

# Lane-Change Intention Inference Based on RNN for Autonomous Driving on Highways

Lin Li , Wanzhong Zhao , Can Xu , Chunyan Wang, Qingyun Chen , and Shijuan Dai

**Abstract**—Recently, inferring lane change intention has received considerable attention. Due to the high nonlinearity and complexity of traffic contexts, traditional methods cannot satisfy the requirements of long-term prediction tasks and lack the ability of capturing nonlinear temporal dependencies. This paper proposes an intention inference model based on Recurrent Neural Networks (RNN), to tackle time series prediction problems. Considering dynamic interaction among surrounding vehicles, our model takes the sequence motion information of surrounding vehicles as inputs and calculates the congestion of different lanes, integrated with vehicle states of the object vehicle. To illustrate the availability of the proposed RNN intention inference model, a motion planning controller considering intention was developed. A Nonlinear Model Predictive Control (NMPC) was established to planning a safe, sub-optimal path for autonomous driving vehicle under collision avoidance constraints. The experiments on the proposed model were conducted, based on two RNN structure Long-Short Term Memory (LSTM) and Generalized Recurrent Unit (GRU), by Tensorflow with NGSIM data. The motion planning controller is modeled and simulated by Carsim with Simulink for some typical scenarios. Subsequently, experimental results demonstrate that RNN achieves best performance, inferring intention with 96% accuracy, compared with other approaches.

**Index Terms**—Autonomous vehicle, lane change, intention inference, machine learning, recurrent neural networks.

## NOMENCLATURE

$\chi_k$	State vector of lateral movement at time step $k$
$u_s$	Lateral jerk
$C_{r,i}$	The congestion of rear vehicle at lane $i$
$C_{p,i}$	The congestion of proceeding vehicle at lane $i$
$v$	The velocity of vehicle, m/s
$d_{th}$	Predetermined threshold between vehicles, m
$y_e$	Lateral position of candidate trajectory, m
$\dot{y}_e$	Lateral velocity of candidate trajectory, m/s
$\ddot{y}_e$	Lateral acceleration of candidate trajectory, m/s <sup>2</sup>
$x_t$	Input vector of RNN intention inference model

Manuscript received October 5, 2020; revised February 5, 2021; accepted May 2, 2021. Date of publication May 11, 2021; date of current version July 8, 2021. This work was supported in part by the National Science Foundation of China under Grant 52072175, and by the Natural Science Foundation of Jiangsu Province under Grant BK20180482. (Corresponding authors: Shijuan Dai; Wanzhong Zhao.)

Lin Li, Wanzhong Zhao, Can Xu, Chunyan Wang, and Qingyun Chen are with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China (e-mail: lili96a@126.com; zhaowanzhong@126.com; xucan2011@126.com; wey2000@126.com; chenqingyuncqy@163.com).

Shijuan Dai is with the School of Material Science and Engineering, Nanjing University of Science and Technology, 210094 Nanjing, China (e-mail: daishijuan1980@163.com).

Digital Object Identifier 10.1109/TVT.2021.3079263

$\hat{y}_t$	Output vector of RNN intention inference model
$y_t$	Actual value of RNN intention inference model
$\varphi$	Heading angle of vehicle, rad
$Y$	Lateral position of vehicle, m
$X$	Longitudinal position of vehicle, m
$\delta_f$	Front wheel angle of autonomous vehicle, rad
$d_w$	The width of the vehicle, m
$d_l$	The length of the vehicle, m
$\sigma_w$	The clearance of vehicle width, m
$\sigma_l$	The clearance of vehicle length, m
$J_{ob}$	Cost function of collision avoidance
$\omega_i$	The weight of $i$ -th Gaussian distribution
$\mu_i$	The mean of $i$ -th Gaussian distribution
$\Sigma_i$	The covariance of $i$ -th Gaussian distribution
$a$	The feature input of the driving style clustering model

## I. INTRODUCTION

### A. Motivation

AUTONOMOUS driving is expected to improve the safety, efficiency and environmental impact of on-road transportation systems. Due to the development of automobile intelligence, there is an urgent need for autonomous vehicles equipped with a system that has more profound comprehension of traffic contexts, especially the intentions of surrounding vehicles. Such system makes it possible for autonomous vehicles to assess the potential risks and focus on the corresponding traffic context perception as early as possible. From the cognitive psychology perspective of view, intention refers to the thoughts that one has before the actions [1]. Accordingly, intentions of surrounding vehicles are the attitudes towards performing future behaviors. Intention inference, also called intention prediction, refers to the prediction of the intentional goal based on a set of incomplete sequence of actions [2]. Just like skilled human drivers, the ability of recognizing other's intention is required for autonomous vehicles, based on their observation and the learned driving skills. The prediction of driving intention is difficult because intention is invisible. Moreover, driving intention is affected by many factors, such as the change of traffic scene, the interaction between other traffic participants, etc. Under the dynamic and complex traffic environment, it is unreliable to only rely on the sensor information at a single moment to predict the driving intention of the surrounding vehicles, owing to the imperfect characteristics of sensor information. If the model can understand the nonlinear temporal dependent relationship in the

environmental information, the accuracy of intention prediction can be effectively improved.

### B. Related Works

Inferring the intention of road users on highways has been extensively studied. The works on intention inference can be simply divided into two aspects: from the high-level, the intention can represent the tactical behavior decisions, e.g. lane change or lane keep; from the low-level, it represents the maneuver input, e.g. longitudinal acceleration or steering wheel angle. This paper mainly focuses on the tactical behavior intention. From the viewpoint of the behavior intention prediction, various approaches exist in the literatures [3], which can be loosely grouped into three categories: firstly, physics-based prediction, which applies dynamic model to describe surrounding vehicles' movement based on the forces that affect the motion. However, these models are limited to short-term motion prediction, since they are unable to consider the decisions made by drivers. This can be addressed by the second approaches, maneuver-based prediction, assuming a vehicle at any time executes exactly one maneuver out of a set of possible driving maneuvers without interacting with other traffic participants. The data-driven approaches such as Logistic Regression, Hidden Markov Models (HMMs) [4], [5], Support Vector Machines (SVMs) [6], [7], Artificial Neural Network (ANN) models [8]–[10] and Bayesian Network (BN) [11], [12] have been investigated for behavior classifiers. While these methods in this class have the limitations of modeling interaction from each other even though they can achieve promising classification performance in some cases. Thirdly, interaction-aware prediction, which considers inter-vehicle dependencies and enriches the feature sets of a maneuver-based prediction by features like the relative distance and velocity to a proceeding object. In [13] and [14], coupled hidden Markov models (CHMMs) and game-theoretic offer ways to model multiple interacting processes. Nevertheless, as the number of possible pairwise dependencies grows in the context of complex traffic situations, the complexity quickly becomes intractable. Lately there is increasing interest in Deep Learning methods considering the powerful ability of capturing non-linear relations [15]. And the models have achieved great success in computer vision and natural language processing [16]. However, most studies just stacked several fully connected layers to fuse context data from multiple sources for inferring intention.

Draw a conclusion from above works, it is still difficult to learn how to infer intentions of surrounding vehicles accurately under dynamic and complex traffic, since most of intention inference models lack the ability of dealing with dynamic traffic context and capturing nonlinear temporal dependencies in observed sequence data. That is to say, the generation of the lane-change intention depends on multi-timesteps information, instead of a moment of information. Due to RNN can process time series data effectively and capture the nonlinear temporal dependence in time series through the recurrent unit in RNN, this paper uses RNN to predict the driving intention of surrounding vehicles. As the number of timesteps in information grows, the temporal dependencies in the observed data should be considered. At the

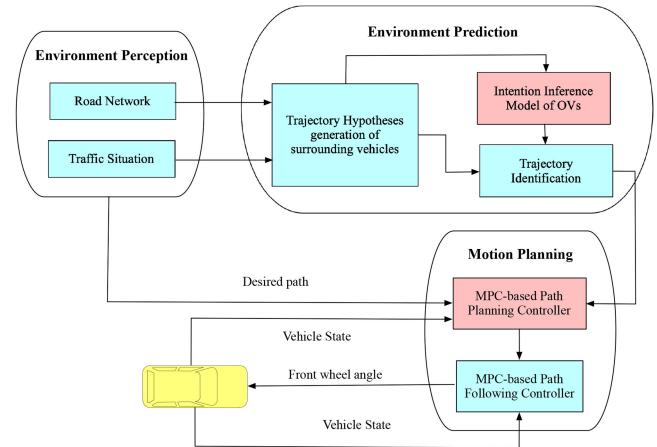


Fig. 1. Overall schematic of the proposed control framework.

same time, the model complexity quickly becomes intractable, since multiple sources such as interaction among vehicles and the vehicle states of object vehicles should be considered, which may lead to difficulty in being applied considering real time. Different from the previous studies [17]–[20], this paper does not directly take the motion state of multiple vehicles around as the input of the model, but calculates the traffic congestion of the lane integrated with the vehicle state of the object vehicle as the input of the model, so as to reduce the dimension of the model input and the complexity of the model. Moreover, most studies used the hand-crafted features so that they could ignore the potential information hidden in the data. It could result in the decreased robustness and accuracy to complex traffic.

In this paper, we proposed a novel RNN-based intention inference model considering dynamic interaction among surrounding vehicles. Our model takes the sequence motion information of Surrounding Vehicles (SV) as inputs and calculates the congestion of different lanes, integrated with vehicle states of the Object Vehicle (OV, the vehicle needs to be predicted). Then Recurrent Neural Network (RNN) is introduced to capture temporal dependencies in the observed data. The proposed model outputs the probabilities of different future potential intentions. The remainder of this paper is organized as follows. Section II introduces the overall framework. Our proposed RNN intention inference model is described in Section III. Section IV, validate the performance of our intention inference model by taking it into the motion planning control. The training process of the RNN is illustrated in Section V. Simulations results and corresponding discussions are included in Section VI, to show the workings of the proposed framework. Conclusions are offered in Section VII.

## II. FRAMEWORK

Fig. 1 shows schematic of the proposed intention inference for object vehicles. Object Vehicles (OV) are the vehicles that need to be predicted, which have the greatest impact on the driving of autonomous vehicle. And control framework for an Autonomous Vehicle (AV) under a complex traffic environment. In the environment perception module, possible routes for object

vehicles are extracted based on the information of the road networks from digital map first, together with the information of surrounding vehicles. Afterwards in environment prediction module, for each route the allowed lane with respect to the traffic rules and regulations are identified, and finally trajectory hypotheses for object vehicles are generated by all possible outcomes of the driver's optimization task when steering the vehicle towards a target state (e.g., a desired lane at a desired speed). To define the trajectory that an object vehicle is following, at first the driver's intention probability distribution is calculated by intention inference model. Eventually the information of intention is fused with the possible trajectories, generating a final specific trajectory. Accordingly, the motion planning module can generate optimal control input sequence and a safer trajectory based on the prediction of OVs.

#### A. Trajectory Hypotheses Generation of Surrounding Vehicles

In order to achieve proper intention of object vehicles, the prediction of surrounding vehicle trajectories (including object vehicles and other vehicles around them) is crucial. Here, we introduce the feasible trajectory generation of surrounding vehicles which based on global route planning. Specifically, from the current position, feasible lanes with which it is confronted are analyzed, assuming the vehicles always want to maintain a certain velocity and steer to the centerline of each lane with least effort. Thereby each vehicle optimizes individual objectives like minimizing the lateral acceleration or the time it takes to reach the center of the target lane [21]. Motivated by this assumption, we employ optimal control, in particular a linear quadratic regulator (LQR), to mimic the behaviors of surrounding vehicles.

For a linear time invariant and time discrete regulator the state space equation is given by

$$\chi_{k+1} = A\chi_k + Bu_{s,k} \quad (1)$$

where  $k = 0, 1, \dots, N - 1$  is the discrete time step and  $N$  is the finite prediction horizon.

In this part we focus on the lateral movement and assume constant velocity for the longitudinal motion. Therefore the state vector is chosen to be  $\chi_k = (y_{e,s,k}, \dot{y}_{e,s,k}, \ddot{y}_{e,s,k})^T$  with relative lateral position to the lane centerline  $y_{e,s,k}$ , velocity  $\dot{y}_{e,s,k}$  and acceleration  $\ddot{y}_{e,s,k}$  of the surrounding vehicles. In this case the input vector  $u_s$ ,  $k$  is the lateral jerk. With a discrete time interval  $T_s$ , the system matrices are

$$A = \begin{bmatrix} 1 & T_s & T_s^2/2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} T_s^3/6 \\ T_s^2/2 \\ T_s \end{bmatrix}$$

The cost function is given by

$$\begin{aligned} J(\chi, u_s) = & \|(\chi - \chi_{ref})\|_P^2 + \sum_{k=0}^{N-1} \|(\chi - \chi_{ref})\|_Q^2 \\ & + \sum_{k=1}^{N-1} \|u_{s,k}\|_R^2 \end{aligned} \quad (2)$$

where  $P > 0$  is the positive semidefinite state penalty,  $Q > 0$  is terminal state penalty, and  $R > 0$  is the positive definite

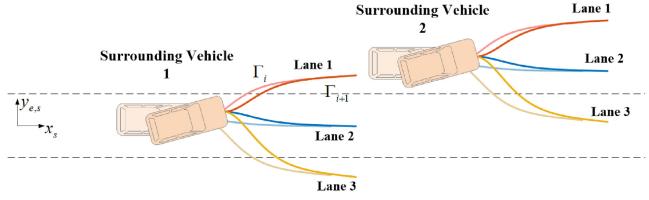


Fig. 2. An example for results of trajectory hypotheses at different time.

input penalty. According to the aforementioned assumption, the vehicle's lane reference  $\chi_{ref}$  corresponding to relative lateral position to lane centerline with lateral velocity and acceleration should be equal to zero.

With the initial state  $\chi_0$ , the optimal control input sequence  $u_s^*$  can be found by solving

$$\min_{u_s} J(\chi, u_s) = J(\chi, u_s^*) \quad (3)$$

for which we use the dynamic programming approach. The optimal state sequence  $\chi^*$  can be computed by applying  $u_s^*$  to the system stated by (1). Fig. 2. shows an example of generation of trajectory hypotheses. The light lines represent the trajectory hypotheses  $\Gamma_i$  generated at time  $i$ . The dark lines are the trajectory hypotheses  $\Gamma_{i+1}$  generated at time  $i + 1$ .

### III. RNN INTENTION INFERENCE MODEL

The factors related to lane-change intention on highways can be divided into two aspects: 1) Comparing the congestion of current lane and adjacent lanes, which can represent the interactive traffic condition of the object vehicle. If the congestion of other lane is lower than current lane, the vehicle will desire to change lane. 2) From the stability standpoint, the past and current motion state of the vehicle should be considered as well. That is to say, the generation of the lane-change intention depends on past, current even future information, instead of a moment of information. Consequently, we utilize Recurrent Neural Networks (RNN) [22]–[25] and its extensions Long-Short Term Memory (LSTM) and Generalized Recurrent Unit (GRU) to build an intention inference model. As shown in Fig. 3, the network is composed of input layer, hidden state layer and output layer, in addition to the fully connection between layers, the connection between past time with current time also existing. To identify the parameters of model, the NGSIM dataset was used in the training and test process, from which the information of natural driving could be learned. The results of Section II. A will be taken as some of the inputs of the proposed intention inference model. The detailed definitions of each layer in the network are explained further in the following parts of the chapter.

#### A. Input Layer of RNN Structure

1) *Congestion of Different Lanes:* The traffic environment and related vehicles during the lane-change process have been demonstrated in Fig. 4. When proceeding vehicle in the current lane is driving too slowly, or the rear vehicle is following too close, the object vehicle is more likely to change-lane. In order to analyze the influence of the surrounding traffic environment

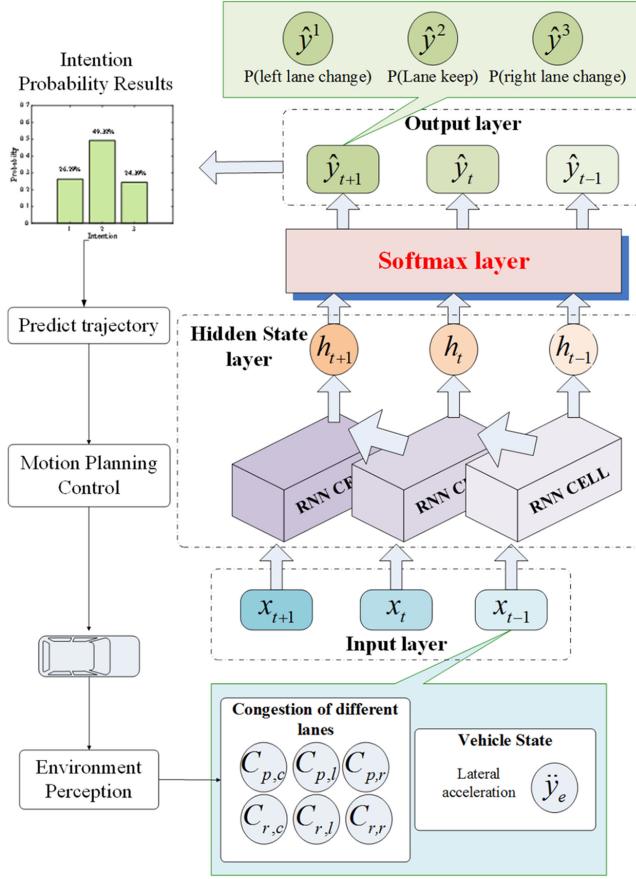


Fig. 3. The model of RNN intention inference.

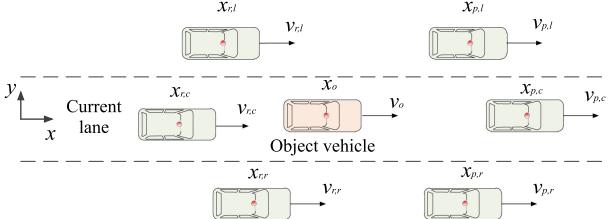


Fig. 4. The traffic environment and related vehicles during the lane-change process for object vehicle.

on the driving intention of the object vehicle, we measure the congestion each lane. The congestion here refers to the capacity of the road. Headway refers to the distance between the front and rear vehicles and the time difference between two vehicles passing a certain point. It is an important basis to reflect the road capacity. Therefore, we calculate the headway to reflect the congestion in each lane. Accordingly, we introduce  $C_{r,i}$ ,  $C_{p,i}$  to interpret the congestion of  $i$  lane.

$$C_{r,i} = \begin{cases} \frac{v_{r,i}}{x_o - x_{r,i} - d_l}, & \text{if } x_o - x_{r,i} \leq d_{th} \\ 0, & \text{if } x_o - x_{r,i} > d_{th} \end{cases} \quad (4)$$

$$C_{p,i} = \begin{cases} \frac{v_o}{x_{p,i} - x_o - d_l}, & \text{if } x_{p,i} - x_o \leq d_{th} \\ 0, & \text{if } x_{p,i} - x_o > d_{th} \end{cases} \quad (5)$$

where  $i \in \{l, c, r\}$  indicates the lane information.  $l$  is the left lane,  $c$  is the current lane and  $r$  is the right lane.  $x_o$  is longitudinal position of object vehicle with the lane coordinate, respectively,  $x_{p,i}$ ,  $x_{r,i}$  are the proceeding and rear vehicle longitudinal position.  $v_o$ ,  $v_{r,i}$  are, respectively, velocity of object vehicle and rear vehicle.  $d_l$  is the length of vehicle body, which we assume the lengths of all vehicles on road are the same.  $d_{th}$  is the predetermined threshold between vehicles. Referring to relevant regulations in China, we known that the vehicles need to keep 50-100 meters away from the others. Therefore, considering safety the thresholds in different lanes are all set as 200 meters. Once the distance is larger than this value, we could ignore the existence of the proceeding or rear vehicle. In our model, the value of congestion reflects how much space have been occupied by surrounding vehicles. When the longitudinal distance between the vehicles becomes closer, the higher C-value will be and the smaller space for driving will be. It is rational that the congestion will increase when other vehicles approaching.

2) *Vehicle States*: In addition to the outer factors, the vehicle itself may have an obvious impact on the lane-change decision. Even if the current lane condition is poor, as the object vehicle is in unstable situation, it is improbable to generate lane-change intention. In this paper we use lateral acceleration  $\ddot{y}_e$  to describe the stability of the object vehicle.

As a result, the inputs of the network at each time is  $x_t$ , given by (6), and a sequence input during a recurrent process is  $X$ , given by (7).

$$x_t = [C_{p,c}, C_{r,c}, C_{p,l}, C_{r,l}, C_{p,r}, C_{r,r}, \ddot{y}_e]^T \quad (6)$$

$$X = [x_{t-N_h}, x_{t-N_h+1}, \dots, x_{t-1}, x_t, x_{t+1}, \dots, x_{t+N}]^T \quad (7)$$

where  $N_h$  is the history time horizon, and  $N$  is the finite prediction horizon.

### B. Hidden State Layer of the RNN Structure

The driving environment of autonomous vehicle is dynamic and complex. The traditional neural network structure can well deal with the nonlinear relationship in the data. However, with the increase of factors considered in the driving intention inference model, the complexity of the model increases sharply, resulting in unstable performance, and can not effectively improve the accuracy of driving intention. Therefore, based on the structure of recurrent neural network, this paper establishes the intention inference model in dynamic traffic scenes. The units between hidden layers are no longer independent, but connected in time order, and process the time series data containing interactive traffic scene information. When inferring the driving intention at time  $t$ , not only the scene information at time  $t$ , but also the scene information at time  $t-1$  should be input. The structure of the recurrent neural network is shown in Fig. 5. We can see that there are connections between the hidden layer units  $h_t$  at the front and back time.

However, in practical application, the traditional RNN can not realize the training process due to the problems of vanishing gradient and the exploding gradient [26]. Therefore, this paper establishes recurrent neural network based on LSTM cells. Each LSTM consists of a memory unit and three gating units, which

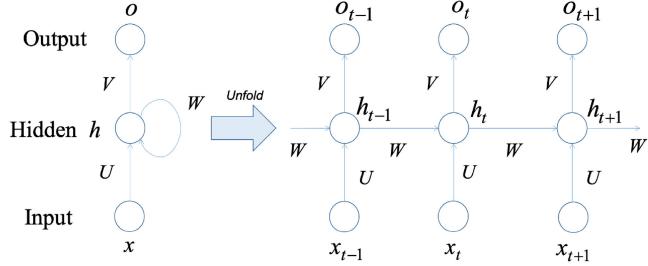


Fig. 5. The structure of RNN network work.

are input gate, output gate and forget gate. The known traffic scene information is taken as the input of the input gate, and the forgetting gate is used to deal with the data uncertainty caused by other traffic complexity or vehicle system disturbance to retain or discard the current effective traffic flow information, so as to output the hidden layer vector containing the traffic scene information through the output gate. Through the learning of real data, the model can learn to judge whether the current traffic scene information is valid for inferring driving intention, and then choose to retain or discard the current input information. This structure can dynamically adjust the weight of driving intention affected by traffic scene information at different times to adapt to dynamic and complex traffic scenes.

LSTM [27] and GRU [28] have a basic unit called memory cell in the hidden state layers, similar to the hidden state unit  $h_t$  in normal RNN. The modeling process is expressed as follows.

$$h_t^l = LSTM(x_t, h_{t-1}^l; W_{LSTM}) \quad (8)$$

$$h_t^g = GRU(x_t, h_{t-1}^g; W_{GRU}) \quad (9)$$

where  $W_{LSTM}$  is the weight coefficient of the model based on LSTM cell and  $W_{GRU}$  is the weight coefficient of the model based on GRU cell.

### C. Output Layer of RNN Structure

Taking the outputs of hidden layers as inputs for the output layer, the output layers can produce  $\hat{y}_t$  as (10). In our model, the output is the vehicle's lane-change intentions, specifically, including lane keep, left lane change and right lane change, noted as  $\hat{y}_t = [\hat{y}_t^1, \hat{y}_t^2, \hat{y}_t^3]$ . Considering the uncertainty of driving intention, softmax function is used to output the probability distribution of different driving intentions

$$\begin{cases} o_t^m = Vh_t^m + b_y \\ \hat{y}_t^m = softmax(o_t^m) \end{cases} \quad (10)$$

where  $V$  is the weight matrix connecting hidden state layers with output layers.  $b_y$  is the biases vector.  $m \in \{l, g\}$ . Softmax function has been widely used in the multi-class problem. Briefly, it can cast original inputs  $o_t$  to values among zero to one, the sum of which is one. Consequently, we can view the results as the probability of each intention.

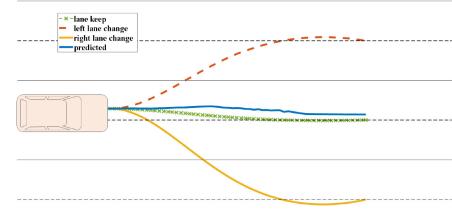


Fig. 6. The trajectory generation based on intention inference results.

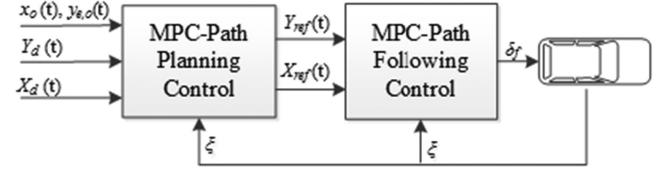


Fig. 7. The motion planning control system for automated vehicle.

### IV. MOTION PLANNING CONTROL BASED ON INTENTION INFERENCE

The aim of this portion is to investigate the importance and performance of the proposed intention inference model, and the model is applied to the motion planning module where a Nonlinear Model Predictive Control problem is formulated [14], [29]. OV perception module collects and provides history motion states for the RNN intention inference model. The prediction model outputs the predicted trajectory of object vehicle. With the information, the motion planning block can calculate the best sequence of front wheel angle and send to the autonomous vehicle. The reference position calculation considers that the autonomous vehicle should drive safely in complex traffic situation. By adding the collision avoidance constraint based on the inference intention of object vehicles, the safety of the reference positions can be guaranteed. And the performance of motion planning control with the intention inference model will be given in the Section VI.

Based on the results of intention inference, at first, we can get the predictive trajectory of object vehicles by the following (11), presented in the Fig. 6.

$$y_{e,t,p} = y_{e,t}^1 * \hat{y}_t^1 + y_{e,t}^2 * \hat{y}_t^2 + y_{e,t}^3 * \hat{y}_t^3 \quad (11)$$

where  $y_{e,t}^1, y_{e,t}^2, y_{e,t}^3$  are the lateral position generated by the feasible trajectory hypotheses introduced in Section II.A.  $\hat{y}_t^1, \hat{y}_t^2, \hat{y}_t^3$  are the probabilities of the feasible lanes.

#### A. Problem Statement

The main work for the motion planning task is to design reasonable objective function, and then calculate the optimal control input under various constraints, which is able to avoid obstacle and minimize deviation to the desired path. See Fig. 7,  $x_o, y_e, o$  are the longitudinal and lateral position of object vehicle, which can get from Section II. A.  $X_d, Y_d$  are the desired paths from higher planning controller.  $X_{ref}, Y_{ref}$  are the local reference paths generated by the path planning control and as the input of the path following control. Finally, the path following

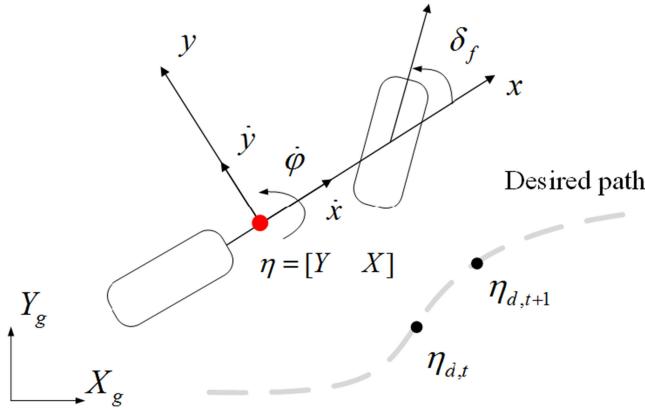


Fig. 8. Vehicle model based on particle motion in the coordinate.

controller outputs the front wheel angle  $\delta_f$ . The motion problem (12) can be formulated as the following optimal problem. The objective function and constraints will be explained in the following subsection.

$$\begin{aligned} & \min_{U_t} J \\ & \text{s.t. } U_{\min} \leq U_t \leq U_{\max} \end{aligned} \quad (12)$$

### B. Vehicle Model

Considering the real time feasibility, the vehicle model based on particle motion is adopted in the motion planning module, as shown in the Fig. 8, and (13). However, higher precision model [30] needs to be used in the control layer for satisfactory control performance.

$$\begin{cases} \ddot{y} = a_y \\ \ddot{x} = 0 \\ \dot{\varphi} = a_y/\dot{x} \\ \dot{Y} = \dot{x} \sin \varphi + \cos \varphi \\ \dot{X} = \dot{x} \cos \varphi - \dot{y} \sin \varphi \end{cases} \quad (13)$$

which can be denoted as (14):

$$\dot{\xi}(t) = f(\xi(t), \mu(t)) \quad (14)$$

where  $\xi = [\dot{y}, \dot{x}, \varphi, Y, X]^T$ ,  $\dot{y}$ ,  $\dot{x}$  are respectively lateral and longitudinal velocity.  $\varphi$  is the heading angle and  $Y$ ,  $X$  are respectively lateral and longitudinal position of the vehicle. The control input is the front wheel angle  $\delta_f$ .

### C. Constraints

Two kinds of constraints are considered in this paper: the dynamic constraint and the obstacle avoidance constraint. The dynamic constraint is denoted as (15):

$$|\mu(t)| < \lambda g \quad (15)$$

where  $\lambda$  is the road adhesion coefficient and  $g$  is inertial acceleration.

We choose the ellipse to describe the collision area, by extending the length and width of the vehicle with clearance  $\sigma_w$ ,  $\sigma_l$ , shown in the Fig. 9. The collision avoidance constraint can

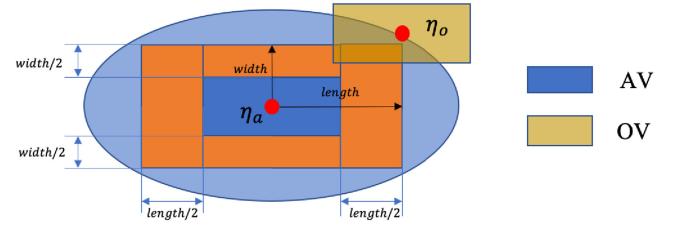


Fig. 9. Elliptical boundary for collision avoidance.

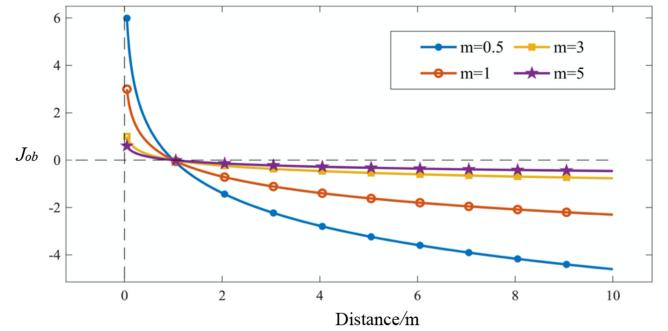


Fig. 10. The collision avoidance function with different coefficient.

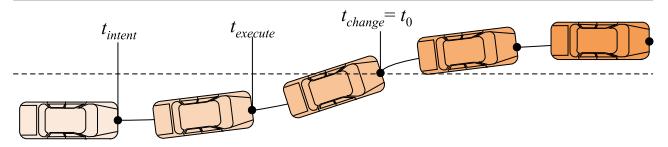


Fig. 11. The definitions of time during lane change process.

be denoted as (16):

$$\left( \frac{Y_a - Y_o}{d_w + \sigma_w} \right)^2 + \left( \frac{X_a - X_o}{d_l + \sigma_l} \right)^2 > 1 \quad (16)$$

where  $Y_a, Y_o$  are respectively the lateral position of the autonomous vehicle and object vehicle.  $X_a, X_o$  are respectively the longitudinal position of the autonomous vehicle and object vehicle.  $d_w$  and  $d_l$  represent the width and length of the vehicles. The length of vehicle is set as 4.5 m and the width of vehicle is chosen as 1.8 m as common.

To address the constrained nonlinear programming problem, we use the sequential unconstrained minimization technique (SUMT), by adding some form of constraint function to the objective function [31]. We choose the common and effective constraint function form as barrier function (17), also called Interior point method.

$$J_{ob}(d(\eta_a, \eta_o)) = -\frac{1}{m} \log \left( \left( \frac{Y_a - Y_o}{d_w + \sigma_w} \right)^2 + \left( \frac{X_a - X_o}{d_l + \sigma_l} \right)^2 - 1 \right) \quad (17)$$

The Fig. 11 shows how the coefficient  $m$  changes the shape of the constraint function.

### D. Objective Function

The objective function is decoupled into control cost and state costs. For control cost, there is cost of front wheel angle. For state

costs, there are cost of deviation to desired path and cost of the distance to object vehicles. Consequently, the objective function is given by (18):

$$J = \sum_{i=1}^{N_p} \| \eta(t+i | t) - \eta_d(t+i | t) \|_Q^2 + \| U_i \|_R^2 + J_{ob,i} \quad (18)$$

where  $\eta = [Y, X]^T$  is the path information,  $J_{ob,i}$  is the collision avoidance function at sample time  $i$ .

## V. LEARNING PROCESS OF MODEL PARAMETERS

In order to obtain the weights matrixes and biases vectors of the RNNs, the training process is essential. This part describes the details of the training for the RNNs with the Next Generation Simulation (NGSIM) database, which provides high-quality real-world traffic datasets. The Next Generation Simulation (NGSIM) database is used in this paper for the learning of our proposed intention inference model. Collected and published by the US Federal Highway Administration in 2005, the NGSIM is one of the largest open datasets of naturalistic driving and has been used in the numerous literatures. More specifically, the area of interest is I-80 freeway in Emeryville, California, of which the covered segment is approximately 500 m in length and 6 lanes (3.66 m or 12 ft each) in width. The dataset contains more than 5000 trajectories of individual vehicles, with a sampling rate at 10 Hz, from which we pick up 100 trajectories of each kind of behaviors (respectively left lane-changing, lane keeping and right lane-changing). The trajectories are divided into training sets (70%) and test sets (30%).

### A. Data Preprocessing

Our investigation mainly focuses on the free lane-change model [32], thus we collected Interstate 80(I-80) freeway dataset. The study area is approximately 500 meters in length and consists of six freeway lanes. Generally, the lane-change process can be divided into three phrases, the generation of lane change intention, the execution of lane change maneuver and the beginning of lane change, which are shown in the Fig. 11. As can be seen in the figure, we can easily identify the beginning time of lane change. For the time of the intention generating, according to the literature [30], the moment is on individual. They note the time from intention generating to the lane change beginning as intention time window, which could be roughly grouped into three categories: conservative, common and aggressive. In our experiments, we respectively choose 5 s, 3.5 s and 2 s for the intention time windows of different driving styles, capturing the 10 frames before and 10 frames after the intention time, see Fig. 12.

### B. Driving Style Clustering

In order to recognize different driving style during lane change process, we analyze the characteristic of the driving using the Gaussian Mixture Model (GMM). GMM is a parametric probability density function represented as a weighted sum of

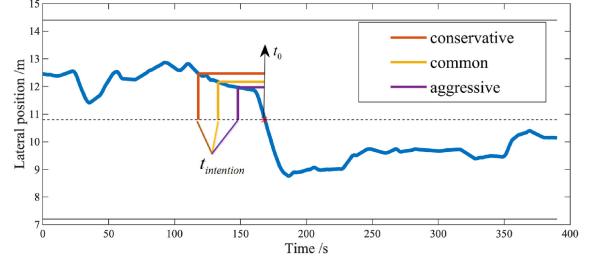


Fig. 12. The intention time windows of different driving styles.

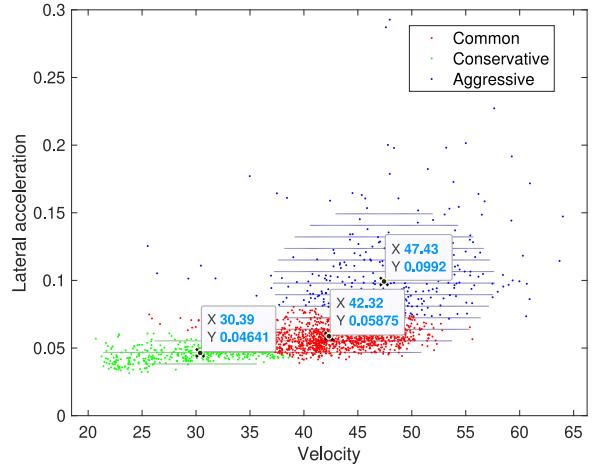


Fig. 13. The results of GMM cluster.

Gaussian component densities [33], given by

$$p(a|\omega_i, \mu_i, \Sigma_i) = \sum_i^M \omega_i N(\mu_i, \Sigma_i) \quad (19)$$

where  $M$  is the number of mixture components,  $\omega_i$  is the weight of  $i$ - th Gaussian distribution with  $0 \leq \omega_i \leq 1$ ,  $\sum_i^M \omega_i = 1$ , and  $\mu_i, \Sigma_i$  are the mean and covariance of  $i$ - th Gaussian distribution, respectively. Here  $M$  represents the kinds of the driving styles.  $a$  is the input feature of driving style clustering model. As mentioned before, the drivers are classified into three kinds and each Gaussian component represents one cluster. We need to obtain the parameters  $\theta$  of Gaussian Mixture Model, specifically, the mean and covariance of each Gaussian distribution and the weights of them, based on EM algorithm [34]. The mean of each Gaussian component is the center of the cluster. The analysis results of our driving style are presented in the Fig. 13. The x-axis is the longitudinal velocity and the y-axis is the lateral acceleration. From the figure, we can see that the green cluster has the minimum longitudinal velocity and lateral acceleration, which represents the conservative drivers, and the blue cluster has the maximum longitudinal velocity and lateral acceleration, which is the aggressive drivers. The red cluster which is in the middle of them represents the common drivers. In a conclusion, the results of the cluster are reasonable. The clustered data is a preparation for the following tests.

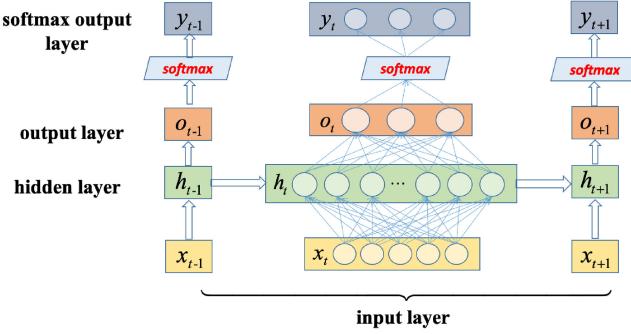


Fig. 14. The whole network structure of the RNN.

TABLE I  
PARAMETERS SETTING OF NETWORKS

Name of parameters	Value
Nodes of input layer	7
Nodes of hidden layer	30
Nodes of output layer	3
Time step	21
Batch size	5

### C. Setting of the Parameters

The whole network structure is presented in Fig. 14 including input layer, hidden state layer, output layer and softmax output layer. According to the sizes of the network and training data, we need to search the most appropriate hyper-parameters during the training process. The final detailed settings of the hyper-parameters can be seen in the Table I.

### D. Learning Algorithm for Weights Matrix and Biases Vector

The network computation stated in Section III can be briefly noted as (20):

$$\hat{y}_t = F(x_t | W, b) \quad (20)$$

Then, we use the cross-entropy as the loss function:

$$Loss(y_t, \hat{y}_t) = -1/n \sum_{t=1}^n \sum_{j=1}^{M=3} (y_t^{(j)}) \ln(\hat{y}_t^{(j)}) \quad (21)$$

where  $y_t$  is the ground truth mapping with  $x_t$ , encoded as the form of one-hot. ([1 0 0], [0 1 0] and [0 0 1], respectively, represent left lane change, lane keep and right lane change.) For the purpose of obtaining the weights and biases, we can formulate an optimal problem (25):

$$W^*, b^* = \arg \min_{W, b} Loss(y_t, \hat{y}_t) \quad (22)$$

To address the problem, the open-source machine intelligence library by Google-Tensorflow was used. TensorFlow enables scientists and engineers to design, develop, and deploy computational models through the use of data flow graphs. We choose the Adam optimizer [35] to update our weights and biases. (see Fig. 15) The nodes in the graph represent mathematical operations, while the edges represent tensors communicated between them.

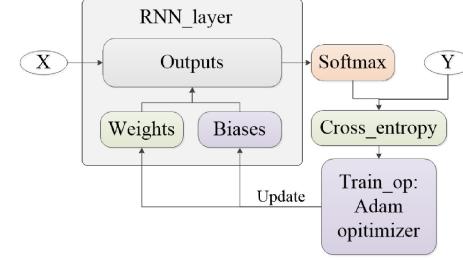


Fig. 15. The data flow graph of Tensorflow.

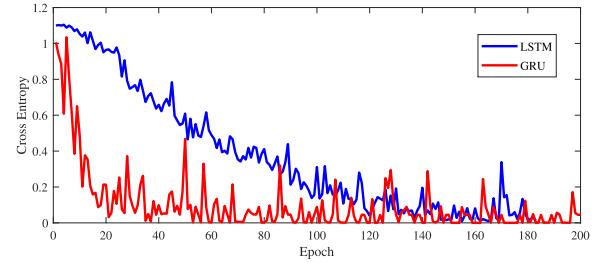


Fig. 16. Training loss of LSTM and GRU.

### E. Learning and Testing Performances

Other machine learning approaches such as BP [10] and RBF [36] are also used in comparison to our RNN intention inference method, based on LSTM and GRU. For the BP and RBF, we only input the data at the intention time. The training processes of LSTM and GRU are presented in the following Fig. 16. After about 200 epochs, the cross entropy of LSTM and GRU both approach to zeros, and the convergence speed of GRU seems faster, for the cross entropy of GRU has reached to zero at epoch 20. The receiver operating characteristic (ROC) curve is used to access our trained model performance, which is widely used to evaluate the classification performance, by considering the true positive rate (TPR) with false positive rate (FPR) and adjusting the threshold from 0 to 1. Note that higher TPR and lower FPR means better performance. Visually speaking, the curves approaching the left upper enjoy better qualities. TPR and FPR are computed by (23), (24), the ROC curves can be seen in the Fig. 17. The line marked blue represents the class of left lane change. The green one is lane keep and the red one is right lane change. It is obvious that LSTM and GRU provide superior performances, and GRU performs better. Then the overall datasets were divided into three classes according to driving styles, in this paper, we use the velocity and lateral acceleration as the features. Fig. 18 shows the ROC curves of different driving styles. We can see that all of them obtain pleasurable performances.

$$TPR = \frac{TP}{TP + FN} \quad (23)$$

$$FPR = \frac{FP}{TN + FP} \quad (24)$$

where  $TP, TN, FP, FN$  are respectively true positives, true negatives, false positives, and false negatives.

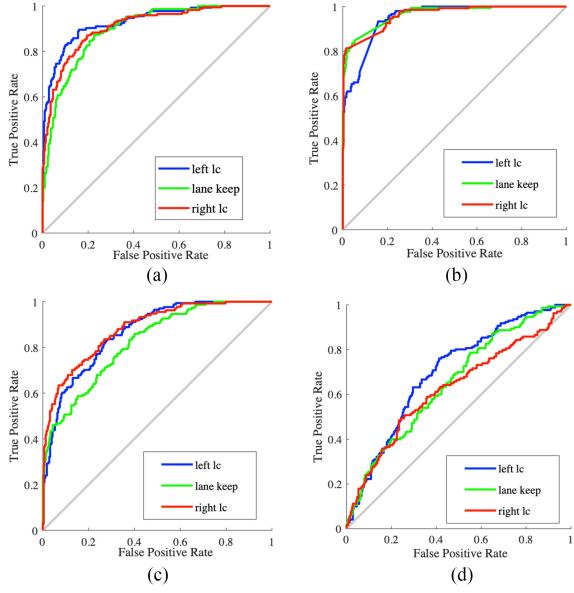


Fig. 17. Classification performances ROC curves of different approaches, (a) LSTM, (b) GRU, (c) RBF, and (d) BP.

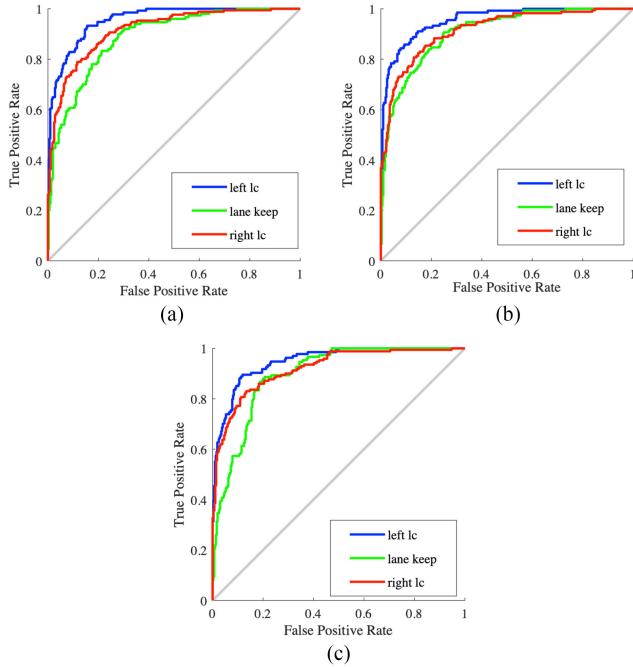


Fig. 18. ROC curves of different driving styles. (a) aggressive drivers (b) common driver, and (c) conservative drivers.

To illustrate the performance for the overall datasets, we use the confusion matrix (e. e. For class A, B, confusion matrix sum up the number of class A is divided into B) and then compute the accuracy, precision and recall of the prediction result, which can be computed by (25), (V-E), (27) test results of different algorithms are shown in the Fig. 19 and Table II– IV.

$$\text{accuracy} = 1/N * \sum_t^N I(\hat{y}_t = y_t) \quad (25)$$

	Output Class 1	Output Class 2	Output Class 3	
Target Class 1	150 (33.0%)	13 (2.9%)	7 (1.5%)	<b>88.2% (11.8%)</b>
Target Class 2	0 (0.0%)	158 (34.7%)	0 (0.0%)	<b>100% (0.0%)</b>
Target Class 3	0 (0.0%)	0 (0.0%)	127 (27.9%)	<b>100% (0.0%)</b>

	Output Class 1	Output Class 2	Output Class 3	
Target Class 1	127 (27.9%)	41 (9.0%)	22 (4.8%)	<b>66.8% (33.2%)</b>
Target Class 2	28 (6.2%)	82 (18.0%)	21 (4.6%)	<b>62.6% (37.4%)</b>
Target Class 3	16 (3.5%)	27 (5.9%)	91 (20.0%)	<b>67.9% (32.1%)</b>

	Output Class 1	Output Class 2	Output Class 3	
Target Class 1	114 (25.1%)	60 (13.2%)	49 (10.8%)	<b>51.1% (48.9%)</b>
Target Class 2	31 (6.8%)	64 (14.1%)	36 (7.9%)	<b>48.9% (51.1%)</b>
Target Class 3	26 (5.7%)	26 (5.7%)	49 (10.8%)	<b>48.5% (51.5%)</b>

	Output Class 1	Output Class 2	Output Class 3	
Target Class 1	74.3% (25.7%)	54.7% (45.3%)	67.9% (32.1%)	<b>65.9% (34.1%)</b>
Target Class 2	66.7% (33.3%)	42.7% (57.3%)	36.6% (63.4%)	<b>49.9% (50.1%)</b>
Target Class 3	66.7% (33.3%)	42.7% (57.3%)	36.6% (63.4%)	<b>49.9% (50.1%)</b>

Fig. 19. Confusion matrix of different approaches. (a) LSTM (b) GRU (c) RBF (d) BP, where 1 -left lane change, 2 - lane keep, and 3 - right lane change.

TABLE II  
ACCURACY OF DIFFERENT APPROACHES

	LSTM	GRU	RBF	BP
accuracy	<b>96.0%</b>	95.6%	65.9%	49.9%

TABLE III  
PRECISION OF DIFFERENT APPROACHES

	LSTM	GRU	RBF	BP
Left LC	<b>96.3%</b>	88.2%	66.8%	51.1%
Lane keep	95.9%	<b>100%</b>	62.6%	48.9%
Right LC	95.9%	<b>100%</b>	67.9%	48.5%

TABLE IV  
RECALL OF DIFFERENT APPROACHES

	LSTM	GRU	RBF	BP
Left LC	97.8%	<b>100%</b>	74.3%	66.7%
Lane keep	<b>94.0%</b>	92.4%	54.7%	42.7%
Right LC	<b>96.5%</b>	94.8%	67.9%	36.6%

$$\text{precision} = TP / (TP + FP) \quad (26)$$

$$\text{recall} = TP / (TP + FN) \quad (27)$$

Compared with previous approaches BP and RBP, the overall performances of LSTM and GRU are dramatically outstanding. The accuracies of LSTM and GRU are both upper than 95%, and are respectively 96.0% and 95.6%. For the precision, the two classes of three have reached to 100% when GRU algorithm was used. The lowest precision of GRU is 88.2%, which is also much higher than the highest precision of RBF, 67.9%. The precisions of LSTM are all around 96%. The results of recall are in the similar circumstance.

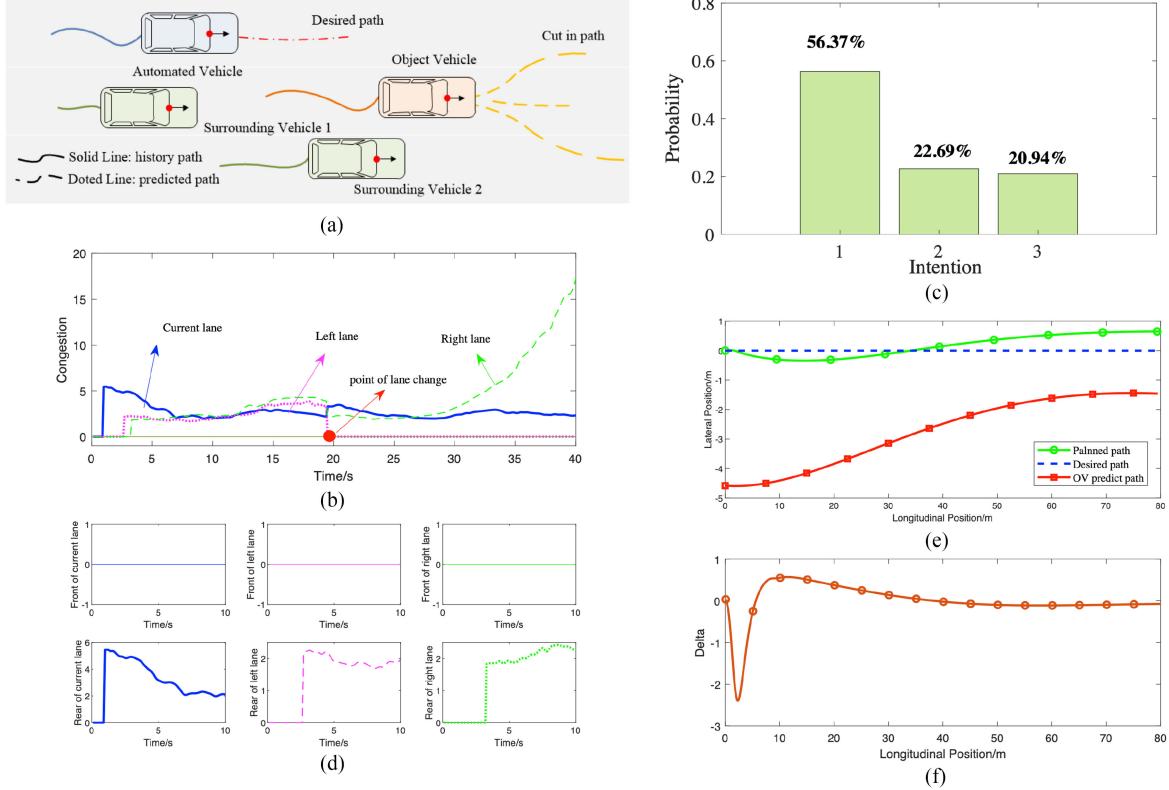


Fig. 20. The simulation results of cut-in scenario. (a) Schematic of cut-in scenario. (b) Congestion of different lanes. (c) Intention inference results. (d) Detailed congestion during control time domain. (e) The planned path compared with original desired path. (f) front wheel angle profile.

## VI. CASES STUDIES

In Section III, the RNN-based intention inference model is proposed, and then employed for the MPC-based motion planning control in Section IV. To illustrate the capability of proposed intention inference model, our experiments were mainly conducted in the typical driving scenarios. We select the cut-in driving scenarios from the NGSIM dataset, which provides the motion information of object and surrounding vehicles appearing in the scenarios. In the architecture, the autonomous vehicle's motion states can be obtained from the CarSim model. OV Perception collects and provides history motion states for the RNN intention inference model. The prediction model outputs the predicted trajectory of object vehicle. With the information, the motion planning block can calculate the best sequence of front wheel angle and send to the Carsim model. Through previous experiments, we find that GRU performs best, so that we use this algorithm for intention inference model in the case study part. This paper mainly focuses on the lateral motion of the autonomous vehicle, as a result, the longitudinal velocity of AV and OV are both set as constant 10 m/s. The detailed parameters used in the simulation are listed in the following Table V. In our simulation, the blue vehicle is the autonomous vehicle (AV). The orange ones are the object vehicles (OVs). And green ones represent other surrounding vehicles, which are around the AV and OVs. The solid line is the history path and the predicted path is represented by dotted line.

TABLE V  
SIMULATION PARAMETERS AND VEHICLE PARAMETERS

Parameters	Description	Values
$S_f$	Front wheel slip rate	0.2
$S_r$	Rear wheel slip rate	0.2
$l_f$	Front CoG distance	1.232 m
$l_r$	Rear CoG distance	1.468 m
$C_{cf}$	Cornering stiffness of front tire	66900 N/rad
$C_{cr}$	Cornering stiffness of rear tire	62700 N/rad
$m$	Vehicle mass	1723 kg
$g$	Acceleration of gravity	$9.8 \text{ m/s}^2$
$I$	Vehicle yaw moment of inertia	$4175 \text{ kgm}^2$
$T_p$	Planning period	0.1 s
$N_p$	Planning horizon	$20 T_p$
$T_t$	Tracking period	0.02 s
$N_t$	Tracking horizon	$20 T_t$

### A. Cut in Scenario

In this case, the autonomous vehicle desired to keep the lane but observed the object vehicle was more likely to execute left lane change. The scenario is described in the Fig. 20(a). Based on the motion information of surrounding vehicles, the congestion for different lanes were calculated for object vehicle. The overall congestion is presented in Fig. 20(b), and the detailed congestion during lane change is given in Fig. 20(d). There is no any vehicle in the front of object vehicle. However, the congestion of current lane was much higher than other two lanes. Our model outputs 56.37% probability of left lane change. Then according to the predicted path of object vehicle, motion planning controller of

autonomous vehicle produced a modified path considering the safety and efficiency, the results are shown in the Fig. 20(e). In addition, Fig. 20(f) presents the corresponding control input, front wheel angle.

## VII. CONCLUSION

In this paper a deep learning methodology is proposed to infer the intention of vehicles considering the dynamic interaction among surrounding vehicles, and the detailed training process of the proposed RNN intention inference model is illustrated. The main contributions of this paper are listed below. 1) Developing an RNN model to inference intention of lane change on highways considering the dynamic interactions among surrounding vehicles and the vehicle state of the object vehicle. 2) Integrating the deep learning with Recurrent Neural Network to learn the driving knowledge and capture the temporal dependencies in observed sequence data. 3) Formulating a Nonlinear Model Predictive Control (NMPC) to handle the motion planning problem that takes into account the intention inference results, where the significance of our proposed RNN intention inference model could be verified. By analysis of congestion to different lanes with vehicle states, the model is able to predict the intention of the object vehicle under dynamic and complex traffic situations. The superiority of our approach has been demonstrated with the ROC curves and test accuracy of 96%, 95.6%, compared with RBF and BP algorithms. In the motion planning parts, our model both presents satisfactory performance in cut in scenarios and overtaking scenarios, providing a safer path for the autonomous vehicles. However, this paper mainly concentrates on the behavior prediction of surrounding vehicles. For further study, we will improve our work on more detailed trajectory prediction based on the intention inference results. In addition, we would carry out real-world experiments in future as to train model with more natural driving data.

## REFERENCES

- [1] M. Bratman, “Intention, plans, and practical reason,” Cambridge, MA: Harvard Univ. Press, vol. 10, 1987.
- [2] Y. Xing *et al.*, “Driver lane change intention inference for intelligent vehicles: Framework, survey, and challenges,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4377–4390, May 2019.
- [3] S. Lefèvre, D. Vasquez, and C. Laugier, “A survey on motion prediction and risk assessment for intelligent vehicles,” *Robomech J.*, vol. 1, no. 1, pp. 1–14, 2014.
- [4] J. Ding, R. Dang, J. Wang, and K. Li, “Driver intention recognition method based on comprehensive lane-change environment assessment,” in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 214–220.
- [5] C. Zong, C. Wang, D. Yang, and H. Yang, “Driving intention identification and maneuvering behavior prediction of drivers on cornering,” in *Proc. IEEE Int. Conf. Mechatronics Automat.*, 2009, pp. 4055–4060.
- [6] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, “Learning-based approach for online lane change intention prediction,” in *Proc. IEEE Intell. Veh. Symp.*, 2013, pp. 797–802.
- [7] B. Morris, A. Doshi, and M. Trivedi, “Lane change intent prediction for driver assistance: On-road design and evaluation,” in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 895–901.
- [8] I. H. Kim, J. H. Bong, J. Park, and S. Park, “Prediction of driver’s intention of lane change by augmenting sensor information using machine learning techniques,” *Sensors*, vol. 17, no. 6, pp. 1–18, 2017.
- [9] V. Leonhardt and G. Wanielik, “Neural network for lane change prediction assessing driving situation, driver behavior and vehicle movement,” in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 1–6.
- [10] C. Chen, L. Liu, T. Qiu, Z. Ren, J. Hu, and F. Ti, “Driver’s intention identification and risk evaluation at intersections in the internet of vehicles,” *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1575–1587, Jun. 2018.
- [11] X. Li, W. Wang, and M. Roetting, “Estimating driver’s lane-change intent considering driving style and contextual traffic,” *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 38–3271, Sep. 2019.
- [12] T. Gindele, S. Brechtel, and R. Dillmann, “Learning driver behavior models from traffic observations for decision making and planning,” *IEEE Intell. Transp. Syst. Mag.*, vol. 7, no. 1, pp. 69–79, 2015.
- [13] M. Brand, N. Oliver, and A. Pentland, “Coupled hidden Markov models for complex action recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 1997, pp. 994–999.
- [14] Y. Huang, H. Wang, A. Khajepour, H. Ding, K. Yuan, and Y. Qin, “A novel local motion planning framework for autonomous vehicles based on resistance network and model predictive control,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 55–66, Jan. 2020.
- [15] D. Lee, Y. P. Kwon, S. McMains, and J. K. Hedrick, “Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC,” *Proc. IEEE Conf. Intell. Transp. Syst.*, vol. 2018, Mar. 2018, pp. 1–6.
- [16] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [17] Y. Jeong, S. Kim, and K. Yi, “Surround Vehicle Motion Prediction Using LSTM-RNN for Motion Planning of Autonomous Vehicles at Multi-Lane Turn Intersections,” in *IEEE Open J. Intell. Transp. Syst.*, vol. 1, 2020, pp. 2–14, doi: [10.1109/OJITS.2020.2965969](https://doi.org/10.1109/OJITS.2020.2965969).
- [18] L. Tang, H. Wang, W. Zhang, Z. Mei, and L. Li, “Driver lane change intention recognition of intelligent vehicle based on long short-term memory network,” *IEEE Access*, vol. 8, pp. 136898–136905, 2020.
- [19] T. Han, J. Jing, and Ü Özgür, “Driving intention recognition and lane change prediction on the highway,” *IEEE Intell. Vehicles Symp. (IV)*, 2019, pp. 957–962, doi: [10.1109/IVS.2019.8813987](https://doi.org/10.1109/IVS.2019.8813987).
- [20] Y. Chen, C. Hu, and J. Wang, “Human-centered trajectory tracking control for autonomous vehicles with driver cut-in behavior prediction,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8461–8471, Sep. 2019.
- [21] T. Rehder, A. Koenig, M. Goehl, L. Louis, and D. Schramm, “Lane change intention awareness for assisted and automated driving on highways,” *IEEE Trans. Intell. Veh.*, vol. 4, no. 2, pp. 265–276, Jun. 2019.
- [22] A. Zyner, S. Worrall, and E. Nebot, “A recurrent neural network solution for predicting driver intention at unsignalized intersections,” *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1759–1764, Jul. 2018.
- [23] K. Min, D. Kim, J. Park, and K. Huh, “RNN-based path prediction of obstacle vehicles with deep ensemble,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10252–10256, Oct. 2019.
- [24] A. Zyner, S. Worrall, and E. Nebot, “Naturalistic driver intention and path prediction using recurrent neural networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1584–1594, Apr. 2020.
- [25] L. Xin, P. Wang, C. Y. Chan, J. Chen, S. E. Li, and B. Cheng, “Intention-aware long horizon trajectory prediction of surrounding vehicles using dual LSTM networks,” in *Proc. IEEE Conf. Intell. Transp. Syst.*, vol. 2018, Nov. 2018, pp. 1441–1446.
- [26] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
- [27] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [28] R. Dey and F. M. Salemt, “Gate-variants of gated recurrent unit (GRU) neural networks,” in *Proc. IEEE Int. Midwest Symp. Circuits Syst.*, 2017, pp. 1597–1600.
- [29] J. Suh, H. Chae, and K. Yi, “Stochastic model-predictive control for lane change decision of automated driving vehicles,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 4771–4782, Jun. 2018.
- [30] E. Tian, “Driver characteristics in the lane change process,” *Proc. Int. Conf. Remote Sens., Environ. Transp. Eng.*, 2011, pp. 3093–3096.
- [31] C. L. Hwang, K. C. Lai, F. A. Tillman, and L. T. Fan, “Optimization of system reliability by the sequential unconstrained minimization technique,” *IEEE Trans. Rel.*, vol. R- 24, no. 2, pp. 133–135, Jun. 1975.
- [32] L. S. Jin, W. P. Fang, Y. N. Zhang, S. B. Yang, and H. J. Hou, “Research on safety lane change model of driver assistant system on highway,” in *Proc. IEEE Intell. Veh. Symp.*, 2009, pp. 1051–1056.
- [33] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, “Probabilistic trajectory prediction with gaussian mixture models,” in *Proc. IEEE Intell. Veh. Symp.*, 2012, pp. 141–146.

- [34] H. Huang, L. P. Bi, H. T. Song, and Y. C. Lu, "A variational EM algorithm for large databases," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Aug. 2005, pp. 3048–3052.
- [35] Z. Zhang, "Improved adam optimizer for deep neural networks," in *Proc. IEEE/ACM 26th Int. Symp. Qual. Serv.*, 2018, pp. 1–2.
- [36] S. Yoon, H. Jeon, and D. Kum, "Predictive cruise control using radial basis function network-based vehicle motion prediction and chance constrained model predictive control," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3832–3843, Oct. 2019.

**Chunyan Wang** received the B.S. and Ph.D. degrees in mechanical engineering from Jilin University, Changchun, China, in 2000 and 2008, respectively. He is currently an Associate Professor with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research focuses on vehicle system dynamics.



**Lin Li** received the B.S. degree in vehicle engineering from Jiangsu University, Zhenjiang, China, in 2018. She is currently working toward the Ph.D. degree with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. Her research interests include decision making and motion prediction of intelligent vehicle.



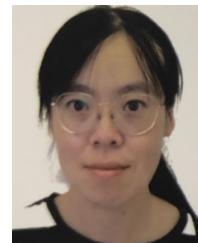
**Qingyun Chen** received the B.S. degree in vehicle engineering from Nanjing Forestry University, Nanjing, China, in 2017. He is currently working toward the Ph.D. degree with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research focuses on the decision making and control of intelligent vehicle.



**Wanzhong Zhao** received the B.S. and M.S. degrees in vehicle engineering from Jiangsu University, Zhenjiang, China, in 2004 and 2005, respectively, and the Ph.D. degree in vehicle engineering from the Beijing Institute of Technology, Beijing, China, in 2009. He is currently a Professor and the Director with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research focuses on vehicle system dynamics.



**Shijuan Dai** received the Ph.D. degree from Southeast University, Nanjing, China, in 2014. She is currently a Senior Experimentalist with the Nanjing University of Science and Technology, Nanjing, China. Her research interests include vehicle design and vehicle materials.



**Can Xu** received the B.S. degree in agricultural mechanization engineering from Nanjing Agricultural University, Nanjing, China, in 2017. He is currently working toward the Ph.D. degree with the Department of Vehicle Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research focuses on the decision making and control of intelligent vehicle.

