

Lane-Level Traffic Speed Forecasting: A Novel Mixed Deep Learning Model

Wenqi Lu^{ID}, Yikang Rui^{ID}, and Bin Ran

Abstract—Lane-level traffic state prediction is one of the most essential issues in the connected automated vehicle highway systems. Accurate and timely traffic state prediction of the lane sections can assist the connected automated vehicles in planning the optimal route and making lane selection. In this article, we tackle the problem of forecasting lane-level short-term traffic speed and propose a novel mixed deep learning (MDL) model by coordinating the convolutional long short-term memory (Conv-LSTM) layers, convolutional layers, and a dense layer in an end-to-end structure. The introduction of the Conv-LSTM neural network enables the proposed MDL model to better capture the spatio-temporal characteristics and correlations of the dynamic lane-based traffic flow synchronously. To improve the efficiency of the proposed model, a feature correlation analysis method based on the maximum information coefficient is presented to measure the relevance between the historical traffic flows and the traffic speeds to be forecasted. Validated by the ground-truth traffic flow data collected by the remote traffic microwave sensors installed on the expressways in Beijing, the MDL model is capable of capturing the fluctuation of the lane-level traffic speeds at different types of lanes effectively during the whole day. Furthermore, the results confirm that the MDL model achieves better predictive performance than several state-of-the-art benchmark models in terms of prediction accuracy and space-time distributions. Our code and data are available at <https://github.com/lwqs93/MDL>.

Index Terms—Traffic speed prediction, lane-level traffic data, spatial-temporal modeling, Conv-LSTM neural network, feature correlation analysis, maximum information coefficient.

I. INTRODUCTION

AS THE core of the smart city, intelligent transportation system (ITS) is an effective way to improve urban planning and ease traffic congestion [1]. Traffic state prediction, which uses traffic flow information collected by various detection devices to forecast traffic patterns for a transient period in the future with the help of relevant algorithms, is a

Manuscript received November 28, 2019; revised May 19, 2020 and August 26, 2020; accepted November 9, 2020. Date of publication December 7, 2020; date of current version March 29, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2019YFB1600100, in part by the National Natural Science Foundation of China under Grant 41971342, in part by the Fundamental Research Funds for the Central Universities under Grant 2242020k30016, in part by the Science and Technology Program of Beijing under Grant Z121100000312101, and in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX20_0136. The Associate Editor for this article was Y. Kamarianakis. (*Corresponding author: Yikang Rui*)

The authors are with the School of Transportation, Southeast University, Nanjing 211189, China (e-mail: lplwq93@126.com; 101012189@seu.edu.cn; bran@seu.edu.cn).

Digital Object Identifier 10.1109/TITS.2020.3038457

significant issue in the framework of ITS [2]. Real-time and accurate traffic state prediction not only assists travelers in finding optimal routes to avoid potential congestion but also helps traffic management departments take timely measures to improve the efficiency of management in traffic systems based on forecasted information [3].

Meanwhile, with the rapid improvement of the connected automated vehicle highway (CAVH) system [4], short-term traffic state forecasting has been gradually shifting from the section-based or network-based methods to lane-based methods [5]. In the environment of the CAVH system, the traffic flows on the road are generally mixed with connected automated vehicles (CAVs) and human-driven vehicles (HVs) [6]. On the one hand, forecasting dynamic traffic flow of lane sections can provide more real-time and detailed traffic state information for the HVs to choose the appropriate travel route and overcome the influence of the limited sight distance [7]. On the other hand, high-efficiency lane-level prediction can assist the CAVs in making lane selection and planning the optimal travel trajectory in terms of the level of service based on the predicted traffic information [8]. Sequentially, the overall distribution of various vehicles on the roads can be balanced and the road capacity will be improved correspondingly.

Although relevant scholars have put forward many algorithms to study the issue of predicting short-term traffic state during the past decades. Several limitations and challenges still exist. 1) The conventional research targets of traffic state prediction mainly focus on the road sections or large-scale networks with the implicit hypothesis that traffic patterns between different lanes are the same or exactly similar. In fact, the traffic flows on different lanes show different yet correlated patterns [9]–[11], and prediction results of road sections or network are too macroscopic to provide effective and precise information for CAVs to choose appropriate lanes. 2) Limited by infrastructure and data acquisition equipment, the sampling time intervals of conventional traffic prediction studies usually range from 5 to 30 min [12]–[15]. It leads to a long prediction horizon, which is difficult for the CAVs to timely perceive and predict the short-term change of the traffic state from the lane-level perspective. 3) Most existing deep learning models [16]–[19] utilize separated structures to extract the spatial and temporal features of the traffic flow independently and ignore the internal relation between the spatio-temporal features of the traffic flow. Thus, the current structures could be insufficient for making lane-level traffic speed prediction.

According to the aforementioned analysis, this article proposes a novel mixed deep learning model (MDL) for forecasting traffic state in the future based on convolutional long short-term memory (Conv-LSTM) neural network. The lanes of roads are employed as the research target and average travel speeds of the lanes are utilized as variables for state representation. The sampling time intervals are shortened to 2 min in order to collect more real-time traffic information. Furthermore, this article proposes a feature correlation analysis to help the MDL model choose the appropriate input time windows of the spatio-temporal variables. Based on the traffic flow data captured by stationary detectors in the ITS or extracted from the vehicle motion information (e.g., position and velocity) shared by the CAVs in the CAVH system, a well-trained MDL model is capable of providing the future traffic speeds of lanes within a few seconds. Therefore, the proposed method can forecast the lane-level traffic state in real time.

The contribution of this article can be summarized as follows:

(1) In this article, we concentrate on studying the lane-based traffic flow and propose a novel mixed deep learning structure named MDL to forecast lane-level traffic speeds. Specifically, the proposed model integrates the Conv-LSTM layers, convolutional layers, and a dense layer in an end-to-end fashion.

(2) To the best of our knowledge, the Conv-LSTM neural network, which can learn the spatio-temporal characteristics synchronously due to its inherent convolutional structure, is first introduced to extract the feature of the lane-level traffic flow.

(3) A feature correlation analysis method based on the maximum information coefficient (FCA-MIC) is proposed to measure the importance of relevance between the historical traffic flow and the traffic speeds to be predicted. The FCA-MIC can assist the MDL model in selecting the input time window and reducing the consumption of the training time.

(4) Examined by the real-world traffic flow data captured by remote traffic microwave sensors (RTMS) installed on the 4th ring road of Beijing, the proposed MDL model illustrates better performance than state-of-the-art models including the deep learning models in terms of prediction accuracy, stability, and space-time distribution.

In the next section, we review the literature on the existing traffic prediction models. Section III presents the methodology including the Conv-LSTM framework, the proposed MDL model, and the FCA-MIC method. In section IV, we introduce the dataset, performance metrics, benchmark methods, and experimental platform. The results of the experiments are presented and analyzed in section V. Finally, the conclusions and future work are discussed in section VI.

II. LITERATURE REVIEW

In this section, a literature review on the traffic flow prediction methods is presented by categorizing them into statistical methods, artificial intelligence-based methods, and deep learning methods.

A. Statistical Methods

The statistical methods which can also be called statistical time series analysis methods are the foundational solution for the traffic flow prediction problem. Some classical statistical methods include the nonparametric regression models [20], the Kalman filtering model [21], traffic flow theory-based methods [22], and generalized autoregressive conditional heteroscedasticity (GARCH) model [23]. One of the most popular statistical approaches is to use Auto-regressive integrated moving average (ARIMA) models. ARIMA and its variants [24]–[26] have been proved useful in many cases. One advantage of statistical methods lies in that they can make very good predictions when the traffic flow varies temporally and regularly. However, these methods only consider the temporal dependency of the traffic flow and are inadequate for capturing the rapid variations of lane-level traffic flow.

B. Artificial Intelligence-Based Methods

To overcome the flaws of the statistical methods, numerous artificial intelligence-based methods including k nearest neighbour [27], support vector regression [28], [29], locally weighted learning [30], tensor-based models [31], [32], and random forest [33] were implemented to deal with the traffic flow prediction problems and obtained certain achievement. Artificial neural networks (ANN) models, which can capture the underlying relationships of traffic flows, are most widely used among these methods [34]–[36]. For example, Yin *et al.* [37] established a fuzzy-neural model to forecast traffic flow in an urban street network and the experimental results indicate that the fuzzy-neural model provides more accurate forecasting results than the back propagation neural network (BPNN) [38]. Kuang and Huang [39] established a radial basis function neural network (RBFNN) to forecast traffic volume. However, though the artificial intelligence-based methods can introduce the spatial and temporal information of the traffic flow as input, it is difficult for these methods even the ANNs to handle a large amount of spatio-temporal data and predict lane-level traffic flow reliably.

C. Deep Learning Methods

Recently, with the emergence of high-performance data storage and processing technology [40], traffic flow prediction methods are shifting from artificial intelligence-based methods to deep learning methods [41]–[45]. As temporal dependencies are the most common features used to develop traffic predictors, many researchers have proposed the deep learning methods to mine the temporal dependency of traffic flow sequence [46], [47]. Ma *et al.* [48] first introduced the long-short term memory neural network (LSTMNN), an extension of recurrent neural networks (RNN), to solve the traffic speed and volume prediction on the highways. Rui *et al.* [49] introduced gated recurrent unit neural network (GRUNN), which was a variant of LSTMNN, to the traffic prediction system. Gu *et al.* [5] built a fusion model (LSTM-GRU) by integrating the LSTMNN and GRUNN to capture the speeds of the lanes effectively. Among the RNN-based models, LSTMNN and GRUNN have become

the favorite choice to mine the temporal characteristics of the time-series issues due to their capability for capturing long-term dependency. However, the RNN-based models always ignore the influence of the upstream and downstream traffic condition and they are weak in mining the spatial relationship of the traffic flow.

As well as temporal dependencies, spatial dependencies have also been exploited in various research in this field and the traffic flow usually has a stronger correlation on nearby locations [50], [51]. Huang *et al.* [52] established a deep learning structure that consisted of a deep belief network and a multi-task regression layer to predict traffic flow. The traffic flow data collected from different positions of the highways were utilized as the input feature variables. Yang *et al.* [53] developed a stack denoise autoencoder method to learn hierarchical representation of urban traffic flow. Convolutional neural network (CNN) has been naturally utilized in some works associated with traffic flow forecasting due to its powerful function of handling the image data. Ma *et al.* [54] established a deep CNN framework for large-scale traffic network speed prediction with space-time matrix converted to an image as the input of the CNN. Liu *et al.* [12] proposed an attention-based CNN structure and used three-dimensional data matrices constructed by flow, speed, and occupancy to predict traffic speed. Ke *et al.* [16] proposed a two-stream multi-channel convolutional neural network model for multi-lane traffic speed prediction considering traffic volume impact.

However, neither CNN-based models nor RNN-based models are perfect for spatio-temporal forecasting problems. Therefore some advanced deep learning structures were put forward based on considering both spatial and temporal features of the traffic flow. Wu *et al.* [17] established a deep learning neural network that combined CNNs and GRUs (CNN-GRU) to make full use of weekly and daily periodicity and spatio-temporal characteristics of traffic flow. Yu *et al.* [18] proposed a spatio-temporal recurrent convolutional network that inherited the advantages of deep CNN and LSTMNN to predict the network-wide traffic state. However, these models still have some limitations. First, the existing hybrid deep learning structures are rarely utilized in handling the problem of forecasting lane-level traffic state. Second, the CNN and LSTMNN are normally separated from each other. Hence, it is difficult for these models to extract the spatio-temporal information effectively and synchronously.

Therefore in this study, we build a mixed deep learning model for lane-level traffic speed prediction. The Conv-LSTM structure is introduced to capture the spatio-temporal characteristics of the lane-level traffic flow simultaneously. For a target lane to be predicted, the influence of traffic flow from the adjacent, upstream, and downstream lanes during the several previous time intervals are taken into consideration to improve the accuracy and robustness of the traffic flow prediction.

III. METHODOLOGY

In this section, we firstly introduce the concept of the Conv-LSTM neural network. Then we briefly describe the structure of the proposed MDL model. Finally, a feature correlation analysis approach named FCA-MIC is presented.

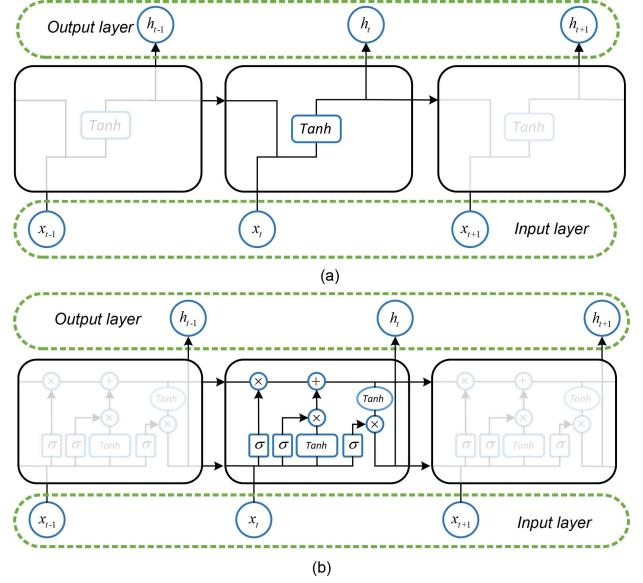


Fig. 1. Standard chain structures of RNN and LSTM. (a)RNN; (b)LSTM.

A. Convolutional Long Short-Term Memory Neural Network

Convolutional long short-term memory neural network (Conv-LSTM) proposed by Shi *et al.* [55] is an advanced deep learning structure which is capable of simultaneously capturing the spatio-temporal characteristics of the tensor-based input. It is an extension of the fully connected LSTMNN. In this section, we first introduce the LSTMNN and then extend it into Conv-LSTM.

As the traditional ANN is weak in dealing with the time-series issues, RNN is presented to learn the temporal information from the input of the previous time intervals. However, RNN has the gradient vanishing problem [56], which makes RNN unable to remember long-term historical information. To solve the aforementioned problems, LSTMNN [57] is proposed and proved to be more efficient than the standard RNN in the task of time series problems such as traffic flow prediction [48].

Fig. 1 shows standard chain structures of a RNN and a LSTMNN. As shown in Fig. 1, the only different part between the standard RNN and the standard LSTMNN lies in the hidden layer. The hidden layer of the LSTMNN is a memory module which contains the cell state c_t . The memory module of a LSTMNN has three special gate structures, by which the LSTMNN allows information to selectively influence the cellular state of the circulatory neural network at every moment. The three gates include an input gate i_t , a forget gate f_t and an output gate o_t . The activation function of all the gates is Sigmoid function which produces a value between 0 and 1 to control how much information can pass through the gate structure. The input of the LSTMNN and output of the hidden layer are vector sequences $x = (x_1, x_2, \dots, x_T)$ and $h = (h_1, h_2, \dots, h_T)$. T is the number of the timestamps. Note that x_t and h_t can share different dimension. Equations of i_t , f_t , o_t , c_t , and h_t , $t = 1, 2, \dots, T$ are demonstrated as follows:

$$i_t = \sigma_s(w_{xi}x_t + w_{hi}h_{t-1} + w_{ci} \circ c_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma_s(w_{xf}x_t + w_{hf}h_{t-1} + w_{cf} \circ c_{t-1} + b_f) \quad (2)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(w_{xc}x_t + w_{hc}h_{t-1} + b_c) \quad (3)$$

$$o_t = \sigma_s(w_{xo}x_t + w_{ho}h_{t-1} + w_{co} \circ c_t + b_o) \quad (4)$$

$$h_t = o_t \circ \tanh(c_t) \quad (5)$$

where \circ refers to hadamard product that produces the element-wise products of two vectors, matrices, or tensors with the same dimension; $\sigma_s(\cdot)$ is the sigmoid function; w_{hi} , w_{ci} , w_{xf} , w_{hf} , w_{cf} , w_{xc} , w_{hc} , w_{xo} , w_{ho} , and w_{co} are the weight matrices; b_i , b_f , b_c , and b_o are the bias matrices.

Although LSTMNN can well capture the temporal characteristics of variables and overcome the gradient vanishing problem existing in RNN [56], the LSTMNN cannot extract the spatial features of the variables in the process of time series prediction. Hence, the Conv-LSTM neural network is presented by considering the advantages of both CNN and LSTMNN.

Different from the dimension of the variables in Equation (1)-(5), the variables in the Conv-LSTM neural network are 3D tensors such as the input tensors $\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_t$, memory cell $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_t$, hidden states $\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_t$, and three gates $\mathcal{I}_t, \mathcal{F}_t, \mathcal{O}_t$. Note that $\mathcal{X}_t, \mathcal{C}_t, \mathcal{H}_t, \mathcal{I}_t, \mathcal{F}_t, \mathcal{O}_t \in \mathbb{R}^{M \times N \times L}$ where M is the number of rows in the grid; N is the number of columns in the grid; L is the length of each grid feature vector. In the framework of the Conv-LSTM, the future state of a specific unit in the input grid is determined according to the input of adjacent units and their past state. As illustrated in Fig. 2, the process of state to state and input to state can be realized by relying on the convolutional process. Equation (1)-(5) can be written as follows:

$$\mathcal{I}_t = \sigma_s(W_{xi} * \mathcal{X}_t + W_{hi} * \mathcal{H}_{t-1} + W_{ci} \circ \mathcal{C}_{t-1} + b_i) \quad (6)$$

$$\mathcal{F}_t = \sigma_s(W_{xf} * \mathcal{X}_t + W_{hf} * \mathcal{H}_{t-1} + W_{cf} \circ \mathcal{C}_{t-1} + b_f) \quad (7)$$

$$\mathcal{C}_t = \mathcal{F}_t \circ \mathcal{C}_{t-1} + \mathcal{I}_t \circ \tanh(W_{xc} * \mathcal{X}_t + W_{hc} * \mathcal{H}_{t-1} + b_c) \quad (8)$$

$$\mathcal{O}_t = \sigma_s(W_{xo} * \mathcal{X}_t + W_{ho} * \mathcal{H}_{t-1} + W_{co} \circ \mathcal{C}_t + b_o) \quad (9)$$

$$\mathcal{H}_t = \mathcal{O}_t \circ \tanh(\mathcal{C}_t) \quad (10)$$

where $*$ refers to the convolutional operator; W_{xi} , W_{hi} , W_{xf} , W_{hf} , W_{xc} , W_{hc} , W_{xo} , and W_{ho} are the convolutional filters that mine the spatial features of the input; W_{ci} , W_{cf} , and W_{co} are weight tensors; b_i , b_f , b_c , and b_o are the bias tensors whose dimension are the same as those of gate tensors.

In the convolution process, the length and width of the convolutional filter are artificially specified. Convolutional filter sizes of 2×2 and 3×3 are widely utilized, and the depth of the convolutional filter is the same as that of the tensor which needs to be processed. To ensure that rows and columns of the input and output are consistent, the zero-padding method can be exploited on the edge of the input tensors to create zeros value around the grids before the convolution process.

Meanwhile, the spatio-temporal variables, such as speed during a time interval is a 2D matrix. Therefore, we need to transfer the original input into 3D tensor by utilizing the mapping function $\mathcal{F}_{2D \rightarrow 3D} : \mathbb{R}^{M \times N} \rightarrow \mathbb{R}^{M \times N \times 1}$, where M and N represent the dimensions of rows and columns of the input. One Conv-LSTM layer can process an input

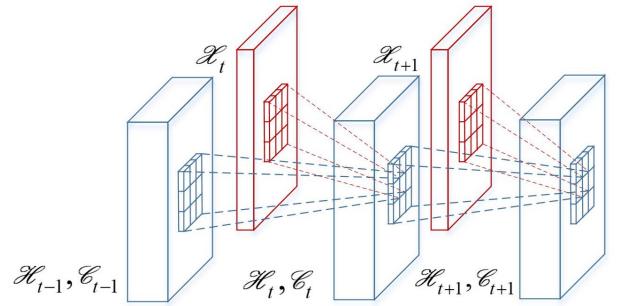


Fig. 2. The convolutional process of the Conv-LSTM with the filter size of 3×3 .

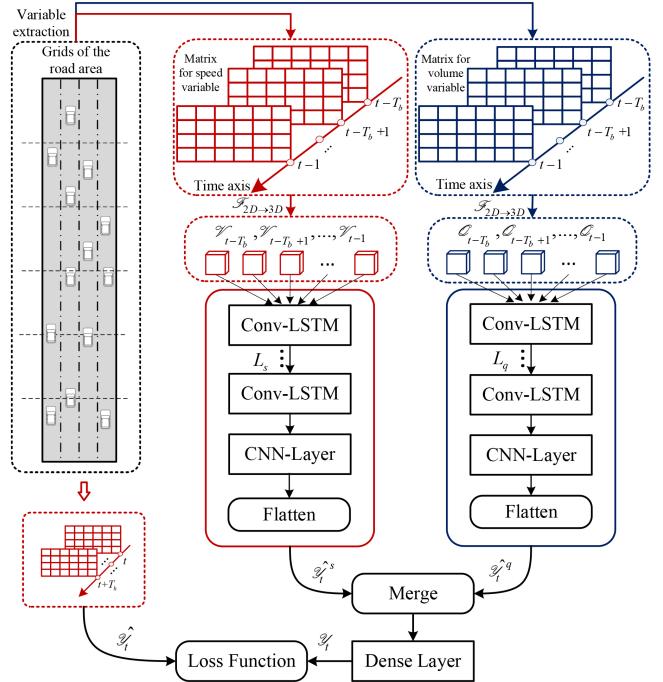


Fig. 3. The structure of the proposed MDL model.

tensor $\mathcal{X} = (\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_T)$ into an output tensor $\mathcal{H} = (\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_T)$, where T denotes the number of the timestamps. Therefore each Conv-LSTM layer can be defined as a mapping function $\mathcal{F}^{CL} : \mathbb{R}^{T \times M \times N \times L} \rightarrow \mathbb{R}^{T \times M \times N \times L'}$, where L refers to the length of one input feature vector in each grid and L' refers to the length of the output feature vector in each grid.

B. Mixed Deep Learning Model

In this section, a mixed deep learning model (MDL) is proposed for forecasting the traffic speeds of lane sections, and the structure of the MDL model is illustrated in Fig. 3.

In the proposed model, the area of the road is simplified as $m \times n$ grids, where m represents the number of the road sections and n represents the number of the lanes at one road section. Note that each grid represents the area of a lane section and a $m \times n$ matrix can store the values of all lane sections for a spatio-temporal variable such as speed, volume, or occupation. Considering the traffic flow theory

that one traffic parameter could be estimated if another two parameters are given, the **traffic speed and volume** are selected as two spatio-temporal variables of the MDL model.

Let denote V_t and Q_t as the speed matrix and volume matrix respectively during the time interval t . Through the use of the mapping function $\mathcal{F}_{2D \rightarrow 3D}$, V_t and Q_t are transfer into 3D tensors \mathcal{V}_t and \mathcal{Q}_t with the dimension of $m \times n \times 1$. The prediction target \mathcal{Y}_t is a vector containing traffic speed of all lane sections during T_h time intervals in the future and length of the \mathcal{Y}_t is $T_h \times m \times n$, where T_h is the prediction horizon.

Due to the fact that the historical travel speed and traffic volume have influence on the speed prediction in different ways, these two kinds of variables are fed into two separate architectures, each of which consists of a series of stacked Conv-LSTM layers and a convolutional operator. The numbers of the stacked Conv-LSTM layers in the architectures of speed and volume are donated as L_s and L_q respectively. T_b represent the input time window of the speed variable and the volume variable. The formulations of the architecture for the two spatio-temporal variables are shown in Equation (11)-(14) given by:

$$\begin{aligned} & (\mathcal{U}_{t-T_b}^{L_s}, \mathcal{U}_{t-T_b+1}^{L_s}, \dots, \mathcal{U}_{t-1}^{L_s}) \\ &= \mathcal{F}_{L_s}^{CL} \dots \mathcal{F}_l^{CL} \dots \mathcal{F}_1^{CL} (\mathcal{V}_{t-T_b}, \mathcal{V}_{t-T_b+1}, \dots, \mathcal{V}_{t-1}) \end{aligned} \quad (11)$$

$$\hat{\mathcal{Y}}_t^s = \sigma_r (W_s * \mathcal{U}_{t-1}^{L_s} + b_s) \quad (12)$$

$$\begin{aligned} & (\mathcal{G}_{t-T_b}^{L_q}, \mathcal{G}_{t-T_b+1}^{L_q}, \dots, \mathcal{G}_{t-1}^{L_q}) \\ &= \mathcal{F}_{L_q}^{CL} \dots \mathcal{F}_l^{CL} \dots \mathcal{F}_1^{CL} (\mathcal{Q}_{t-T_b}, \mathcal{Q}_{t-T_b+1}, \dots, \mathcal{Q}_{t-1}) \end{aligned} \quad (13)$$

$$\hat{\mathcal{Y}}_t^q = \sigma_r (W_q * \mathcal{G}_{t-1}^{L_q} + b_q) \quad (14)$$

where $\mathcal{U}_{t-t'}^{L_s}$, $t' = 1, 2, \dots, T_b$ and $\mathcal{G}_{t-t'}^{L_q}$, $t' = 1, 2, \dots, T_b$ are output hidden tensors in the highest-level layers of the speed and volume respectively; W_s and W_q are the convolutional operators to further capture the spatial correlation of high-dimensional output tensors of the Conv-LSTM layers; b_s and b_q are the bias tensors; $\sigma_r(\cdot)$ is the ReLU activation function.

After obtaining the two outputs of the convolutional layers $\hat{\mathcal{Y}}_t^s$ and $\hat{\mathcal{Y}}_t^q$, we flatten them and fuse them into one vector by using the concatenation as the fusion function. This fusion process is mathematically represented as follows:

$$\hat{\mathcal{Y}}_t = W_y \times Conc[F(\hat{\mathcal{Y}}_t^s), F(\hat{\mathcal{Y}}_t^q)] + b_y \quad (15)$$

where W_y and b_y are the weight and bias respectively; $F(\cdot)$ is the flatten function and $Conc(\cdot)$ is the concatenation function.

Table I gives the pseudo-code of training an MDL model. In the training process, the objective function of the structure shown in Equation (16) is to minimize the mean squared error between the predicted speeds and observed speeds, through which the weights and biases in the structure can be calibrated.

$$\min_{W,b} \left\| \hat{\mathcal{Y}}_t - \mathcal{Y}_t \right\|_2^2 + \alpha \|W\|_2^2 \quad (16)$$

where $\hat{\mathcal{Y}}_t$ is the predicted speed vector and \mathcal{Y}_t is the observed speed vector; $\alpha \|W\|_2^2$ is an L_2 -norm regularization term that

TABLE I
THE PSEUDO-CODE OF TRAINING AN MDL MODEL

Algorithm: The process of training an MDL model

Input: Speed matrices $\{V_1, V_2, \dots, V_n\}$;
Volume matrices $\{Q_1, Q_2, \dots, Q_n\}$;
Input time window: T_b ;
The number of the Conv-LSTM layers: L_s, L_q ;

Output: The trained MDL model

1. Procedure MDL Training
2. Initialize a null set $S \leftarrow \emptyset$
3. **for** all available time intervals t ($1 \leq t \leq n$) **do**
4. $\mathcal{S}_t^v \leftarrow [\mathcal{V}_{t-T_b}, \mathcal{V}_{t-T_b+1}, \dots, \mathcal{V}_{t-1}]$
5. $\mathcal{S}_t^q \leftarrow [\mathcal{Q}_{t-T_b}, \mathcal{Q}_{t-T_b+1}, \dots, \mathcal{Q}_{t-1}]$
6. Put the training samples $(\{\mathcal{S}_t^v, \mathcal{S}_t^q\}, \mathcal{Y}_t)$ into S
7. **end**
8. Initialize all the weighted and intercept parameters
9. **repeat**
10. Randomly extract a batch of samples S^b from S
11. Estimate the parameters by minimizing the objective function shown in Equation (16)
- 12 **until** convergence criterion met

reduces the overfitting phenomenon; The weighted parameters in $\hat{\mathcal{Y}}_t$ are defined as W ; α is a regularization parameter.

C. Feature Correlation Analysis Method

For the proposed MDL model, it is necessary to find the historical traffic flows which are highly correlated with target traffic speeds to construct input feature tensors. In this section, a feature correlation analysis method named FCA-MIC is presented based on the maximum information coefficient (MIC), which is capable of identifying a wide range of relationships between pairs of variables in large datasets [58]. The core idea of the MIC is that if a relationship exists between two variables, a grid can be drawn on the scatterplot of the two variables that partitions the data to encapsulate that relationship. By calculating the average value of the MICs between the target speeds and traffic feature variables at different lanes, the FCA-MIC method can measure their relationships effectively and help the MDL model select the input time window.

The detailed process of using the FCA-MIC method to capture nonlinear associations of the target speed and traffic feature variable is described as follows.

It is assumed that the v_i^t is the travel speed of the i -th lane section at the time interval t . x_i^t is the historical value of the feature such as speed or volume of the i -th lane section at the time interval t . Let $\mathbf{v}_i^t = \{v_i^w \mid w \in T_w\}$ represent the sequence of traffic speed to be forecasted at i -th lane section and $\mathbf{x}_i^{t-k} = \{x_i^{w-k} \mid w \in T_w\}$ represent the sequence of a feature variable such as speed or volume at i -th lane section whose elements are k time intervals before the corresponding elements in \mathbf{v}_i^t , where k is time gap. $T_w = \{j, j+1, \dots, j+s-1\}$ is the index set of the time intervals, where j is the index of the first time interval and s is the length of the sequence.

Firstly, according to the approximate probability density distribution of the two variables, the mutual information (MI)

between v_i^t and x_i^{t-k} can be calculated formulated as:

$$\text{MI}(v_i^t, x_i^{t-k}) = \sum_{v_i^t, x_i^{t-k}} p(v_i^t, x_i^{t-k}) \log_2 \frac{p(v_i^t, x_i^{t-k})}{p(v_i^t)p(x_i^{t-k})} \quad (17)$$

where MI represents the amount of information shared by the two variables; $p(v_i^t)$ and $p(x_i^{t-k})$ are edge density of v_i^t and x_i^{t-k} respectively.

Then, the value of the MIC between v_i^t and x_i^{t-k} can be obtained by regularizing the MI as follows:

$$\text{MIC}(v_i^t, x_i^{t-k}) = \max_{|v_i^t||x_i^{t-k}| < B(m)} \frac{\text{MI}(v_i^t, x_i^{t-k})}{\log_2(\min(|v_i^t|, |x_i^{t-k}|))} \quad (18)$$

where $|v_i^t|$ and $|x_i^{t-k}|$ are the number of grids by dividing the domain of v_i^t and x_i^{t-k} ; m is the number of the ordered pairs and $m = s$; $B(m)$ is the limiting condition for $|v_i^t| \times |x_i^{t-k}|$, and $B(m) = m^{0.6}$ [58].

Finally, the mean value (MMIC) of the $\text{MIC}(v_i^t, x_i^{t-k})$, $i = 1, 2, \dots, I$ is used to measure the dependence between the target speed and feature variable with a gap of k time intervals. The MMIC is calculated as:

$$\text{MMIC}(V_t, X_{t-k}) = \frac{1}{I} \sum_{i=1}^I \text{MIC}(v_i^t, x_i^{t-k}) \quad (19)$$

where $V_t = \{v_1^t, v_2^t, \dots, v_I^t\}$ represents the set of target speed of all lane sections at the time interval t and $X_{t-k} = \{x_1^{t-k}, x_2^{t-k}, \dots, x_I^{t-k}\}$ represents the set of traffic feature at the time interval $t - k$.

IV. EXPERIMENT

A. Data Description

In this article, the data¹ used for validating the proposed MDL model were provided by Beijing Municipal Commission of Transport and they were captured by the remote traffic microwave sensors (RTMS) installed on the 4th ring road in Beijing. As shown in Fig. 4, 9 road sections covering 36 lane sections were used as research targets and the length of each lane section is approximately 1.1 km. The traffic flow data were collected from 2014.1.6 to 2014.1.19 and from 2014.2.17 to 2014.3.2 with the updating frequency of 2 min in 24 hours a day, where the key information of each record includes the traffic speed and volume of a lane section. The number of the records collected at each lane section is 20,160 and the entire dataset comprises of 725,760 records with the sampling accuracy higher than 95% [59]. Considering the continuity of traffic flow on the urban expressways and the original traffic flow data of lanes were captured with high quality, the original dataset were appropriately utilized to test algorithms.

Note that the raw traffic volume of the lanes mainly contains the volume of different types of vehicles. By using the conversion factors [60], the volume of different types of vehicles was converted to the traffic volume of the passenger car unit. Besides, the traffic speed of the lane is the average

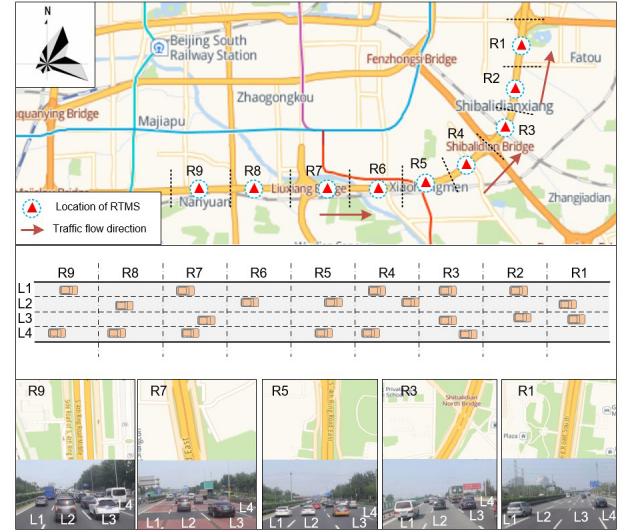


Fig. 4. The observation of the experimental lane sections.

speeds of the vehicles which pass the lane sections during a sampling time interval. To ensure a more reliable result, the missing and erroneous records were properly remedied by using the historical averaged based data imputation approach [35]. Meanwhile, it is assumed that the traffic speeds to be forecasted are mainly affected by the upstream or downstream traffic flow of the mainline and influence of the traffic flow from the ramps are not considered in this experiments.

To validate the prediction performance of different models and fairly compare the prediction accuracy, the dataset was divided into two parts: the training dataset and the testing dataset. The data of the first three weeks were used to calibrate the parameters of the models, and the data in the last week were used to test the models.

B. Evaluation Metrics and Experimental Environment

In this article, we evaluate the models via the four measures of effectiveness: mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE), and Theil inequality coefficient (TIC) [61], given by:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (20)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (21)$$

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{\hat{y}_i - y_i}{y_i} \right| \times 100\% \quad (22)$$

$$\text{TIC} = \frac{\sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i)^2} + \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i)^2}} \quad (23)$$

where \hat{y}_i is the predicted value and y_i is the measured value during the i -th time interval. n is the number of the total predicted samples.

¹<https://github.com/lwqs93/MDL/blob/master/DATASET.txt>

TABLE II
STRUCTURE OF PARTIAL NN-BASED MODELS FOR TRAFFIC SPEED PREDICTION

NN-based models	Input Feature	Hidden layer number	Hidden units number (bottom-top)	Activation function
BPNN	speed+volume	1	[256]	Sigmoid
RBFNN	speed+volume	1	[256]	Gaussian
MLP	speed+volume	2	[256 128]	ReLU
GRUNN	speed	1	[128]	ReLU
LSTMNN	speed	1	[128]	ReLU
LSTM-GRU	speed	2	[128 128]	ReLU

Note that MAE, RMSE, and MAPE are utilized to represent the accuracy of the prediction models. TIC reflects the stability and fitting degree between the predicted value and the observed value. The smaller the TIC is, the higher the fitting degree is [61].

The experimental platform of this research is a Dell computer with Intel(R) Core(TM) i7-8700 CPU@3.20 GHz. The Python 3.6 with the Keras 2.0, Tensorflow 1.0, Scikit-learn 0.20, and XGboost 0.9 is utilized to realize the proposed method and benchmark models.

C. Benchmark Models

Eight types of base models are employed in this research, including multi-layer perceptron (MLP), back-propagation neural network (BPNN), RBFNN [62], extreme gradient boosting (XGBoost) [63], LSTMNN [48], GRUNN [49], LSTM-GRU [5], TM-CNN [16], and CNN-GRU [17]. To ensure the fairness of these experiments, slight modifications were made to accommodate the experimental data to those models.

For the MLP, BPNN, RBFNN, XGBoost, and LSTM-GRU model, the traffic speed or volume of the target lanes and their adjacent lanes, upstream lanes, and downstream lanes during the previous time intervals are employed as the input variables. The structures of the BPNN, RBFNN, and MLP are illustrated in Table II. For the XGBoost, the maximum depth of a tree is chosen to be 5 with the learning rate set as 0.01. For the RNN-based models, the LSTMNN, GRUNN, and LSTM-GRU only consider the temporal characteristics of the traffic flow by using the speed data. As shown in Table II, there is one hidden layer with 128 hidden units in the LSTMNN and GRUNN model. The LSTM-GRU model consists of two hidden layers and each layer contains 128 hidden units.

The CNN-GRU model introduces double-layer CNN and double-layer GRUNN to capture the spatio-temporal features of the traffic flow by using traffic speed data. Each layer of the CNN uses 15 filters with a size of a 3×3 and each hidden layer of the GRUNN contains 128 hidden units. The TM-CNN model employs input tensors of speed and volume with the dimension of $T_c \times 9 \times 4$ where T_c is the number of the channels, which is equal to the input time window. Two double-layer CNNs are utilized to capture the spatio-temporal characteristics and merged by a dense layer with the ReLU activation function. For the proposed MDL model, L_s and L_q are both set as 1 with the filter number and filter size chosen to be 10 and 3×3 respectively in the Conv-LSTM layers. The

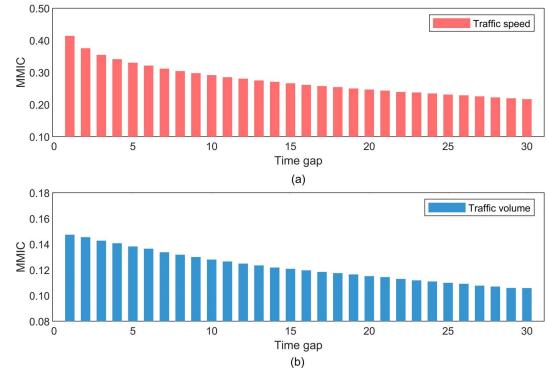


Fig. 5. The results of the feature correlation analysis. (a) traffic speed; (b) traffic volume.

CNN layer uses 5 filters with a size of 3×3 to handle the output of the Conv-LSTM layer. The optimization algorithm is the Adamax, which is an efficient and robust minibatch gradient descent approach.

For all models, the training epoch is set to be 50 and the size of a minibatch is 64. The mean square error between the predicted values and the observed values is selected as the objective function, and 10% training data are employed for cross-validation to reduce overfitting.

V. RESULT AND DISCUSSION

A. The Results of Feature Correlation Analysis

In this section, the FCA-MIC method is employed to measure the relevance between the traffic feature variables (speed and volume) during different past time intervals and the traffic speeds to be forecasted. The values of MMIC for speed and volume variable with the time gap (k) ranging from 1 to 30 were calculated. As shown in Fig. 5, with the increase of the time gap, the MMICs of the speed and volume show a similar descending trend. Meanwhile, when the input time window is the same, the MMICs of the speed variable are larger than those of the volume variable.

Hence, considering time consumption of training NN-based models and preserve the significant spatio-temporal information as much as possible, the input time window for feature variables in the different models are all set as 15, which means the traffic flow information during past 30 min are utilized to construct input tensors of the models.

B. Prediction Results of the MDL Model

Fig. 6 illustrates the prediction results of the MDL model at different road sections by taking the lane-based prediction results as a whole at each road section. As shown in Fig. 6, MAEs and MAPEs of the different road sections are similar, indicating that the MDL model is stable and suitable for multiple scenarios. Meanwhile, Fig. 6 shows that the prediction errors of the R1 and R9 are slightly larger than those of other road sections, which may be caused by the missing of the upstream or downstream traffic flow information for the two road sections.

TABLE III

THE PREDICTION RESULTS OF THE MDL MODEL AT DIFFERENT TYPES OF LANES

Time	Lane Type	MAE	RMSE	MAPE	TIC
Weekdays	Inside lane	6.00	7.89	11.23%	0.0644
	Middle lane	6.07	7.86	10.68%	0.0591
	Outside lane	6.14	8.14	13.98%	0.0616
	Total	6.07	7.94	11.64%	0.0609
Weekends	Inside lane	6.19	8.10	14.42%	0.0677
	Middle lane	6.32	8.21	12.77%	0.0631
	Outside lane	6.43	8.49	15.07%	0.0650
	Total	6.32	8.26	13.76%	0.0646

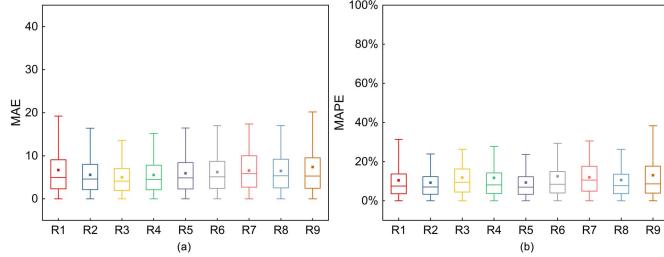


Fig. 6. The prediction results of the MDL model at different road sections. (a)MAE; (b)MAPE.

Table.III demonstrates the overall prediction result of the MDL model by dividing the lanes into inside lanes, middle lanes, and outside lanes. It is revealed that the MAEs, RMSEs, MAPEs, and TICs of the inside lanes and middle lanes are lower than those of outside lanes on the weekdays and weekends. The possible reason for this phenomenon is that the frequent speed change of the merging vehicles on the outside lanes may disturb the continuity of the traffic speed and increase the difficulty of prediction. In addition, it is found that the prediction results on the weekdays are more accurate than those on the weekends, which may be because the training data of the weekdays are much more than those of the weekends.

Fig. 7 displays the prediction results of the MDL model on a weekday and a weekend. As revealed in Fig. 7, the proposed model can fit the tendency and volatility of the traffic speeds well during the whole day. Even at the evening peak of the weekday when the traffic speed of lanes changes sharply, the MDL model can still capture the trend of the traffic speeds and make predictions stably and reliably.

C. Comparison of Prediction Results of Different Models

Table. IV presents the error metrics of different models in the task of making a one-step-ahead prediction ($T_h = 1$) by considering the 7-day results as a whole. It is found that the MDL model works best among these models, which outperforms the second-best predictor CNN-GRU with the improvements of 0.33 and 1.29% on MAE and MAPE, and outperforms the TM-CNN with improvements of 0.56 and 1.77% on RMSE and MAPE. The experimental results confirm that the proposed model can learn rich spatio-temporal information of traffic flow by utilizing the Conv-LSTM structure which is capable of mining the spatio-temporal feature syn-

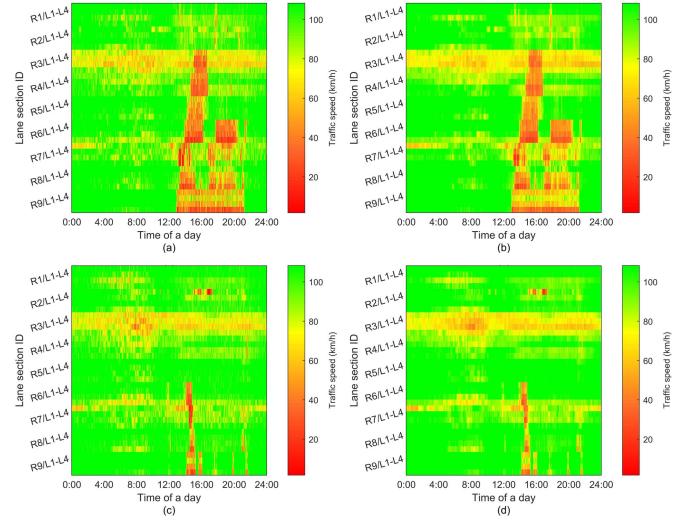


Fig. 7. The prediction results of the MDL model. (a) On a weekday (2014.2.24); (b) On a weekend (2014.3.1).

TABLE IV
THE OVERALL PREDICTION PERFORMANCE OF THE DIFFERENT MODELS

Model	MAE	RMSE	MAPE	TIC
BPNN	6.82	8.90	15.19%	0.0683
RBFNN	7.07	9.41	16.81%	0.0724
MLP	6.74	8.86	14.81%	0.0681
XGBoost	6.65	8.78	14.36%	0.0679
GRUNN	7.06	9.50	16.16%	0.0752
LSTMNN	7.00	9.27	15.61%	0.0713
LSTM-GRU	6.61	8.75	14.21%	0.0677
TM-CNN	6.59	8.59	14.03%	0.0659
CNN-GRU	6.47	8.37	13.55%	0.0639
MDL	6.14	8.03	12.26%	0.0620

chronously and efficiently from the target lane sections and their surrounding lane sections.

In addition, it is interesting to find that the BPNN, MLP, and XGBoost exhibit better performance than the LSTMNN and GRUNN in terms of the MAE, RMSE, MAPE, and TIC. This may be because the BPNN, MLP, and XGBoost consider both spatial and temporal variables as the input of the models, and the LSTMNN and GRUNN only consider the temporal feature of the target lane. For lane-level traffic speed prediction, the sudden change of traffic flow from surrounding lanes may affect the traffic pattern of the target lanes significantly. Hence, learning the spatial features in the short-term may be as important as or even more important than capturing the long-term temporal dependency.

The purposes of the MAEs, RMSEs, MAPEs and TICs are to measure the errors between the ground-truth values and the forecasted values. Meanwhile, the predicted accuracy of the spatio-temporal distribution is vital for lane-level traffic speed prediction as well. Thus, the average correlation (AC) is employed to measure the performance of different models on time and space dimensions. The AC of spatial distribution (AC_S) and the AC of temporal distribution (AC_T) are defined

TABLE V
COMPARISON OF THE AVERAGE CORRELATION OF DIFFERENT METHODS

Model	Inside lanes		Middle lanes		Outside lanes	
	AC_S	AC_T	AC_S	AC_T	AC_S	AC_T
BPNN	0.8611	0.7213	0.8998	0.7225	0.7041	0.7215
RBFNN	0.8463	0.6831	0.8722	0.6846	0.6770	0.6748
MLP	0.8653	0.7126	0.9037	0.7117	0.7029	0.7098
XGBoost	0.8556	0.7280	0.9021	0.7250	0.6910	0.7187
GRUNN	0.8501	0.7173	0.8919	0.7200	0.6899	0.7093
LSTMNN	0.8497	0.7089	0.8731	0.7118	0.6858	0.7058
LSTM-GRU	0.8597	0.7356	0.9031	0.7312	0.6906	0.7303
TM-CNN	0.8702	0.7307	0.9077	0.7460	0.7246	0.7258
CNN-GRU	0.8767	0.7602	0.9107	0.7643	0.7363	0.7567
MDL	0.8813	0.7715	0.9173	0.7713	0.7442	0.7672

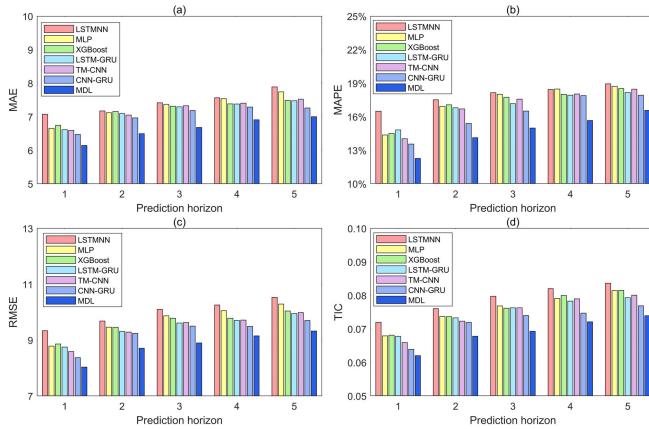


Fig. 8. Comparison of the different models with different prediction horizons.
(a)MAE;(b)MAPE;(c)RMSE;(4)TIC.

as follows:

$$AC_S = \frac{1}{n_s} \sum_{s=1}^{n_s} \text{Corr}(F_{s:}, T_{s:}) \quad (24)$$

$$AC_T = \frac{1}{n_t} \sum_{t=1}^{n_t} \text{Corr}(F_{:t}, T_{:t}) \quad (25)$$

where $\text{Corr}(\cdot)$ represents the Pearson correlation coefficient between the two vectors; AC_T represents the similarity between the predicted value and observed value at the same time interval; AC_S represents the similarity between the predicted value and observed value at the same lane; n_t and n_s denote the number of predicted vectors on the time dimension and space dimension respectively; $F_{s:}$ and $F_{:t}$ denote the predicted speed at space point s and at time interval t ; $T_{s:}$ and $T_{:t}$ denote the observed speed at space point s and at time interval t .

Table. V gives the comparison of the average correlation of different models on spatio-temporal dimension at different kinds of lanes. As shown in Table. V, the MDL has the highest AC_s and AC_t values among the models at different types of lanes, indicating that the proposed model has a powerful capability to extract the spatio-temporal features within traffic flow of lanes.

Forecasting traffic state over several time intervals in the future allows a wider range of applications to take advantage of prediction results. Fig. 8 provides the prediction performance

of models including the MLP, XGBoost, LSTMNN, TM-CNN, LSTM-GRU, CNN-GRU, and MDL model with the prediction horizon T_h increasing from 1 to 5, which correspond to the future time intervals from 2 min to 10 min. There is a clear phenomenon that each model performs better in near-term future time intervals. It can be found that the prediction accuracy of the MDL model is always higher than those of benchmark models with different prediction horizons and the longer horizon did not deteriorate the prediction accuracy significantly. It is worth noting that when the prediction horizon reaches up to 5, the prediction error of the MDL model is still less than that of the LSTMNN with the prediction horizon set as 1.

D. Sensitivity Analysis of the Parameters

In this section, we further conduct the sensitivity analysis and parameter tuning on the MDL model, where four kinds of parameters are investigated, including the input time window, the number of filters, filter size, and the layer structure. For each training process in the comparison experiment, the number of training epochs is set as 50 and the batch size is set as 64. The prediction horizon is chosen to be 1 ($T_h = 1$).

At first, we compare the MDL models with the input time window (T_b) increasing from 5 to 30 with a step of 5 for the speed and volume variables. The filter number and filter size are set as 10 and 3×3 respectively. Table.VI reveals that the accuracy and fitness degree of the MDL model rises with the input time window increasing from 5 to 15. However, when the input time windows become further larger ($T_b > 15$), the MAEs, RMSEs, MAPEs, and TICs remain relatively stable and the time consumption rises a lot. Therefore, it is supposed that if the input time window for speed and volume are both chosen to be 15, the input tensors may contain sufficient traffic flow information for forecasting lane-level traffic speeds in the future. The results further confirm that the FCA-MIC method can effectively measure the relevance of the variables and help the MDL model select the input time window.

As the filter size of the Conv-LSTM layer decides the view of extracting the spatial features, it is necessary to learn the relationship between the model performance and the filter size. Hence, we examine the MDL model with the different filter sizes including 1×1 , 2×2 , 3×3 , and 4×4 . Meanwhile, the L_s and L_q are both set as 1. The input time window for the two variables are fixed as 15. From Fig. 9(a)-(d), it can be observed that the MDL models with filter sizes of 2×2 and 3×3 are more accurate and stable than those with filter sizes of 1×1 and 4×4 when they have the same number of filters. This may be because the 2×2 and 3×3 filters can better extract the spatio-temporal influence of the adjacent lanes on the target lane section. Note that if the filter size is fixed, the accuracy and the fitting degree of the MDL model firstly increase and then drop a little with the number of filters increasing from 5 to 30. Fig. 9(e) indicates that the training time increases remarkably with the increase of the filter number and filter size. Therefore, 10-15 filters with sizes of 2×2 or 3×3 are ideal settings for the Conv-LSTM layers in the MDL model based on the prediction accuracy and training efficiency.

TABLE VI

COMPARISON OF THE PREDICTION PERFORMANCE OF THE MDL MODELS WITH DIFFERENT INPUT TIME WINDOWS

Input time window	MAE	RMSE	MAPE	TIC	Training time(s)
$T_b = 5$	6.30	8.22	13.37%	0.0640	628.24
$T_b = 10$	6.16	8.05	12.95%	0.0627	1087.68
$T_b = 15$	6.14	8.03	12.26%	0.0620	1503.52
$T_b = 20$	6.11	8.01	12.22%	0.0624	2557.39
$T_b = 25$	6.10	8.02	12.13%	0.0625	2930.02
$T_b = 30$	6.11	8.03	12.38%	0.0625	3502.60

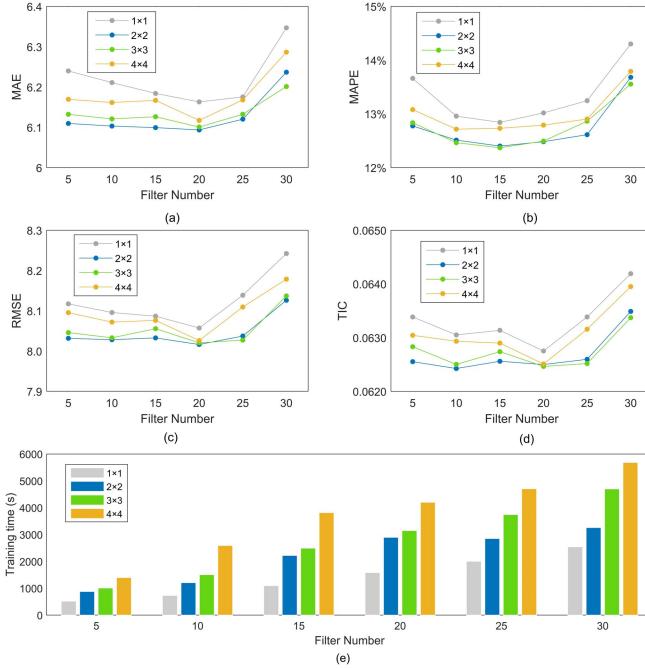


Fig. 9. Prediction performance of the MDL models with different numbers of filters and filter size. (a)MAE; (b)MAPE; (c)RMSE; (d)TIC; (e) Training time.

In addition to the input time window, filter size, and the number of filters, the layer structure of the MDL model may be another factor that affects the model performance. Table.VII compares the prediction results of the MDL models with different L_s and L_q under the condition that the number of filters and the filter size are set as 15 and 3×3 . It is illustrated that with more Conv-LSTM layers stacked in the MDL structure, the MAEs, RMSEs, MAPEs, and TICs have no obvious improvements while the training time increases sharply. Consequently, the structure with one Conv-LSTM layer can be efficient to capture the spatio-temporal characteristics of features for the lane-level speed forecasting.

VI. CONCLUSION

Lane-level short-term traffic state forecasting is of great benefit to both CAVs and HVs in the CAVH system. In this article, we establish a mixed deep learning model named MDL for predicting lane-level traffic speeds. The proposed architecture integrates the Conv-LSTM layers, convolutional layers, and a dense layer to improve prediction performance,

TABLE VII

COMPARISON OF THE PREDICTION PERFORMANCE OF THE MDL MODELS WITH DIFFERENT NUMBERS OF CONV-LSTM LAYERS

Layer structure	MAE	RMSE	MAPE	TIC	Training time(s)
$L_s = L_q = 1$	6.13	8.06	12.37%	0.0627	2491.07
$L_s = 2$; $L_q = 1$	6.10	8.02	12.13%	0.0633	5249.13
$L_s = 1$; $L_q = 2$	6.18	8.10	12.19%	0.0645	5374.72
$L_s = L_q = 2$	6.11	8.03	12.60%	0.0625	9310.68

and is fed with input tensors of traffic flow. To choose the efficient input time window for the MDL model, the FCA-MIC method is put forward to measure the relevance between the historical traffic flow and traffic speeds to be predicted. In addition, the real-world lane-level traffic flow data collected by the RTMSs located at 36 lane sections of 4th ring road in Beijing with the updating frequency of 2 min were utilized to examine the proposed MDL model. The prediction accuracy, fitting degree, and space-time distributions of the proposed model are compared with those of the state-of-the-art models including the BPNN, RBFNN, MLP, XGBoost, LSTMNN, GRUNN, LSTM-GRU, TM-CNN, and CNN-GRU model. Furthermore, the sensitivity of critical parameters in the MDL model is analyzed.

Some useful conclusions can be drawn in this study. 1) The proposed MDL model can capture the volatility and spatio-temporal dependency of lane-level traffic flow efficiently on different types of lanes during both weekdays and weekends. The prediction errors of the inside lanes and middle lanes are lower than those of the outside lanes. 2) Comparisons on the testing data reveal that the MDL model outperforms the benchmark models in terms of accuracy, stability, and spatio-temporal distributions. 3) With longer prediction horizon, the MDL model maintains a stable performance and prediction accuracy does not deteriorate obviously. 4) The sensitivity analysis indicates that the input time window, filter number, filter size, and the layer structure can affect the performance of the MDL model. The structure consisting of a Conv-LSTM layer with 10-15 filters and the filter size of 2×2 or 3×3 can achieve satisfactory prediction results with acceptable time consumption.

For future work, it is valuable to further study how to reduce the complexity and time consumption of our method. Meanwhile, the traffic flow from the ramps should be taken into account to improve the performance of the proposed model. Besides, it would be interesting to evaluate the adaptability of the proposed model on a larger and more complex urban network with hundreds of road links.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and editors whose insightful comments have helped significantly to improve the quality of this study.

REFERENCES

- [1] N. Zhang, F.-Y. Wang, F. Zhu, D. Zhao, and S. Tang, "DynaCAS: Computational experiments and decision support for ITS," *IEEE Intell. Syst.*, vol. 23, no. 6, pp. 19–23, Nov. 2008.

- [2] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
- [3] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.
- [4] B. Ran *et al.*, "Connected automated vehicle highway systems and methods," U.S. Patent 10380886 B2, Aug. 13, 2019.
- [5] Y. Gu, W. Lu, L. Qin, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: A fusion deep learning model," *Transp. Res. C, Emerg. Technol.*, vol. 106, pp. 1–16, Sep. 2019.
- [6] Z. Yao, R. Hu, Y. Wang, Y. Jiang, B. Ran, and Y. Chen, "Stability analysis and the fundamental diagram for mixed connected automated and human-driven vehicles," *Phys. A, Stat. Mech. Appl.*, vol. 533, pp. 1–16, Nov. 2019.
- [7] D. Tian, G. Wu, P. Hao, K. Boriboonsomsin, and M. J. Barth, "Connected vehicle-based lane selection assistance application," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 7, pp. 2630–2643, Jul. 2019.
- [8] T. Song, N. Capurso, X. Cheng, J. Yu, B. Chen, and W. Zhao, "Enhancing GPS with lane-level navigation to facilitate highway driving," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 4579–4591, Jun. 2017.
- [9] C. F. Daganzo, "A behavioral theory of multi-lane traffic flow. Part II: Merges and the onset of congestion," *Transp. Res. B, Methodol.*, vol. 36, no. 2, pp. 159–169, Feb. 2002.
- [10] P. G. Michalopoulos, D. E. Beskos, and Y. Yamauchi, "Multilane traffic flow dynamics: Some macroscopic considerations," *Transp. Res. B, Methodol.*, vol. 18, nos. 4–5, pp. 377–395, Aug. 1984.
- [11] C. F. Daganzo, "A behavioral theory of multi-lane traffic flow. Part I: Long homogeneous freeway sections," *Transp. Res. B, Methodol.*, vol. 36, no. 2, pp. 131–158, Feb. 2002.
- [12] Q. Liu, B. Wang, and Y. Zhu, "Short-term traffic speed forecasting based on attention convolutional neural network for arterials," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, pp. 1–18, Sep. 2018.
- [13] Y. Liu, Z. Liu, H. L. Vu, and C. Lyu, "A spatio-temporal ensemble method for large-scale traffic state prediction," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 35, no. 1, pp. 26–44, Jun. 2019.
- [14] T. T. Tchrakian, B. Basu, and M. O'Mahony, "Real-time traffic flow forecasting using spectral analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 519–526, Jun. 2012.
- [15] M.-C. Tan, S. C. Wong, J.-M. Xu, Z.-R. Guan, and P. Zhang, "An aggregation approach to short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 60–69, Mar. 2009.
- [16] R. Ke, W. Li, Z. Cui, and Y. Wang, "Two-stream multi-channel convolutional neural network for multi-lane traffic speed prediction considering traffic volume impact," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2674, no. 4, pp. 459–470, Mar. 2020.
- [17] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 166–180, May 2018.
- [18] H. Yu, Z. Wu, S. Wang, Y. Wang, and X. Ma, "Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks," *Sensors*, vol. 17, no. 7, pp. 1501–1517, 2017.
- [19] L. Li, L. Qin, X. Qu, J. Zhang, Y. Wang, and B. Ran, "Day-ahead traffic flow forecasting based on a deep belief network optimized by the multi-objective particle swarm algorithm," *Knowl.-Based Syst.*, vol. 172, pp. 1–14, May 2019.
- [20] S. Clark, "Traffic prediction using multivariate nonparametric regression," *J. Transp. Eng.*, vol. 129, no. 2, pp. 161–168, Mar. 2003.
- [21] Y. Xie, Y. Zhang, and Z. Ye, "Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 22, no. 5, pp. 326–334, Jul. 2007.
- [22] L. Li, X. Chen, Z. Li, and L. Zhang, "Freeway travel-time estimation based on temporal-spatial queueing model," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1536–1541, Sep. 2013.
- [23] J. Guo, W. Huang, and B. M. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 50–64, Jun. 2014.
- [24] M. Van Der Voort, M. Dougherty, and S. Watson, "Combining Kohonen maps with ARIMA time series models to forecast traffic flow," *Transp. Res. C, Emerg. Technol.*, vol. 4, no. 5, pp. 307–318, Oct. 1996.
- [25] B. M. Williams, "Multivariate vehicular traffic flow prediction: Evaluation of ARIMAX modeling," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1776, no. 1, pp. 194–200, Jan. 2001.
- [26] Y. Kamarianakis and P. Prastacos, "Space-time modeling of traffic flow," *Comput. Geosci.*, vol. 31, no. 2, pp. 119–133, Mar. 2005.
- [27] S. Cheng, F. Lu, P. Peng, and S. Wu, "Short-term traffic forecasting: An adaptive ST-KNN model that considers spatial heterogeneity," *Comput., Environ. Urban Syst.*, vol. 71, pp. 186–198, Sep. 2018.
- [28] W.-C. Hong, "Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm," *Neurocomputing*, vol. 74, nos. 12–13, pp. 2096–2107, Jun. 2011.
- [29] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6164–6173, Apr. 2009.
- [30] M. Shuai, L. Han, K. Xie, G. Song, and M. A. Xiujun, "An adaptive traffic flow prediction mechanism based on locally weighted learning," *Acta Sci. Nat. Univ. Pekinensis*, vol. 46, no. 1, pp. 64–68, Jan. 2010.
- [31] H. Tan, Y. Wu, B. Shen, P. J. Jin, and B. Ran, "Short-term traffic prediction based on dynamic tensor completion," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2123–2133, Aug. 2016.
- [32] K. Tang, S. Chen, Z. Liu, and A. J. Khattak, "A tensor-based Bayesian probabilistic model for citywide personalized travel time estimation," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 260–280, May 2018.
- [33] F. Guo, R. Krishnan, and J. Polak, "The influence of alternative data smoothing prediction techniques on the performance of a two-stage short-term urban travel time prediction framework," *J. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1–13, Jan. 2017.
- [34] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach," *Transp. Res. C, Emerg. Technol.*, vol. 13, no. 3, pp. 211–234, Jun. 2005.
- [35] J. Tang, L. Fang, Y. Zou, W. Zhang, and Y. Wang, "An improved fuzzy neural network for traffic speed prediction considering periodic characteristic," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 1–11, Sep. 2017.
- [36] K. Y. Chan, T. S. Dillon, J. Singh, and E. Chang, "Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and Levenberg–Marquardt algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 644–654, Jun. 2012.
- [37] H. Yin, S. C. Wong, J. Xu, and C. K. Wong, "Urban traffic flow prediction using a fuzzy-neural approach," *Transp. Res. C, Emerg. Technol.*, vol. 10, no. 2, pp. 85–98, Apr. 2002.
- [38] A. Rahimizadeh, E. Ghorbani, and S. Rahimizadeh, "Daily network traffic prediction based on backpropagation neural network," *Spine*, vol. 8, no. 7, pp. 1278–1281, Dec. 2014.
- [39] A. Kuang and Z. Huang, "Short-term traffic flow prediction based on RBF neural network," *Syst. Eng.*, vol. 22, no. 2, pp. 63–65, Apr. 2004.
- [40] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011.
- [41] J. Ke, H. Zheng, H. Yang, and X. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 591–608, Dec. 2017.
- [42] L. N. N. Do, H. L. Vu, B. Q. Vo, Z. Liu, and D. Phung, "An effective spatial-temporal attention based neural network for traffic flow prediction," *Transp. Res. C, Emerg. Technol.*, vol. 108, pp. 12–28, Nov. 2019.
- [43] L. Li, X. Qu, J. Zhang, Y. Wang, and B. Ran, "Traffic speed prediction for intelligent transportation system based on a deep feature fusion model," *J. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 605–616, Mar. 2019.
- [44] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, Mar. 2017.
- [45] N. G. Polson and V. O. Sokolov, "Deep learning for short-term traffic flow prediction," *Transp. Res. C, Emerg. Technol.*, vol. 79, pp. 1–17, Jun. 2017.
- [46] J. Wang, R. Chen, and Z. He, "Traffic speed prediction for urban transportation network: A path based deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 100, pp. 372–385, Mar. 2019.
- [47] Y. Jia, J. Wu, and M. Xu, "Traffic flow prediction with rainfall impact using a deep learning method," *J. Adv. Transp.*, vol. 2017, pp. 1–10, Aug. 2017.
- [48] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 187–197, May 2015.
- [49] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proc. 31st Youth Acad. Annu. Conf. Chin. Assoc. Autom. (YAC)*, Nov. 2016, pp. 324–328.

- [50] S. Yang, S. Shi, X. Hu, and M. Wang, "Spatiotemporal context awareness for urban traffic modeling and prediction: Sparse representation based variable selection," *PLoS ONE*, vol. 10, no. 10, Oct. 2015, Art. no. e0141223.
- [51] S. Deng, S. Jia, and J. Chen, "Exploring spatial-temporal relations via deep convolutional neural networks for traffic flow prediction with incomplete data," *Appl. Soft Comput.*, vol. 78, pp. 712–721, May 2019.
- [52] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2191–2201, Oct. 2014.
- [53] H.-F. Yang, T. S. Dillon, and Y.-P.-P. Chen, "Optimized structure of the traffic flow forecasting model with a deep learning approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2371–2381, Oct. 2017.
- [54] X. Ma, H. Yu, Y. Wang, and Y. Wang, "Large-scale transportation network congestion evolution prediction using deep learning theory," *PLoS ONE*, vol. 10, no. 3, Mar. 2015, Art. no. e0119044.
- [55] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. NIPS*, Jun. 2015, pp. 802–810.
- [56] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [57] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [58] D. N. Reshef *et al.*, "Detecting novel associations in large data sets," *Science*, vol. 334, no. 6062, pp. 1518–1524, Dec. 2011.
- [59] F. Zhan, X. Wan, Y. Cheng, and B. Ran, "Methods for multi-type sensor allocations along a freeway corridor," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 134–149, Jul. 2018.
- [60] *Highway Capacity Manual*, Nat. Res. Council Transp. Res. Board, Washington, DC, USA, 2000.
- [61] J. Song, L. Wei, and Y. Ming, "A method for simulation model validation based on Theil's inequality coefficient and principal component analysis," *Commun. Comput. Inf. Sci.*, vol. 402, pp. 126–135, Nov. 2013.
- [62] J. Z. Zhu, J. X. Cao, and Y. Zhu, "Traffic volume forecasting based on radial basis function neural network with the consideration of traffic flows at the adjacent intersections," *Transp. Res. C, Emerg. Technol.*, vol. 47, pp. 139–154, Oct. 2014.
- [63] X. Dong, T. Lei, S. Jin, and Z. Hou, "Short-term traffic flow prediction based on XGBoost," in *Proc. IEEE 7th Data Driven Control Learn. Syst. Conf. (DDCLS)*, May 2018, pp. 854–859.



Wenqi Lu received the B.E. degree in transportation engineering from Hohai University, Nanjing, China, in 2016, and the M.S. degree in transportation planning and management from Beijing Jiaotong University, Beijing, China, in 2019. He is currently pursuing the Ph.D. degree with the School of Transportation, Southeast University, Nanjing. His current research interests include connected and automated vehicles, traffic theory, and intelligent transportation systems.



Yikang Rui received the Ph.D. degree in geographic information science from the KTH Royal Institute of Technology. He was a Post-Doctoral Fellow with Nanjing University from 2014 to 2016. He is currently an Associate Research Fellow with the School of Transportation, Southeast University, Nanjing, China. His current research interests include spatio-temporal data model, traffic network analysis, and vehicle-road collaborative decision planning.



Bin Ran currently serves as the Director of the Joint Research Institute on Internet of Mobility, founded by Southeast University, China, and the University of Wisconsin-Madison, Madison, WI, USA. He also holds the title of a National Distinguished Expert in China and the Vilas Distinguished Achievement Professor and the Director of the Intelligent Transportation System (ITS) Program at the University of Wisconsin-Madison. His research interests include intelligent transportation systems and connected automated vehicle highway (CAVH) systems. He is also the Founding President of the Chinese Overseas Transportation Association (COTA). He is also an Associate Editor of the *Journal of Intelligent Transportation Systems*.