Journal of Advanced Transportation

Vehicle Trajectory Prediction by Knowledge-driven LSTM Network in Urban Environments

Shaobo Wang,[1,2,3,4] Pan Zhao,[1,3,4] Biao Yu,[1,3,4] Weixin Huang,[1,2,3,4] and Huawei Liang,[1,3,4]

[1] Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China/P. R.China.
[2] University of Science and Technology of China, Hefei 230026, China/P. R.China.
[3] Anhui Engineering Laboratory for Intelligent Driving Technology and Application, Hefei, Anhui, China.
[4] Innovation Research Institute of Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Hefei, Anhui, China.

Correspondence should be addressed to Pan Zhao and Huawei Liang; pzhao@hfcas.ac.cn; hwliang@iim.ac.cn

Abstract

An accurate prediction of future trajectories of surrounding vehicles can ensure safe and reasonable interaction between intelligent vehicles and other types of vehicles. Vehicle trajectories are not only constrained by a priori knowledge about road structure, traffic signs, and traffic rules but also affected by posterior knowledge about different driving styles of drivers. The existing prediction models cannot fully combine the prior and posterior knowledge in the driving scene and perform well only in a specific traffic scenario. This paper presents a long short-term memory (LSTM) neural network driven by knowledge. First, a driving knowledge base is constructed to describe the prior knowledge about a driving scenario. Then, the prediction reference baseline (PRB) based on driving knowledge base is determined by using the rule-based online reasoning system. Finally, the future trajectory of the target vehicle is predicted by an LSTM neural network based on the prediction reference baseline, while the predicted trajectory considers both posterior and prior knowledge without increasing the computation complexity. The experimental results show that the proposed trajectory prediction model can adapt to different driving scenarios and predict trajectories with high accuracy due to the unique combination of the prior and posterior knowledge in the driving scene.

1. Introduction

Since the 1980s, autonomous vehicles have been regarded as effective solutions to the problems of road safety, traffic congestion, and energy crisis. However, autonomous vehicles still face many driving difficulties in the real urban traffic environment. A major problem is how to interact safely and reasonably with other types of vehicles in a driving scene. Experienced human drivers can predict the future trajectory of other vehicles in a driving scene, thereby making safe, reasonable, and efficient decisions. Accurately predicting the future trajectory of a vehicle not only can reduce or eliminate the collision risk when autonomous vehicles perform complex driving maneuvers, such as merge, lane change, and overtaking, but

also can improve the driving efficiency and comfort of autonomous vehicles [1]. In a real urban traffic scenario, the vehicle's driving trajectory is not only constrained by prior knowledge, such as that about the road structure, traffic signs, and traffic rules, but also by uncertain posterior knowledge, including subjective driving intentions of the driver. The influence of driving knowledge on vehicle trajectory is shown in Figure 1, where it can be seen that when the road structure constraints are not considered, the predicted future trajectory, denoted as the red curve, is incorrect. As shown in Figure 1(b), there is a large slow-moving truck in front of the target vehicle. In such a case, based on human driving experience, the target vehicle is likely to adopt a lane change strategy. Therefore, how to fully combine the prior and posterior knowledge in a driving scene in the prediction process is crucial for accuracy improvement of the long-term trajectory prediction and safe interaction with other vehicles.

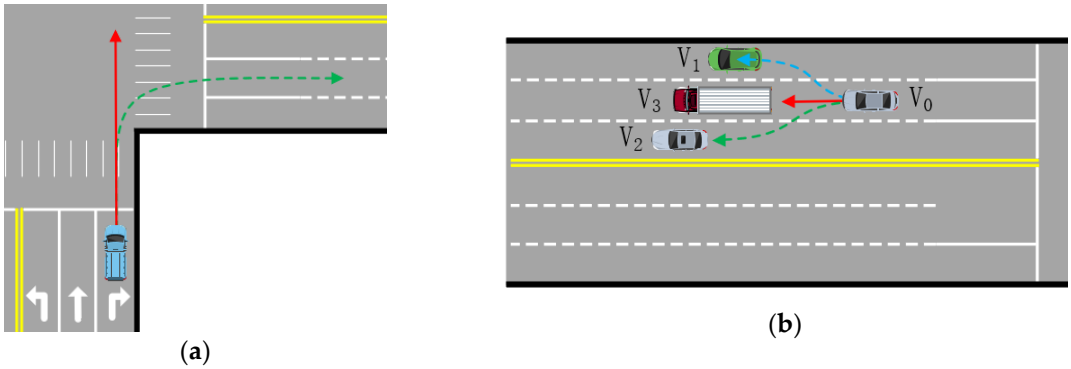


Figure 1. Influence of driving knowledge on trajectory prediction: (a) Influence of road structure on trajectory prediction; (b) Influence of driving experience on trajectory prediction.

According to the specific prediction process, the existing prediction models can be roughly divided into three categories: physics-based models, maneuver-based models, and learning-based models [2]. The physics-based models use vehicle kinematics and dynamics model to predict the future position of a target vehicle, and they include the Constant Turn Rate and Acceleration model [3], Switching Kalman Filters [4], Monte Carlo simulation [5]. However, these models ignore the prior and posterior knowledge about a driving scenario, such as road structure, traffic rules, and driver's subjective intentions, which limits these models to short-term prediction (less than 1 s) [6].

Maneuver-based models divide the prediction process into two parts. First, driving intention is estimated according to the physical state of a vehicle, information about the road network, and driver behavior, and then the predicted trajectory is fitted based on the driving intention. For maneuvers classification in more complex scenarios, discriminative learning algorithms, including the Multi-Layer Perceptions (MLP) [7], Logistic regression [8], Relevance Vector Machines (RVM) [9], and Support Vector Machines (SVM) [10], have been very popular. Complex vehicle motion is decomposed into predefined driving action sequences, which makes driving intention easier to identify and classify, and the prediction result is more stable and accurate than that of the physics-based models, and the prediction horizon is longer. However, in complex traffic scenarios, the traditional algorithms, such as finite vector machines and conditional random fields, have the problem of low scene adaptability, while Bayesian network and Markov model can solve the problem of driving maneuver classification in uncertain environments. In addition, the state space of the above models is extremely large, and these models are prone to "curse of dimensionality", and unable to real-time prediction. Recently, artificial neural networks have been used to classify vehicle driving actions, but the existing high-quality calibrated datasets are limited and cannot cover all possible driving scenarios (data

sparsity) [11], which makes the network training difficult and challenging, and scene adaptability is low.

Learning-based models skip the step of maneuver recognition and perform trajectory prediction directly based on the historical observation of a target vehicle, so the posterior knowledge in driving scenarios can be effectively learned, and incorrect driving motion recognition can be avoided. Recently, artificial neural networks have been used to predict future trajectories of vehicles, bicycles, and pedestrians [12–14]. As a type of recurrent neural network (RNN), the long short-term memory (LSTM) neural network has been proven to be very effective in solving the time series problems, and thus have been widely used in pedestrian trajectory prediction, intersection vehicle destination prediction, and highway vehicle trajectory prediction. However, in previous works, specific-scenario models, such as lane change models for non-intersection sections and left/right turn models for intersection areas, have been proposed [15–17], and the training data needed manual annotation, which increased the training difficulty of the model. In [18], an encoder-decoder LSTM model is proposed for predicting vehicle trajectory by using an occupancy grid map, the maximum prediction horizon of this model is two seconds, which is not sufficient for applications.

When an intelligent vehicle is driving in a real urban environment, the driving scene changes dynamically over time, which means that the prediction model should automatically adapt to a driving scene. In order to solve the problem of vehicle adaptability to the driving scene, many studies incorporated the Maneuver-based and Learning-based models. In [19], two LSTMs were used to identify high-level driver intentions and analyze low-level complex vehicle motion dynamics. This method is better geography-adaptive than the traditional LSTM networks. An LSTM model for interaction aware motion prediction of surrounding vehicles on freeways was presented in [20]. This model assigns confidence values to maneuvers being performed by vehicles and outputs a multi-modal distribution over future motion based on these values. The mentioned methods predict the multi-modal trajectory based on maneuver classes, which improves the road adaptability, but the prior knowledge in driving scenarios is not used. In [21], a long short-term memory (LSTM) network was employed to anticipate the driving policy of a vehicle (such as forward, yield, turn left, and turn right) using its sequential history observations. The policy was then used to guide a low-level optimization-based context reasoning process. This method combines the prior knowledge in the driving scene and constructs the cost map to perform the second optimization of the previously obtained driving intention to generate the final predicted trajectory, but the driving intention estimation of the upper-level does not utilize the prior knowledge of the driving scene, and the weight of the function cannot be adjusted adaptively to a driving scenario. Deo and Trivedi [22] adopt a convolutional social pooling LSTM based model. This approach predicts a distribution of future vehicle trajectory dependent on maneuver, but this approach ignores the impact of the interaction of the road users. Dai et al. [23] proposed a Spatio-Temporal LSTM based model, which considers the spatial interactions of the surrounding vehicles, but the constraints of other prior knowledge like road structure, traffic rules and driving experience are not considered. The Dual Learning Model (DLM) which takes information from two different inputs to predict vehicle trajectory was presented in [24]. This model embeds the Occupancy Map and Risk Map into the trajectory model to consider a comprehensive definition of risk in the traffic scene, but the computational complexity usually grows exponentially if the dimensionality of the feature space increases. Thus, it becomes difficult to meet the on-line requirement.

In this article, an integrated trajectory prediction model, which combines knowledge reasoning and LSTM neural networks, is proposed. The contribution of this study can be summarized as follows:

1. In order to consider the constraints of the prior knowledge. The Prediction Reference Baseline obtained by knowledge reasoning is introduced into the LSTM network, where the proposed model can effectively combine the prior knowledge without increasing the computation complexity.
2. In order to learn the spatial interactions of the surrounding vehicles and solve combinatorial explosion problem caused by a large number of condition attributes. A method of deterministic scene evaluation is employed to classify and analyze the main conditions that affect the future trajectory of a vehicle from the perspectives of safety, legitimacy, and reasonableness, which simplifies modeling of the spatial interactions.
3. In order to improve the adaptability of the proposed model. The Frenet coordinates based on the PRB are used to train the LSTM network, and it is not necessary to annotate the training data set manually according to the specific driving scenario. The results of the field test prove the adaptive performance of the proposed model.
4. The performance of the proposed model is evaluated with state-of-the-art methods on a naturalistic highway driving dataset (NGSIM), the results show that our proposed model outperforms the state-of-the-art methods.

The rest of the paper is organized as follows. The Prediction Reference Baseline determination method and the proposed LSTM network is presented in Section 2. The proposed prediction model is evaluated by both simulations and real-traffic urban roadways experiments, and the obtained results are presented and discussed in Section 3. Finally, the main conclusions, limitations, and future work are presented in Section 4.

2. Materials and Methods

2.1. Problem Formulation and Method Overview

2.1.1. Problem Formulation

The proposed trajectory prediction model is divided into two layers. The first layer determines the PRB of a target vehicle and the second layer predicts the future trajectory based on the PRB. PRB is a trajectory that indicates the driving intention of the target vehicle based on prior driving knowledge, which connect the on-line reasoning system and the LSTM network.

The process of driving intention prediction, for a target vehicle V_i at time t is presented in Figure 2, where it can be seen that it is necessary to understand and evaluate driving scene $S_{v_i}^t$, and generate the scene evaluation parameters $E_{v_i}^t$ of the target vehicle, including the safety assessment $SAFE_{v_i}^t$, legitimacy assessment $LEGAL_{v_i}^t$, and reasonable assessment $REASONABLE_{v_i}^t$, which is expressed as:

$$E_{v_i}^t = \left\{SAFE_{v_i}^t, LEGAL_{v_i}^t, REASONABLE_{v_i}^t\right\}. \quad (1)$$

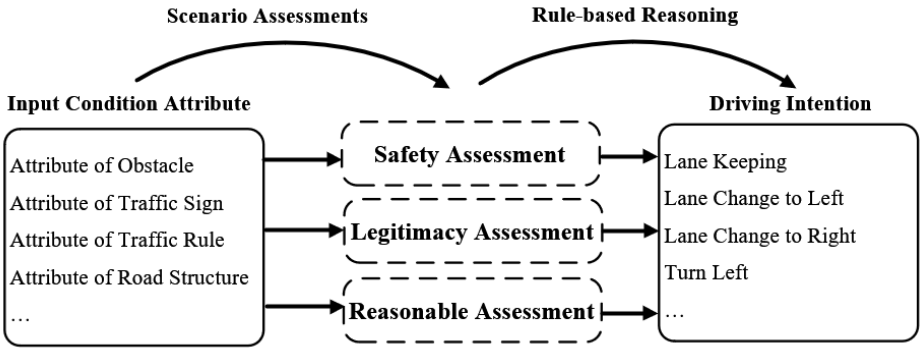


Figure 2. Driving intention prediction process.

According to the prior knowledge of driving scenarios, such as traffic rules and driving experience, the driving intention $b_{v_i}^t$ of the target vehicle is inferred based on the prolog online reasoning system. A maneuver $b_{v_i}^t$ is classified by the lateral movement of the vehicle, which is expressed by a finite set $\mathcal{B}$:

$$b_{v_i}^t \in \mathcal{B} := \{LK, LCL, LCR, TR, TL, GS, SS, \dots\}. \tag{2}$$

Finite set $\mathcal{B}$ includes the following maneuvers: lane keeping (LK), lane change to left (LCL), lane change to right (LCR), turn right (TR), turn left (TL), go straight at intersection (GS), stop before the stop line (SS).

Finally, driving intention $b_{v_i}^t$ is fitted to the PRB $PRB_{v_i}^t$ by the cubic Bezier curves.

The second layer predicts the future vehicle trajectory. First, the coordinate transformation is performed on the historical trajectory of the target vehicle based on the PRB $PRB_{v_i}^t$, as shown in Figure 3. In Figure 3, $s_{v_i}^t$ denotes the distance the target vehicle has traveled along the PRB, and $l_{v_i}^t$ is the transverse distance between the target vehicle and the $PRB_{v_i}^t$. The absolute position denoted as $(lat_{v_i}^t, lng_{v_i}^t)$ is transformed to the Frenet coordinates that is denoted as $(s_{v_i}^t, l_{v_i}^t)$. The set of observations vectors denoted as $O_{v_i}^{t(n)}$ is used for trajectory prediction of the target vehicle, and it is given by:

$$O_{v_i}^{t(n)} = \left\{s_{v_i}^t, l_{v_i}^t, \kappa_{v_i}^t, \theta_{v_i}^t, v_{v_i}^t, a_{v_i}^t\right\},\ for\ n = t - M + 1, \cdots, t, \tag{3}$$

where a set $(s_{v_i}^t, l_{v_i}^t)$ denotes the Frenet coordinates, $\kappa_{v_i}^t$ denotes the curvature, $\theta_{v_i}^t$ represents vehicle heading, $v_{v_i}^t$ denotes the vehicle speed, $a_{v_i}^t$ denotes the vehicle acceleration, and M is the input step of the network.

The network output $P_{v_i}^{t(n)}$ is expressed as:

$$P_{v_i}^{t(n)} = \left\{s_{v_i}^t, l_{v_i}^t, v_{v_i}^t\right\},\ \ for\ n = t + 1, t + 2, \cdots, t + K, \tag{4}$$

where K denotes the output step of the network.

Finally, the predicted trajectory denoted as $Tra_{v_i}^{t(n)}$ is obtained by transforming the reference coordinates to the absolute coordinates represented by the latitude and longitude, which is expressed as:

$$Tra_{v_i}^{t(n)} = \left\{lat_{v_i}^t, lng_{v_i}^t, v_{v_i}^t\right\}, \quad for\ n = t+1, t+2, \cdots, t+K, \tag{5}$$

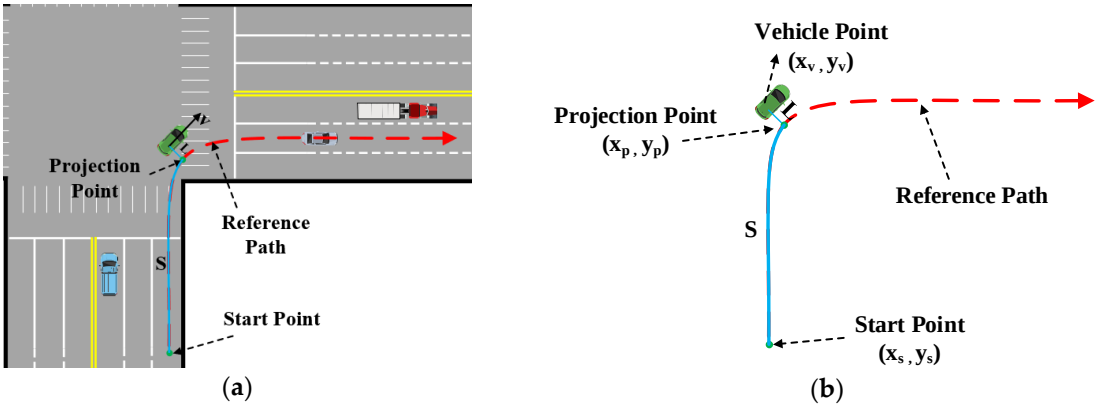


Figure 3. Frenet coordinate of a vehicle: (a) Driving scenario; (b) Description of Frenet coordinate.

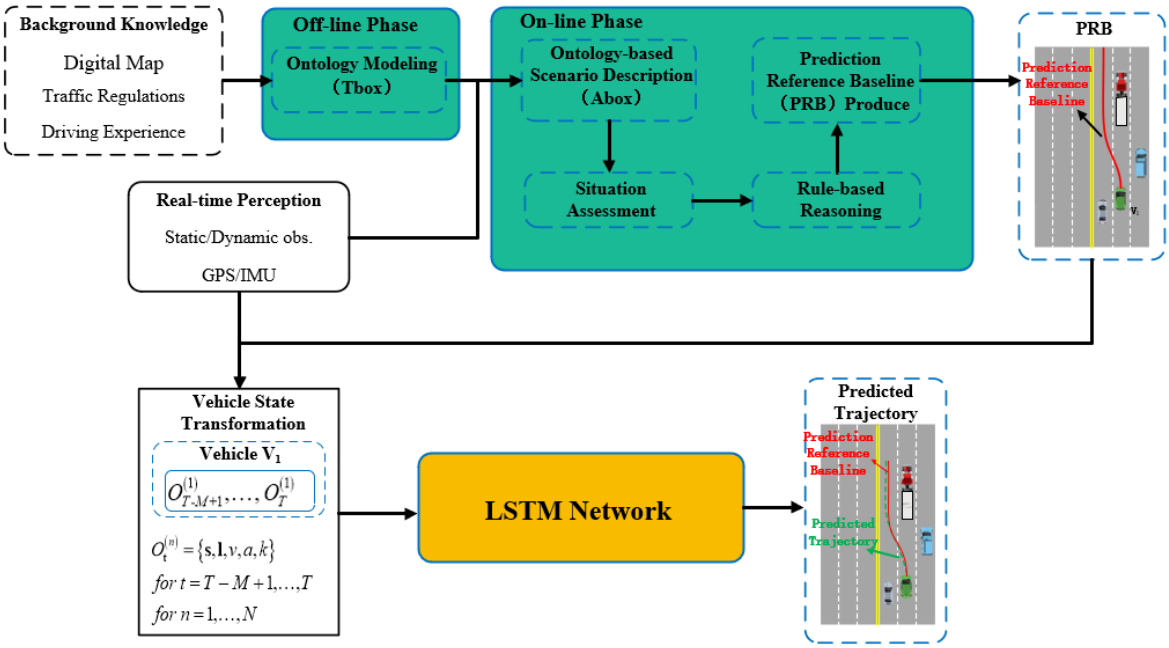


Figure 4. Overview of the proposed trajectory prediction model.

2.1.2. Overview of the Proposed Approach

This paper proposes a trajectory prediction model based on knowledge reasoning and LSTM neural network. The architecture of the proposed model is shown in Figure 4, where it can be seen that the proposed model consists of two phases: PRB determination phase and trajectory prediction phase. During the PRB determination phase, by analyzing the relationship between "human-vehicle-road" in the driving scene and extracting the knowledge of road network, traffic participants, and road traffic facilities, the conceptual ontology model of the driving scene is established. The main conditional attributes that affect the behavior decision-making process are classified and analyzed from the perspectives of safety, legitimacy, and

reasonableness using the proposed deterministic situation assessment method, and situation parameters in the horizontal and vertical directions are obtained. The behavior prediction rule base is constructed using the situation parameters, traffic rules, and driving experience. Based on the prolog online reasoning system, the behavioral prediction rules are matched with the factual knowledge obtained by the conceptual ontology model, and the driving intentions are inferred. Finally, a third-order Bezier curve is used to fit the driving intention to a PRB. The trajectory prediction phase uses the LSTM network to learn the continuous features of the historical trajectory of a target vehicle on the basis of the PRB and generates the final predicted trajectory.

2.2. Prediction Reference Baseline Determination

The architecture of the proposed PRB Determination method is presented in Figure 5, where it can be seen that this method consists of online and offline phases. The offline phase establishes the conceptual ontology model (TBOX) of a driving scene and extracts the behavioral prediction rules based on traffic rules and driving experience. According to the conceptual ontology model, the road network and real-time environment perception information are used to instantiate the entities and related relationships in the driving scene (ABOX). The entities and entity relationships in the driving scene are classified by the deterministic scene assessment method and analyzed from the perspectives of safety, legitimacy, and reasonableness. The scene evaluation parameters in both horizontal and vertical directions are generated, and the behavioral prediction rules are matched with the scene evaluation parameters by using the prolog online reasoning system. The driving intentions are inferred, and finally, the third-order Bézier curve is used to fit the driving intention to the PRB.

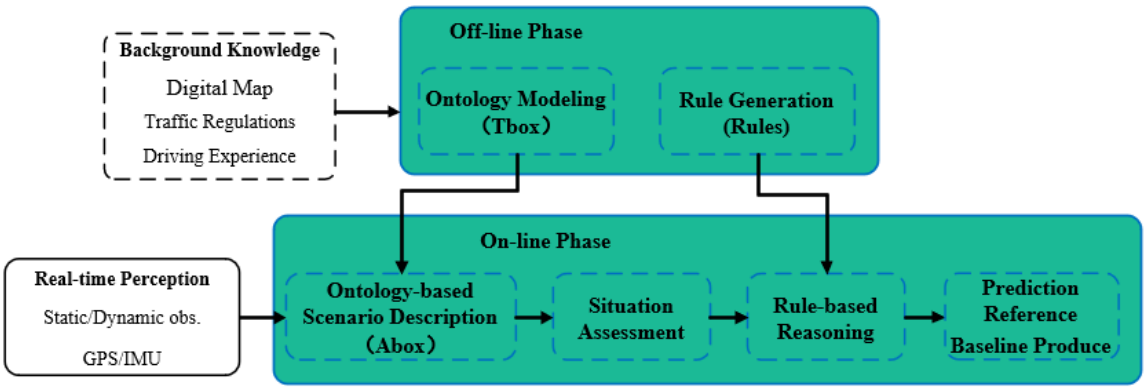


Figure 5. The architecture of the proposed Prediction Reference Baseline Determination method.

2.2.1. Semantic Modeling of Driving Scene

In a driving scenario, there are various road element entities, such as traffic participants, road networks, and road traffic facilities in urban driving scenarios. The environment perception system can provide only the spatial location of each entity, but it cannot describe the correlation between entities, and make full use of prior information, such as traffic rules and driving experience, which is crucial for improving the prediction model adaptability to the driving scene. Ontology, as a form of knowledge expression, is used to model the concepts of specific domains and relationships between concepts, which can be used to model driving scenarios effectively [25–27].

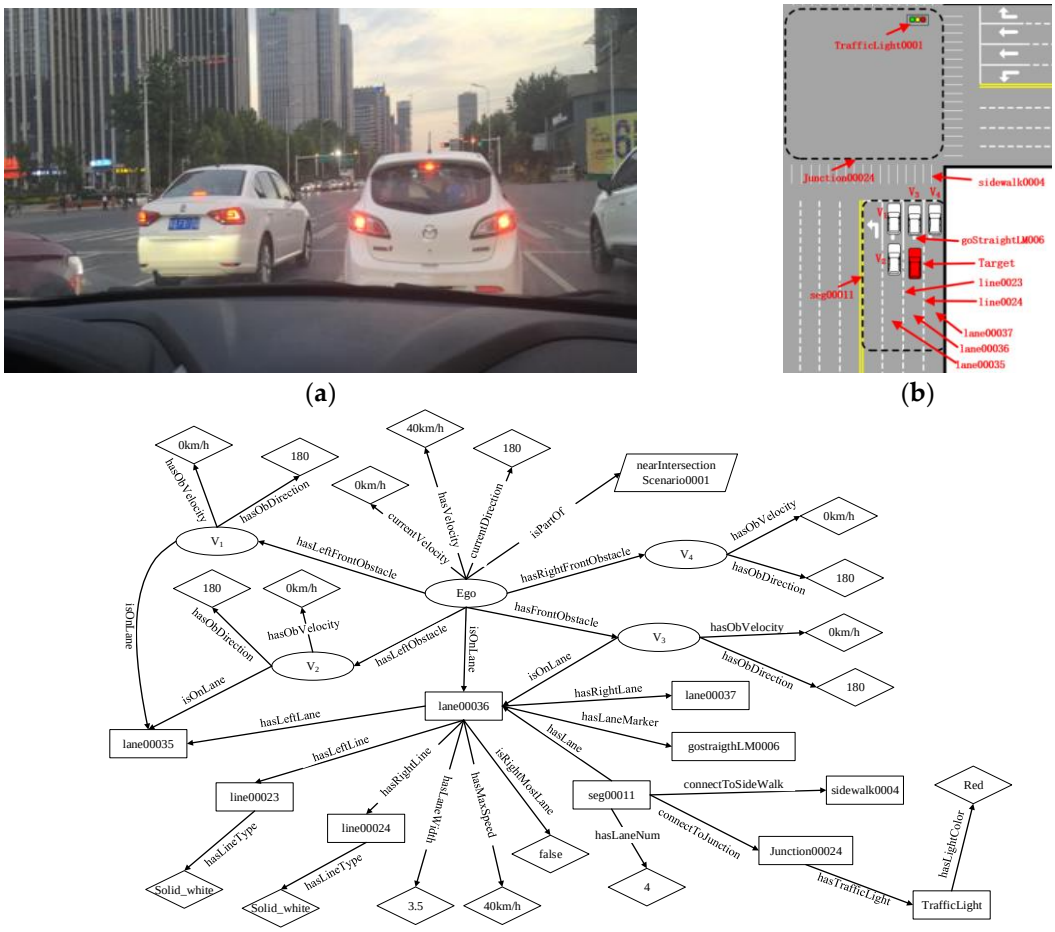


Figure 6. Scene modeling: (a) Real traffic scenario; (b) Concrete driving scene; (c) Semantic description of the concrete scene.

The conceptual ontology model is divided into two module types: entities and attributes. This study takes the target vehicle as a perspective and summarizes five entity types on the basis of [28]:

1) Target Vehicle

The target vehicle entity describes the vehicle to be predicted.

2) Behavior

The behavior entity is a collection of driving maneuvers of a vehicle. Three behavior types are designed: LongtiBehavior, LatiBehavior, and AdvancedBehavior. The LongtiBehavior represents basic vertical driving behavior and includes four behaviors: accelerate, decelerate, keep, and stop. The LatiBehavior represents basic horizontal driving behavior and includes three behaviors: ChangeToLeft, ChangeToRight, and KeepLane. The AdvancedBehavior represents advanced driving behavior and includes two behaviors: Overtake and Merge.

3) Obstacle

The obstacle entity represents a collection of obstacle entities encountered by a vehicle during driving. This work divides obstacles according to the behavior characteristics of obstacle entities in driving scenarios into two categories: StaticObstacle and DynamicObstacle.

4) Road network

The road network entity represents the topological connection of roads by intersecting points and lines. RoadType includes different road types. RoadPart describes the components of the road network and is divided into AreaEntities and PointEntities. AreaEntities refer to road entities that can be abstracted into lines and areas, such as lane, side walk, junction, and segment, while PointEntities refers to road entities that can be abstracted into points, such as road signs, traffic signs, and traffic lights.

5) Driving Scenario

The driving scene entity refers to a collection of road entity elements encountered when a vehicle travels in different road areas. In this work, driving scenarios are divided into three categories: InSpecialAreascenario (special area driving scenario), OnRoadscenario (road driving scenario), and NearSpecialAreascenario (near special region driving scenario). InSpecialAreascenario category can be further divided into IntersectionScenario (intersection scene), TunnelScenario (tunnel scene), BridgeScenario (elevated scene), and UturnScenario (UTurn scene).

The object attribute is used to describe the relationship between concept classes. This attribute restricts the described relationship regarding the domain and range. The data attribute restricts the described relationship through the definition and value domains. The definition domain is a class type.

The described ontology modeling process of driving scenario is equivalent to filling the background knowledge of the TBox that constitutes the ontology knowledge base, but the situational knowledge in the ABox is still lacking. According to the road elements of a real driving scenario, the driving scenario needs to be re-expressed using the conceptual model of the TBox, which is an instantiation of the ontology model. A real traffic scenario is displayed in Figure 6(a); a concrete driving scene that includes instances of defined classes is presented in Figure 6(b), and its semantic description is presented in Figure 6(c). The instances of RoadNetwork are added to the ABox as prior knowledge, and instances of the obstacle are asserted in real time.

2.2.2. Situation Assessment

After obtaining a semantic description of a driving scene, it is necessary to determine and evaluate condition attributes that affect the driving intention in the driving scene, so as to estimate the driving intention of a target vehicle. In order to solve the problem of combinatorial explosion due to numerous condition attributes [29], a deterministic scenario assessment method is adopted to classify and analyze the key attributes that affect driving intentions from the perspectives of safety, legitimacy, and reasonableness.

Deterministic scenario assessment methods use the threat assessment indicators: TTC (time to collision), THW (time headway), TTB (time to brake), DST (deceleration to safety time), and MSM (minimal safety margin) in rule-based systems, and the probability of collision is

estimated as a binary value. For instance, Glaser et al. [30] used the TTC and TIV (time intervehicles) indicators to evaluate the possibility of collision. Samyeul et al. [31] proposed a distributed reasoning method by dividing the current and adjacent lanes into the front and rear areas, and the TTB and MSM indicators were used to evaluate the possibility of collision in the front area, while the TTC and MSM indicators were used to evaluate the collision of rear area collision possibility.

The proposed deterministic scenario assessment method consists of two parts. First, the driving scenario is determined by querying the knowledge base with the current vehicle position. Then, a reasoning structure of an obstacle is constructed in eight regions of interest to make safety assessment, and a binary result (safe or dangerous) is calculated for each region using critical indicators TTC and TIV. Finally, legitimacy and reasonableness assessments are made to predict the maneuver of the target vehicle.

1) Safety assessment

Safety primarily refers to whether the surrounding obstacles pose a threat to a vehicle, especially in the area ahead, but it also refers to whether the left or right lane can provide a safe lane change. This paper constructs the eight-direction obstacle inference model. For each area, the TTC and TIV indicators are used for safety assessment. The TTC indicator is defined as a time when two vehicles continue to collide on the same trajectory at the current speed, and it is defined by:

$$TTC = \frac{D_i}{V - V_i}, \tag{6}$$

where D_i denotes the relative distance between the following vehicle and followed vehicle, V denotes the speed of the following vehicle, and V_i is the speed of followed vehicle.

The threshold value TTC_{th} is used to judge whether a vehicle is dangerous in high-speed scenarios. The risk assessment formula is as follows:

$$R_{TTC}(t) = \begin{cases} 0, & t \geq TTC_{th} \\ 1, & t < TTC_{th} \end{cases}. \tag{7}$$

when the calculated collision time t between the following vehicle and followed vehicle is greater than TTC_{th}, the current scene is considered to be safe; otherwise, it is considered to be dangerous.

The TIV indicator is used to detect low-speed difference scenarios. When the speeds of two vehicles are similar in value, the TIV indicator is used to judge the degree of danger, and it is calculated by:

$$TIV = \frac{D_i}{V}. \tag{8}$$

threshold TIV_{th} is used to distinguish between safe and dangerous scenes in low workshop distance scenes. The TIV risk assessment formula is as follows:

$$R_{TIV}(t) = \begin{cases} 0, & t \geq TIV_{th} \\ 1, & t < TIV_{th} \end{cases}. \tag{9}$$

when the calculated vehicle interval time t between the following vehicle and followed vehicle is greater than TIV_{th}, the current scene is considered to be safe; otherwise, it is considered to be dangerous, and in that case, the following car needs to perform a certain action to avoid a possible collision.

In each region, only when $R_{TTC}(t)$ and $\mathrm{R}_{TIV}(t)$ are calculated safety synchronously, the region is considered to be safe. The risk assessment of an region is determined as:

$$r_{Region}(t) = \begin{cases} 1, \mathrm{R}_{TTC}^{Region}(t) + \mathrm{R}_{TIV}^{Region}(t) \geq 1 \\ 0, \quad \text{otherwise} \end{cases}, \tag{10}$$

where $Region$ represents one of the eight regions, as shown in Figure 7; when $r_{Region}(t)$ has a value of zero, the area is considered to be safe; and, when $r_{Region}(t)$ has a value of one, the area is considered to be dangerous.

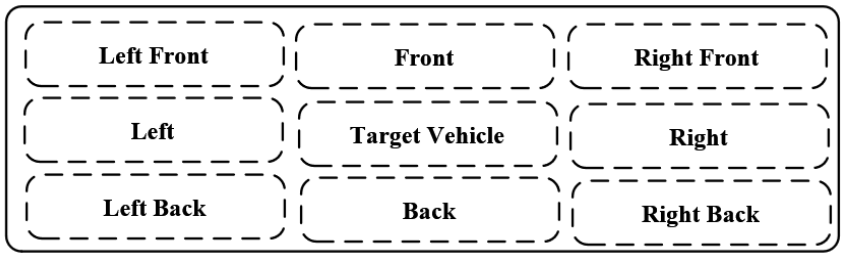


Figure 7. Example individuals in traffic scenario.

The degree of danger in the area ahead is calculated by:

$$r_F(t) = \begin{cases} 1, & \mathrm{R}_{TTC}^{F}(t) + \mathrm{R}_{TIV}^{F}(t) \geq 1 \\ 0, & \text{otherwise} \end{cases}, \tag{11}$$

when the calculated values of the TTC and TIV indicators in the current area are greater than the predefined threshold, the following result message is obtained: safeToGo (targetVehicle, keep); when the calculated values of TTC and TIV indicators in the current area are both less than the predefined deceleration threshold, but greater than the corresponding parking threshold, the result message is: safeToGo (targetVehicle, dec), and in this case, a lane change can be performed to improve driving efficiency; otherwise, the result message is: safeToGo (targetVehicle, stop).

The safety assessment of adjacent lanes is conducted using the same assessment formula as that of the area ahead. If the left lane is taken as an example, then the degree of danger is expressed as:

$$r_L(t) = \begin{cases} 1, & \text{has left vehicle at time } t \\ 0, & \text{otherwise} \end{cases}, \tag{12}$$

when there are vehicles in the left area, the left lane can be considered to be dangerous; otherwise, the safety of the left front and left rear areas are respectively evaluated by:

$$r_{FL}(t) = \begin{cases} 1, & \mathrm{R}_{TTC}^{FL}(t) + \mathrm{R}_{TIV}^{FL}(t) \geq 1 \\ 0, & \text{otherwise} \end{cases}, \tag{13}$$

$$r_{BL}(t) = \begin{cases} 1, & R_{TTC}^{BL}(t) + R_{TIV}^{BL}(t) \geq 1 \\ 0, & otherwise \end{cases}, \tag{14}$$

Therefore, the safety assessment of the left lane is as follows:

$$Risk_L(t) = \begin{cases} 1, & r_L(t) = 1 \ or \ r_{FL}(t) + r_{BL}(t) \geq 1 \\ 0, & otherwise \end{cases}, \tag{15}$$

If the left lane is safe, safeToLeft (targetVehicle, true) will be generated; otherwise, the scene evaluation parameters will be instantiated as safeToLeft (targetVehicle, false).

2) Legitimacy assessment

The legality assessment includes three assumptions. First, when a vehicle is driving on the road, it cannot exceed the maximum speed limit of the road; second, when the vehicle is driving to the pre-intersection, it is necessary to pay attention to the change in traffic lights and obey the traffic rules; third, when a vehicle is about to change the lane, the adjacent lane should allow lane changes.

3) Reasonableness assessment

Reasonableness assessment generally refers to whether lane changing and other driving behaviors affect the current goal of a target vehicle. Based on the current lane of the target vehicle, the specific road section or lane to be driven can be known. For instance, on the one hand, if the next area to be driven by the target vehicle is an intersection, and the distance between the target vehicle and the stop line is less than δ, then lane change is not recommended. On the other hand, if the distance between the target vehicle and the stop line is greater than δ, lane change can be performed. Reasonableness assessment introduces a situation parameter set $(reasonableToLeft,\ reasonableToRight)$, which is defined as data properties in the ontology model.

2.2.3. Rule-Based Reasoning

The driving intention is determined based on traffic rules and driving experience, where the traffic rules are mainly used to limit the driving behavior while the driving experience is utilized to summarize the understanding and cognition of human drivers in different scenes and obtain some rules that are not specific traffic rules but are conducive to the reasonable driving. According to the different driving scenarios defined in the driving knowledge base, the rules stored in the driving knowledge base are divided into several categories. Different scenarios have different key road entities. For instance, unlike OnRoadScenario ,in NearIntersectionScenario, traffic lights are considered. Also, the classification of traffic rules can reduce the rule search space and reasoning time.

In order to save computing resources and reduce reasoning time, SWI-Prolog language is used to write rule knowledge, which is represented as a set of driving scene-driving behavior mapping pairs, where driving behavior is described as a rule head, and the driving scene is described as a rule body. On the basis of [28], this paper adds the scene assessment as an

intermediate link of mapping driving scene to the driving behavior and reorganizes 57 rules. Some of the prediction rules are presented in Table 1.

The online reasoning process can be described as follows. First, the real-time facts related to the scene are used as input, and each rule statement is matched. If all the facts of the corresponding rule are matched, the matched prediction result will be obtained, and the next rule statement will be matched until each rule is matched. When all matched results are obtained, the final result denotes the predicted driving intention.

Table 1. Prediction rules used in this study.

ID	SWRL Rules
Rule #1	TargetVehicle(target), currentRoadState(target,"ApprJunction"), isOnSegment(target,Seg), connectToJunction(Seg,Junc), intersection(Junc), hasTrafficLight(Junc,TL), (hasLightColor(TL,"red"); hasLightColor(TL,"yellow")), connectToStopLine(Seg,SL), distToStopLine(SL,DL), DL < 10.
	legalToGo(target,SS)
Rule #2	TargetVehicle(target), currentRoadState(target,"ApprJunction"), isOnSegment(target,Seg), connectToJunction(Seg,Junc), intersection(Junc), hasTrafficLight(Junc,TL), hasLightColor(TL,"green"), connectToStopLine(Seg,SL), distToStopLine(SL,DL), DL =< 20.
	LegalToGo(target,acc)
Rule #3	TargetVehicle(target), currentVelocity(target,V), hasFrontObstacle(target,FO), distToObstacle(FO,DF), (DF/V) < 3, (DF/V) >= 2.
	SafeToGo(target,dec)

2.2.3. Prediction Reference Baseline Fitting

After obtaining the driving intention of the target vehicle, the driving intention is converted into the Prediction Reference Baseline using the cubic Bezier curves. As shown in Figure 8, first, a target lane is selected based on the driving intention and road network, where p_0 represents the current position of the target vehicle, and p_3 is selected from the centerline of the target lane with distance L_d from p_0, and L_d is obtained based on the driving experience. The Prediction Reference Baseline is divided into three parts by p_0 and p_3: the predicted extension, the historical extension, and the intention segment. In addition, L_p and L_h are determined by the input and output steps of the LSTM neural network.

As shown in Figure 8(c), the cubic Bezier curve constructed by four control points is used to generate the intention segment, which is expressed as:

$$C(t) = B_{0,3}(t)P_0 + B_{1,3}(t)P_1 + B_{2,3}(t)P_2 + B_{3,3}(t)P_3, \quad (16)$$

where $B_{i,3}$ is the Bernstein polynomial and it is given by:

$$B_{i,3}(t) = \binom{3}{i}\left(\frac{t_1 - t}{t_1 - t_0}\right)^{3-i}\left(\frac{t - t_0}{t_1 - t_0}\right)^{i}, t \in \langle 0,1,2,3 \rangle. \quad (17)$$

The coordinate system (X'', Y'') is built with the origin P_0 at the vehicle center. The x-axis direction is the vehicle's initial heading, the terminal state will be the end point P_3, and P_1 and

P_2 are obtained by moving forward for distance d along the vehicle's initial heading direction from the start point and backward for distance d along the terminal heading from the end point P_3, respectively. The position of the control points in the above coordinate system is expressed as:

$$\mathrm{P}_0 = \begin{bmatrix}0\\0\end{bmatrix}, \mathrm{P}_1 = \begin{bmatrix}d\\0\end{bmatrix}, \mathrm{P}_2 = \begin{bmatrix}L_x - d\cos\omega\\L_y - d\sin\omega\end{bmatrix}, P_3 = \begin{bmatrix}L_x\\L_y\end{bmatrix}, \tag{18}$$

where L_x and L_y are lateral and longitudinal offsets of the terminal state P_3 to P_0, respectively; ω is the angle between the terminal heading and the direction of the x-axis. The terminal heading is defined as the tangential direction of the closest point on the reference path to P_3.

Equation (16) and (17) can be rewritten by applying (18), so the Bezier curve can be represented as:

$$\mathrm{x}(t) = (3d + 3d\cos\omega - 2L_x)t^3 - 3(2d + d\cos\omega - L_x)t^2 + 3dt, \tag{19}$$

$$y(t) = \left(3d\,sin\,\omega - 2L_y\right)t^3 - 3\left(d\,sin\,\omega - L_y\right)t^2. \tag{20}$$

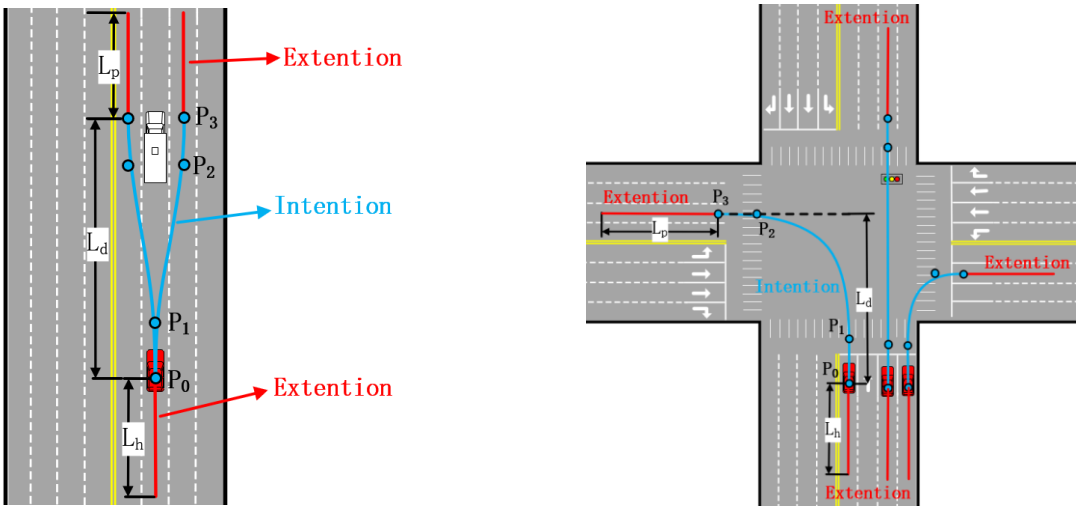
Besides, the curvature of the generated path can be derived by applying (20) and (21), which leads to:

$$\kappa(t) = \frac{x'(t)y''(t) - y'(t)x''(t)}{(x'(t)^2 + y'(t)^2)^{3/2}}, \tag{21}$$

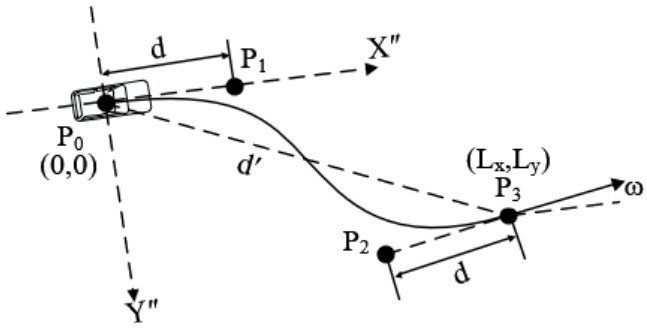
The maximum of the curvature should satisfy the condition given by equation (22) to meet the vehicle's nonholonomic constraint.

$$\kappa(t_m) \le \frac{\tan(\varphi_{max})}{L} \tag{22}$$

In equation (22), L denotes the vehicle wheelbase, and φ_{max} denotes the maximum steering angle of the vehicle.



(a) (b)



(c)

Figure 8. Schematic diagram of Prediction Reference Baseline fitting: (a) Lane-changing scene; (b) Intersection turning scene; (c) The cubic Bezier curves.

The maximum of the curvature $\kappa(t_m)$ is a function of d. The suitable value of d that satisfies the vehicle's nonholonomic constraint can be found by brutal searching from $\frac{d'}{6}$ to $\frac{d'}{2}$, where d' denotes the distance between P_0 and P_3. The processing time can be reduced by building a look-up table that matches a given set with the corresponding maximum curvature of the Bezier curve.

2.3. LSTM Network Driven by Knowledge

Since different drivers have different driving styles, in order to accurately predict the future trajectory of a vehicle, in this work, an LSTM neural network is employed to learn the continuous features of the historical trajectory. The LSTM is an RNN type that can effectively overcome the problem of gradient disappearance [32]. The LSTM is composed of a unit memory that stores the previous input sequence information and a gating mechanism that controls the information flow between input, output, and unit memory. There are three gates in the core design of the LSTM network, namely the input gate, the forget gate, and the output gate. The specific network structure is shown in Figure 9. The forget gate is used to control how much information in c_{t-1} is retained in c_t. The input gate determines how much information of x_t remains in c_t, and finally, the output gate determines how much information in the output o_t is output to h_t by the control unit c_t. The work of the LSTM is described by the following recursive equations:

$$f_t = \sigma(W_f \cdot [h_{t-1},\ x_t] + b_f), \tag{23}$$

$$i_t = \sigma(W_i \cdot [h_{t-1},\ x_t] + b_i)\ , \tag{24}$$

$$\widetilde{c_t} = \tan h\,(W_c \cdot [h_{t-1},\ x_t] + b_c), \tag{25}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t}, \tag{26}$$

$$o_t = \sigma(W_o \cdot [h_{t-1},\ x_t] + b_o), \tag{27}$$

$$h_t = o_t \odot \tan h(c_t)\ , \tag{28}$$

where x_t denotes the input vector, $\sigma(x)$ denotes the activation function, W denotes the linear transformation matrix, b denotes the offset vector; i_t, f_t, and $\widetilde{c_t}$ are gate vectors, c_t represents the amount of cell memory, and lastly, h_t denotes the output.

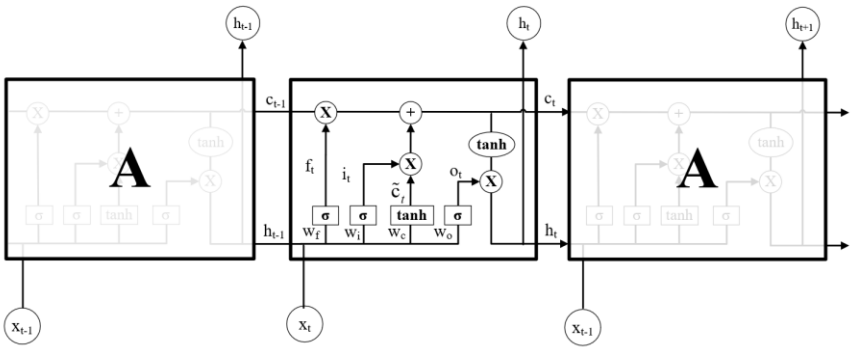


Figure 9. The internal structure of an LSTM cell.

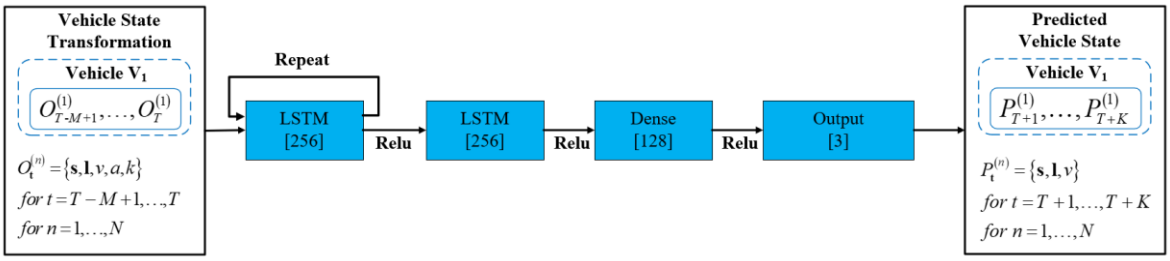


Figure 10. Network structure used as a reference design.

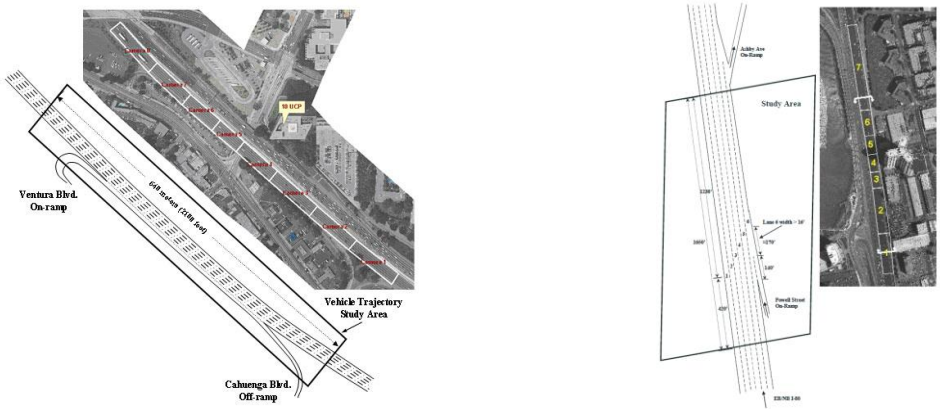
In this work, the network presented in Figure 10 is used as a reference structure. This network has two layers consisting of 256 LSTM cells, followed by one time-distributed layer consisting of 128 neurons, and the final dense output layer containing as many cells as the number of outputs. The network input is a tensor of track histories of a vehicle. The network output consists of the future coordinates and velocity of the vehicle. Since the prior knowledge about the driving scene is expressed by a priori reference trajectory, the network can learn the posterior knowledge about the driving scene only from the relative relationship between the historical trajectory and the prior reference trajectory. Compared with the existing prediction models based on the LSTM network, the proposed prediction model reduces the network training difficulty and decreases demand for the computing performance of the vehicle platform.

3. Results and Discussion

3.1. Data Preparation and Model Training

3.1.1. The training dataset

The Next Generation Simulation (NGSIM) dataset in I-80 and US101 sections are used for model training and testing [33], this dataset is derived from the US Federal Highway Administration, which is currently the largest public natural driving public data source, and thus has been widely used in the literature [34,35]. The layouts and top-down views of the US101 AND I-80 sections are shown in Figure 11. Each data frame includes many vehicle's parameters, including the position, velocity, yaw rate, size, and others. The sampling frequency of the dataset is 10 Hz; therefore, in this work, Δt is set to 0.1 s.



(**a**) (**b**)

Figure 11. Layouts and top-down views of the sites used for the collection of the NGSIM: (a) US101; (b) I80.

3.1.2. Data Preparation

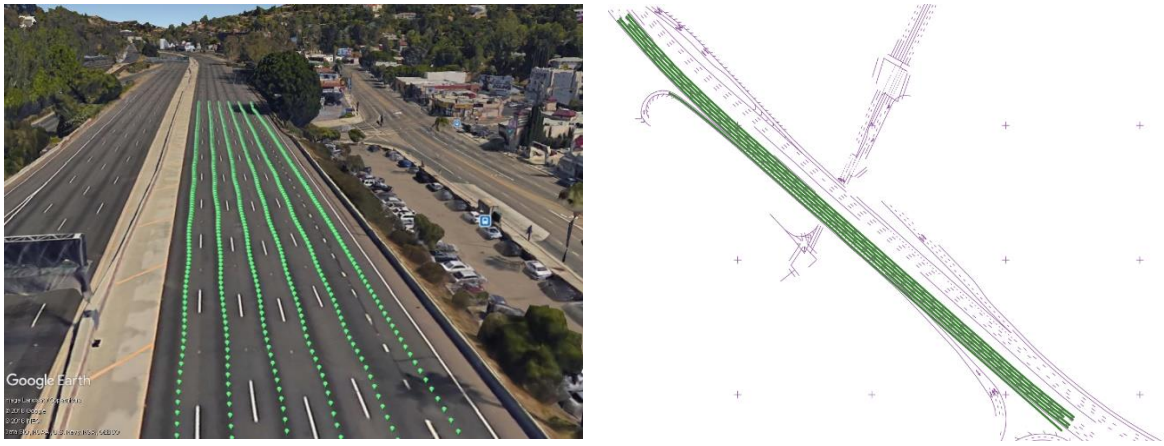
The vehicle positioning data in the NGSIM dataset are obtained by video analysis, so the recorded trajectory contains a lot of noise [27]. Therefore, the vehicle kinematics model and the road geometric are used to filter the original data, which is expresses as:

$$0 < \kappa_i < \kappa_{max}, \tag{29}$$

$$\theta_{min} < \theta < \theta_{max}, \tag{30}$$

$$\theta_{r_i} > \theta_{rate}, \tag{31}$$

$$\theta_{r_i} = \frac{(\theta_{i+1} - \theta_i)}{t_2 - t_1}. \tag{32}$$



(**a**) (**b**)

Figure 12. Extraction of the lane's centerline: (a) Extraction results obtained by Google Earth; (b) Shapefile of the US101 highways.

The vehicle position is transformed to the Frenet coordinates based on the centerline. As shown in Figure 12(b), the centerline of each lane is extracted and fitted using the shapefile, and the centerline that the target vehicle was initially driven is selected as a reference baseline. For each original coordinate point (x_v, y_v), the corresponding mapping point (x_p, y_p) on the reference baseline is determined, and the Frenet coordinates (s_v, l_v) are obtained by:

$$l_v = \sqrt{(x_p - x_v)^2 + (y_p - y_v)^2}, \tag{33}$$

$$s_v = S(x_p, y_p), \tag{34}$$

where l_v denotes the Euclidean distance between (x_v, y_v) and (x_p, y_p), $S(x_p, y_p)$ denotes the length from the mapping point (x_p, y_p) to the starting point of the reference trajectory.

In addition, four other features: curvature $\boldsymbol{\kappa}$, velocity $\boldsymbol{V}$, acceleration $\boldsymbol{a}$, and heading $\boldsymbol{\theta}$, are also selected so as to compose the observation vector $\boldsymbol{o}$ with the Frenet coordinates $(\boldsymbol{s}, \boldsymbol{l})$.

3.1.3. Training Details

There were 8311 filtered trajectories; 80% of the trajectories were selected as the training set, 10% as the test set, and the remaining 10% was used as the verification set to observe if the model is over-fitted.

The network was trained using mini-batches with a size of 64. Due to the limitation on a sensor measurement range and noise in practical application scenarios, it was difficult to track dynamic vehicles stably for a long time, so the network was trained using windows that consisted of 30 inputs, representing a total of 3s past observations. The ADAM optimizer was used; the learning rate was 0.0005 and ReLU activation with α = 0.1. The loss function adopted the MSE (mean square error) between the predicted sequence and the ground truth sequence; the code used to generate the model was written in Keras, and the training was performed on GPU using the TensorFlow backend with a batch size of 32; the model training contained 16 epochs.

3.2. Testing Results and Discussion

3.2.1. The Impact of the Prediction Reference Baseline

To investigate the impact of considering Prediction Reference Baseline on the accuracy of the proposed method. The RMSE performance of the proposed model is present with three modifications.

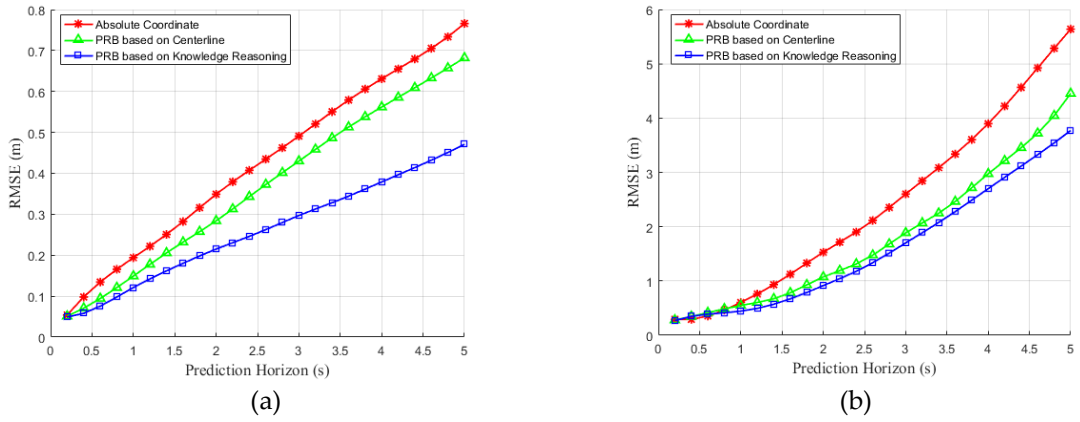


Figure 13. Effectiveness of considering Prediction Reference Baseline in the vehicle trajectory prediction: (a) Lateral position error; (b) Longitudinal position error.

In one experiment, the system is trained and tested with the absolute coordinate, in the second, the centerline that the target vehicle was initially driven is selected as a PRB, and finally in the third experiment, the PRB of the target vehicle is determined by the method in Section 2, while the other attributes of the three model are unchanged. Figure 13 shows the accuracy of the trajectory prediction for different time horizons, the RMSE value of the model is decreased by adding the PRB for both lateral and longitudinal trajectory.

3.2.2. Comparative study

To evaluate the proposed approach, we purse a direct comparison with state-of-the-art vehicle trajectory prediction using the same dataset (i.e., NGSIM). The results show that the proposed method outperforms the state-of-the-art model and decreases the overall RMSE value of the system by 10 percent on average. Table 2, summarizes the RMSE values comparing the proposed methods with the baseline trajectory prediction models in the literature [20], [22], [23], [24], [36].

The comparison results show that the proposed Knowledge-driven LSTM Network has better performance in RMSE for NGSIM. Note that as compared to the baseline [24], the prediction accuracy gets a slight improvement, but the proposed method enhances the real-time performance and much reduces the computational complexity due to the reduction of the feature space dimension.

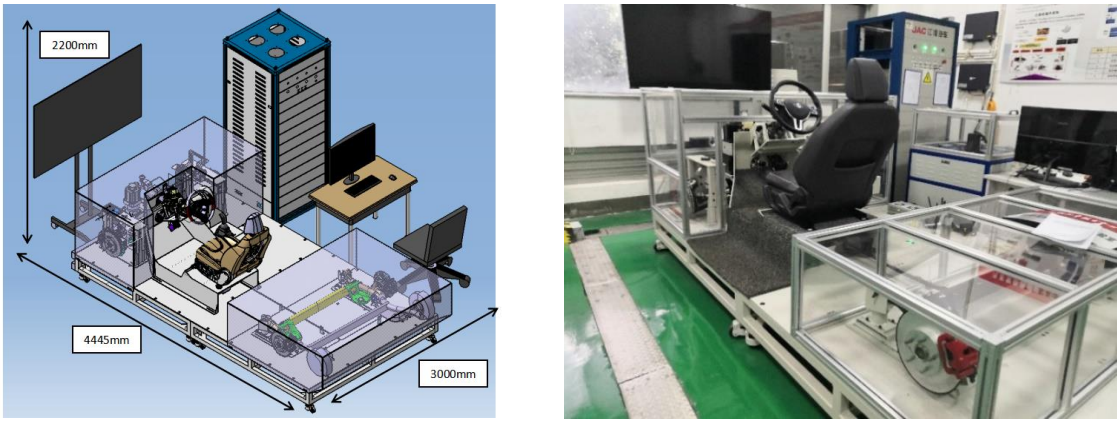
Table 2. RMSE comparison of the proposed method with the baseline models and state-of-the-art.

Prediction Horizon (s)	**M-LSTM [20]**	**CS-LSTM [22]**	**NLS-LSTM [36]**	**ST-LSTM [23]**	**DLM [24]**	**Proposed method**
1	0.58	0.61	0.56	0.58	0.41	**0.41**
2	1.26	1.27	1.22	1.21	0.95	**0.89**
3	2.12	2.09	2.02	1.97	1.72	**1.64**
4	3.24	3.10	3.03	2.85	2.64	**2.47**
5	4.66	4.37	4.30	3.89	3.87	**3.68**

3.3. Simulation Results and Discussion

3.3.1. Simulation Experimental Platform

The simulation experiments were based on the JAC's automatic driving hardware-in-the-loop test platform. The experimental simulation platform is presented in Figure 13, where it can be seen that the experimental platform included the real vehicle braking system, steering system, sensor system, and network communication system, which had dSPACE (Matlab/Simulink) as a core. The controller rapid prototyping platform was built, virtual reality interfaces and environment-aware sensor modules were provided using the PreScan software, and the CarSim software was used to run the vehicle dynamic model and provide a platform that could be quickly verified for automatic driving algorithm testing.



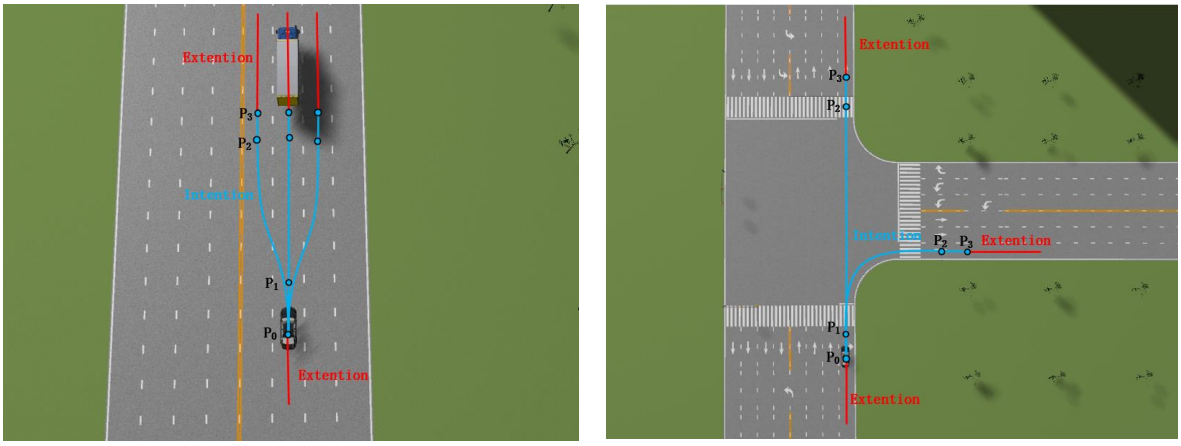
(**a**) (**b**)

553 Figure 14. Experimental simulation platform: (a) Design of the simulation platform; (b) Photo of the simulation
554 platform.

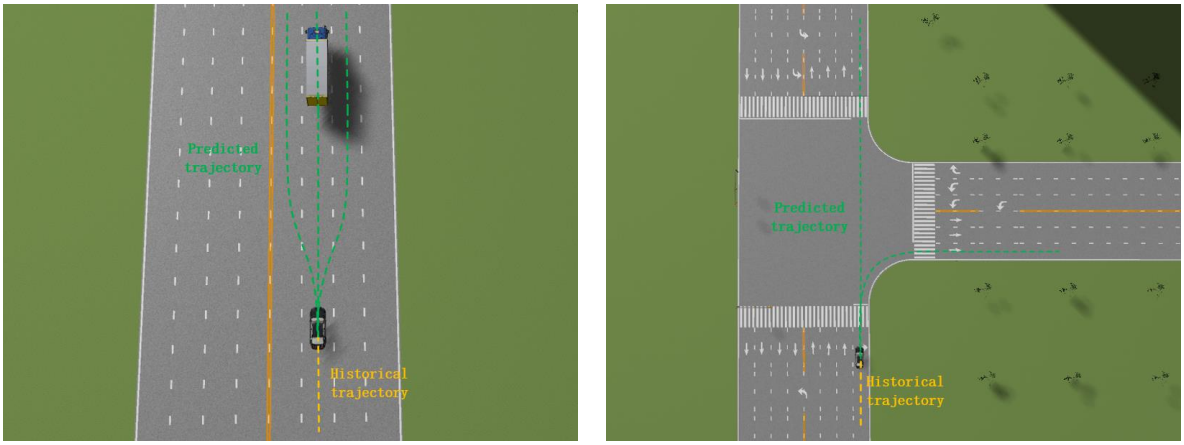
555

556 Figure 15. Simulation scenario layout.

557



(**a**) (**b**)



(c) (d)

Figure 16. Simulation results: (a) On the road scenario; (b) Intersection scenario. (c) Simulation results of the on the road scenario; (d) Simulation results of the intersection scenario.

3.3.2. Simulation Results

The simulation scenario shown in Figure 15 was established according to the real urban traffic scenario. Two typical traffic scenarios were selected to verify the adjustment effect of the priori reference trajectory on the predicted trajectory. Figure 16(a) shows the driving scene on the road, and Figure 16(b) shows the driving scene at the intersection. In the first scene, three reference trajectory intent segments were fitted to lane keeping (LK), lane change left (LCL), and lane change right (LCR), as shown by the blue curve in Figure 16(a). The historical observation vector of the target vehicle denoted the network input, and it was unchanged; the output network vector was converted according to three Prediction Reference Baseline to obtain three predicted trajectories, as shown by the green curve in Figure 16(c). In Figure 16(b), the intersection driving scenario is presented, where two Prediction Reference Baseline intent segments of go straight (GS) and turn right (TR) are fitted respectively; the converted network output results are shown by the green curve in Figure 16(d). Since the network learns the relative relationship between the historical and Prediction Reference Baseline, even at the same network input, the predicted trajectory will be affected by the Prediction Reference Baseline. The experimental results prove that the priori knowledge about the driving scene can be used to adjust the predicted trajectory effectively based on the Prediction Reference Baseline.

3.4. Real-World Urban Traffic Scenarios

3.4.1. Experimental Platform Construction

In order to verify if the simulation results obtained in coincide well with the real-world scenario results, an instrumented vehicle was used to collect data, as shown in Figure 17(a). The vehicle loading sensors included an IBEO four-layer laser scan instrument, a Velodyne HDL-64E lidar, two high-resolution cameras, and a differential GPS/INS (SPAN-CPT) system. The sensor configuration of the vehicle and its sensing range are shown in Figure 17(b). The differential GPS module provided the information on the position, speed, and heading of the ego vehicle. Based on our previous work [37], moving obstacles were detected and tracked by a four-layer laser scanner, which was located at the front of the vehicle. According to the space-time relationship between the moving obstacles, such as pedestrians and vehicles, and experimental vehicles, the position, speed, size, and type of the sports vehicles can be measured. We

conducted a real vehicle experiment in Hefei, Anhui Province, China. The test road is shown in Figure 18(a). The prediction model was exemplarily implemented on NVIDIA Xavier platform using the C++ programming language.

Figure 17. Experimental platform: (a) The Pioneer IV autonomous vehicle; (b) Perception range of the autonomous vehicle.

Before conducting the actual vehicle experiment, the high-resolution maps were collected to establish based on our experimental vehicle. There were more than 1,140 road entities on the map, including the stop signs, lane markings, and lane lines, covering approximately 8 km of the roadways (as Figure 18 shows).

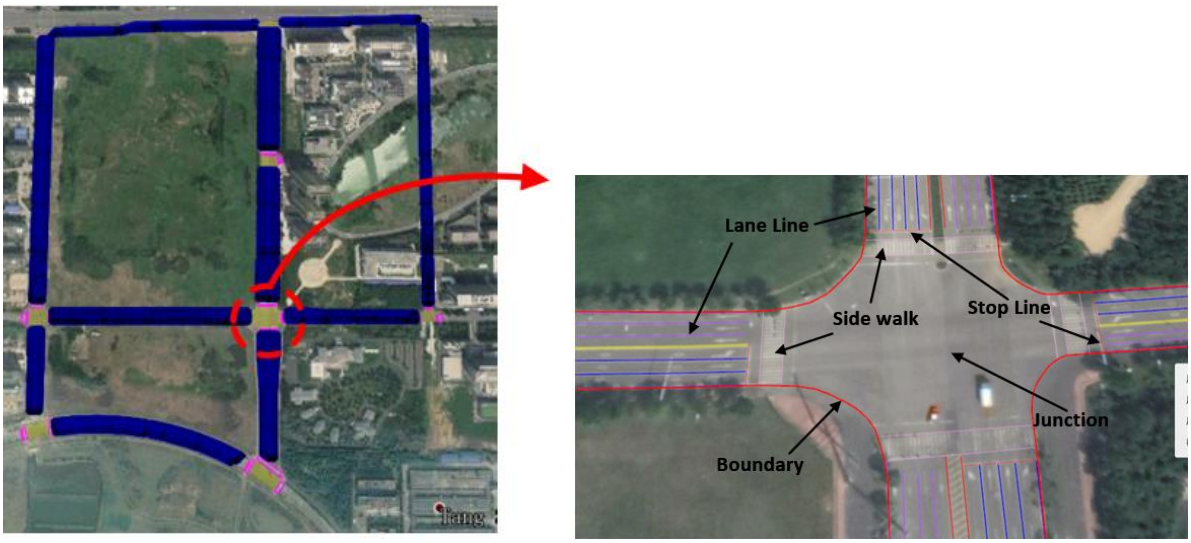


Figure 18. Testing roadway layout: (a) The high-resolution map of the experimental roadway; (b) An enlarged view of the map data.

3.4.2. Field Test and Discussion

The experimental driving route was located on a typical urban roadway, with a total length of about 4.7 km. It includes multiple intersections, Y-shaped intersections, T-shaped intersections, and other common urban road scenarios. Due to the long experimental route and a large number

of scenes encountered, it was inconvenient to conduct the prediction process for each scene. Therefore, two typical scenes were selected for detailed trajectory prediction process analysis.

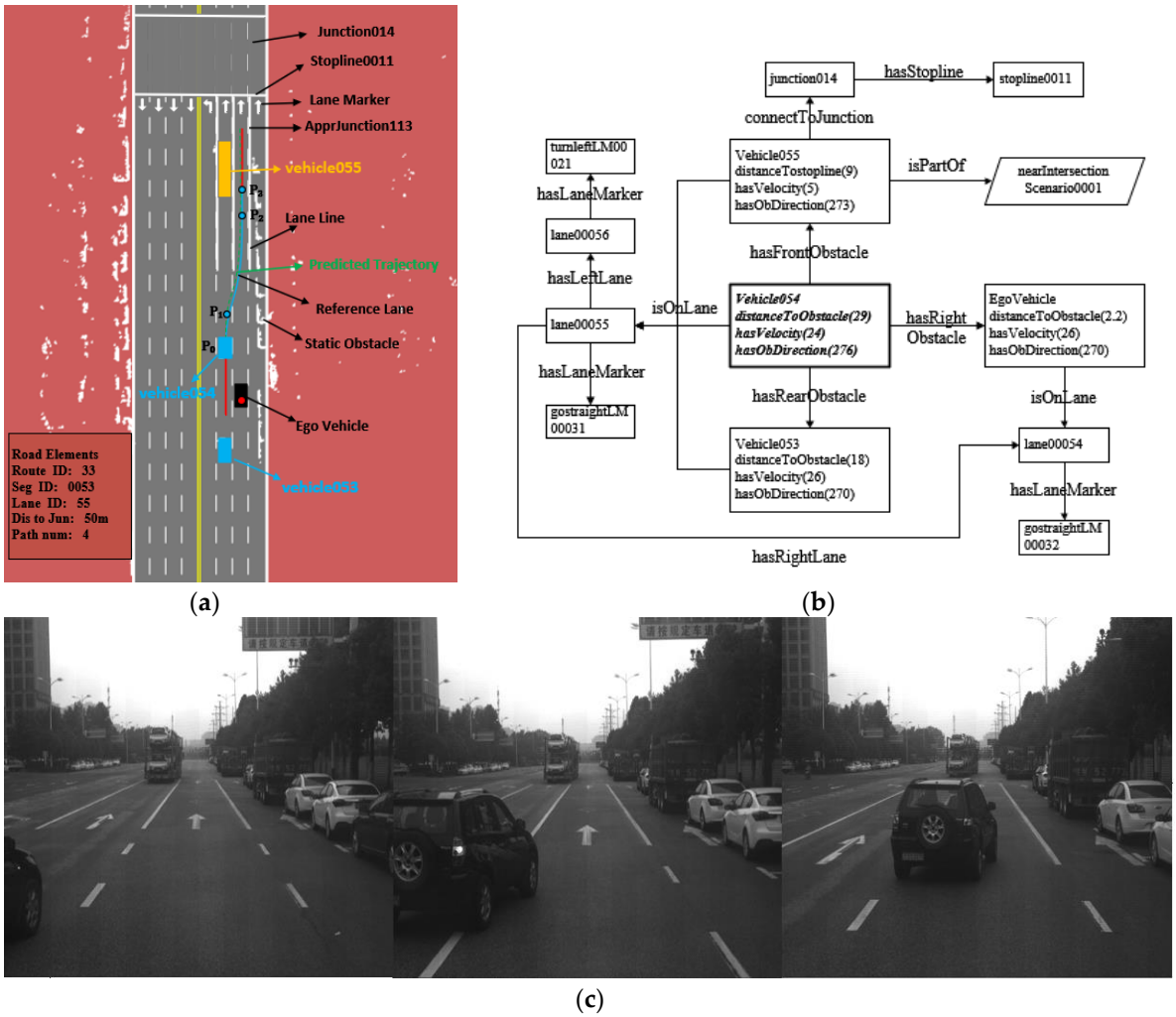


Figure 19. The results of Scenario 1: (a) Trajectory prediction interface; (b) Scenario reasoning; (c) Experimental scenarios.

In Scenario 1, vehicle054 was on the road. The input conditions of the scene evaluation module are shown in Figure 17(b). In front of the target vehicle, there was a large truck denoted as vehicle055 that was moving with a speed of 5 km/h. At that time, the speed of vehicle054 was 24 km/h. The calculated value of the TIV was less than the deceleration threshold. Therefore, it was judged that the target vehicle would have an intension to change lane. Vehiche054 was driving on lane00055, which was a straight lane. Through an associated search in the conceptual ontology model of the driving scene, it was learned that the right lane was also a straight lane, and the lane line was a white dotted line; the prolog rule of legality is expressed as follows:

legalToRight(target,true): targetVehicle(target), isOnLane(target,Lane), hasRightLine(Lane,Line), hasLineType(Line,"dotted_white").

The next section of the target vehicle to travel was the intersection, and the distance to the intersection was greater than 30 m, so the effectiveness of changing lanes was satisfied; thus,

the lane change did not affect the current target of the target vehicle; the prolog rule of reasonableness is as follows:

reasonableToRight(target,true): targetVehicle(target), currentRoadState(target,"ApprJunction"), isOnSegment(target,Seg), connectToJunction(Seg,Junc), intersection(Junc), connectToStopLine(Seg,SL), distToStopLine(SL,DL), DL >= 30.

The autonomous vehicle was driving in the right back region of the target vehicle, and the TTC and TIV values were both less than the corresponding acceleration threshold but greater than the corresponding parking threshold; the prolog rule of safety is as follows:

safeToRight(target,true): targetVehicle(target), hasRightObstacle(target,null), hasRightFrontObstacle(target,null), hasRightBackObstacle(target,egovehicle).

The final prolog rule is as follows:

canChangeToRight(target,true): safeToRight(target,true), reasonableToRight(target,true), legalToRight (target,true).

After obtaining the driving intention of the target vehicle, the prior reference trajectory (the blue curve in Figure 17(a)) was fitted, and the historical trajectory was transformed into Frenet coordinates based on the prior reference trajectory and then fed to the LSTM network input; the predicted trajectory is shown by the green dotted line in Figure 20(a). The trajectory prediction results of vehicle054 could effectively reduce the reaction time of autonomous vehicle while avoiding collisions caused by vehicle054 cutting in.

In Scenario 2, vehicle121 was in the pre-intersection scenario. The input conditions of the scenario evaluation module are shown in Figure 20(b). The current speed of vehicle121 was 24 km/h, and the lane it traveled was lane000103, which was a right-turn lane. In this scenario, the main factor affecting the target vehicle's driving intention was the traffic light. The prolog rule of legality is as follows:

legalToTurnRight(ego,acc): targetVehicle(target), currentRoadState(target,"ApprJunction"), isOnSegment(target,Seg), connectToJunction(Seg,Junc), intersection(Junc), hasTrafficLight(Junc,TR), hasLightColor(TR,"green"), connectToStopLine(Seg,SL), distToStopLine(SL,DL), DL =< 0

The target vehicle was on the right turn, and the prolog rule of reasonableness is as follows:

reasonableToLeft(ego,true): egoVehicle(ego), currentRoadState(ego,"ApprJunction"), isOnSegment(ego,Seg), connectToJunction(Seg,Junc), intersection(Junc), connectToStopLine(Seg,SL), distToStopLine(SL,DL), DL >= 30.

The final rule of safety is as follows:

safeToTurnRight(target,true):- targetVehicle(target),hasFrontObstacle(target,null),hasRightFrontObstacle(target,null),hasRightBackObstacle(target,null).

The final rule is expressed as:

canTurnRight(target,true): safeToTurnRight(target,true), reasonableToTurnRight(target,true), legalToTurnRight (target,true).

The blue curve in Figure 20(a) represents the prior reference trajectory fitted based on the turn right driving intention. The predicted trajectory is shown by the green dotted line in Figure 20(a).

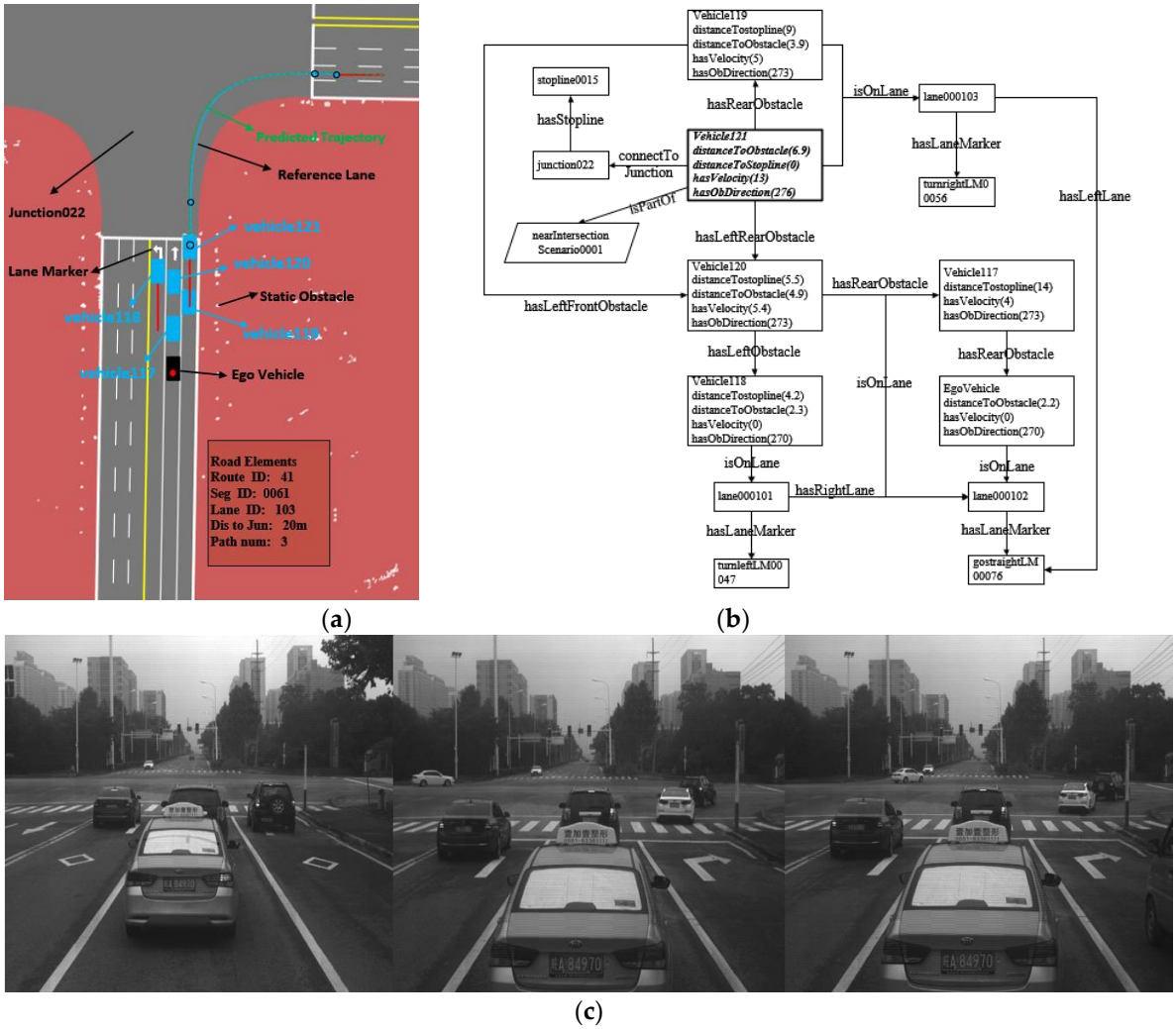


Figure 20. The results for Scenario 2: (a) Trajectory prediction interface; (b) Scenario reasoning; (c) Experimental scenarios.

The experimental results of Scenarios 1 and 2 verify that the proposed LSTM network can effectively combine the prior and posterior knowledge in the driving scene, and the lane change behavior can be predicted before the variation of vehicle kinemics. The proposed LSTM network can be iteratively adapted to a driving scenario without manual annotation during the network training, which significantly reduces the training complexity and solves the sparse data problem. Due to the unique combination of knowledge reasoning, the proposed prediction model can make a more accurate and reasonable estimation of future trajectories of surrounding vehicles than the existing models in different environments.

4. Conclusions

This paper combines a maneuver-based and learning-based trajectory prediction models and proposes an improved trajectory prediction model based on the LSTM neural network driven by driving knowledge. In order to achieve better use of a prior driving knowledge in driving scenarios and solve the problem of the combinatorial explosion caused by a large number of conditional attributes, the multi-source and heterogeneous information of the driving scenario is modeled based on ontology, and a driving knowledge base, including the driving experience and traffic rules, is constructed. Then, the conditional attributes that affect driving intentions are classified and analyzed from the perspectives of safety, legitimacy, and reasonableness, and situation parameters in the horizontal and vertical directions are generated by the deterministic scene evaluation method. Finally, using the obtained situation parameters and the driving knowledge base, the driving intention is inferred based on the prolog online reasoning system. In order to make the prediction results effectively combine the posterior knowledge and solve the problem of insufficient adaptability of the existing learning-based prediction models, this paper converts the driving intention of the target vehicle into a prior reference trajectory, and the Frenet coordinates based on prior reference trajectory are used as a coordinate frame for the LSTM neural network. The prior driving knowledge existing in the driving scene can be used to adjust the predicted trajectory in the form of the prior reference trajectory without increasing the network complexity but ensuring efficient operation of the proposed model on an embedded platform.

The proposed prediction model was verified by simulations and experiments. The simulation results showed that the prior reference trajectory could effectively adjust the output of the LSTM neural network, making the predicted trajectory meet the constraints of the prior knowledge in a driving scenario. The real-world-experiment results show that the proposed prediction model can significantly reduce the computing performance requirements while ensuring real-time performance on the embedded platform. Also due to the full combination of prior and posterior knowledge in the driving scene, the target vehicle's lane-changing behaviour can be predicted on average 2.05 s (for LCL) or 2.71 s (for LCR) in advance, and the precision can be improved by 12.5% for long-term predictions and is more robust, flexible, and adaptive in complex traffic scenarios.

Even though the proposed model has advantages in trajectory prediction, there are still some limitations, such as that indexes of scenario assessment are not comprehensive enough. In order to overcome this limitation, in future work, the evaluation index will be considered from the perspective of human-vehicle interaction and multi-vehicle interaction. Furthermore, data from a more complex scenario will be collected and used to verify the proposed prediction model.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding Statement

This work was supported by National Key Research and Development Program of China (Nos.2016YFD0701401, 2017YFD0700303 and 2018YFD0700602), Youth Innovation Promotion Association of the Chinese Academy of Sciences (Grant No. 2017488), Key Supported Project in the Thirteenth Five-year Plan of Hefei Institutes of Physical Science, Chinese Academy of Sciences (Grant No. KP 2019 16) Equipment Pre-research Program (Grant No. 301060603) Natural Science Foundation of Anhui Province (Grant No. 1508085MF133) and Technological Innovation Project for New Energy and Intelligent Networked Automobile Industry of Anhui Province.

Acknowledgments

The authors gratefully acknowledge the help of the team members, whose contributions were essential for the development of the intelligent vehicle. The authors would also like to thank the Institute of Applied Technology, Hefei Institute of Physical Science, and the Academy of Sciences of China for supporting this study.

References

[1] Stéphanie Lefèvre, D. Vasquez , and C. Laugier . A survey on motion prediction and risk assessment for intelligent vehicles. ROBOMECH Journal 1.1(2014):1-14.

[2] Dai, Shengzhe , L. Li , and Z. Li . Modeling Vehicle Interactions via Modified LSTM Models for Trajectory Prediction. IEEE Access (2019):38287-38296.

[3] Kaempchen N, Weiss K, Schaefer M, Dietmayer KCJ (2004) IMM object tracking for high dynamic driving maneuvers In: Proc. IEEE intelligent vehicles symposium, pp 825–830.

[4] Veeraraghavan H, Papanikolopoulos N, Schrater P (2006) Deterministic sampling-based switching Kalman filtering for vehicle tracking In: Proc.IEEE intelligent transportation systems conference, pp 1340–1345

[5] Broadhurst, Adrian, Simon Baker, and Takeo Kanade. Monte Carlo road safety reasoning. IEEE Proceedings. Intelligent Vehicles Symposium, 2005.. IEEE, 2005.

[6] Bahram, Mohammad, et al. A combined model-and learning-based framework for interaction-aware maneuver prediction. IEEE Transactions on Intelligent Transportation Systems 17.6 (2016): 1538-1550.

[7] Ortiz, Michaël Garcia, et al. Behavior prediction at multiple time-scales in inner-city scenarios. 2011 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2011.

[8] Klingelschmitt, Stefan , et al. Combining behavior and situation information for reliably estimating multiple intentions. Intelligent Vehicles Symposium IEEE, 2014.

[9] Morris, Brendan , A. Doshi , and M. Trivedi . Lane change intent prediction for driver assistance: On-road design and evaluation. 2011 IEEE Intelligent Vehicles Symposium (IV) IEEE, 2011.

[10] Kumar P, Perrollaz M, Lef ‘evre S, Laugier C (2013) Learning-based approach for online lane change intention prediction In: Proc. IEEE Intelligent Vehicles Symposium, pp 797–802

[11] Zyner, Alex , et al. Long short term memory for driver intent prediction. 2017 IEEE Intelligent Vehicles Symposium (IV) IEEE, 2017.

[12] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, Social LSTM: Human Trajectory Prediction in Crowded Spaces, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, jun 2016, pp. 961–971.

[13] D. J. Phillips, T. A. Wheeler, and M. J. Kochenderfer, Generalizable Intention Prediction of Human Drivers at Intersections, 2017 IEEE Intelligent Vehicles Symposium (IV), pp. 1665–1670, 2017.

[14] F. Altché, and A. De La Fortelle, An LSTM network for highway trajectory prediction, IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 353-359, October 2017.

[15] C. Ding, W. Wang, X. Wang, and M. Baumann, A neural network model for driver’s lane-changing trajectory prediction in urban traffic flow, Mathematical Problems in Engineering, vol. 2013, 2013.

[16] J. Zheng, K. Suzuki, and M. Fujita, Predicting driver’s lane changing decisions using a neural network model, Simulation Modelling Practice and Theory, vol. 42, pp. 73–83, 2014.

[17] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, Learning-based approach for online lane change intention prediction, in 2013 IEEE Intelligent Vehicles Symposium (IV). IEEE, jun 2013, pp. 797–802.

[18] S. H. Park, B. Kim, C. M. Kang, C. C. Chung and J. W. Choi, Sequence-to-Sequence Prediction of Vehicle Trajectory via LSTM Encoder-Decoder Architecture, 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, 2018, pp. 1672-1678.

[19] Xin, Long , et al. Intention-aware Long Horizon Trajectory Prediction of Surrounding Vehicles using Dual LSTM Networks. 2018 IEEE International Conference on Intelligent Transportation Systems (ITSC) IEEE, 2018.

[20] N. Deo and M. M. Trivedi, Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs, 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, 2018, pp. 1179-1184

[21] Ding, Wenchao , and S. Shen . Online Vehicle Trajectory Prediction using Policy Anticipation Network and Optimization-based Context Reasoning. 2019 International Conference on Robotics and Automation (ICRA) IEEE, 2019.

[22] N. Deo and M. M. Trivedi, Convolutional social pooling for vehicle trajectory prediction, 2018, arXiv:1805.06771. [Online]. Available: https://arxiv.org/abs/1805.06771.

[23] Dai S , Li L , Li Z . Modeling Vehicle Interactions via Modified LSTM Models for Trajectory Prediction[J]. IEEE Access, 2019:38287-38296.

[24] Khakzar M , Rakotonirainy A , Bond A , et al. A Dual Learning Model for Vehicle Trajectory Prediction[J]. IEEE Access, 2020, 8(99):21897-21908.

[25] Hummel, Britta, Werner Thiemann, and Irina Lulcheva. Scene understanding of urban road intersections with description logic. Dagstuhl Seminar Proceedings. Schloss Dagstuhl-Leibniz-Zentrum fr Informatik, 2008.

[26] Armand, Alexandre, David Filliat, and Javier Ibañez-Guzman. Ontology-based context awareness for driving assistance systems. 2014 IEEE intelligent vehicles symposium proceedings. IEEE, 2014.

[27] Ralf Kohlhaas, Thomas Bittner, Thomas Schamm, and J Marius Zollner. Semantic state space for high-level maneuver planning in structured traffic scenes. Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on. IEEE, 2014: 1060–1065.

[28] Xinli, Geng , et al. A Scenario-Adaptive Driving Behavior Prediction Approach to Urban Autonomous Driving. Applied ences 7.4(2017):426.

[29] Glaser S, Vanholme B, et al. Maneuver-Based Trajectory Planning for Highly Autonomous Vehicles on Real Road With Traffic and Driver Interaction. IEEE transactions on Intelligent Transportation Systems 2010,11(3): 589-606.

[30] N. Samyeul and A. Kyounghwan, Decision-Making Framework for Automated Driving in Highway Environments, IEEE Transactions on Intelligent Transportation Systems, 2018,19(1): 58-71.

[31] Noh, Samyeul, Kyounghwan An, and Wooyong Han. Situation assessment and behavior decision for vehicle/driver cooperative driving in highway environments. 2015 IEEE International Conference on Automation Science and Engineering (CASE). IEEE, 2015.

[32] Hochreiter, Sepp, and Jürgen Schmidhuber. Long short-term memory. Neural computation 9.8 (1997): 1735-1780.

[33] US Department of Transportation, NGSIM: Next Generation Simulation, 2008, Available: http://www.ngsim.fhwa.dot.gov. [Accessed: June 6, 2017].

[34] A. Bender, J. R. Ward, S. Worrall, and E. M. Nebot, Predicting driver intent from models of naturalistic driving, in 2015 IEEE 18th International Conference on Intelligent Transportation Systems. IEEE, 2015, pp. 1609–1615.

[35] Yoon, Seungje, and Dongsuk Kum. The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles. 2016 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2016.

[36] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet and F. Nashashibi, Non-local Social Pooling for Vehicle Trajectory Prediction, 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 2019, pp. 975-980.

[37] Huang, Rulin, et al. A practical point cloud based road curb detection method for autonomous vehicle. Information 8.3 (2017): 93.