STA3DCNN: Spatial-Temporal Attention 3D Convolutional Neural Network for Citywide Crowd Flow Prediction

Gaozhong Tang, Zhiheng Zhou, and Bo Li(✉)

School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510640, China
eegztang@mail.scut.edu.cn, {zhouzh,leebo}@scut.edu.cn

Abstract. Crowd flow prediction is of great significance to the construction of smart cities, and recently became a research hot-spot. As road conditions are constantly changing, the forecasting crowd flows accurately and efficiently is a challenging task. One of the key factors to accomplish this prediction task is how to temporally and spatially model the evolution trend of crowd flows. In previous works, capturing features is carried out mainly by utilizing the structure based on a recurrent neural network which is effective to capture temporal features from time sequence. However, it is inefficient for capturing spatial-temporal features which is critical for the prediction task. In this paper, we develop an elementary module, a 3D convolution layer based on the self-attention mechanism (3DAM), which can extract spatial features and temporal correlation simultaneously. Our proposed spatial-temporal attention 3D convolution prediction network (STA3DCNN) is composed of 3DAMs. Finally, we conduct comparative and self-studying experiments to evaluate the performance of our model on two benchmark datasets. The experimental results demonstrate that the proposed model performs effectively, and outperforms 9 representative methods.

Keywords: Crowd flow prediction · Spatial-temporal features · 3D convolution neural networks · Self-attention mechanism

1 Introduction

Crowd flow prediction is of great significance for developing modern intelligent transportation system (ITS) in smart cities. It aims to predict the changes of crowd distribution in a certain period of time in cities according to the historical distribution of crowds. Accurate and real-time prediction of crowds plays a guiding role in planning the vehicle trajectory, alleviating the crowd congestion, and providing an assistant reference for road construction planning. However, it is still a challenging task due to the difficulty of efficiently fitting the nonlinear characteristics caused by the dynamic temporal and spatial changes of crowd flow.

© Springer Nature Switzerland AG 2021
T. Mantoro et al. (Eds.): ICONIP 2021, LNCS 13111, pp. 353–365, 2021.
https://doi.org/10.1007/978-3-030-92273-3_29

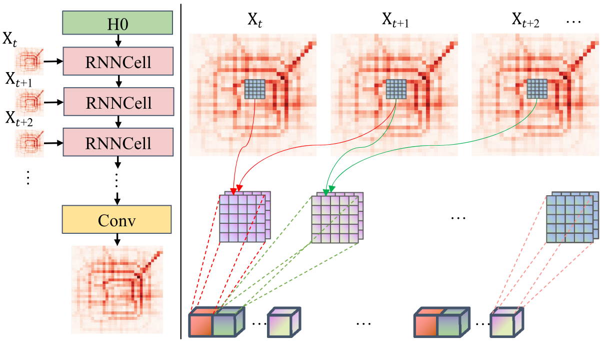

Fig. 1. The left part of the vertical line shows the structure based on RNN, while the right part is the 3DCNN diagram. 3DCNN is able to draw information by sliding convolutional kernels along the time dimension.

In the past few years, many works have been presented on the task of crowd flow prediction. Some traditional time series prediction algorithms and machine learning algorithms have been applied, such as ARIMA [16], linear regression [19], support vector regression [21], and Bayesian Analysis [20]. These models are simple and convenient to be deployed in the task, however, they are not suitable for solving the problem possessing the nonlinear characteristic, *e.g.* crowd flow prediction. With the rising of deep learning, those methods based on deep neural networks (DNNs) with outstanding nonlinear fitting capability were applied to the task of time sequence analysis. Recurrent neural network (RNN) [7] is specially designed for time sequence prediction, and the networks are able to achieve satisfying performance on processing sequential tasks [9,10,12,25]. However, the instability of crowd-flow data caused by dynamic changes in time and space is difficult to be fully fitted by the RNN-based models. Moreover, the said model is at the risk of vanishing gradient, which makes models hard to be trained and converge. RNNs are implemented with a serial structure that is appropriate to capture timing features via step-by-step iterations (as shown on the left part of the vertical line in Fig. 1) whereas the serial structure is short in the computational efficiency.

In our work, a 3D convolutional neural network (3DCNN) based on the self-attention mechanism (SAM) is introduced to remedy the defect mentioned above. As shown on the right part of the vertical line in Fig. 1, 3DCNN can extract temporal features by sliding convolution kernel along the time dimension, while the spatial feature can also be aggregated by the receptive field of convolution kernels. It is more efficient to extract the spatial feature and the temporal features simultaneously from dynamic crowd flows. In addition, 3DCNN can extract features of the entire input time series using a short network connection, which is utilized to alleviate the problem of vanishing gradient. Relying on the self-attention mechanism, our model can build a spatial connection between two regions at a long distance, meanwhile, the temporal feature of the entire input time sequence can be captured.

In this paper, we propose a spatial-temporal attention network based on a 3D convolutional neural network (STA3DCNN) to predicting crowd flows. The specific works of this paper include: i) We introduce 3DCNN and SAM to solve the problem of crowd flow prediction. 3D convolution layer and SAM are cascaded together as an elementary ingredient, 3DAM, which will be used to form modules to extract the spatial-temporal feature from various scales. ii) In order to capture the temporal feature sufficiently, we consider the feature in two modes including the high correlation features of adjacent time and cyclical pattern features in crowd flows. Then, we implement a network with two branches by employing 3DAM, which aims to automatically capture the spatial-temporal evolution features and the stable spatial-temporal cycle pattern features hidden in crowd flows, respectively. iii) We implement a bi-modal fusion module (BFM) to fuse evolution features and cyclical pattern features, and to accomplish the final prediction. The results of the experiments prove that our fusion method performs better than the baseline fusion method.

In summary, the main contributions of our work can be summarized as below:

1) We introduce 3DCNN and self-attention mechanism into the crowd flow prediction framework. Our model can efficiently extract spatial-temporal features.
2) We model crowd-flow data based on the time correlation and the cyclical pattern, while a bi-mode features fusion module is designed to merge spatial-temporal features of different modes.
3) We propose a crowd flow prediction model, STA3DCNN, and conduct experiments on two benchmark datasets to demonstrate that our model is feasible, efficient, and accurate on predicting the trend of crowd flows.

2 Related Work

In the past few years, a lot of works have been proposed on the task of crowd flow prediction. Traditional prediction methods, including linear regression [19], support vector regression [21], ARIMA [5], Bayesian model [20], are easily implemented. However, the prediction accuracy of these models is hard to satisfy the expectation, due to these models is not well fitted the nonlinear characteristic of crowd-flow data.

In recent years, deep learning shows an outstanding ability to fit nonlinear data and has the capacity on digging the latent information from data. Methods built on deep learning have been successfully utilized in time series forecasting tasks. LSTM [14] and GRU [3] have been used to extract temporal features for crowd flow prediction. By modifying the LSTM cell, Liu *et al.* [11] introduced the attention mechanism into the method and implemented an attentive crowd flow machine (ACFM) which can adaptively exploit diverse factors affecting changes in crowd flows. Do *et al.* [4] employed the attention mechanism to design a new model which consists of an encoder and a decoder based on GRU. Inspired by densely connected networks, Xu *et al.* [22] used historically dense structures to analyze historical data, and then used two serial LSTM units for the temporal feature extraction and prediction.

The spatial feature is vital for predicting the trend of crowd flows. Graph neural network (GNN) is a popular way to extract spatial features of crowd flows [1,23,27]. In previous works, the transportation network can be considered as a graph composed of nodes (areas) and edges (roads). For example, Zhao *et al.* [27] integrated the graph convolution operation into GRU cells, and extracted spatial information while performing time-dependent extraction.

Although many impressive works have been presented in the literature, there is still a challenge to capture the dynamic temporal and spatial characteristics of crowd flows regarding balance effectiveness and efficiency. In our work, we propose a crowd flow prediction method based on DNNs, STA3DCNN, which can capture spatial-temporal features effectively and efficiently.

3 Preliminary

In this section, we briefly introduce the data processing methods, preliminary concepts, and the definition of the problem of crowd flow prediction.

Region Partition. Crowds may move from one area to another along the road in a city, such as from residential areas to central business districts. Because the city may contain hundreds of thousands of road links, modeling the changes of crowd flow based on road links is too complicated to accomplish. Then, the city can be partitioned into grids, and we analyze the crowd-traffic flow in each grid. Similar to previous study [26], we divide a city's road map into $h \times w$ small areas along the longitude and the latitude, by which data can be easily processed by models. We make a compromise between the calculation amount and the prediction granularity in city partition, so that the regional crowd flow density can be better modeled.

Crowd Flow Prediction Problem. We count the flow of people entering and leaving each area within a period of time as the research object. $X_t^d \in R^{2\times h\times w}$ denotes the change of crowd distribution at the time period t on day d, where the 1st channel represents the inflow, and the 2nd one represents the outflow. The task of crowd flow prediction is to estimate the quantity of crowd flows which will enter and leave an area according to historical crowd-flow data. Then, the problem of crowd flow prediction can be transformed into the below problem: Given sequence $[X_{t-k}^d, X_{t-k+1}^d, \dots, X_{t-1}^d]$ of previous k time slots, to predict the crowd-flow map X_t^d at the t_{th} time interval of d_{th} day.

4 Methodology

Figure 2 shows the architecture of STA3DCNN which is designed from 3 perspectives: 1) Regarding dynamic spatial-temporal features, we implement an elementary module, 3DAM, which is comprised of a 3D convolutional layer and SAM.

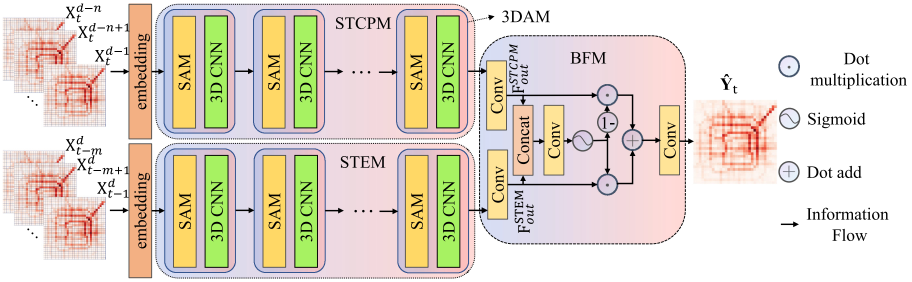

Fig. 2. Overview of the spatial-temporal 3D convolutional neural network (STA3DCNN).

3DAM can make the model pay more attention to the temporal feature among closely correlated time steps and the spatial dependence hidden in highly relevant regions. 2) 3DAM is used to form two modules which are the spatial-temporal evolution feature extraction module (STEM) and the spatial-temporal cyclical patterns feature extraction module (STCPM). The two modules work in a parallel way to extract evolution features and cyclical pattern features, respectively. It should be noticed that the spatial information can be implanted in features due to the operation characteristics of 3DAM. 3) A bi-modal feature fusion and prediction module (BFM) is implemented to merge the spatial-temporal evolution feature and spatial-temporal cyclical patterns feature by BFM and accomplish the final prediction.

4.1 3D Convolutional Neural Networks Combining with Self-Attention Mechanism

3DAM aims to fully capture the temporal feature and the spatial features of crowd flow. As shown in Fig. 3, 3DAM is composed of SAM and a 3D convolutional layer. Different from RNNs which can learn the sequence information according to the order of feeding data, the 3D convolutional network can not completely model the sequential relationship implied in the whole input time series. All of the temporal features are calculated within the receptive field of the convolution kernel in parallel, however, the model is hard to distinguish the temporal relationship between different feature channels. Hence, we implement SAM to extract the feature of the sequential relationship. In addition, the spatial feature between distant regions can be captured simultaneously by the module.

Here, SAM is expressed as

$$F_{out}^{seq} = F_{in}^{seq} \cdot f_{sig}(f_{up}(f_{pool}(F_{in}^{seq} * w_1))) \tag{1}$$

where F_{in}^{seq} denotes the input of SAM, F_{out}^{seq} denotes the output of SAM, w_1 denotes parameters of convolution kernel with kernel size 3 × 3 followed by an activation unit (ReLU), f_{sig} denotes a sigmoid operation, and f_{up} and f_{pool} respectively denote up-sampling operation and pooling operation with kernel

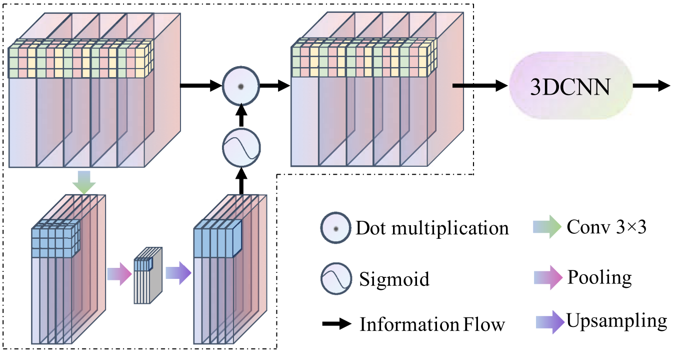

Fig. 3. Overview of our elementary module 3DAM. The part surrounded with dash line is the SAM, while the part 3DCNN accomplish the extraction of the spatial-temporal feature.

size 4×4. By updating weights of the above structure, the spatial-temporal can be captured, so as to extract effective information for crowd flow prediction.

Combining with a 3D convolutional layer, the module can be used to fully extract the spatial-temporal features. 3DCNN can extract the dependence of adjacent time through sliding convolution kernels along the time dimension. By adjusting the size of the convolution kernel, the receptive field in the time dimension is also changed. Then, with the 3D convolution kernel sliding along the time dimension with various strides, we can extract features of different time scales. The size of the convolution kernel in spatial dimension determines its spatial receptive field which can promote the capability of 3DCNN to extract spatial features by setting spatial receptive field. Thus, 3DAM can simultaneously capture temporal features and spatial features of crowd flows. We set the 3D convolutional layer with kernel size $2 \times 3 \times 3$.

4.2 Spatial-Temporal Feature Extraction

As shown in Fig. 2, STEM and STCPM are utilized to extract the spatial-temporal features. The two modules can model the change of crowd flow from two various perspectives. The closer the data is in the time dimension, the stronger the correlation is between the data. Then, STEM is designed to capture such trends in short-term and features with high correlation. Moreover, people follow a relatively fixed travel schedule during the day in urban. For example, crowd flow has two traffic peaks in the morning and evening, corresponding to the beginning and the ending of production activities. The distribution of crowd flow can be guided by the changes of crowd flow in the same period dating back to the past few days due to the cyclical change. STCPM can extract the pattern features of this cyclical change, which can provide varying perspectives for prediction.

STEM. This module is implemented to extract evolution features of crowd flow in a short period of time. Regarding the change of crowd flows is continuously evolving with time, we use the crowd-flow data of several past and adjacent moments as inputs of STEM. We utilize a series of stacked 3DAMs to capture the evolutionary change. Crowd-flow data is firstly encoded by the embedding module into 96 channels aiming to improve the diversity of feature expression. Then, the dynamic spatial-temporal features of crowd flows are extracted by the cascaded 3DAMs. The features of different time intervals have different implicit associations with the current time features, and the self-attention module in 3DAM can aggregate features from the hidden relationship. Then, 3DCNN layers are employed for spatial-temporal features extraction. The output of STEM, $F_{out}^{STEM} \in R^{96 \times h \times w}$, encodes the spatial-temporal evolution features of crowd-flow data.

STCPM. The module is used to extract cyclical patterns of crowd flows. Due to the change of producing activities and daily life in the city, the spatial and temporal distribution of people's travel trajectory often exhibit a certain regularity and periodicity. Therefore, we set weeks as the cycle of the cyclical pattern, and we acquire data at the same period of each day which is selected from the past more than a week. The data can provide enough information for our model to capture the cyclical pattern which varies from day to day within a cycle. The historical data of the same period is firstly encoded by the embedding module aiming to enlarge the amount of channels. We pass the encoded features through SAM to automatically emphasize those features with higher similarity. Then, 3DCNN in 3DAM fulfills the extraction of spatial-temporal features from crowd-flow data. The outputs of STCPM is denoted as $F_{out}^{STCPM} \in R^{96 \times h \times w}$ which encodes the cyclical patterns of crowd flow, and will be fused with F_{out}^{STEM} in the subsequent bi-mode fusion module for the final prediction.

4.3 Bi-Modal Fusion Module

After we obtained the two features, the temporal evolution, and the cyclical pattern, we can generate the final prediction by fusing features. As shown in Fig. 2, the bi-modal fusion module (BFM) is implemented to merge the two features and accomplish the final prediction. We use the self-weighting method to automatically learn the weights of the features extracted from the two modules. Specifically, we calculate the weights by concatenating F_{out}^{STEM} and F_{out}^{STCPM} together, and pass the feature through a convolution layer and a sigmoid function. BFM can be represented as:

$$F = f_{sig}(w_2 * [F_{out}^{STEM}, F_{out}^{STCPM}]), \tag{2a}$$

$$Y = w_3 * (F_{out}^{STEM} \odot F + F_{out}^{STCPM} \odot (1 - F)) \tag{2b}$$

where Y denotes the final crowd flow prediction, w_2 and w_3 denote the weights of the convolutional layer, f_{sig} denotes the sigmoid function, and '$\odot$' denotes dot multiplication. Processed by BFM, the features of temporal evolution and cyclical pattern are fully integrated, which can enhance the performance of the prediction network.

5 Experiments

We comprehensively evaluate STA3DCNN in three perspectives: performance evaluation, efficiency evaluation, and self-studying tests. We adopt PyTorch [13] toolbox to implement the proposed method. Our model is compared with 9 representative methods presented in recent years, including 3 traditional time-series analysis models, and 6 deep learning-based methods, in which two GNN-based prediction methods are included. The performance evaluation we made is the next-step prediction. The efficiency evaluation is performed on estimating the RAM (random access memory) consumption and the operating efficiency. Via self-studying experiments, we analyze the effectiveness of each component of the proposed model.

5.1 Datasets and Evaluation Criteria

Datasets. Our method has been evaluated on two public benchmark datasets: TaxiBJ dataset [25] and BikeNYC dataset [26]. The crowd-flow maps are generated from the trajectory data in the two datasets by the method presented in Sect. 3. Similar to previous works, we use the last four weeks in TaxiBJ and the last ten days in BikeNYC as a test set, while the rest data of the two datasets are used for training models.

Evaluation Criteria. We evaluate the performance of the proposed method by two popular evaluation metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). Specifically, their definitions are respectively shown as below:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\widehat{Y}_i - Y_i\right)^2}, \tag{3a}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|\widehat{Y}_i - Y_i\right| \tag{3b}$$

where $\widehat{Y}_i$ and Y_i respectively represent the predicted crowd-flow map and its ground truth, and n denotes the number of crowd-flow maps used for validation.

5.2 Experimental Setup

In our work, the length of sequence inputted into the entire model is 16, while the length of sequence fed into STEM and STCPM are set to 8, respectively. The size of one minibatch is set to 128, and the initial learning rate in our model is

10^{-3}. Weights of all convolutional layers are initialized according to Kaiming's work [6]. We optimize the parameters of our network via Adam optimization [15] by minimizing mean square error loss with a GTX Titan Xp GPU in an end-to-end manner.

5.3 Experiment Results and Analysis

Evaluation on Crowd Flow Prediction. We apply STA3DCNN to predict the next-step crowd flow, meanwhile we make a comparison with 9 representative methods: Historical Average (HA) [18], Auto-Regressive Integrated Moving Average (ARIMA) [16], XGBoost [2], Convolution LSTM (ConvLSTM) [17], Spatio-temporal graph convolutional neural network (STGCN) [24], Spatial-Temporal Dynamic Network (STDN) [23], Context-Aware Spatial-Temporal Neural Network (DeepSTN+) [10], Spatial-temporal Graph to Sequence Model (STG2Seq) [1], and Dual Path Network (DPN) [8].

Table 1 shows the performance of our method and the 9 representative methods on the two benchmark datasets. The first 3 methods in Table 1 are traditional time series prediction methods, and the rest are deep learning-based models. Compared with models based on deep learning, the traditional models are hard to achieve satisfactory performance, because the traditional models are inadequate to fit nonlinear data. Among the deep learning-based models, our STA3DCNN achieves the best results on both two datasets. Specifically, compared with the best performing method, RMSE of our model is respectively decreased by 3.04% and 3.47% on the two datasets, while MAE is decreased by 5.45% and 0.88%, respectively. It is should be noticed that STGCN and STG2Seq

Table 1. Evaluation of crowd flow prediction on TaixBJ and BikeNYC datasets. The top-2 results are highlighted in red and blue, respectively.

Models	TaxiBJ		BikeNYC		Number of Para.	Infer. Time
	RMSE	MAE	RMSE	MAE	(Megabytes)	(Sec.)
HA [18]	57.79	-	21.57	-	-	**0.04**
ARIMA [16]	22.78	-	10.07	-	-	110
XGBoost [2]	22.93	-	6.91	-	-	3.45
ConvLSTM [17]	18.79	11.46	6.6	2.44	**0.76**	7.74
STGCN [24]	19.10	11.57	4.76	2.44	6	3.10
STDN [23]	17.83	**9.90**	5.79	2.41	37.8	1.49
DeepSTN+ [10]	17.65	10.03	4.96	2.31	1074	4.84
STG2Seq [1]	17.60	10.47	**4.61**	**2.28**	23.04	13.84
DPN [8]	**16.80**	-	-	-	-	-
STA3DCNN	**16.29**	**9.36**	**4.45**	**2.26**	**3.1**	**1.28**

are built upon GNN for spatial feature extraction. Comparatively, STA3DCNN employing 3DCNNs to extract spatial features is more effective.

Computational Efficiency of STA3DCNN. The efficiency evaluation is fulfilled in two perspectives: calculating the RAM (random access memory) consumption and estimating the operating efficiency. The former one is evaluated by counting the parameter of models, while the latter one is evaluated by estimating how long it is needed to infer and generate the final prediction. The experimental results are shown in the last two columns of Table 1.

We evaluate the model based on deep learning only due to, generally, the model built upon deep learning contains massive parameters. From the penultimate column of Table 1, it can be observed that the amount of parameters of ConvLSTM, STGCN, and our method are far less than the rest three models. Although ConvLSTM has the fewest parameters, our model surpasses a large margin in operating efficiency. In all models, the operating efficiency of HA is best, while our model is the most efficient among deep learning-based models. The inference time of STDN is slightly higher than our model, whereas the parameter quantity of STDN is more than ten times that of our model. Although the performance of STA3DCNN on the two efficiency criteria is not the best, the comprehensive performance is higher than the rest models.

Table 2. Evaluation of each component composing STA3DCNN. ‘w/o’ in each sub-test means ‘without’.

Module	TaxiBJ		BikeNYC	
	RMSE	MAE	RMSE	MAE
STA3DCNN-w/o-STEM	43.23	21.40	7.04	3.24
STA3DCNN-w/o-STCPM	17.88	9.93	4.69	2.52
STA3DCNN-w/o-BFM	16.89	9.81	4.52	2.35
STA3DCNN-w/o-Att	16.45	9.49	4.50	2.31
STA3DCNN	16.29	9.36	4.45	2.26

Ablation Study. We verify the effectiveness of each component forming STA3DCNN by self-studying. The model is comprised of 4 key components: 3DAM, STEM, STCPM, and BFM. Among them, STEM and STCPM are two spatio-temporal feature extraction branches. Therefore, we verify each component via 4 sub-tests: 1) STA3DCNN-w/o-STEM: Verifying the cyclical patterns feature captured by STCPM to predict crowd flow without evolution features; 2) STA3DCNN-w/o-STCPM: Evaluating the evolution features captured by STEM without cyclical patterns; 3) STA3DCNN-w/o-BFM: Verifying BFM by fusing the features extracted by STEM and STCPM with the same weight; 4) STA3DCNN-w/o-att: Verifying the self-attention module in 3DAM.

The results of ablation tests on each component are shown in Table 2, from which we can observe: 1) Both STEM and STCPM are effective on this prediction task, which proves the effectiveness of evolution features and cyclical pattern features. Satisfactory performance is hard to be achieved by only using one of the two features. Besides, the contribution of evolution features to the task is greater than that of cyclical patterns features. 2) Compared with the fusion pattern via simply averaging, BFM can integrate features more effectively. Specifically, BFM can reduce RMSE by 3.55% and MAE by 4.59% on the TaxiBJ dataset, while RMSE and MAE are respectively decreased by 1.57% and 3.83% on the BikeNYC dataset. 3) The result of the STA3DCNN-w/o-att test demonstrates that SAM in our 3DAM module can effectively capture temporal and spatial features.

6 Conclusion

In this paper, we propose a spatial-temporal attention 3D convolutional neural network for the task of crowd flow prediction. Instead of utilizing RNN structure to extract features, our work introduces a more effective and efficient structure based on 3DCNN and the self-attention mechanism. We firstly implement the 3DAM module which served as an elementary module. Then, by using 3DAM, we implement the extraction module for the evolution features of the crowd flow and the cyclical pattern features. Finally, a feature fusion module is implemented to fuse features. The experimental results demonstrate that our method is effective and efficient in predicting crowd flows, and performs better than 9 representative models. The results of the ablation study demonstrate that the modules forming STA3DCNN are effective, and can incrementally improve the performance of the proposed model. In the future, we will explore the interpretability of spatio-temporal features from the actual physical meaning of the region for further improving the performance of the method.

Acknowledgment. This research was supported by National Key R&D Program of China (No. 2017YFC0806000), National Natural Science Foundation of China (No. 11627802, 51678249, 61871188), State Key Lab of Subtropical Building Science, South China University of Technology (2018ZB33), and the State Scholar- ship Fund of China Scholarship Council (201806155022).

References

1. Bai, L., Yao, L., Kanhere, S.S., Wang, X., Sheng, Q.Z.: Stg2seq: spatial-temporal graph to sequence model for multi-step passenger demand forecasting. In: IJCAI 2019 (2019)
2. Chen, T., Guestrin, C.: Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (2016)
3. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NIPS 2014 Workshop on Deep Learning, December 2014 (2014)

4. Do, L.N., Vu, H.L., Vo, B.Q., Liu, Z., Phung, D.: An effective spatial-temporal attention based neural network for traffic flow prediction. Transp. Res. Part C Emerg. Technol. **108**, 12–28 (2019)
5. Hamed, M.M., Al-Masaeid, H.R., Said, Z.M.B.: Short-term prediction of traffic volume in urban arterials. J. Transp. Eng. **121**(3), 249–254 (1995)
6. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1026–1034 (2015). https://doi.org/10.1109/ICCV.2015.123
7. Jain, L.C., Medsker, L.R.: Recurrent Neural Networks: Design and Applications, 1st edn. CRC Press Inc, USA (1999)
8. Li, H., Liu, X., Kang, Y., Zhang, Y., Bu, R.: Urban traffic flow forecast based on dual path network. In: Journal of Physics: Conference Series, vol. 1453, p. 012162 (2020)
9. Li, W., Wang, J., Fan, R., Zhang, Y., Guo, Q., Siddique, C., Ban, X.J.: Short-term traffic state prediction from latent structures: accuracy vs. efficiency. Transp. Res. Part C Emerg. Technol. **111**, 72–90 (2020). https://doi.org/10.1016/j.trc.2019.12.007
10. Lin, Z., Feng, J., Lu, Z., Li, Y., Jin, D.: Deepstn+: context-aware spatial-temporal neural network for crowd flow prediction in metropolis. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 1020–1027 (2019)
11. Liu, L., et al.: Dynamic spatial-temporal representation learning for traffic flow prediction. IEEE Trans. Intell. Transp. Syst. **22**, 7169–7183 (2020)
12. Liu, Y., Blandin, S., Samaranayake, S.: Stochastic on-time arrival problem in transit networks. Transp. Res. Part B Method. **119**, 122–138 (2019). https://doi.org/10.1016/j.trb.2018.11.013
13. Paszke, A., et al.: Pytorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems, pp. 8026–8037 (2019)
14. Sak, H., Senior, A., Beaufays, F.: Long short-term memory recurrent neural network architectures for large scale acoustic modeling. Comput. Sci. **29**, 338–342 (2014)
15. Sharma, M., Pachori, R., Rajendra, A.: Adam: a method for stochastic optimization. Pattern Recogn. Lett. **94**, 172–179 (2017)
16. Shekhar, S., Williams, B.M.: Adaptive seasonal time series models for forecasting short-term traffic flow. Transp. Res. Rec. **2024**(1), 116–125 (2007)
17. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., kin Wong, W., chun Woo, W.: Convolutional lstm network: a machine learning approach for precipitation nowcasting. In: NIPS 2015 Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, vol. 28, pp. 802–810 (2015)
18. Smith, B.L., Demetsky, M.J.: Traffic flow forecasting: comparison of modeling approaches. J. Transp. Eng. **123**(4), 261–266 (1997)
19. Sun, H., Liu, H.X., Xiao, H., He, R.R., Ran, B.: Use of local linear regression model for short-term traffic forecasting. Transp. Res. Rec. **1836**(1836), 143–150 (2003)
20. Wang, J., Deng, W., Guo, Y.: New bayesian combination method for short-term traffic flow forecasting. Transp. Res. Part C Emerg. Technol. **43**, 79–94 (2014)
21. Wu, C.H., Ho, J.M., Lee, D.: Travel-time prediction with support vector regression. IEEE Trans. Intell. Transp. Syst. **5**(4), 276–281 (2004)
22. Xu, L., Chen, X., Xu, Y., Chen, W., Wang, T.: ST-DCN: a spatial-temporal densely connected networks for crowd flow prediction. In: Shao, J., Yiu, M.L., Toyoda, M., Zhang, D., Wang, W., Cui, B. (eds.) APWeb-WAIM 2019. LNCS, vol. 11642, pp. 111–118. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26075-0_9

23. Yao, H., Tang, X., Wei, H., Zheng, G., Li, Z.: Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 5668–5675 (2019)
24. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18 (2018)
25. Zhang, J., Zheng, Y., Qi, D.: Deep spatio-temporal residual networks for citywide crowd flows prediction. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
26. Zhang, J., Zheng, Y., Qi, D., Li, R., Yi, X.: Dnn-based prediction model for spatio-temporal data. In: Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 1–4 (2016)
27. Zhao, L., et al.: T-GCN: a temporal graph convolutional network for traffic prediction. IEEE Trans. Intell. Transp. Syst. **21**, 3848–3858 (2019)