Wireless Networks (2019) 25:2143–2156
https://doi.org/10.1007/s11276-018-1803-3

Vehicle trajectory prediction algorithm in vehicular network

Lei-lei Wang[1,2] · Zhi-gang Chen[1,2] · Jia Wu[1,2]

Published online: 25 July 2018

Abstract

Vehicular ad hoc network has become an important component of the intelligent transportation system, what's more, the vehicle trajectory prediction has gradually become one of the hotter issues in this research. Vehicle trajectory prediction cannot only provide accurate location services, but also can monitor traffic conditions in advance, and then recommend the best route for the vehicle. For this purpose, this research established a new method for vehicle trajectory prediction (TPVN), which is mainly applied to predict the vehicle trajectory in the short term. Based on the regularity of vehicle movement, the algorithm is helpful to predict the vehicle trajectory so as to estimate the position of the vehicle motion probability. To improve the prediction accuracy, the motion patterns are divided into two types: simple pattern and complex pattern. The advantage of the TPVN algorithm is that the calculation result not only predicts the movement behavior of vehicles in different motion patterns but also the probability distribution of all possible trajectories of the vehicle in the future. Simulation on a large number of true trajectory datasets shows that the performance of TPVN outperforms than other classical algorithms.

Keywords VANET · Vehicle trajectory prediction · Regularity of motion · Motion pattern

1 Introduction

Vehicular ad hoc network (VANET) is created by applying the principle of the mobile ad hoc network (MANET) [1, 2], the spontaneous creation of wireless networks for data exchange, to the vehicle domain. This type of network has its own unique characteristics: high mobility, insufficient power supply, uncertainty distribution, frequent network disconnections, and frequent topology changes differentiate them from MANET [3].

✉ Zhi-gang Chen
czg@csu.edu.cn

✉ Jia Wu
jiawu5110@163.com

Lei-lei Wang
wll_1234@163.com

1 School of Software, Central South University, Changsha City, Hunan Province 410075, China

2 "Mobile Health" Ministry of Education-China Mobile Joint Laboratory, Changsha 410083, China

In the wake of the rapid development of mobile communication technology and global location technology, advanced driver assistance systems (ADAS), automated vehicles (AVs), vehicle collision warning system (CWS) and location-based service (LBS) have a wide range of research interests owing to their great potential to be used in more efficient, cleaner and safer transportation systems [4]. Meanwhile, the vehicle trajectory prediction has slowly become one of the hotter issues in this research.

At present, VANET has become an important component framework of the intelligent transportation system (ITS) to provide navigation, road safety, and other roadside services. The study by IEEE Communications Association shows that more than 90% traffic emergencies are caused by human errors. When the traffic accident happened, there is only a limited time for the driver to react, so there still have a risk of collisions. With the development of the vehicle's future trajectory prediction technology, drivers can be aware of nearby vehicles and traffic conditions timely, so that they can have enough time to make appropriate decisions to avoid potential collisions [5]. Accurate and reliable future vehicle trajectory prediction algorithms can detect potential collisions in advance and reduce the risk of vehicle collisions. In real life, vehicle movement is controlled by human behavior, rather than the irregular random motion in the vehicular network [6]. The regularity of the vehicle movement be analyzed to get the correct trajectory prediction of the vehicle in the future, which has a very important influence on the rapid data transmission, traffic management, and human daily life, and it also has great research value and wide application field.

The regularity of vehicle motion enables us to predict the future trajectory of the vehicle with a certain probability [7]. Vehicle trajectory prediction, according to geographical position, can be divided into the following two categories: short-distance prediction and long-distance prediction. The long-distance prediction is to predict the whole path of the vehicle when the initial point and destination point of the vehicle is given. Since the relative required distance of the long-distance prediction is too far, the prediction algorithm is usually very difficult to achieve high accuracy. Short-distance prediction is generally used to predict the trajectory [8] of the current vehicle at the next moment. According to the current vehicle movement state and historical travel trajectory record, short-distance prediction can achieve higher accuracy. This paper mainly related to the prediction of the vehicle trajectory at the next moment in short distance.

However, an issue to be addressed is the moving trajectory and the unique position information of the moving vehicle may not be obtained during the time of send two positioning information. Technically, the increasingly complex and highly uncertain traffic environment is an important reason [9]. For this topic, the vehicle should have the ability to comprehend the traffic environment and to formulate trajectory planning [10] and tracking [11]. As we all know, in real life, the movement of vehicles often receives the restriction of human behavior and traffic rules. Therefore, the movement of vehicles has certain regularity. By analyzing the movement regularity of the vehicle can help to get additional vehicle movement information to predict the vehicle trajectories in complex environments.

To solve these problems, this research presents a new vehicle trajectory prediction (TPVN) algorithm. This approach is based on the regularity of vehicle movement to establish the trajectory probability of different vehicle motion patterns. The motion patterns are divided into two types: single motion pattern and complex motion pattern, which can obtain additional vehicle information to ensure the accuracy of trajectory prediction.

The main contribution of this article lies in the following aspects:

1. Processing and analyzing historical trajectory data, by analyzing the regularity of vehicle movement, the vehicle motions pattern are divided into two types: simple pattern and complex pattern. It can help to get additional vehicle movement information to predict the vehicle trajectories in complex environments.
2. Using Gaussian mixed model (GMM) to model the trajectory probability of different motion patterns, and using the maximum likelihood estimation (EM) algorithm to obtain the model parameters, so that the probability of trajectory probability model based on historical trajectory data model is maximized. GMM can adapt to the changes in the environment with little limitation.
3. To avoid the drawbacks of the trajectory analysis method in discrete states, the Gaussian process regression (GPR) is used to estimate all probabilities of possible trajectories of the vehicles in the future. And for the prediction of the moving object trajectory, it can accurately measure the prediction error according to the probability model.
4. In accordance with the tool *MATLAP2016a* to simulate in the large number of data and to evaluate the performance of the TPVN algorithm, the prediction time is reduced, and then the prediction accuracy is improved in the simulation.

The remained of this paper is structured as follows. In Sect. 2, we describe and analyze the related works. The definitions and computing methods for vehicle trajectory prediction are presented in Sect. 3. Simulation results and performance analysis are presented in Sect. 4. The last section concludes this paper.

2 Related work

At present, research on vehicular network focuses on trajectory prediction. Existing trajectory prediction algorithms have been improved in various fields. Some methods applied in trajectory prediction are as follows.

Xie et al. [4] proposed the interactive multiple model trajectory prediction (IMMTP), a method that combines physics-based and maneuver-based methods, which using naturalistic driving data to apply and analyze the lane-change. The physics-based trajectory prediction approach can guarantee the prediction accuracy in the short term and the maneuver-based trajectory prediction approach can estimate the future trajectory for a long-term insight. In IMMTP, in order to get a more prediction accuracy in the complex network environments, the vehicle-to-vehicle (V2V) communication devices be used to help the vehicle to achieve the additional information. The time-to-critical-collision-probability (TTCCP) [12] presents an integrated Bayesian maneuver method for trajectory prediction and critical evaluation, not limited to specific driving conditions. The distribution of advanced driving actions is inferred by Bayesian inference for each vehicle in the traffic scene. To achieve this purpose, the domain is modeled via the Bayesian network with both diagnostic evidence and causality maneuver, allowing the test of irrational driving habits and seamless applications are from a high degree of structure to unstructured environments. The maneuver-based probabilistic trajectory prediction model is used to predict the configuration of each vehicle. The random elements in the designed model take into account the uncertainty of future driver's driving performance.

Liu et al. [13] proposes a distributed position estimation algorithm to improve the prediction accuracy by using the distance between cooperating vehicles. This algorithm performs better only when the signal strength is good. However, due to trees, tunnels, buildings, environmental losses, GPS receivers cannot receive signals in the city, resulting in location prediction affected. The approach presented in [14], proposed a location prediction model without GPS. Using the communication devices vehicle-to-Internet (V2I) and dead reckoning to predict of the position, its accuracy is not suitable for many keys VANET applications because it reaches the position error for the prediction of a minimum of 8.79 m. The paper [15] presents an approach that by focusing on the local approximation to spread the vehicle's current known error covariance matrix. This allows evaluating the collision risk between the ego vehicle and the target object. However, because the research of vehicle trajectory is still in the initial stage, the current study mostly trajectory of the vehicle is modeled and analyzed to achieve monitoring and identification purposes, or to predict the vehicle arrival time, destination and motion etc.

How to obtain the high accuracy trajectory prediction of the moving vehicle is such a difficult issues that need to be solved. There have been some research results, such as clustering, anomaly detection of moving targets, positioning and movement trajectory prediction, among which driving vehicle trajectory prediction technology is continuously improving, but as a result of not mature theory and technology, many of the models can't adapt to the needs of the mobile vehicle trajectory prediction [16–19]. Due to the regularity of human activities, traffic conditions have a very strong temporal and spatial correlation. The popularization of roadside equipment and GPS equipment facilitates the collection of steering information and can define the dynamic relationship between roads by steering information as a clustering basis, so as to gather the most influential roads among each other into one group.

Current research efforts focus on modeling the traffic flow [20] and then getting traffic data for analysis. However, it is difficult to find a deterministic traffic model, and the existing models all make trade-offs between prediction accuracy and scene limits. Noting that one road blockage may cause increased traffic pressure on the surrounding roads, some studies began to consider starting from the linkages between roads to predict future traffic conditions. This problem is divided into two aspects: on the one hand, the clustering problem, that is, which points are incorporated into the set of effects on a particular road; and on the other hand, how the influencing parameters are quantified. The measurement method in [21] is static, that is, to set some fixed influence roads for certain roads. But this is not reasonable, because traffic conditions are dynamic. The quantitative relationship between the roads given in [22] is based on the distance between the roads. Obviously, this method ignores the situation of vehicles traveling on the road. How much traffic on a road should consider.

Other studies have begun to take the neural network approach [23] to automatically learn the non-linear relationship between roads. The trajectory prediction method uses neural networks called Long-Term Memory (LSTM) to analyze the time behavior and predict future coordinates of vehicles around, but, because of the calculation is too complicated when all the road conditions are taken into account, so many methods in determining the input side do not very good. Ref. [24] added social relevance considerations and introduced a dynamic clustering approach to determine neural network input. But the definition of mutual influence on the road is more obscure, which affects the effect of clustering.

The approach [25] introduced how to predict the vehicle running path under the condition of given initial node and the destination node. By trying to find out the operating law of the vehicle, it can predict the vehicle's moving position at a certain time in the future. The approach [26], it is the main research work that closes to us, only for vehicle trajectory prediction in short distance. In order to predict the future trajectory of the vehicle, many mathematical models are used to extract the vehicle motion model and predict the trajectory of the vehicle. Qiao et al. [27] proposed a trajectory prediction method based on Gaussian mixture models (GMTP), this method using the Gaussian mixture model to model the motion pattern and calculate the motion pattern probability. It assumes that the trajectory satisfies the Gaussian model, then using the Gaussian method to predict the object movement trajectories. In Yong et al. [28], the author uses Kalman filter to predict the potential destination of the user or to predict the traffic flow. For short-range prediction research, the traditionally fixed order Markov model and the complex hidden Markov model are applied to the prediction of the trajectory.

Based on the analysis and summary of relevant work, this paper discusses and demonstrates the application of trajectory prediction in vehicular network.

3 System model design

The accuracy of vehicle trajectory prediction is of great significance for vehicle information exchange and data transmission. However, the diversity of the moving objects movement environment always makes the problem more complicated. How to accurately predict the trajectory of moving objects has become an urgent problem to be solved. Despite the mobile object trajectory prediction technology is still in constant perfect, but as a result of theory and technology is not mature, most models not well adapted to the needs of the development of mobile computing technology [29]. For example, it is impossible to accurately predict the motion trajectory by simply using the Markov motion model to predict the vehicle motion trajectory.

In this work, we first explain the trajectory sequence and its characters. Then, by analyzing the historical trajectory data, we divide the vehicle movement motion patterns into two types: single motion pattern and complex motion pattern, and establish the trajectory probability model of different motion patterns by GMM. Finally, we analyze all the possible trajectories of the vehicle in the future by GPR.

3.1 Trajectory sequence

To present the TPVN method better, we first need to explain the trajectory and its characters. Given vehicle moving object database *DB*, which stores a large number of historical trajectory information of vehicle at different times, the ordered set of moving information in time is called trajectory, expressed in $DB = \{Tr_1, Tr_2, \ldots Tr_n\}$, and the number of trajectories is defined as |*DB*|.

Definition 1 Trajectory sequence $Tr = \{s_1, s_2, \ldots s_g\}$ represents a sequence of ordered discrete trajectory points $s_i = \{(x_i, y_i, v_i, t_i)|1 \le i \le g\}$. Where v_i represents the current position of the vehicle speed value, t_i represents the time point, $i \in [1, g), t_i < t_{i+1}, (x_i, y_i)$ represents two-dimensional coordinates of moving objects.

Definition 2 Trajectory data: For two-dimensional space plane *X-axis* and *Y-axis* modeling, the trajectory in two directions are used to represent the trajectory data $D = \left\{(s_x^1, s_x^2, \ldots s_x^m)^T, (s_y^1, s_y^2, \ldots s_y^m)^T\right\} = \{X, Y\}$. Where $s_X^i = (x_1, x_2, \ldots x_m)^T$ denotes the projection vector set of the *i*th trajectory in the *X* direction, $s_y^i = (y_1, y_2, \ldots y_m)^T$ denotes the projection vector set of the *i*th trajectory in the *Y-axis* direction, {*X*, *Y*} represents the trajectory vector set on the trajectory dataset *D*.

3.2 Trajectory probability model

In real life, the movement of the vehicle is affected by the driver's behavior, so the movement of the vehicle has certain regularity under certain circumstances. Therefore, by analyzing the historical trajectory and the regularity of vehicle movement, the vehicle motion pattern can be divided into two types: single motion pattern and complex motion pattern. For the single motion pattern that means there have many trajectory have the same motion pattern. It can only just use a simple Gaussian process model to present. On the contrary, the complex motion pattern means the trajectory have more than one typical motion pattern. Therefore, it cannot be shown with a simple Gaussian process but with the Gaussian mixed model to present.

3.2.1 Single motion pattern

In the single motion pattern, many trajectories have the same motion pattern, and it is considered that the moving object is independent of each other in the x and y directions. A trajectory needs two Gaussian processes (in the x and y directions) to represent it. In this paper, the trajectories are divided into *g-dimensional* in two directions, $X = (x_1, x_2, x_3, \ldots x_g,)^T$ and $Y = (y_1, y_2, y_3, \ldots y_g,)^T$, g represents the number of observation points of the trajectory. For n trajectory with the same motion mode, Gaussian process is used to model it. Due to the noise data often existing in historical GPS data, here we assume that the noise data also obeys Gaussian distribution, that is,$\xi \sim N(0, \psi)$, the trajectory probability model of single motion pattern can be obtained as follows:

$$P(X) = \prod_{i=1}^{m} \sigma_i \varphi(x_i|\mu_{x,i}, \psi_{x,i}) \tag{1}$$

$$P(Y) = \prod_{i=1}^{m} \sigma_i \varphi(y_i|\mu_{y,i}, \psi_{y,i}) \tag{2}$$

where σ_i is the weight of the *i*th Gaussian process, which satisfy $\sigma_i \ge 0$, $\sum_{i=1}^{m} \sigma_i = 1$. $\mu_{x,i}, \psi_{x,i}$ and $\mu_{y,i}, \psi_{y,i}$ represent the mean and covariance of the *i*th trajectory mode state in the X and Y directions, respectively. And $\varphi(x_i|\mu_{x,i}, \psi_{x,i})$ represents the feature vector x_i of the trajectories in the X direction conforms to the distribution density of the Gaussian process trajectory mode:

$$\varphi(x_i|\mu_{x,i}, \psi_{x,i}) = \frac{1}{\sqrt{2\pi}|\psi_{x,i}|} \exp\left[-\frac{(x_i - \mu_{x,i})(x_i - \mu_{x,i})^T}{2\psi_{x,i}^2}\right] \tag{3}$$

Where μ, ψdenotes the mean and covariance matrix of the motion pattern of the Gaussian process in the X direction. The expectation is $\mu=\frac{1}{m}\sum_{i=1}^{m}[E(x_1)+E(x_2)+\cdots E(x_m)]$, covariance matrix is $\psi=\frac{1}{m}\sum_{i=1}^{m}(x_i-\mu_{x,i})(x_i-\mu_{x,i})^T$.

3.2.2 Complex motion pattern

In the scene of a more complex motion pattern, there are more than one typical motion pattern, which is difficult to describe with a Gauss process. Therefore, the trajectory sequence prediction can be carried out by using the Gaussian mixed model (GMM) for multiple Gauss processes. Gaussian mixture model is a finite mixture model, the model of each state with a Gaussian process function, according to data produced by the linear combination of multiple Gaussian process model. In the process of imitation learning path representation, through the teaching data input were coded to study the model. After reconstruction and Gaussian mixture regression on the data generalization output, imitation to get continuous trajectory. And in a scene with multiple trajectory patterns, a trajectory can be attached to multiple motion patterns, and the Gaussian mixed model is needed to accurately depict a trajectory.

Supposing that every data point x_i can be obtained through transformation the data point Z_i($Z \sim N(0, I)$, I is a unit matrix and z_i is obtained through Gauss sampling) for $\Lambda z_i + \mu$ (Λ is a matrix) deformation. Since GPS data is a historical trajectory that often has noise data, we assumed that the noise data also obeys a Gaussian distribution, that is $\xi \sim N(0, \psi)$. For the trajectory training dataset $D = \{X, Y\}$, the probability trajectory model of the complex motion pattern can be obtained as follows:

$$P(x_i|z_{x,i}) = \sum_{i=1}^{m} \sigma_i \varphi(x_i|\mu_{x,i} + \Lambda z_{x,i,}, \psi_{x,i}) \quad (4)$$

$$P(y_i|z_{y,i}) = \sum_{i=1}^{m} \sigma_i \varphi(y_i|\mu_{y,i} + \Lambda z_{y,i}, \psi_{y,i}) \quad (5)$$

Where σ_i is the weight of the ith Gaussian process and satisfy $\sigma_i \geq 0, \sum_{i=1}^{m} \sigma_i = 1$, $\varphi(\cdot)$ represents the Gauss process probability density function, and m represents the number of Gauss processes. $\mu_{x,i} + \Lambda z_{x,i}, \psi_{x,i}$ and $\mu_{y,i} + \Lambda z_{y,i}, \psi_{y,i}$ represent the mean and covariance of the ith trajectory mode state in the X and Y directions, respectively.

Therefore, for the trajectory training set $D = \{X, Y\}$, the Gaussian mixture model likelihood function of the entire trajectory training set is:

$$P(X|Z) = \prod_{i=1}^{m} P(x_i|z_{x,i}) \quad (6)$$

$$P(Y|Z) = \prod_{i=1}^{m} P(y_i|z_{y,i}) \quad (7)$$

Using the Gaussian mixed model to predict the trajectory of the vehicle in complex motion pattern, the most important is to estimate the parameters $\sigma, \mu, \Lambda, \psi$ accurately. In generally, we can use the expectation–maximization algorithm (referred to as EM) to estimate the parameters of the GMM model. A set of parameters was found through iterative training to maximize the value of $P(X|Z)$ or $P(Y|Z)$, that is:

$$(\sigma, \mu, \Lambda, \psi) = \arg \max_{(\sigma,\mu,\Lambda,\psi)} P(X|Z) \quad (8)$$

Because each x_i and y_i are transformed by the corresponding z_i, plus the noise, so X and Y are the posterior distributions of $z : X|Z \sim N(\mu_x + \Lambda z_x, \psi_x)$ and $Y|Z \sim N(\mu_y + \Lambda z_y, \psi_y)$. Thus, Z and X obey the joint Gaussian distribution:

$$\begin{bmatrix} X \\ Z \end{bmatrix} \sim \mathrm{N}(\mu_{XZ}, \Sigma) \quad (9)$$

From $Z \sim N(0, I)$, we can see that $E[Z] = 0$, therefore $E[X] = \mu$. All elements of the matrix Λ are constants, and its expectations are still itself, so you can get: $\mu_{XZ} = \frac{\mu}{\vec{0}}$. Solving Σ requires calculation: $\Sigma_{ZZ} = E[(Z- \ E[Z])(Z-E[Z]^T)], \Sigma_{XZ} = E[(Z-E[Z])(X-E[X]^T)]$ and $\Sigma_{XX} = E[(X-E[X])(X-E[X]^T)]$. Because of $Z \sim N(0, I)$ and $\Sigma_{ZZ} = Cov(Z) = I$, therefore $E[(Z-E[Z])(X-E[X]^T)] = \Lambda^T$ and $E[(X-E[X])(X-E[X]^T)] = \Lambda\Lambda^T + \psi$ can be obtained from $E[ZZ^T] = Cov(Z)$, $E[Z\xi^T] = E[Z]E[\xi^T] = 0$. It can be concluded that:

$$\begin{bmatrix} X \\ Z \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \mu \\ \vec{0} \end{bmatrix}, \begin{bmatrix} \Lambda & \Lambda\Lambda^T + \psi \\ \mathrm{I} & \Lambda^T \end{bmatrix}\right) \quad (10)$$

So the edge distribution of X is as follows $X \sim N(\mu, \Lambda\Lambda^T + \psi)$. Given the training trajectories datasets $\{x_i; i = 1, 2, \ldots, m\}$, the logarithmic likelihood function of the parameter μ, Λ, ψ is established.

$$\ell(\mu, \Lambda, \psi) = \log \prod_{i=1}^{m} \frac{1}{(2\pi)^{m/2}|\Lambda\Lambda^T + \psi|^{1/2}} \times \exp\left(-\frac{1}{2}(x_i-\mu)^T(\Lambda\Lambda^T+\psi)^{-1}(x_i-\mu)\right) \quad (11)$$

Next, we need to further solve the probability of motion mode i, which need calculate the parameter$\sigma$and the value when the partial derivative of parameter μ, Λ, ψ in Eq. (11) is zero. Using EM algorithm to solve, which is divided into two steps *E-step* (seeking expectations) and *M-step* (seeking extreme value), the specific methods are as follows:

1. *E-step* The probability of the trajectory vector x_i of the trajectory sequence in the *X-axis* direction belongs to the motion mode i:

$$P(i|x_i, z_{x,i}) = \frac{\sigma_i P(x_i|i, z_{x,i})}{P(x_i|z_{x,i})} = \frac{\sigma_i \varphi(x_i|\mu_{x,i} + \Lambda z_{x,i}, \psi_{x,i})}{\sum_{i=1}^{m} \sigma_i \varphi(x_i|\mu_{x,i} + \Lambda z_{x,i}, \psi_{x,i})} \tag{12}$$

2. *M-step* using the maximum expectation method to obtain the parameterσiterative estimation formula:

$$\sigma = \sum_{i=1}^{m} P(i|x_i, z_{x,i}) \tag{13}$$

Maximizing the likelihood function of the parameter μ, Λ, ψ and finding the iterative estimation formula of the parameter μ, Λ, ψ:

$$\sum_{i=1}^{m} \int_{z_i} \log \frac{P(x_i, z_{x,i}; \mu_{x,i}, \Lambda_{x,i}, \psi_{x,i})}{z_{x,i}} dz_{x,i} = \sum_{i=1}^{m} E\left[\log P(x_i|z_{x,i}; \mu_{x,i}, \Lambda_{x,i}, \psi_{x,i}) + \log P(z_{x,i}) - \log N(z_{x,i})\right] \tag{14}$$

So it can be obtained:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} x_i \tag{15}$$

$$\Lambda = \left(\sum_{i=1}^{m} (x_i - \mu_{x,i}) \mu_{x_i|z_{x,i}}^T\right) \left(\sum_{i=1}^{m} \mu_{x_i|z_{x,i}} \mu_{x_i|z_{x,i}}^T + \Sigma_{x_i|z_{x,i}}\right)^{-1} \tag{16}$$

$$\psi = \frac{1}{m} \sum_{i=1}^{m} x_i x_i^T - x_i \mu_{x_i|z_{x,i}}^T \Lambda^T - \Lambda \mu_{x_i|z_{x,i}}^T x_i^T + \Lambda\left(\mu_{x_i|z_{x,i}} \mu_{x_i|z_{x,i}}^T + \Sigma_{x_i|z_{x,i}}\right) \Lambda^T \tag{17}$$

where $\mu_{x_i|z_{x,i}}$ and $\Sigma_{x_i|z_{x,i}}$ can be obtained by the Eq. (10), that is: $\mu_{x_i|z_{x,i}} = \mu_{x,i} + (\Lambda \Lambda^T + \psi_{x,i}) \Lambda^{T^{-1}} z_{x,i}$, $\Sigma_{x_i|z_{x,i}} = \Lambda - (\Lambda \Lambda^T + \psi_{x,i}) \Lambda^{T^{-1}} \mathrm{I}$. *E-step* estimates the weight of the Gaussian model by conjecturing the parameters of each Gauss model. *M-step* is based on the weight of the Gaussian model estimated by *E-step*, and then again determines the Gaussian model parameter. Repeating the above two steps until the fluctuation is small and approximately reaches the extreme value.

3.3 Predictive analysis

Entering the training dataset $D = \{(x_i, y_i)\}_{i=1}^{m} = \{X, Y\}$, and the vector as $(x_i, y_i) \in R^g$, which obeys a joint Gaussian distribution. Assuming that the trajectory datasets $\{X, Y\}$ are a multidimensional motion vector, we can randomly generate m independent linear combinations of Gaussian processes $\{\varphi_1, \varphi_2 \ldots \varphi_m\}$. Trajectory training dataset $D_{train} = (x, y)$, x is the input data, y is the output data, trajectory test dataset $D_{test} = (x^*, y^*)$, x^* is the input test data, y^* is the output test data. Then the joint probability density function of y and y^* satisfy the following formula:

$$P(y, y^*) = \sum_{i=1}^{m} \sigma_i \varphi(y, y^*|\mu_i, \psi_i) \tag{18}$$

where $\varphi(y|\mu_i, \psi_i)$ is the Gauss probability density function, $\mu_i = [\mu_{iy}, \mu_{iy^*}]^T, \psi_i = \begin{bmatrix} \psi_{iy} & \psi_{iyy^*} \\ \psi_{iy^*y} & \psi_{iy^*} \end{bmatrix}$. And σ_i is the weight of the ith Gauss process, satisfying $\sigma_i \geq 0$, $\sum_{i=1}^{m} \sigma_i = 1$.

The trajectory prediction can be used to calculate the conditional expectation between the training output and the test output. The trajectory prediction in this paper is to modify the preliminary prediction results of the improved Gauss mixture model. And further predict the vehicle trajectories satisfying different motion models. So, *y^*'s* regression function for y, that is, the predictive value of y^* is:

$$\mathrm{y}^* = E(y^*|x, y, x^*) = \sum_{i=1}^{m} \sigma_i(y) f_i^{\wedge}(y) \tag{19}$$

Where $\sigma_i(y)$ represents the mixed weight of the ith Gauss process in the m Gauss process, and the representation is: $\sigma_i(y) = \sigma_i \varphi(y, \mu_i, \psi_i) / \sum_{i=1}^{m} \sigma_i \varphi(y, \mu_i, \psi_i)$ and $f_i^{\wedge}(y) = \mu^* + \psi \psi^T$. The Eq. (19) is called the trajectory of Gauss mixed regression model, the basic idea is: First, using the conditional density function to model the trajectory data. The second, using EM algorithm to estimate the corresponding parameters, in accordance with the conditional distribution of normal distribution data the regression function of m Gauss components is obtained. Finally, using the Eq. (19) the regression function is weighted to complete the trajectory regression prediction.

3.4 Algorithm framework

The TPVN algorithm framework shown in Fig. 1 and the processes can be divided into four steps:

Step 1 Pre-process historical trajectory data stored in database DB.

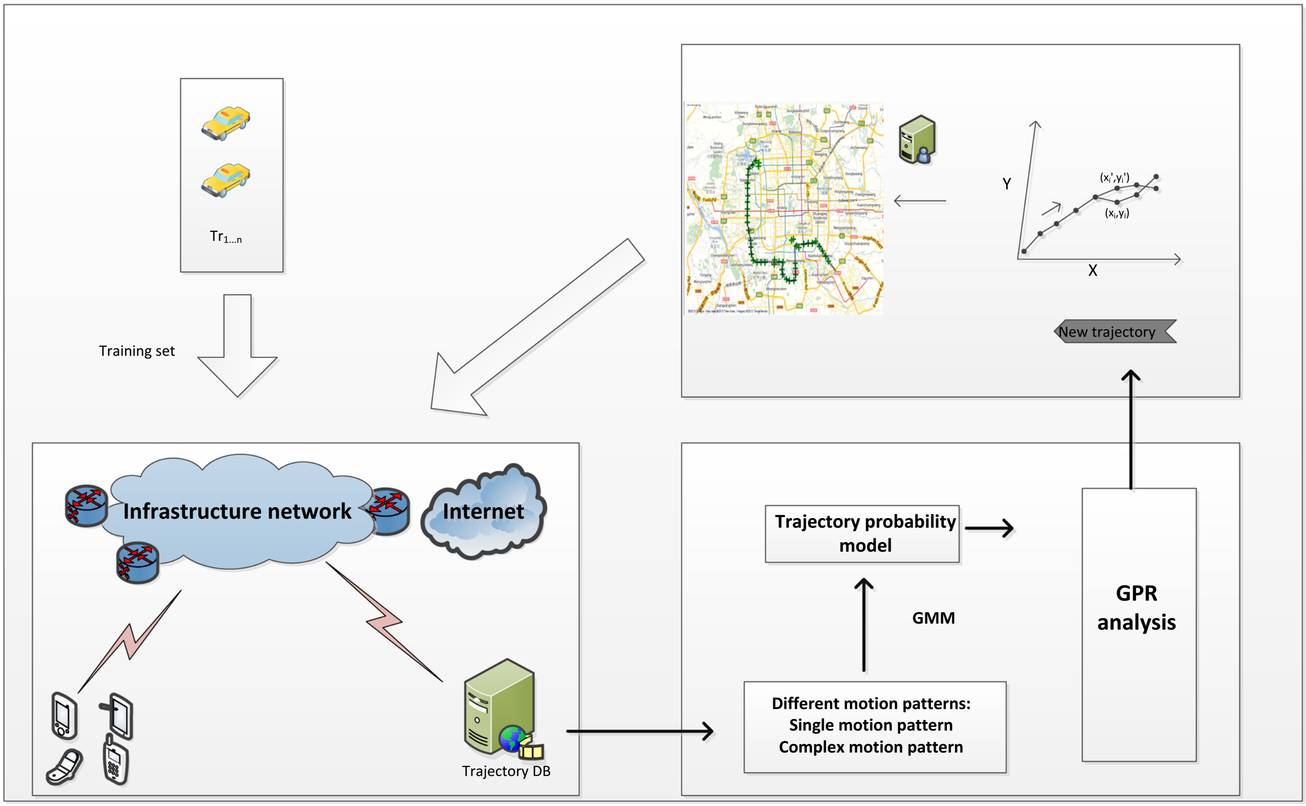


Fig. 1 The framework of TPVN

Step 2 GMM was used to model the trajectory data of different motion pattern. The EM algorithm is used to estimate the parameters of complex motion pattern, and the prediction probability is maximized based on the historical trajectory data.

Step 3 GPR was used to train different motion pattern prediction models, and the most probable trajectory of the vehicle is predicted based on the trajectory data.

Step 4 Output the predicted trajectory route.

3.4.1 Algorithm principle

In Sect. 3.2, according to the characteristics of single and complex of historical trajectory, the models under different conditions are established. Based on this, TPVN algorithm combines GMM, linear regression of least squares method and GR to predict different motion modes.

The classical linear regression method is used to process the spatiotemporal trajectory data and predict the future moving position points.$\{S_1, S_2, \ldots, S_g\}$ is the discrete points of the known historical trajectory, $S_i = \{(x_i, y_i, v_i, t_i)|1 \leq i \leq g\}$, Θ indicates GPR regression prediction equation (Eq (19)). According to the known historical trajectory prediction, the first step is expressed as $\left(S'_{g+1} = \Theta\left(S_g, S_{g-1}, \ldots, S_{g-l-1}\right)\right)$. And then we use the historical trajectory to predict step 2, and so on. Step n, denoted as $S'_{g+n} = \Theta\left(S_{g+n-1}, S_{g+n-2}, \ldots, S_{g+n-l}\right)$. It can be seen that to predict the position of the trajectory at the next moment, the previous prediction value needs to be used as a new step to predict the trajectory position point, and the iterative approach is used to continuously predict the future position point.

In the actual prediction, the historical trajectory is modeled first, and then the regression prediction is performed. The vehicle trajectories are predicted separately in the X and Y directions, and the respective trajectory prediction regression function Θ_x and Θ_y were obtained by modeling separately.$x'_{g+1} = \Theta_x\left(x_g, x_{g-1}, \ldots, x_{g-l-1}\right)$,$y'_{g+1} = \Theta_y\left(y_g, y_{g-1}, \ldots, y_{g-l-1}\right)$.

The training data set $D = \{x_i, y_i\}_{i=1}^{m} = (X, Y)$ is divided into two directions,$D_x = \{x_{i-1}, \Delta x_i\}_{i=2}^{m}$ and $D_y = \{y_{i-1}, \Delta y_i\}_{i=2}^{m}$. Given trajectory information $\{x_1, x_2, \ldots, x_g\}$ predicts the next location point x'_{g+1}. It can be expressed as $x'_{g+1} = \Theta_x(x) + \xi_{x,g}$. And it can be transformed into $\Delta x = \left(\Delta x_2, \Delta x_3, \ldots, \Delta x_g\right)$ prediction to find the incremental Δx_{g+1} of the next position. ξ is the noise data, denoted as $\xi \sim N(0, \psi)$. Then the increment Δx_{g+1} is expressed as:

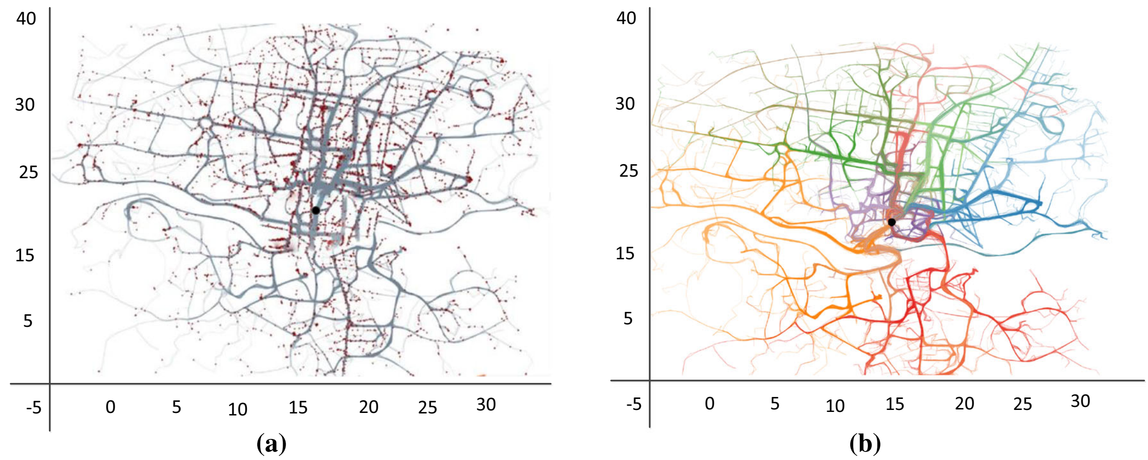


Fig. 2 Example of trajectory prediction. **a** Real trajectories, **b** prediction trajectory

$$\Delta x_{g+1} = \Theta_x(\Delta x) + \xi_{x,g} \tag{20}$$

Using the known observed values $\mathrm{x} = \{x_1, x_2, \ldots, x_{g-1}\}$ and $\Delta x = \{\Delta x_1, \Delta x_2, \ldots, \Delta x_g\}$ to predict Δx_{g+1} and then g +1 position point is obtained in the X direction coordinate, $x'_{g+1} = x_g + \Delta x_{g+1}$. The key to this process is to get the incremental regression function $\Theta_x(\Delta x)$. For simple and complex motion modes, GR predictive regression equation was adopted to obtain:

$$\Delta x_{g+1} = \Theta_x(\Delta x) = \sum_{i=1}^{m} \sigma_i(x) \mathrm{f}_i^{\wedge}(\Delta x) \tag{21}$$

where $\sigma_i(x) = \sigma_i \varphi(x, \mu_i, \psi_i) / \sum_{i=1}^{m} \sigma_i \varphi(x, \mu_i, \psi_i)).$ $\sum_{i=1}^{m} \sigma_i \varphi (x, \mu_i, \psi_i)$ and $f_i^{\wedge}(\Delta x) = \mu^* + \psi \psi^T$. Extracting a part of the historical data $\{S_1, S_2, \ldots, S_g\}$ from the trajectory $Tr = \{s_1, s_2, \ldots s_g\}$ returns to find the predicted increments Δx_{g+1} and Δy_{g+1} of the $g+1$ location point in the X and Y directions. Therefore, the predicted value of the location point is obtained:

$$S'_{g+1} = (x'_{g+1}, y'_{g+1}) = ((x_g + \Delta x_{g+1}), (y_g + \Delta y_{g+1})) \tag{22}$$

Then, according to the trajectory $g+1$ real location point $S_{g+1} = (x_{g+1}, y_{g+1})$, the prediction error ΔS_{g+1} of $g+1$ position point was obtained quantitatively.

$$\begin{aligned} \Delta S_{g+1} &= \sqrt{(x'_{g+1} - x_{g+1})^2 + (y'_{g+1} - y_{g+1})^2} \\ &= \sqrt{(x_g + \Delta x_{g+1} - x_{g+1})^2 + (y_g + \Delta y_{g+1} - y_{g+1})^2} \end{aligned} \tag{23}$$

3.4.2 Algorithm design

For different motion patterns, the trajectory prediction model is established. Each prediction model adapts to traffic scenarios by taking into account driver behavior and environmental factors. We divide the motion pattern into two types: single motion patter and complex motion pattern. In the single trajectory motion pattern, many trajectories have the same motion pattern. On the contrary, in the scene of a more complex motion pattern, there are more than one typical motion pattern, which is difficult to describe with a Gauss process. Therefore, the trajectory sequence prediction can be carried out by using the Gaussian mixed model (GMM) for multiple Gaussian processes. On the Fig. 2, we can see an example of trajectories prediction.

According to the above algorithm framework, we build an algorithm to explain the proposed method, as shown in Table 1.

For this algorithm, we use Gaussian process to model parameters that can be acquired based on different historical data adaptive training. In model training, parameters are obtained by optimizing edge likelihood, and the covariance matrix is inverted for every gradient calculation, the time complexity is $O(n^2 * m)$, n represents the number of trajectories, m represents the number of gradient calculations. Although the training cost of the model is high, the predicted time complexity is $O(n^2)$, where n represents the number of predicted trajectories.

4 Simulations

We implement the proposed algorithm TPVN in the *MATLAP2016a* simulator and evaluate TPVN by performance comparison with GPR prediction algorithm based on Gauss regression and Kalman prediction algorithm based on Kalman filter. The experimental data mainly come from the data of *MIT* parking lots, where more than 40,000 pieces of trajectory data were collected by *MIT*

Table 1 TPVN algorithm

Algorithm Vehicle trajectory prediction algorithm——TPVN
Input: Historical trajectory dataset: $Tr = \{s_1, s_2, \ldots s_g\}$
Output: Vehicle speed at the next moment: $\mathrm{v}_{t+\Delta t}$

$Tr_{test} = \{s_1, s_2, \ldots s_g\}$
if Single motion pattern:
$P(X) = \prod_{i=1}^{m} \omega_i \varphi(x_i \mid \mu_i, \psi_i)$
$P(Y) = \prod_{i=1}^{m} \omega_i \varphi(y_i \mid \mu_i, \psi_i)$
else Complex motion pattern:
$P(X \mid z) = \prod_{i=1}^{m} \sum_{i=1}^{m} \sigma_i \varphi(x_i \mid \mu_i + \Lambda z_i, \psi_i)$
$P(Y \mid z) = \prod_{i=1}^{m} \sum_{i=1}^{m} \sigma_i \varphi(y_i \mid \mu_i + \Lambda z_i, \psi_i)$
end if
for i=1 *to m*
Gauss regression training prediction model
$\mathrm{y}^* = \mathrm{E}(\mathrm{y}^* \mid \mathrm{x}, \mathrm{y}, \mathrm{x}^*) = \sum_{i=1}^{m} \sigma_i(\mathrm{y}) \mathrm{f}_i(\mathrm{y})$
end for
end

laboratories, which can be downloaded by *MITtrajsingle* [30]. Simulation experiments are carried out to analyze the performance of the algorithm. First of all, in order to explain the performance of the proposed algorithm TPVN more specifically, the prediction algorithm GPR based on Gauss regression and the prediction algorithm based on Kalman filter (Kalman) are used to compare the prediction error and prediction time of TPVN algorithm. Secondly, in order to further verify the performance advantage of TPVN algorithm in prediction time, the algorithm is compared with the existing trajectory prediction algorithms TPMO and HMTP.

TPVN is compared with other classical methods to validate its performance. Simulation focuses on the following parameter:

EDS: It is used to calculate the geometrical space error between the real and the predicted position, that is, the geometric space errors between the predicted trajectory points and the actual trajectory points are calculated by using the root mean square error *EDS*.

$$EDS = \frac{1}{n}\sum_{j=1}^{m}\sqrt{(x_j' - x_j)^2 + (y_j' - y_j)^2} \tag{24}$$

where $(x_j, y_j)$is the real trajectory position, (x_i', y_i') is the predicted trajectory position, and n is the number of predicted trajectory points.

4.1 Prediction error

The prediction algorithm GPR based on Gauss regression and the prediction algorithm Kalman based on Kalman filter are used to compare the prediction error and prediction time of TPVN algorithm. The experimental results take the average of all the trajectory prediction errors (*EDS*) under each test set to evaluate the prediction error. The effect of different prediction time on prediction error is verified under the condition that the number of test trajectories from 1 to 1000 and the number of test trajectories is 1000.

The prediction error analysis of three kinds of prediction algorithms under different trajectory test datasets 1–1000 is shown in Fig. 3. Compared with the other two algorithms, the TPVN prediction error is the smallest as the number of test trajectories increases. The *EDS* error remains below 50 m, and as the number of test trajectories increases, the prediction accuracy of TPVN is relatively high and stable.

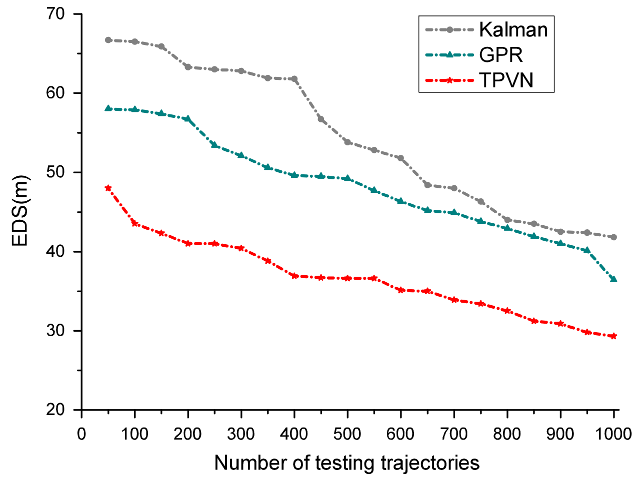


Fig. 3 Prediction error comparison under different number of testing trajectories

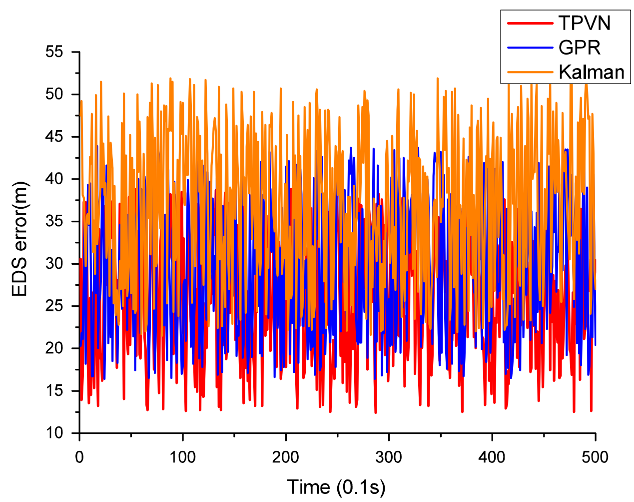


Fig. 4 Prediction error comparison under different prediction time

The experimental results show that the *EDS* error of GPR prediction is about 30 m higher than TPVN average, but it is more accurate than Kalman prediction. The reason is that GPR is more accurate for handling the trajectories with simpler motion patterns, and the dataset trajectories used in the experiment are more complex and wide-ranging, so it is difficult to describe with a single Gaussian process. The TPVN algorithm analyzes the trajectory prediction models of different motion patterns. Trajectory prediction based on Kalman filter only carries out linear regression prediction of trajectory and does not analyze complex trajectory model, so the prediction error is the largest. Compared with GPR and Kalman filter algorithm, TPVN prediction accuracy increased by 80.1% and 71.2% on average.

When the number of test trajectories is 1000, the influence of different prediction time on prediction error as shown in Fig. 4. With the change of simulation time, the prediction error is also changing. The *EDS* prediction error of TPVN is relatively stable at about 25 m, while GPR and Kalman are relatively maintained at 32 and 40 m respectively. This is because this paper is mainly aimed at the prediction of vehicle motion trajectory under short distance, and TPVN has a good effect on motion trajectory prediction under short distance. The TPVN algorithm uses Gaussian regression to predict the different motion patterns and evaluates the most probable trajectory of the vehicle in the future. The GMM is better than pure Gaussian regression and Kalman filter.

Table 2 The EDS improvement of TPVN compared with the other algorithms (%)

Algorithm	GPR	Kalman
Number of testing trajectories	19.9	28.8
Different prediction time	21.9	37.5

Compared with the other algorithms, TPVN has the prediction error reduced by some extent, which is shown in Table 2.

4.2 Prediction time

In order to further verify the performance of the proposed method, the prediction time cost of this algorithm is observed. In Fig. 5, the TPVN method prediction time is very small, compared with the GPR and Kalman filter methods, the average reduction is 56.7 and 32.1%. This is because the Kalman filter substitutes the position information of the previous points into the regression analysis for the next position prediction of each trajectory. When the number of the predicted trajectories increases, the prediction time increases linearly. For the TPVN approach, it can predict the trajectories with uniform model parameters at the same time, and the trajectories depict by the Gaussian process can be predicted only once.

Therefore, when the number of predicted trajectories increases, as long as no more trajectory patterns are added, the prediction time will not be greatly increased or fluctuated. When the number of training trajectories reaches a certain scale, the trajectory patterns are abundant and can contain most of the motion patterns. The TPVN prediction model trained by the Gaussian process can able to deal with the trajectory prediction problems of various complex motion patterns. Due to the limited road traffic environment, the motion patterns of the trajectories added by the test dataset are consistent with the previous patterns. Just when the number of trajectories in the test set increases without any change in motion patterns, there is not much change in the training set data import model for predicting. Therefore, the TPVN prediction error fluctuates less.

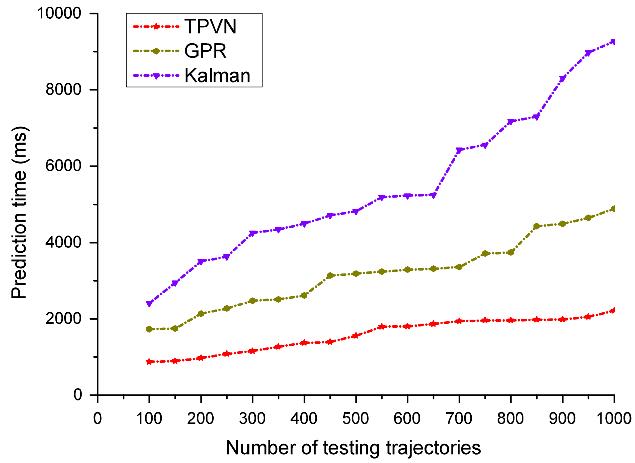


Fig. 5 Prediction time comparison

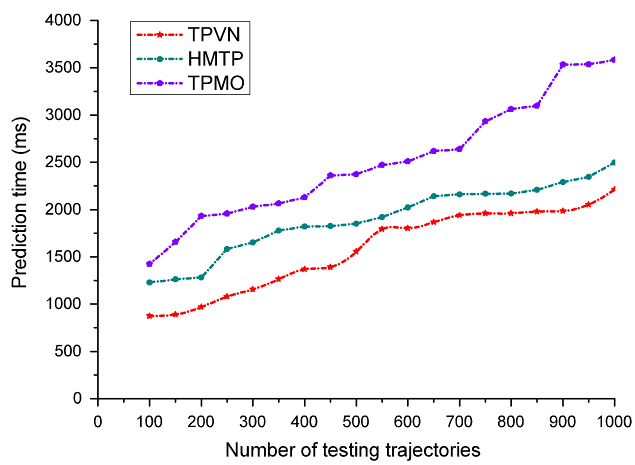


Fig. 6 Prediction time comparison between typical trajectory prediction algorithms

In order to further verify the performance advantages of the proposed algorithm, TPVN is compared with the existing trajectory prediction algorithms with good performance, such as TPMO and HMTP. TPMO and HMTP mainly use hit rate as the evaluation index of prediction effectiveness. However, this paper uses the root mean square error *EDS* to evaluate the accuracy of trajectory prediction, which cannot be compared on a measurement scale. Therefore, this paper compares the prediction time performance of the three algorithms. Verifying that the number of test trajectories varies within the range of 1–1000, and the effect on the experimental results are shown in Fig. 6.

Experimental results show that TPVN is slightly better than HMTP based on Hidden Markov Model and is superior to TPMO based on the time-continuous Bayesian network. The average TPVN prediction time is approximately 3/4 of HMTP and is 1/2 of TPMO. The reason is: TPVN is a Gauss nonlinear probability statistical model, and the training time cost is very low. But HMTP model needs to use HMM model to extract hidden state and observation state, so the cost is relatively high. And The TPMO algorithm not only needs to use the hotspot mining algorithm to cluster trajectories but also need to build a path of continuous time Bayesian networks is extremely time-consuming.

Compared with the other algorithms, TPVN has the prediction time reduced by some extent, which is shown in Table 3.

4.3 Anti-interference analysis

Most of the GPS trajectory data exist some noise data, which will have a great impact on the prediction accuracy of the algorithm. This section verifies the anti-interference performance of the three prediction algorithms to noise data. The influence of noise data on error *EDS* is verified under the condition that the number of test trajectories is 1000, and the influence of noise data on the prediction time is analyzed under the condition that the number of test trajectories is 500 and 1000. The horizontal axis represents the proportion of noise points.

As shown in Fig. 7, the Kalman filtering algorithm is sensitive to noise data. With the increase of noise data, the prediction error increases continuously and approximately linearly. The reason is that: Kalman filter is the autoregressive filter, which uses Kalman filter from a series of observation data with noise estimate process of the internal state is observed, so the noise data changes have a great influence on the error. For TPVN and GPR model, prediction before trajectory clustering model is to categorize noise effectively trajectory points, to predict algorithm can differentiate very well between noise trajectory, thus will not affect other trajectory data prediction.

In order to further verify the performance of this algorithm, the time cost of the three prediction models under different noise data is compared. As we can see from

Table 3 The EDS improvement of TPVN compared with the other algorithms (%)

Algorithm	GPR	Kalman	HMTP	TPMO
number of Testing trajectories	43.3	67.9	23.2	47.9

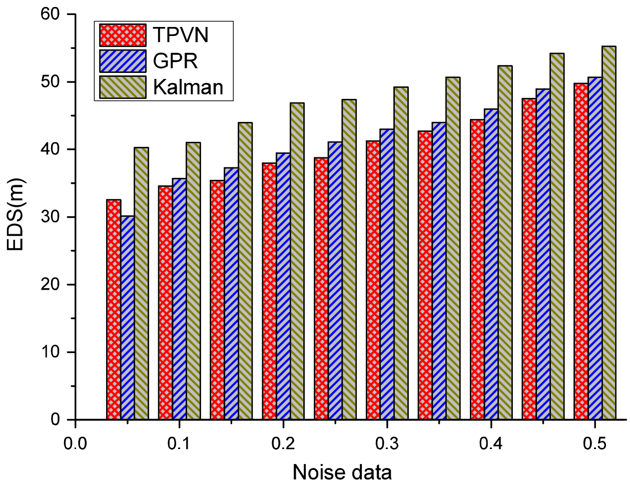


Fig. 7 Prediction error comparison under different portion of noise data

Fig. 8, the time cost of the Kalman filter algorithm is relatively high. The reason is that the Kalman filter model assumes that the real state at time k is evolved from the state at time $i-1$, so each step must be analyzed and determined according to the previous step. Thus its time efficiency is relatively low. While, the TPVN and GPR algorithms excavated the frequent patterns of trajectories before the prediction, and the trajectories formed by the mean points represent a large number of predicted routes with the same motion pattern, so the time cost is low. In addition, the time cost of TPVN prediction is lower than GPR algorithm, which further proves the timeliness of the algorithm.

Table 4 The improvement of noise data anti-interference TPVN compared with the other algorithms (%)

Algorithm	GPR	Kalman
Different portion of noise data	4.4	17.7
Number of testing trajectories		
Number of test trajectories = 500	58.2	73.9
Number of test trajectories = 1000	54.5	63.0

Compared with the other algorithms, TPVN has the EDS error and prediction time reduced by some extent, which is shown in Table 4.

5 Conclusions

Trajectory prediction of mobile objects is a new and challenging research topic, what's more, for the vehicle network which taking the vehicle as the research object has a more challenging to make a fairly accurate prediction of its trajectory. In this research, by analyzing the behavior state of the vehicle's historical trajectory, the vehicle motion mode is divided into two categories: single and complex. The Gaussian process was used to model the two motion patterns and the probability trajectory model was obtained. The advantages of this algorithm is that the parameters of the algorithm implementation process does not need to consider too much, the vehicle can according to different motion probability model, the statistical distribution characteristics of a trajectory to get the position information of different motion patterns.

Future research work, in order to improve the adaptability of the algorithm to environmental factors, we will

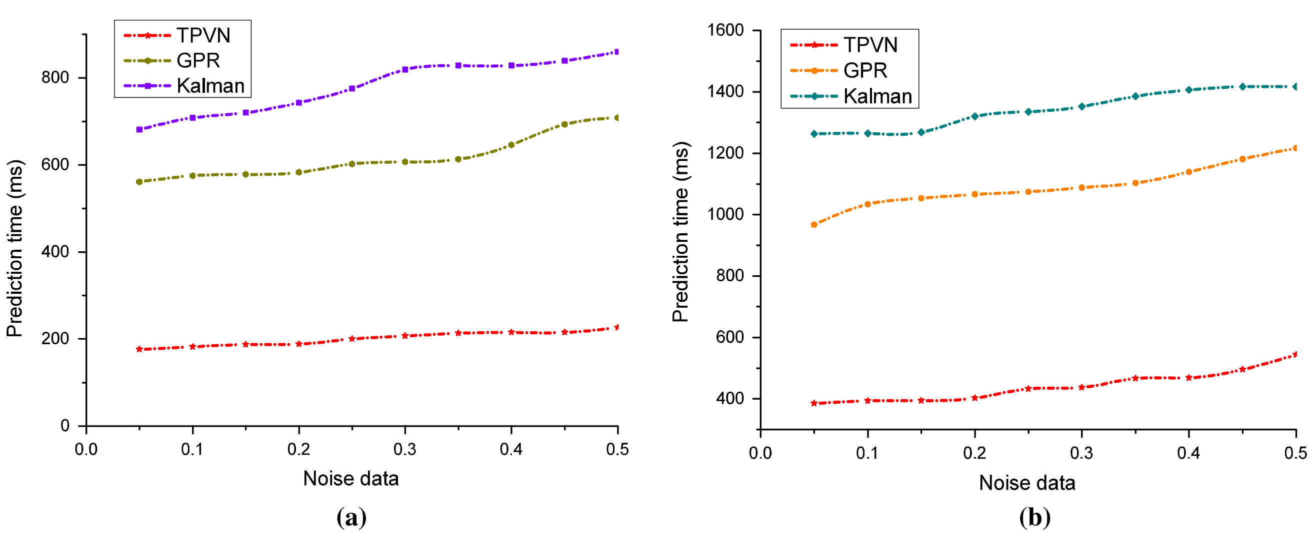


Fig. 8 Prediction time comparison under different portion of noise data. **a** Number of test trajectories = 500, **b** number of test trajectories = 1000

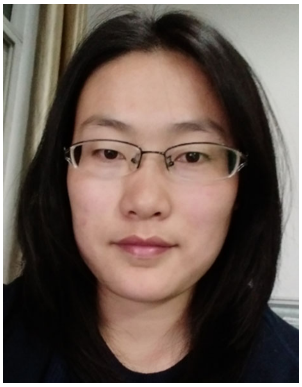
consider more realistic factors such as traffic lights, traffic jams and so on. And consider the impact of road shape on prediction.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Grant Nos. 71633006, 616725407). This work is supported by the China Postdoctoral Science Foundation funded project (Grant No. 2017M612586). This work is supported by the Postdoctoral Science Foundation of Central South University (Grant No. 185684). Also, this work was supported partially by "Mobile Health" Ministry of Education-China Mobile Joint Laboratory.

Author contributions L.W. and J.W. conceived and designed the experiments; L.W. performed the experiments; L.W analyzed the data; Z.C. and J.L contributed reagents/materials/analysis tools; L.W. wrote the paper.

References

1. Jia, W. U., Chen, Z., & Zhao, M. (2017). Effective information transmission based on socialization nodes in opportunistic networks. *Computer Networks*, *129*(1), 297–305.
2. Wu, J., & Chen, Z. (2017). Human activity optimal cooperation objects selection routing scheme in opportunistic networks communication. *Wireless Personal Communications, 95*(3), 3357–3375. https://doi.org/10.1007/s11277.
3. Liu, P., Kurt, A., & ÖZgüNer, U. (2014). Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification. In *IEEE, international conference on intelligent transportation systems* (pp. 942–947). IEEE.
4. Xie, G., Gao, H., Qian, L., et al. (2018). Vehicle trajectory prediction by integrating physics- and maneuver-based approaches using interactive multiple models. *IEEE Transactions on Industrial Electronics, 65*(7), 5999–6008.
5. Ramakrishnan, B., Nishanth, R. B., Joe, M. M., et al. (2017). Cluster based emergency message broadcasting technique for vehicular ad hoc network. *Wireless Networks, 23*(1), 233–248.
6. Taylor, J., Zhou, X., Rouphail, N. M., et al. (2015). Method for investigating intradriver heterogeneity using vehicle trajectory data: A dynamic time warping approach. *Transportation Research Part B, 73*, 59–80.
7. Hu, W. C., Yang, H. J., & Kaabouch, N. (2016). Secure spatial trajectory prediction based on traffic flows. In *IEEE international conference on electro information technology* (pp. 0723–0727). IEEE.
8. Kong, X., Xu, Z., Shen, G., et al. (2016). Urban traffic congestion estimation and prediction based on floating car trajectory data. *Future Generation Computer Systems, 61*(C), 97–107.
9. Xu, X., Lian, C., Wang, J., He, H. G., & Hu, D. (2016). Actor–critic reinforcement learning for autonomous control of unmanned ground vehicles. *Science Robotics, 354*, 42–47.
10. Wang, L., Chen, Z., & Wu, J. (2017). An opportunistic routing for data forwarding based on vehicle mobility association in vehicular ad hoc networks. *Information, 8*(4), 140.
11. Qiao, S., Shen, D., Wang, X., et al. (2015). A self-adaptive parameter selection trajectory prediction approach via hidden markov models. *IEEE Transactions on Intelligent Transportation Systems, 16*(1), 284–296.
12. Schreier, M., Willert, V., & Adamy, J. (2016). An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments. *IEEE Transactions on Intelligent Transportation Systems, 17*(10), 2751–2766.
13. Liu, K., & Lim, H. B. (2012). Positioning accuracy improvement via distributed location estimate in cooperative vehicular networks. In 15th *international IEEE conference on intelligent transportation systems (ITSC), 2012*, (pp. 1549–1554). IEEE.
14. Zhang, R., Cao, L., Bao, S., et al. (2017). A method for connected vehicle trajectory prediction and collision warning algorithm based on V2V communication. *International Journal of Crashworthiness*, *22*(1), 15–25.
15. HouéNou, A., Bonnifait, P., & Cherfaoui, V. (2014). Risk assessment for collision avoidance systems. In *IEEE international conference on intelligent transportation systems* (pp. 386–391). IEEE.
16. Liu, P., Kurt, A., & Ner, U. G. (2014). Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification. In *2014 IEEE 17th international conference on proceedings of intelligent transportation systems (ITSC)* (pp. 942–947), November 7–12, 2014. IEEE.
17. Houenou, A., Bonnifait, P., & Cherfaoui, V. (2013) Vehicle trajectory prediction based on motion model and maneuver recognition. In *2013 IEEE/RSJ international conference on proceedings of the intelligent robots and systems (IROS)* (pp. 4363–4369), November 6–9, 2013. IEEE.
18. Wen, X., Shao, L., Xue, Y., & Fang, W. (2015). A rapid learning algorithm for vehicle classification. *Information Sciences, 295*(1), 395–406.
19. Yin, X., Wang, B., & Li, W. (2015). Background subtraction for moving cameras based on trajectory-controlled segmentation and label inference. *KSII Transactions on Internet and Information Systems, 9*(1), 4092–4107.
20. Hu, W. C., Yang, H. J., & Kaabouch, N. (2016). Secure spatial trajectory prediction based on traffic flows. In *IEEE international conference on electro information technology* (pp. 0723–0727). IEEE.
21. Kong, X., Xu, Z., Shen, G., et al. (2016). Urban traffic congestion estimation and prediction based on floating car trajectory data. *Future Generation Computer Systems, 61*(C), 97–107.
22. Mei, Y. C., Angeline, L., Chin, R. K. Y., et al. (2007). Vehicle trajectory clustering for traffic intersection surveillance. In *IEEE international conference on consumer electronics-Asia* (pp. 1–4). IEEE.
23. Kim, B. D., Kang, C. M., Lee, S. H., et al. (2017). Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. https://doi.org/10.1109/ITSC.2017.8317943.
24. Pérez-Hurtado, I., Capitán, J., Caballero, F., et al. (2016). Decision-theoretic planning with person trajectory prediction for social navigation. In L. Reis, A. Moreira, P. Lima, L. Montano, & V. Muñoz-Martinez (Eds.), *Robot 2015: Second Iberian Robotics Conference. Advances in Intelligent Systems and Computing* (Vol. 418). Springer.
25. Ye, N., Wang, Z. Q., Malekian, R., et al. (2015). A method of vehicle route prediction based on social network analysis. *Journal of Sensors, 2*, 1–9.
26. Schreier, M., Willert, V., & Adamy, J. (2016). An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments. *IEEE Transactions on Intelligent Transportation Systems, 17*(10), 2751–2766.
27. Qiao, S. J., Jin, K., Han, N., & Tang, C. J. (2015). Trajectory prediction algorithm based on Gaussian mixture model. *Journal of Software, 26*(5), 21–32.
28. Yong, K. K., Han, J. W., & Park, H. (2015). Trajectory prediction for using real data and real meteorological data. *Lecture Notes in Electrical Engineering, 331*, 89–103.
29. Besse, P. C., Guillouet, B., Loubes, J. M., et al. (2016). Destination prediction by trajectory distribution based model. *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2017.2749413.
30. MITtrajsingle: http://www.ee.cuhk.edu.hk/~xgwang/MITtrajsingle.html.



Lei-lei Wang (1992–), female, master's degree, research direction: opportunity network.

Zhi-gang Chen (1964–), male, Ph.D., professor, member of CCF (E200005226S), research direction: network and distributed computing.

Jia Wu (1983–), male, Ph.D., member of CCF (E200039369M), research direction: Opportunities Network and Software Engineering.