

Research Article

An Improved Deep Spatial-Temporal Hybrid Model for Bus Speed Prediction

Huawei Zhai ^{1,2}, Licheng Cui ³, Weishi Zhang ¹, Xiaowei Xu ² and Ruijie Tian¹

¹Information Science and Technology School, Dalian Maritime University, Dalian 116026, China

²University of Arkansas at Little Rock, Little Rock, AR, USA

³Public Security Information Department, Liaoning Police College, Dalian 116036, China

Correspondence should be addressed to Licheng Cui; clcicheng@163.com, Weishi Zhang; teesiv@dmlu.edu.cn, and Xiaowei Xu; xwxu@ualr.edu

Received 11 May 2020; Revised 26 July 2020; Accepted 18 August 2020; Published 27 September 2020

Academic Editor: Mahmoud Mesbah

Copyright © 2020 Huawei Zhai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

For resolving or alleviating the transportation problems, it is necessary to efficiently manage the public transportation and provide public transport services with high quality and advocate green travel, which rely on accurate traffic data. In order to obtain more accurate bus speed in the future, this paper proposed a novel dynamic hierarchical spatial-temporal network model based on Grey Relation Analysis (EGRA), the convolutional neural network (CNN), and the gated recurrent unit (GRU). The proposed model is named the DHSTN; it exploited EGRA to analyze and choose the suitable candidate line sections with high impacts on the target section and, then, construct a multilayer structure based on the CNN, GRU, and attention mechanism to analyze and capture the spatial and temporal dependency, and finally, the extreme learning machine (ELM) is exploited for the fusion of the long-term and short-term dependency to predict the bus speed variation in the next time interval. Comparative experiments indicate that the DHSTN has better performances, the mean absolute error is around 2.6, and it meets the real requirements.

1. Introduction

With the urbanization, urban population grows rapidly, the increase in the aggregate amount of urban automobiles is still high on the way, and urban public resources are limited or insufficient; these lead to the urban transportation problems which are more and more serious such as traffic congestion, traffic overload, and traffic accidents which are the major problems facing widespread life. Thus, green travels advocated by related organizations and urban public transport system are the main forces of urban transportation, which play an important role. Urban buses are the basic of urban public transport system, and it is crucial to analyze the urban buses and their relative information, such as bus speed and bus travel time.

Bus speed is the vital information, and it is essential for improving the performances of the urban public transport system [1]. For the public transportation agencies, they can manage the whole public transportation, schedule the buses

and other resources to meet the travel requirements efficiently, and hence, mitigate crowdedness at bus stations and in buses and increase ridership [2]. For travelers, they can plan their trips based on these information reasonably. Besides, accurate and timely information about the bus speed and time is crucial; it could attract more travelers, reduce the waiting time, and improve their satisfaction [3, 4]. Therefore, for providing public transportation agencies and travelers with more timely and accurate information, efficient models need to be developed to predict the bus speed more accurately.

As is known, the bus speed prediction has gradually attracted the attention of researchers and public transportation agencies [5], it is significantly different from traditional traffic speed prediction, and its impacts are more, such as traffic congestions and bus operation schedules. But, nowadays, lots of cities set the bus lane in traffic peak hours, and to some extent, this alleviates the problems within prediction. Thus, the bus speed prediction is more similar to

the traffic speed prediction in some time without considering scheduling and other relative factors, and it can refer to more traffic speed prediction models, but it remains facing many challenges.

Similar to traffic speed prediction, it is necessary to consider sufficient factors to construct the prediction model for achieving accurate results. Thus, this paper proposes an efficient prediction model for bus speed, called the Dynamic Hierarchical Spatial-Temporal Network, (DHSTN). The contributions of the DHSTN mainly involve four aspects: (1) The dynamic weights method is proposed to calculate the bus travel average speed of the bus line section; it makes full use of the each bus speed passing the bus line section and improves the accuracy of average speed. (2) The Entropy-based Grey Relation Analysis (EGRA) is introduced to analyze the spatial correlations of nearby line sections and the target section, in order to extract the spatial line sections from candidates as the analyzed objects together with target section. (3) For making full use of the long-term temporal features, the attention mechanism is introduced to analyze the long-term temporal feature, so as to improve the prediction accuracy. (4) A multilayer structure is proposed; it combines the CNN, GRU, and attention mechanism, in order to capture the spatial and long-term/short-term temporal features within the travel speed.

2. Related Works

In the recent years, the bus speed prediction has gradually drawn researchers' attention, and various prediction models have been proposed. Different from traditional traffic speed prediction, these models rarely used linear models (such as ARIMA [6, 7]) to predict the bus speed because the bus speed is not a standard problem, and the prediction results are not better. For analyzing the nonlinear features within bus speed data, many nonlinear models are exploited by researchers, such as the Bayesian network [8], Support Vector Machine (SVM) [9], radial basis function neural networks (RBFNN) [10], Kalman filter [11], back propagation neural network (BPNN) [12], and extreme learning machine (ELM). The core idea of nonlinear models is to capture the nonlinear relationship within speed data and mine more potential information without prior knowledge [13]. Both the linear and nonlinear prediction models are based on the time-dependent features of bus speed data and only analyze the short-term temporal dependency.

Each prediction model mentioned above has its own limitations and advantages; to some extent, it captures one or more features of bus speed, and it is hard to cover all or most of features and provide better prediction performances by a single model [14]. Thus, for making full use of each prediction model, researchers proposed the hybrid models, which combine two or more prediction models, and they make up for the limitations of the single model and show better prediction performances. Yang et al. [15] proposed the GA-SVM model based on SVM and genetic algorithm (GA) in analyzing the features of bus travel speed data, it still considered the time period, road length, weather, and other factors, and testing results showed that GA-SVM is superior

to the single SVM model and Artificial Neural Network (ANN). Hashi et al. [16] proposed a novel hybrid model based on SVM and the Kalman filter to predict bus arrival time, and test results showed that it has better performances than GA-SVM.

The linear prediction models, nonlinear prediction models, and the hybrid prediction models mentioned above are exploited to capture the temporal dependency within the data and still have good performances. As is known, the bus speed has obvious temporal features, which are made full use of by many prediction models, but it is noted that the speed has still spatial dependency. Thus, in analyzing the features within the data, the prediction models should capture the spatial and temporal dependency. Liu et al. [17] proposed a comprehensive prediction model based on the long short-term memory (LSTM) and ANN, and it utilized the spatial-temporal features vector to predict the bus arrival time. Cui et al. [18] proposed a novel multivariate conditional autoregressive (MCAR) model, and it is exploited to analyze the spatial and temporal correlations of the bus speeds. Sun et al. [2] proposed a hybrid prediction model based on the LSTM and attention mechanism, it can capture the spatial and temporal features, but it is weak in capturing the long-term temporal dependency. Hu et al. [8] proposed a novel Bayesian model to characterize space-time interaction patterns and to construct a bus speed prediction model further. Gu et al. [19] proposed a fusion model based on LSTM, EGRA, and the gated recurrent unit (GRU), and a double-layer structure is constructed based on the LSTM and GRU in order to resolve the problem that the single-layer LSTM or GRU cannot capture the short-term temporal dependency. Lu et al. [20] proposed a novel hybrid model for lane-level traffic prediction; it is based on the complete ensemble empirical mode (CEEMDAN) and extreme gradient boosting (XGBoost), CEEMDAN is exploited to decompose the raw data into two components, and the XGBoost is used to analyze each component and predict its changes, respectively, and finally, the prediction results are integrated to obtain the final prediction result; experimental results confirm that the hybrid model outperforms the state-of-the-art models in the term of prediction accuracy and stability. Guo et al. [21] proposed a novel prediction fusion framework; it can combine multiple individual predictors and use different fusion strategies to improve the model performances. Gu et al. [22] proposed an improved Bayesian combination model with deep learning (IBCM-DL); it analyzes the relevance between the historical traffic data and current data, and three subpredictors are incorporated into the IBCM framework to take advantages of each model. Bachu et al. [23] developed a new prediction method which considers both temporal and spatial variation in travel time; first, it rewrites the conservation of vehicles equation in terms of flow and density instead of speed, then discretizes the novel equation, and finally, it is used in the prediction scheme which is based on the Kalman filter, and experimental results confirm that the proposed method is able to perform better. Ma et al. [24] also proposed a novel segment-based approach to predict bus travel time using a combination of real-time taxi and bus datasets, and the

experimental results confirm that the novel approach improves the accuracy of bus travel time prediction, especially under abnormal traffic conditions.

3. Preliminaries

In the section, after analyzing the urban bus net and bus trajectory data, we give a novel method for getting the average speed of each bus line section.

Different from the urban road net, the urban bus line is irregular, and the urban bus net is complex, as shown in Figure 1, in which the black points are the stations, green means that the traffic in the line sections is smooth, yellow means it is slow traffic, and red means it is congested. Before analyzing the bus running trajectory data, the bus net needs to be gridded based on the stations. But, it is generally known that several bus lines may overlap in some area; this leads to the stations to get too close, and the bus net being gridded is too complex. So, the stations in the overlapping area of bus lines are merged, in order to simplify the bus net after being gridded, and the result is shown in Figure 2.

Based on the gridding bus net, the bus running data can be matched to the corresponding line section, by the following equations:

$$\min(\text{lon}_A, \text{lon}_B) < \text{lon}_x < \max(\text{lon}_A, \text{lon}_B), \quad (1)$$

$$\min(\text{lat}_A, \text{lat}_B) < \text{lat}_x < \max(\text{lat}_A, \text{lat}_B), \quad (2)$$

where $(\text{lon}_A, \text{lat}_A)$ and $(\text{lon}_B, \text{lat}_B)$ are the position of any two adjacent stations and $(\text{lon}_x, \text{lat}_x)$ is the trajectory point position of any running bus.

When the bus runs from station i to station $i + 1$, its running speed is not constant, and it is sometimes in the process of acceleration or deceleration, as shown in Figure 3.

The state of acceleration and deceleration is not conducive to get the average speed from station i to station $i + 1$ (or line section). Thus, the trajectory point data in the speed buffer area is not considered.

In general, the average speed $V_{i,j}$ of bus j in line section i is obtained by the following equation:

$$V_{i,j} = \frac{L(i)}{T(i,j)} = \frac{L(i)}{T_{N_e} - T_{N_s}}, \quad (3)$$

where $L(i)$ is the length of line section i , T_{N_s} and T_{N_e} are the projection time of bus j passing endpoints $N_s(i)$ and $N_e(i)$ of line section i , and $T(i,j) = T_{N_e} - T_{N_s}$. But, due to various reasons, in real bus trajectory data, it is hard to get time T_{N_s} and T_{N_e} . Based on these bus trajectory points in the line section, a novel method (Dynamic Weight Computing Method, DWCM) for computing the average speed is given, and it performs in two sequential steps.

The first step is to compute the average speed of each bus passing the line section. Based on the running trajectory of the bus, the first trajectory point and the last trajectory point data in the line section are selected to compute its average speed as follows:

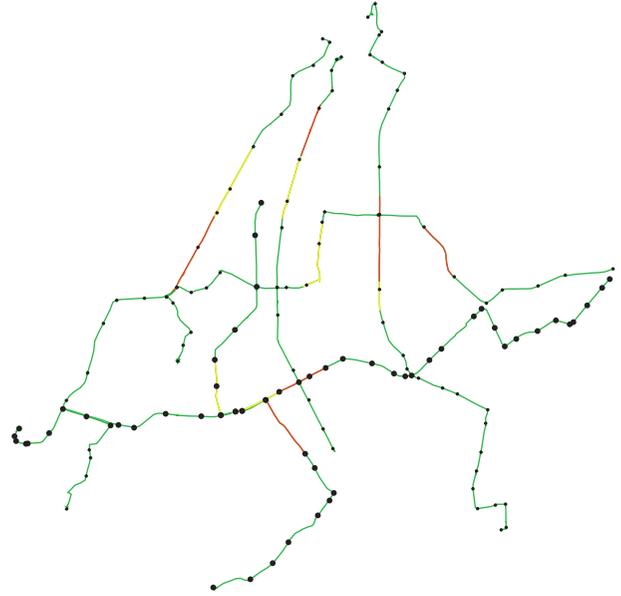


FIGURE 1: Part of the urban bus net marking stations.

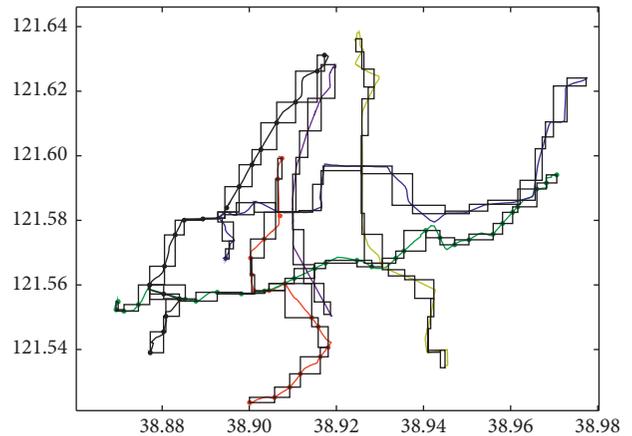


FIGURE 2: Part of the urban bus net after being gridded.

$$V_{i,j} = \frac{L_j(i)}{T^t(i,j)} = \frac{L_j(i)}{T_n - T_1}, \quad (4)$$

where T_1 and T_n are the sampling time of the first trajectory point and the last trajectory point in the line section, respectively, and $L_j(i)$ is the length of the line between the first trajectory point and the last trajectory point.

The second step is to compute the average speed of buses in the line section. In general, the average of one or several buses cannot fully reflect the real average speed of the line section, and it is partial. Thus, the impact factor λ , a novel conception is proposed, and it can dynamically show the influences of each bus average speed on the real average speed of the line section. It is obtained by the following equations:

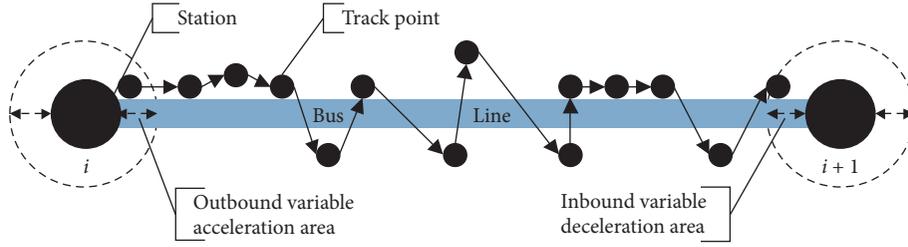


FIGURE 3: The changes of the bus speed from station i to station $i + 1$.

$$\lambda(i, j) = \frac{L_j(i)}{L(i)}, \quad (5)$$

$$w_j = \frac{\lambda(i, j)}{\lambda(i, 1) + \lambda(i, 2) + \dots + \lambda(i, N_i)}, \quad (6)$$

where N_i is the number of buses passing line section i . Finally, based on $V_{i,j}$ and $\lambda(i, j)$, the average speed of the line section can be obtained by the following equation:

$$V_i = \sum_{j=1}^{N_i} w_j V_{i,j}. \quad (7)$$

4. Methodology

4.1. Temporal Dependency Mining. As a basic part of the urban intelligent transportation system, urban bus speed exhibits higher correlation in a short time period, due to the nature of the transportation system [25], and the short-term temporal dependency is the key factor for prediction. Moreover, similar to traffic flow, it also exists long-term temporal dependency within bus speed data. The special example is that because of extreme congestion or extreme weather, the buses will take a long time to travel from the last line section to current line section. Hence, during temporal dependency mining, both the short-term and long-term temporal dependency must be taken into account. Traditional methods only consider the short-term temporal dependency and ignore coexistence of short-term temporal dependency and long-term temporal dependency. In the recent years, many research findings have explored the fusion methods to mine the temporal dependency within the data [25–29]. The long short-term memory network (LSTM) is an important RNN and introduced to analyze temporal features of traffic data [27, 30], and it can fully capture both short-term and long-term memories within traffic data and avoid gradient vanishing or exploding problems.

As is known, the LSTM is an ideal structure [19], has good performances, and avoids some unnecessary problems, but its learning parameters are a little more, to some extent, which affects the time performance. The Gated Recurrent Unit (GRU) is a variant of the LSTM; it is essentially an LSTM without an output gate, and its structure is simpler and learning parameters are less, but the GRU can achieve equal or even better performances [31]. The GRU only has two gates, update gate z_t and reset gate r_t , z_t is to control the degree which the previous hidden state h_{t-1} inputs current

new hidden state h_t , and r_t is to control the degree of ignoring the previous hidden state. Thus, for current input x_t , z_t , r_t , h_t , and the final output y_t are calculated by the following equation [32]:

$$\begin{aligned} z_t &= \sigma(w_z x_t + w_{hz} h_{t-1} + b_z), \\ r_t &= \sigma(w_r x_t + w_{hr} h_{t-1} + b_r), \\ \hat{h}_t &= \tan h(w_h x_t + U_h(r_t \odot h_{t-1} + b_h)), \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t, \\ y_t &= \sigma(w_y h_t), \end{aligned} \quad (8)$$

where w_z , w_r , w_h , w_y , w_{hr} , w_{hz} , and U_h are the weight matrix. $\sigma(\cdot)$ and $\tan h(\cdot)$ are the activation functions, $\sigma(x) = 1/(1 + e^{-x})$, $\sigma(x) \in [0, 1]$, and $\tan h(x) = (e^x - e^{-x})/(e^x + e^{-x})$. \odot represents the Hadamard product of two vectors.

4.2. Spatial Dependency Mining. Only analyzing the temporal features of bus speed will weaken the prediction performances. It is well known that traffic data shows the spatial features obviously; for example, the traffic state of the current line section may be affected by its upstream or downstream line sections. In what follows, it is to focus on exploring the spatial features within bus speed, for which the convolutional neural network (CNN) is selected. The CNN originates from tackling the image processing problems, and it is a highly efficient tool and, naturally, has been utilized in many works [26, 33–36]. Moreover, analyzing the whole urban bus net is a very complex task; it needs lots of data, more resources, and much more time. In real application, it may find out various problems, e.g., only analyzing bus arrival time. These need to transform the whole net to many small segments, which is analyzed easily. In the study, based on the characteristics of current bus lines and data, the 1D-CNN is exploited to mine the spatial features within the data, and thus, it performs the 1D convolution on the vectors

$$x_k = [v_1(t - \tau + k), v_2(t - \tau + k), \dots, v_p(t - \tau + k)]^T \quad (1 \leq k \leq \tau - 1)$$

of matrix V from upstream to downstream by the equation (9) (V is the speed matrix, $V = [v_1, v_2, \dots, v_p]^T$, and p is the line section id, where $v_i = [v_i(t - \tau), v_i(t - \tau + 1), \dots, v_i(t - 1)]$).

$$h_m(x_k) = g(w_m \otimes h_{m-1}(x_k) + b_m), \quad (9)$$

where $h_m(x_k)$ and $h_{m-1}(x_k)$ are the output and input of the m -th convolution layer, respectively, b_m represents the bias,

w_m represents the convolutional weights, $g(\cdot)$ represents the activation function, and \otimes is the convolution.

One of advantages of the deep neural network is that it could improve its performances by increasing hidden layers; the more the hidden layer is, the better its representation capability is, but it is weak in time performances and data demands and also leads to be over fitting easily, and the appropriate hidden layers can balance the performances and make the better results. Thus, based on multispect factors, it sets the hidden layers m as 3, $m = 3$. As mentioned above, the CNN is weak in the time performances, for converging fast, the Rectified Linear Units (ReLUs) are exploited as the nonlinear activation function, and it can also tackle the gradient vanishing or exploding problems [37].

It is well known that, for the CNN, the convolutional layer is the key component, in which, the filter is applied to capture the features from input data. Moreover, Springenberg et al. [38] proposed a novel CNN, whose pooling layer is replaced by a convolutional layer; it has state-of-the-art or competitive performances on the object recognition with small images. Thus, after fully analyzing existing data and its limited space dimension, the special 1D-CNN without the pooling layer is exploited to capture the spatial features.

4.3. Attention. The GRU described above only uses previous data, which are usually a few hours' data, and it does not consider the long-term dependency within the data; long-term dependency has great influences on the final prediction and it is still. As is known, urban bus speed data show obvious periodicity, the same as other traffic data. Thus, it needs to capture both long-term and short-term dependency during prediction and consider the previous short-term data and long-term historical data. Since the shallow single-layer GRU does not capture the long-term memory [39], it needs to be extended further.

In this paper, the data of previous d days are used to capture the long-term information, and for each day, only limited data near predicted time t are chosen, which can reduce influences of redundant data on the final performances. In analyzing the periodicity of bus speed, it finds out the same problem as traffic flow, which is the temporal shifting of periodicity [40]. Thus, it needs to analyze the importance of each input data on the final prediction result, $\text{imp}(x_k, y_t)$, for this the attention mechanism is exploited to analyze each input data. Some researches [40–42] show $\text{imp}(x_k, y_t)$ could be analyzed by linear or nonlinear methods.

The attention mechanism is that, before inputting data into the GRU, it needs to analyze its importance, as shown in Figure 4. We set $x_k = (x_{k,1}, x_{k,2}, \dots, x_{k,n})$, where it is assumed that the n values near the time t have impacts on the prediction, $w_{k,j}$ ($1 \leq j \leq n$) is the weight of each input, and the relation between y_t and x_k is

$$y_t = w_k x_k + b_k, \quad (10)$$

where b_k is the bias. Equation (10) is a multiple regression function; it needs to get optimal \hat{w}_k by minimizing the error

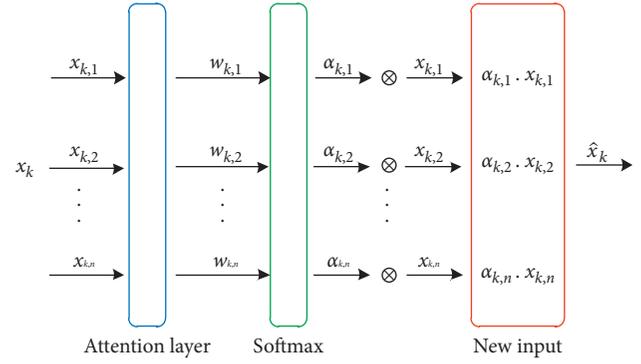


FIGURE 4: The attention mechanism before GRU.

e between y_t and the predicted value y_t^0 , $e = f(y_t, y_t^0)$, and f is the function calculating the error. For the time performance, the idea of Extreme Learning Machine (ELM) is exploited to solve the problem, and w_k and b_k are learning parameters. Finally, the importance of $\alpha_{k,j}$ is obtained by the following equation:

$$\alpha_{k,j} = \frac{\exp(\hat{w}_{k,j})}{\sum_{j=1}^n \exp(\hat{w}_{k,j})}, \quad (11)$$

and then, the new input of the GRU is $\alpha_k \cdot x_k$, where $\hat{w}_{k,j}$ is the analyzing results by minimizing the error e .

4.4. Fusion Method. Based on the analysis mentioned above, a novel fusion is proposed, called the Dynamic Hierarchical Spatial-Temporal Network (DHSTN). Figure 5 is the model architecture. Considering the periodicity and continuity of the urban bus speed, continuous time series data of previous days and the same day last three week are selected as the inputs of the DHSTN with the near-term bus speed data, which are highly correlated to the future bus speed.

The DHSTN is includes six steps. (a) The first step is the feature extracting, showing the “a” in Figure 5. As is known, bus speed data have obvious spatial feature, and it needs to choose the nearby line sections from candidates. For more correlated spatial variables, the correlation analysis is exploited to recognize the line sections with high impacts on the target section. After analyzing lots of related methods, the Entropy-based Grey Relation Analysis (EGRA) is chosen to analyze the candidate line sections, and detail description refers to the literature [43, 44]. (b) The second step is the spatial dependency mining and, in fact, spatial feature analysis has already started in the first step, and in spatial mining, the CNN is chosen, detailed in the section “Spatial Dependency Mining.” (c) The third step is the attention. In the paper, the previous d days data x_k^s , $1 \leq s \leq d$, are chosen to analyze the long-term dependency, for each previous day, and it needs to know the importance of near-term time data $x_{k,i}^s$ on the final predicting time y_t^0 , in order to generate more reasonable input data \hat{x}_k^s . (d) The fourth step is the temporal dependency mining, including the short-term and long-term dependency. The short-term dependency analysis is in

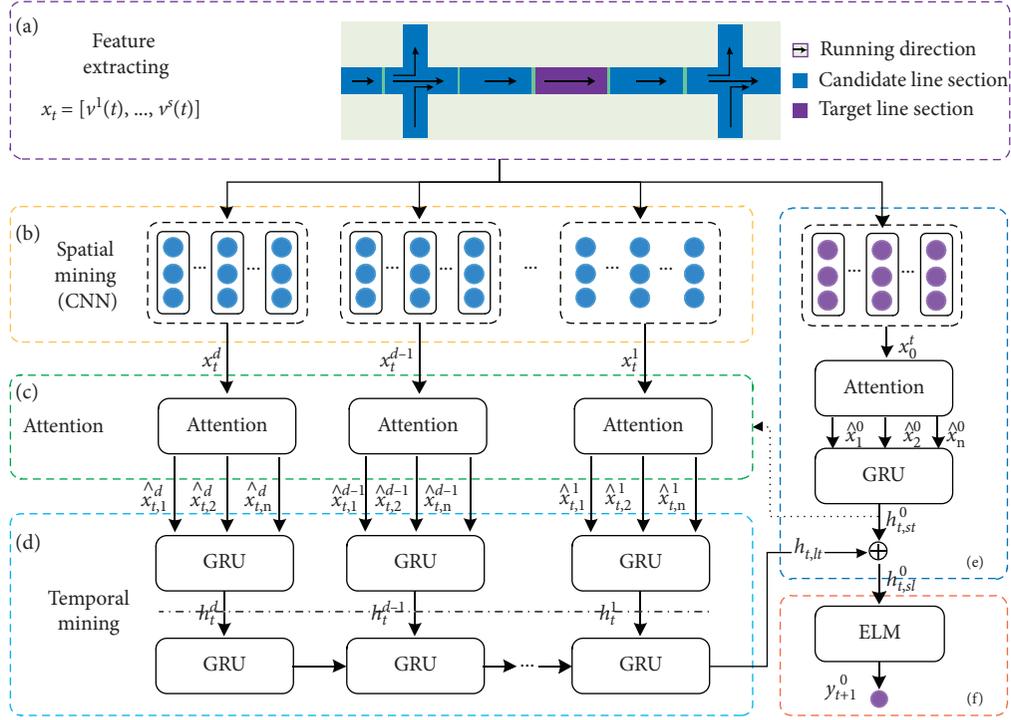


FIGURE 5: The architecture of the DHSTN.

the first GRUs, and for each previous day \hat{x}_k^s , the final time t data h_t^s are obtained. The long-term dependency analysis is in the second GRUs, and for each h_t^s , the long-term information $h_{t,ls}$ could be preserved by $h_{t,ls}^q = \text{GRU}(h_{t,ls}^{q+1}, h_t^q)$, $1 \leq q \leq d-1$, and $h_{t,ls}^q$ is the output of the $(q+1)$ -th GRU and h_t^q is the input of the q -th GRU. (e) The fifth step is the short-term dependency analysis for the target line section. It also includes spatial and temporal dependency mining, which is similar to the step (b), (c), and (d), and the main difference is that step (e) uses one GRU layer to analyze the short-term dependency $h_{t,st}^0$ of target line section. The final output $h_{t,sl}^0$ is the concatenation of $h_{t,st}^0$ and $h_{t,ls}^0$, $h_{t,sl}^0 = h_{t,st}^0 \oplus h_{t,ls}^0$, which preserves the short-term and long-term temporal dependency information. (f) The sixth step is the final prediction, and it uses ELM to fit $y_{t+1}^0 = f(w_f h_{t,sl}^0 + b_f)$ and get the final result. It is noted that the data during the whole processing are all centered on the target line section, and the superscript "0" indicates the day of time to predict.

5. Experiments

For verifying the proposed fusion model (DHSTN), real bus speed datasets are used to conduct experiments by comparing with other models (ARIMA, MLP [45], RBFNN [46], TM-CNN [47], FDL [19], DNN-BTF [25], RS-DHSTN (its feature extracting strategy is replaced with random selection), KNN-DHSTN (its feature extracting strategy is the k -nearest neighbor method), and NAM-DHSTN (it is a special DHSTN without attention module)). These data are sampled from the Huangpu road of Dalian city, shown in Figure 6,

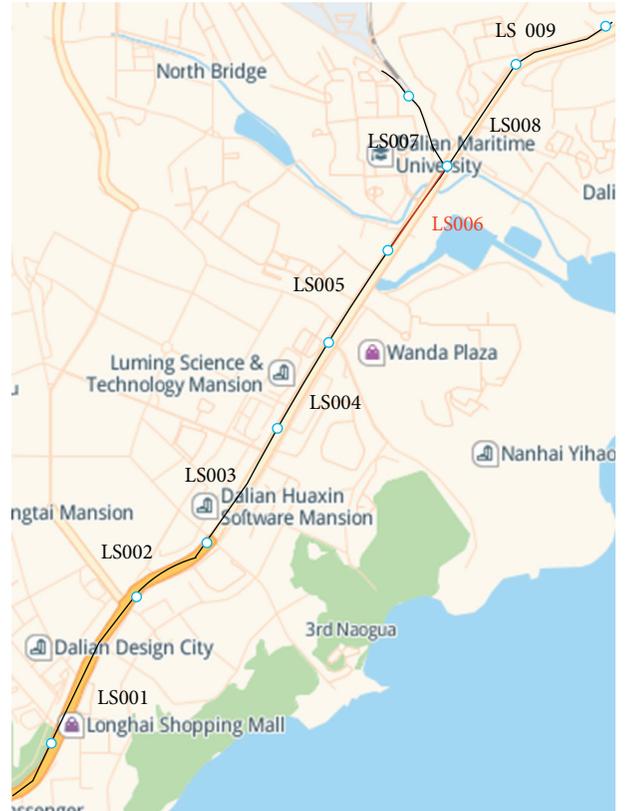


FIGURE 6: The observation and experimental bus line sections on No. 3 bus route of Dalian.

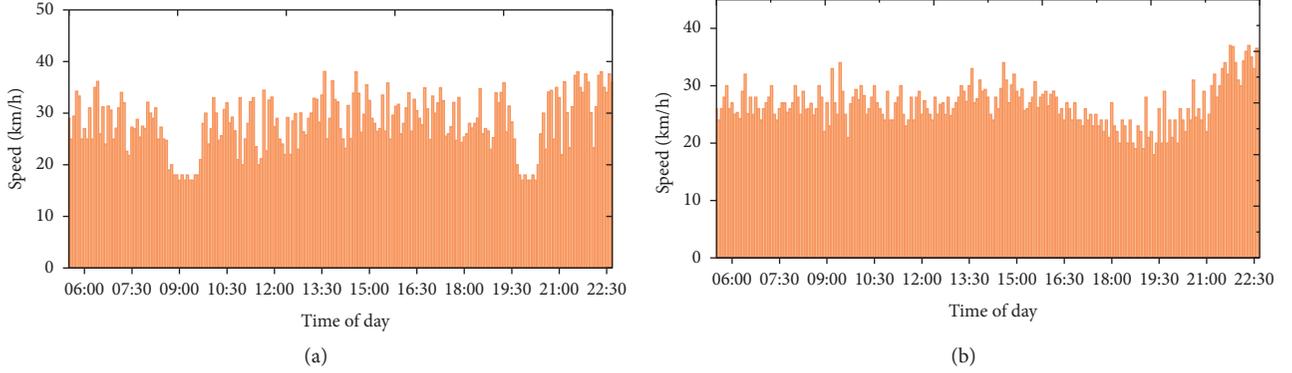


FIGURE 7: The average speed of a day. (a) Weekday. (b) Weekend.

where the line sections include LS001, LS002, \dots , LS009, and LS006 is the target line section. The sampling period is from November 5, 2018, to November 18, 2018, and from December 3, 2018, to December 16, 2018. For real requirements, original data are aggregated into time series in a 5-minute interval so as to generate an average speed time series, based on the equation (4)–(7), shown in Figure 7. Moreover, the model performances are evaluated by the mean absolute error (MAE), the mean absolute percent error (MAPE), and the root mean-squared error (RMSE).

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - y'_i|, \quad (12)$$

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - y'_i|}{y_i}, \quad (13)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2}. \quad (14)$$

5.1. Average Speed Analysis. For testing the performances of the average speed analysis method (DWCM) proposed in the paper, we use the method based on speed integral (SI method) proposed by Quiroga and Augusto [48]. Figure 8 is the average speed analyzing result. As is shown, comparing with the SI-method, the proposed method could compute the average speed of line sections more accurately.

In addition, the unstable states of bus entering and leaving the station have impacts on analyzing average speed. For verifying these, the comparing tests are carried out under considering/not considering the unstable states (C-US and NC-US). Table 1 is the result error of the two methods. Under considering the unstable state, the MAPE of the DWCM is only 6.53%, which is far less than that of the SI method, 9.47%. As is shown, it is indicated that independent of whether the unstable states are considered or not, the DWCM method for analyzing the average speed is better than the SI method.

Figure 7(a) is the average speed of a weekday. The tendency of speed is different from traditional traffic speed,

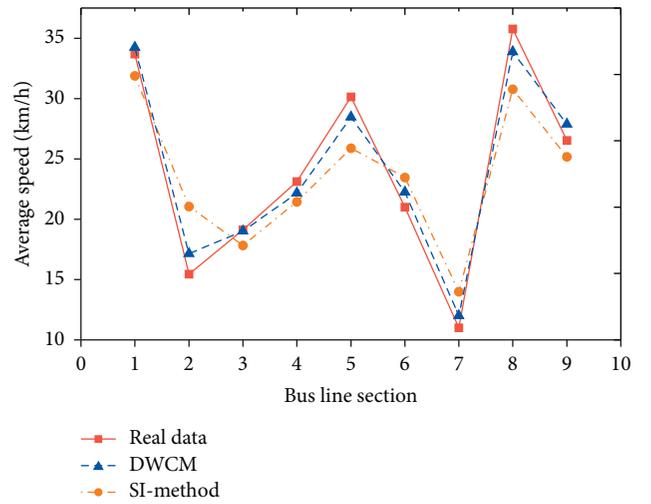


FIGURE 8: The average speed analyzing results.

TABLE 1: Comparative analysis of different methods.

Error	DWCM		SI method	
	C-US ¹	NC-US ²	C-US ¹	NC-US ²
MAE	2.57	3.38	3.88	5.19
MAPE	6.53%	8.17%	9.47%	11.06%
MAX (MAE)	5.08	8.57	11.17	14.88

¹C-US is considering the unstable state. ²NC-US is not considering the unstable state.

TABLE 2: GRGs of the target line section.

	Line section	GRGs
1	LS001	0.5182
2	LS002	0.5933
3	LS003	0.6967
4	LS004	0.7727
5	LS005	0.8593
6	LS006	1
7	LS007	0.8107
8	LS008	0.8205
9	LS009	0.7188

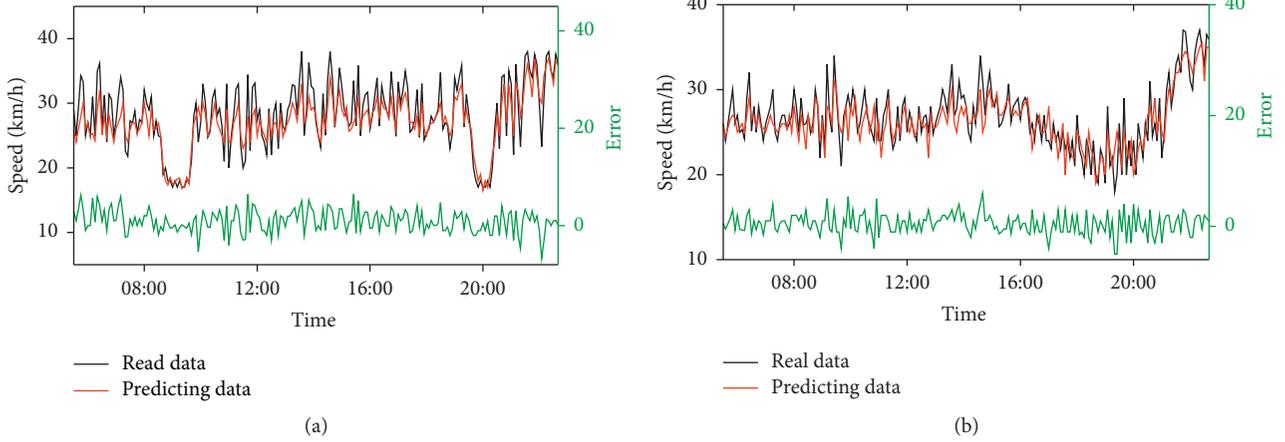


FIGURE 9: Prediction results of the DHSTN at target line section LS006. (a) On November 15, 2018 (weekday). (b) On December 15, 2018 (weekend).

on weekdays, its peak time is delay, and its changes are not much greater than that of traditional traffic speed in peak time. The reason is that, in the peak time, there are bus lanes, whose time is from 6:30 am to 8:30 am in the morning and from 16:00 pm to 19:00 pm. After the time, the traffic peak is almost over. Because of the works and other reasons, the traffic state on weekends is stable, and the bus speed is much smoother, as shown in Figure 7(b).

5.2. Feature Extraction. Table 2 is the results of extracting spatial features within the bus line sections given in Figure 6. As is shown, GRG values of the candidate line sections near the target section are larger, and it means that these sections are more likely to be chosen as inputs of the DHSTN. Besides, GRG values of upstream sections are larger than that of downstream ones, and it indicates that the upstream line state has greater impacts on the target section than that of the downstream ones. If $\lambda = 4$, the candidates with top 5 GRGs are selected to construct the input matrix.

5.3. Prediction and Analysis. Figure 9 shows the prediction results of the DHSTN on a weekday/weekend. From the figure, it is known that the DHSTN is capable of efficiently fitting the tendency of bus speed and also capturing the sudden change with an excellent performance. Table 3 is the analysis of prediction performances of the DHSTN. It indicates that the DHSTN is working better for the weekend than that for the weekday due to relatively stable speed, and its performance for weekends is superior to that for weekdays, which is improved of 0.61, 0.89%, and 0.56 on MAE, MAPE, and RMSE, respectively, because on weekends, the buses run more smoothly.

To evaluate the performances of the DHSTN and test the importance of the two modules of the DHSTN, the feature extracting module and attention module, the comparison experiments with the RS-DHSTN, KNN-DHSTN, and NAM-DHSTN are conducted. Table 4 shows the average evaluation results for week datasets. From Table 4, it is

TABLE 3: Prediction performance of the DHSTN.

	MAE	MAPE (%)	RMSE
Weekday	2.9925	6.9477	3.4943
Weekend	2.3898	6.0518	2.9338

TABLE 4: Comparison analysis of the DHSTN itself.

	MAE	MAPE (%)	RMSE
DHSTN	2.6912	6.3628	3.2141
RS-DHSTN	4.9470	10.3966	5.5799
KNN-DHSTN	3.7953	8.3750	4.1701
NAM-DHSTN	3.2396	7.5636	3.8937

observed that the DHSTN is better and has smaller error. Comparing with random selection and the KNN, the DHSTN improves by 45.60% and 29.09% on MAE, respectively, and 38.80% and 24.03% on MAPE, respectively. It indicates that the feature extracting module is the basic, and better candidate selection strategy could improve the performances of the DHSTN. Comparing with the NAM-DHSTN, it improves by 16.93%, 15.88%, and 17.45% on MAE, MAPE, and RMSE, respectively, and it means that the attention module could analyze the importance of each near-term time data and improve the prediction accuracy.

For verifying the performances of the DHSTN further, several methods are exploited to carry out the comparative experiments. Table 5 is the comparison of prediction results by considering a week as a whole. From the table contents, the DHSTN works better than other methods, it outperforms the second better model DNN-BTF with the improvement of 21.06% and 8.36% on MAE and MAPE, respectively, and compared with the nonparameter method MLP, it improves with 41.30%, 49.62%, and 47.69% on MAE, MAPE, and RMSE, respectively. The results indicate that the DHSTN can efficiently capture the spatial and temporal features of urban bus speed and has better performances. Among these models, ARIMA and MLP only analyzed the

TABLE 5: Comparison analysis of different methods.

	MAE	MAPE (%)	RMSE
ARIMA	4.1999	12.2068	6.3360
MLP	4.5849	12.9016	6.1442
RBFNN	5.0257	13.4208	6.9826
TM-CNN	3.5414	10.2262	4.3560
FDL	3.4091	6.8202	4.5939
DNN-BTF	3.0696	6.9830	3.7008
DHSTN	2.6912	6.3628	3.2141

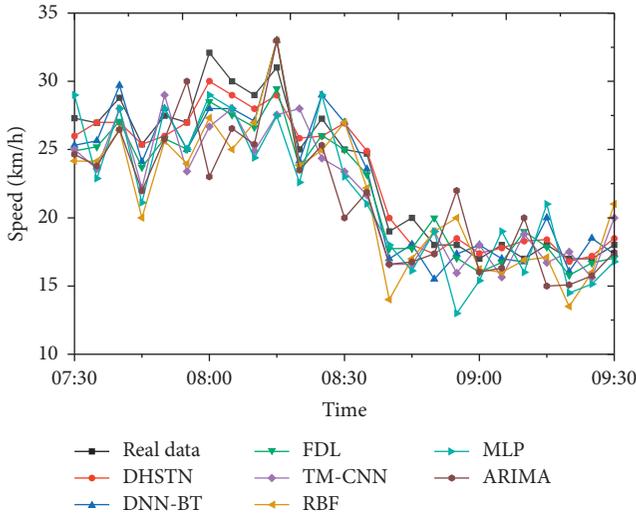


FIGURE 10: Part prediction results of the DHSTN on weekdays.

time series information, but their performances are better than those of RBFNN, which considered spatial and temporal features. It means that the temporal features are essential for short-term prediction, and spatial information may lead predictor worse, so reasonable utilization of spatial and temporal information is of vital importance.

Figures 10 and 11 are the part prediction results of each model on the weekday and weekend, respectively (from 7:30 to 9:30 on the weekday and from 16:30 to 18:30 on the weekend). From the changing tendency, these models can capture the features of bus speed, while the spatial-temporal models are better. The difference between real data and prediction results of the DHSTN is around 2.6, and the least difference of other predictors is about 3.1. Thus, the DHSTN proposed in the paper has the better performances.

Table 6 is the comparison analysis of different models on weekdays and weekends. The error of the DHSTN is the least among these models, the least MAE is 2.3898, comparing with others, and its accuracy is improved by 41.38%, 45.73%, 26.72%, 16.71%, and 12.25% on weekdays and 46.25%, 49.84%, 56.05%, 39.76%, 31.03%, and 19.18% on weekends, respectively. It is noted that the MAE and RMSE of each models on weekends are similar to that on weekdays, while the MAPE on weekdays is much larger than that on weekends, which means that the prediction results on weekends are better. The reason is that different from weekdays, the urban traffic on weekends is more stable and the bus speed is smoother.

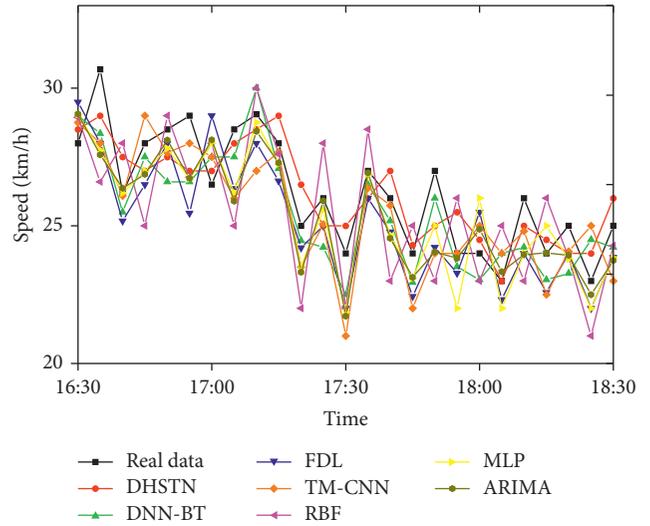


FIGURE 11: Part prediction results of the DHSTN on weekends.

TABLE 6: Comparison analysis of different methods on weekdays/weekends.

Methods	Weekdays			Weekends		
	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE
ARIMA	4.2876	10.18	6.7325	4.1464	7.06	6.3364
MLP	4.7640	11.15	6.5344	4.5689	7.03	6.1384
RBFNN	5.4381	11.85	7.0823	5.1463	8.47	6.9952
TM-CNN	3.8112	8.49	4.5543	3.7653	5.96	4.4355
FDL	3.3529	7.35	4.4938	3.4652	6.84	4.6941
DNN-BTF	3.1825	7.13	3.8927	2.9568	6.51	3.5088
DHSTN	2.7925	6.95	3.4943	2.3898	6.05	2.9338

6. Conclusions

In this paper, the original bus speed data were deeply analyzed and constructed as the time series with the time interval of 5 minutes based on the proposed dynamic weight computing method to generate the bus travel average speed in each line section. Referring to relative models, the Dynamic Hierarchical Spatial-Temporal Network model (DHSTN) is proposed, which consists of the GRU, CNN, EGRA, and attention mechanism. First, it exploits EGRA to analyze the candidate line sections based on the correlations, in order to choose the suitable candidates with high impacts on the target section. In the next, the DHSTN constructs multilayer structure by using the CNN, GRU, and attention mechanism to analyze and capture the spatial dependency, long-term and short-term temporal dependency. Finally, by ELM, it fuses the spatial dependency of nearby line sections and temporal dependency of the target section to predict the bus speed variation in the next time interval. For verifying the model performances, the comparative experiments are conducted with ARIMA, MLP, RBF, TM-CNN, FDL, and DNN-BTF,

and the results show that the proposed prediction model can accurately capture and predict the bus speed variation.

Data Availability

The urban bus travel speed data used to support the findings of this study have not been made available, because the data are from the operation system of relevant enterprise, according to the operation specifications, these data are still in the confidential stage, and we are not authorized to release the data.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 3132019354, Scientific Research Fund of Liaoning Provincial Education Department under Grant JYT2019LQ01, National Natural Science Foundation of China under Grant 51679025, and Science and Technology Program Foundation of Liaoning Province under Grant 2019JH2/10300052.

References

- [1] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: where we are and where we're going," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 3–19, 2014.
- [2] Y. Sun, G. Jiang, S. K. Lam, S. Chen, and P. He, "Bus travel speed prediction using attention network of heterogeneous correlation features," in *Proceedings of the 2019 SIAM International Conference on Data Mining*, Calgary, Canada, May 2019.
- [3] A. Comi, M. Zhuk, V. Kovalyshyn, and V. Hilevych, "Investigating bus travel time and predictive models: a time series-based approach," *Transportation Research Procedia*, vol. 45, pp. 692–699, 2020.
- [4] R. H. Jeong, *The Prediction of Bus Arrival Time using Automatic Vehicle Location Systems Data*, Texas A & M University, 2004.
- [5] N. Julio, R. Giesen, and P. Lizana, "Real-time prediction of bus travel speeds using traffic shockwaves and machine learning algorithms," *Research in Transportation Economics*, vol. 59, pp. 250–257, 2016.
- [6] H. Yang, Y. Zhang, L. Zhang, and T. Xu, "Research on floating car speed short-time prediction with wavelet-ARIMA under data missing," in *Green Intelligent Transportation Systems*, pp. 289–298, Springer, Berlin, Germany, 2019.
- [7] M. Napiah and I. Kamaruddin, "ARIMA models for bus travel time prediction," *Journal of Industrial Engineering and Management Malaysia*, vol. 71, pp. 49–58, 2010.
- [8] B. Hu, K. Xie, H. Cui, and H. Lin, "A Bayesian spatiotemporal approach for bus speed modeling*," in *Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 497–502, Auckland, New Zealand, October 2019.
- [9] J. Wang, B. Yu, and Y. zhong, "Bus travel-time prediction based on bus speed," *Proceedings of the Institution of Civil Engineers-Transport*, vol. 163, pp. 3–7, 2010.
- [10] X. Wang, H. Han, J. Qie, S. Li, C. Zhang, and H. Wang, "RBFNN-bagging-model-based study on bus speed prediction," *Green, Smart and Connected Transportation Systems*, Springer, Berlin, Germany, pp. 1009–1024, 2020.
- [11] Y. Zhu, D. Zhang, and J. W. Zhang, "The prediction of bus speed based on Kalman filter," *Advanced Materials Research*, vol. 962–965, pp. 2818–2822, 2014.
- [12] J. Pan, X. Dai, X. Xu, and Y. Li, "A self-learning algorithm for predicting bus arrival time based on historical data model," in *Proceedings of the 2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, pp. 1112–1116, Hangzhou, China, October 2012.
- [13] Y. Bai, Z. Sun, B. Zeng, J. Deng, and C. Li, "A multi-pattern deep fusion model for short-term bus passenger flow forecasting," *Applied Soft Computing*, vol. 58, pp. 669–680, 2017.
- [14] Z. Ma, J. Xing, M. Mesbah, and L. Ferreira, "Predicting short-term bus passenger demand using a pattern hybrid approach," *Transportation Research Part C: Emerging Technologies*, vol. 39, pp. 148–163, 2014.
- [15] M. Yang, C. Chen, L. Wang, X. Yan, and L. Zhou, "Bus arrival time prediction using support vector machine with genetic algorithm," *Neural Network World*, vol. 26, no. 3, pp. 205–217, 2016.
- [16] A. O. Hashi, S. Z. M. Hashim, T. Anwar, and A. Ahmed, "A Robust hybrid model based on kalman-SVM for bus arrival time prediction," *Advances in Intelligent Systems and Computing*, Springer, Berlin, Germany, pp. 511–519, 2020.
- [17] H. Liu, H. Xu, Y. Yan, Z. Cai, T. Sun, and W. Li, "Bus arrival time prediction based on LSTM and spatial-temporal feature vector," *IEEE Access*, vol. 8, pp. 11917–11929, 2020.
- [18] H. Cui, K. Xie, B. Hu, H. Lin, and R. Zhang, "Analysis of bus speed using a multivariate conditional autoregressive model: contributing factors and spatiotemporal correlation," *Journal of Transportation Engineering, Part A: Systems*, vol. 145, no. 4, 2019.
- [19] Y. Gu, W. Lu, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: a fusion deep learning model," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 1–16, 2019.
- [20] W. Lu, Y. Rui, Z. Yi, B. Ran, and Y. Gu, "A hybrid model for lane-level traffic flow forecasting based on complete ensemble empirical mode decomposition and extreme gradient boosting," *IEEE Access*, vol. 8, pp. 42042–42054, 2020.
- [21] F. Guo, J. W. Polak, and R. Krishnan, "Predictor fusion for short-term traffic forecasting," *Transportation Research Part C: Emerging Technologies*, vol. 92, pp. 90–100, 2018.
- [22] Y. Gu, W. Lu, X. Xu, Z. Shao, and H. Zhang, "An improved bayesian combination model for short-term traffic prediction with deep learning," *Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, 2020.
- [23] A. Bachu, L. Vanajakshi, and S. Subramanian, "Bus travel time prediction using a time-space discretization approach," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 308–332, 2017.
- [24] J. Ma, J. Chan, G. Ristanoski, S. Rajasegarar, and C. Leckie, "Bus travel time prediction with real-time traffic information," *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 536–549, 2019.
- [25] W. Yuankai, H. Tan, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.
- [26] T. Bogaerts, A. D. Masegosa, J. S. Angarita-Zapata, E. Onieva, and P. Hellinckx, "A graph CNN-LSTM neural network for

- short and long-term traffic forecasting based on trajectory data,” *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 62–77, 2020.
- [27] 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,” *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.
- [28] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, “Modeling spatial-temporal dynamics for traffic prediction,” 2018.
- [29] 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,” *Transportation Research Part C: Emerging Technologies*, vol. 108, pp. 12–28, 2019.
- [30] Y. Duan, L. Yisheng, and F. Y. Wang, “Travel time prediction with LSTM neural network,” in *Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1053–1058, Rio de Janeiro, Brazil, November 2016.
- [31] K. Cho, B. V. Merriënboer, C. Gulcehre et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, October 2014.
- [32] H. Abbasimehr, M. Shabani, and M. Yousefi, “An optimized model using LSTM network for demand forecasting,” *Computers & Industrial Engineering*, vol. 143, Article ID 106435, 2020.
- [33] Z. Cui, R. Ke, and Y. Wang, “Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction,” 2018.
- [34] Y. Miao, J. Han, Y. Gao, and B. Zhang, “ST-CNN: spatial-temporal convolutional neural network for crowd counting in videos,” *Pattern Recognition Letters*, vol. 125, pp. 113–118, 2019.
- [35] P. Zhang, T. Zhuo, W. Huang, K. Chen, and M. Kankanhalli, “Online object tracking based on CNN with spatial-temporal saliency guided sampling,” *Neurocomputing*, vol. 257, pp. 115–127, 2017.
- [36] Y. Hu and X. Lu, “Learning spatial-temporal features for video copy detection by the combination of CNN and RNN,” *Journal of Visual Communication and Image Representation*, vol. 55, pp. 21–29, 2018.
- [37] G. Hinton, “Rectified linear units improve restricted Boltzmann machines Vinod Nair,” in *Proceedings of ICML*, pp. 807–814, Haifa, Israel, June 2010.
- [38] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, “Striving for simplicity: the all convolutional net,” 2014.
- [39] A. Karpathy, J. Johnson, and L. Fei-Fei, “Visualizing and understanding recurrent networks,” 2015.
- [40] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, “Revisiting spatial-temporal similarity: a deep learning framework for traffic prediction,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 5668–5675, 2019.
- [41] B. Yang, S. Sun, J. Li, X. Lin, and Y. Tian, “Traffic flow prediction using LSTM with feature enhancement,” *Neurocomputing*, vol. 332, pp. 320–327, 2019.
- [42] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, September 2015.
- [43] Y. Zhang, Y. Zhang, and A. Haghani, “A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model,” *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 65–78, 2013.
- [44] Q. Zhang, J. Deng, and Y. Shao, “A grey correlational analysis by the method of degree of balance and approach,” *Journal of Huazhong University of Science and Technology*, vol. 23, pp. 94–98, 1995.
- [45] A. Raza and M. Zhong, “Hybrid lane-based short-term urban traffic speed forecasting: a genetic approach,” in *Proceedings of the 2017 4th International Conference on Transportation Information and Safety (ICTIS)*, pp. 271–279, Banff, Canada, August 2017.
- [46] A. W. Kuang and Z. X. Huang, “Short-term traffic flow prediction based on RBF neural network,” *Systems Engineering*, vol. 22, no. 2, pp. 63–65, 2004.
- [47] 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,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 4, pp. 459–470, 2020.
- [48] C. Quiroga and C. Augusto, *An Integrated GPS-GIS Methodology for Performing Travel Time Studies*, Louisiana State University, Baton Rouge, LA, USA, 1997.