



Urban traffic congestion estimation and prediction based on floating car trajectory data

Xiangjie Kong^a, Zhenzhen Xu^{a,*}, Guojiang Shen^b, Jinzhong Wang^a, Qiuyuan Yang^a, Benshi Zhang^a

^a School of Software, Dalian University of Technology, Dalian 116620, China

^b College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

HIGHLIGHTS

- Floating car trajectory data can be used to predict traffic congestion effectively.
- A Fuzzy Comprehensive Evaluation method with dynamic adaptive weight is introduced.
- A Traffic Flow Prediction method utilizing particle swarm optimization is proposed.
- Experiments verify methods' performances in accuracy, instantaneity and stability.

ARTICLE INFO

Article history:

Received 1 July 2015

Received in revised form

23 September 2015

Accepted 10 November 2015

Available online xxxx

Keywords:

Floating car trajectory data

Particle swarm optimization

Congestion estimation

Traffic flow prediction

Fuzzy comprehensive evaluation

ABSTRACT

Traffic flow prediction is an important precondition to alleviate traffic congestion in large-scale urban areas. Recently, some estimation and prediction methods have been proposed to predict the traffic congestion with respect to different metrics such as accuracy, instantaneity and stability. Nevertheless, there is a lack of unified method to address the three performance aspects systematically. In this paper, we propose a novel approach to estimate and predict the urban traffic congestion using floating car trajectory data efficiently. In this method, floating cars are regarded as mobile sensors, which can probe a large scale of urban traffic flows in real time. In order to estimate the traffic congestion, we make use of a new fuzzy comprehensive evaluation method in which the weights of multi-indexes are assigned according to the traffic flows. To predict the traffic congestion, an innovative traffic flow prediction method using particle swarm optimization algorithm is responsible for calculating the traffic flow parameters. Then, a congestion state fuzzy division module is applied to convert the predicted flow parameters to citizens' cognitive congestion state. Experimental results show that our proposed method has advantage in terms of accuracy, instantaneity and stability.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Urban traffic congestion has become a critical problem that not only affects the people's daily lives, but also restricts the stable development of society and economy [1,2]. Hence, it is urgent to ease the traffic congestion, especially the recurrent congestion. Nevertheless, urban traffic flow is complex and constantly changing, and then it is difficult for travelers to acquire the current and the future traffic condition at road sections.

There are two major challenging problems that should be answered to perform urban traffic congestion estimation and

prediction. Firstly, how to estimate and predict the traffic congestion in large-scale urban areas? Floating car, namely Global Position System (GPS)-equipped taxi, is an effective way to collect the real-time traffic flow data in a large-scale road network, which can be regarded as ubiquitous mobile sensors probing a city's rhythm and pulse [3]. In addition, floating car has more acceptable cost than the traditional methods with fixed sensors collecting data at fixed trunks or major intersections. Secondly, how to improve the accuracy, instantaneity and stability of traffic congestion estimation and prediction? The existing methods have not addressed the three performance aspects systematically.

We propose a novel method using floating car trajectory data to improve the overall performance of traffic congestion estimation and prediction. The paper is based on our previous work [4]. The

* Corresponding author.

E-mail address: xzz@dlut.edu.cn (Z. Xu).

major difference between this work and the prior one lies on the proposal of new traffic congestion estimation method and the improvement of traffic congestion prediction combined with the new estimation module.

In the matter of traffic congestion estimation, we propose a Fuzzy Comprehensive Evaluation (FCE) method in which the weights and the fuzzy matrix of multi-indexes can be adapted according to the traffic flows. Assigning the multi-indexes weights in accordance with the congested traffic flow dynamically improves the estimation accuracy and instantaneity performance in comparison to the fixed weighting method.

For traffic congestion prediction, we propose a novel method including Traffic Flow Prediction (TFP) and Congestion State Fuzzy Division (CSFD) modules. The former predicts traffic flow parameters by using Particle Swarm Optimization (PSO) algorithm. The latter converts the predicted traffic flow parameters to citizens' cognitive congestion state using the proposed FCE method. TFP module is composed by three sub modules: Traffic Volume Prediction (TVP), Traffic Speed Prediction (TSP) and PSO. TVP sub module predicts the traffic volume, while TSP sub module is for predicting average speed. Furthermore, PSO sub module optimizes the punish coefficients and the multi-kernel functions' parameters of Support Vector Machine (SVM) in TVP and TSP sub modules. The reason for choosing PSO algorithm is that it can get the optimum solution in a short time with a low computing complexity which meets the performance requirements of congestion prediction in terms of accuracy, instantaneity and stability.

The rest of this paper is organized as follows. In Section 2, we present related works about urban traffic congestion estimation and prediction. Then, our proposed congestion estimation and prediction method is introduced in Sections 3 and 4 respectively. In Section 5, experiment results are described. Finally, the paper is concluded in Section 6.

2. Related work

In this section, we present an overview of urban traffic congestion estimation and prediction methods based on the recent literature.

Pattara-Atikom et al. [5] estimated the traffic congestion using weighted exponential moving averages of measured GPS speed. Yoon et al. [6] proposed a simple yet effective method using the spatial and temporal speed information in order to estimate traffic status on surface streets with GPS location data. The authors in [7] estimated the traffic state by calculating the average velocities along GPS-equipped vehicles' tracks. Similarly, an improved method was presented in [8] that makes use of both time-varying and space-varying information to predict the urban traffic state based on an adaptive cubic surface traffic flow model.

Kong et al. [9] presented a systematic solution to efficiently predict traffic state by extracting the spatio-temporal average velocity from a large number of GPS probe vehicles. The method was based on a curve-fitting and vehicle-tracking mechanism. In order to improve the estimation accuracy, the same authors in [10] calculated mean speed at road section from multi-source traffic data to estimate the traffic states. Zhang et al. [11] proposed a weighted approach to estimate traffic state using GPS data by increasing the weights of recent velocity information. Li et al. [12] presented a hybrid learning framework to appropriately combine estimation results of freeway traffic density state from multiple macroscopic traffic flow models. Feng et al. [13] proposed a cooperative approach to estimate arterial travel time states including Bayesian and Expectation Maximization algorithms using GPS probe data.

The above-mentioned methods estimated the traffic state using one specific parameter such as average velocity, travel time

or traffic density. However, the uncertainty and complexity of traffic state have not been studied in these methods sufficiently. To address these shortcomings, Lu et al. [14] evaluated traffic congestion state using an adaptive neuro-fuzzy inference system in which a series of fuzzy logic rules was extracted. Pongpaibool et al. [15] presented a traffic congestion estimation system from video data using manually tuned fuzzy logic. However, vehicle volume and velocity were used in this method without considering the road space information. Chen et al. [16] rendered a tracking-based method using Pareto optimal decision theory and fuzzy comprehensive judgment in order to estimate the traffic state. Shankar et al. [17] explored advantages of fuzzy inference systems to evaluate the level of road traffic congestion using traffic density and speed information.

Related to traffic flow and congestion prediction, Su et al. [18] proposed a hybrid traffic flow prediction model using Genetic algorithm (GA) to optimize the input parameters. Then, the SVM method was used to update the prediction function through an online learning process. Similarly, Cao et al. [19] used an improved PSO algorithm to preprocess the input parameters of the SVM method. Xu et al. [20] presented a spatio-temporal variable selection method based on Support Vector Regression (SVR) model to predict traffic volume. In this method, the spatial and temporal information of all available road segments was utilized.

Hong et al. [21] presented a SVR traffic flow forecasting model using Gaussian Radial Basis Function (RBF) kernel. In this method, a hybrid Genetic Algorithm with Simulated Annealing is used to forecast the RBF suitable parameters accurately. Li et al. [22] applied SVR model with Gauss loss function (Gauss-SVR) to forecast urban traffic flow and proposed a Chaotic Cloud Particle Swarm Optimization algorithm to optimize the parameters of Gauss-SVR model. Wang et al. [23] proposed a traffic speed forecasting model using chaos-wavelet analysis and SVM to choose the appropriate kernel function. Wang et al. [24] proved that selecting the appropriate SVR parameters improves the prediction of traffic flow in terms of the instantaneity and accuracy performance metrics. Gong et al. [25] proposed a traffic volume forecasting model based on SVR and analyzed the nature of the RBF kernel function.

The discussed methods above only considered one kernel function of the SVM method to improve the accuracy of urban traffic state prediction, while the road traffic congestion cannot be predicted by these methods accurately. The reason is that various SVM kernel functions have different prediction accuracy and adaptability. In addition, the predicted traffic flow cannot intuitively forecast the future traffic congestion for the travelers and the traffic administrators. To tackle these shortcomings, Hen et al. [26] proposed an accurate particle filter method to predict multi-step traffic state using the speed measurements. Similarly, Dunne et al. [27] proposed a regime-based multivariate traffic condition prediction method using an Artificial Neural Network (ANN) structure with adaptive learning strategies. Min et al. [28] presented a scalable multivariate spatial-temporal autoregressive model to predict the traffic volume and speed jointly. Zhang et al. [29] proposed a robust traffic congestion prediction method based on hierarchical fuzzy rule-based systems and GA, which combines the variable selection, ranking and lateral tuning of the membership functions with optimization of the rule base.

Closely related to traffic congestion estimation and prediction, Herring et al. [30] proposed two statistical learning algorithms which use data from GPS-equipped smart phones. In this method, logistic regression and spatio-temporal autoregressive moving average models are employed to estimate and forecast the arterial traffic conditions. Castro et al. [31] proposed a method to construct a model of traffic density based on large scale taxi traces, and used the model to predict future traffic conditions according to

the probabilistic transition matrix. To conduct a comprehensive and accurate traffic flow analysis, Zhou et al. [32] proposed a traffic condition estimation and prediction method based on Least Squares Support Vector Machine (LS-SVM) classification and regression using the floating car data.

The above-mentioned methods have not considered the traffic capacity as well as the spatial information of the roads. In addition, most of them have considered only one single performance metric and there is a lack of a systematic method to address accuracy, instantaneity and stability at the same time. In order to tackle this issue, we propose a new method to estimate and predict traffic congestion to improve the three performance metrics simultaneously. Our proposed method is described in Sections 3 and 4.

3. Traffic congestion estimation

In this section, we describe FCE congestion estimation method to generate appropriate factors and evaluation sets for traffic congestion. In addition, the weight coefficients of each evaluation factor as well as the membership functions are determined. Finally, the whole process of our proposed method is described.

3.1. Determine factor and evaluation sets of traffic congestion

Understanding traffic flow in urban areas is the first step in predicting traffic congestion. Traffic flow mainly includes traffic volume, traffic density, and traffic speed. Up to now, one or two traffic parameters have been used in existing methods which are insufficient to explore the underlying traffic dynamics deeply. We believe that selecting appropriate traffic flow parameters can characterize the traffic congestion state accurately. In addition, it has better computational complexity to deal with real-time scenarios.

In fact, the volume and speed of urban traffic can be extracted from the floating car trajectory data directly. Taking the length and the lane number of the road section as well as the traffic volume into consideration, we can analyze the traffic density to reflect the level of road congestion. Furthermore, considering the traffic capacity and traffic volume, the road saturation can be adopted to intuitively reflect the capacity of road traffic facilities or actual load condition of the road. Traffic speed can also reflect dynamic operation conditions of road traffic. Thus, road saturation (*Sat*), traffic density (*Den*), and traffic speed (*Spe*) are leading factor sets to characterize the traffic congestion accurately.

When exploring traffic flow, we measure traffic congestion state through investigation. Then, we choose appropriate traffic congestion state as the evaluation sets. Note that identifying the cognitive traffic congestion state plays an important role in travel navigation and traffic guidance systems. Through investigation, we determine ‘very smooth’, ‘smooth’, ‘mild congestion’, ‘moderate congestion’, and ‘serious congestion’ as the evaluation sets.

3.2. Determine weight coefficients of the factors

After creating the traffic congestion factor sets in the previous subsection, we need to determine the weight coefficient of each factor. There are two methods to determine the weight coefficients of factors, namely subjective weighting and objective weighting. The subjective weighting method determines the weights by experts’ experience. However, the results obtained from this method is subjective and random since every expert has different weights for each factor. The latter method determines the weights according to the correlation between each factor, which are sometimes inconsistent with realistic data.

In order to avoid the shortcomings of the subjective and objective weighting methods, we determine the weight coefficients based on the traffic flow characteristics and the experience of experts. The weight coefficient set of factors is expressed as $W = (w_1, w_2, w_3, \dots, w_i)$, and $w_1 + w_2 + w_3 + \dots + w_i = 1$ where weight coefficients are normalized between 0 and 1. Our proposed method improves the estimation accuracy and real-time performance comparing to the fixed weighting method.

3.3. Determine the membership functions

Determining the membership functions is the main step for a fuzzy comprehensive evaluation method. The membership functions determine the relationship between factor sets (road saturation, traffic density, and traffic speed) and evaluation sets ('very smooth', 'smooth', 'mild congestion', 'moderate congestion', and 'serious congestion') for traffic congestion. For this purpose, we choose trapezoidal membership functions as our membership function, and its function graph is shown as follows.

In Fig. 1, the parameters a, b, c, d are the thresholds corresponding to five traffic congestion states, and k_1, k_2, \dots, k_8 are linear values corresponding to a, b, c, d .

Accordingly, the functional expression is shown in Eq. (1).

$$\begin{aligned} V_1(u) &= \begin{cases} 1 & u \leq k_1 \\ (k_2 - u) / (k_2 - k_1) & k_1 < u < k_2 \\ 0 & u \geq k_2 \end{cases} \\ V_2(u) &= \begin{cases} (u - k_1) / (k_2 - k_1) & k_1 < u < k_2 \\ 1 & k_2 \leq u \leq k_3 \\ (k_4 - u) / (k_4 - k_3) & k_3 < u < k_4 \\ 0 & u \geq k_4 \end{cases} \\ &\dots \\ V_5(u) &= \begin{cases} 0 & u \leq k_7 \\ (u - k_7) / (k_8 - k_7) & k_7 < u < k_8 \\ 1 & u \geq k_8 \end{cases} \end{aligned} \quad (1)$$

where $V_1(u), V_2(u), V_3(u), V_4(u)$ and $V_5(u)$ are all within the range of [0, 1] and each of them denotes one traffic congestion state. The piecewise functions $V_3(u)$ and $V_4(u)$ are similar to $V_2(u)$, which are omitted.

3.4. FCE congestion estimation method

Traffic congestion factor and evaluation sets.

The traffic congestion factor sets are expressed as $U = \{u_1, u_2, u_3\}$ corresponding to the road saturation, traffic density, and traffic speed. The evaluation sets are also expressed as $V = \{v_1, v_2, v_3, v_4, v_5\}$ corresponding to ‘very smooth’, ‘smooth’, ‘mild congestion’, ‘moderate congestion’ and ‘serious congestion’. Moreover, we denote that u_1 (or u_2) $\rightarrow \{v_1, v_2, v_3, v_4, v_5\}$ and $u_3 \rightarrow \{v_5, v_4, v_3, v_2, v_1\}$, which mean that the smaller road saturation (or traffic density) indicates the lighter traffic congestion, while the smaller road section average speed indicates more serious traffic congestion.

Determining weights of the evaluation factors

Considering the characteristics of traffic flow before and after the morning peak, we adapt different weights of evaluation factors in different periods. To be specific, the weights of each evaluation factor is expressed as $W_a = \{w_1, w_2, w_3\}$ before the morning peak and $W_b = \{w_4, w_5, w_6\}$ after the morning peak.

Performing the single factor fuzzy evaluation

For the i th factor in factor set U , we get the membership r_{ij} of the j th evaluation in evaluation set V through the trapezoidal

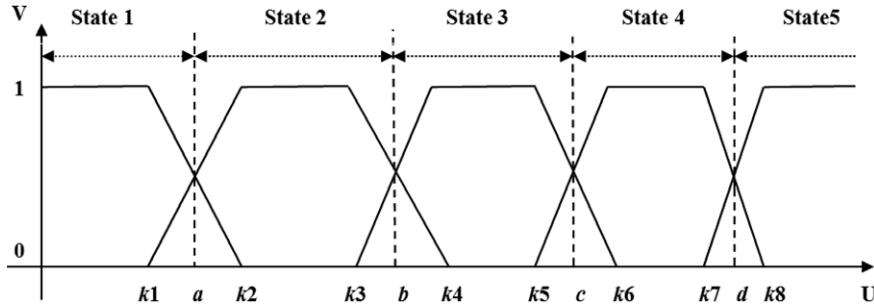


Fig. 1. Membership function graph of traffic congestion estimation.

membership function. The single factor fuzzy evaluation set is expressed as $R_i = \{r_{i1}, r_{i2}, r_{i3}, r_{i4}, r_{i5}\}$ in Eq. (2).

$$R = \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} \end{pmatrix}. \quad (2)$$

Performing the fuzzy comprehensive evaluation

After identifying the weights and performing the single factor fuzzy evaluation, the fuzzy comprehensive evaluation matrix B is calculated using a fuzzy transformation based on Eqs. (3) and (4) as follows:

$$B = W \circ R = W \circ \begin{pmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} \end{pmatrix} = (b_1, b_2, b_3, b_4, b_5) \quad (3)$$

$$b_j = \sum_{i=1}^n w_i \times r_{ij}, \quad i = 1, 2, \dots, 6, j = 1, 2, \dots, 5 \quad (4)$$

where \circ is the fuzzy compositional operation and b_j is the fuzzy comprehensive evaluation index which means the membership of the j th factor of the evaluation object.

Determining the traffic congestion state

Based on the maximum membership principle, the biggest membership b_j is calculated as the final evaluation index b , namely traffic congestion state, as follows.

$$b = \max(b_1, b_2, b_3, b_4, b_5). \quad (5)$$

4. Traffic congestion prediction

In this section, we describe the congestion prediction method, which includes the TFP and the CSFD modules. The TFP module is used to predict the traffic flow parameters and consists of TVP, TSP, and PSO sub modules. The TVP sub module is used for predicting traffic volume, while the TSP sub module is used for predicting average traffic speed. And the PSO sub module is applied to optimize the punish coefficients and the parameters of the multi-kernel functions of SVM in TVP and TSP. Furthermore, the CSFD module converts the predicted traffic flow parameters to citizens' cognitive state with the help of FCE method which we have described in detail in the previous section. Fig. 2 depicts the architecture of our proposed method.

SVM and PSO algorithms are chosen as optimization methods, since SVM has effective nonlinear mapping and generalization abilities and can solve small sample, over learning and local minimum problems, while PSO is a heuristic algorithm that has an advantage in search speed and stability.

4.1. The TFP module

TFP module includes TVP, TSP, and PSO sub modules. In TVP and TSP sub modules, we use LIBSVM library [33] to calculate ε -Support Vector Regression (ε -SVR). The regression function for prediction is calculated using Eq. (6) as follows:

$$y = f(x) = \sum_{i=1}^N (a_i - a_i^*) K(x_i, x) + b \quad (6)$$

where a_i, a_i^* are the Lagrange multipliers related to punish coefficient c . $K(x_i, x)$ is the kernel function, and b is the bias.

In addition, we use linear in Eq. (7), polynomial in Eq. (8), and radial basis function (RBF) in Eq. (9) as the kernel function $K(x_i, x)$ of Eq. (6) respectively.

$$K(x_i, x) = (x_i \times x) \quad (7)$$

$$K(x_i, x) = (\lambda(x_i \times x) + e)^d \quad (8)$$

$$K(x_i, x) = \exp\left(-\frac{\|x_i - x\|^2}{2\sigma^2}\right). \quad (9)$$

Most of the existing congestion prediction methods such as [19–24] only consider one kernel function: linear, polynomial, or RBF. Different from these methods, our proposed TFP module considers multiple kernel functions since different kernel functions have different prediction accuracies and fitting abilities. In addition, according to Eqs. (7)–(9), the polynomial kernel function has more parameters than linear and RBF kernel functions. Considering the computational complexity and model selectivity, we mainly focus on linear and RBF kernel functions.

The TVP sub module

Assuming that the current time is denoted by t , we aim to predict the traffic volume of time $t+1$ at some road sections. The input variables include six parameters $Spe_{t-2}, Spe_{t-1}, Spe_t, Vol_{t-2}, Vol_{t-1}, Vol_t$ and the output variable is Vol_{t+1} . Among these, Spe_i indicates the average speed of time i , and Vol_i indicates the traffic volume of time i . Then, the TVP model is trained considering the punish coefficient c and multiple kernel functions. At last, the TVP model is tested using the real floating car data.

The TSP sub module

Training and testing methods in TSP sub module are similar to those in TVP sub module, except that the input variables in TSP include $Vol_{t-2}, Vol_{t-1}, Vol_t, Spe_{t-2}, Spe_{t-1}, Spe_t$ and the output variable is Spe_{t+1} .

The PSO sub module

In this module, we aim to optimize the punish coefficient c and multiple kernel functions based on kernel parameters of the SVM model in TVP and TSP sub module.

- Firstly, we determine the parameters of PSO algorithm such as maximum evolution number, maximum population number, cross validation number, the range of punish coefficient c ,

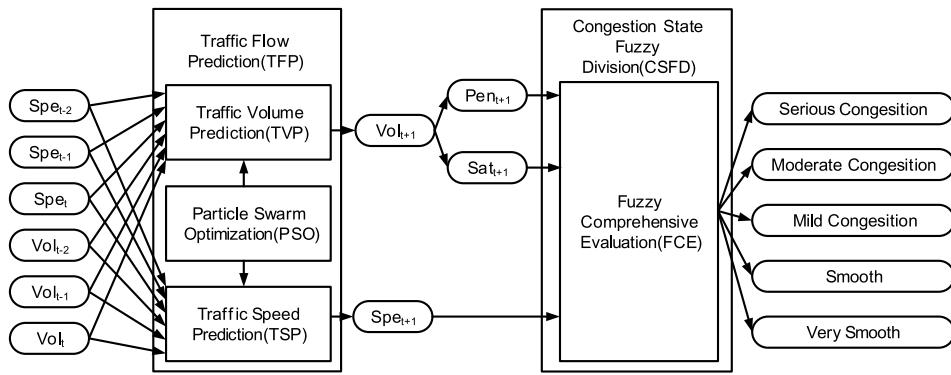


Fig. 2. Architecture of proposed method.

the corresponding kernel parameter kp of the selected kernel function, the position and speed of the particle swarm, etc. In this module, the parameters c and kp are selected as the position of the particle swarm.

- Secondly, the parameters c and kp of the current particle are selected as $pbest$, and the optimal value of all particles as $gbest$. Then, the fitness function value is identified in order to construct and update the position and speed using Eqs. (10) and (11) [34] as follows.

$$v_i(t+1) = \omega \times v_i(t) + c_1 \times r_1 \times (p_{best} - x_i(t)) + c_2 \times r_2 \times (g_{best} - x_i(t)) \quad (10)$$

$$x_i(t+1) = x_i(t) + v_i(t+1). \quad (11)$$

- Thirdly, if $pvalue$ is better than $pbest$, set $pbest = pvalue$.
- Fourthly, if $gvalue$ is better than $gbest$, set $gbest = gvalue$.
- Finally, if the evolution number reaches the maximum value, the optimization stops. Otherwise, the procedure returns to the second step.

After PSO optimizes the parameters of SVM in TVP and TSP sub modules, it transmits the parameters back to TVP and TSP. Then, the traffic volume and average traffic speed are predicted in case different kernel functions are selected in the SVM model. We compare the predicted traffic volume in the case of different kernel functions in the SVM model of TVP and select the best result with minimum error. The TSP sub module follows the same process to obtain the best result. In addition, the corresponding kernel parameters in the SVM improve the prediction accuracy and ensure the real-time performance.

4.2. The CSFD module

We calculate traffic volume (Vol) and average traffic speed (Spe) values using TFP which we have introduced in the preceding section. However, the congestion state fuzzy division cannot be performed directly since the predicted parameters do not characterize the traffic congestion state accurately. Considering the length, the lane number of the road section and Vol , we calculate the traffic density (Den). Then, the road saturation (Sat) is calculated using traffic capacity and Vol . Finally, congestion state division is applied in the FCE method using Sat , Den , and Spe . The detailed process of how FCE method converts these traffic flow parameters to citizens' cognitive congestion state is described in Section 3.4.

5. Experiment and discussion

In this section, we first describe the floating car data. Then, the experiment is conducted to show the performance of our traffic congestion estimation and prediction method. In experiments, we

select a road section from Lian Hua Qiao to Liu Li Qiao at third ring near Beijing west railway station as the research area in Fig. 3, of which exists seriously recurrent traffic congestion.

5.1. The floating car data

The floating car data are the real traffic GPS data collected by 12,000 taxis in Beijing China over a period of one month (November 2012) [35]. The traffic data are recorded once per minute approximately. The format of a GPS data is shown in Fig. 4.

We first preprocess the data in order to eliminate noisy sample points, of which perceived positions are chaotic. In the second step, the sample points with the same vehicle ID are linked to each other according to their time correlate on. Then, we capture the floating car trajectories on the urban road network space. In the next step, map matching is carried out according to the floating car trajectories, the latitude and longitude of the vehicles, and the urban geographic information. Finally, the traffic flow parameters are extracted in five minutes.

The traffic flow parameters include the traffic volume and traffic speed. The traffic volume is equal to the floating car number divided by a floating car detection ratio. The floating car detection ratio equals the total number of floating cars divided by the total number of vehicles on the road. Based on the annual report of Beijing traffic development in 2012 [36], the floating car detection ratio is 19%. The traffic speed is equal to the average speed of floating car in five minutes. The average speed of each floating car trajectory can be acquired by GPS sample points.

5.2. Performance indexes

We consider the performance indexes from the TFP and CSFD modules, which include the traffic flow prediction and congestion state division indexes. These concepts are identified in the rest of this subsection.

Traffic flow prediction indexes

The prediction accuracy indexes include mean absolute error ($maerr$), mean absolute relative error ($marerr$), mean square error ($mserr$) and mean square relative error ($msrerr$) as shown in following equations,

$$maerr = \frac{1}{N} \sum_{t=1}^N |p_{predict}(t) - R_{real}(t)| \quad (12)$$

$$marerr = \frac{1}{N} \sum_{t=1}^N \frac{|p_{predict}(t) - R_{real}(t)|}{R_{real}(t)} \quad (13)$$

$$mserr = \sqrt{\frac{1}{N} \sum_{t=1}^N (p_{predict}(t) - R_{real}(t))^2} \quad (14)$$

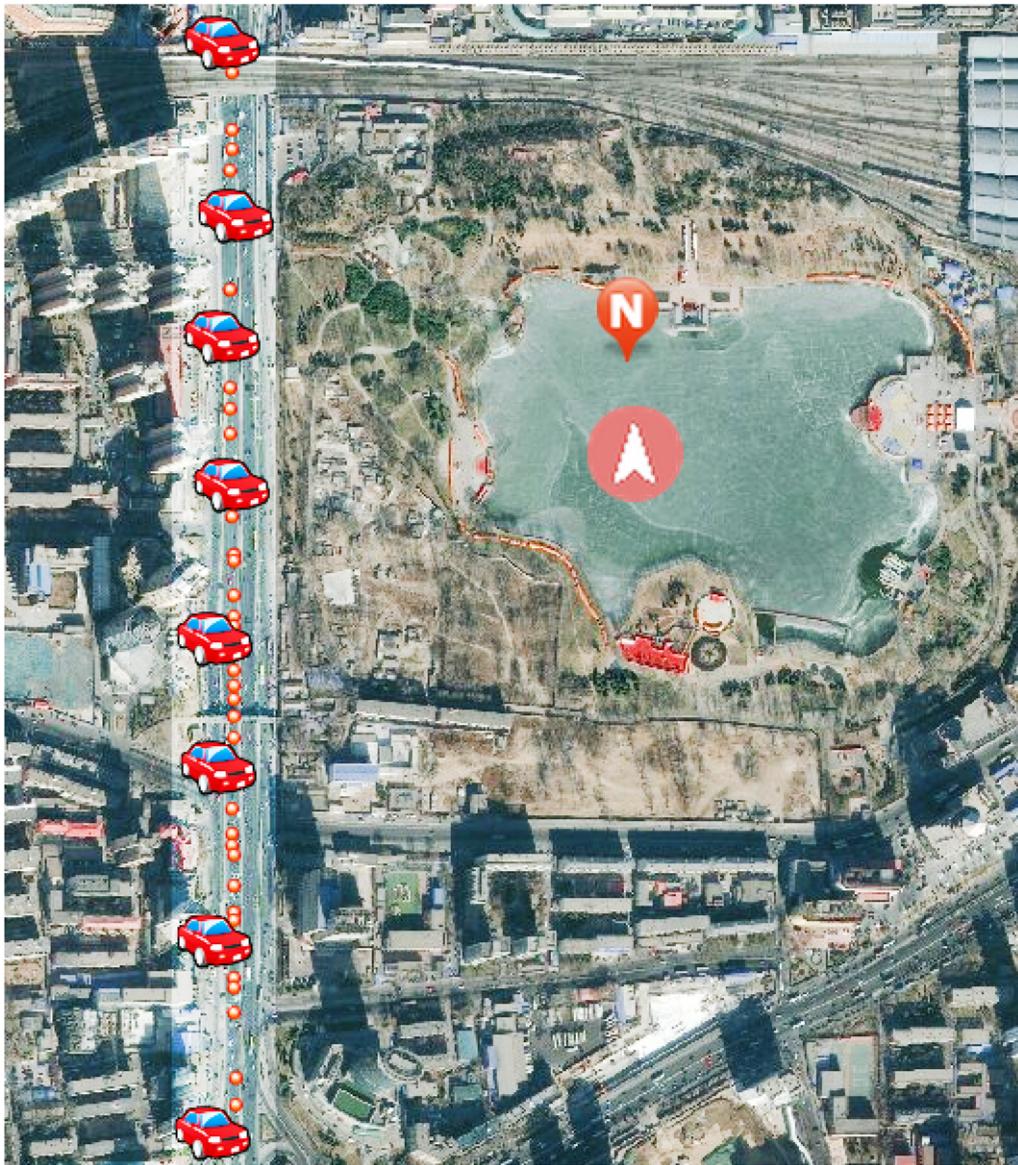


Fig. 3. Experiment area.

$$msrerr = \sqrt{\frac{1}{N} \sum_{t=1}^N \left(\frac{p_{\text{predict}}(t) - R_{\text{real}}(t)}{R_{\text{real}}(t)} \right)^2} \quad (15)$$

where P_{predict} denotes the predicted value and R_{real} denotes the real value.

The real-time performance indexes include the time for training model (tr_time) and traffic flow prediction (pre_time). The *stability* performance indexes are related to the process of the punish coefficient selection and the prediction accuracy.

Congestion state division indexes

The congestion state prediction *accuracy* is given by the formula that the prediction congestion states divide by the real congestion state. The *instantaneity* performance indexes include the time to perform congestion state division. The *stability* performance indexes are also related to the process of the punish coefficient selection and the prediction accuracy.

5.3. FCE congestion estimation

Though the traffic flow parameters are extracted, it can be seen that traffic congestion still cannot be characterized accurately. The

Taxi ID	Trigger	Operating	Time	Longitude	Latitude	Speed	Orientation	Status
431498	1	116.4243011	43	88	1	20121101095636	40.0727348	

Fig. 4. Data format of a GPS data entry.

reason is that different roads have different road spaces and traffic capacities. Considering the length and lane number of road section as well as the traffic volume, the traffic density can be acquired with the help of Google Earth. The experiment area is 1.127 square kilometers with 8 lanes. The road saturation can also be calculated using the traffic capacity and traffic volume. We acquire the traffic capacity from China highway capacity manual, which defines that the maximum traffic capacity of multi-lane highway designed with 80 km/h is 1800 pcu/h/lane. In other words, the maximum traffic capacity of the explored road in this paper is 150 pcu/5 min/lane.

Then, the congestion state division in our proposed FCE method is performed using the road saturation, the traffic density, and the traffic average speed. Accordingly, the weight sets are assigned as $W_a = [0.43, 0.27, 0.3]$ and $W_b = [0.23, 0.17, 0.6]$. Fig. 5 shows

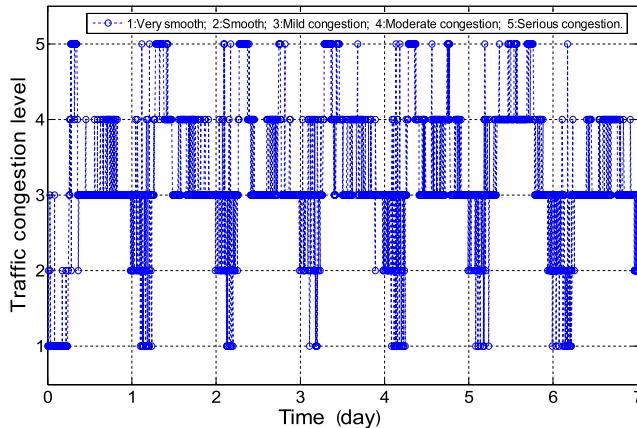


Fig. 5. Traffic congestion estimation during a week.

Table 1

Performance comparison among different kernel functions of proposed method about traffic volume prediction.

Metrics	Linear	Polynomial	RBF
<i>maerr</i>	0.5457	0.6772	0.5319
<i>marerr</i>	22.1943	22.4959	21.8899
<i>mserr</i>	31.1367	31.6596	30.2740
<i>msrerr</i>	1.5327	2.1554	1.3666
<i>tr_time</i>	0.0930 s	0.1120 s	0.1210 s
<i>pre_time</i>	0.0100 s	0.0120 s	0.0340 s

Table 2

Performance comparison among different kernel functions of proposed method about average speed prediction.

Metrics	Linear	Polynomial	RBF
<i>maerr</i>	0.1576	0.1737	0.1604
<i>marerr</i>	4.8738	5.0802	4.8986
<i>mserr</i>	6.7124	6.8462	6.7971
<i>msrerr</i>	0.2354	0.2642	0.0589
<i>tr_time</i>	0.0920 s	0.1130 s	0.1190 s
<i>pre_time</i>	0.0090 s	0.0110 s	0.0330 s

the traffic congestion estimation using our FCE method during a week (from Monday to Sunday).

5.4. Traffic flow prediction

Through estimating the traffic congestion, we aim to induce travel cost and prevent congestion from further spreading. To this target, a congestion prediction method is applied, which includes the traffic flow prediction and the congestion state fuzzy division. Thus, we do experiments from these two aspects.

The traffic volume and traffic speed prediction results are explored in our experiments. We conduct the experiments with different kernel functions and optimization techniques.

Prediction with different kernel functions

We compare the results of prediction with three different kernel functions: linear, polynomial and RBF. In order to evaluate the performance of these kernel functions, we do not apply any optimization techniques in this part. The default parameters are shown as follows: insensitive parameter ε is 0.01, punish coefficient c is 0.8, RBF parameter σ is $\sqrt{2}/2$, polynomial kernel function parameter λ is 0.17 and e is 0. We compare the three kernel functions in terms of six evaluation metrics: *maerr*, *marerr*, *mserr*, *msrerr*, *tr_time* and *pre_time*. The experiment results of traffic volume prediction are shown in Fig. 6 and Table 1. Similarly, the traffic speed prediction results are shown in Fig. 7 and Table 2.

We can reach that RBF kernel performs better than linear and polynomial kernels in traffic volume prediction according to

Table 3

Traffic volume prediction performance of kernel functions in different periods.

Metrics	Linear	Polynomial	RBF
<i>marerr</i>	22.1943	22.4959	21.8899
<i>marerr_befmor</i>	14.0327	16.6359	13.9580
<i>marerr_mor</i>	41.9103	40.4290	39.8513
<i>marerr_betmoev</i>	28.6937	27.6906	27.5881
<i>marerr_eve</i>	22.7470	23.0084	26.2030
<i>marerr_afteve</i>	14.9937	14.8832	14.8502

Table 4

Average speed prediction performance of kernel functions in different periods.

Metrics	Linear	Polynomial	RBF
<i>marerr</i>	4.8738	5.0802	4.8986
<i>marerr_befmor</i>	7.7589	7.7101	7.7928
<i>marerr_mor</i>	3.3871	5.4953	3.6888
<i>marerr_betmoev</i>	4.0058	4.0662	3.8725
<i>marerr_eve</i>	2.7675	2.9984	3.1744
<i>marerr_afteve</i>	3.6400	3.6637	3.6409

Table 1. From Table 2, linear kernel function outperforms the other two kernels on average speed prediction in terms of prediction accuracy and real-time index.

In order to analyze the results of traffic flow prediction before and after morning and evening peak accurately, we divide one day into five periods and the corresponding prediction indexes are the mean absolute relative error before morning peak (*marerr_befmor*), morning peak (*marerr_mor*), between morning and evening peak (*marerr_betmoev*), evening peak (*marerr_eve*), and after evening peak (*marerr_afteve*). The prediction results of traffic volume in different periods are shown in Table 3, and the traffic speed prediction results are shown in Table 4.

Fig. 6 and Table 3 demonstrate that traffic volume prediction performance of three kernel functions fluctuates from morning peak to evening peak, because the research area is next to Beijing Railway Station and the traffic conditions are complicated in the daytime. With respect to average speed prediction, performance fluctuates before morning peak from Fig. 7 and Table 4 due to the less number of travelers and the randomness of driving behavior. In general, the three kernel functions perform better in the smooth parts of curves. Furthermore, linear and RBF kernels perform better than polynomial kernel in terms of traffic volume and average speed predictions. Hence, we mainly focus on the linear and RBF kernel functions in the following experiments.

Prediction with different optimization techniques

We compare the results of prediction with different optimization techniques. Specifically, PSO optimization method is used in the RBF kernel function (PSO-R), as well as the linear kernel function (PSO-L) of SVM. In addition, GA optimization method is used in the RBF kernel function (GA-R), as well as the linear kernel function (GA-L) of SVM.

In order to evaluate the performance of PSO and GA optimization methods, their common parameters are set as follows: the maximum evolution number is 100, the maximum population number is 20, the cross validation number of SVM is 3, punish coefficient c is [0.1, 100], RBF kernel parameter σ is [0.01, 1000]. Moreover, the crossover probability and mutation probability of GA are set to 0.4 and 0.01, respectively.

We compare PSO-R, PSO-L, GA-R, and GA-L methods in terms of following evaluation metrics, *maerr*, *marerr*, *tr_time*, *pre_time* and *stability*, respectively. The evaluation results for the traffic volume prediction are shown in Fig. 8 and Table 5. Similarly, the traffic speed prediction results are shown in Fig. 9 and Table 6.

From these experimental results, we can see that PSO optimization method is generally better than GA method in terms of prediction accuracy (*maerr* and *marerr*). Furthermore, GA consumes too

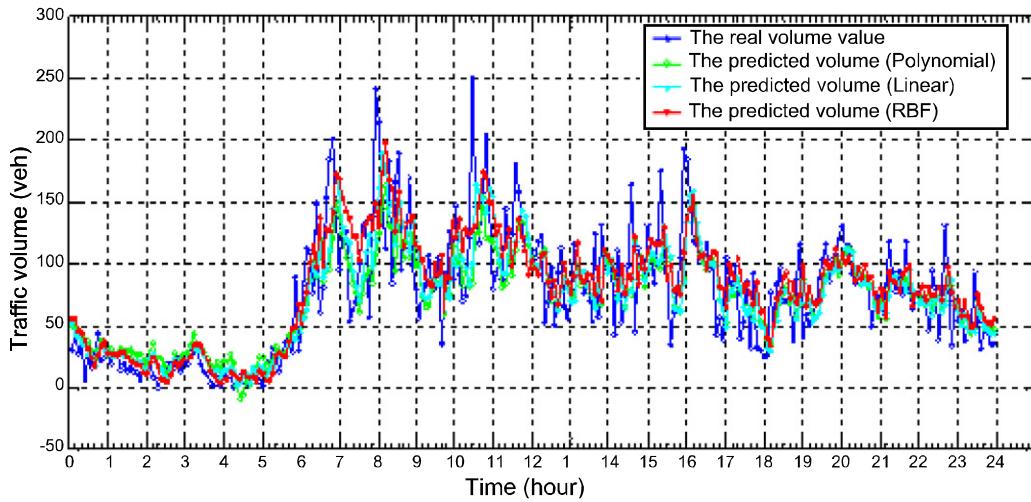


Fig. 6. Comparison among the real traffic volume and prediction results of proposed method with different kernel functions.

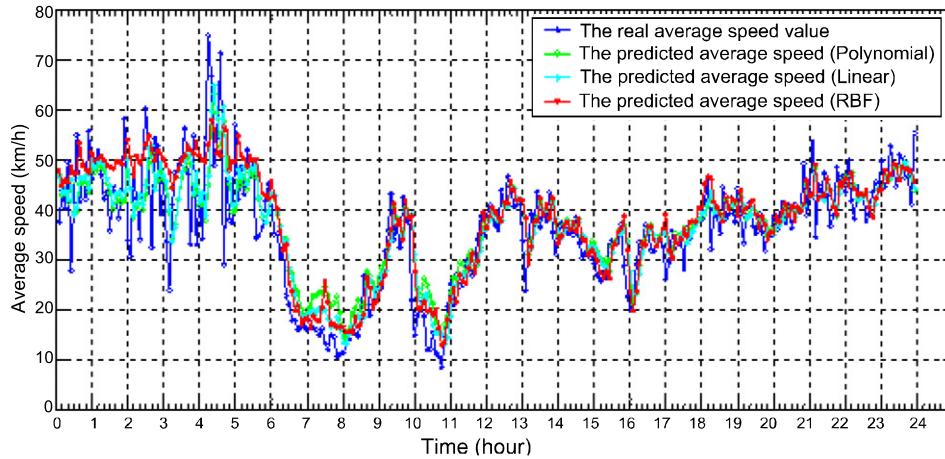


Fig. 7. Comparison among the real average speed and prediction results of proposed method with different kernel functions.

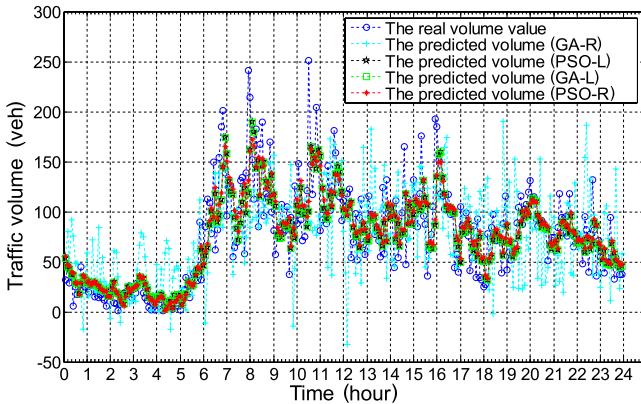


Fig. 8. Comparison among the real traffic volume and prediction results of proposed method with different optimization techniques.

much time in the process of parameter optimization and the operation of selection, crossover and mutation, which results in a disadvantage to the real-time performance. Besides, PSO-R has the best comprehensive performance from the evaluation results. We can conclude that different kernel functions in SVM have different prediction accuracies and fitting abilities, hence, we take advantages of the SVM multi-kernel functions to improve the performance of our proposed method.

Table 5

Performance comparison among different optimization techniques of proposed method about traffic volume prediction.

Metrics	GA_R	PSO_L	GA_L	PSO_R
maerr	37.9212	22.2051	22.2027	21.5979
marerr	1.0464	0.5472	0.5476	0.5370
tr_time	1.0770 s	1.7780 s	1.5140 s	0.2190
pre_time	0.0310 s	0.0160 s	0.0150 s	0.0460 s
c, σ	0.18, 8.2	100, 0.1	91.84	100
Stability	No	Yes	No	Yes

Table 6

Performance comparison among different optimization techniques of proposed method about average speed prediction.

Metrics	GA_R	PSO_L	GA_L	PSO_R
maerr	9.92	4.87	4.86	4.74
marerr	0.36	0.15	0.15	0.14
tr_time	2.09 s	1.04 s	2.02 s	0.14 s
pre_time	0.046 s	0.016 s	0.016 s	0.031 s
c, σ	91.4, 35.2	40.99	100	100, 0.1
Stability	No	No	Yes	Yes

5.5. Congestion state prediction

The performance of congestion state fuzzy division is explored in this subsection. We conduct the experiments without and with optimization techniques.

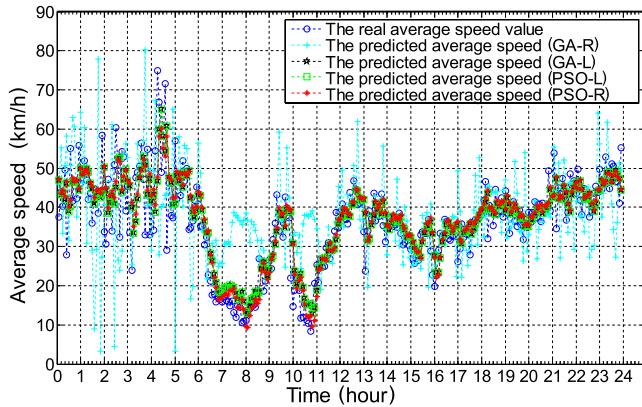


Fig. 9. Comparison among the real average speed and prediction results of proposed method with different optimization techniques.

Table 7
Congestion state prediction performance of kernel functions in different periods.

Metrics	Linear	RBF
accuracy	74.22%	74.22%
acc_befmor	66.67%	67.86%
acc_mor	76.92%	69.23%
acc_betmoev	69.79%	72.92%
acc_eve	83.33%	79.19%
acc_afteve	84.75%	83.05%

Prediction without optimization techniques

We compare the performance results of linear and RBF kernel functions in this part. In order to evaluate the performance of these kernel functions, we do not apply any optimization techniques. Fig. 10 shows the congestion state prediction based on linear kernel, and Fig. 11 depicts the results based on RBF kernel.

Similar to the previous subsection, we divide one day into five periods to analyze the results more accurately. The corresponding indexes are the accuracy before morning peak (*acc_befmor*), morning peak (*acc_mor*), between morning and evening peak (*acc_betmoev*), evening peak (*acc_eve*) and after evening peak (*acc_afteve*). The prediction results of congestion state in different periods are shown in Table 7.

From Figs. 10, 11 and Table 7, we can see that linear and RBF kernels have the same overall accuracies, which can properly fit the real congestion state. Moreover, linear kernel outperforms RBF kernel in the periods of morning peak, evening peak and after evening peak. It demonstrates that each kernel has its own

Table 8

Performance comparison among different optimization techniques of proposed method about traffic congestion prediction.

Metrics	GA-SVM-RL	PSO-SVM-R	PSO-SVM-RL
accuracy	67.25%	74.22%	74.91%
acc_befmor	53.57%	70.24%	69.05%
acc_mor	75.0%	87.5%	87.5%
acc_betmoev	62.5%	67.71%	68.75%
acc_eve	83.33%	75%	83.33%
acc_afteve	84.75%	84.75%	84.75%
instantaneity	6.094 s	0.701 s	4.695 s
stability	No	Yes	Yes

advantages in congestion state prediction, hence, we use the multi-kernel function to improve prediction performance.

Prediction with optimization techniques

In this part, we present our experiment results of traffic congestion prediction with optimization techniques in multi-kernel function. We use PSO and GA optimization techniques in linear and RBF kernel functions of SVM, and then form three schemes: PSO optimizing RBF kernel (PSO-SVM-R), PSO optimizing RBF and linear kernels (PSO-SVM-RL), and GA optimizing RBF and linear kernels (GA-SVM-RL). The results for congestion state prediction are shown in Figs. 12–14. We compare PSO-SVM-R, PSO-SVM-RL and GA-SVM-RL methods in terms of *accuracy*, *instantaneity* and *stability* metrics, and we also present the accuracies of different periods as shown in Table 8.

The results show that PSO-SVM-RL method has better prediction accuracy than the PSO-SVM-R method. Especially, PSO-SVM-RL achieves 83.33% accuracy in evening peak while the accuracy of PSO-SVM-R is 75%. We can conclude that the PSO optimization method has better performance on the multi-kernel function of SVM rather than the single kernel function. Besides, both the PSO-SVM-RL and PSO-SVM-R methods outperform the GA-RL method in all the performance metrics, which means that the PSO optimization method has better performance on the multi-kernel and single kernel function of SVM in comparison to the GA optimization method. Finally, the PSO-SVM-RL has the best comprehensive performance in our experiments, which can fit the accuracy, real-time and stability requirements.

6. Conclusion

In this paper, a new fuzzy comprehensive evaluation (FCE) method was used to estimate and predict the traffic congestion in large scale urban areas. In this method, floating car trajectory data was used to map the floating car sample points with the same

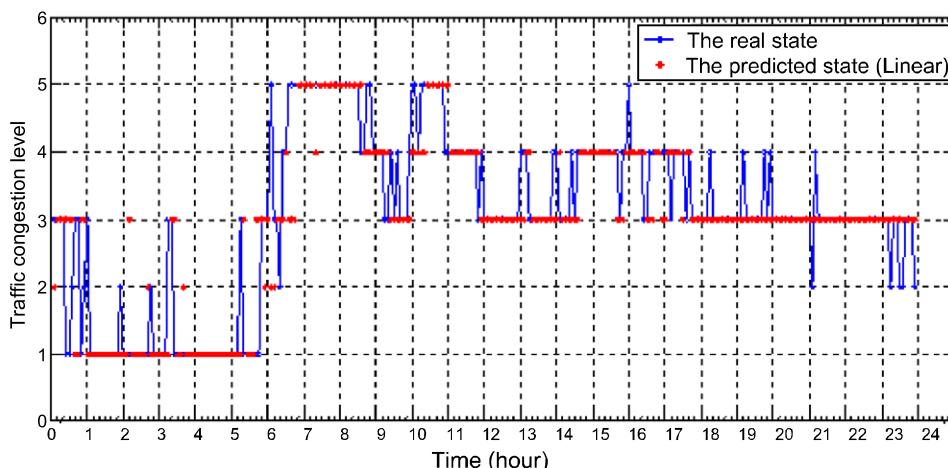


Fig. 10. Comparison between the real congestion state and prediction result of proposed method with linear kernel.

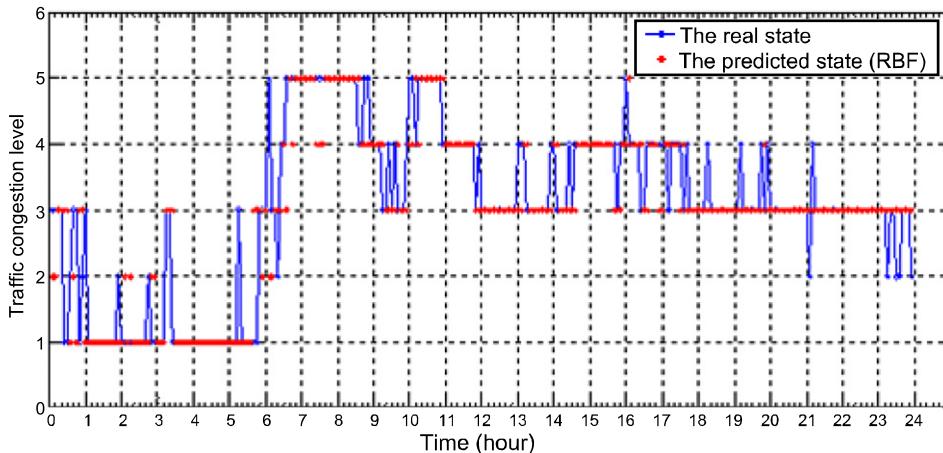


Fig. 11. Comparison between the real congestion state and prediction result of proposed method with RBF kernel.

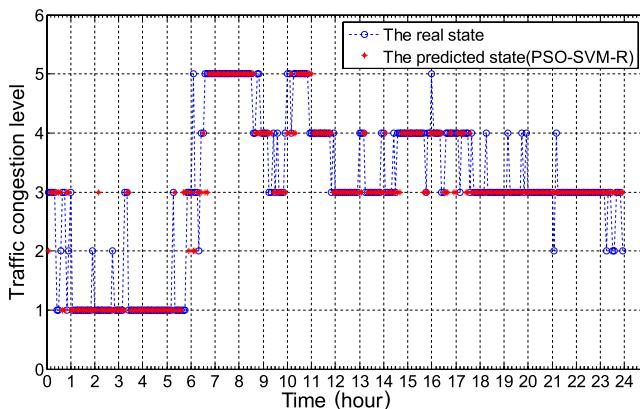


Fig. 12. Comparison between the real congestion state and prediction result of proposed method with PSO-SVM-R.

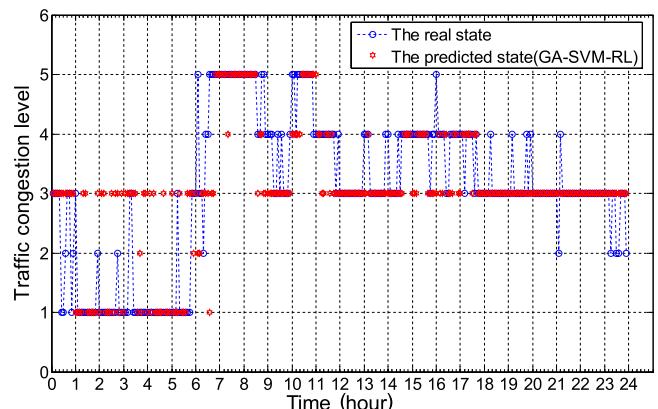


Fig. 14. Comparison between the real congestion state and prediction result of proposed method with GA-SVM-RL.

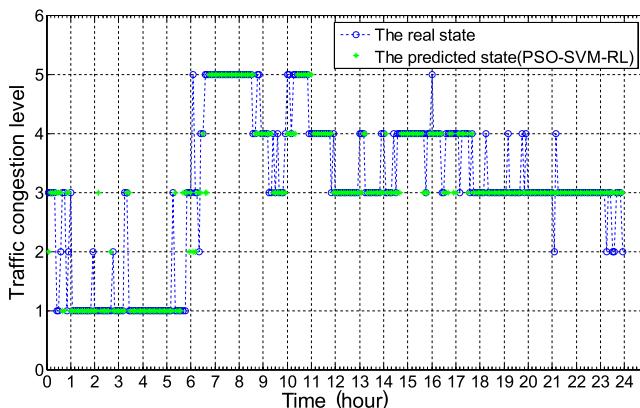


Fig. 13. Comparison between the real congestion state and prediction result of proposed method with PSO-SVM-RL.

vehicle ID to the car trajectories according to their time correlation. Then, the traffic flow parameters were extracted to estimate the traffic flow. In order to improve the traffic estimation performance metrics in terms of the accuracy, instantaneity and stability, we applied FCE method in which the weights of multi-indexes were adapted according to the traffic flows. In order to predict traffic congestion, the PSO optimization method was used to optimize the punish coefficients and multiple kernel functions' parameters of SVM in the process of traffic flow parameters prediction.

In the future, we plan to extend our congestion estimation and prediction methods in a way that the weights of multi-indexes can be adapted according to the traffic flows. Our simulation results revealed that the predicted traffic flow parameters have a certain delay compared to the real values. Therefore, we plan to improve the prediction instantaneity in our future work.

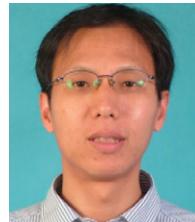
Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grant Nos. 61203165 and 61174174, the Foundation of Key Laboratory of System Control and Information Processing No. DUT15YQ112, Ministry of Education, P.R. China No. SCIP2012001, and the Fundamental Research Funds for Central Universities No. DUT15QY06.

References

- [1] Y. Zheng, L. Capra, O. Wolfson, et al., Urban computing: concepts, methodologies, and applications, *ACM Trans. Intell. Syst. Technol.* 5 (3) (2014) 38.
- [2] M.B. Younes, A. Boukerche, A performance evaluation of an efficient traffic congestion detection protocol (ECODE) for intelligent transportation systems, *Ad Hoc Netw.* 24 (2015) 317–336.
- [3] Y. Zheng, Trajectory data mining: An overview, *ACM Trans. Intell. Syst. Technol.* (2015) <http://dx.doi.org/10.1145/2743025>.
- [4] Q.Y. Yang, J.Z. Wang, X.M. Song, et al., Urban traffic congestion prediction using floating car trajectory data, in: *Algorithms and Architectures for Parallel Processing*, Springer International Publishing, 2015, pp. 18–30.
- [5] W. Pattara-Atikom, P. Pongpaibool, S. Thajchayapong, Estimating road traffic congestion using vehicle velocity, in: *Proceedings of 6th International Conference on ITS Telecommunications*, Chengdu, China, 2006, pp. 1001–1004.

- [6] J. Yoon, B. Noble, M. Liu, Surface street traffic estimation, in: Proceedings of the 5th International ACM Conference on Mobile Systems, Applications and Services, New York, USA, 2007, pp. 220–232.
- [7] Y. Chen, L. Gao, Z. Li, et al., A new method for urban traffic state estimation based on vehicle tracking algorithm, in: Proceedings of Intelligent Transportation Systems Conference, Seattle, WA, 2007, pp. 1097–1101.
- [8] W. Shi, Q.J. Kong, Y. Liu, A GPS/GIS integrated system for urban traffic flow analysis, in: Proceedings of 11th International IEEE Conference on Intelligent Transportation Systems, Beijing, China, 2008, pp. 844–849.
- [9] Q.J. Kong, Q. Zhao, C. Wei, et al., Efficient traffic state estimation for large-scale urban road networks, *IEEE Trans. Intell. Transp. Syst.* 14 (1) (2013) 398–407.
- [10] Q.J. Kong, Z. Li, Y. Chen, et al., An approach to urban traffic state estimation by fusing multisource information, *IEEE Trans. Intell. Transp. Syst.* 10 (3) (2009) 499–511.
- [11] J.D. Zhang, J. Xu, S.S. Liao, Aggregating and sampling methods for processing GPS data streams for traffic state estimation, *IEEE Trans. Intell. Transp. Syst.* 14 (4) (2013) 1629–1641.
- [12] L. Li, X. Chen, L. Zhang, Multimodel ensemble for freeway traffic state estimations, *IEEE Trans. Intell. Transp. Syst.* 15 (3) (2014) 1323–1336.
- [13] Y. Feng, J. Hourd, G.A. Davis, Probe vehicle based real-time traffic monitoring on urban roadways, *Transp. Res. C* 40 (2014) 160–178.
- [14] J. Lu, L. Cao, Congestion evaluation from traffic flow information based on fuzzy logic, in: Proceedings of the 6th International IEEE Conference on Intelligent Transportation Systems, vol. 1, Shanghai, China, 2003, pp. 50–53.
- [15] P. Pongpaibool, P. Tangamchit, K. Noodwong, Evaluation of road traffic congestion using fuzzy techniques, in: Proceeding of 2007 IEEE Region 10 Conference, Taipei, Taiwan, 2007, pp. 1–4.
- [16] Y. Chen, Y. Liu, A new method for GPS-based urban vehicle tracking using pareto frontier and fuzzy comprehensive judgment, in: Proceedings of 11th International IEEE Conference on Geoscience and Remote Sensing Symposium, Barcelona, Spain, 2007, pp. 683–686.
- [17] H. Shankar, P.L.N. Raju, K.R.M. Rao, Multi model criteria for the estimation of road traffic congestion from traffic flow information based on Fuzzy logic, *J. Transp. Technol.* 2 (2012) 50.
- [18] H. Su, S. Yu, Hybrid GA based online support vector machine model for short-term traffic flow forecasting, in: *Advanced Parallel Processing Technologies*, Springer, Berlin, Heidelberg, 2007, pp. 743–752.
- [19] C.T. Cao, J.M. Xu, Improved particle swarm optimized SVM for short-term traffic flow predication, in: Proceedings of the 26th Chinese Control Conference, Zhangjiajie, Hunan, China, 2007, pp. 6–9.
- [20] Y. Xu, B. Wang, Q. Kong, et al., Spatio-temporal variable selection based support vector regression for urban traffic flow prediction, in: Proceeding of the 93rd Annual Meeting of the Transportation Research Board, Washington, DC, 2014, pp. 14–1994.
- [21] W.C. Hong, Y. Dong, F. Zheng, et al., Hybrid evolutionary algorithms in a SVR traffic flow forecasting model, *Appl. Math. Comput.* 217 (15) (2011) 6733–6747.
- [22] M.W. Li, W.C. Hong, H.G. Kang, Urban traffic flow forecasting using Gauss-SVR with cat mapping, cloud model and PSO hybrid algorithm, *Neurocomputing* 99 (2013) 230–240.
- [23] J. Wang, Q. Shi, Short-term traffic speed forecasting hybrid model based on Chaos-Wavelet analysis-support vector machine theory, *Transp. Res. C* 27 (2013) 219–232.
- [24] F. Wang, G. Tan, C. Deng, et al., Real-time traffic flow forecasting model and parameter selection based on ε -SVR, in: Proceedings of the 7th IEEE World Congress on Intelligent Control and Automation, Chongqing, China, 2008, pp. 2870–2875.
- [25] J. Gong, L. Qi, M.Y. Liu, et al., Forecasting urban traffic flow by SVR, in: Proceedings of the 25th Chinese IEEE Control and Decision Conference, Guiyang, China, 2013, pp. 981–984.
- [26] H. Chen, H.A. Rakha, S. Sadek, Real-time freeway traffic state prediction: A particle filter approach, in: Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems, ITSC, Washington, DC, USA, 2011, pp. 626–631.
- [27] S. Dunne, B. Ghosh, Regime-based short-term multivariate traffic condition forecasting algorithm, *J. Transp. Eng.* 138 (4) (2011) 455–466.
- [28] W. Min, L. Wynter, Real-time road traffic prediction with spatio-temporal correlations, *Transp. Res. C* 19 (4) (2011) 606–616.
- [29] X. Zhang, E. Onieva, A. Perallos, et al., Hierarchical fuzzy rule-based system optimized with genetic algorithms for short term traffic congestion prediction, *Transp. Res. C* 43 (1) (2014) 127–142.
- [30] R. Herring, A. Hofleitner, S. Amin, et al., Using mobile phones to forecast arterial traffic through statistical learning, in: Proceedings of the 89th Transportation Research Board Annual Meeting, Washington DC, USA, 2010, pp. 10–2493.
- [31] P.S. Castro, D. Zhang, S. Li, Urban traffic modelling and prediction using large scale taxi GPS traces, in: *Pervasive Computing*, Springer, Berlin, Heidelberg, 2012, pp. 57–72.
- [32] X. Zhou, W. Wang, L. Yu, Traffic flow analysis and prediction based on GPS data of floating cars, in: Proceedings of the 2012 International Conference on Information Technology and Software Engineering, Springer, Berlin, Heidelberg, 2013, pp. 497–508.
- [33] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, *ACM Trans. Intell. Syst. Technol.* 2 (3) (2011) 1:27.
- [34] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of the IEEE International Conference on Neural Networks, vol. 4 (2), 1995, pp. 1942–1948.
- [35] <http://www.datatang.com/data/44502>.
- [36] Beijing traffic development research center. The transportation development annual report at 2012 of Beijing city, 2012. <http://www.bjtrc.org.cn/JGS.aspx?id=5.2&Menu=GZCG>.



Xiangjie Kong received the Ph.D. degree from Zhejiang University, Hangzhou, China in 2009. Currently, he is an associate professor in School of software, Dalian University of technology, Dalian, China. He has served as Workshop Chair or PC Member of a number of conferences. He has published over 30 scientific papers in international journals and conferences (with 10+ indexed by ISI SCIE). His research interests include big traffic data, social computing, and cyber-physical systems. He is a Member of IEEE.



Zenzhen Xu received the Ph.D. degree from Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China in 2009. She is currently an Assistant Professor at School of Software in Dalian University of Technology, China. Her research interests include intelligent transportation systems, modeling, scheduling, simulation and optimization of complex system.



Guojiang Shen received the B.Sc. degree in Control Theory and Control Engineering and the Ph.D. degree in Control Science and Engineering from Zhejiang University, Hangzhou, China, in 1999 and 2004, respectively. He is currently a Professor in the College of Computer Science and Technology, Zhejiang University of Technology. His current research interests include intelligent control theory and application, advanced control technology and application, and urban road traffic modeling and control technology.



Jinzhong Wang received the B.S. degree in Computer Education from Anshan Normal University, Anshan, China, in 2002 and the M.S. degree in Computer Application Technology from Liaoning University, Shenyang, China, in 2005. He is currently working toward the Ph.D. degree with the School of Software, Dalian University of Technology, Dalian, China.



Qiuyuan Yang received the Bachelor's degree in Software Engineering from Dalian University of Technology, Dalian, China, in 2013. She is currently working toward the Master's degree from the Mobile and Social Computing Laboratory, School of Software, Dalian University of Technology. Her research interests include mobile social networks, big traffic data and socially aware networking.



Benshi Zhang received the M.S. degree in Software Engineering from Dalian University of Technology, Dalian, 116620, China, 2014. Since 2012, he has been a member of Mobile and Social Computing Laboratory, School of Software, Dalian University of Technology. His research interests include intelligent transportation systems, particularly traffic congestion estimation and prediction by taxi trajectory data.