained when trajectory Tr_k intersects area R_{ij}, $|\cdot|$ is the total number of trajectory segments in the above set.

By calculating the indoor regional crowd flow volume using the above method, it is possible to obtain the state sequence comprising crowd flow matrices. For the convenience of data processing, the value of the crowd flow volume was scaled to the range of [0, 1]. Taking the trajectory data on the first floor of a shopping mall on 1 May 2019, in the experimental dataset as an example, the plan view of the first floor of the shopping mall is shown in Figure 2. A total of 96 crowd flow matrices were acquired during a 15 min time interval, and each matrix contained 2400 grid units, as shown in Figure 3. The closer the color in the picture is to a cool tone (i.e., blue), the smaller crowd flow volume is. On the contrary, the closer the color is to a warm tone (i.e., yellow), the greater the crowd flow volume is.

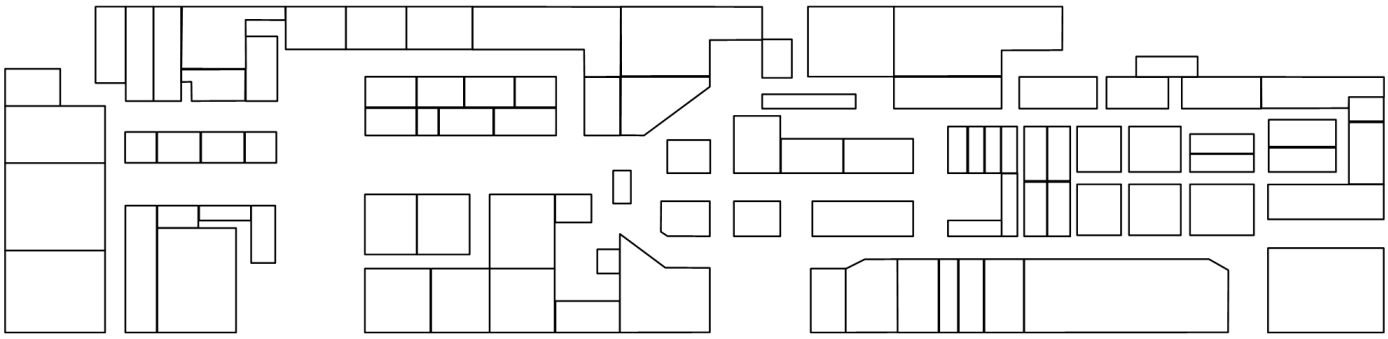

Figure 2. Plan view of the first floor of the shopping mall.

Indoor regional crowd flow volumes within a certain area have closeness, periodicity and tendency in terms of time. As shown in Figure 4a, the numerical difference between crowd flow volumes in adjacent periods during business hours is small. However, with the arrival of morning and evening peak times, the crowd flow volumes change greatly. From Figure 4b,c, it is possible to observe that the values of crowd flow volumes during the same period of each day or week are similar. However, with the influence of working days and rest days, as well as other factors such as weather conditions and holiday arrangements, crowd flow volumes produce fluctuations.

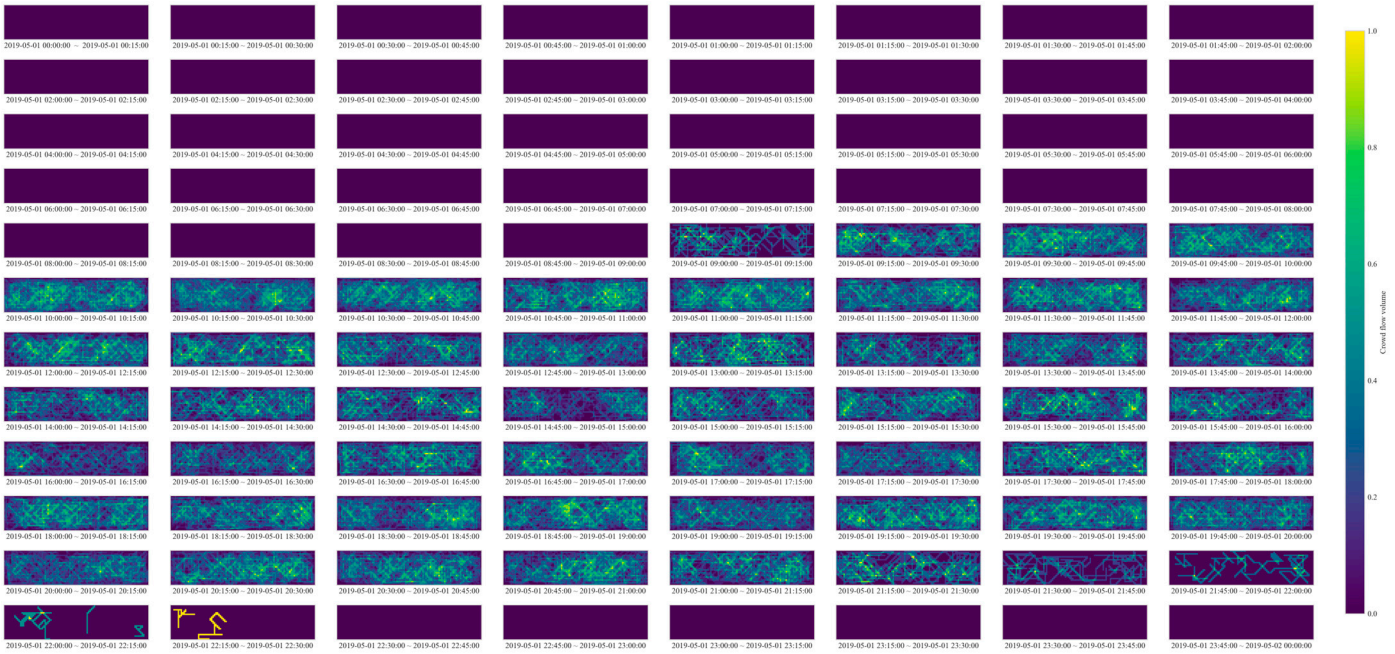

Figure 3. Visualization of indoor regional crowd flow volume in experimental area.

The space and time of indoor regional crowd flow volume do not only display correlation and dependence, but also reveal outliers and randomness. Thus, it is necessary to learn the spatial correlation and time dependence of crowd flow volume from multiple perspectives by extracting and selecting several characteristics of crowd flow volume. Simultaneously, we must fully consider the outliers and randomness that may affect the inherent spatial–temporal pattern of crowd flow volume. By effectively fusing spatial correlation, time dependence and some external factors, the prediction performance of indoor regional crowd flow prediction model can be significantly enhanced.

3.2. Indoor Regional Crowd Flow Prediction Model

The results of the spatial–temporal correlation analysis show that indoor regional crowd flow volume is spatially correlated, time dependent and random under the influence of external factors. To establish the inherent spatial–temporal patterns in indoor activities, we modeled the correlation, dependence and randomness in the crowd flow matrix from multiple perspectives, such as space, time, external factors, etc.

Since indoor regional crowd flow volume can be expressed as a two-dimensional matrix with a grid structure, CNNs can be directly applied in order to extract spatial correlation features. By stacking several CNN blocks, spatial correlation can be captured on multiple scales, enabling the spatial pattern to be learnt from local to global levels. Crowd flow volume displays similarities during adjacent periods and shows inherent periodicity over a long temporal duration. Considering the influence of periodicity in the model can allow researchers to effectively improve prediction accuracy and reduce the difficulty of model fitting. Therefore, we construct independent branches for the extraction of temporal closeness, periodicity and tendency features. External factors, such as weather conditions and holiday arrangements, will disturb short-term crowd flow prediction tasks and increase the randomness in the process of model training. Thus, it is necessary to embed features through independent model branches and provide globally shared external factors contexts for spatial–temporal pattern learning branches.

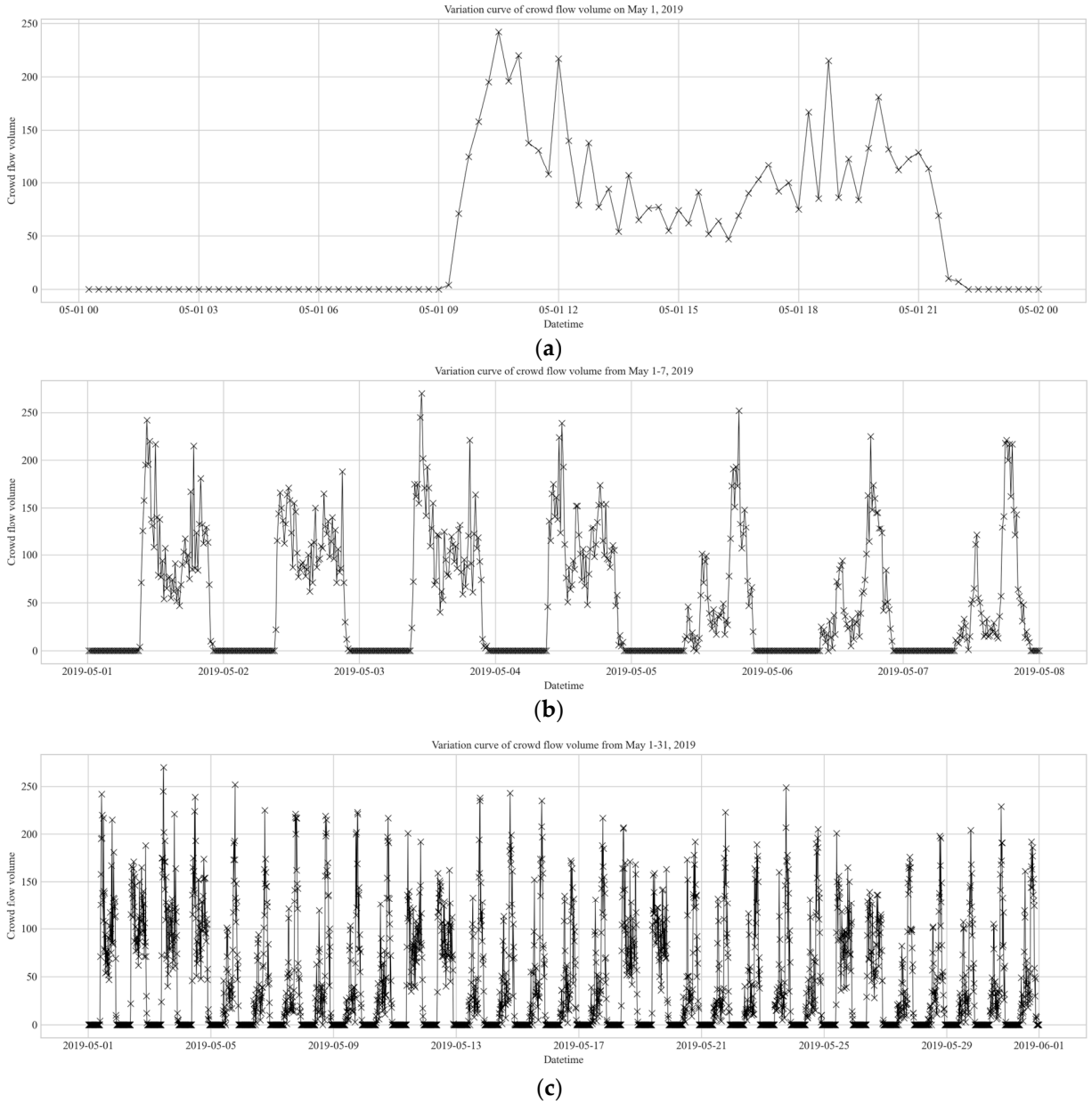

Figure 4. Variation curve of indoor regional crowd flow volume in experimental area. (**a**) Variation curve of crowd flow volume on 1 May 2019; (**b**) variation curve of crowd flow volume from 1 to 7 May 2019; (**c**) variation curve of crowd flow volume from 1 to 31 May 2019.

Based on the above analysis, we constructed an integrated learning model composed of four model branches: temporal closeness, periodicity, tendency and external factors. Subsequently, we fused the outputs of multiple branches to acquire the predicted value of crowd flow volume for a certain period in the future. Finally, we used the back-propagation method to train the model, finding that the model performed best using the validation dataset.

We dubbed the indoor crowd flow prediction model proposed in this paper as STA-CFPNet. By adding STATT to the three branches of temporal closeness, periodicity and tendency, the model can strengthen the data features, which are conducive to spatial correlation and time dependence learning. Additionally, this model can be used to reduce the difficulty of subsequent block fitting and accelerate the overall convergence speed of the model. For external factors, the branch centered around feature embedding and extraction, based on a fully convolutional structure, can be used learn the globally shared

external factors contexts for the research area. This allows for the realization of the feature expression of randomness during any time period, and effectively improves the ability to fit randomness in the model. To fuse the output feature maps of multiple model branches, we integrate the feature fusion module at the output of the model, aiming to dynamically combine the feature maps of multiple model branches using the model training process and add a nonlinear activation function.

The architecture of STA-CFPNet is shown in Figure 5. The branches C, P and Q at the bottom of Figure 5 are used to learn the spatial–temporal patterns implied in the crowd flow matrix. We divide the time axis into three segments: distant history, near history and recent. Then, we extract X_C, X_P and X_Q three tensors composed of crowd flow matrix sequences as the inputs of each model branch:

$$X_C = \left[X_{T_i-1}, X_{T_i-2}, \cdots, X_{T_i-l_c}\right], \tag{2}$$

$$X_P = \left[X_{T_i-p}, X_{T_i-2\times p}, \cdots, X_{T_i-l_p\times p}\right], \tag{3}$$

$$X_Q = \left[X_{T_i-q}, X_{T_i-2\times q}, \cdots, X_{T_i-lq\times q}\right], \tag{4}$$

where l_c, l_p and l_q are the length of X_C, X_P and X_Q, respectively, and p and q are two measurement units of periodicity and the tendency to express the length of time intervals, respectively.

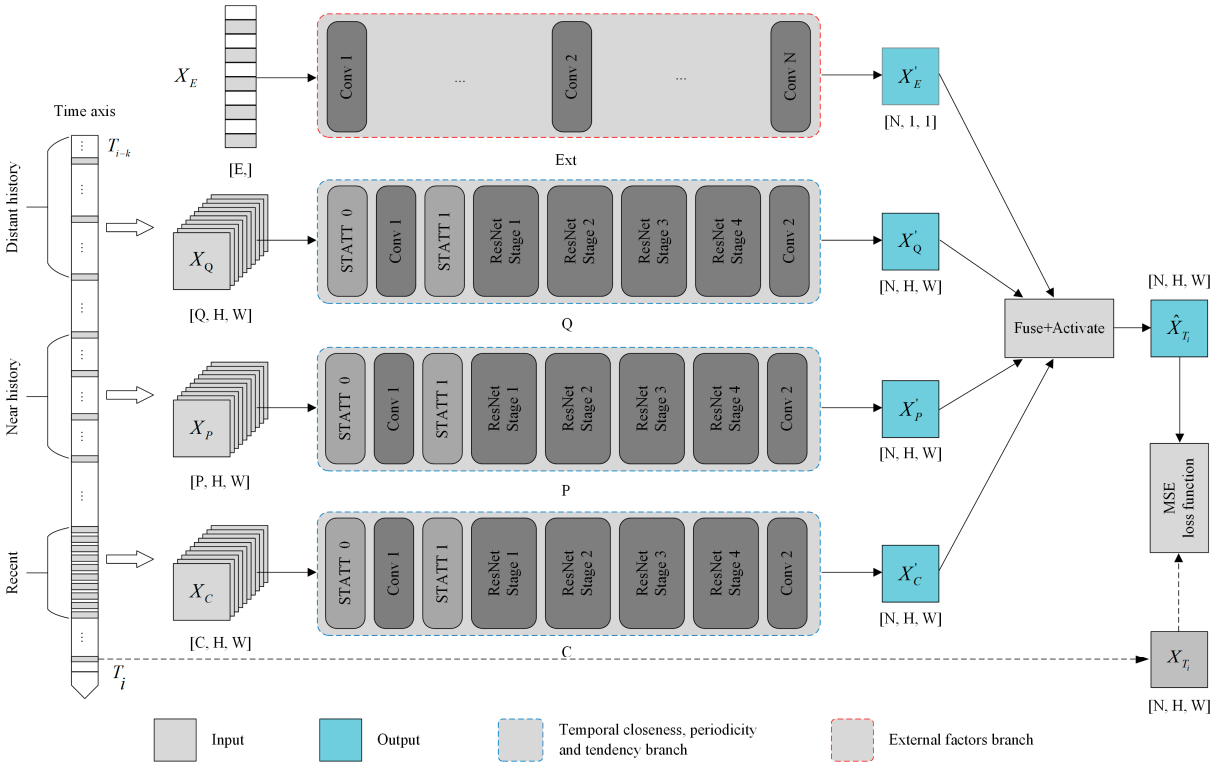

Figure 5. STA-CFPNet architecture.

The Ext branch is used for feature embedding and the extraction of external factors. The model input X_E for the Ext branch is relatively simple provided that weather conditions, holiday arrangements, business status and other external factors are encoded into distinguishable vectors. For example, we can use "one-hot encoding" to encode each external factor and then splice the individual vectors.

After passing the output feature maps X'_C, X'_P, X'_Q and X'_E of each branch through the "fuse and activate" module, the final output X'_{T_i} of the crowd flow prediction model is

obtained. Then, the loss between predicted value X'_{T_i} and true value X_{T_i} can be calculated using the "loss function". Afterward, we utilize the optimization algorithm to back-propagate the model and adjust the model parameters until a convergence state is reached, allowing the acquisition of the prediction model with the best performance in terms of the verification dataset.

3.2.1. Spatial–Temporal Pattern Learning Branch Architecture

By continuously sliding along the input feature map, the convolution kernel of the CNN performs a linear weighting operation between the sliding window on the feature map and the corresponding elements on the convolution kernel. This allows researchers to extract the local spatial features of the input feature map. Compared with the fully connected layer, the CNN layer possesses the characteristics of a local connection, weight sharing and hierarchical expression. A local connection indicates that an element on the output feature map is only related to the local area on the input feature map. Weight sharing means that the weight coefficients of the convolution kernel are globally shared on the same input feature map. Hierarchical expression means that feature representations can be extracted on multiple scales by stacking multiple CNN layers.

Operating based on the analysis given above, we use the structure of stacking multiple CNN layers to extract the spatial correlation features hidden in the crowd flow matrix. Their model input structures are similar for the temporal closeness, periodicity and tendency model branches. Therefore, we adopted the same model structure for C, P and Q model branches. The architecture of a spatial–temporal pattern learning branch is shown in Figure 6.

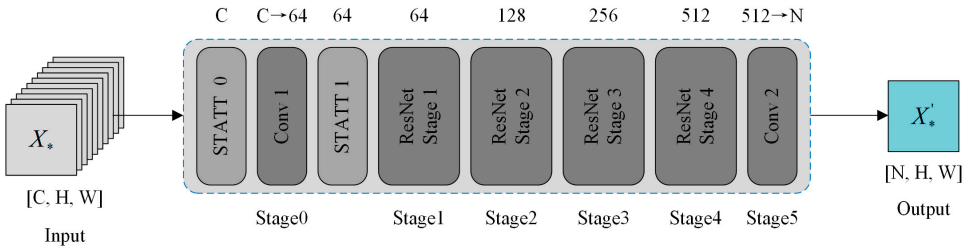

Figure 6. Spatial–temporal pattern learning branch architecture.

The branch is divided into five stages, in which Stage 0 and Stage 5 are convolution layers, and the sizes of the convolution core are [64, 3, 3] and [N, 3, 3], respectively. Their primary functions include feature extraction and channel adjustment. Conv1 adjusts the number of feature map channels from C to 64, and Conv2 adjusts the number of feature map channels from 512 to N. ResNet Stages 1–4 are residual blocks proposed in [17]. The convolution kernels in the residual units are [64, 3, 3], [128, 3, 3], [256, 3, 3] and [512, 3, 3], respectively, which can increase the depth of the model branch and extract spatial features on multiple scales. Compared with the ResNet50, ResNet101 and ResNet152 models proposed in [17], the depth of the model used in this study is relatively shallow. Conv1, ResNet Stages 1–4 and Conv2 form the backbone of the spatial–temporal pattern learning branch. The model input size is [C, H, W] and the output size is [N, H, W], where H and W are the height and width of the feature map, respectively, C is the length of the crowd flow matrix sequence, and N is the number of channels in the output feature map, which is determined on the basis of the specific needs of the crowd flow prediction task. For example, the cumulative flow prediction can be 1, and the inflow and outflow prediction can be 2.

According to the calculation principle of multi-channel convolution, a convolution kernel performs linear weighting operations with the sliding window on the input feature map, and then perform tensor addition operations on each channel of the output feature map, thereby achieving feature extraction in the spatial dimension and feature fusion in the channel dimension. By stacking multiple layers of CNNs, spatial–temporal pattern learning

from low to high layers can be achieved. However, the traditional CNN architecture design is too general to effectively pay attention to the data characteristics beneficial to spatial–temporal pattern learning and model the spatial correlation and time dependence implied in the crowd flow matrix. When operating on the premise of limited model depth, it is especially difficult to achieve good results in the task of crowd flow prediction. An attention mechanism was first applied in the research fields of machine translation [18] and image classification [19]. Inspired by the human visual attention mechanism, we designed a special attention learning branch to make the network pay attention to the data characteristics that offer benefits to the machine learning task in the training process, reducing the complexity of the model learning process and improving the convergence speed of the model training process.

There exists a significant spatial correlation between the crowd flow volume V_{mn}^{i} of area R_{mn} during period T_i and the surrounding areas. The correlation is strong between the areas close to each other, while elsewhere, it is weak. To characterize this spatial correlation, the model learns a location mask matrix of area R_{mn} during period T_i from the surrounding areas. Meanwhile, the values of crowd flow volume V_{mn}^{j} in area R_{mn} express time dependence during several adjacent periods T_j, a value which gradually decreases with the evolution of time. The model may learn a channel mask vector in the training process to characterize the importance of crowd flow volume during each period. In order to mine the hidden spatial–temporal dynamic patterns of people during indoor activities, this paper proposes a STATT block for the learning of location mask matrix and channel mask vector, which can calibrate the features of input data and shallow feature maps, strengthening the data features beneficial to spatial correlation and time dependence learning, reducing the fitting difficulty of subsequent blocks and improving the overall convergence speed of the model. The architecture of the STATT block is shown in Figure 7.

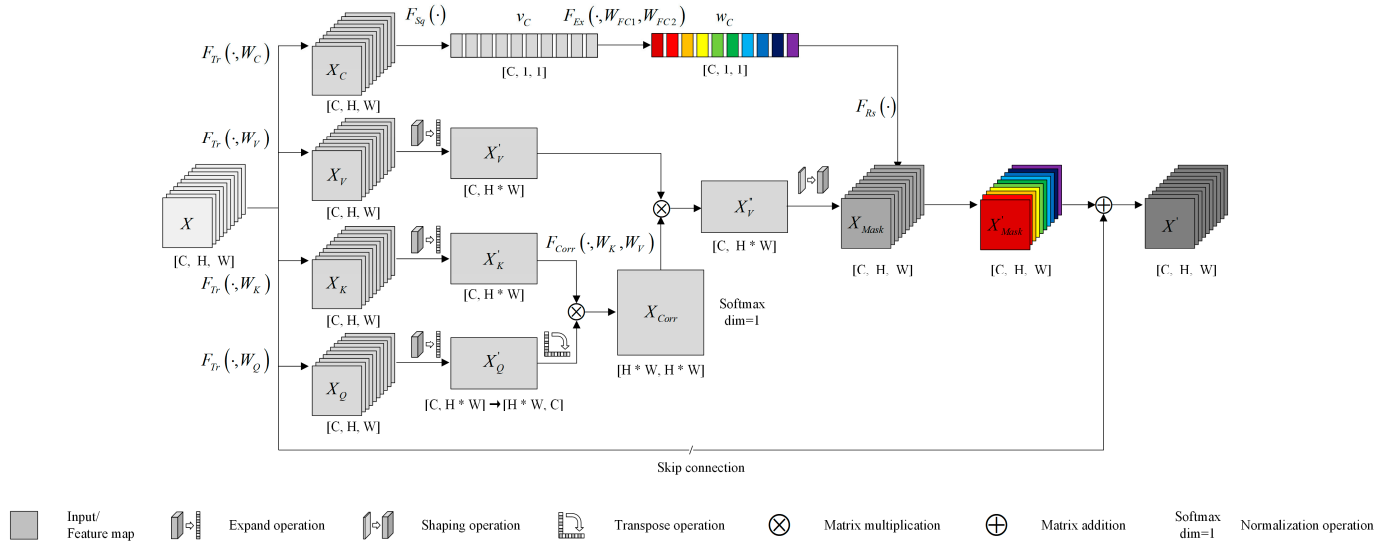

Figure 7. STATT block architecture.

The STATT block is designed to display a residual structure. It is possible to calibrate the input feature map in the residual part of the block, use the skip connection to perform tensor addition with the output of the residual part, and finally obtain the output feature map of the block. The input and output size of the block are both [C, H, W].

In order to acquire the location mask matrix $X_{Mask} \in \mathbb{R}^{C\times H\times W}$, we first learn three feature mapping functions $F_{Tr}(\cdot, W_Q)$, $F_{Tr}(\cdot, W_K)$ and $F_{Tr}(\cdot, W_V)$ in order to map the input feature map $X \in \mathbb{R}^{C\times H\times W}$ to the hidden layer feature space, obtaining $X_Q \in \mathbb{R}^{C\times H\times W}$, $X_K \in \mathbb{R}^{C\times H\times W}$ and $X_V \in \mathbb{R}^{C\times H\times W}$. Then, we expand the feature maps of the above hidden layer feature space along the latter two dimensions to obtain $X'_Q \in \mathbb{R}^{C\times(H\times W)}$,

$X'_K \in \mathbb{R}^{C\times(H\times W)}$ and $X'_V \in \mathbb{R}^{C\times(H\times W)}$. In order to calculate the correlation between any two positions $\langle i, j\rangle$ on the feature map X, it is necessary to use a correlation calculation function $F_{Corr}(\cdot, W_Q, W_K)$ to obtain the correlation matrix $X_{Corr} \in \mathbb{R}^{(H\times W)\times(H\times W)}$ and normalize the matrix along the second dimension. Then, it is necessary to multiply the hidden layer feature map X'_V and the correlation matrix X_{Corr} and assign the position attention coefficient to each position on the feature map to obtain the output of this stage $X''_V \in \mathbb{R}^{C\times(H\times W)}$. Based on the above calculation process, it is possible to obtain the location mask matrix X_{Mask} as long as X''_V is reshaped to the size of the input feature map of the STATT block.

$F_{Tr}(\cdot, W_Q)$, $F_{Tr}(\cdot, W_K)$, $F_{Tr}(\cdot, W_V)$ and $F_{Corr}(\cdot, W_Q, W_K)$ can be formally expressed as:

$$F_{Tr}(\cdot, W_Q) = \sigma(W_Q X + b_Q), \tag{5}$$

$$F_{Tr}(\cdot, W_K) = \sigma(W_K X + b_K), \tag{6}$$

$$F_{Tr}(\cdot, W_V) = \sigma(W_V X + b_V), \tag{7}$$

$$F_{Corr}(X, W_Q, W_K) = Softmax\left(X^T W_Q^T W_K X\right), \tag{8}$$

where W_Q, W_K and W_V are convolution kernels with a size of 1×1; b_Q, b_K and b_V are the paranoid terms of linear mapping; $\sigma(\cdot)$ is the nonlinear activation function; and $Softmax(\cdot)$ is the normalization function.

By combining the above four formulas, we can acquire the calculation formula of X''_V:

$$X''_V = F_{Tr}(X, W_V)F_{Corr}(X, W_Q, W_K) = W_V X Softmax(X^T W_Q^T W_K X), \tag{9}$$

Based on X''_V, we can shape the tensor from [C, H*W] to [C, H, W] to acquire the location mask matrix X_{Mask}. The feature map of each channel in the location mask matrix has been calibrated for specific features, highlighting the data features beneficial to spatial correlation modeling. The location attention learned using the STATT block displays global invariance, i.e., the spatial correlation between any positions on the feature map is consistent.

To obtain the channel mask vector $w_C \in \mathbb{R}^{C\times1\times1}$, we first map the input feature map of the STATT block to the hidden layer feature space, obtaining $X_C \in \mathbb{R}^{C\times H\times W}$. The specific form of feature mapping function $F_{Tr}(\cdot, W_C)$ is as follows:

$$F_{Tr}(\cdot, W_C) = \sigma(W_C X + b_C), \tag{10}$$

where W_C is the convolution kernel with a size of 1×1, b_C is the paranoid term of linear mapping and $\sigma(\cdot)$ is the nonlinear activation function.

Then, we convert the feature map X_C of the hidden layer feature space into a coding vector that can characterize the global feature of the channel dimension. The specific form of the feature compression function $F_{Sq}(\cdot)$ is as follows:

$$F_{Sq}(X_C) = GlobalAvgPool(X_C), \tag{11}$$

where $GlobalAvgPool(\cdot)$ is the global average pooling function [20].

The coding vector v_C represents the global feature of the channel dimension. To capture the nonlinear relationship between any two channels, we must learn a feature activation function $F_{Ex}(\cdot, W_{FC1}, W_{FC2})$, the specific form of which is as follows:

$$F_{Ex}(v_C, W_{FC1}, W_{FC2}) = \sigma(W_{FC2} ReLU(W_{FC1} v_C)), \tag{12}$$

where W_{FC1} and W_{FC2} are convolution kernels with a size of 1×1 [21], and $ReLU(\cdot)$ and $\sigma(\cdot)$ are nonlinear activation functions.

Combining Formulas (11) and (12), we acquire the channel mask vector $w_C \in \mathbb{R}^{C\times1\times1}$, representing the importance of the channel dimension. Then, we calibrate the features

of the location mask matrix X_{Mask} again, multiplying the feature map of each channel in X_{Mask} by the weight coefficient of the corresponding channel in w_C to obtain the output $X'_{Mask} \in \mathbb{R}^{C\times H\times W}$ of the residual part of STATT block. The specific form of feature scaling function $F_{Rs}(\cdot)$ is as follows:

$$F_{Rs}(X_{Mask}, w_C) = w_C \cdot X_{Mask}, \tag{13}$$

Finally, it is possible to obtain the output feature map of STATT block $X' \in \mathbb{R}^{C\times H\times W}$:

$$X' = X + F_{Rs}(X_{Mask}, w_C) = X + w_C \cdot X_{Mask}, \tag{14}$$

3.2.2. External Factor Learning Branch Architecture

The Ext branch is used for feature embedding and the extraction of external factors; its input $X_E \in \mathbb{R}^E$ is a coded representation of external factors. Since it does not have the attribute of spatial location, it cannot be directly used to guide the learning of spatial–temporal pattern learning branches to improve the prediction performance of the model. In order to integrate the external factor vector into the learning process of the STA-CFPNet model, we constructed a feature embedding and extraction branch composed of multiple convolution layers to map the external factor vector into scalar values that can represent the global external factors contexts. The architecture of the external factor learning branch is shown in Figure 8.

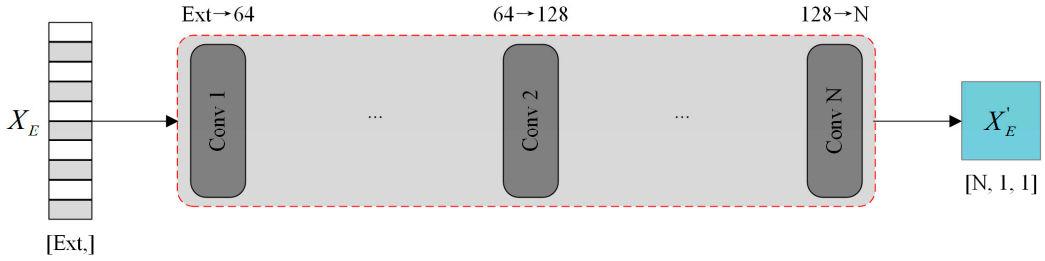

Figure 8. External factor learning branch architecture.

The output of the external factor learning branch $X'_E \in \mathbb{R}^{N\times 1\times 1}$ is calculated as follows:

$$X'_E = W_N(ReLU(W_2 ReLU(W_1 X_E))), \tag{15}$$

where W_1, W_2 and W_N are convolution kernels with a size of 1×1 and $ReLU(\cdot)$ is the nonlinear activation function.

3.2.3. Model Fusion Method

In order to integrate the output feature maps of multiple model branches, we used the fusion function $F_{Fuse}(\cdot)$ to merge the output feature maps X'_C, X'_P, X'_Q and X'_E, indicating the temporal closeness, periodicity, tendency and external factor branches. This operation allowed us to obtain the final output $\hat{X}_{T_i} \in \mathbb{R}^{N\times H\times W}$ of the model:

$$\hat{X}_{T_i} = Tanh(Fuse(X'_C, X'_P, X'_Q, X'_E)), \tag{16}$$

where $Fuse(\cdot)$ is the feature fusion function and $Tanh(\cdot)$ is the nonlinear activation function.

3.3. Model Training Method

By inputting the crowd flow matrix sequences X'_C, X'_P, X'_Q and external factor vector X'_E into each branch of STA-CFPNet, we can acquire the predicted value $\hat{X}_{T_i}$ of crowd flow volume during period T_i. We used mean squared error (MSE) to measure the difference between the real value and the predicted value:

$$L(\theta) = \left\| X_{T_i} - \hat{X}_{T_i} \right\|_2^2, \tag{17}$$

where θ represents all learnable parameters of STA-CFPNet.

We add a regularization term to the loss function in order to adjust the complexity of the model and avoid over-fitting problems, and obtain the objective function of STA-CFPNet:

$$\hat{\theta} = \underset{\theta}{argmin}(L(\theta) + \lambda J(\theta)), \tag{18}$$

where θ represents all learnable parameters of STA-CFPNet, $J(\theta)$ is the regularization term, usually taking L_1 or L_2 norm, and λ is the regularization coefficient.

The pseudo code of the STA-CFPNet training process is shown in Algorithm 1.

Algorithm 1: STA-CFPNet training algorithm

Input: crowd flow matrix: $\{X_{T_0}, X_{T_1}, \cdots, X_{T_{n-1}}\}$
external factor vector: $\left\{X_{Ext_{T_0}}, X_{Ext_{T_1}}, \cdots, X_{Ext_{T_{n-1}}}\right\}$
sequence length of crowd flow matrix: l_c, l_p, l_q
periodicity interval unit: p
tendency interval unit: q
Output: STA-CFPNet M
//construct training sample
$S \leftarrow \varnothing$
for all available time periods $t(1 \le t \le n-1)$ do
  $X_C = \left[X_{T_i-1}, X_{T_i-2}, \cdots, X_{T_i-l_c}\right]$
  $X_P = \left[X_{T_i-p}, X_{T_i-2\times p}, \cdots, X_{T_i-l_p\times p}\right]$
  $X_Q = \left[X_{T_i-q}, X_{T_i-2\times q}, \cdots, X_{T_i-lq\times q}\right]$
  construct training sample$\left(\{X_C, X_P, X_Q, X_E\}, X_{T_i}\right)$ and add it to dataset S in order
//train STA-CFPNet
initialize all learnable parameters of STA-CFPNet θ [22]
repeat
  randomly select a subset S_b from dataset S
  find a set of parameters $\hat{\theta}$ on current subset S_b that minimizes the objective function
until the convergence condition is reached
output the trained STA-CFPNet M

4. Results and Discussion

4.1. Experiment Settings

To verify the effectiveness of the model proposed in this paper, we chose seven benchmark models that can be used for the crowd flow prediction task, the details of which are as follows. The first three are commonly used benchmark models in this field. Conversely, the last four are time series prediction models based on deep learning. These provide sufficient technical implementation details. It should be noted that TFFNet was proposed in a highly cited article from the past three years, and DeepIndoorCrowd is a comparative method drawn from another highly relevant study published this year.

HA: The historical average (HA) method uses the average value of the crowd flow volumes during the same period in history as the predicted value of the crowd flow volume.

ARIMA: ARIMA is a classical method for time series prediction. This method includes many variants, such as SARIMA.

SARIMA: SARIMA is a statistical model for learning data with seasonal characteristics. The model considers both temporal closeness and periodicity.

DeepST [23]: The deep-learning-based prediction model for spatial–temporal data (DeepST) has four branches, which are used to learn temporal closeness, periodicity, tendency and external factors, respectively.

ST-ResNet: ST-ResNet has three branches, an architecture similar to that of DeepST and uses residual units as basic blocks in each branch.

TFFNet: TFFNet and DeepST have a similar four-branch structure. However, TFFNet has a deeper model architecture, which is capable of extracting deeper spatial–temporal

dependence and integrating external factors including weather, weekdays and weekends, as well as holidays.

DeepIndoorCrowd: DeepIndoorCrowd comprehensively considers historical temporal features, future temporal features, semantic features of indoor stores and spatial features.

The characteristics of the above models are shown in Table 2.

Table 2. Characteristics of benchmark models.

Model	Spatial Correlation	Time Dependence			External Factors
		Closeness	Periodicity	Tendency	
HA	×	×	√	√	×
ARIMA	×	√	×	×	×
SARIMA	×	√	√	×	×
DeepST	√	√	√	√	√
ST-ResNet	√	√	√	√	√
TFFNet	√	√	√	√	√
DeepIndoorCrowd	√	√	√	√	×

× indicates incapable, √ indicates capable.

To verify the generalization ability of STA-CFPNet, this study used two datasets to test the performance of STA-CFPNet. IDSBJ is 1 month of indoor trajectory data of a shopping mall in Beijing, covering weekdays, weekends and holidays. Due to the small number of indoor crowd flow volume datasets published, we used TaxiBJ [23] as a supplement, which includes taxicabs GPS trajectory data in four time intervals in Beijing. The details of the two datasets are shown in Table 3.

Table 3. Experimental datasets details (holidays include adjacent weekends).

Dataset	IDSBJ	TaxiBJ
Location	Beijing	Beijing
Data type	Indoor trajectory	Taxicab GPS trajectory
Time span	1 to 31 May 2019	1 July 2013–30 October 2013 1 March 2014–30 June 2014 1 March 2015–30 June 2015 1 November 2015–10 April 2016
Time interval	15 min	30 min
Grid size	24 × 100	32 × 32
Sampling rate	60 s	60 s
Crowd flow matrices	2976	22459
Weekdays	23 days	376 days
Weekends	8 days	152 days
Holidays	4 days	41 days
Weather conditions	2 types (e.g., good, poor)	16 types (e.g., sunny, rainy)

We used the root mean squared error (RMSE) as the performance index to measure the prediction accuracy of the model:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - \hat{X}_i)}, \tag{19}$$

where X_i is the true value, $\hat{X}_i$ is the predicted value and N is the total number of samples in the dataset. The smaller RMSE, the better performance of STA-CFPNet.

The principal hardware configurations of the server used for model training are Intel CORE i7-10875H * 1, NVIDIA GTX 2080Ti * 1, 32 GB RAM and 1 TB SSD. We used PyTorch 2.0 to realize STA-CFPNet and the benchmark models, and all benchmark models used the super parameters provided in the original papers as the training super parameters.

4.2. Comparison of Model Prediction Accuracy

We set the following super parameters for STA-CFPNet: initial learning rate of 0.005, learning rate attenuation coefficient of 0.95, weight attenuation coefficient of 0.0001, length of temporal closeness sequence of 8, length of periodicity sequence of 1, and length of tendency sequence of 1; the division ratio of the training dataset and test dataset was 8:2, and the verification dataset was the last 20% of the training dataset. We trained STA-CFPNet and the benchmark models on two training datasets for 100 rounds and all models converged smoothly. The comparison of the models' prediction accuracy and computation time is shown in Table 4.

Table 4. Comparison of models' prediction accuracy and computation time.

Model	RMSE		Training Time (min)		Inference Time (s)	
	IDSBJ	**TaxiBJ**	**IDSBJ**	**TaxiBJ**	**IDSBJ**	**TaxiBJ**
HA	13.2241	218.1444	-	-	-	-
ARIMA	6.8770	102.3351	45.3	151.1	14.80	49.33
SARIMA	7.0633	106.7840	53.0	176.6	16.77	55.91
DeepST	0.3137	46.8211	133.2	388.0	31.72	105.73
ST-ResNet	0.2324	43.7945	112.7	365.6	31.28	102.19
TFFNet	0.1454	29.2633	135.0	424.5	28.74	95.58
DeepIndoorCrowd	0.1857	33.7945	147.9	536.0	33.15	110.52
STA-CFPNet	0.0430	28.1566	130.7	435.5	29.52	98.4

It can be seen from Table 4 that, compared with the use of traditional time sequence prediction models on the two datasets, the accuracy index of STA-CFPNet relative to ARIMA increased by 99.3% and 72.5%, respectively, and the accuracy index of STA-CFPNet relative to SARIMA increased by 99.4% and 73.6%, respectively. STA-CFPNet can learn the implicit spatial correlation and time dependence in the crowd flow matrix at the same time and extract effective auxiliary information from the external factor vector. Due to the above fact, STA-CFPNet greatly surpassed ARIMA and SARIMA in the task of crowd flow volume prediction.

Compared with the current popular deep learning models, STA-CFPNet also has great advantages. For the two datasets, the accuracy index of STA-CFPNet relative to DeepST increased by 86.3% and 39.9%, respectively; the accuracy index of STA-CFPNet relative to ST-ResNet increased by 81.4% and 35.7%, respectively; the accuracy index of STA-CFPNet relative to TFFNet increased by 72.5% and 3.8%, respectively; and the accuracy index of STA-CFPNet relative to DeepIndoorCrowd increased by 76.8% and 16.7%, respectively. STA-CFPNet adds the STATT block to the three branches of temporal closeness, periodicity and tendency, providing calibrated location and channel features for subsequent block learning of the model. Due to the above fact, STA-CFPNet greatly reduces the difficulty of model training and effectively improves the prediction performance of the model. In addition, feature embedding and the extraction of external factor vectors using a fully convolutional structure can reduce the number of learnable parameters of the external factors branch and learn the location-independent external factors contexts for the research area. The output feature maps of the four branches of temporal closeness, periodicity, tendency and external factors are fused via tensor summation, which also improves the performance of the model to a certain extent.

Table 4 shows that the STA-CFPNet is not superior in terms of computational cost. On the contrary, traditional statistics-based crowd flow prediction models require less computation time due to their simple model structures and fewer model parameters. Although deep-learning-based crowd flow prediction models can achieve better results, the computation time of the model is long owing to the complex model structures and large number of model parameters. The question of how to reduce the computational consumption will constitute the focus of our next study.

4.3. Effectiveness Analysis of Model Architecture

For the purpose of studying the impact of structural changes on the performance of the model, we fine-tuned the components of STA-CFPNet to form four variants. Then, we tested these on two training datasets. Each variant adds the STATT block in Stage 0 and Stage 1 of temporal closeness, and brings the periodicity and tendency branches, and uses the fully convolutional structure to embed and extract the features of the external factor vector. Finally, the model adopts tensor summation to fuse the output feature maps of multiple branches. To compare the performance of each model fairly in the crowd flow prediction task, we selected the same super parameters for each variant of STA-CFPNet. The performance of each model variant is shown in Table 5. The model architecture and super parameter settings are shown in Table 6.

Table 5. Model architecture effectiveness analysis results.

Model	RMSE	
	IDSBJ	**TaxiBJ**
STA-CFPNet-CXXX	0.0897	93.3217
STA-CFPNet-CXXE	0.0799	79.6672
STA-CFPNet-CPQX	0.0831	87.3231
STA-CFPNet-CPQE	0.0430	28.1566

Table 6. Model architecture and super parameter settings.

Model	Description	Super Parameter Setting
STA-CFPNet-CXXX	Using temporal closeness branch	Initial learning rate is 0.005, learning rate attenuation coefficient is 0.95 and weight attenuation coefficient is 0.0001; the length of temporal closeness sequence is 8, the length of periodicity sequence is 1 and the length of tendency sequence is 1; the division ratio of training dataset and test dataset is 8:2, and the verification dataset is the last 20% of the training dataset.
STA-CFPNet-CXXE	Using temporal closeness and external factor branches	
STA-CFPNet-CPQX	Using temporal closeness, periodicity and tendency branches	
STA-CFPNet-CPQE	Using temporal closeness, periodicity, tendency and external factor branches	

Table 5 shows that, when tested on the dataset IDSBJ, compared with STA-CFPNet-CXXX, STA-CFPNet-CXXE and STA-CFPNet-CPQX, the accuracy index of STA-CFPNet-CPQE increased by 52%, 46.8% and 48.3%, respectively. When tested on the TaxiBJ dataset, compared with STA-CFPNet-CXXX, STA-CFPNet-CXXE and STA-CFPNet-CPQX, the accuracy index of STA-CFPNet-CPQE increased by 69.8%, 64.7% and 67.8%, respectively. Based on the above analysis, it is possible to establish that STA-CFPNet-CPQE, which integrates temporal closeness, periodicity, tendency and external factor branches, obtains the best prediction accuracy on both datasets. This indicates that integrating the above four branches can effectively learn the spatial correlation, time dependence and randomness introduced by external factors implied in a crowd flow matrix, improving the performance of the model.

Simultaneously, using the IDSBJ dataset, the accuracy index of STA-CFPNet-CPQX improved by 7.4% compared with STA-CFPNet-CXXX, and the accuracy index of STA-CFPNet-CXXE improved by 10.9% compared with STA-CFPNet-CXXX. Using the TaxiBJ dataset, the accuracy index of STA-CFPNet-CPQX improved by 6.4% compared with STA-CFPNet-CXXX, and the accuracy index of STA-CFPNet-CXXE improved by 14.6% compared with STA-CFPNet-CXXX. These results prove that integrating the external factor branch into the model can improve the prediction accuracy better than integrating the periodicity and tendency branches. In particular, when the input length of the periodicity and tendency sequences are short, that is, when working in the short-term crowd flow prediction problem, it is possible to only use the temporal closeness and external factor branches and properly fuse the output feature maps.

4.4. Effectiveness Analysis of STATT Block

To verify the effectiveness of the STATT block proposed in this paper, we fine-tuned the architecture of STA-CFPNet, removing the STATT block from the three branches of temporal closeness, periodicity and tendency, and tested it on two training datasets. With the aim of comparing the performance of each model in the crowd flow prediction task fairly, we selected the same super parameters for each variant of STA-CFPNet. The effectiveness analysis results for the STATT block are shown in Table 7. The model architecture and super parameter settings are shown in Table 8.

Table 7. STATT block effectiveness analysis results.

Model	RMSE	
	IDSBJ	**TaxiBJ**
STA-CFPNet-CPQE-with-STATT	0.0430	28.1566
STA-CFPNet-CPQE-without-STATT	0.0669	68.4889

Table 8. Model architecture and super parameter settings.

Model	Description	Super Parameter Setting
STA-CFPNet-CPQE-with-STATT	Using temporal closeness, periodicity, tendency and external factor branches with STATT block	Initial learning rate is 0.005, learning rate attenuation coefficient is 0.95 and weight attenuation coefficient is 0.0001; the length of temporal closeness sequence is 8, the length of periodicity sequence is 1 and the length of tendency sequence is 1; the division ratio of training dataset and test dataset is 8:2, and the verification dataset is the last 20% of the training dataset.
STA-CFPNet-CPQE-without-STATT	Using temporal closeness, periodicity, tendency and external factor branches without STATT block	

As shown in Table 7, after removing the STATT block of STA-CFPNet-CPQE-with-STATT, the accuracy index RMSE of STA-CFPNet-CPQE-without-STATT on the IDSBJ and TaxiBJ datasets decreased by 35.7% and 58.5%, respectively. Using the STATT block can effectively improve the performance of the model in crowd flow prediction tasks. Furthermore, Figure 9 shows that the STATT block not only improves the performance of the model, but also accelerates the convergence of the model training process. Compared with STA-CFPNet-without-STATT, the training time of STA-CFPNet-with-STATT was greatly reduced under the same training rounds and accuracy index constraints.

STA-CFPNet-CPQE-with-STATT adds STATT blocks before Stage 0 and Stage 1 of the temporal closeness, periodicity and tendency branches. This achieves the feature recalibration of the input data and shallow feature maps and provides more effective information for feature extraction and pattern learning of the subsequent network, reducing the interference of invalid information on the network fitting process in the process of model learning. However, the STATT block introduces additional learnable parameters, increasing the occupation of video memory and training time. Therefore, when using this block, we must consider the impact of location and quantity on the performance of the model.

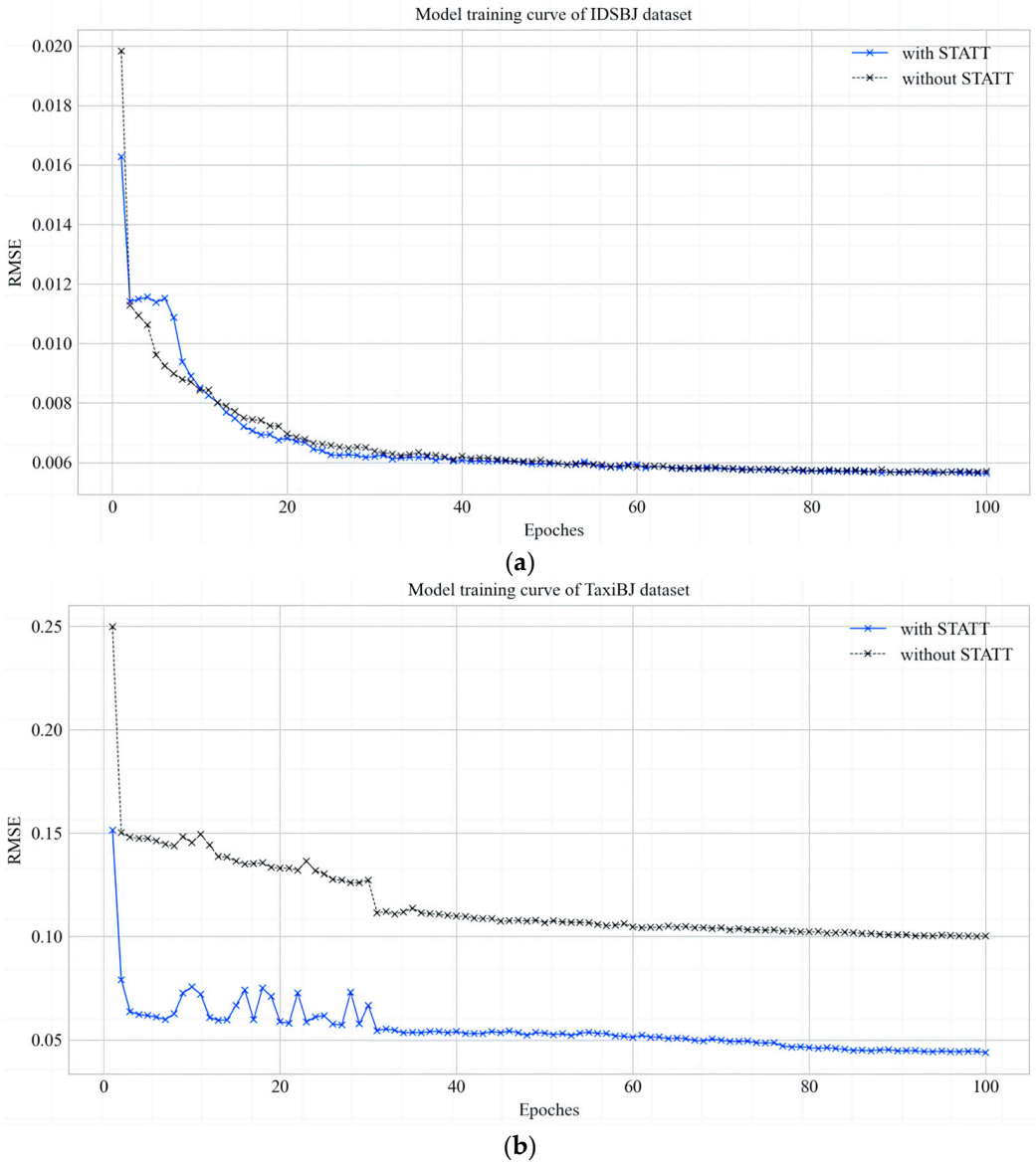

Figure 9. STA-CFPNet training curve. (**a**) Model training curve for IDSBJ dataset; (**b**) model training curve for TaxiBJ dataset.

4.5. Effectiveness Analysis of External Factor Extraction Method

For the sake of verifying the effectiveness of the external factor extraction method, we fine-tuned the architecture of STA-CFPNet, modifying the structure of the external factor branch into a fully convolutional network and fully connected network. Then, we extracted features from the external factor vector in the training dataset and fused them with the other three branches of the model. With the aim of fairly comparing the two external factor extraction methods, we selected the same super parameters for each variant of STA-CFPNet. The effectiveness analysis results for the external factor extraction method are shown in Table 9. The model architecture and super parameter settings are shown in Table 10.

Table 9. External factor extraction method effectiveness analysis results.

Model	RMSE	
	IDSBJ	**TaxiBJ**
STA-CFPNet-CPQE-Conv	0.0430	28.1566
STA-CFPNet-CPQE-FC	0.0918	75.3992

Table 10. Model architecture and super parameter settings.

Model	Description	Super Parameter Setting
STA-CFPNet-CPQE-Conv	Using fully convolutional structure for external factors branch	Initial learning rate is 0.005, learning rate attenuation coefficient is 0.95 and weight attenuation coefficient is 0.0001; the length of temporal closeness sequence is 8, the length of periodicity sequence is 1 and the length of tendency sequence is 1; the division ratio of training dataset and test dataset is 8:2, and the verification dataset is the last 20% of the training dataset.
STA-CFPNet-CPQE-FC	Using fully connected structure for external factors branch	

Table 9 shows that using a fully convolutional structure for feature embedding and extraction of external factor vectors is much better than using a fully connected structure. On the IDSBJ and TaxiBJ datasets, the accuracy index RMSE of STA-CFPNet-CPQE-Conv compared with STA-CFPNet-CPQE-FC improved by 53.2% and 66.6%, respectively.

Compared with the fully connected structure, the fully convolutional structure has fewer parameters, reducing over-fitting problems. Moreover, the existing deep learning framework displays better optimization for convolution operation, which can make the training process more efficient. As shown in Figure 10a, the size of the feature map output by the fully convolutional structure is [1, 1, 1]. There were globally shared external factors contexts for the whole research area, i.e., each position on the output feature map was affected by the same external factor. Figure 10b shows that the output vector size of the fully connected structure was [FC2,]. Then, according to the size of the research area, this was shaped into a feature map with a size of [1, H, W]. The external factor features learned from each location on the feature map did not possess meaningful spatial location attributes, i.e., the external factor features of each location on the feature map were not related to the spatial location, which brings additional complexity to the fusion of subsequent multiple branch output feature maps. Therefore, it is suggested to use a fully convolutional structure in the external factor extraction operation of the model.

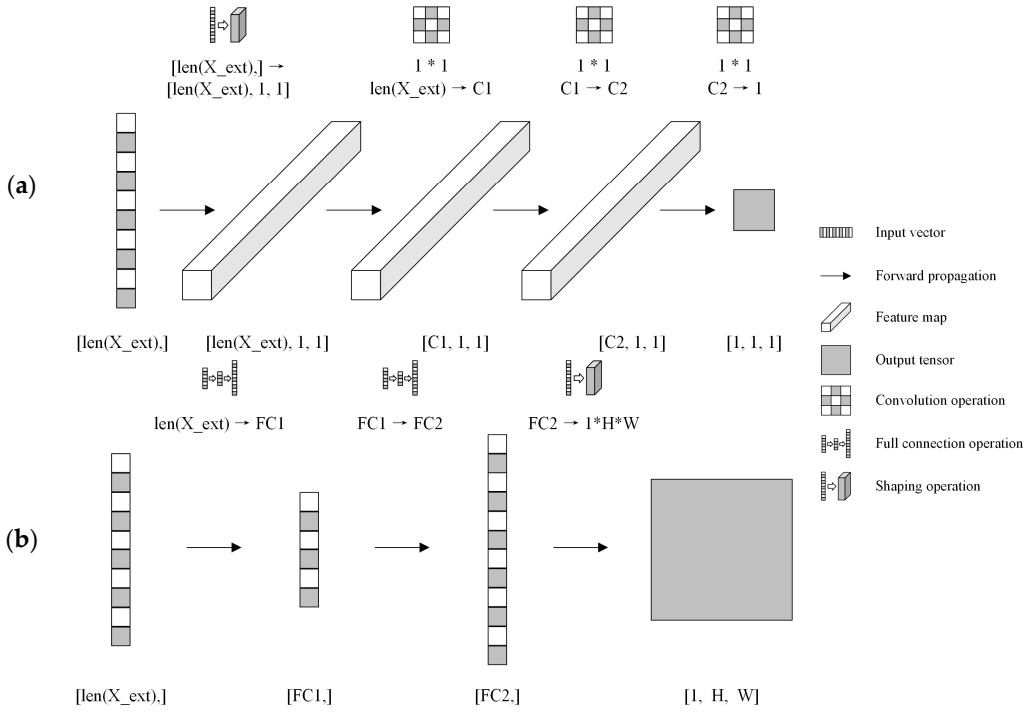

Figure 10. Comparison of fully convolutional structure and fully connected structure. (**a**) Fully convolutional structure; (**b**) fully connected structure.

5. Conclusions

This paper proposes a deep-learning-based crowd flow prediction model for indoor regions named STA-CFPNet. This model can utilize deep convolutional structures to capture spatial correlations from regions near and far. By establishing three branches, namely, temporal closeness, periodicity and tendency, it is possible to model the time dependence implicit in the crowd flow matrix via multi-channel convolution calculations. By designing the STATT block based on the residual structure, we can highlight the data features that are beneficial for spatial–temporal dependence modeling and reduce the interference from invalid information on STA-CFPNet. The fusion of external factors (e.g., weather conditions, holiday arrangements, working days and rest days, etc.) can improve the prediction accuracy of STA-CFPNet. In this research, we evaluated the effectiveness of STA-CFPNet and other benchmarks when applied to the experimental datasets. Additionally, we explored the impacts induced by different model structures and external factor extraction methods. The experimental results indicate that STA-CFPNet exceeds the benchmark models on the experimental datasets in the field of indoor regional crowd flow prediction and verify the effectiveness of using STATT blocks and an external factor extraction method based on a fully convolutional structure.

STA-CFPNet comprehensively considers multiple factors that affect indoor regional crowd flow predictions in order to obtain robust prediction results. However, a large amount of computation is required in the process of generating crowd flow matrices and the training process of the model. As such, we need to introduce high-performance computing methods to solve the problem. In addition, there are currently few publicly available indoor trajectory datasets. This, to some extent, limits research on this issue. In the future, we will consider the application of STA-CFPNet to many other indoor scenarios such as indoor path planning, indoor hotspot prediction and so on.

Author Contributions: Conceptualization, Q.T., S.S. and W.S.; methodology, Q.T. and S.S.; software, S.S.; validation, Q.T. and S.S.; formal analysis, Q.T. and S.S.; investigation, Q.T. and S.S.; resources, W.S., J.B. and C.W.; data curation, S.S.; writing—original draft preparation, Q.T. and S.S.; writing—review and editing, Q.T. and S.S.; visualization, Q.T.; supervision, J.B. and C.W.; project administration, J.B. and C.W.; funding acquisition, W.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China, grant number 42071343.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhang, J.; Zheng, Y.; Sun, J.; Qi, D. Flow prediction in spatio-temporal networks based on multitask deep learning. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 468–478. [CrossRef]
2. Deng, Z.; Yu, Y.; Yuan, X.; Wan, N.; Yang, L. Situation and development tendency of indoor positioning. *China Commun.* **2013**, *10*, 42–55. [CrossRef]
3. Williams, B.M.; Hoel, L.A. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *J. Transp. Eng.* **2003**, *129*, 664–672. [CrossRef]
4. Voort, M.V.D.; Dougherty, M.; Watson, S. Combining kohonen maps with arima time series models to forecast traffic flow. *Transp. Res. Part C Emerg. Technol.* **1996**, *4*, 307–318. [CrossRef]
5. Williams, B. Multivariate vehicular traffic flow prediction: Evaluation of ARIMAX modeling. *Transp. Res. Rec. J. Transp. Res. Board* **2001**, *1776*, 194–200. [CrossRef]
6. Gao, Y.C.; Zhou, C.J.; Rong, J.; Wang, Y.; Liu, S.Y. Short-term traffic speed forecasting using a deep learning method based on multitemporal traffic flow volume. *IEEE Access* **2022**, *10*, 82384–82395. [CrossRef]
7. Zhang, J.; Zheng, Y.; Qi, D.; Li, R.; Yi, X.; Li, T. Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artif. Intell.* **2017**, *259*, 147–166. [CrossRef]
8. Sun, S.Y.; Wu, H.Y.; Xiang, L.G. City-wide traffic flow forecasting using a deep convolutional neural network. *Sensors* **2020**, *20*, 421. [CrossRef] [PubMed]

9. Wang, S.Z.; Miao, H.; Li, J.Y.; Cao, J.N. Spatio-temporal knowledge transfer for urban crowd flow prediction via deep attentive adaptation networks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4695–4705. [CrossRef]
10. Wang, S.Z.; Cao, J.N.; Chen, H.; Peng, H.; Huang, Z.Q. SeqST-GAN: Seq2Seq generative adversarial nets for multi-step urban crowd flow prediction. *ACM Trans. Spat. Algorithms Syst.* **2020**, *6*, 1–24. [CrossRef]
11. Li, F.X.; Feng, J.; Yan, H.; Jin, D.P.; Li, Y. Crowd flow prediction for irregular regions with semantic graph attention network. *ACM Trans. Intell. Syst. Technol.* **2022**, *13*, 1–14. [CrossRef]
12. Sun, J.; Zhang, J.; Li, Q.F.; Yi, X.W.; Liang, Y.X.; Zheng, Y. Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 2348–2359. [CrossRef]
13. Zhang, S.Y.; Deng, B.C.; Yang, D.Q. CrowdTelescope: Wi-Fi-positioning-based multi-grained spatiotemporal crowd flow prediction for smart campus. *CCF Trans. Pervasive Comput. Interact.* **2023**, *5*, 31–44. [CrossRef]
14. Trivedi, A.; Silverstein, K.; Strubell, E.; Lyyer, M.; Shenoy, P. WiFiMod: Transformer-based indoor human mobility modeling using passive sensing. In Proceedings of the ACM SIGCAS Conference on Computing and Sustainable Societies, New York, NY, USA, 28 June–2 July 2021.
15. Chu, C.; Zhang, H.C.; Wang, P.X.; Lu, F. DeepIndoorCrowd: Predicting crowd flow in indoor shopping malls with an interpretable transformer network. *Trans. GIS* **2023**, *27*, 1699–1723. [CrossRef]
16. Tedjopurnomo, D.A.; Bao, Z.; Zheng, B.; Choudhury, F.M.; Kai, Q. A survey on modern deep neural network for traffic prediction: Trends, methods and challenges. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 1544–1561. [CrossRef]
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
18. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
19. Mnih, V.; Heess, N.; Graves, A. Recurrent models of visual attention. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Cambridge, MA, USA, 8–13 December 2014.
20. Lin, M.; Chen, Q.; Yan, S.C. Network in network. *arXiv* **2013**, arXiv:1312.4400.
21. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
22. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
23. Zhang, J.; Zheng, Y.; Qi, D.; Li, R.; Yi, X. DNN-based prediction model for spatial-temporal data. In Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Burlingame, CA, USA, 31–33 November 2016.