Traffic Parameters Prediction Using a Three-Channel Convolutional Neural Network

Di Zang, Dehai Wang, Jiujun Cheng, Keshuang Tang, Xin Li

Traffic Parameters Prediction Using a Three-channel Convolutional Neural Network

Di Zang[1,2], Dehai Wang[1,2], Jiujun Cheng[1,2], Keshuang Tang3, Xin Li[4]

[1] Department of Computer Science and Technology,
Tongji University, Shanghai, China
[2] The Key laboratory of Embedded System and Service Computing, Ministry of Education,
Tongji University, Shanghai, China
[3] Department of Transportation Information and Control Engineering, Tongji University,
Shang-hai, China
[4] Shanghai Lujie Electronic Technology Co., Ltd., Pudong, Shanghai, China
zangdi@tongji.edu.cn

Abstract. Traffic three elements consisting of flow, speed and occupancy are very important parameters representing the traffic information. Prediction of them is a fundamental problem of Intelligent Transportation Systems (ITS). Convolutional Neural Network (CNN) has been proved to be an effective deep learning method for extracting hierarchical features from data with local correlations such as image, video. In this paper, in consideration of the spatiotemporal correlations of traffic data, we propose a CNN-based method to forecast flow, speed and occupancy simultaneously by converting raw flow, speed and occupancy (FSO) data to FSO color images. We evaluate the performance of this method and compare it with other prevailing methods for traffic prediction. Experimental results show that our method has superior performance.

Keywords: Deep Learning; Convolutional Neural Network; Traffic Prediction; Intelligent Transportation System

1 Introduction

Currently, real time traffic information prediction has got more and more attention of individual drivers, business sectors and governmental agencies with the increasing numbers of vehicles and the development of ITS. Flow, speed and occupancy as three elements of traffic describe the different characteristics of traffic and record the spatiotemporal evolution over a period of time. Accurate prediction of them can facilitate people's travel and reduce traffic congestion and accidents. Meanwhile, it can help traffic managers allocate traffic resources systematically and improve regulatory efficiency.

There exist the spatiotemporal correlations of traffic data due to the consecutive evolution of traffic state on the time-space dimension. It's necessary to retain and leverage inherent spatiotemporal correlations when forecasting traffic information. In addition, there are undoubtedly inner correlations between traffic parameters such as flow, speed and occupancy within a time unit, most of which have been explained

theoretically and mathematically. Intuitively, when traffic flow is high, speed usually won't be low. Therefore, it's reasonable to take these important correlations into account in the modeling to make the prediction more robust.

Existing traffic prediction methods can be mainly divided into three categories: time-series methods, nonparametric methods and deep leaning methods.

Autoregressive moving average (ARIMA) [1] model is one of the representative time-series models, which focuses on finding the patterns of the temporal evolution formation by two steps: moving average (MA) and autoregressive (AR) considering the essential traffic information characteristics. Many variants of this model, such as subset ARIMA, seasonal ARIMA, KARIMA, ARIMAX, were proposed to improve prediction accuracy. However, these models are inept at extracting spatiotemporal feature for prediction because of ignoring spatiotemporal correlations.

Nonparametric methods were widely used because of its advantages, such as their ability to deal with multi-dimensional data, implementation flexibility. In [2], a prediction model was built by support vector regression (SVR) and optimized by particle swarm algorithm. Chang et al proposed a dynamic multi-interval traffic flow prediction using k-nearest neighbors (KNN) [3]. In addition, various artificial neural network (ANNs) [4-5] were designed to predict traffic information. Unfortunately, these shallow architectures failed to learn deep features.

Recently, deep learning models such as deep belief network (DBN) [6], recurrent neural network (RNN) [7], and long short-term memory (LSTM) [8] have been used for traffic prediction owe to their excellent capability of extracting complex features and generalizability and strong forecasting performance. However, these models usually put the time and space into same dimension and they violate the two-dimensional basis of spatiotemporal features. CNN has been demonstrated to have excellent performance in large-scale image-processing tasks [9]. Ma et al. [10] first applied CNN to traffic speed prediction by converting network traffic to gray images and achieved significant improvement on average accuracy.

In this paper, we propose a CNN-based method to forecast traffic information from a comprehensive perspective. We design a novel way in which quite complete spatiotemporal features can be extracted from the data by converting raw flow, speed and occupancy (FSO) data to FSO images. Furthermore, we use the features to forecast the traffic flow, speed, occupancy simultaneously. Finally, we demonstrate the performance of the proposed model and compare it with other prevailing methods.

The rest of the paper is organized as follows: In section 2, converting raw FSO data to FSO images and CNN model for traffic prediction are introduced. In Section 3, experiments and results are detailed. Finally, conclusions are drawn in section 4.

2 Methodologies

Raw FSO data with space and time dimensions can be integrated and converted to FSO color images as shown in Fig.1. Then a CNN model based on LeNet-5 architecture is designed to learn the mapping relationship between spatiotemporal features and FSO images. Finally, we compute predictive performance indicators by mapping the learned features back to original FSO data space.

2.1 Data conversion

Traffic data such as flow, speed, occupancy collected by sensors such as inductive loop detectors at a certain time interval, which is usually not more than 5 minutes, record the evolution of traffic conditions in a particular region.

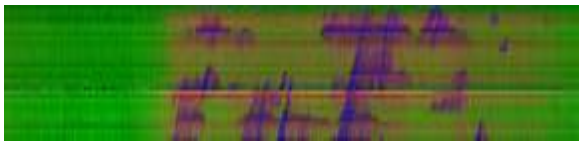

Fig. 1. FSO image converted from FSO data of a day. The area containing purple blocks corresponds to the period when the traffic volume is large i.e. day time whereas the green area corresponds to the opposite side.

For the width of the image, time sequences are fitted linearly into the width-axis chronologically. Thus, the length of time interval determines the width of the image. For a 5-minute interval, there will be 288 data sequences on the time dimension corresponding to the image width of 288 pixels. In general, short intervals like 5 seconds are meaningless for traffic prediction. Therefore, in most cases data sequences need to be aggregated to generate available data by dealing with several adjacent time intervals.

For the height of the image, we simply fit the number of sensors ordered spatially into the height-axis. We can also make height-axis compact and informative, in the same way as width dimension, by aggregating data from serval adjacent sensors. Notably, different traffic data may use different aggregation methods. For example, flow use accumulation while speed and occupancy use mean.

Finally, three time-space matrix generated by the method mentioned above are directly merged into FSO color images for flow as green channel, speed as red channel and occupancy as blue channel. A FSO image can be denoted as:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1N} \\ p_{21} & p_{22} & \cdots & p_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ p_{M1} & p_{M2} & \cdots & p_{MN} \end{bmatrix} \quad (1)$$

where M is the number of sensors, N is the length of time units and pixel p_{ij} is a triple consisting of flow, speed and occupancy.

2.2 CNN architecture

Fig.2 illustrates the proposed three-channel convolutional neural network model. In this case, traffic data of flow, speed and occupancy are respectively encoded into three channels corresponding to the red, green and blue color channels. These three traffic parameters are then learned by different convolutional kernels to generate feature maps

which represent the fusion of traffic information. After the consecutive processing of convolution and pooling, the fully-connected layer maps the extracted traffic features back to the original space for final prediction as the output of the model.

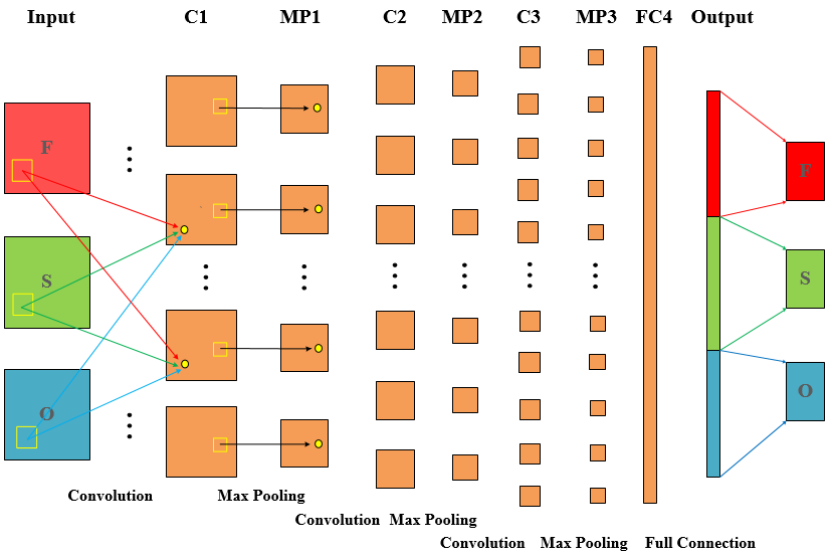

Fig. 2. Proposed CNN model for extracting traffic spatiotemporal features. C, MP and FC represent convolution, max pooling and fully-connected layer, respectively.

2.2.1 Input layer and output layer

First, raw FSO data are converted to FSO images. However, different from CNN models for classification, for an FSO image as input, the output of the model is also a FSO image right next to it. This means that the network not only learns hierarchical features but also learns to map the extracted features back to original space, which is similar to the mechanism of the autoencoder (AE).

Given the lengths of input and output time units as τ and ε, respectively and prediction interval between input and output as π, which is set to 1 in this paper, the input of the model can be written as:

$$x^i = \begin{bmatrix} p_{1,i} & p_{1,i+1} & \cdots & p_{1,i+\tau-1} \\ p_{2,i} & p_{2,i+1} & \cdots & p_{2,i+\tau-1} \\ \vdots & \vdots & \ddots & \vdots \\ p_{M,i} & p_{M,i+1} & \cdots & p_{M,i+\tau-1} \end{bmatrix} \tag{2}$$

where i is the sample index in the range of $[1,\ N-\tau-\pi-\varepsilon+2]$, and M is defined as in formula (1), $p_{i,j}$ is a triple consisting of flow, speed and occupancy. Accordingly, the predicted FSO image can be written as:

$$y^i = \begin{bmatrix} p_{1,i+\tau+\pi-1} & p_{1,i+\tau+\pi} & \cdots & p_{1,i+\tau+\pi+\varepsilon-2} \\ p_{2,i+\tau+\pi-1} & p_{2,i+\tau+\pi} & \cdots & p_{2,i+\tau+\pi+\varepsilon-2} \\ \vdots & \vdots & \ddots & \vdots \\ p_{M,i+\tau+\pi-1} & p_{M,i+\tau+\pi} & \cdots & p_{M,i+\tau+\pi+\varepsilon-2} \end{bmatrix} \quad (3)$$

2.2.2 Convolutional layer and pooling layer

Convolutional layer plays a major role in extracting the spatiotemporal features. The previous layer's features are convolved with learnable kernels and put through the activation function to form more complex features. Generally, a convolutional layer is usually followed by a pooling layer to reduce parameters of the model and make the learned features more robust. Rectified linear unit (ReLU) activation function and max pooling procedure are used in our model because of their respective superior performance. The output of convolutional and pooling layers can be written as:

$$x_j{}^l = mp\left(\varphi\left(\sum_{i=1}^{c^{l-1}} x_i{}^{l-1} * k_{ij}{}^l + b_j{}^l\right)\right) \quad (4)$$

where l is the index of layers and j is the index of feature maps in the l th layer, $x_i{}^{l-1}$, $x_j{}^l$, $k_{ij}{}^l$ and $b_j{}^l$ denote the input, output, kernels and bias of the layer, respectively. $*$ indicates convolution operation and mp denotes max pooling procedure. φ is ReLU activation function defined as:

$$\varphi(x) = max(0, x). \quad (5)$$

2.2.3 Fully-connection layer

Fully-connection layer provides an effective way to map the spatiotemporal features back to original FSO image space. The output of this layer as predictive value of the model can be written as:

$$\hat{y} = \sigma(w^l x^{l-1} + b^l) \quad (6)$$

where w^l and b^l are the weights and bias of the layer, respectively. $\hat{y}$ are predicted values ,that is, the output of the model. σ is sigmoid activation function defined as:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (7)$$

Finally, Mean squared errors (MSEs) are employed as loss function to measure the distance between predictions and ground-truth, which are optimized by mini-batch gradient descent (mini-batch GD) algorithm. MSE can be defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}^i - y^i)^2 \tag{8}$$

where $\hat{y}^i$ and y^i are predicted values and true values of the ith sample, respectively, and N is the number of samples.

3 Experiments and Results

In this section, we first detail the generation of samples. Then we display the trained model from different aspects. Finally, we give the experimental results.

3.1 Data Description

The proposed method is evaluated on the data of West Yan'an Road in Shanghai, which were collected every 5 minutes from 35 individual inductive loops during the whole year of 2012. In fact, there are a total of 361 days of data because of the absence of data from March 20 to March 23. For practical reasons, there are inevitably some missing pieces in the data. Therefore, a proper mending method is applied to the data with spatiotemporal adjacent records. As shown in Fig.1, the green area from 21:00 pm of previous day to 7:00 am of next day corresponds to the traffic state that there are few vehicles on the road. It is incompatible with the characteristics of CNN model and almost impossible to extract spatiotemporal features because of lacking representational and differentiable patterns when using image patches to train the network. Therefore, the data with obvious traffic patterns from 7:00 am to 21:00 pm of a day are chosen to generate samples.

For the time interval of 5 minutes as a time unit, there are 168 time sequences from 7:00 am to 21:00 pm. Accordingly, when converting these sequences to FSO images, the width of the images will be 168. As a result, there are 361 FSO images of 35×168×3 available for intercepting image patches as samples. When the lengths of input and output are set to τ and ε, respectively, which means using 5τ-minute FSO data to forecast next 5ε-minute FSO data, the number of the image patches of a day will be $168 - \tau - \varepsilon + 1$. In total, there are $361 \cdot (168 - \tau - \varepsilon + 1)$ samples. First 90% of them are used as training data and the rest are used as test data.

3.2 Model display

The model is designed based on LeNet-5 architecture. The parameters of the model as listed in Table 1 are set based on the principle that the network converges to a better solution and that the train time of the network is acceptable.

Table 1. Parameters of the model.

Layer	Parameter Dim	Feature Dim	Parameter Num
Input	—	(3,35,35)	—
C1	Kernel(8,3,5,5)	(8,31,31)	608
MP1	Pooling(2,2)	(8,16,16)	—
C2	Kernel(16,8,3,3)	(16,14,14)	1168
MP2	Pooling(2,2)	(16,7,7)	—
C3	Kernel(32,16,3,3)	(32,5,5)	4640
MP3	Pooling(2,2)	(32,3,3)	—
FC4	Weight(288,105)	105	147968
Output	—	105	—

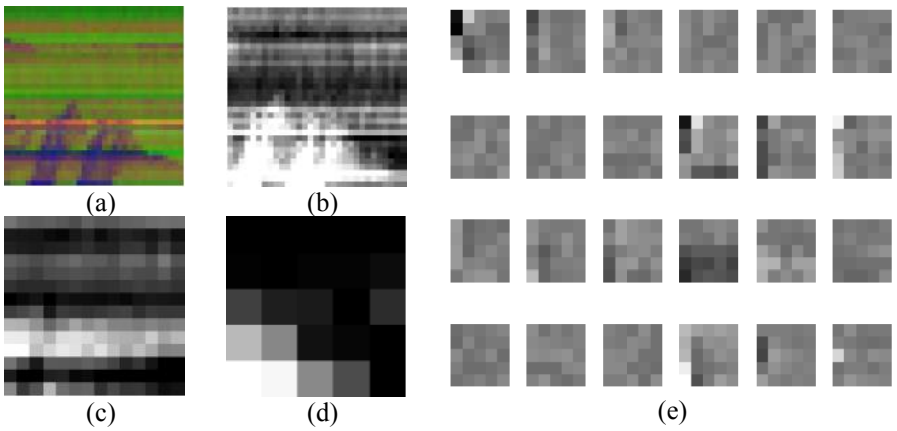

Fig. 3. (a): FSO input image. (b): Feature map of the first convolutional layer. (c): Feature map of the second convolutional layer. (d): Feature map of the third convolutional layer. Hierarchical spatiotemporal features of FSO data from simple to complex and concrete to abstract are extracted automatically through the model. (e): Kernels of the first convolutional layer. The model has learnt homogeneous kernels through training, which is consistent with the characteristics of traffic data.

3.3 Results and comparison

We first evaluate the proposed method for short-term traffic prediction with the lengths of input and output are set to 35 and 1, respectively. This means we use 175-min FSO data to forecast next 5-min data. As mentioned in section 3.1, 43211 training samples are used to train the model and 4802 test samples are used to validate the model. We compare three indexes: mean absolute errors (MAE), mean relative errors (MRE) and

relative mean square errors (RMSE) with other prevailing methods for short-term traffic prediction: ANN, DBN and random work (RW). These methods are all optimized by mini-batch SGD algorithm with batch size set to 300 except RW. Results show that our model has the best average performance as shown in Table 2 (AVG indicates average value). Notably, under the same index, three channels of flow, speed and occupancy differ due to their different data fields ranging from 10 to 1017, 1.0 to 116.0 and 1.1 to 98.7, respectively.

Table 2. Prediction performance of CNN and other models on test data.

TYPE	CNN			ANN			DBN			RW		
	MAE	RMSE	MRE (%)	MAE	RMSE	MRE (%)	MAE	RMSE	MRE	MAE	RMSE	MRE (%)
F	23.8	31.7	8.3	24.1	32.0	8.6	24.3	32.9	8.4	27.5	39.1	8.2
S	3.0	4.4	5.6	3.3	4.6	6.0	3.0	4.4	5.7	3.8	6.2	8.7
O	1.7	2.9	14.3	1.8	2.9	14.9	1.8	2.9	15.4	2.4	4.4	14.3
AVG	9.5	18.6	9.4	9.7	18.7	9.8	9.7	19.2	9.8	11.2	23.0	10.4

Table 3. Prediction performance of CNN on different time span.

TYPE	5-min Prediction			15-min Prediction			30-min Prediction		
	MAE	RMSE	MRE (%)	MAE	RMSE	MRE (%)	MAE	RMSE	MRE (%)
F	23.8	31.7	8.3	26.1	34.9	9.0	25.8	33.9	9.2
S	3.0	4.4	5.6	3.3	4.9	6.2	3.2	4.6	6.1
O	1.7	2.9	14.3	1.8	3.0	14.4	2.1	3.2	18.1
AVG	9.5	18.6	9.4	10.4	20.4	9.9	10.4	19.8	11.2

As shown in Table 3, we further evaluate the performance of our model on forecasting longer time span when setting the input length to 35 i.e. 175 minutes. From the table we can see that, with the increase of time span, the prediction accuracy is decreasing but within the acceptable range. This is due to the fusion of three-channel information which can effectively filter out influences of the outliers in channels.

Finally, we present the prediction curves of the 343th day at the 26th loop and the 357th day at the 22th loop from 10:00 am to 21:00 pm as shown in Fig.4.

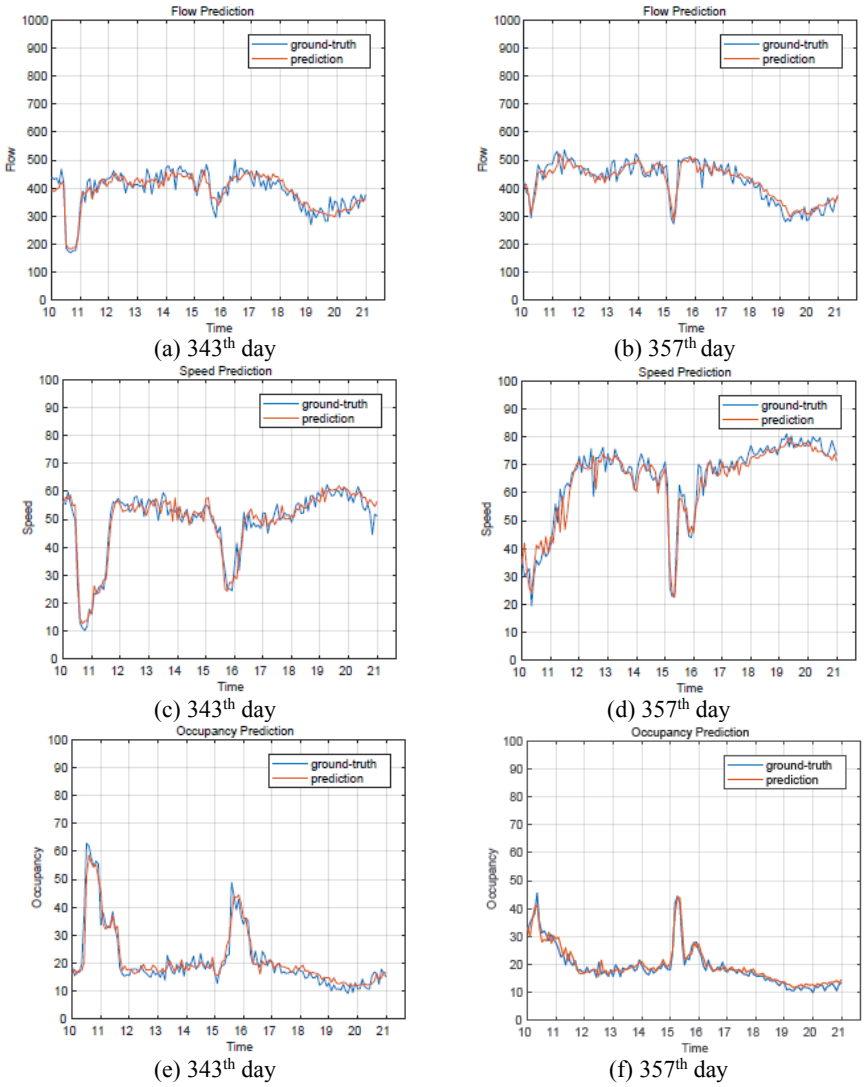

Fig. 4. (a), (c) and (e) are the flow, speed and occupancy predictions of the 343th day at the 26th loop, respectively. (b), (d) and (f) are the flow, speed and occupancy predictions of the 357th day at the 22th loop, respectively.

4 Conclusion

In this paper, we proposed a CNN-based method to forecast traffic information. Unlike most single prediction models, traffic flow, speed and occupancy are simultaneously

predicted by the model to provide more complete traffic information. Spatiotemporal traffic features of traffic data can be learned automatically by converting flow, speed, and occupancy data to color images as the input of the model.

We evaluated the proposed method on the data of West Yan'an road in Shanghai and compared it with ANN, DBN and RW methods. And results show that the proposed method is superior to others. In addition, we explored the prediction performance on different tasks of 5-min prediction, 15-min prediction and 30-min prediction and results show that proposed model has certain robustness as the prediction accuracy descends but within an acceptable range.

Acknowledgments. This work has been supported by the Fundamental Research Funds for the Central Universities of China and by National Natural Science Foundation of China under grant 61472284.

References

1. Ahmed, M.S., Cook, A.R.: Analysis of Freeway Traffic Time-series Data by Using Box–Jenkins techniques. Transp. Res. Rec. 722, 1-9 (1979).
2. Jeong, Y.S., Byon, Y.J., Castro-Neto, M.M., Easa, S.M.: Supervised Weighting-online Learning Algorithm for Short-term Traffic Flow Prediction. IEEE Transactions on Intelligent Transportation Systems, 14(4), 1700-1707 (2013).
3. Chang, H., Lee, Y., Yoon, B., Baek, S.: Dynamic Near-term Traffic Flow Prediction: System Oriented Approach Based on Past Experiences. IET Intelligent Transport Systems, 6(3), 292-305 (2012).
4. Chan, K.Y., Dillon, T.S., Singh, J., Chang, E.: Neural-network-based Models for Short-term Traffic Flow Forecasting Using a Hybrid Exponential Smoothing and Levenberg–Marquardt Algorithm. IEEE Transactions on Intelligent Transportation Systems, 13(2), 644-654 (2012).
5. Kumar, K., Parida, M., Katiyar, V.K.: Short Term Traffic Flow Prediction for a Non Urban Highway Using Artificial Neural Network. Proc. Soc. Behav. Sci. 104, 755-764 (2013).
6. Huang, W., Song, G., Hong, H., Xie, K.: Deep Architecture for Traffic Flow Prediction: Deep Belief Networks With Multitask Learning. IEEE Transactions on Intelligent Transportation Systems, 15(5), 2191-2201 (2014).
7. Ma, X., Yu, H., Wang, Y., Wang, Y.: Large-Scale Transportation Network Congestion Evolution Prediction Using Deep Learning Theory. PLoS one, 10(3) (2015).
8. Tian, Y., Pan, L.: Predicting Short-Term Traffic Flow by Long Short-Term Memory Recurrent Neural Network. In: Smart City/SocialCom/SustainCom (SmartCity), 2015 IEEE International Conference on. IEEE, pp. 153-158. (2015).
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (2012).
10. Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., Wang, Y.: Learning Traffic as Images: A Deep Convolution Neural Network for Large-scale Transportation Network Speed Prediction. Sensors, 17(4), 818 (2017).