



Short-term prediction of lane-level traffic speeds: A fusion deep learning model[☆]



Yuanli Gu^a, Wenqi Lu^b, Lingqiao Qin^{c,*}, Meng Li^a, Zhuangzhuang Shao^a

^a Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, Beijing Jiaotong University, 3 Shangyuan Cun, Haidian District, Beijing 100044, PR China

^b School of Transportation, Southeast University, 2 Southeast University Road, Jiangning District, Nanjing 211189, PR China

^c TOPS Laboratory, Department of Civil and Environmental Engineering, University of Wisconsin-Madison, 1415 Engineering Dr., Madison, WI 53706, USA

ARTICLE INFO

Keywords:

Short-term prediction
Traffic speed of lanes
Long short-term memory neural network
Gated recurrent unit neural network

ABSTRACT

Accurate and robust short-term traffic prediction is an important part of advanced traveler information systems. With the development of intelligent navigation and autonomous driving, it is necessary to explore the lane-level predictions of traffic speeds. However, most existing traffic prediction models concentrate on forecasting the traffic flow characteristics of the entire road sections rather than those of certain lanes. This paper proposes a fusion deep learning (FDL) model to predict lane-level traffic speed. First, the entropy-based grey relation analysis is introduced to choose lane sections that are strongly correlated with the lane section to be predicted. Second, a two-layer deep learning framework is established by combining the long short-term memory (LSTM) neural network and the gated recurrent unit (GRU) neural network. Third, the ground-truth data of several lane sections captured by remote traffic microwave sensors (RTMS) on the 2nd Ring Road of Beijing are utilized to examine the FDL model and compare it with several benchmark models. The experimental result indicates that in addition to capturing the fluctuations of traffic speed at the lane level, the FDL model has better performance than the benchmark models in terms of prediction accuracy and stability.

1. Introduction

Short-term traffic prediction can be defined as the process of estimating the anticipated traffic state at a future time. Real-time and accurate short-term traffic prediction can help traffic managers take effective strategies for alleviating traffic congestion and optimizing traffic assignments. Similarly, individual travelers can use the prediction information to make their mobility decisions more effectively. Therefore, short-term traffic prediction is a crucial issue in an Intelligent Transportation System (ITS).

With the development of Big Data and connected vehicle (CV) technology, the traditional approaches for traffic flow prediction, which takes the entire road sections as the research object, cannot satisfy the requirement of new technologies. The objects of short-term traffic state prediction are urgent to be more sophisticated as the short-term traffic states of lanes in a road section differ in traffic speed, volume and occupancy (Kwon et al., 2003). Therefore, the lane-level traffic state forecasting is becoming an essential issue of Intelligent Vehicle Infrastructure Cooperative Systems which is a new development of ITS. Making short-term prediction of

[☆] This article belongs to the Virtual Special Issue on “Traffic flow modeling”.

* Corresponding author.

E-mail addresses: yugu@bjtu.edu.cn (Y. Gu), lingqiao.qin@wisc.edu (L. Qin).

traffic speed of lanes can describe the future state of urban expressways or highways more exquisitely and efficiently, which is fundamental for high-precision navigation, intelligent driving, active traffic management, and other advanced applications. Specifically, accurately predicting the traffic speed of lane sections can support the lane-changing behavior of CVs and provide CVs with lane-level route guidance. Overall, the aforementioned technique can increase the capacity and safety of high-grade roads with the help of lane-level traffic state prediction.

In essence, lane-level traffic flow prediction can learn from the traditional traffic flow prediction methods. Over the past few decades, many different traffic flow prediction approaches have emerged, including statistical time series methods (Nicholson and Swann, 1974), traffic flow theory-based methods (Kai and Schreckenberg, 1992; Takenouchi et al., 2019), artificial intelligence models (Lv et al., 2015; Polson and Sokolov, 2017) and hybrid methods (Guo et al., 2018; Wang et al., 2014; Zhang et al., 2014). Although many advanced methods have emerged to predict the traffic speed or volume of road sections, limited research efforts have been implemented on short-term prediction for lane-level traffic characteristics. The limitation of research on lane-level prediction may result from two causes:

- (1) Difficulty for researchers to collect high-quality data to describe the traffic state of specific lanes;
- (2) The traffic characteristics of the target lane sections are influenced by adjacent, upstream, or downstream lane sections, which needs efficient methods to extract the valid information.

To overcome the problem (1) and forecast the traffic state of lanes efficiently, travel speed was chosen to measure the traffic state of urban expressway. Compared with travel time and occupancy, traffic speed can be readily collected by loop detectors, global positioning systems devices installed on floating cars and remote traffic microwave sensors (RTMS) mounted on the side of the road. As the most popular non-intrusive traffic detectors, RTMS do not cause temporary lane closures for installation or traffic flow interruption. They can detect traffic volume, occupancy and speed in multiple lanes without causing interference (Zhan et al., 2018). In addition, the research conducted by Yu and Prevedouros (2013) revealed that the speed measurement of RTMS can achieve up to 95% accuracy, and is higher than that of a single loop detector. Consequently, we utilized the traffic speed data of lanes captured by RTMS as the predictor in this study.

Recently, deep learning neural networks and their combinations have led a series of breakthroughs for application on complex and big datasets such as images, languages, and speech (Irie et al., 2016; Shin et al., 2016; Wu et al., 2017). With the help of deep learning technique and considering the complex spatio-temporal correlation described as (2), we propose a fusion deep learning (FDL) model, which combines the feature extraction method and a two-layer deep learning structure, to achieve the short-term prediction for traffic speed of lanes. Based on the aforementioned discussion, the improvement and contributions of this study mainly involve the following four aspects:

- (1) Entropy-based grey relation analysis is introduced to extract vital spatial feature variables from the candidate variables and coordinates them in the input matrix of one end-to-end learning structure for forecasting the travel speed of lanes.
- (2) A two-layer deep learning structure is proposed, which combines the LSTM and the GRU, in order to better capture the spatio-temporal features of the travel speed of lanes.
- (3) Validated by real-world traffic speed data of the lanes on urban expressways 2nd Ring Road of Beijing, the proposed FDL model outperforms both the traditional and the state-of-the-art benchmark algorithms, including two deep learning methods.
- (4) The analysis of the influence of spatial features and the RNN-based structure on the FDL model is presented, which illustrates that the accuracy and stability of the FDL model are influenced by these two factors to some extent.

The rest of the paper is organized as follows: Section 2 reviews the existing research on approaches for short-term traffic flow prediction. Section 3 presents the methodology to realize the proposed FDL model for lane-level speed prediction. Section 4 provides a detailed description of the experiment, assessment criteria, and benchmark models. In Section 5, the experimental results are analyzed and compared with the benchmark models. Section 6 presents the conclusions of this study and future work.

2. Literature review

Overall, the existing traffic flow prediction methods can be divided into three categories, including parametric methods, non-parametric methods, and deep learning methods.

2.1. Parametric methods

Parametric methods have a definite model structure, and these models can be calibrated with ground truth traffic data. Parametric methods also offer explicit formulas to give valuable interpretations of traffic characteristics. Classical parametric methods mainly consist of exponential smoothing (Chan et al., 2012), autoregressive integrated moving average model (ARIMA) (Hamed et al., 1995) and multivariate time series models (Min and Wynter, 2011). Specifically, many researchers applied the data analysis techniques developed by Box et al. (1976) to predict short-term freeway traffic flow and found that the ARIMA model can represent freeway time-series data in a highly accurate manner. Inspired by the superior capability to cast traffic dynamic of ARIMA, numerous extended ARIMA models (Chen et al., 2011; Ding et al., 2010) began to emerge. However, parametric approaches make strong assumptions about traffic dynamics and they are inferior at predicting the traffic flow with irregular fluctuations.

2.2. Non-parametric methods

Different from parametric methods, flexible non-parametric methods can provide a distinct way to achieve traffic flow prediction since their structure and parameters are not fixed. Specially, these methods have relaxed assumptions for inputs, and they are more capable of processing outliers, missing data, and noisy data (Karlaftis and Vlahogianni, 2011). There are many non-parametric approaches which exhibit superior performances in traffic forecasting, including support vector machines (Wang and Shi, 2013; Castro-Neto et al., 2009; Vanajakshi and Rilett, 2007), and Kalman filter (KF) (Guo et al., 2014; Lippi et al., 2013) research and artificial neural networks (ANN) (Karlaftis and Vlahogianni, 2011). Note that ANN and its variants (Li et al., 2019; Laña et al., 2019) have exclusive advantages of identifying underlying non-linear relationships inside historical traffic data. Dougherty and Cobbett (1997) studied the back-propagation neural network (BPNN) to forecast the speed and occupancy of traffic flow, and achieved promising results. Kuang and Huang (2004) set up a radial basis function neural network (RBFNN) to predict traffic volume of a freeways. Khotanzad and Sadek (2003) combined a multi-layer perceptron and a fuzzy neural network to high-speed network traffic prediction, which indicates that ANN performs better than autoregressive models. Besides, some researchers used non-parametric methods for lane-level traffic flow prediction. For instance, Ghosh et al. (2009) established a multivariate forecasting algorithm based on ANN to predict the predict traffic flow and speed of lanes in freeways, and the traffic flow and speed observations from uncongested and congested regimes were regime-adjusted to ensure consistent system dynamics. Raza and Zhong (2017) introduced the genetic algorithms to generate the input which is highly correlated with the output. In their study, the multi-layer perceptron (MLP) and locally weighted regression (LWR) were used to predict 5-min short-term traffic speed for lanes of an urban road. Though multiple variables were used for prediction, the non-parametric based methods were weak in learning the spatio-temporal features of lane-level traffic flow.

2.3. Deep learning methods

With the rapid development of data storage and data processing techniques, traditional non-parametric methods have difficulty in dealing with multi-source data (Vlahogianni et al., 2014). The short-term traffic predictions are shifting from non-parametric methods to deep learning methods (Wu et al., 2018; Cui et al., 2018; Huang et al., 2014; Li et al., 2019). For instance, a novel stacked auto-encoder model was established by Yang et al. (2016) to learn the hierarchical representation of urban traffic flow. Zhang et al. (2018) proposed a convolutional neural network to fully utilize the topological characteristics of urban crowd flow.

One challenge of traffic forecasting is to capture spatio-temporal correlations of the traffic flow. In terms of temporal correlations, recurrent neural network (RNN) (Van Lint et al., 2002) plays a particularly effective role in capturing traffic dynamic evolution as it can make an effective use of historical input data via its inner memory units. However, the conventional RNN models exhibit a weakness on handling long temporal dependency issues due to the problems of vanishing gradient and exploding gradient. To address that, the long short-term memory (LSTM) was introduced to the field of traffic prediction by Ma et al. (2015). Distinguished from aforementioned RNN-based variants, LSTM is capable of recognizing inherent correlations inside long-time span traffic inputs and choosing the optimal time lags automatically within the training process. After that, LSTM has been widely employed into the traffic information prediction field and shown some promising results (Jia et al., 2017; Zhao et al., 2017; Wang et al., 2019). One of the famous LSTM variants is the GRU model, which has fewer neurons but can achieve equal or better performance than LSTM (Cho et al., 2014). The GRU model has been used in the context of machine translation, but it is still few used for traffic prediction (Wu et al., 2018). In terms of spatial correlations, Ke et al. (2019) proposed a multi-channel convolutional neural network (CNN) model for multi-lane speed prediction considering the impact and traffic volume.

However, neither CNN nor LSTM are perfect models for spatio-temporal forecasting problems. CNN fails to capture the temporal dependencies, while LSTM is incapable of characterizing local spatial correlations. In order to capture spatio-temporal dependencies simultaneously in an end-to-end structure, many studies have emerged in recent years. Ma et al. (2017) established a deep CNN for large-scale traffic network speed prediction with space-time matrix converted to an image as the input of the CNN. Liu et al. (2018) proposed an attention CNN and used three-dimensional data matrices constructed by flow, speed, and occupancy to predict traffic speed.

However, extracting spatial features from the lane sections of urban expressways or highways is different from that of large-scale networks. Utilizing the CNN model to capture spatial dependencies of lane sections of high-grade roads will cause many unnecessary parameters and interfere with forecasting accuracy. Consequently, in this paper a novel deep learning architecture, the fusion deep learning (FDL) model, is established to capture the spatio-temporal characteristics in short-term speed prediction of lane sections. To reduce the computation complexity, firstly an entropy-based grey relation analysis is introduced to characterize the spatial variables for lane-level travel speed prediction. Then a deep learning structure with two kinds of RNN-based hidden units attempts to learn the spatio-temporal correlations of the extracted variables. In this way, an efficient model to make short-term prediction for traffic speed of lanes is obtained.

3. Methodology

3.1. Entropy-based Grey Relation Analysis (EGRAs)

Lane-level traffic flow on urban expressways have obvious spatial and temporal characteristics. The traffic speed of a lane section at a certain time interval is affected not only by the traffic speed of the lane section during previous time intervals, but also by the

traffic speed of adjacent, upstream, and downstream lane sections. In order to consider more spatial variables without consuming too much time during the training process of the proposed model, correlation analysis was applied to recognize the lane sections that have strong impacts on the target lane section.

There are many methods for correlation analysis between the target lane section and the candidate lane sections. One method is the entropy-based grey relation analysis (EGRA), which was first proposed by [Zhang et al. \(1995\)](#) to improve the deficiency of the grey relation analysis. The EGRA method is very effective in determining the critical factors that significantly influence certain defined objectives ([Wang et al., 2014](#)).

Assuming that S_0 is the target lane section which needs to be forecasted; $S_m, m = 0, 1, 2, \dots, M$ are the candidate lane sections that are related to S_0 , and the candidate lane sections usually contain the target lane section itself. M is the number of related lane sections, and it should be large enough to cover most of the lane sections which have an influence on S_0 ; $V_{S_0} = \{v_{S_0}(t) | t \in T\}$ represents the historical speed series at target lane section S_0 , and $V_{S_m} = \{v_{S_m}(t) | t \in T\}$ is the historical speed series at candidate lane section S_m ; T is the length of the historical traffic speed sequence, which represents the number of historical time intervals; $v_{S_0}(t)$ and $v_{S_m}(t)$ are the speed values of lanes sections S_0 and S_m respectively at the same time interval t .

The grey relational coefficient between target lane speed and candidate lane speed at one time interval can be calculated as following:

$$\gamma(v_{S_0}(t), v_{S_m}(t)) = \frac{\Delta min + \rho \Delta max}{|v_{S_0}(t) - v_{S_m}(t)| + \rho \Delta max} \quad (1)$$

where $\Delta min = \min_{m \in M} \min_t |v_{S_0}(t) - v_{S_m}(t)|$; $\Delta max = \max_{m \in M} \max_t |v_{S_0}(t) - v_{S_m}(t)|$; and ρ is the distinguishing coefficient which has a range of $[0, 1]$.

The grey relational coefficient $\gamma(v_{S_0}(t), v_{S_m}(t))$ calculated by (Eq. (1)) illustrates the relevance degree between the target lane speed $v_{S_0}(t)$ and the candidate lane speed $v_{S_m}(t)$ in sequences of the grey system. To satisfy the rule of grey entropy, the grey relation coefficient needs to be transformed into a grey relation density $p(m, t)$. Then grey relation density $p(m, t)$ can be obtained as following:

$$p(m, t) = \frac{\gamma(v_{S_0}(t), v_{S_m}(t))}{\sum_{t=1}^T \gamma(v_{S_0}(t), v_{S_m}(t))}, m = 0, 1, 2, \dots, M; \quad (2)$$

After obtaining the grey relation density $p(m, t)$, we can calculate the entropy of the grey relational coefficient of each of the candidate lane sections. The entropy demonstrates the degree of relevance of the grey relational coefficients and it can be represented as:

$$E(m) = \frac{-\sum_{t=1}^T p(m, t) \ln p(m, t)}{\ln T}, m = 0, 1, 2, \dots, M \quad (3)$$

where $-\sum_{t=1}^T p(m, t) \ln T$ shows the grey entropy between V_{S_0} and V_{S_m} ; $\ln T$ is the maximum entropy which makes the value of $E(m)$ between 0 and 1.

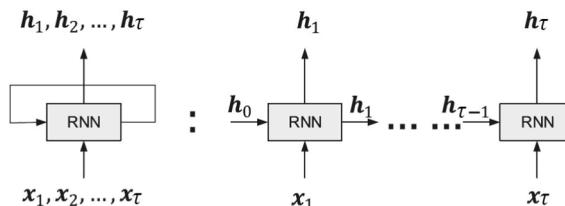
Finally, $E(m)$ is multiplied by the average of grey relational coefficient of candidate lane section S_m to gain the grey relevancy grade (GRG) which can be revealed as following:

$$GRG(m) = \frac{E(m)}{T} \sum_{t=1}^T \gamma(v_{S_0}(t), v_{S_m}(t)), m = 0, 1, 2, \dots, M \quad (4)$$

In this paper, the values of the GRGs are utilized as the numerical measurement of the relevancy between the speed series at the target lane section and at the candidate lane sections. According to the definition of the GRG, the larger the GRG is, the heavier the influence of the candidate lane section on the target section. After obtaining the GRG of each candidate lane section, the GRGs are ranked in decreasing order and the first λ candidate lane sections are chosen to reconstruct the input matrix of the deep learning model.

3.2. LSTM and GRU

Traditional ANNs lack the ability to capture the characteristics of time series since they ignore taking the temporal variables into



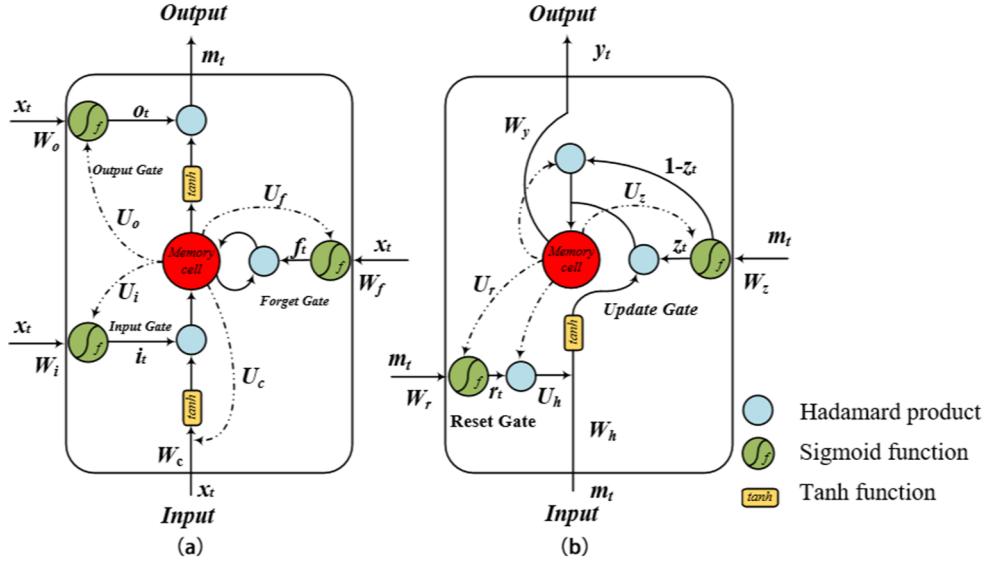


Fig. 2. The structure of memory blocks. (a) LSTM; (b) GRU.

consideration. In order to mitigate this shortcoming, a RNN is proposed, and the connection between the units in the RNN is organized by time series. The inner architecture of the RNN is demonstrated in Fig. 1.

For a hidden layer in the RNN, it receives a τ time-interval vector sequence $x = (x_1, x_2, \dots, x_\tau)$, and the output of the hidden layer is a sequence $h = (h_1, h_2, \dots, h_\tau)$. Note that x_t can be a vector or scalar and h_t may have different dimension than x_t . The hidden unit h_t at the time interval t stores the previous information including the hidden vector $h = (h_1, h_2, \dots, h_{t-1})$ and the input vector $(x_1, x_2, \dots, x_{t-1})$. Together with the input x_t at time interval t , h_t is passed to the following time interval $t + 1$. Hence, the RNN can memorize the information from multiple previous time intervals. Although the RNN exhibits strong ability in catching temporal characteristics, it fails to store information for a long-term memory.

To handle the aforementioned problems of the RNNs, a LSTM is proposed as a special RNN variant. The standard LSTM has an input vector sequence x and a hidden vector sequence C with τ time intervals. The only different component between a standard LSTM architecture and a RNN architecture lies in the memory block of the hidden unit cells.

As illustrated in Fig. 2(a), the memory block of the LSTM contains a memory cell and three multiplicative gates including an input gate, a forget gate, and an output gate, which are denoted as i_t , f_t , and o_t respectively at time interval t . The memory cell, which can be defined as C_t , is capable of memorizing the temporal state. In addition, the input gate and output gate control the input and output activations into the block, respectively. The forget gate enables the memory blocks to reset when the information is outdated. The i_t , f_t , o_t , C_t , and the output sequence of the hidden layer m_t can be calculated by following the equations:

$$f_t = \sigma(W_f x_t + U_f C_{t-1} + b_f) \quad (5)$$

$$i_t = \sigma(W_i x_t + U_i C_{t-1} + b_i) \quad (6)$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c C_{t-1} + b_c) \quad (7)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_{t-1} \quad (8)$$

$$o_t = \sigma(W_o x_t + U_o C_{t-1} + b_o) \quad (9)$$

$$m_t = o_t \circ \tanh(C_t) \quad (10)$$

where \circ refers to the Hadamard product, which calculates the element-wise products of two vectors, matrices, or tensors with the same dimensions. W_f , W_i , W_c , and W_o are weight matrices which connect the hidden layer input to gates; U_f , U_i , U_c , and U_o are weight matrices which connect the previous cell output state to gates and to the input cell state; b_f , b_i , b_c , and b_o are bias vectors. σ and \tanh are two non-linear activation functions, which are calculated as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (11)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (12)$$

Though the LSTM is an ideal structure to overcome the vanishing gradient of the RNN, the GRU, a simpler variant of the LSTM, is proposed in this study. It is also reported that the GRU achieves equal or better performance than the LSTM (Cho et al., 2014).

Compared with the LSTM, the memory block of the GRU is simpler, which can be shown in Fig. 2(b). The memory block only has two gates including the update gate and the reset gate, which are defined as z_t and r_t . Note that the update gate z_t determines whether the hidden state (H) will be updated with a new hidden state and the reset gate r_t controls the degree of state information in the previous moment which needs to be ignored. If the GRU has an input vector m_t , the z_t , r_t , H_t , and the output vector sequence of the hidden layer y_t can be calculated by following the equations:

$$z_t = \sigma(W_z m_t + U_z H_{t-1} + b_z) \quad (13)$$

$$r_t = \sigma(W_r m_t + U_r H_{t-1} + b_r) \quad (14)$$

$$\tilde{H}_t = \tanh[W_h m_t + U_h(r_t \circ H_{t-1} + b_h)] \quad (15)$$

$$H_t = (1 - z_t) \circ H_{t-1} + z_t \circ \tilde{H}_t \quad (16)$$

$$y_t = \sigma(W_y H_t) \quad (17)$$

where \circ refers to the Hadamard product; σ is the sigmoid function, which is set to be in the range of $[0, 1]$ to control the information flow; \tanh is a non-linear activation function squashing numbers to the range of $[-1, 1]$; W_z , W_r , W_h , W_y , U_z , U_r , and U_h are weight matrices; and b_z , b_r , and b_h are bias vectors. It is obvious the network tends to learn long-term memories if most update gates z are near 0 or if most reset gates r are near 0. Conversely, it tends to learn short-term memories.

3.3. Fusion Deep Learning (FDL) method

In this section, a novel fusion deep learning (FDL) model is proposed, which is demonstrated in Fig. 3 to realize the short-term traffic forecasting for lane speeds. Since the shallow single-layer LSTM or the GRU only capture short-term memories (Karpathy et al., 2015), a double-layer structure was established, including a LSTM layer and a GRU layer, to extract the long-term memories and short-term memories of traffic speed of lane sections in contrast to previous single layer models.

For a target lane section S_0 , the input of the FDL model is represented as $X_t = [x_{t-\tau}, x_{t-(\tau-1)}, \dots, x_{t-1}]^T$, and τ is the number of the time lag. $x_t = [v_{S_0}(t), \dots, v_{S_i}(t), \dots, v_{S_{\lambda-1}}(t)]$ is a vector containing the speed of the target lane section S_0 and the candidate lane sections S_i , $i = 1, 2, \dots, \lambda - 1$ at the time interval t . These candidate lane sections are selected from the feature extraction process using the EGRA. According to the result of the feature extraction, the reconstructed X_t can be characterized as a 2D speed matrix:

$$X_t = \begin{bmatrix} x_{t-1} \\ x_{t-2} \\ \vdots \\ x_{t-\tau} \end{bmatrix} = \begin{bmatrix} v_{S_0}(t-1), \dots, v_{S_i}(t-1) \dots, v_{S_{\lambda-1}}(t-1) \\ v_{S_0}(t-2), \dots, v_{S_i}(t-2) \dots, v_{S_{\lambda-1}}(t-2) \\ \vdots \\ v_{S_0}(t-\tau), \dots, v_{S_i}(t-\tau) \dots, v_{S_{\lambda-1}}(t-\tau) \end{bmatrix} \quad (18)$$

To achieve the goal of forecasting the speed of the target lane section at time interval t , the first hidden layer of the FDL model employs the LSTM to capture the spatio-temporal characteristics from variables of the input. The GRU is introduced as the second

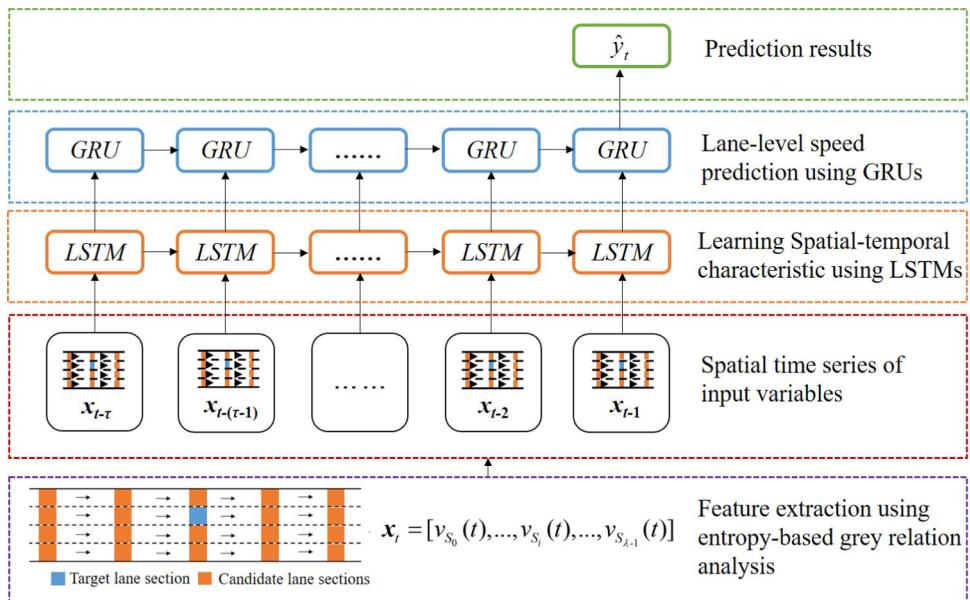


Fig. 3. The framework of the FDL model.

Table 1

Pseudo-code of the process to train a FDL.

Algorithm: The Training process of the FDL model

```

1 Input: Observations of historical lane-based speed  $V_{Sm} = \{v_{Sm}(t) | t \in T\}$  in training set
2      Time lag:  $\tau$ 
3      Spatial dimension:  $\lambda$ ;
4 Output: The FDL model with learnt parameters
5 Phase 1: Feature extraction using ERGA
6      for all candidate lane sections  $m (0 \leq m \leq M)$  do
7          Calculate GRG( $m$ ) according to Eqs. (1)–(4)
8      end for
9      Sort the GRG( $m$ ) in descending order and select  $\lambda$  candidate lane sections with larger GRGs
10 End Phase 1
11 Phase 2: Procedure LSTM-GRU Training
12      Initialize a null set:  $L \leftarrow \emptyset$ 
13      for all available time intervals  $t (1 \leq t \leq T)$  do
14           $x_t \leftarrow [v_{S_0}(t), v_{S_1}(t), \dots, v_{S_{\lambda-1}}(t)]$ 
15      end for
16      for all available time intervals  $t (\tau \leq t \leq T)$  do
17           $X_t \leftarrow [x_{t-1}, x_{t-2}, \dots, x_{t-\tau}]$ 
18          A training observation  $(X_t, v_{S_0}(t))$  is put into  $L$ 
19      end for
20      Initialize all the weighted and intercept parameters
21      repeat
22          Randomly extract a batch of samples  $L_b$  from  $L$ 
23          Estimate the parameters by the minimizing the objective function shown in Eq. (19) within  $L_b$ 
24      until convergence criterion met
25 End Phase 2

```

hidden layer to forecast the target speed at time interval t based on the output of the LSTM layer. During the training procedure of the FDL model, the training object is to minimize the mean squared error between the estimated speed values and ground-truth speed values. The objective function of the architecture is shown as follows:

$$\text{error} = E[(\hat{y}_t - y_t)^2] \quad (19)$$

where \hat{y}_t denotes the predicted speed of the FDL model and y_t is the ground-truth value.

The weight and bias can be learned through the training process. As shown in Table 1, the training process can be divided into two phases including the first phase for extracting the spatial features and the second phase for learning the parameters.

4. Experiment

4.1. Dataset description and preprocessing

To verify the performance of the proposed FDL model in terms of accuracy and stability, the RTMS data provided by the Beijing Municipal Commission of Transport are utilized as the dataset. The dataset contains the traffic speed information of several lanes on the 2nd Ring Road of Beijing, which were collected from Jan 6th, 2014 to Jan 19th, 2014 and from Feb 17th, 2014 to Feb 23rd, 2014 with an updating frequency of 2 min. The distance between the adjacent road sections which installed RTMSs is approximately 0.5 km, and the direction of the traffic is from south to north. A detailed description of the lane sections is shown in Table 2 and Fig. 4. Noted that there were 15120 records of each lane section with a data validity rate higher than 95%. To ensure a more reliable result, missing and erroneous records were properly remedied using time-adjacent records (Karlaftis and Vlahogianni, 2011).

The aggregated average speed in those 720 time intervals of a day, the aggregated averaged daily traffic speed of a week, and the average traffic speed of those 21 lane sections are given in Fig. 5. It can be seen that the different time intervals in a day have different

Table 2

Detailed information of the observation lane sections

Road section ID	Name of Road section	Number of lanes	Lane section ID
H2002	The North of the Dongzhimen Bridge	3	H2002-L1, H2002-L2, H2002-L3
H2003	The South of the Dongzhimen Bridge	3	H2003-L1, H2003-L2, H2003-L3
H2004	The North of the Dongjisitiao Bridge	3	H2004-L1, H2004-L2, H2004-L3
H2005	The South of the Dongjisitiao Bridge	3	H2005-L1, H2005-L2, H2005-L3
H2006	The North of the Chaoyangmen Bridge	3	H2006-L1, H2006-L2, H2006-L3
H2007	The South of the Chaoyangmen Bridge	3	H2007-L1, H2007-L2, H2007-L3
H2008	The North of the Chaoyangmen Bridge	3	H2008-L1, H2008-L2, H2008-L3

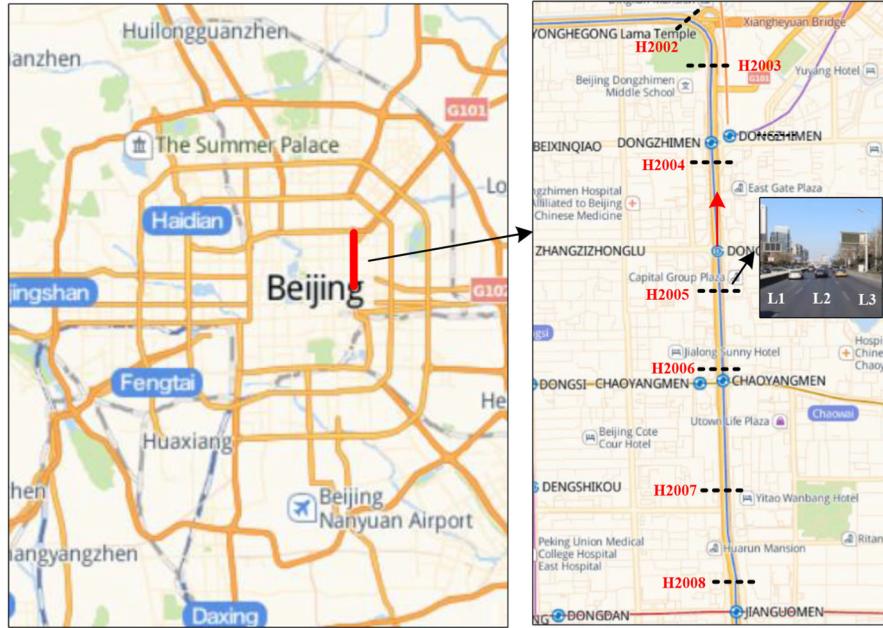


Fig. 4. The observation and experimental lane sections on 2nd Ring Road of Beijing.

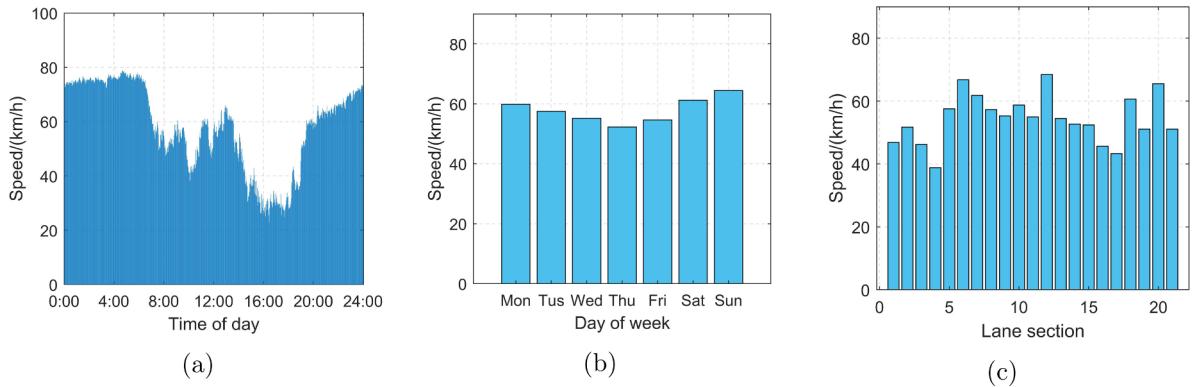


Fig. 5. The average speed of (a) 720 time intervals of a day; (b) different days in a week; (c) 21 lane sections.

average speeds and the working time intervals have a lower average speed. In addition, the distribution of average speeds during the week also varies, and the average speed of weekends is higher than that of weekdays. It is worth noting that average speed of the lane sections at the same road section are different.

In order to prevent the inconsistencies of data distribution and to avoid gradient explosions, the input data was processed by scaling the attributes to a specified maximum and minimum value (usually 1 – 0) by min-max normalization. The objective of the experiment is to predict the traffic speed of the three lane sections (H2005-L1, H2005-L2, H2005-L3) on the road section of the South of Dongsishitiao Bridge in the next time intervals. The three-week dataset is separated into two parts. The first part (6th Jan - 19th Jan) is used as training data, and the second part (17th Feb - 23rd Feb) is used as testing data. According to the sampling frequency, a single unit of the time lag is set as 2 min. Therefore if the time lag τ is set as 10, it means the FDL model utilizes a set of data with 10 time lags covering 20 min to predict the future speed value. In general, a shorter time lag is more suitable for short-term traffic forecasting, whereas a relatively long time lag is suitable for long-term traffic forecasting. Note that the spatial dimension denoted as λ is the number of related lane sections, which illustrates that the FDL model extracts λ feature lane sections to reconstruct the input matrix. Based on the aforementioned descriptions of the data, each sample of the input data X_t is a 2-D matrix with a dimension of $[\tau, \lambda]$.

4.2. Model implementation

In this section, the structure of the FDL model is comprised of two hidden layers for spatio-temporal features, and the number of hidden neuron in each hidden layer is 200. The number of training epochs is set to be 500, and the batch size is set to be 50. Note that

10% of training data is set as validation to reduce overfitting problem. The number of the spatial feature gained from the feature extraction process is 5 and time lag is chosen to be 10. Therefore, the dimension of the input matrix is [5, 10]. The optimizer is set as Adam which uses an adaptive learning rate for stochastic optimization.

Apart from the proposed model, several benchmark methods are also considered. These benchmark algorithms include one traditional time-series forecasting models (ARIMA), two non-parametric methods (LWR and KF), two state-of-the-art machine learning methods (MLP and RBFNN), and two deep learning approaches (CNN and LSTM). The settings of all these methods above are given as follows.

The ARIMA model (Levin and Tsao, 1980) integrates the autoregressive (AR), integrated (I), and moving average (MA) parts and considers trends, cycles, and non-stationary characteristics of a dataset simultaneously. The optimal parameters p , d , and q for ARIMA models were determined based on the best Akaike Information Criterion value. The parameters p , d , and q of the ARIMA is obtained as (2, 1, 3). The KF model (Grewal and Andrews, 1993) is employed for forecasting by extracting the most recent measured traffic flow. The order of AR is regarded as the number of previous traffic flow data that the KF based on, and the order of AR is set as 5 in this paper. Similarly, the LWR (Cleveland et al., 1988) predicts the future traffic flow using the past traffic flow data, and training observations closer to the predicted point generally receive higher weights with Gaussian weighing function. The MLP model (Raza and Zhong, 2017) and the RBFNN model (Kuang and Huang, 2004) both have 15 input neurons which contain the traffic speed of the target lane section and its two adjacent lane sections at the same road section during the past five time intervals. The two model both have one hidden layer containing 15 hidden units and one output layer. The radial basis function of the RBFNN model is chosen to be Gaussian kernel. The CNN model (Ke et al., 2019) divides the road area into a 7×3 grid according to the number of road sections and lanes. The traffic speed of the lane section at one time interval is viewed as one channel of a picture. The previous five time intervals are utilized to form the input picture with 5 channels. The input is a 3D tensor [5, 7, 3] and the filter of the CNN model is 2×2 . The CNN model uses two conventional layers to learn the spatial features and uses two full connection layers with softmax activation function to obtain the predicted result in the next time interval. The LSTM model (Ma et al., 2015) can capture nonlinear traffic dynamic effectively and automatically determine the optimal time lags. The LSTM model composes of one input layer, one LSTM layer with memory blocks, and one output layer. The largest time lag is 10, and the other hyper-parameters are set the same as those of the FDL model.

4.3. Evaluation criteria and experimental platform

The effectiveness of the FDL model and other benchmark methods are evaluated via four criteria, which are mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE) and variance of absolute percentage error (VAPE) (Wang et al., 2014) given by the following equations:

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

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

$$MAPE = \frac{1}{L} \sum_{i=1}^L \left(\frac{|\hat{y}_i - y_i|}{y_i} \right) \quad (22)$$

$$VAPE = \sqrt{\frac{L \sum_{i=1}^n \left(\frac{|\hat{y}_i - y_i|}{y_i} \right)^2 - \left[\sum_{i=1}^L \left(\frac{|\hat{y}_i - y_i|}{y_i} \right) \right]^2}{L(L-1)}} \quad (23)$$

where \hat{y}_i is the predicted value; y_i is the measured value; L is the length of the prediction sequence. Note that the MAE, the MAPE and the RMSE are utilized to evaluate the accuracy of the prediction models and the VAPE is used to evaluate the stability of the prediction models. In addition, our experiment platform is a personal computer with Core(TM) i7-8700 CPU@3.20 GHz. Python 3.6, Tensorflow 1.0.0, and Keras 1.0 are used to realized the models.

5. Results and analysis

5.1. The results of feature extraction

Fig. 6 gives the result of spatial feature extraction. As shown in Fig. 6, the candidate lane sections which are closer to the target lane sections obtain larger GRG values. It means that the candidate lane sections surrounding the target lane section are more likely to be selected as the input of the FDL model. In addition, with the same distance to the target lane sections, the GRGs of the upstream lane sections are larger than those of the downstream lane section, which illustrates that the traffic state of the upstream lane sections has more significant impact on the downstream lane sections. If $\lambda = 5$, the candidate lane sections with these top 5 GRGs are chosen to reconstruct the input matrix of the FDL model based on the result of spatial feature extraction.

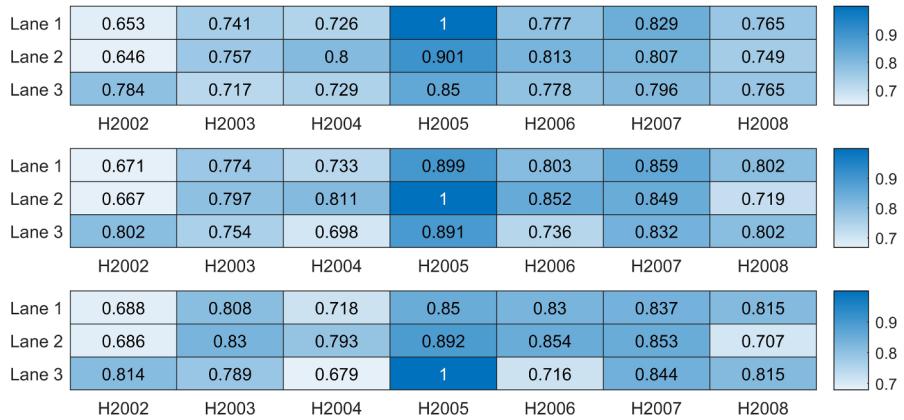


Fig. 6. The GRGs of the target lane sections.

5.2. Prediction results of the FDL model at different lane sections

Fig. 7 shows the prediction results of the FDL on a weekday and on a weekend. In Fig. 7, the FDL model is capable of capturing the tendency and volatility of the traffic speed of the different lane sections. Even during the peak periods on weekdays when vehicle speed fluctuates greatly, the FDL model has a dominant performance in fitting the sudden change of travel speed.

Table 3 compares the prediction performance of the FDL model at different lanes. It demonstrates that the proposed model works best at the middle lane not only on weekdays but also on weekends due to the stable traffic flow. At the inside lane where the overtaking and furious driving often take place, the RMSE and VAPE are lightly higher than those of the other lane sections.

In addition, the performance of the proposed model on weekends is superior to that of the proposed model on weekdays with the improvement of 1.46% and 3.01% on MAPE and VAPE respectively due to the higher average speed and lower traffic volume on weekends.

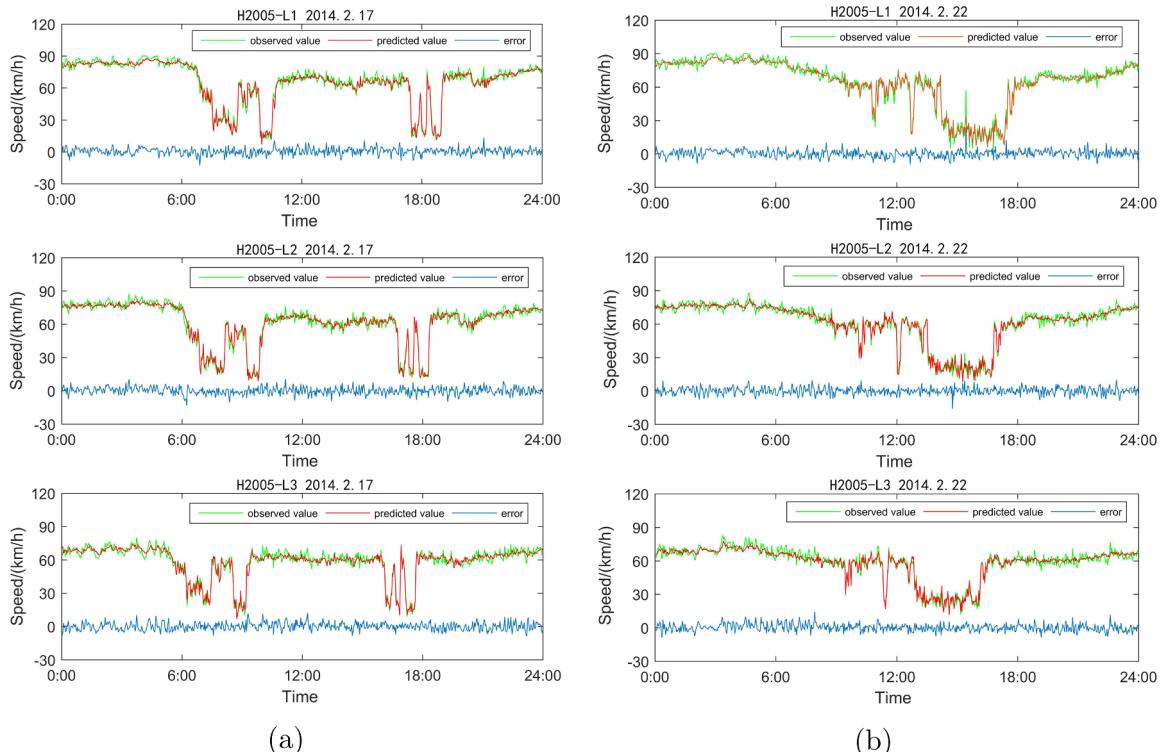


Fig. 7. Prediction results of the FDL model at different lane sections: (a) on a weekday (2014.2.17); (b) on a weekend (2014.2.22).

Table 3

Comparison of the results of the FDL model at different lane sections.

Lane ID	Weekdays				Weekends			
	MAE	MAPE	RMSE	VAPE	MAE	MAPE	RMSE	VAPE
H2005-L1	2.59	6.43%	3.48	10.76%	2.52	4.76%	3.24	8.07%
H2005-L2	2.5	6.16%	3.19	7.75%	2.59	4.95%	3.34	6.13%
H2005-L3	2.54	6.20%	3.24	9.03%	2.6	4.69%	3.36	4.30%
Average	2.54	6.26%	3.30	9.18%	2.57	4.80%	3.31	6.17%

5.3. Performance comparison of the FDL model and benchmark models

Table 4 shows the comparison of the overall results of the FDL model and the benchmarks by considering the 7-day results as a whole. It is found that the FDL model works best among these methods, which outperforms the second best predictor LSTM with the improvement of 2.76% and 8.01% on MAPE and VAPE, and outperforms the best non-parametric method KF with the improvement of 1.33, 5.14%, and 16.47% on MAE, MAPE and VAPE respectively. The experimental results illustrate that the FDL model can efficiently learn the spatio-temporal features of the lane-level traffic flow by using the EGRA and RNN-based structure. Among these different models, the ARIMA, the KF, and the LSTM considered the time-series information of the traffic flow, and the CNN and the RBFNN considered both spatial and temporal features of traffic flow. However, the ARIMA, the KF, and the LSTM work better than the CNN and the RBFNN. It illustrates that the temporal features of the traffic flow are essential information in short-term traffic prediction, and the utilization of more spatial features may not improve the prediction results and even lead to a larger prediction error since the random fluctuations of the traffic flow at other lane sections may cause a disturbance to the target lane sections.

Fig. 8 gives the overall prediction errors produced by these different methods. As shown in **Fig. 8a** and b, the proposed method reveals better performance than other models in terms of the maximum, minimum and the median of errors. In addition, it can be found that the FDL model has a smaller distance between Q1 and Q3 and the error distribution of the FDL model is more concentrated than those of other models. Therefore, the RMSE and the VAPE of the proposed method shown in **Fig. 8c** are lower than those of other methods, indicating that the FDL model is more accurate and stable.

Table 5 compares the prediction results of different methods on weekdays and weekends. It can be found that the MAE and RMSE of each method on weekdays are similar to those of each method on weekends, while the MAPE and VAPE of the weekdays are much larger than those of the weekends. The reason is that the traffic flow on the weekends is much smoother and the average speeds of the weekends is larger than that of the weekdays. However, the FDL model has better performance than other benchmark models in terms of accuracy and robustness.

The purposes of the MAEs, RMSEs, MAPEs and VAPEs are to measure the prediction errors between the ground-truth values and the forecasted values. Meanwhile, the predicted accuracy of the spatio-temporal distribution is vital as well. Hence, we measure the prediction result of the different methods at multiple lane sections and during different time intervals. The average correlation (AC) is introduced to evaluate the spatial and temporal distribution as follows.

$$AC_T = \frac{1}{n_t} \sum_{t=1}^{n_t} \text{Corrcoef}\left(F_t, T_t\right) \quad (24)$$

$$AC_S = \frac{1}{n_s} \sum_{s=1}^{n_s} \text{Corrcoef}\left(F_s, T_s\right) \quad (25)$$

where AC_T represents the similarity between the predicted value and observed value at the same time intervals; AC_S represents the similarity between the predicted value and observed value at the same lane section; n_t and n_s denote the number of predicted vectors on the time dimension and space dimension respectively; F_t and T_t denote the forecast speed at space point s and at time interval t ; and T_s and T_t denote the forecast speed at space point s and at time interval t .

Table 6 demonstrates the comparison of the average correlation of different methods on spatio-temporal dimension. As shown in

Table 4

The overall prediction results of different methods.

Methods	MAE	MAPE	RMSE	VAPE
ARIMA	3.92	10.99%	5.81	22.96%
LWR	4.68	11.99%	5.31	17.24%
MLP	3.9	11.05%	5.65	24.01%
KF	3.88	10.99%	6.05	24.04%
RBFNN	4.45	12.41%	6.44	26.47%
CNN	4.38	12.42%	5.93	22.31%
LSTM	3.37	8.61%	4.69	16.58%
FDL	2.55	5.85%	3.31	8.57%

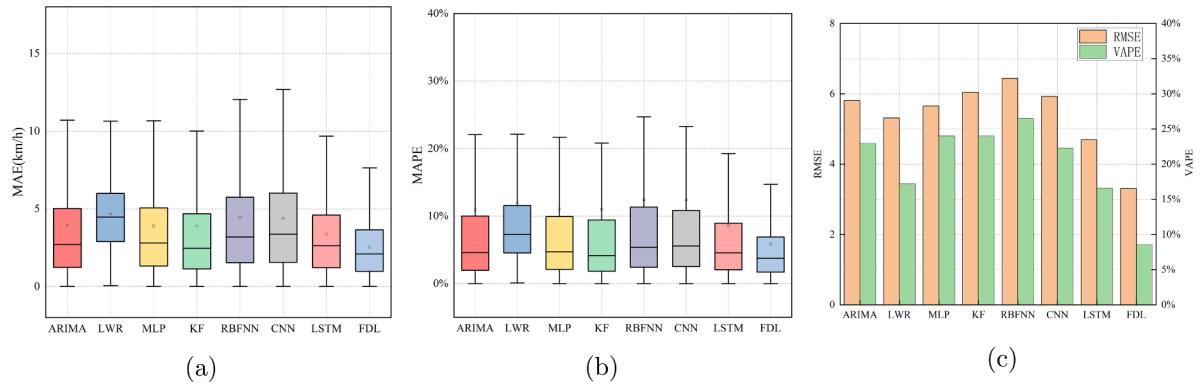


Fig. 8. Comparison of the prediction errors for each method: (a) MAE; (b) MAPE; (c) RMSE & VAPE.

Table 5

Result comparison of different methods on weekdays and on weekends.

Methods	Weekdays				Weekends			
	MAE	RMSE	MAPE	VAPE	MAE	RMSE	MAPE	VAPE
ARIMA	4.02	5.96	12.00%	24.62%	3.68	5.39	8.44%	17.83%
LWR	4.71	5.36	13.01%	18.72%	4.60	5.20	9.42%	12.32%
MLP	4.00	5.79	12.17%	25.94%	3.64	5.28	8.20%	17.87%
KF	4.08	6.35	12.17%	25.83%	3.36	5.20	7.98%	18.35%
RBFNN	4.60	6.66	13.69%	28.68%	4.07	5.83	9.14%	19.39%
CNN	4.59	6.19	14.06%	24.53%	3.84	5.19	8.23%	14.47%
LSTM	3.43	4.77	9.42%	17.72%	3.19	4.50	6.55%	13.02%
FDL	2.54	2.18	6.26%	9.26%	2.57	3.31	4.81%	6.38%

Table 6

Comparison of the average correlation of different methods.xmllabelt0030

Mehtods	AC_T	AC_S
ARIMA	0.806517	0.956489
LWR	0.781103	0.963139
MLP	0.816557	0.958601
KF	0.856253	0.953128
RBFNN	0.804509	0.946415
CNN	0.798009	0.958443
LSTM	0.818575	0.971842
FDL	0.845968	0.985987

Table 6, the proposed method is capable of providing reliable prediction in terms of spatio-temporal distributions. To be specific, the FDL method displays the highest AC in the spatial dimension and exhibits the second highest AC in temporal dimension. It is interesting to find that the KF model has the highest AC_T among these different methods, which shows its superiority in dealing with the discrete-data non-linear problem and in estimating the future state.

In addition, Fig. 9 reveals the performance of the traffic prediction using the neural network (NN) model in terms of MAE, MAPE and RMSE over 24 h. The testing NN-based models include the MLP, the RBFNN, the CNN, the LSTM, and the FDL. The 7-day samples are used to conduct on studying the error distribution of these NN-based methods during the whole day. It can be learned from Fig. 9 that the MAPEs of these five NN-based models during the peak hours are relatively higher than those of off-peak hours, and the MAPEs of these five NN-based models during the morning peak hours are lower than those of the evening peak hours. Besides, due to the decrease in the traffic volume, the MAEs and RMSEs of these models at night and early morning are lower than those during the working time. The proposed model shows its advantage in prediction accuracy and stability for 24 h. The MAEs and the RMSEs of the FDL model fluctuate slightly throughout the day. Meanwhile, the MAPEs only rise to approximately 12% during the peak hour of the entire day.

5.4. Comparison of the FDL models with multiple inputs and structures

For the FDL model, the input number of the spatial features and structure of the RNN-based layers are likely to be associated with the accuracy and stability of the prediction result. In this subsection, we carried out the experiment by testing the FDL models with

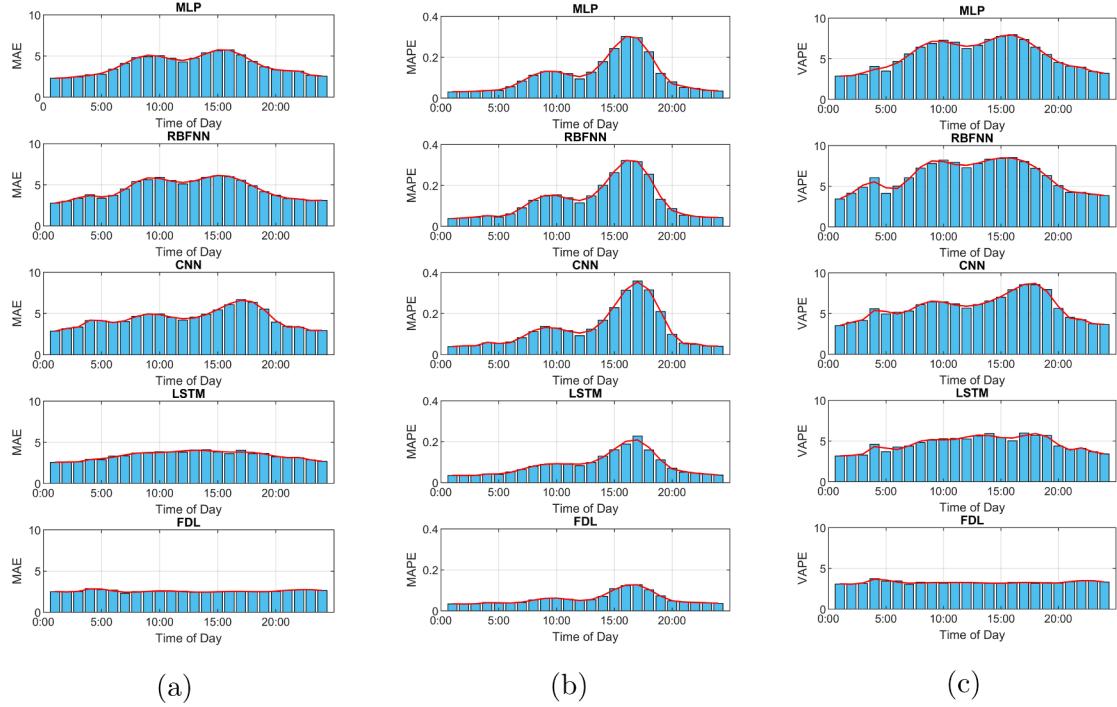


Fig. 9. Prediction error analysis of the NN-based methods in 24 h: (a) MAE; (b) MAPE; (c) RMSE.

different numbers of the input spatial features and different RNN-based structures. Four different structures including one GRU layer, one LSTM layer, LSTM-GRU stacked layers, and the GRU-LSTM stacked layers are tested combined with the GREAs method. The input number of each testing structure increases from 1 to 15. The results in Fig. 10 shows that input number of the spatial features and the structure both have an influence on the accuracy and stability of the FDL model. For a particular structure, with the increase of the input number of the spatial features, the MAE and VAPE firstly drop and then the two indexes stable or rise a little. This means that the appropriate increase of the spatial features contributes to improving the performances of the FDL models with different structures. However, if too many spatial features are utilized as the input, the prediction accuracy and stability especially will be affected by more useless and repetitive information. According to Fig. 10, it is assumed that approximately 5 to 10 spatial features are enough to capture the fluctuations of traffic speeds in the target lane considering the performance and time consumption.

In addition, compared with single-layer structures, the double-layer structures show better performance in extracting spatio-temporal traits of the traffic speed even with less spatial features. Meanwhile, with the same input and similar time consumption, the results of LSTM-GRU structure are slightly superior to those of GRU-LSTM structure in terms of MAE, RMSE, MAPE and VAPE. When the input number of spatial features is set as five, the LSTM-GRU structure outperforms the GRU-LSTM structure with the improvements of 0.4% and 1.03% on MAPE and VAPE respectively. This could be due to that the memory cell of the LSTM contains more parameters than that of the GRU, and it is more reliable to use the LSTM layer to extract the temporal information of the spatial features directly.

To further analyze the FDL models with the two structures, comparison of the two models with different input time lags was implemented in Table 7. Each time lag equals to a 2-min time interval. The number of the both FDL models was set as 5 which was proved to be efficient in Fig. 9. As shown in Table 7, the accuracy and stability of the two FDL models both rise significantly with the time lags increasing from 1 to 15. Then the MAE, MAPE, RMSE, and VAPE fluctuate smoothly with the time lags more than 15. It means that the appropriate increase of the time lags of the input can improve forecast performance of the FDL models. It is also found that the LSTM-GRU structure works better than the GRU-LSTM structure in terms of the accuracy and stability with the same time lags, and the time cost of the LSTM-GRU structure is little more than that of the GRU-LSTM structure. When the time lag is chosen to be 20, the MAE and VAPE of the LSTM-GRU structure are 2.23 and 5.81% respectively, and the LSTM-GRU structure achieves its best performance.

6. Conclusion

In this paper, a deep learning approach, named FDL, is proposed for lane-level speed prediction. The proposed architecture is fused by entropy-based grey relation analysis (EGRA) and a two-layer RNN-based structure. Firstly, the EGRA method is introduced to extract spatial variables and reconstruct the input matrix of the deep learning framework. Secondly, a two-layer deep learning framework is established for capturing temporal properties by combining the LSTM and the GRU. Thirdly, the real-life data collected by RTMS on 2nd Ring Road of Beijing are utilized to examine the new method in terms of accuracy and robustness. Finally, the

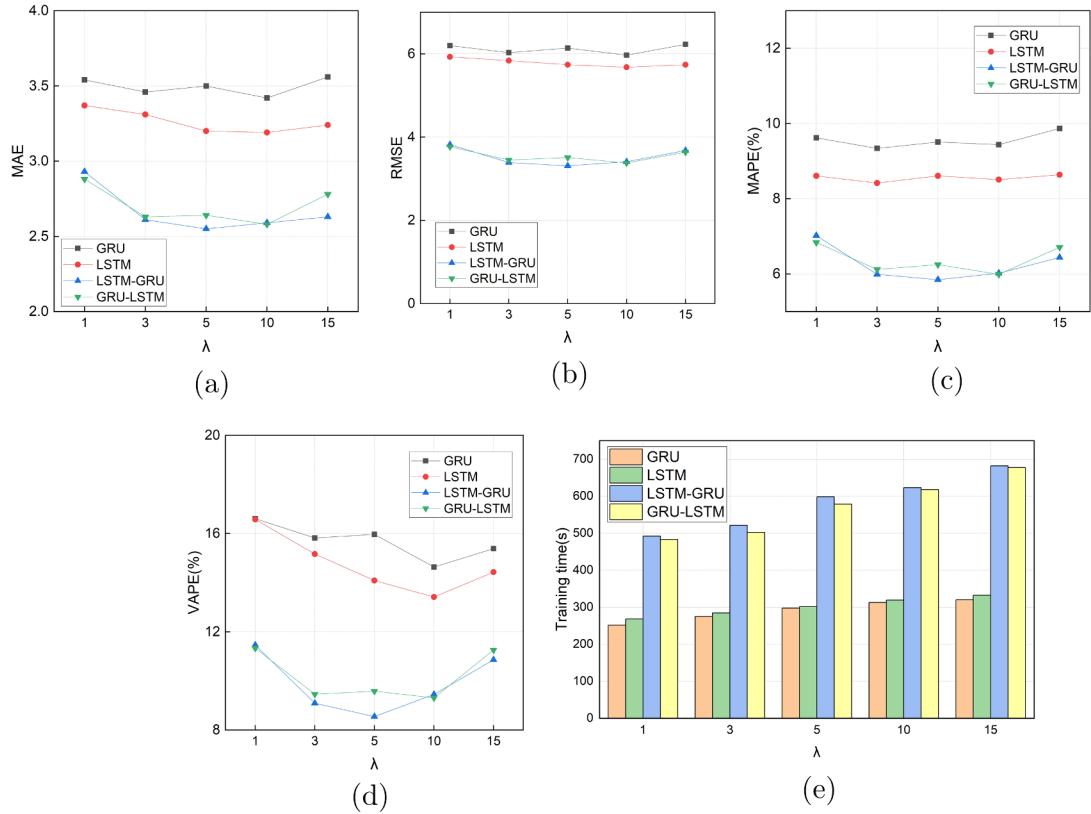


Fig. 10. Comparison of the FDL models with different structures and input features: (a) MAE; (b) RMSE; (c) MAPE; (d) VAPE; (e) Training time.

proposed model is compared with several benchmark algorithms including ARIMA, KF, LWR, MLP, RBFNN, CNN and LSTM, and an analysis of influence factors on the FDL model is carried out.

The experimental result illustrates the future speed of the target lane sections is highly affected by the previous speeds of adjacent lane sections, especially the upstream lane sections. Comparisons with the baseline models show that the FDL model achieves superior prediction accuracy and robustness over all of those benchmark models on both weekdays and weekends. The FDL models error is the lowest and predicted speeds are most robust during the 24 h among NN-based benchmark models. In addition, an appropriate increase of the spatial features can improve the performance of the FDL model, and the LSTM-GRU stacked structure turns out to be more efficient to learn spatio-temporal features from the dataset.

Future work will concentrate on exploring more deep learning structures based on the fusion theory and introducing multi-source input variables including volume and occupancy to improve the accuracy of the model. Dataset with longer temporal dimension and broader spatial dimension could be employed to train and test the proposed model. Furthermore, with the help of the lane-level traffic state forecasting techniques, the lane-level route guidance and lane-changing behavior of CVs based on the prediction result of lane-level travel speed can be studied. In addition, the FDL model can be applied to other fields such as missing traffic data imputation, traffic congestion forecasting and traffic accident detection.

Table 7
Comparison of the FDL models with different time lags.xmllabel0035

Methods	Time lag	1	5	10	15	20	25	30
LSTM-GRU	MAE	3.78	2.95	2.55	2.35	2.23	2.32	2.25
	RMSE	5.49	3.98	3.31	3.02	2.90	2.98	2.94
	MAPE	10.91%	7.63%	5.85%	5.14%	4.78%	5.31%	4.80%
	VAPE	23.93%	14.53%	8.55%	6.80%	5.81%	6.39%	5.98%
GRU-LSTM	Traning time (s)	140.01	299.65	598.29	775.26	1045.94	1188.30	1461.24
	MAE	3.81	3.01	2.64	2.39	2.32	2.38	2.29
	RMSE	5.52	4.00	3.51	3.13	3.06	3.09	2.94
	MAPE	10.91%	7.87%	6.25%	5.15%	5.06%	5.12%	4.89%
	VAPE	23.92%	15.21%	9.58%	6.51%	6.23%	6.34%	6.12%
	Traning time (s)	116.65	260.72	578.64	626.76	823.32	1062.13	1341.97

Acknowledgement

This research is supported by the surface project of the National Natural Science Foundation of China (No. 71273024), the National Key Basic Research Program (2012CB725403).

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at <https://doi.org/10.1016/j.trc.2019.07.003>.

References

- Box, G.E., Jenkins, G.M., Reinsel, G.C., 1976. Time series analysis, forecasting and control. *J. Time* 31 (4), 238–242.
- Castro-Neto, M., Jeong, Y.S., Jeong, M.K., Han, L.D., 2009. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert Syst. Appl. Int. J.* 36 (3), 6164–6173.
- Chan, K.Y., Dillon, T.S., Singh, J., Chang, E., 2012. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and Levenberg-Marquardt algorithm. *IEEE Trans. Intell. Transp. Syst.* 13 (2), 644–654.
- Chen, C., Hu, J., Meng, Q., Zhang, Y., 2011. Short-time traffic flow prediction with ARIMA-GARCH model. In: Proceedings of the IEEE Intelligent Vehicles Symposium, pp. 607–612.
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv: 1406.1078*.
- Cleveland, W.S., Devlin, S.J., Grosse, E., 1988. Regression by local fitting. *J. Econom.* 37 (1), 87–114.
- Cui, Z., Ke, R., Wang, Y., 2018. Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction. In: International Workshop on Urban Computing, pp. 22–25.
- Ding, Q.Y., Wang, X.F., Zhang, X.Y., Sun, Z.Q., 2010. Forecasting traffic volume with space-time arima model. *Adv. Mater. Res.* 156–157, 979–983.
- Dougherty, M.S., Cobbett, M.R., 1997. Short-term inter-urban traffic forecasts using neural networks. *Int. J. Forecast.* 13 (1), 21–31.
- Ghosh, B., Basu, B., O'Mahony, M., 2009. Multivariate short-term traffic flow forecasting using time-series analysis. *IEEE Trans. Intell. Transp. Syst.* 10 (2), 246–254.
- Grewal, M.S., Andrews, A.P., 1993. Kalman Filtering: Theory and Practice Using MATLAB. Wiley-IEEE Press.
- Guo, F., Polak, J.W., Krishnan, R., 2018. Predictor fusion for short-term traffic forecasting. *Transp. Res. Part C: Emerg. Technol.* 92, 90–100.
- Guo, J., Huang, W., Williams, B.M., 2014. Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. *Transp. Res. Part C: Emerg. Technol.* 43, 50–64.
- Hamed, M.M., Al-Masaied, H.R., Said, Z.M.B., 1995. Short-term prediction of traffic volume in urban arterials. *J. Transp. Eng.* 121 (3), 249–254.
- Huang, W., Song, G., Hong, H., Xie, K., 2014. Deep architecture for traffic flow prediction: deep belief networks with multitask learning. *IEEE Trans. Intell. Transp. Syst.* 15 (5), 2191–2201.
- Irie, K., Tüske, Z., Alkhoul, T., Schlüter, R., Ney, H., 2016. LSTM, GRU, highway and a bit of attention: an empirical overview for language modeling in speech recognition. In: Proceedings of the Annual Conference of the International Speech Communication Association, pp. 3519–3523.
- Jia, Y., Wu, J., Xu, M., 2017. Traffic flow prediction with rainfall impact using a deep learning method. *J. Adv. Transp.* 2017 (722), 1–10.
- Kai, N., Schreckenberg, M., 1992. A cellular automaton model for freeway traffic. *J. Phys. I* 2 (12), 2221–2229.
- Karlaftis, M.G., Vlahogianni, E.I., 2011. Statistical methods versus neural networks in transportation research: differences, similarities and some insights. *Transp. Res. Part C: Emerg. Technol.* 19 (3), 387–399.
- Karpathy, A., Johnson, J., Fei-Fei, L., 2015. Visualizing and understanding recurrent networks. In: International Conference on Learning Representations (ICRL), pp. 1–12.
- Ke, R., Li, W., Cui, Z., Wang, Y., 2019. Two-stream multi-channel convolutional neural network (TM-CNN) for multi-Lane traffic speed prediction considering traffic volume impact. *arXiv preprint arXiv: 1903.01678*.
- Khotanzad, A., Sadek, N., 2003. Multi-scale high-speed network traffic prediction using combination of neural networks. In: Proceedings of the International Joint Conference on Neural Networks, vol. 2. pp. 1071–1075.
- Kuang, A., Huang, Z., 2004. Short-Term traffic flow prediction based on RBF neural network. *Syst. Eng.* 22 (2), 63–65.
- Kwon, J., Varaiya, P., Skabardonis, A., 2003. Estimation of truck traffic volume from single loop detectors with lane-to-lane speed correlation. *Transp. Res. Rec.* 1856 (1), 106–117.
- Laña, I., Lobo, J.L., Capecci, E., Ser, J.D., Kasabov, N., 2019. Adaptive long-term traffic state estimation with evolving spiking neural networks. *Transp. Res. Part C: Emerg. Technol.* 101, 126–144.
- Levin, M., Tsao, Y.-D., 1980. On forecasting freeway occupancies and volumes. *Transp. Res.* 773, 47–49.
- Li, L., Qin, L., Qu, X., Zhang, J., Wang, Y., Ran, B., 2019. Day-ahead traffic flow forecasting based on a deep belief network optimized by the multi-objective particle swarm algorithm. *Knowl.-Based Syst.* 172, 1–14.
- Lippi, M., Bertini, M., Frasconi, P., 2013. Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning. *IEEE Trans. Intell. Transp. Syst.* 14 (2), 871–882.
- Liu, Q., Wang, B., Zhu, Y., 2018. Short-term traffic speed forecasting based on attention convolutional neural network for arterials. *Comput.-Aided Civ. Infrastruct. Eng.* 33 (6), 1–18.
- Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.Y., 2015. Traffic flow prediction with big data: a deep learning approach. *IEEE Trans. Intell. Transp. Syst.* 16 (2), 865–873.
- Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., Wang, Y., 2017. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors* 17 (4), 818.
- Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y., 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part C: Emerg. Technol.* 54, 187–197.
- Min, W., Wynter, L., 2011. Real-time road traffic prediction with spatio-temporal correlations. *Transp. Res. Part C: Emerg. Technol.* 19 (4), 606–616.
- Nicholson, H., Swann, C., 1974. The prediction of traffic flow volumes based on spectral analysis. *Transp. Res.* 8 (6), 533–538.
- Polson, N.G., Sokolov, V.O., 2017. Deep learning for short-term traffic flow prediction. *Transp. Res. Part C: Emerg. Technol.* 79, 1–17.
- Raza, A., Zhong, M., 2017. Hybrid lane-based short-term urban traffic speed forecasting: a genetic approach. International Conference on Transportation Information and Safety 271–279.
- Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M., 2016. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans. Med. Imag.* 35 (5), 1285–1298.
- Takenouchi, A., Kawai, K., Kuwahara, M., 2019. Traffic state estimation and its sensitivity utilizing measurements from the opposite lane. *Transp. Res. Part C: Emerg. Technol.* 104, 95–109.
- Van Lint, J., Hoogendoorn, S., van Zuylen, H., 2002. Freeway travel time prediction with state-space neural networks: modeling state-space dynamics with recurrent neural networks. *Transp. Res. Rec. J. Transp. Res. Board* 1811, 30–39.
- Vanajakshi, L., Rilett, L.R., 2007. Support vector machine technique for the short term prediction of travel time. In: IEEE Intelligent Vehicles Symposium, Proceedings, pp. 600–605.
- Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2014. Short-term traffic forecasting: where we are and where were going. *Transp. Res. Part C: Emerg. Technol.* 43, 3–19.

- Wang, J., Deng, W., Guo, Y., 2014. New bayesian combination method for short-term traffic flow forecasting. *Transp. Res. Part C: Emerg. Technol.* 43, 79–94.
- Wang, J., Chen, R., He, Z., 2019. Traffic speed prediction for urban transportation network: a path based deep learning approach. *Transp. Res. Part C: Emerg. Technol.* 100, 372–385.
- Wang, J., Shi, Q., 2013. Short-term traffic speed forecasting hybrid model based on chaos–wavelet analysis–support vector machine theory. *Transp. Res. Part C: Emerg. Technol.* 27, 219–232.
- Wu, S., Zhong, S., Liu, Y., 2017. Deep residual learning for image steganalysis. *Multimedia Tools Appl.* 77 (9), 10437–10453.
- Wu, Y., Tan, H., Qin, L., Ran, B., Jiang, Z., 2018. A hybrid deep learning based traffic flow prediction method and its understanding. *Transp. Res. Part C: Emerg. Technol.* 90, 166–180.
- Yang, H.F., Dillon, T.S., Chen, Y.P., 2016. Optimized structure of the traffic flow forecasting model with a deep learning approach. *IEEE Trans. Neural Netw. Learn. Syst.* 28 (10), 2371–2381.
- Yu, X., Prevedouros, P.D., 2013. Performance and challenges in utilizing non-intrusive sensors for traffic data collection. *Adv. Remote Sens.* 02 (2), 45–50.
- Zhan, F., Wan, X., Cheng, Y., Ran, B., 2018. Methods for multi-type sensor allocations along a freeway corridor. *IEEE Intell. Transp. Syst. Mag.* 10 (2), 134–149.
- Zhang, J., Zheng, Y., Qi, D., Li, R., Yi, X., Li, T., 2018. Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artif. Intell.* 255, 147–166.
- Zhang, Q., Deng, J., Shao, Y., 1995. A grey correlational analysis by the method of degree of balance and approach. *J. Huazhong Univ. Defence Technol.* 23 (11), 94–98.
- Zhang, Y., Zhang, Y., Haghani, A., 2014. A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model. *Transp. Res. Part C: Emerg. Technol.* 43 (1), 65–78.
- Zhao, Z., Chen, W., Wu, X., Chen, P.C.Y., Liu, J., 2017. Lstm network: a deep learning approach for short-term traffic forecast. *IET Intell. Transp. Syst.* 11 (2), 68–75.