Research
Vehicle Engineering—Article

A Probabilistic Architecture of Long-Term Vehicle Trajectory Prediction for Autonomous Driving

Jinxin Liu, Yugong Luo *, Zhihua Zhong, Keqiang Li, Heye Huang, Hui Xiong

State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100084, China

ARTICLE INFO

ABSTRACT

In mixed and dynamic traffic environments, accurate long-term trajectory forecasting of surrounding vehicles is one of the indispensable preconditions for autonomous vehicles to accomplish reasonable behavioral decisions and guarantee driving safety. In this paper, we propose an integrated probabilistic architecture for long-term vehicle trajectory prediction, which consists of a driving inference model (DIM) and a trajectory prediction model (TPM). The DIM is designed and employed to accurately infer the potential driving intention based on a dynamic Bayesian network. The proposed DIM incorporates the basic traffic rules and multivariate vehicle motion information. To further improve the prediction accuracy and realize uncertainty estimation, we develop a Gaussian process-based TPM, considering both the short-term prediction results of the vehicle model and the driving motion characteristics. Afterward, the effectiveness of our novel approach is demonstrated by conducting experiments on a public naturalistic driving dataset under lane-changing scenarios. The superior performance on the task of long-term trajectory prediction is presented and verified by comparing with other advanced methods.

1. Introduction

Over the past few years, there has been widespread and extensive investigation about autonomous driving in different areas, particularly in the field of intelligent transportation. One major challenge to fully realize autonomous driving is achieving favorable situational awareness and a comprehensive understanding of the stochastic environment. To address part of this problem, one feasible solution is to forecast the trajectory of the surrounding vehicles, which can provide an anticipatory assessment of the driving situation around the ego vehicle, thereby avoiding imminent or potential threats to safe driving [1,2].

Although interaction-aware trajectory prediction considering the influence of multiple vehicles has been an advanced research focus [3,4], while a majority of autonomous vehicles (AVs) cannot perceive the motion state of vehicles over a distance in a mixed traffic environment without communication technology. Thus, our work concentrates on the trajectory prediction of the preceding or adjacent vehicle from the ego vehicle perspective.

To date, some researchers have dedicated themselves to vehicle trajectory prediction. In terms of approaches primarily applied in this research area, the current methods can be classified into three categories [5–7]: vehicle model-based prediction, maneuver-based prediction, and deep learning-based prediction. Vehicle model-based prediction is a straightforward and simple method [8]. It only uses basic motion models, including kinematic vehicle models such as the constant velocity (CV) model, constant acceleration (CA) model, and constant turn rate and acceleration (CTRA) model [9], as well as dynamic vehicle models such as the two-wheeled "bicycle" model [10]. Considering the effect of model uncertainty, a variety of filtering algorithms can be applied to these linear or nonlinear models, such as Kalman filtering (KF), extensive KF (EKF) [11], unscented KF (UKF) [12], and particle filtering (PF) [13]. Generally, this method can achieve good performance in short-term (less than one second [5]) prediction since it utilizes the laws of physics. However, it is inadequate for long-term prediction due to the lack of consideration of high-level vehicle information. It should be noted that we define a short-term prediction if the prediction duration is less than one second, and a long-term prediction if the prediction duration is more than two seconds.

As to the maneuver-based prediction, it assumes that the future vehicle trajectory is consistent with the recognized intention that

the vehicle tends to execute [5,14]. Therefore, many state-of-the-art studies need to estimate driving behavior or intention first and subsequently make trajectory predictions [15,16]. Specifically, discriminative classifiers such as support vector machines [17] and generative models such as hidden Markov models (HMMs) [18] are frequently exploited for intention estimation. A full-scale description and advanced research about this can be found in Refs. [6,19,20]. For trajectory prediction based on the recognized intention, motion pattern-based methods such as Gaussian processes (GPs) [21,22] and other intention-based methods [23] are predominantly used. Li et al. [24] divided the vehicle trajectories into several typical patterns using a Gaussian mixture model. According to these patterns, the traffic modeling and motion uncertainties were derived from GP. Schreier et al. [23] established a Bayesian network to infer the driving maneuvers of each vehicle. Then, a probabilistic trajectory prediction model (TPM) was built through motion planning approaches integrating stochastic elements. In general, while maneuver-based prediction is prone to have an initial low accuracy, it is relatively suitable for long-term prediction with uncertainties due to a high-level reasoning of vehicles.

Furthermore, there are a few methods combining the vehicle model and maneuver model for trajectory prediction [25–27]. Houenou et al. [25] forecasted the trajectory considering both vehicle kinematics and maneuver recognition to take advantage of the prediction accuracy in the short and long term. In Ref. [26], Xie et al. used an interactive multiple model for trajectory prediction, which combined physics- and maneuver-based approaches and achieved a more accurate trajectory within a long prediction horizon. However, there still exist some limitations. Most of the model parameters are defined manually, and the driving motion characteristics (Fig. 1) are not taken into account.

For deep learning-based prediction, numerous trajectory prediction frameworks are based on deep neural networks such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory (LSTM) networks, or a combination of them [28–33]. An LSTM encoder–decoder model employing convolutional social pooling, which can generate a multimodal predictive distribution over prospective trajectories, was built in Ref. [28]. In Ref. [29], an encoder–decoder architecture based on relational RNNs was introduced. The encoder explored the patterns of previous trajectories, while the decoder created the potential trajectory sequence. Yan et al. [30] developed an LSTM encoder–decoder framework with two spatial-attention mechanisms to improve the prediction accuracy. Mo et al. [31] used CNN–LSTMs to predict interaction-aware trajectories of connected vehicles and LSTMs to make personalized prediction of vehicle states [32,33]. Although these methods are rather appropriate to deal with interactions and perform well in long-term prediction, it is

challenging to introduce traffic rules or vehicle models to improve the prediction accuracy.

However, the scope of our paper focuses on the mixed and non-connected traffic environments under which the intelligent vehicles will drive in the recent and future years, it is difficult to provide sufficient information for these deep learning-based models. Regardless of whether the behavior of the predicted vehicle is influenced by other factors, the historical motion trajectory of it always exists objectively, and we can judge the tendency of the predicted vehicle based on its actual historical information.

Motivated by the aforementioned research gaps, a comprehensive scheme of long-term vehicle trajectory prediction under uncertainty is proposed. The main contributions of this study are briefly summarized as follows:

(1) We propose an integrated architecture for long-term vehicle trajectory prediction driven by both the vehicle model and naturalistic driving data. The overall probabilistic framework is interpretable and can reduce data dependencies and deal with prediction uncertainties in a dynamic environment.

(2) A driving inference model (DIM) is designed to reveal and extrapolate high-level vehicle information concerning driving intention and motion characteristics, incorporating the basic road rules and low-level vehicle motion elements in longitudinal and lateral directions.

(3) Considering the vehicle motion characteristics and integrating the short-term prediction results of the kinematic vehicle model, a TPM is developed to ensure the precision of the entire prediction process.

The remainder of this paper is organized as follows. Section 2 presents an overview of the system architecture. Section 3 and Section 4 introduce the details of the method development, consisting of the driving inference module and trajectory prediction module. Comprehensive experiments are compared and analyzed using the naturalistic driving dataset in Section 5, and concluding remarks are drawn in Section 6.

2. System architecture

This paper is motivated by the problem of allowing AVs to have better situational awareness in a complex traffic environment. For example, AVs need to automatically decide the next maneuver and perform trajectory planning after predicting the future trajectories of adjacent vehicles. Existing approaches to forecasting vehicle trajectories are various but short of considering the comprehensive factors that highly affect the final prediction accuracy. In this paper, under a nonconnected environment, we conclude that the significant factors can be divided into two categories: high-level information, including driving intention
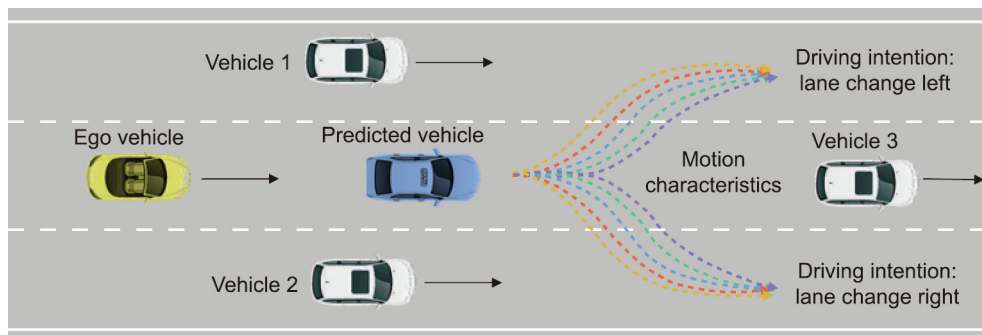

Fig. 1. The illustration of motion characteristics. The yellow ego vehicle is an AV, and the blue vehicle is one of the surrounding vehicles that will be predicted. Under one particular driving intention, the predicted vehicle may have various motion movements. In this paper, we identify these features as motion characteristics.

and motion characteristics, and low-level information, including physical movement and traffic rules.

Our study presents a probabilistic architecture integrating these elements, through which AVs can predict the long-term trajectory of a surrounding vehicle to fulfill superior decision-making in the traffic environment. The architecture overview, shown in Fig. 2, is mainly composed of a driving inference module, vehicle model-based prediction, and long-term trajectory prediction module. Note that the naturalistic driving data are used to provide the necessary information, which can be obtained through driving intention calibration and driving characteristic classification based on a sequence clustering algorithm.

The primary objective of the driving inference module is to generate the probability of the driving intention and motion characteristics via the DIM. We first define the structure of the DIM based on a dynamic Bayesian network. Then, the parameters of the DIM can be learned through the data training process. Finally, given the historical and environmental information of the predicted vehicle, the probabilistic inference of the DIM will be output. Based on the vehicle motion information in the past, we can make accurate trajectory predictions in the short term through a kinematic vehicle model. First, a nonlinear vehicle model is established. After that, we implement the PF to filter the historical trajectory points of the vehicle. In addition, future short-term prediction position points can be generated by the PF. We call both the filtered historical points and the future short-term points "support points."

For the long-term trajectory prediction module, we build the TPM based on the GP according to the inference results of the DIM. Afterward, the model parameters are acquired using a data learning method. Eventually, the long-term future trajectory with a certainty description can be predicted by fusing the short-term support points. In this paper, the TPM is our final model for long-term vehicle trajectory prediction and DIM is an indispensable part for driving intention inference.

3. Driving inference module

This driving inference module aims to recognize the driving intention and motion characteristic probabilities, which will be used as inputs to the trajectory prediction module. This section has three parts: DIM construction, model data training, and model probabilistic inference. Moreover, we will illustrate the details of these aspects in the following.

3.1. DIM construction

Here, we first define the DIM structure based on the theory of a dynamic Bayesian network [34]. With the assumption of the first-order Markov chain, the DIM is constructed as a directed acyclic graphical model. To determine the structure of a dynamic Bayesian network, we need to first define the prior network and the transition network. The prior network defines the connection between the nodes at the initial time $t = 1$, and the transition network defines the connection between time t and $t + 1$.

In Fig. 3, we present the transition network of the DIM, which contains two types of nodes, hidden nodes $\{H_1, H_2, M_1, M_2\}$ and observable nodes $\{O_1, O_2, O_3, O_4\}$. Specifically, nodes $\{H_1, H_2\}$ denote the high-level abstract information, the driving intention and motion characteristic, separately; $\{M_1, M_2\}$ denote mixture parameters with fixed values. Nodes $\{O_1, O_3, O_4\}$ represent the observed motion variables containing the longitudinal or lateral
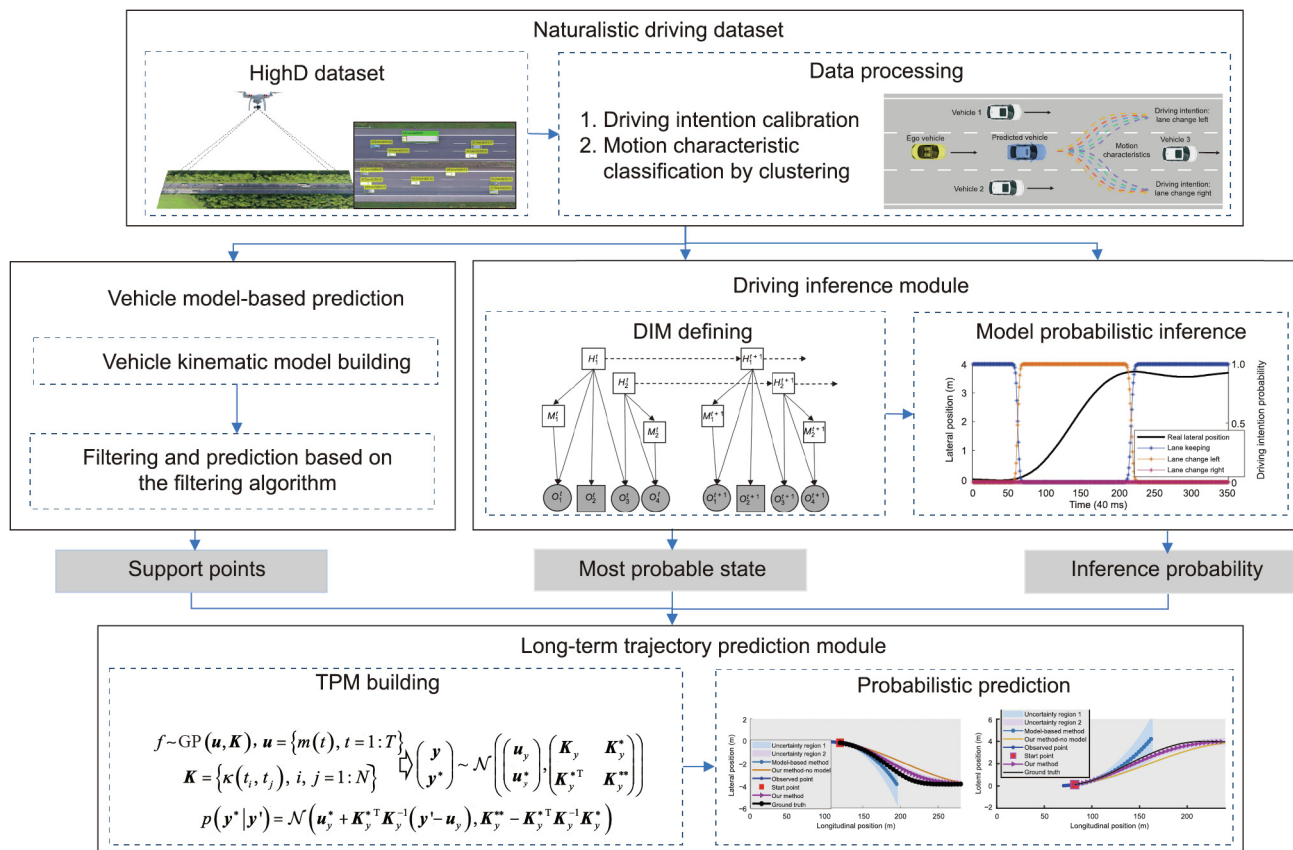

Fig. 2. Probabilistic architecture diagram for long-term vehicle trajectory prediction. The three main blocks, the driving inference module, the vehicle model-based prediction, and the long-term trajectory prediction module, are presented in detail, and the information flow among them is explained. The definition of parameters can be found in corresponding sections.
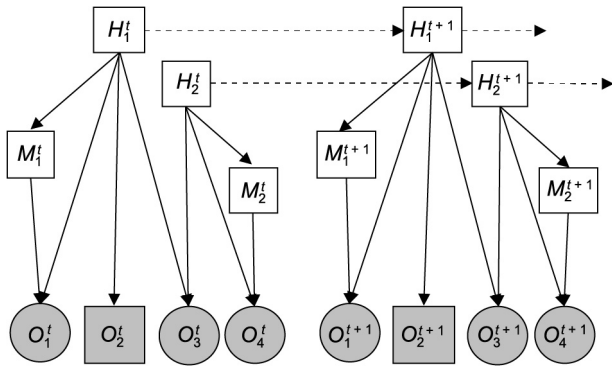
Fig. 3. The designed structure of the DIM based on a dynamic Bayesian network. The squares and circles denote discrete and continuous nodes, respectively; the clear and shaded nodes denote hidden and observed nodes, respectively. The nodes in the network represent the predefined variables, and the connections between nodes in the form of arrows represent the conditional probability distributions (CPDs).

position, velocity, or acceleration; node $\{O_2\}$ will be represented by a four-dimensional vector $\boldsymbol{o}_2$, which is composed of Boolean values, representing whether it is the leftmost or rightmost lane and whether there is an adjacent vehicle on the left or right. If the value of $\{O_2\}$ is $[0, 0, 1, 1]$, it means that the vehicle is located in a lane that is not the leftmost or rightmost lane, but there are adjacent vehicles both on the left and right sides. As shown in Fig. 1, the lane where the predicted blue vehicle is located is not the leftmost or rightmost one. However, there exist white vehicles alongside the predicted vehicle both on the left and right sides. In a real driving environment, the historical trajectory of a vehicle sometimes deviates from the centerline of the lane, causing the misrecognition of driving intentions. Therefore, we introduce the information about basic road or driving rules into the DIM to restrict the driving intention and improve the accuracy rate of the DIM for driving intention recognition.

After defining the structure of the DIM, we can set up the joint and conditional probability distributions (CPDs) of the DIM. The DIM can be modeled as a stochastic process over a set of random variables $Z^t = \{H_1^t, H_2^t, M_1^t, M_2^t, O_1^t, O_2^t, O_3^t, O_4^t\}$ at time t. The joint probability distribution of the variables within time T can be defined as follows:

$$P(Z^{1:T}) = \prod_{t=1}^{T} \prod_{i=1}^{N} P(Z_i^t | \mathrm{Pa}(Z_i^t)) \tag{1}$$

where the symbol P means the probability; Z_i^t is the ith node at time t; $\mathrm{Pa}(Z_i^t)$ are the parents of Z_i^t in the network; and the symbol N means the number of nodes contained in the variables Z^t.

As to the CPDs in the DIM, they can be categorized into three types of matrices: a prior distribution matrix $\boldsymbol{\pi}$ of hidden variables, a transfer matrix $\boldsymbol{A}$ of hidden variables, and an observation matrix $\boldsymbol{B}$. The first two matrices are expressed as follows:

$$\begin{aligned} \boldsymbol{\pi} &= \{\pi_i\}, \pi_i = P(H_i^1), i = 1, 2 \\ \boldsymbol{A} &= \{a_i\}, a_i = P(H_i^{t+1} | H_i^t), i = 1, 2 \end{aligned} \tag{2}$$

where $\{\pi_1, \pi_2\}$ represent the prior probabilities of nodes $\{H_1, H_2\}$ at time $t = 1$ and $\{a_1, a_2\}$ denote the transition probabilities of nodes $\{H_1, H_2\}$ between time t and $t + 1$;

The last matrix $\boldsymbol{B}$ can be expressed as:

$$\begin{aligned} \boldsymbol{B} &= \{b_i\}, b_i = P(M_i^t | H_i^t), i = 1, 2 \\ b_3 &= P(O_2^t | H_1^t); b_4 = P(O_1^t | H_1^t, M_1^t) \\ b_5 &= P(O_3^t | H_1^t, H_2^t); b_6 = P(O_4^t | H_2^t, M_2^t) \end{aligned} \tag{3}$$

where $\{b_1, b_2\}$ represent the conditional probabilities of nodes $\{H_1, M_1\}, \{H_2, M_2\}$ at time t. Similarly, $\{b_3, b_4, b_5, b_6\}$ refer to the conditional probabilities between the observable nodes $\{O_1, O_2, O_3, O_4\}$ and the hidden nodes $\{H_1, H_2, M_1, M_2\}$ at time t.

For the conditional probabilities of b_3 in Eq. (3), we will need to introduce the constraints of traffic rules according to the current driving environment. Take the scene of highway as an example, the probability of b_3 that a vehicle is in the leftmost or rightmost lane with the intention of left or right lane change is zero, and the probability that there is an adjacent vehicle in the left or right lane with the intention of left or right lane change is zero. For the predicted blue vehicle in Fig. 1, the probability of b_3 that there are adjacent vehicles in the left and right with the intention of left or right lane change is zero. These conditional probabilities will affect the subsequent probabilistic inferences.

It is noted that the DIM is time-homogeneous, assuming the parameters in CPDs are time-invariant. In addition, $\{b_4, b_5, b_6\}$ can be expressed using Gaussian distributions since nodes $\{O_1, O_3, O_4\}$ are continuous variables.

3.2. Data training of the model

After specifying the CPDs in the DIM, the parameters in $\{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}\}$ are defined as the model parameter λ. The task of model data training is to learn the model parameter λ, given the observation sequence $\boldsymbol{O} = \{\boldsymbol{o}^t = o_i^t\}(t = 1 : T; i = 1, 2, 3, 4)$ of the observed nodes $\{O_1, O_2, O_3, O_4\}$. Since there may exist missing data in $\boldsymbol{O}$, we can utilize the maximum likelihood estimation (MLE) method with the expectation–maximization (EM) algorithm [35] to optimize and obtain the λ, consisting of two steps—the expectation and maximization steps.

First, we define the likelihood function $F(\lambda)$ as:

$$F(\lambda) = P(\boldsymbol{O}|\lambda) = \sum_{\boldsymbol{S}} P(\boldsymbol{O}, \boldsymbol{S}|\lambda) \tag{4}$$

where $\boldsymbol{S}$ is the state sequence $\boldsymbol{S} = \{\boldsymbol{s}^t = s_i^t\}(t = 1 : T; i = 1, 2)$ of the hidden nodes $\{H_1, H_2\}$.

Then, using the initialized parameter $\bar{\lambda}$, $R(\lambda|\bar{\lambda})$, the expectation of the complete likelihood function $\log P(\boldsymbol{S}, \boldsymbol{O}|\lambda)$ under the conditional distribution $P(\boldsymbol{S}|\bar{\lambda}, \boldsymbol{O})$, can be expressed as:

$$R(\lambda|\bar{\lambda}) = E[\log P(\boldsymbol{S}, \boldsymbol{O}|\lambda)|\bar{\lambda}, \boldsymbol{O}] = \sum_{\boldsymbol{S}} P(\boldsymbol{S}|\bar{\lambda}, \boldsymbol{O}) \log P(\boldsymbol{O}, \boldsymbol{S}|\lambda) \tag{5}$$

where $E[\cdot]$ represents the expectation function.

In order to calculate $R(\lambda|\bar{\lambda})$, we use the forward and backward algorithm [35,36]. The related forward variable $\alpha^t(\boldsymbol{s}^t)$ and backward variable $\beta^t(\boldsymbol{s}^t)$ can be expressed as:

$$\begin{aligned} \alpha^t(\boldsymbol{s}^t) &= P(\boldsymbol{O}^{1:t}, \boldsymbol{S}^t = \boldsymbol{s}^t | \lambda) \\ \beta^t(\boldsymbol{s}^t) &= P(\boldsymbol{O}^{t+1:T} | \boldsymbol{S}^t = \boldsymbol{s}^t, \lambda) \end{aligned} \tag{6}$$

The specific computational procedure for $\alpha^t(\boldsymbol{s}^t)$ and $\beta^t(\boldsymbol{s}^t)$ can be found in Ref. [35], which has a derivation of the forward–backward algorithm for HMMs. Based on $R(\lambda|\bar{\lambda})$, which can be expressed by Eq. (6), the model parameter λ can be obtained by maximizing $R(\lambda|\bar{\lambda})$ over λ using Jensen's inequality theory [35,37]. By treating the model training task as an optimization problem with constraints on the likelihood function $F(\lambda)$, which is subject to some normalization restrictions, a standard Lagrange optimization can be constructed by using Lagrange multipliers to find new estimated parameter λ. The whole iterative procedures are shown in Algorithm 1.

3.3. Probabilistic inference of the model

With the model parameter λ derived from model training, we can make the probability inference of the DIM, including the inference of the driving intention and the motion characteristics. This part aims to find the most likely state sequence $\boldsymbol{S}$, given the observation sequence $\boldsymbol{O}$ within the time T.

In the beginning, we define an interim variable $r^t(\boldsymbol{s}^t)$ using the Bayesian formula:

$$r^t(\boldsymbol{s}^t) = P\left(\boldsymbol{S}^t = \boldsymbol{s}^t | O^{1:T}, \lambda\right) = \frac{P\left(\boldsymbol{O}^{1:T} | \boldsymbol{S}^t = \boldsymbol{s}^t, \lambda\right) P\left(\boldsymbol{S}^t = \boldsymbol{s}^t\right)}{P\left(\boldsymbol{O}^{1:T} | \lambda\right)} \quad (7)$$

where $P\left(\boldsymbol{S}^t = \boldsymbol{s}^t | \boldsymbol{O}^{1:T}, \lambda\right)$ means the probability of $\boldsymbol{S}^t$ being in state $\boldsymbol{s}^t$ at time t, given $\boldsymbol{O}^{1:T}$ and the model parameter λ.

Then, using the variables $\{\alpha^t(\boldsymbol{s}^t), \beta^t(\boldsymbol{s}^t)\}$ in Eq. (6), the $r^t(\boldsymbol{s}^t)$ can be expressed as:

$$r^t(\boldsymbol{s}^t) = \frac{\alpha^t(\boldsymbol{s}^t) \beta^t(\boldsymbol{s}^t)}{\sum_{\boldsymbol{s}^t} \alpha^t(\boldsymbol{s}^t) \beta^t(\boldsymbol{s}^t)} \quad (8)$$

Finally, the most likely state $\boldsymbol{S}^t$ at time t can be calculated by solving the following optimization problem. And the entire process is described in Algorithm 1.

$$\boldsymbol{S}^t = \arg\max_{\boldsymbol{s}^t} \left(r^t(\boldsymbol{s}^t)\right) \quad (9)$$

Algorithm 1. Data training and probabilistic inference of the DIM.

Data training:
1. Initialize λ^0, convergence threshold ε and the observation sequence
   $\boldsymbol{O} = \{\boldsymbol{o}^t = o_i^t\}(t = 1:T; i = 1,2,3,4)$
2. **for** $j = 1:n$ **do**
3.   Calculate the expectation: $R\left(\lambda^j | \lambda^{j-1}\right) = E\left[\log P(\boldsymbol{S}, \boldsymbol{O} | \lambda^j) | \lambda^{j-1}, \boldsymbol{O}\right]$
4.   Maximize $R\left(\lambda^j | \lambda^{j-1}\right)$ using Jensen's inequality theory
5.   Output the model parameter λ^j
6.   **if** $\left|\log P\left(\boldsymbol{O} | \lambda^j\right) - \log P\left(\boldsymbol{O} | \lambda^{j-1}\right)\right| < \varepsilon$ **then**
7. $\lambda \leftarrow \lambda^j$
8.   **end if**
9. **end for**

Probabilistic inference:
10. Initialize λ and the observation sequence
   $\boldsymbol{O} = \{\boldsymbol{o}^t = o_i^t\}(t = 1:T; i = 1,2,3,4)$
11. **for** $t = 1:T$ **do**
12.   Calculate the forward variable: $\alpha^t(\boldsymbol{s}^t) = P\left(\boldsymbol{O}^{1:t}, \boldsymbol{S}^t = \boldsymbol{s}^t | \lambda\right)$
13.   Calculate the backward variable: $\beta^t(\boldsymbol{s}^t) = P\left(\boldsymbol{O}^{t+1:T} | \boldsymbol{S}^t = \boldsymbol{s}^t, \lambda\right)$
14.   Calculate the interim variable: $r^t(\boldsymbol{s}^t) = \frac{\alpha^t(\boldsymbol{s}^t) \beta^t(\boldsymbol{s}^t)}{\sum_{\boldsymbol{s}^t} \alpha^t(\boldsymbol{s}^t) \beta^t(\boldsymbol{s}^t)}$
15.   Calculate the most likely state: $\boldsymbol{S}^t = \arg\max_{\boldsymbol{s}^t} \left(r^t(\boldsymbol{s}^t)\right)$
16. **end for**
17. Output the most likely state $\boldsymbol{S}^t$

---

4. Trajectory prediction module

The purpose of this module is to make probabilistic trajectory predictions of the predicted vehicle, depending on the recognized probabilities of the driving intention and the motion characteristics from the DIM. This section will mainly focus on three aspects: vehicle model-based prediction, TPM building, and probabilistic model prediction. Next, we will introduce them in detail.

4.1. Vehicle model-based prediction

As mentioned before, the vehicle model-based method has advantages in short-term predictions. Here, we will use the CTRA kinematic vehicle model [9] to obtain the support points as the inputs to the trajectory prediction module, including the filtered historical and future trajectory points.

First, the state space $\boldsymbol{s}(t)$ and state transition expression can be expressed as follows:

$$\boldsymbol{s}(t) = (x, y, v, a, \theta, \omega)^T$$
$$\boldsymbol{s}(t + \Delta t) = \boldsymbol{s}(t) + \Delta \boldsymbol{s}(t) \quad (10)$$

where (x, y) mean the longitudinal and lateral positions of the vehicle, respectively; (v, a) represent the velocity and acceleration of the vehicle, respectively, in the driving direction; and (θ, ω) denote the rotation angle and yaw rate of the vehicle, respectively. Besides, Δt refers to the running period, which is consistent with the data sample frequency, and $\Delta \boldsymbol{s}(t)$ can be obtained by Eq. (11):

$$\begin{pmatrix} \frac{v + a\Delta t}{\omega} \sin(\theta + \omega\Delta t) - \frac{v}{\omega}\sin\theta + \frac{a}{\omega^2}(\cos(\theta + \omega\Delta t) - \cos\theta) \\ \frac{v}{\omega}\cos\theta - \frac{v + a\Delta t}{\omega}\cos(\theta + \omega\Delta t) + \frac{a}{\omega^2}(\sin(\theta + \omega\Delta t) - \sin\theta) \\ a\Delta t \\ 0 \\ \omega\Delta t \\ 0 \end{pmatrix}$$
$$(11)$$

To address the model uncertainty, the UKF algorithm is used [38]. The state and observation equations with uncertainty can be described as follows:

$$\boldsymbol{s}(t + \Delta t) = w(t) + q(t)$$
$$\boldsymbol{g}(t) = h(t) + r(t) \quad (12)$$

where $w(\cdot)$ is the motion function; $q(\cdot)$ is the system noise function, which is defined here as Gaussian noise; $\boldsymbol{g}$ is the observation space; $h(\cdot)$ is the observation function; and $r(\cdot)$ is the observation noise.

Then, the filtered historical points $\{\boldsymbol{x}_{\text{hf}} = x^{1:t}, \boldsymbol{y}_{\text{hf}} = y^{1:t}\}$ and the predicted future points $\{\boldsymbol{x}_{\text{sf}} = x^{t:t+\Delta t}, \boldsymbol{y}_{\text{sf}} = y^{t:t+\Delta t}\}$ can be calculated by iterating the state equation. Note that t is the current time; Δt is the duration time of a short-term prediction; lower case hf means historical points sequence; and sf means future short-term points sequence. Besides, the above method is suitable for lateral movement prediction, such as lane changing. Since the prediction of longitudinal movement is simple, we choose the CA model with KF to generate the points $\{\boldsymbol{x}_{\text{hf}}, \boldsymbol{y}_{\text{hf}}\}$ and $\{\boldsymbol{x}_{\text{sf}}, \boldsymbol{y}_{\text{sf}}\}$. Sometimes, the rotation angle and angular velocity of the predicted vehicle are difficult to obtain, and we can use $\{\boldsymbol{x}_{\text{hf}}, \boldsymbol{y}_{\text{hf}}\}$ to calculate them indirectly.

4.2. TPM building

Here, we will build the TPM for each kind of driving intention and motion characteristic. For example, given the vehicle trajectory dataset $D = \left\{\left(\boldsymbol{x}_i = x_i^{1:T_i}, \boldsymbol{y}_i = y_i^{1:T_i}\right), i = 1:N\right\}$ under one particular intention and characteristic (where N means the number of trajectories and T_i denotes the length of the ith trajectory), we can express a distribution $P(f = \{f_x, f_y\} | D)$ with the assumption of $\{x = f_x(t), y = f_y(t)\}$. Then, based on a GP [39], the TPM can be built with the distribution defined over parameters at a finite time, which consists of the mean vector $\boldsymbol{u}$ and the covariance matrix $\boldsymbol{K}$:

$$f \sim \mathcal{N}(\boldsymbol{u}, \boldsymbol{K}), \quad \boldsymbol{u} = \{m(t), \ t = 1:T\}$$
$$\boldsymbol{K} = \{\kappa(t_i, t_j), \ i, j = 1:N\} \quad (13)$$

where the symbol f and $\mathcal{N}$ denote the function of vehicle trajectory and Gaussian distribution separately; $m(\cdot)$ is the mean function and

κ is a definite positive kernel representing the dependency between function values at times t_i and t_j.

Since the observed data may have noise, we denote the covariance matrix with noise as $\boldsymbol{K}' = \boldsymbol{K} + \sigma_n^2 \boldsymbol{I}_N$, where σ_n is the standard noise deviation and $\boldsymbol{I}_N$ is a unit matrix. For the trajectories derived from lane change scenarios in the highway, the mean and covariance functions can be expressed as follows:

$$
\begin{aligned}
m(t) &= a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \\
\kappa(t, t') &= \sigma_f^2 \exp\left(-\frac{1}{2\gamma^2}(t - t')^2\right) + \sigma_n^2 \delta_{t,t'}
\end{aligned}
\tag{14}
$$

where $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5\}$ are the parameters of the mean function represented as a quintic polynomial, whose optimized curve can continuously smoothen the vehicle trajectory, and the speed and acceleration of vehicle motion are continuous; $\boldsymbol{k} = \{\ell, \sigma_f, \sigma_n\}$ are the parameters of the covariance function where the squared exponential kernel is employed for its good smoothing performance; and $\delta_{t,t'}$ is the Kronecker delta.

Similarly, for the trajectories derived from lane-keeping scenarios, the mean function can be changed to linear. After establishing the TPM, we need to determine the model parameter $\theta = \{\boldsymbol{a}, \boldsymbol{k}\}$. It is crucial to learn suitable values of θ because the final prediction accuracy depends on the properness of the TPM parameters directly. To make the prediction results more reliable and reasonable, we learn the parameters from the training data D instead of defining θ manually.

Next, we will introduce the process of parameter learning. Taking the lateral movement as an example, first, the corresponding log marginal likelihood $L(\theta)$ can be expressed as follows:

$$
\begin{aligned}
L(\theta) = &-\frac{1}{2}\log|\boldsymbol{K} + \sigma_n^2\boldsymbol{I}_N| - \frac{N}{2}\log(2\pi) \\
&-\frac{1}{2}(y - \boldsymbol{u})^{\mathrm{T}}(\boldsymbol{K} + \sigma_n^2\boldsymbol{I}_N)^{-1}(y - \boldsymbol{u})
\end{aligned}
\tag{15}
$$

Then, $\{\theta_a, \theta_k\}$, the parameters of $\{\boldsymbol{a}, \boldsymbol{k}\}$, can be obtained by optimizing $L(\theta)$ using the partial derivatives of Eq. (14):

$$
\begin{aligned}
\frac{\partial L(\theta)}{\partial \theta_a} &= -\frac{1}{2}(y - \boldsymbol{u})^{\mathrm{T}}(\boldsymbol{K} + \sigma_n^2\boldsymbol{I}_N)^{-1}\frac{\partial \boldsymbol{u}}{\partial \theta_k} \\
\frac{\partial L(\theta)}{\partial \theta_k} &= \frac{1}{2}\mathrm{trace}\left((\boldsymbol{K} + \sigma_n^2\boldsymbol{I}_N)^{-1}\frac{\partial \boldsymbol{u}}{\partial \theta_k}\right) + \\
&\quad \frac{1}{2}(y - \boldsymbol{u})^{\mathrm{T}}\frac{\partial \boldsymbol{K}}{\partial \theta_k}(\boldsymbol{K} + \sigma_n^2\boldsymbol{I}_N)^{-1}\frac{\partial \boldsymbol{K}}{\partial \theta_k}(y - \boldsymbol{u})
\end{aligned}
\tag{16}
$$

Finally, we can apply an optimization method using conjugate gradients to effectively figure out the optimal model parameters. As for the longitudinal movement, the model selection is similar to the lateral case.

4.3. Probabilistic prediction of the model

Here, we will first introduce the typical prediction based on the TPM. Considering the lateral trajectory prediction case, we denote $\boldsymbol{y}$ as the known observation points and $\boldsymbol{y}^*$ as the unknown future points. The joint probability density can be expressed as:

$$
\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{y}^* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{u}_y \\ \boldsymbol{u}_y^* \end{pmatrix}, \begin{pmatrix} \boldsymbol{K}_y & \boldsymbol{K}_y^* \\ \boldsymbol{K}_y^{*\mathrm{T}} & \boldsymbol{K}_y^{**} \end{pmatrix}\right)
\tag{17}
$$

where $(\boldsymbol{u}_y, \boldsymbol{K}_y)$ refer to the mean and variance of $\boldsymbol{y}$; $(\boldsymbol{u}_y^*, \boldsymbol{K}_y^{**})$ represent the mean and variance of $\boldsymbol{y}^*$; and $\boldsymbol{K}_y^*$ means the covariance between $\boldsymbol{y}$ and $\boldsymbol{y}^*$.

Then, we incorporate the support points of the vehicle based on prediction. The new observation points will be replaced by $\boldsymbol{y}' = $

$\boldsymbol{y}_{\mathrm{hf}} + \boldsymbol{y}_{\mathrm{sf}}$. Moreover, for the longitudinal prediction, the new observation points will be replaced by $\boldsymbol{x}' = \boldsymbol{x}_{\mathrm{hf}} + \boldsymbol{x}_{\mathrm{sf}}$. Afterward, the conditional prediction probability of $\boldsymbol{y}^*$ given $\boldsymbol{y}'$ can be presented by:

$$
p(\boldsymbol{y}^*|\boldsymbol{y}') = \mathcal{N}\left(\boldsymbol{u}_y^* + \boldsymbol{K}_y^{*\mathrm{T}}\boldsymbol{K}_y^{-1}(\boldsymbol{y}' - \boldsymbol{u}_y), \quad \boldsymbol{K}_y^{**} - \boldsymbol{K}_y^{*\mathrm{T}}\boldsymbol{K}_y^{-1}\boldsymbol{K}_y^*\right)
\tag{18}
$$

Finally, the future predicted points in the long term can be obtained from the mean function $\boldsymbol{u}_y^* + \boldsymbol{K}_y^{*\mathrm{T}}\boldsymbol{K}_y^{-1}(\boldsymbol{y}' - \boldsymbol{u}_y)$ in Eq. (18), and the corresponding prediction uncertainty $\boldsymbol{K}_y^{**} - \boldsymbol{K}_y^{*\mathrm{T}}\boldsymbol{K}_y^{-1}\boldsymbol{K}_y^*$ can be described by the covariance function in Eq. (18). Since the predicted point at each moment obeys a Gaussian distribution. The lateral uncertainty of the point can be represented by the values of the related variance. Similarly, when calculating the longitudinal trajectory prediction, we can use the same method to obtain the conditional prediction probability $p(\boldsymbol{x}^*|\boldsymbol{x}')$. Finally, by calculating $p(\boldsymbol{x}^*|\boldsymbol{x}')$ and $p(\boldsymbol{y}^*|\boldsymbol{y}')$, we can obtain the trajectory prediction points $(\boldsymbol{x}^{t:T}, \boldsymbol{y}^{t:T})$. The entire vehicle trajectory prediction process is shown in Algorithm 2.

Algorithm 2. Probabilistic vehicle trajectory prediction.

Input: $(x, y, v, a)^{1:t}$, driving intention probability $\boldsymbol{P}_{\mathrm{Intent}}^t$, and motion characteristic probability $\boldsymbol{P}_{\mathrm{Mcharact}}^t$

Output: prediction points $(\boldsymbol{x}^{t:T}, \boldsymbol{y}^{t:T})$

1 **if** the probability of lane keeping $P_{\mathrm{Lk}}^t = \max\{\boldsymbol{P}_{\mathrm{Intent}}^t\}$ **then**

2 Calculate the points based on CV and KF: $\{\boldsymbol{x}_{\mathrm{hf}}, \boldsymbol{y}_{\mathrm{hf}}\}, \{\boldsymbol{x}_{\mathrm{sf}}, \boldsymbol{y}_{\mathrm{sf}}\}$

3 **else**

4 Calculate the points based on CTRA and PF: $\{\boldsymbol{x}_{\mathrm{hf}}, \boldsymbol{y}_{\mathrm{hf}}\}, \{\boldsymbol{x}_{\mathrm{sf}}, \boldsymbol{y}_{\mathrm{sf}}\}$

5 **end if**

6 Output the input of TPM: $\boldsymbol{y}' = \boldsymbol{y}_{\mathrm{hf}} + \boldsymbol{y}_{\mathrm{sf}}, \boldsymbol{x}' = \boldsymbol{x}_{\mathrm{hf}} + \boldsymbol{x}_{\mathrm{sf}}$

7 Calculate the most likely characteristic: $P_{\mathrm{max\_Charact}}^t = \max\{\boldsymbol{P}_{\mathrm{Mcharact}}^t\}$

8 Calculate the most likely intention: $P_{\mathrm{max\_Intent}}^t = \max\{\boldsymbol{P}_{\mathrm{Intent}}^t\}$

9 Select θ to match $P_{\mathrm{max\_Intent}}^t$ and $P_{\mathrm{max\_Charact}}^t$

10 Calculate trajectory prediction points $(\boldsymbol{x}^{t:T}, \boldsymbol{y}^{t:T})$ using $p(\boldsymbol{x}^*|\boldsymbol{x}')$ and $p(\boldsymbol{y}^*|\boldsymbol{y}')$

---

5. Experiments

Since the straight driving and lane change are the situations we encounter most frequently in the process of naturalistic driving, in this section, we will validate the developed DIM and TPM in the widely applicable highway scenario. According to the contributions proposed in this paper, we will validate the superiorities of our developed DIM. Then, the overall performance of our final TPM will be shown from different aspects.

First, the data processing method will be introduced, including the driving intention calibration and motion characteristic classification. Then, the inference probability of the DIM and the evaluation of the DIM will be presented. Finally, the trajectory prediction results of the TPM will be shown and analyzed. Besides, a comprehensive evaluation and comparison will be made to further demonstrate the effectiveness of our proposed method.

5.1. Data processing

Here, we use a large-scale naturalistic vehicle trajectory dataset from German highways called the highD dataset [40] to verify our proposed method for long-term trajectory prediction. The highD dataset contains 16.5 hours of measurements from six locations with 110 000 vehicles. Moreover, it records 5600 complete lane-

changing scenarios. Compared to the commonly used Next Generation SIMulation (NGSIM) dataset, the vehicles in highD dataset have a more reasonable speed distribution, which is more close to our real driving environment. Thus, it is appropriate to perform model training or learning for our built DIM and TPM. The training and test sets are mainly including three parts: the calibration of driving intentions; the calibration of motion characteristics calibration; and the determination of observation variables. For the test sets, they only include observation variables.

First, according to the scene of highway, driving intentions will be divided into three categories: left lane change, lane keeping, and right lane change. The start time identification of the lane-change intention is defined when the lateral offset of the vehicle exceeds 0.1 m relative to the average lateral position of the vehicle. For the calibration of driving intention, we define the trajectory before the start time as the straight driving phase, and the trajectory afterwards as the lane change phase.

Then, we will make a calibration of the motion characteristics. Since it is difficult to directly calibrate this abstract variable, we employ a sequence clustering algorithm, which is based on the k-means cluster method, to classify the motion characteristics [41]. Generally, the cluster number C is hard to determine since it is not a probabilistic model and there is no likelihood. Hence, we use the following mean square error (MSE) to solve this problem:

$$MSE(Q, C) = \frac{1}{|Q|} \sum_{\boldsymbol{c}_i \in Q} \| \boldsymbol{c}_i - \hat{\boldsymbol{c}}_i \|^2 \tag{19}$$

where Q represents the trajectory samples; $\boldsymbol{c}_i$ is the ith trajectory sample in Q; and the centroids $\hat{\boldsymbol{c}}_i$ can be derived from:

$$\hat{\boldsymbol{c}}_i = \boldsymbol{u}_{d_i}, \quad d_i = \arg\min_c \| \boldsymbol{c}_i - \boldsymbol{u}_c \|^2 \tag{20}$$

where $\boldsymbol{u}_c$ means the cth cluster center, $\boldsymbol{u}_{d_i}$ denotes the optimal cluster (i.e., the d_ith cluster center).

Subsequently, we use the lateral acceleration sequences to carry out the clustering of the motion characteristics, and the results of the left lane change case are shown below. With the illustrated sequence data of lateral acceleration in Fig. 4(a), we can try different values of K using the clustering algorithm and calculate the MSE separately. According to Fig. 4(b), the number of clusters can be set to 3, which corresponds to the knee point in the error curve. Therefore, the motion characteristic can be identified by three clustering centroids, as shown in Fig. 4(c). A difference can be seen in the drop between the peaks and troughs of these curves. The blue curve represents motion characteristic 1; the yellow curve represents motion characteristic 2; and the orange curve represents motion characteristic 3. Similarly, we can obtain the clustering centroids in the case of right lane change, which is shown in Fig. 4(d).

Finally, we will make the determination of observation variables according to the different requirements of models. We can obtain the corresponding training and test sets for the TPM, which contains the position sequences (x,y) under different kinds of driving intentions and motion characteristics. However, to prepare the training and test sets for the DIM, we need to define the motion variables of nodes $\{O_1, O_3, O_4\}$ in the DIM. Since the motion characteristic is mainly determined by lateral and longitudinal acceleration (a_x, a_y) and the driving intention is primarily characterized by lateral position y, lateral velocity v_y, and lateral acceleration a_y, the related nodes are set to $\{O_1 = (a_x, a_y), O_3 = v_y, O_4 = (y, a_y)\}$. Note that this assignment is the empirical result of trying a few different combinations. Finally, we can obtain the state and observation sequences ($\boldsymbol{S}, \boldsymbol{O}$) for the DIM training.

5.2. Results analysis and evaluation of the DIM

With the results of data processing, the DIM parameters can be learned from the training set of DIM. Then, we can perform probabilistic model inference using the method described in Section 2. Subsequently, the test set is utilized to show the inference performance. Here, we illustrate the outcomes under the cases of left lane change and right lane change, which can be seen in Figs. 5 and 6, respectively. The dotted lines in color refer to the probabilities of the driving intention or motion characteristics over time, and the solid black line indicates the true lateral position or lateral acceleration of the vehicle. In addition, the probability of the driving intention is depicted in Figs. 5(a) and 6(a), and the probability of the motion characteristics is presented in Figs. 5(b) and 6(b).

Next, we will quantitatively analyze the performance of the DIM since it acts as the essential input to the TPM and has a significant influence on the trajectory prediction accuracy. Since the colored lines in Figs. 5 and 6 do not fluctuate frequently when the lane change occurs, we will only evaluate the DIM using the accuracy rate. If the probability of one specific sequence exceeds 90%, we define it as a correct case. To further demonstrate our designed DIM's performance, we make comparisons using the traditional models which includes HMM, HMM with mixture of Gaussians output (GMM–HMM) [34,42], and our previously built driving characteristic and intention estimation (DCIE) model [37]. The statistical results of the DIM are shown in Table 1. We can see that the accuracy rate of the DIM for the driving intention and motion characteristics reaches 94.5% and 92.3%, respectively, while the results of the DCIE model are 92.4% and 90.1%. Moreover, both the HMM and GMM–HMM have lower accuracy than our model.

In order to further demonstrate the improvement in the accuracy rate of the DIM for driving intention recognition after introducing traffic rules, we tested the model without node O_2^t in Fig. 3. The accuracy rate of this model for driving intention recognition is 93.87%, and the DIM improved by 0.67% over it. Therefore, the performance of our proposed model that introduces traffic rules is promoted.

In conclusion, our designed DIM model can effectively infer the probabilities of driving intention or motion characteristics. From the visualized results, our DIM has good response properties, and the advantages in inference ability and accuracy can be verified compared to the other models.

5.3. Results analysis and comparison of the TPM

As mentioned before, each TPM corresponds to one specific driving intention and motion characteristic. Thus, given the training sets of the TPM, we can learn the model parameters of each TPM from the data.

There are three main procedures to make long-term trajectory predictions via our proposed method. First, through the DIM, we can determine the start time when the probability of a driving intention exceeds 90%. Afterward, we can identify the most likely driving intention and motion characteristics at that time. Then, we choose the right TPM model according to the most likely probability. The final probabilistic trajectory prediction can be made with the chosen TPM and the corresponding vehicle model.

In the case of left lane change, the trajectory prediction results of our proposed method are shown in Fig. 7. In Fig. 7(a), we show the outcomes of the prediction at one second ahead of the start time. We should note that the observed points in Fig. 7 are the filtered points from the vehicle model-based prediction module. Since the probability of a left lane change intention is over 90% at the start time, the future long-term trajectory can be forecasted via the TPM, as shown in Fig. 7(b). Furthermore, we can make the
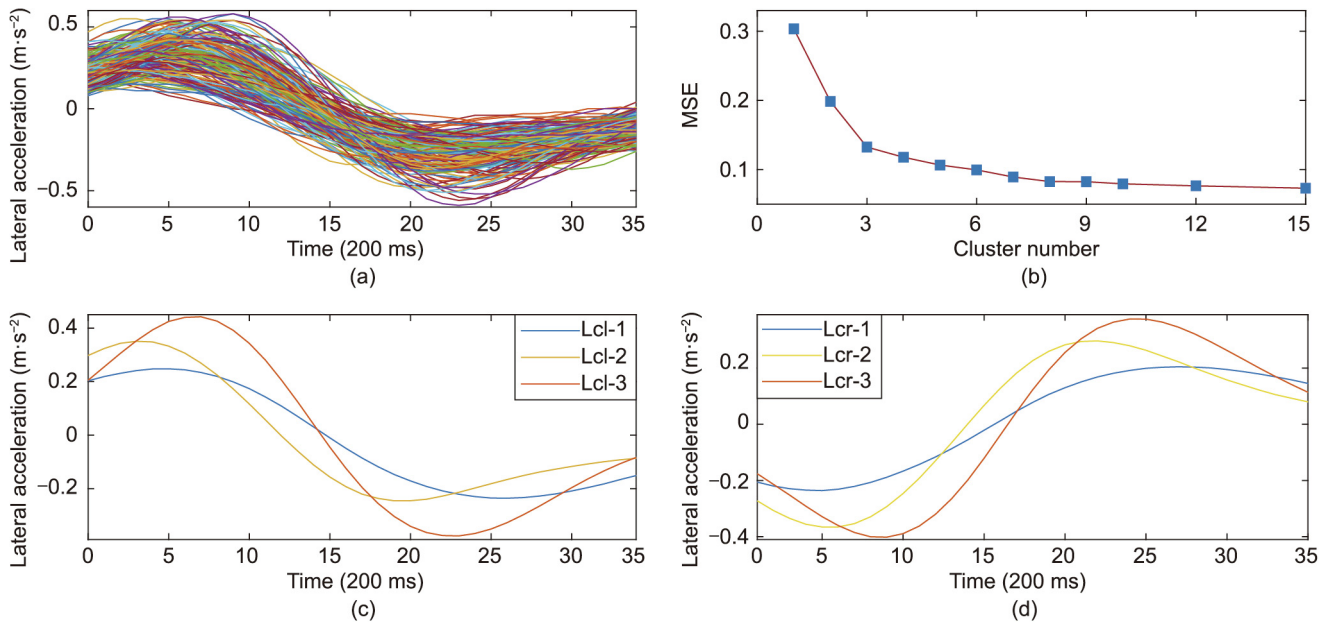
Fig. 4. (a) The sequence data of the lateral acceleration in the left lane change case; (b) the change in the MSE over different cluster numbers; (c) the clustering centroids of the motion characteristics in the left lane change case (Lcl-1, Lcl-2, and Lcl-3 respectively represent the motion characteristics 1, 2, and 3 in the process of left lane change); (d) the clustering centroids of the motion characteristics in the right lane change case (Lcr-1, Lcr-2, and Lcr-3 respectively represent the motion characteristics 1, 2, and 3 in the right lane change process).
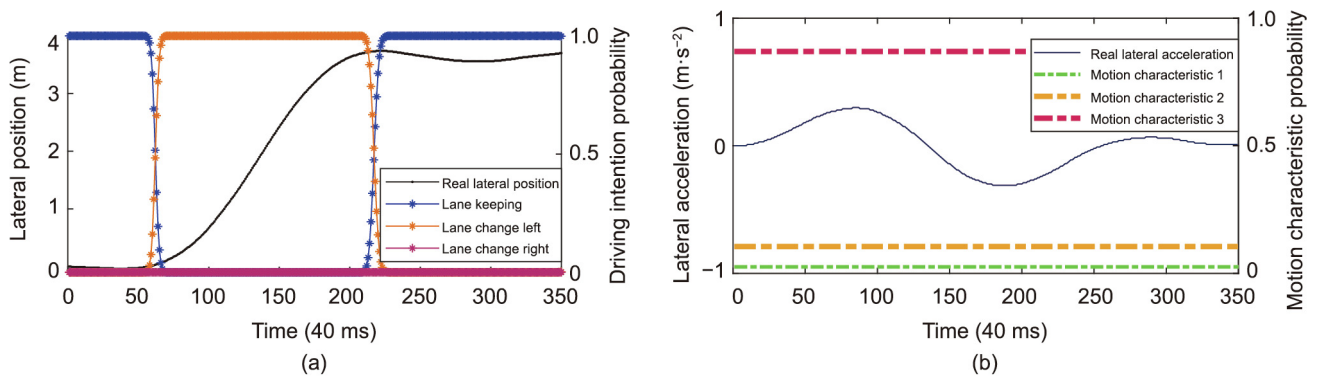
Fig. 5. The probability results of the DIM in the left lane change case. (a) The probability of the driving intention; (b) the probability of the motion characteristic.
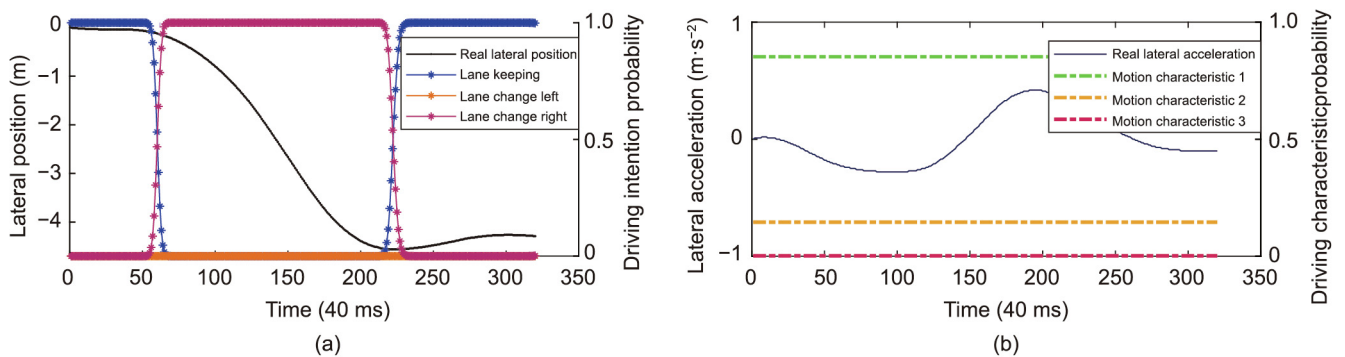
Fig. 6. The probability results of the DIM in the right lane change case. (a) The probability of the driving intention; (b) the probability of the motion characteristic.

multimodal trajectory prediction (shown in Fig. 7(a)) via our proposed method because the probabilities of left lane change and lane-keeping intentions are both below 50% at that time. One thing to note is that we should choose the CV model when dealing with the lane-keeping case.

In addition, one phenomenon that can be found in Fig. 7 is that the quality of the TPG-based prediction will be improved with the extended length of the observed sequence. Since our TPM is based on the GP, it is easy to understand that the model performance will be improved when additional reliable known points are provided.

Table 1
Accuracy rate of different models.

Accuracy rate	Model			
	HMM (%)	GMM–HMM (%)	DCIE (%)	DIM (ours) (%)
Driving intention	86.10	90.40	92.40	94.50
Motion characteristic	81.90	89.20	90.10	92.30

Fortunately, based on the support points of vehicle model-based prediction, we can not only obtain more reliable filtered points but also lengthen the known points using the short-term prediction points.

In Fig. 8, the uncertainty regions of the predicted trajectories based on the historical observation sequences are shown, which are represented by the purple ellipse with the horizontal axis representing the longitudinal uncertainty and the vertical axis representing the lateral uncertainty. According to the theoretical part of the TPM in Section 4, the predicted point at each moment obeys a Gaussian distribution, and the longitudinal and lateral variances of each point can be obtained according to Eq. (18). The values of variances are used to determine the length of axes in the ellipse. As can be seen in Fig. 8, the ellipse gradually increases since the uncertainty rises over time. Moreover, the true future trajectories in both Figs. 8 and 9 are always enveloped in the uncertainty regions, which indicates that our method is able to provide a reasonable description of the range of prediction uncertainty.

To achieve a comprehensive analysis of our proposed method, we use two trajectory prediction methods for comparison. The first is vehicle model-based prediction, which has been introduced frequently in this paper, and the second prediction method is based on our designed DIM and TPM without using the vehicle model. Actually, the second approach is a maneuver-based method. Here, the comparative results under the left lane change and right lane change intentions are illustrated in Figs. 9(a) and (b), respectively.

As shown in Fig. 9, the vehicle model-based prediction (the blue line) has a high accuracy in the short term but has a low accuracy in the long term. In contrast, the maneuver-based prediction (the orange line) has good performance in the long term and is capable of guaranteeing overall accuracy. For our proposed method, the prediction accuracy during the whole process is further improved by incorporating the vehicle model. Additionally, the TPM has an advantage in uncertainty description using the covariance functions. As depicted in Fig. 9, the uncertainty region generated by our method is more reasonable and reliable than that generated by the first method.

To evaluate the prediction performance, two evaluation criteria, the average displacement error (ADE) and the final displacement error (FDE)—are used to analyze the results [43]. In Table 2, comparative evaluations of the ADE/FDE using different methods are presented. In general, the tendency of the ADE/FDE during the whole prediction corresponds to the analysis of the three methods, which is mentioned above. The best results are in bold in the table.
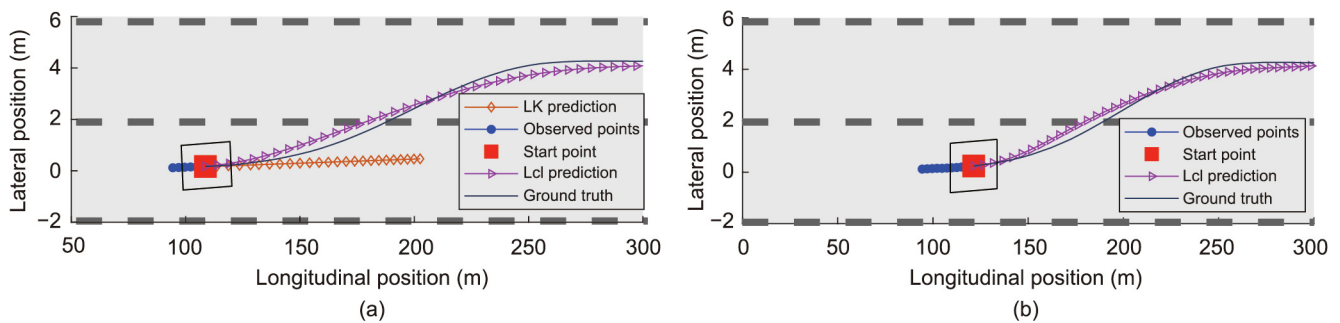
Fig. 7. Trajectory prediction of our proposed method with the intention of left lane change. (a) The results of the trajectory prediction at one second ahead of the start time; (b) the results of the trajectory prediction at the start time. The red solid square refers to the start point, and the quadrilateral refers to the predicted vehicle; the blue lines denote the observed points, and the black lines denote the true future trajectory; the red lines denote the predicted trajectory of left lane change, and the orange line denotes the predicted trajectory of lane keeping. LK: lane keeping; Lcl: lane change left.
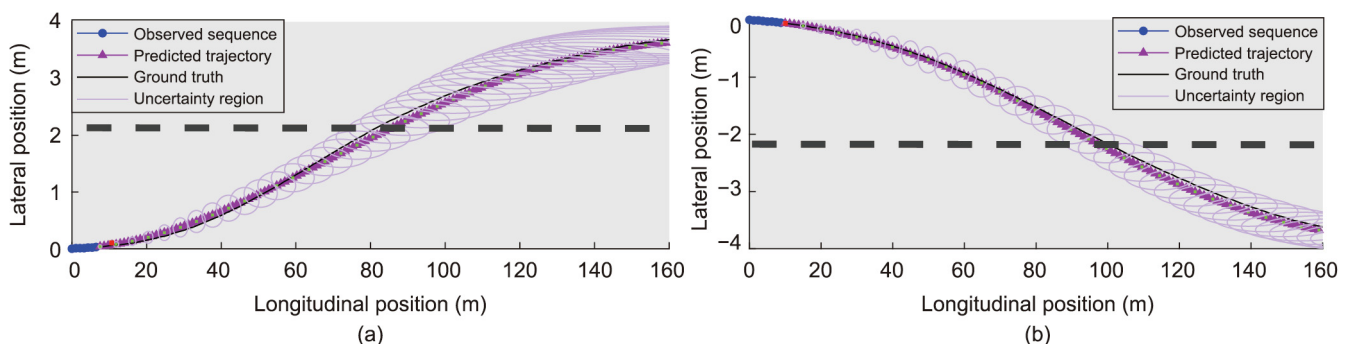
Fig. 8. Trajectory prediction of our proposed method with the uncertainty region. (a) The case of left lane change; (b) the case of right lane change. The red solid square refers to the start point; the blue lines denote the observed points; the black lines denote the true future trajectory; the red lines denote the predicted trajectory. Ellipses in purple represent the prediction uncertainty derived from the covariance function in Eq. (18); the horizontal axis refers to the longitudinal uncertainty; and the vertical axis indicates the lateral uncertainty.
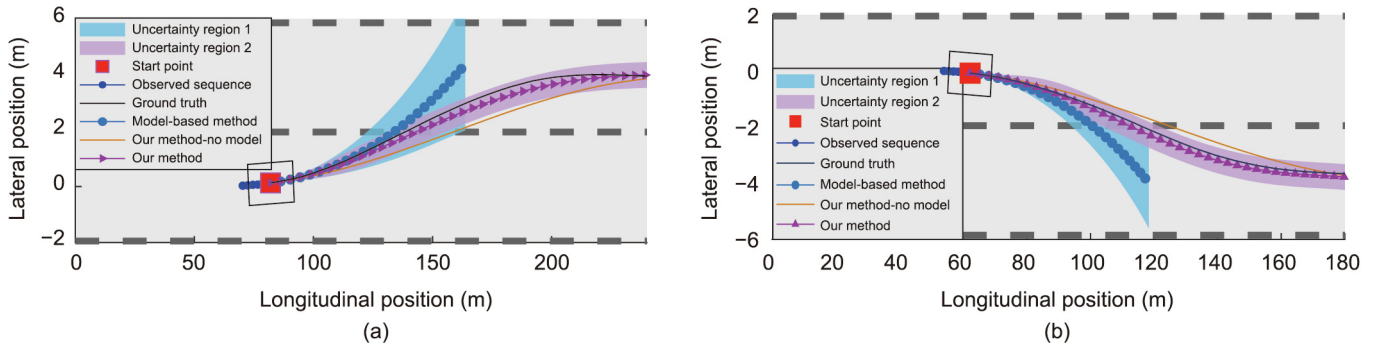
Fig. 9. Comparisons of different vehicle trajectory prediction methods. (a) The case of a left lane change; (b) the case of a right lane change. The blue shaded part indicates the prediction uncertainty region of the model-based method, the red shaded part indicates that of our method, and the orange line refers to the results of our method without using the vehicle model.

In Section 1, we have mentioned three types of trajectory prediction methods: vehicle model-based prediction, maneuver-based prediction, and deep learning-based prediction. In Table 2, we will compare a total of three methods. The first corresponds to the vehicle model-based prediction, and the second and the third correspond to the maneuver-based prediction. In Table 3 [3,28–30], we will show the comparisons between our method and the deep learning-based prediction.

In Table 2, the second one is our proposed method without incorporating the vehicle kinematic model. From Table 2, we can see that the ADEs of our method without model are 0.204 and 0.367 m for the prediction times of one and two seconds, which are 4.08% and 27.43% higher than our method's results of 0.196 and 0.288 m. In addition, for the prediction times of fourth and fifth seconds, the ADEs of our method are 0.972 and 1.261 m, respectively, which are 50.03% and 60.33% lower than the first results of 1.945 and 3.179 m, and 7.95% and 9.99% lower than the second results of 1.056 and 1.401 m. Similarly, compared with the other methods, the FDEs of our method are lower. Therefore, our proposed method improves the prediction accuracy both in the short and long term, compared to our method without vehicle model. Meanwhile, our method enhances the long-term prediction accuracy to a great extent compared to the model-based method. Thus, integrating the short-term prediction results of the kinematic vehicle model, the TPM can improve the precision of the entire prediction process.

The above analysis demonstrates the effectiveness of our proposed framework for long-term trajectory prediction. To further illustrate the benefits of our approach, in Table 3 [3,28–30], we show comparisons of other state-of-the-art methods applied in the public highD dataset. The best results for the ADE are labeled in bold, and the second best results are underlined. It should be noted that the results of the sequence-to-sequence method using convolutional social pooling [28] are derived from Ref. [30]. We can see that this method has high accuracy in the long term (the prediction duration is more than 2 s) but low accuracy in the short term (the prediction duration is 1 s). However, the second and fourth methods have excellent performance in the beginning but bad performance at the final stage, which accounts for the difficulty of guaranteeing the prediction accuracy both in the short term and long term. Although the third method based on the LSTM encoder–decoder model with the lane attention mechanism achieves the optimal results in the fourth and fifth seconds (one possible reason is the difference in the test set), the accuracy within the first three seconds is lower than our method.

To further validate the advantage of our method in the overall prediction time domain, we define a comprehensive evaluation index (CEI): the average value of ADE over the entire prediction period. The CEI can be expressed as follows:

$$\text{CEI} = \frac{1}{T}\sum_{i=1}^{T}\text{ADE}(i) \tag{21}$$

Table 2
The ADE/FDE (m) comparisons of different methods.

Method	Prediction horizon				
	1 s	2 s	3 s	4 s	5 s
Mode-based method	**0.184/0.204**	0.306/0.548	1.163/1.724	1.945/2.751	3.179/4.363
Our method without model	0.204/0.349	0.367/0.503	0.629/0.891	1.056/1.322	1.401/1.771
Our method	0.196/0.237	**0.288/0.415**	**0.502/0.667**	**0.972/1.138**	**1.261/1.474**

Table 3
The ADE/CEI (m) comparisons of other state-of-the-art methods.

Method	Prediction horizon					CEI
	1 s	2 s	3 s	4 s	5 s	
CS-LSTM [28]	1.18	1.16	1.22	1.44	1.77	1.354
L-RRNN [29]	<u>0.22</u>	<u>0.63</u>	1.31	2.22	3.38	1.552
LSTM encoder–decoder model-lane attention [30]	0.69	0.66	<u>0.69</u>	**0.83**	**1.09**	<u>0.792</u>
3D CNN-LSTM [3]	0.26	0.65	1.20	1.91	2.74	1.352
Ours	**0.18**	**0.29**	**0.50**	<u>0.97</u>	<u>1.23</u>	**0.634**

CEI: comprehensive evaluation index; CS-LSTM: LSTM with convolutional social pooling; L-RRNN: relational RNN with per lane embedding; 3D CNN-LSTM: LSTM with 3D convolutional kernel layers.

where T is the total prediction time and ADE(i) means the value of ADE at the ith prediction time.

In Table 3, [3,28–30] it can be seen that different methods have an obvious difference in the prediction performance under different prediction durations. The CEI can effectively measure the comprehensive prediction effectiveness of a certain prediction method in the short and long term. We can see that the CEI value of the method that dominates when the prediction time is 5 s is 0.792, while our method has the best CEI value of 0.634. Our method improves by 19.5% over it. Therefore, our designed method has clear advantages over others in the whole prediction time.

In summary, our proposed method can obtain good prediction performance in both the short term and long term. Compared to the other methods, the effectiveness and reliability of our method are superior. It is capable of ensuring higher prediction accuracy and describing the prediction uncertainty more appropriately during the whole prediction process.

6. Conclusions

In this paper, we proposed an integrated probabilistic architecture to predict the long-term trajectory of the surrounding vehicle by combining low- and high-level environmental information. This architecture consists of two novel components: a DIM based on a dynamic Bayesian network that incorporates physical movement and traffic rules and a TPM based on a GP that considers the vehicle motion characteristics and short-term prediction results together. We first use the predicted vehicle's historical data to obtain the probability of both the driving intention and motion characteristics, which also considers the basic road rules. According to the above results, the vehicle trajectory can be predicted correspondingly, which is fused with the vehicle's physical motion. In addition, the region of prediction uncertainty can be presented since our proposed framework is probabilistic.

Experiments on the public highD dataset show that the proposed architecture is effective and reliable in highway scenarios. Compared to other methods, the proposed model has the following advantages: ① an interpretable probabilistic framework to guarantee prediction accuracy and feasibility; ② the capability of handling sequential data to take advantage of high-level information consisting of the driving intention and motion characteristics; and ③ the guidance of autonomous driving systems with better situational understanding. In future work, we will extend the vehicle trajectory prediction architecture to adapt to more complex scenarios and consider employing the prediction results to estimate the collision probability. By considering the dynamic interaction between vehicles, we can provide an early warning for autonomous driving. There are also some improvements that need to be made in some aspects. Since the structure of DIM is artificially constructed based on experience, a considerable amount of time is spent on the adjustment of the model structure. For the good scalability of our probabilistic prediction framework, further refinements are still in proceeding, including the self-learning and optimization of the model structure, and the application of this approach to more complex scenarios. We are also conducting experimental work on real vehicles in Chinese highways, hoping to verify the performance of our model in Chinese highway scenarios.

Acknowledgments

Compliance with ethics guidelines

Jinxin Liu, Yugong Luo, Zhihua Zhong, Keqiang Li, Heye Huang, and Hui Xiong declare that they have no conflict of interest or financial conflicts to disclose.

References

[1] Zhou J, Zhou Y, Wang B, Zang J. Human–cyber–physical systems (HCPSs) in the context of new-generation intelligent manufacturing. Engineering 2019;5(4):624–36.
[2] Seif HG, Hu X. Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry. Engineering 2016;2(2):159–62.
[3] Krüger M, Novo AS, Nattermann T, Bertram T. Interaction-aware trajectory prediction based on a 3D spatio-temporal tensor representation using convolutional–recurrent neural networks. In: Proceedings of 2020 IEEE Intelligent Vehicles Symposium (IV); 2020 Oct 19–Nov 13; Las Vegas, NV, USA. New York City: IEEE; 2020. p. 1122–7.
[4] Huang H, Wang J, Fei C, Zheng X, Yang Y, Liu J, et al. A probabilistic risk assessment framework considering lane-changing behavior interaction. Sci China Inf Sci 2020;63(9):1–15.
[5] Lefèvre S, Vasquez D, Laugier C. A survey on motion prediction and risk assessment for intelligent vehicles. ROBOMECH J 2014;1:1.
[6] Mozaffari S, Al-Jarrah OY, Dianati M, Jennings P, Mouzakitis A. Deep learning-based vehicle behaviour prediction for autonomous driving applications: a review. IEEE Trans Intell Transp Syst 2020;23(1):33–47.
[7] Rudenko A, Palmieri L, Herman M, Kitani KM, Gavrila DM, Arras KO. Human motion trajectory prediction: a survey. Int J Robot Res 2020;39(8):895–935.
[8] Sorstedt J, Svensson L, Sandblom F, Hammarstrand L. A new vehicle motion model for improved predictions and situation assessment. IEEE Trans Intell Transp Syst 2011;12(4):1209–19.
[9] Schubert R, Richter E, Wanielik G. Comparison and evaluation of advanced motion models for vehicle tracking. In: Proceedings of 2008 11th International Conference on Information Fusion; 2008 Jun 30–Jul 3; Cologne, Germany. New York City: IEEE; 2008. p. 1–6.
[10] Lin CF, Ulsoy AG, LeBlanc DJ. Vehicle dynamics and external disturbance estimation for vehicle path prediction. IEEE Trans Control Syst Technol 2000;8(3):508–18.
[11] Kamann A, Bielmeier JB, Hasirlioglu S, Schwarz UT, Brandmeier T. Object tracking based on an extended Kalman filter in high dynamic driving situations. In: Proceedings of 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC); 2017 Oct 16–19; Yokohama, Japan. New York City: IEEE; 2017. p. 1–6.
[12] Caveney D. Stochastic path prediction using the unscented transform with numerical integration. In: Proceedings of 2007 IEEE Intelligent Transportation Systems Conference; 2007 Sep 30–Oct 3; Bellevue, WA, USA. New York: IEEE; 2007. p. 848–53.
[13] Berntorp K, Hoang T, Di Cairano S. Motion planning of autonomous road vehicles by particle filtering. IEEE Trans Intell Veh 2019;4(2):197–210.
[14] Schreier M, Willert V, Adamy J. Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems. In: Proceedings of 17th International IEEE Conference on Intelligent Transportation Systems (ITSC); 2014 Oct 8–11; Qingdao, China. New York City: IEEE; 2014. p. 334–41.
[15] Liu P, Kurt A, Özgüner Ü. Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification. In: Proceedings of 17th International IEEE Conference on Intelligent Transportation Systems (ITSC); 2014 Oct 8–11; Qingdao, China. New York City: IEEE; 2014. p. 942–7.
[16] Deo N, Rangesh A, Trivedi MM. How would surround vehicles move? A unified framework for maneuver classification and motion prediction. IEEE Trans Intell Veh 2018;3(2):129–40.
[17] Kumar P, Perrollaz M, Lefèvre S, Laugier C. Learning-based approach for online lane change intention prediction. In: Proceedings of 2013 IEEE Intelligent Vehicles Symposium (IV); 2013 Jun 23–26; Gold Coast, QLD, Australia. New York City: IEEE; 2013. p. 797–802.
[18] Streubel T, Hoffmann KH. Prediction of driver intended path at intersections. In: Proceedings of 2014 IEEE Intelligent Vehicles Symposium Proceedings; 2014 Jun 8–11; New York City: IEEE; 2014. p. 134–9.
[19] Brown K, Driggs-Campbell K, Kochenderfer MJ. Modeling and prediction of human driver behavior: a survey. 2020. arXiv:2006.08832.
[20] Xing Y, Lv C, Wang H, Wang H, Ai Y, Cao D, et al. Driver lane change intention inference for intelligent vehicles: framework, survey, and challenges. IEEE Trans Veh Technol 2019;68(5):4377–90.
[21] Wiest J, Höffken M, Kreßel U, Dietmayer K. Probabilistic trajectory prediction with Gaussian mixture models. In: Proceedings of 2012 IEEE Intelligent Vehicle Symposium; 2012 Jun 3–7; Madrid, Spain. New York City: IEEE; 2012. p. 141–6.
[22] Gao H, Zhu J, Zhang T, Xie G, Kan Z, Hao Z, et al. Situational assessment for intelligent vehicles based on stochastic model and Gaussian distributions in typical traffic scenarios. IEEE Trans Syst Man Cybern Syst. In press.

[23] Schreier M, Willert V, Adamy J. An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments. IEEE Trans Intell Transp Syst 2016;17(10):2751–66.

[24] Li J, Wu J, Sun H, Jiang Y, Deng W, Zhu B. Traffic modeling considering motion uncertainties. SAE Internat 2017.

[25] Houenou A, Bonnifait P, Cherfaoui V, Yao W. Vehicle trajectory prediction based on motion model and maneuver recognition. In: Proceedings of 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2013 Nov 3–7; Tokyo, Japan. New York City: IEEE; 2013. p. 4363–9.

[26] Xie G, Gao H, Qian L, Huang B, Li K, Wang J. Vehicle trajectory prediction by integrating physics- and maneuver-based approaches using interactive multiple models. IEEE Trans Ind Electron 2018;65(7):5999–6008.

[27] Schlenoff C, Madhavan R, Kootbally Z. PRIDE: a hierarchical, integrated prediction framework for autonomous on-road driving. In: Proceedings of 2006 IEEE International Conference on Robotics and Automation; 2006 May 15–19; Orlando, FL, USA. New York City: IEEE; 2006. p. 2348–53.

[28] Deo N, Trivedi MM. Convolutional social pooling for vehicle trajectory prediction. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop (CVPRW); 2018 Jul 18–22; Salt Lake City, UT, USA. New York City: IEEE; 2018. p. 1468–76.

[29] Messaoud K, Yahiaoui I, Verroust-Blondet A, Nashashibi F. Relational recurrent neural networks for vehicle trajectory prediction. In: Proceedings of 2019 IEEE Intelligent Transportation Systems Conference (ITSC); 2019 Oct 27–30; Auckland, New Zealand. New York City: IEEE; 2019. p. 1813–8.

[30] Yan J, Peng Z, Yin H, Wang J, Wang X, Shen Y, et al. Trajectory prediction for intelligent vehicles using spatial-attention mechanism. IET Intell Transp Syst 2020;14(13):1855–63.

[31] Mo X, Xing Y, Lv C. Interaction-aware trajectory prediction of connected vehicles CNN-LSTM networks. In: Proceedings of IECON 2020 46th Annual Conference of the IEEE Industrial Electronics Society; 2020 Oct 18–21; Singapore. New York City: IEEE; 2020. p. 5057–62.

[32] Xing Y, Lv C, Cao D. Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles. IEEE Trans Veh Technol 2020;69(2):1341–52.

[33] Xing Y, Lv C, Cao D, Lu C. Energy oriented driving behavior analysis and personalized prediction of vehicle states with joint time series modeling. Appl Energy 2020;261:114471.

[34] Murphy KP. Dynamic Bayesian networks: representation, inference and learning [dissertation]. Berkeley: University of California, Berkeley; 2002.

[35] Rabiner LR. A tutorial on hidden Marko models and selected applications in speech recognition. Proc IEEE 1989;77(2):257–86.

[36] Minka TP. Expectation propagation for approximate Bayesian inference. In: Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence; 2001 Aug 2–5; Seattle, WA, USA. San Francisco: Morgan Kaufmann Publishers Inc.; 2001. p. 362–9.

[37] Liu J, Luo Y, Xiong H, Wang T, Huang H, Zhong Z. An integrated approach to probabilistic vehicle trajectory prediction via driver characteristic and intention estimation. In: Proceedings of 2019 IEEE Intelligent Transportation Systems Conference (ITSC); 2019 Oct 27–30; Auckland, New Zealand. New York City: IEEE; 2019. p. 3526–32.

[38] Xiao W, Zhang L, Meng D. Vehicle trajectory prediction based on motion model and maneuver model fusion with interactive multiple models. SAE Int J Adv Curr Pract Mobil 2020;2:3060–71.

[39] Williams CK, Rasmussen CE. Gaussian processes for machine learning. Cambridge: MIT Press; 2006.

[40] Krajewski R, Bock J, Kloeker L, Eckstein L. The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways for validation of highly automated driving systems. In: Proceedings of 2018 21st International Conference on Intelligent Transportation Systems (ITSC); 2018 Nov 4–7; Maui, HI, USA. New York City: IEEE; 2018. p. 2118–25.

[41] Bishop CM. Pattern recognition and machine learning. New York: Springer; 2006.

[42] Wang H, Wang W, Yuan S, Li X. Uncovering interpretable internal states of merging tasks at highway on-ramps for autonomous driving decision-making, 2021. arXiv: 2102.07530.

[43] Yang T, Nan Z, Zhang H, Chen S, Zheng N. Traffic agent trajectory prediction using social convolution and attention mechanism. 2020. arXiv: 2007.02515.