

ORIGINAL ARTICLE

Open Access



Target Vehicle Selection Algorithm for Adaptive Cruise Control Based on Lane-changing Intention of Preceding Vehicle

Jun Yao, Guoying Chen and Zhenhai Gao*

Abstract

To improve the ride comfort and safety of a traditional adaptive cruise control (ACC) system when the preceding vehicle changes lanes, it proposes a target vehicle selection algorithm based on the prediction of the lane-changing intention for the preceding vehicle. First, the Next Generation Simulation dataset is used to train a lane-changing intention prediction algorithm based on a sliding window support vector machine, and the lane-changing intention of the preceding vehicle in the current lane is identified by lateral position offset. Second, according to the lane-changing intention and collision threat of the preceding vehicle, the target vehicle selection algorithm is studied under three different conditions: safe lane-changing, dangerous lane-changing, and lane-changing cancellation. Finally, the effectiveness of the proposed algorithm is verified in a co-simulation platform. The simulation results show that the target vehicle selection algorithm can ensure the smooth transfer of the target vehicle and effectively reduce the longitudinal acceleration fluctuation of the subject vehicle when the preceding vehicle changes lanes safely or cancels their lane change maneuver. In the case of a dangerous lane change, the target vehicle selection algorithm proposed in this paper can respond more rapidly to a dangerous lane change than the target vehicle selection method of the traditional ACC system; thus, it can effectively avoid collisions and improve the safety of the subject vehicle.

Keywords: Lane-changing intention, Target vehicle selection, Support vector machine, Adaptive cruise control

1 Introduction

The problems of traffic congestion have become more and more serious. As a result, adaptive cruise control (ACC), as a key technology of advanced driver assistance systems (ADASs), has been widely studied and gradually introduced into the lives of ordinary people. According to statistical reports, lane changes are the main cause of car crashes [1–4]. When the preceding vehicle changes lanes, traditional ACC systems simply declare the target vehicle (i.e., the vehicle that the subject vehicle follows) as the

closest one currently in the subject vehicle's lane; thus, these systems cannot comprehensively consider lane-changing vehicles. Under these condition, large fluctuations in longitudinal acceleration can occur; these greatly reduce ride comfort and may even present collision risks [5, 6]. To prevent this, one key technology is that of reliable lane-changing intention prediction, which can recognize that the preceding vehicle intends to change lanes before it crosses the lane line. This allows the subject vehicle to respond in advance of the preceding vehicle's lane-changing action, thereby reducing acceleration fluctuations and minimizing collision risks. The most relevant methods thus far reported for predicting the preceding vehicle's lane-changing intention can be

*Correspondence: gaozh@jlu.edu.cn
State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130022, China

roughly classified into four categories: fuzzy logic-based, support vector machine (SVM)-based, hidden Markov model (HMM)-based, and deep learning-based.

The fuzzy logic-based method uses relative motion information between the subject and preceding vehicles as the input variable; with this, the lane-changing intention of the preceding vehicle can be obtained, to effectively realize human control strategies and experience. Moon et al. [7, 8] introduced a lane-changing intention predictor based on fuzzy logic; this used the relative lateral distance and relative lateral speed between the preceding and subject vehicles as the input, and it used fuzzy rules to determine the lane-changing probability of the preceding vehicle. This method assumed that the vehicles with smaller lateral relative distances and larger lateral relative speeds were more likely to change lane. The fuzzy rules presented in the literature are primarily based on the fitting curve of relative speed and relative distance under the preceding vehicle's cut-in condition. However, the fuzzy logic controller largely depends on human experience, and it cannot objectively identify lane-changing intentions.

The SVM-based method selects the appropriate feature vector using relative motion information, and it obtains the optimal SVM parameters through training, to predict the lane-changing intention of the preceding vehicle. Ma et al. [9, 10] used data collected from actual traffic environments as training samples, to identify cut-in maneuvers for adjacent-lane vehicles based on fuzzy support vector machines (FSVMs). To improve the training accuracy of the cut-in identifier, a fuzzy membership coefficient was introduced for each sample to solve the FSVM, and a grid optimization was conducted on the FSVM parameters. Woo et al. [11] defined the feature vector as comprising the distance from the centerline, the lateral velocity, and the potential feature. The potential feature characterizes the likelihood of lane-changing by analyzing the location relationship between the preceding vehicle and its surrounding vehicles. By adding the potential feature, the proposed SVM algorithm can eliminate the false predictions produced by zigzag driving.

The HMM-based method mostly uses the observed state information of the preceding vehicle to identify independent and invisible lane-changing intentions. Ma established a mixed Gaussian-HMM to describe the lane changing behavior of adjacent vehicles. The driver's decision states were segmented and described by the model parameters [12]. Furthermore, the lateral distance between the preceding vehicle and the center of the host vehicle was used to characterize the changes in decision states. Using results from Ref. [12], Zhang [13] classified lane-changing maneuvers into the safe and dangerous lane-changing processes, according to collision risk.

Based upon the characteristics of lane keeping and lane changing, as well as the characteristics of safe and dangerous lane changes, the HMM-based lane-changing identification method was designed to use a sliding time window, and the driving state of each time window was judged in turn. Mitrovic proposed a simple and reliable method for identifying driving events using a HMM [14]. By collecting real-vehicle experimental data and manually selecting observation sequences for training and verification, each observation sequence was classified into specific types of events, and the HMM model parameters of each driving event were trained separately. The observation sequence from the training set was evaluated using multiple models. By comparing the probability of the observation sequence calculated by each HMM model, the event corresponding to the highest HMM model was selected as the estimated result.

The deep learning-based method predicts the preceding vehicle's lane-changing intention or driving trajectory using a neural network. This method requires a huge dataset for parameter training to improve prediction results. Zhang et al. [15] used the speech-recognition framework as an example, and they mapped the behavior of the preceding vehicle (i.e., lane-changing or lane-keeping) to different speech words. Because the motion information of the preceding and surrounding vehicles was both continuous and time-varying, words of different sizes corresponded to different driving styles during lane changes. The speech recognition model could be effectively applied to recognize the preceding vehicle's lane-changing behavior. Yoon et al. [16] calculated the lane-change likelihood of multiple target lanes and trajectories of surrounding vehicles using a radial basis function network (RBFN). The RBFN prediction algorithm used the classification distribution and future trajectory in parallel to estimate the probability of each lane becoming the driver's target lane, and it converted the RBFN into a probability model which incorporated uncertainty. Lee et al. [17] proposed a lane-changing intention recognizer based on a convolutional neural network (CNN). This method transformed real-world driving data into a simplified bird's-eye view, which facilitated a CNN-based inference approach with low computation cost and robustness against noisy inputs.

Most of the current literature has sought to predict the lane-changing intention of the preceding vehicle in the adjacent lane (as shown in Figure 1); however, the prediction results for the lane-changing intention of the preceding vehicle in the current lane (as shown in Figure 2) also determine the longitudinal acceleration of the subject vehicle. For example, when the preceding vehicle in the current lane changes lanes and a low-speed commercial vehicle or stationary object appears ahead in the current

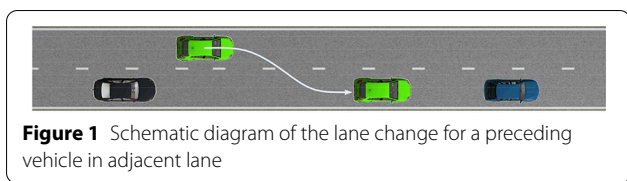


Figure 1 Schematic diagram of the lane change for a preceding vehicle in adjacent lane

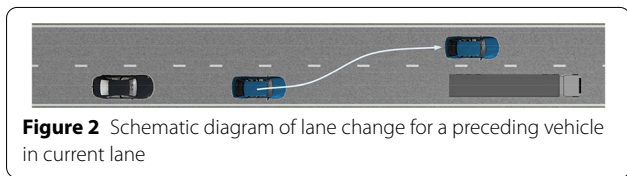


Figure 2 Schematic diagram of lane change for a preceding vehicle in current lane

lane, the subject vehicle will also experience acceleration fluctuations or even collision risks. Therefore, this paper studies the lane-changing intention prediction algorithm for both the preceding vehicle in the adjacent lane and the preceding vehicle in the current lane. Because most of the previous studies used SVMs to identify the lane-changing intention of the preceding vehicle, they only selected a certain feature vector and kernel function of the SVM, and they failed to explain the reasons for their selections. This study compares the prediction accuracies of different types of SVM, selects the RBF as the kernel function, and analyzes the influence of different sliding window sizes on the prediction accuracy. Moreover, most previous research has only studied the successful lane changes of the preceding vehicle, without considering the failure or cancellation thereof. This work studies target vehicle selection when the preceding vehicle fails to change lanes.

The remainder of this paper is structured as follows: Section 2 illustrates the system architecture, Section 3 introduces the lane-changing intention prediction algorithm, Section 4 introduces the target vehicle selection algorithm, Section 5 studies the longitudinal motion control algorithm, Section 6 evaluates the proposed algorithm in a simulation, and Section 7 concludes the paper.

2 System Architecture

The overall framework proposed in this paper is shown in Figure 3. It is primarily divided into three components: lane-changing intention prediction, target vehicle selection, and longitudinal motion control. First, the lane-changing intention of the preceding vehicle was primarily predicted by the sliding window SVM algorithm. We used the Next Generation Simulation (NGSIM) dataset to train the parameters of the SVM and determine the size of the sliding window. The lane-changing intention of the preceding

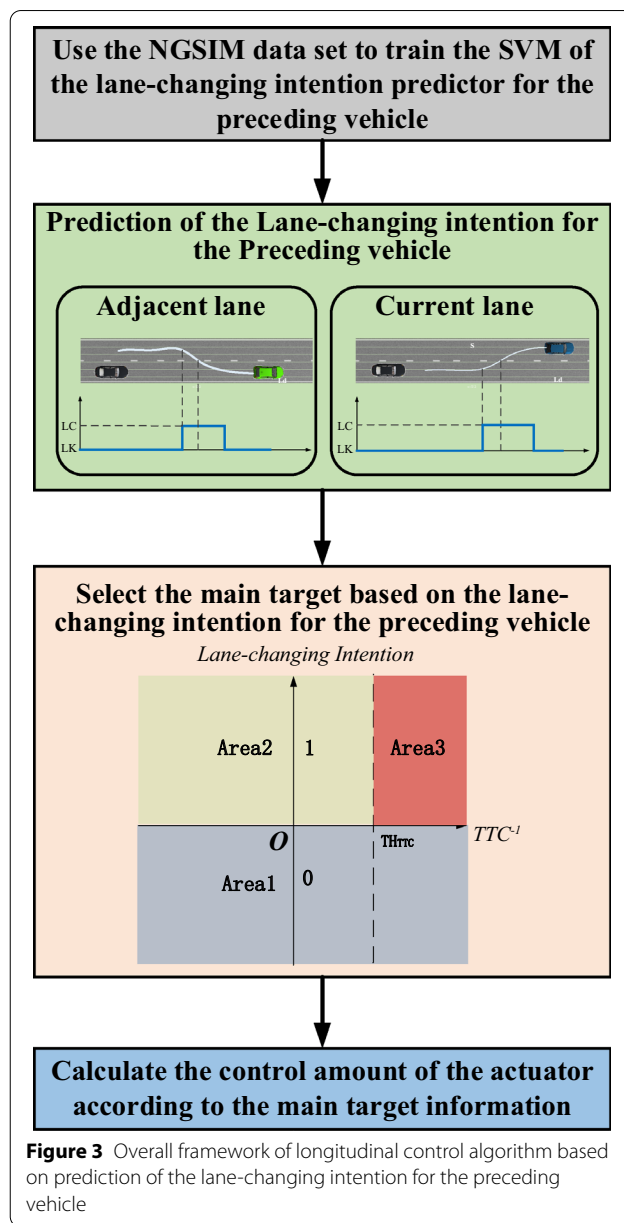


Figure 3 Overall framework of longitudinal control algorithm based on prediction of the lane-changing intention for the preceding vehicle

vehicle in the current lane was predicted via the lateral relative distance offset. The next step was to select the target vehicle. The target vehicle selection determines the target vehicle under three different conditions: safe lane-changing, dangerous lane-changing, and lane-changing cancellation. The longitudinal motion control generated the actuator control value using the state information of the target vehicle. The actuator control quantity was composed of two components: the feedforward and feedback control quantities.

3 Lane-changing Intention Prediction Algorithm based on Sliding Window SVM

When the preceding vehicle changes lanes, the traditional ACC system cannot comprehensively consider the preceding vehicle in the current lane as well as the lane-changing vehicle. Large fluctuations arise in longitudinal acceleration under this condition, which greatly reduces the ride comfort and may even cause collision risks. To avoid the violent fluctuation in longitudinal acceleration caused by the jump of the target vehicle, the sliding window SVM algorithm was adopted to identify the lane-changing intention of the preceding vehicle.

3.1 NGSIM Dataset Preprocessing

This study used the public dataset recorded by the NGSIM program (initiated by the Federal Highway Administration in 2002) to train the sliding window SVM [18]. This program used high-definition cameras installed above the road to record vehicle driving data, and it used video processing software to obtain the vehicle trajectory data at intervals of 0.1 s. The lane-changing vehicle data on the US101 highway in the NGSIM dataset were used to train the lane-changing intention prediction SVM in this work.

The study area schematic and camera coverage of the NGSIM US101 highway data are shown in Figure 4. After simple filtering, 6100 individual vehicle driving data points were obtained. We studied the free lane-changing behavior of passenger cars; thus, reasonable lane-changing vehicle data must meet the following constraints:

- (1) Because this work studies the free lane changing of cars, it is necessary to restrict the types of vehicles to 2-cars.
- (2) Lanes 7 and 8 of US101 highway are both ramps, Lane 6 is the auxiliary lane of the ramp entrance, Lane 1 is the leftmost lane, and Lane 5 is the rightmost lane, adjacent to Lane 6. To avoid the influence of the forced lane-changing behavior data produced by vehicles entering and exiting the ramp, the lane-changing vehicle data used in this paper exclude the vehicle trajectories containing Lanes 6, 7, and 8 in their driving lane ID, and we ensure that the lane ID in the vehicle trajectory data undergoes a change.
- (3) To prevent vehicle lane ID changes caused by vehicles driving near the lane line at all times, we compared the deviations of lateral position between the start and end of the lane change, ensuring that this deviation exceeded 2.75 m.

Through artificial selection, 184 reasonable lane-changing vehicle trajectories were obtained. Because the subject vehicle can only obtain the relative position and speed information of the preceding vehicle through its sensors, it was necessary to calculate the relative lateral distance and lateral speed of the preceding vehicle relative to the road centerline of the target lane. By subtracting the local coordinates of the target lane centerline from those of the lane-changing vehicle, the relative lateral distance of the lane-changing vehicle relative to the centerline of the target lane was obtained. To reduce the influence of NGSIM dataset measurement errors, Kalman filter was used to calculate the relative lateral velocity v_y and relative lateral acceleration a_y of the preceding vehicle relative to the road centerline of the target lane. The estimated relative lateral velocity v_y and acceleration a_y are shown in Figures 5 and 6, respectively.

The relative lateral velocity calculated by Kalman filter was essentially identical to that obtained by the local coordinate Y 's difference in the original NGSIM dataset; however, the spike was effectively suppressed. By comparing the relative lateral accelerations computed via Kalman filter of the acceleration data obtained by velocity difference, we found that such filtering could well restrain the fluctuation generated by the difference.

3.2 SVM Algorithm Design

3.2.1 SVM Algorithm

SVM is a very popular algorithm in machine learning. It is mainly used to identify a suitable hyperplane in a multi-dimensional space as a classification plane, to maximize the minimum spacing of positive and negative samples in the sample space. The samples that satisfy the

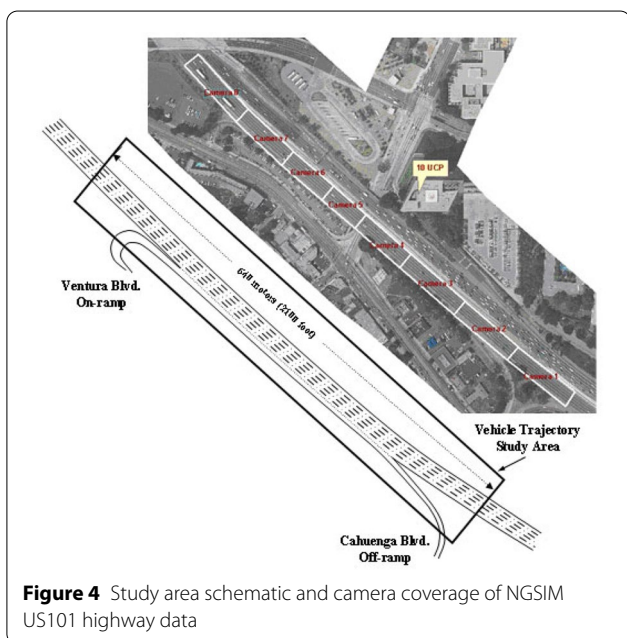


Figure 4 Study area schematic and camera coverage of NGSIM US101 highway data

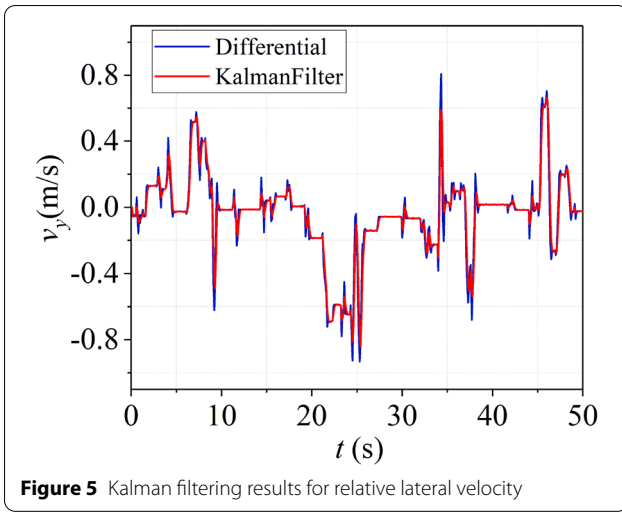


Figure 5 Kalman filtering results for relative lateral velocity

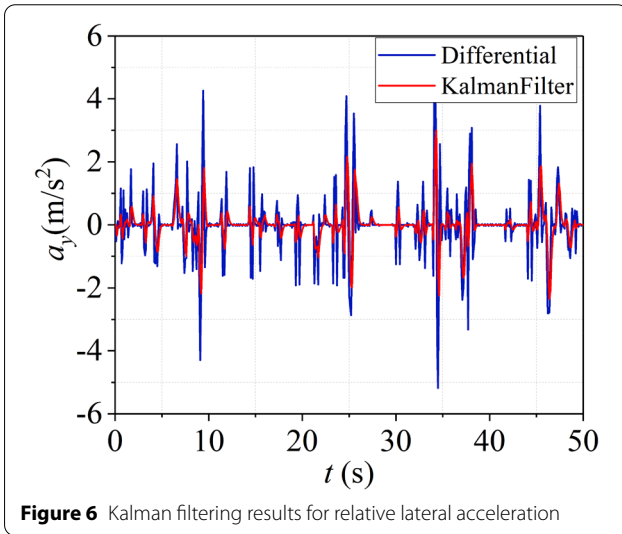


Figure 6 Kalman filtering results for relative lateral acceleration

minimum spacing are called support vectors. For linearly inseparable cases, the SVM can use the kernel function to transform the nonlinear classification scenario into a linearly separable situation in the high-dimensional sample space [19–21]. Commonly used kernel functions include the polynomial and Gaussian kernel functions.

We assume that the classification function is

$$h_{\bar{w},b}(\bar{x}) = g(\bar{w}^T \bar{x} + b), \tag{1}$$

where $g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$, \bar{w} and b are the training parameters, and \bar{x} is the feature vector.

In the model that predicts the lane-changing intention of the preceding vehicle, $h_{\bar{w},b}(\bar{x}) = 1$ indicates that the preceding vehicle intends to change lanes, and $h_{\bar{w},b}(\bar{x}) = 0$ indicates that the preceding vehicle does not

intend to change lanes and will continue to drive in the original one. The optimization objective of the SVM is to maximize the geometric margins between the positive and negative samples. The definition of geometric margin $\gamma^{(i)}$ is

$$\begin{cases} \gamma^{(i)} = y^{(i)} \left(\left(\frac{\bar{w}}{\|\bar{w}\|} \right)^T \bar{x}^{(i)} + \frac{b}{\|\bar{w}\|} \right), \\ \gamma = \min_{i=1,2,\dots,m} \gamma^{(i)}, \end{cases} \tag{2}$$

where m represents the number of samples in the training set, and γ denotes the smallest margin. The original optimization problem of the SVM is as follows:

$$\begin{aligned} \max_{\gamma, \bar{w}, b} & \quad \gamma \\ \text{s.t.} & \quad y^{(i)} (\bar{w}^T \bar{x}^{(i)} + b) \geq \gamma, \quad i = 1, \dots, m \\ & \quad \|\bar{w}\| = 1. \end{aligned} \tag{3}$$

The non-convex constraint $\|\bar{w}\| = 1$ in the original optimization problem means that the original problem is very difficult to solve. Thus, it must be transformed into a convex optimization problem:

$$\begin{aligned} \min_{\gamma, w, b} & \quad \frac{1}{2} \|\bar{w}\|^2, \\ \text{s.t.} & \quad y^{(i)} (\bar{w}^T \bar{x}^{(i)} + b) \geq 1, \quad i = 1, \dots, m. \end{aligned} \tag{4}$$

Through Lagrange duality, the above convex optimization problem can be transformed into a quadratic programming problem, expressed as

$$\begin{aligned} \max_{\lambda} W(\lambda) &= \sum_{i=1}^m \lambda_i - \frac{1}{2} \sum_{i,j=1}^m y^{(i)} y^{(j)} \lambda_i \lambda_j \langle \bar{x}^{(i)}, \bar{x}^{(j)} \rangle \\ \text{s.t.} & \quad \lambda_i \geq 0, \quad i = 1, \dots, m \\ & \quad \sum_{i=1}^m \lambda_i y^{(i)} = 0, \end{aligned} \tag{5}$$

where $\langle \bar{x}^{(i)}, \bar{x}^{(j)} \rangle$ represents the kernel function value of $\bar{x}^{(i)}$ and $\bar{x}^{(j)}$, and λ represents the Lagrange multiplier.

3.2.2 SVM Feature Vector Selection

The feature vectors selected in Ref. [10] include the longitudinal relative distance between the subject and preceding vehicles, lateral relative distance, longitudinal relative speed, lateral relative speed, longitudinal relative acceleration, lateral relative acceleration, and subject vehicle speed; these are shown in Figure 7. However, the training samples are limited and cannot cover all feature vectors that may arise in the SVM; for example, the present speed of the subject vehicle never appears in the training sample; furthermore, the current longitudinal relative distance, longitudinal relative

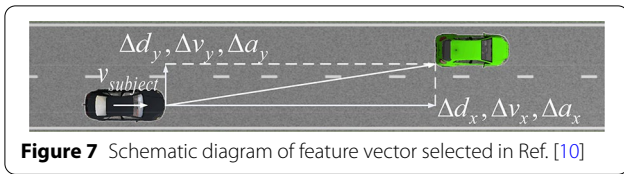


Figure 7 Schematic diagram of feature vector selected in Ref. [10]

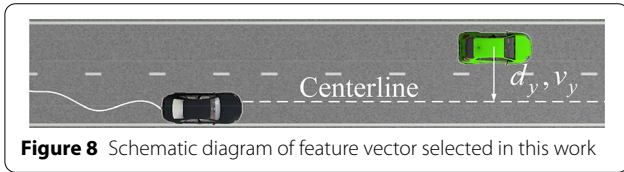


Figure 8 Schematic diagram of feature vector selected in this work

speed, and longitudinal relative acceleration exceed the range of the feature vector in the training sample. In the above cases, the accuracy of the lane-changing intention prediction obtained via the SVM is very low. Ref. [11] selected the lateral relative distance, lateral relative speed, and potential feature of the preceding vehicle (relative to the centerline of the subject vehicle's driving lane) as the feature vector. The potential feature, which analyzes the position relationship between the preceding vehicle and its surrounding traffic vehicles, represents the lane-changing risk degree of the preceding vehicle. This feature is added to reduce false predictions when the preceding vehicle is performing zigzag driving in its original lane. However, millimeter-wave radar and cameras, as the main sensor systems of ADAS, do not obtain comprehensive and accurate motion state information for traffic vehicles surrounding the preceding vehicle. In addition, this paper assumes that the preceding vehicle's zigzag driving in the original lane does not necessarily indicate it as failing to change lanes. It may indicate the inexperienced driving of novice drivers, or that the target vehicle is in the target lane adjustment stage after a lane change. The potential feature cannot be used to solve all zigzag driving misjudgments.

The feature vectors selected in this work include the lateral relative distance d_y and lateral relative speed v_y of the preceding vehicle relative to the centerline of the subject vehicle's driving lane as shown in Figure 8. When using only the relative motion information at the present moment as the feature vector, a short-term misjudgment often occurs owing to the jump of the motion state. However, the lane-changing intention prediction of the preceding vehicle at the current moment is often related to the relative motion information over several previous cycles. Therefore, this paper takes the relative motion information of the preceding vehicle relative to the centerline of the subject vehicle's driving lane in the

previous k cycles as the feature vector. The feature vector \bar{x}_t at time t can be expressed as

$$\bar{x}_t = [\bar{D}_y, \bar{V}_y], \tag{6}$$

$$\bar{D}_y = [d_{y,t-(k-1)}, d_{y,t-(k-2)}, \dots, d_{y,t}], \tag{7}$$

$$\bar{V}_y = [v_{y,t-(k-1)}, v_{y,t-(k-2)}, \dots, v_{y,t}], \tag{8}$$

where \bar{D}_y is the feature of the lateral relative distance, and \bar{V}_y is the feature of the lateral relative speed with respect to the centerline of the subject vehicle's driving lane.

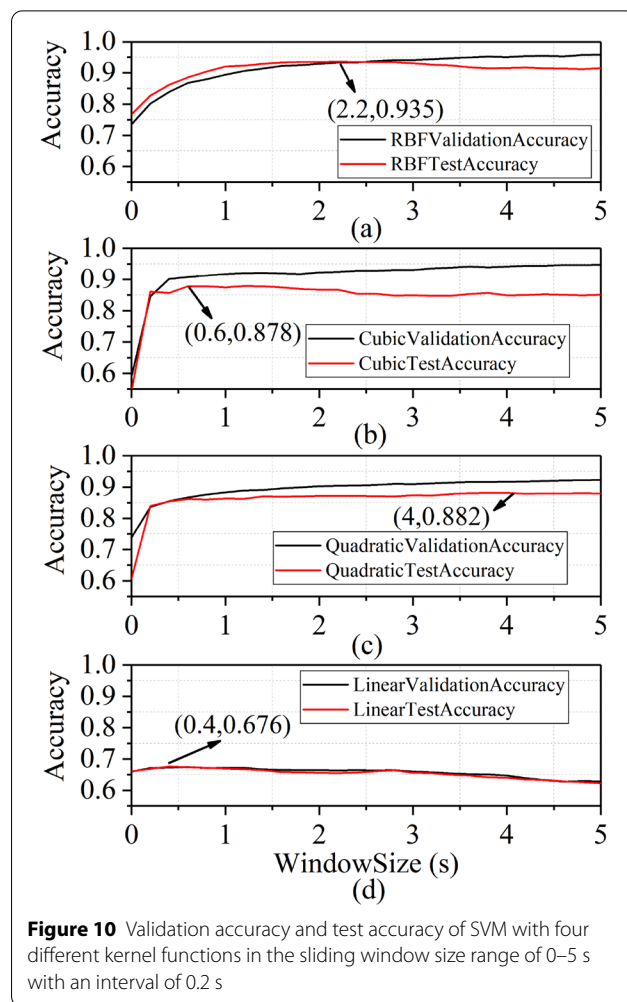
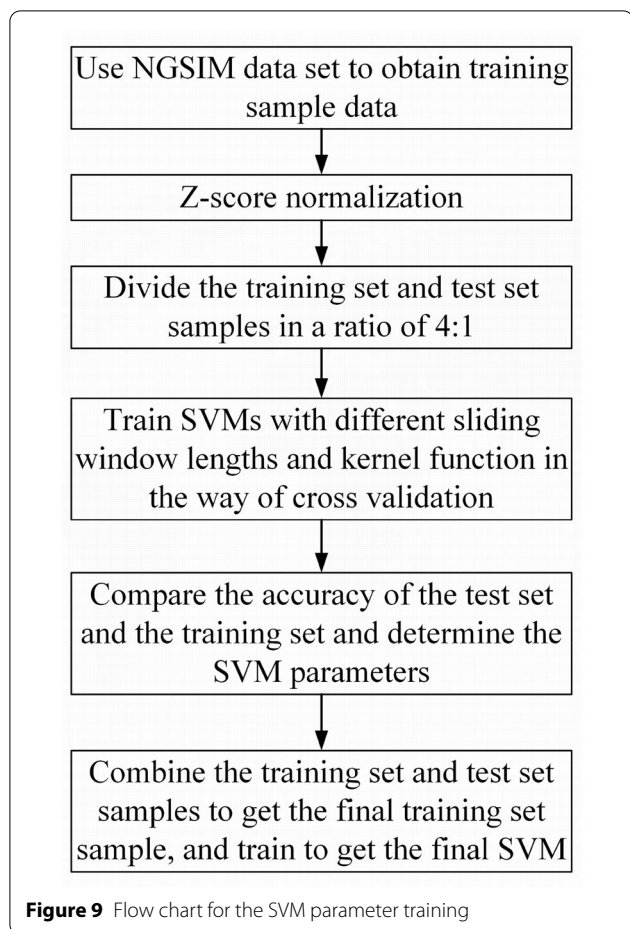
Selecting the relative motion information of the preceding vehicle relative to the centerline of the subject vehicle's driving lane as the feature vector (instead of the relative motion information of the preceding vehicle relative to the subject vehicle) can, on the one hand, mitigate the influence of the subject vehicle's lateral movement on the lane-changing intention prediction. On the other hand, it is very convenient to convert the relative lateral distance into d coordinates under *Frenet* coordinates when driving in curves [22, 23].

3.2.3 SVM Parameter Training

To resolve the influences of different feature units, the z-score normalization was used to standardize the features. The mean value of each feature after processing was zero, and the standard deviation was 1. Prior to SVM parameter training, the NGSIM dataset was divided into training and test set samples in the ratio 7:3. The numbers of training and test set samples were 10080 and 4273, respectively. SVMs with different parameters were trained using training set samples, and the SVM prediction accuracy was tested by test set samples. Meanwhile, we used the cross-validation method to divide the training set data into N copies ($N = 5$ in this paper). In each training process, $N - 1$ of these were selected for training, and the remaining copy was used as the validation set. Through n -training, the group of parameters with the highest accuracy from the validation set was selected as the final training result. The flow chart of the SVM parameter training is shown in Figure 9.

Linear, quadratic, cubic, and radial basis functions were selected as the kernel function to train the SVM. Meanwhile, to determine the size of the sliding window, we trained the SVM with four different kernel functions in a window size range of 0–5 s with an interval of 0.2 s. The training results are shown in Figure 10.

In Figure 10(d), we can see that when the sliding window size was 0.4 s, the test set accuracy of the linear kernel function SVM reached the maximum value of



0.676. Because the quadratic, cubic, and RBF kernel functions can map sample features to higher spaces and achieve nonlinear classification, the validation and test set accuracies of the above three kernel function SVMs were improved to different degrees compared with the linear kernel function SVM. The RBF kernel function SVM had the most prominent improvement. The test set accuracy of the RBF kernel function SVM reached 0.935 when the sliding window was 2.2 s. Therefore, we selected the RBF kernel function SVM to predict the lane-changing intention of the preceding vehicle.

Comparing the accuracies of the test and verification sets, we found that the test set accuracies of the above three kernel function SVMs were lower than the validation set accuracy to some extent. When the size of the sliding window was increased, the number of features increased, and overfitting occurred during SVM training. When the size of the sliding window increased, the validation set accuracy could be continuously improved. However, when the sliding window

size exceeded a certain range, the test-set prediction accuracy decreased when the size of the time window increased (this was particularly clear for the RBF kernel function SVM); that is, in terms of the size of the sliding window, longer does not necessarily entail better.

As shown in Figure 10(a), when the sliding window size was 2.2 s, the test set accuracy of the RBF kernel function SVM was maximal. Therefore, we selected the RBF kernel function SVM with a sliding window size of 2.2 s to predict the lane-changing intention of the preceding vehicle. After determining the SVM kernel function and sliding window size, we combined the test and training set samples to form a new training set, and trained using this set to obtain the final lane-changing intention prediction SVM. The parameters of the final SVM for the preceding vehicle lane-changing intention prediction are shown in Table 1. Here, KernelScale is the parameter γ of the RBF, where the RBF has the

Table 1 Parameters of the final SVM for the lane-changing intention prediction of the preceding vehicle

Parameters	Value
SVM kernel function	RBF
Sliding window size	2.2 s
KernelScale	8.5
BoxConstraint	20.5

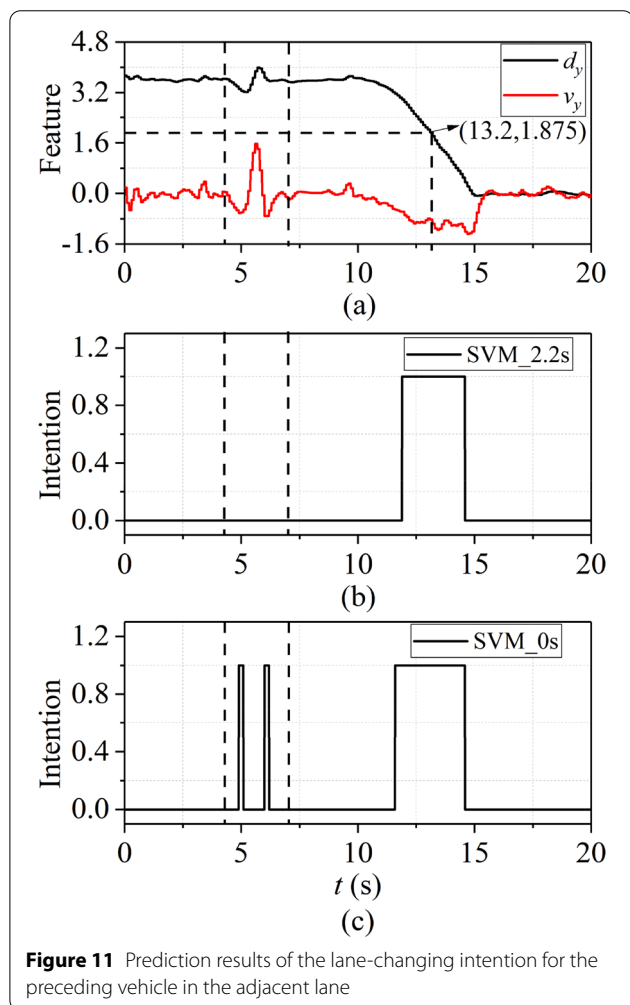


Figure 11 Prediction results of the lane-changing intention for the preceding vehicle in the adjacent lane

following form: $K(\bar{x}^{(i)}, \bar{x}^{(j)}) = \exp(-\gamma \|\bar{x}^{(i)} - \bar{x}^{(j)}\|)$. BoxConstraint is a positive value that controls the penalty imposed on observations with large residuals [24].

3.3 Prediction Results of Lane-changing Intention for Preceding Vehicle in the Adjacent Lane

The prediction results of the lane-changing intention for the preceding vehicle in the adjacent lane are shown in Figure 11: (a) SVM features, (b) lane-changing intention

prediction obtained by SVM_2.2 s, and (c) lane-changing intention prediction obtained by SVM_0 s. It can be seen from Figure 11(a) that the preceding vehicle performed zigzag driving in the original lane within 4.3–7 s of the start of the simulation. Furthermore, the lane change began at 10.5 s and ended at 15 s. The overall lane changing time was 4.5 s. It can be seen from Figure 11(b) that the lane-changing intention prediction SVM based on the sliding window designed in this study (denoted as SVM_2.2 s) predicted that the preceding vehicle had a lane-changing intention at 11.9 s. From Figure 11(a), we see that the preceding vehicle passed through the lane line of the lane in which the subject vehicle was located at 13.2 s; thus, the lane-changing intention prediction SVM based on the sliding window identified the lane-changing intention of the preceding vehicle 1.3 s in advance. Figure 11(c) shows the prediction results of the SVM that used only the motion state information of the current moment as the feature vector (denoted SVM_0 s). Short-term misjudgments were observed at 4.9 s and 6 s. SVM_0 s only used the motion state information at the current moment as the feature vector; hence, it easily made misjudgments when the motion state jumped during zigzag driving. The lane-changing intention prediction SVM designed in this paper employed the motion state information of the entire sliding window (the window size was 2.2 s); thus, it could deal with the disturbance of motion state changes produced by zigzag driving.

Compared with the traditional ACC target vehicle selection algorithm, the time advantage of the SVM-based lane-changing intention prediction output was related to many factors, including the initial relative lateral distance when the preceding vehicle began to change lanes, the overall lane-changing time, and more. Figure 12 shows the prediction results of the preceding vehicle’s lane-changing intention under three different overall lane-changing times, with the overall lane-changing times of 3.1 s, 5.0 s, and 6.9 s corresponding to Figure 12(a), (b), and (c), respectively; the lane-changing intention prediction SVM designed in this study identified the lane-changing intention of the preceding vehicle 0.9 s, 1.7 s, and 2.3 s, in advance of the traditional ACC target vehicle selection algorithm as shown in Table 2. When the overall lane-changing time increased, the advance time increased accordingly. Therefore, the advance time cannot be used as the only criterion to judge the quality of the lane-changing intention prediction SVM.

3.4 Prediction Results of Lane-changing Intention for Preceding Vehicle in the Current Lane

When the preceding vehicle in the current lane changes lanes, if a low-speed vehicle or stationary object appears ahead in the current lane, the subject vehicle will

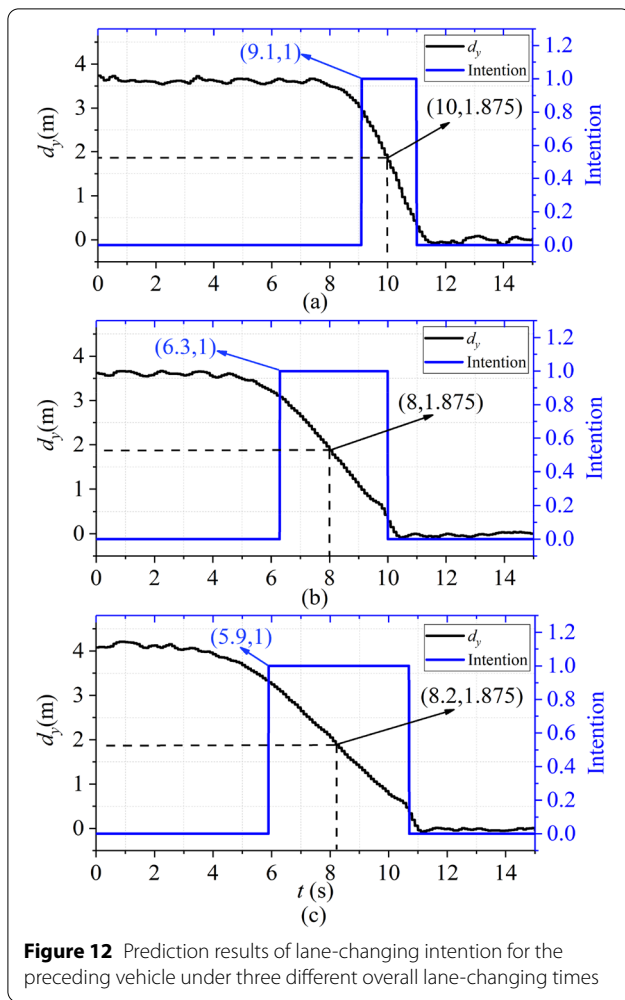


Figure 12 Prediction results of lane-changing intention for the preceding vehicle under three different overall lane-changing times

Table 2 Advance time for different overall lane-changing times (s)

Overall lane-changing time	Advance time
3.1	0.9
5.0	1.7
6.9	2.3

experience acceleration fluctuations or collision risks. Therefore, it is important to identify the lane-changing intention of the preceding vehicle in the current lane. However, because the lane-changing intention prediction SVM designed in this paper used the lateral relative distance and lateral relative speed of the preceding vehicle (relative to the centerline of the target lane) as the feature vector, it cannot be directly applied to predict the lane-changing intention of the preceding vehicle in the current lane. To solve this problem, the lateral

relative distance of the preceding vehicle in the current lane was offset left and right, respectively. The offset distance was one lane width, as shown in Figure 13.

The lateral relative distance offset does not affect the magnitude of lateral relative velocity. By inputting the offset lateral relative distance and lateral relative speed as feature vectors into the lane-changing intention prediction SVM, the lane-changing intention of the preceding vehicle in the current lane (with the left and right adjacent lanes as the target lane) could be identified. When the lane width was shifted to the left, the target lane of the preceding vehicle changed from the right adjacent lane to the current one, and the right lane-changing intention of the vehicle in the current lane was identified using the lane-changing intention prediction SVM. Likewise, when the lane width was shifted to the right, the target lane of the preceding vehicle changed from the left adjacent lane to the current lane, and the left lane-changing intention of the vehicle in the current lane was identified. The prediction results of the lane-changing intention for the preceding vehicle in the current lane are shown in Figure 14: (a) features of the left lane-changing intention prediction SVM, (b) left lane-changing intention prediction, (c) features of the right lane-changing intention prediction SVM, and (d) right lane-changing intention prediction.

Because the relative lateral distance after shifting to the left by one lane width varied as 3.7–7.5 m, the prediction results of the right lane-changing intention for the preceding vehicle were always zero, which means that the preceding vehicle did not have a right lane-changing intention. However, when the lateral relative distance offset to the right and the lateral relative velocity were taken as the feature vector, the left lane-changing intention of the preceding vehicle could be identified by the SVM at 10.6 s. Compared with the traditional ACC target vehicle selection algorithm, the lane-changing intention of the preceding vehicle in the current lane could be identified 1 s in advance.

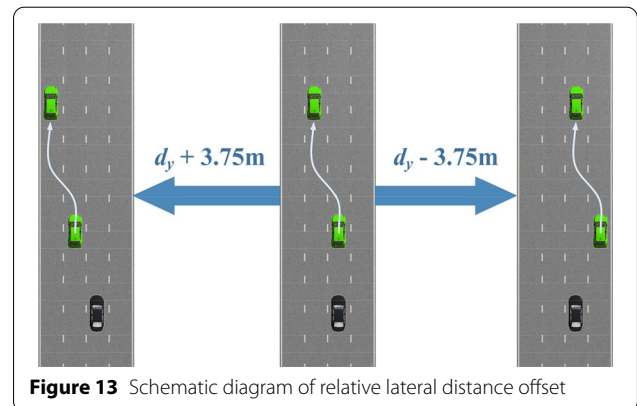


Figure 13 Schematic diagram of relative lateral distance offset

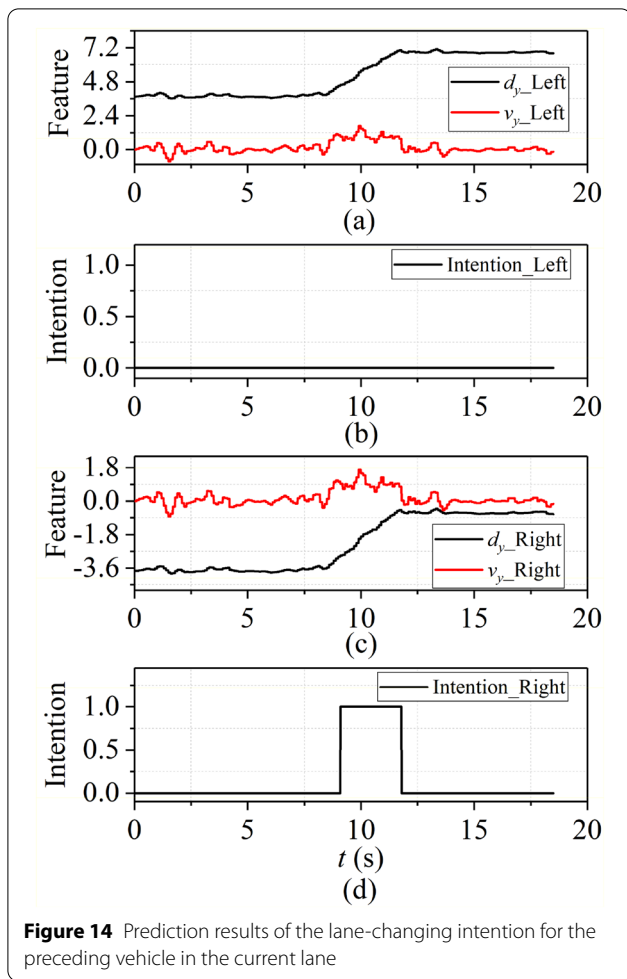


Figure 14 Prediction results of the lane-changing intention for the preceding vehicle in the current lane

4 Target Vehicle Selection based on the Prediction of the Lane-Changing Intention for the Preceding Vehicle

This study neglects the situation in which both the preceding vehicle in the current lane and that in the adjacent lane change lanes simultaneously; it only considers the situation in which one of them changes lane; here, we take the lane change of the preceding vehicle in the adjacent lane as an example to illustrate the target vehicle selection process.

To select the target vehicle, it is necessary to calculate the collision risk of each target. The collision risk is represented by TTC^{-1} in this study [25, 26]. TTC^{-1} can be calculated as

$$TTC^{-1} = -\frac{v_x}{d_x}, \tag{9}$$

where d_x is the longitudinal relative distance, v_x is the longitudinal relative speed between the preceding and subject vehicles, which equals the difference between the longitudinal speed of subject vehicle $v_{subject}$ and that of the preceding vehicle $v_{preceding}$, as shown in Figure 15.

When TTC^{-1} exceeds zero, it means that the preceding vehicle is approaching and there is a risk of collision. The collision threat increases with the increase of TTC^{-1} . When TTC^{-1} is less than zero, it indicates that the preceding vehicle is far from the subject vehicle and there is no collision risk.

According to the lane-changing intention (denoted as *Intention*) and the collision threat of each target, the targets in the adjacent lane can be classified into three types; these are represented by *DriveStaeue* [7], as shown in Figure 16.

Area 1 indicates that the preceding vehicle has no lane-changing intention ($Intention = 0$). In this case, the *DriveStaeue* is equal to zero. Area 2 indicates that the preceding vehicle has a lane-changing intention but there is no collision risk ($Intention = 1, TTC^{-1} < Th_{TTC}$); in this case, the *DriveStaeue* is equal to 1. Area 3 means that the preceding vehicle has lane-changing intention and there is a risk of collision ($Intention = 1, TTC^{-1} \geq Th_{TTC}$). In this case, the *DriveStaeue* is equal to 2.

Because there may be multiple vehicles with lane-changing intention in the adjacent lane ahead, it is necessary to select the "most threatening" of them as the target vehicle

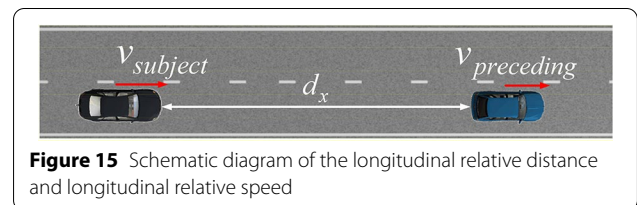


Figure 15 Schematic diagram of the longitudinal relative distance and longitudinal relative speed

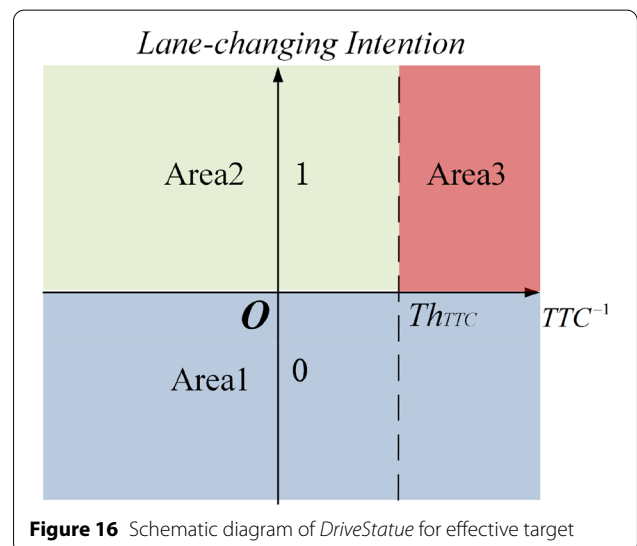


Figure 16 Schematic diagram of *DriveStaeue* for effective target

in the adjacent lane. Firstly, according to the *DriveStatue* of the targets in the adjacent lane, we can obtain the driving status with the highest priority *RDS* at the current time as the representative *DriveStatue*, expressed as

$$RDS = \max\{DriveStatue_i\}, i = 1, 2, 3, \dots, n, \quad (10)$$

where n is the number of targets in the adjacent lane.

Among the targets whose *DriveStatue* is the *RDS* in the adjacent lane, that with the smallest longitudinal relative distance from the subject vehicle is selected as the target vehicle in the adjacent lane. According to the different *RDS* values, the fusion methods for the target vehicle in the adjacent lane and the target closest to the subject vehicle in the current lane (i.e., the target vehicle obtained by the traditional ACC target vehicle selection algorithm, referred to as the target vehicle in the current lane) also differ. The values $d_{x,inlane}$ and $v_{x,inlane}$ represent the longitudinal relative distance and speed, respectively, between the target vehicle in the current lane and the subject vehicle; $d_{x,adjacent\ lane}$ and $v_{x,adjacent\ lane}$ represent the longitudinal relative distance and speed, respectively, between the target vehicle in the adjacent lane and the subject vehicle.

Case 1: $RDS = 0$, there is no vehicle in the adjacent lanes with lane-changing intention; thus, the target vehicle in the current lane can be directly selected as the target vehicle; that is,

$$d_{x,main} = d_{x,inlane}, \quad (11)$$

$$v_{x,main} = v_{x,inlane}. \quad (12)$$

Case 2: $RDS = 1$, the target vehicle in the adjacent lane has a lane-changing intention and there is no risk of collision, which means the target vehicle in the adjacent lane changes lanes safely. In this case, the target vehicle selection must fuse the target vehicle in the current lane with that in the adjacent lane, using

$$d_{x,main} = \alpha d_{x,inlane} + (1 - \alpha) d_{x,adjacent\ lane}, \quad (13)$$

$$v_{x,main} = \alpha v_{x,inlane} + (1 - \alpha) v_{x,adjacent\ lane}, \quad (14)$$

$$\alpha = \min \left\{ \frac{\left| |d_{y,Init}| - |d_{y,adjacent\ lane}| \right|}{\left| |d_{y,Init}| - 0.875 \right|}, 1 \right\}, \quad (15)$$

where $d_{y,Init}$ is the lateral relative distance of the target vehicle in the adjacent lane (relative to the center line of the lane in which the subject vehicle is located) when that vehicle is first detected as having a lane-changing intention, and $d_{y,adjacent\ lane}$ is the lateral relative distance of the target vehicle in the adjacent lane relative to the center line of the subject vehicle's lane.

During the lane-changing process of the target vehicle in the adjacent lane, $d_{y,adjacent\ lane}$ changes from $d_{y,Init}$ to 0.875 m (when the lateral relative distance of the target in the adjacent lane is less than 0.875 m, this target can be considered as the target in the current lane), and α smoothly transfers from zero to 1.

Case 3: $RDS = 2$, the target vehicle in the adjacent lane has a lane-changing intention and there is a risk of collision, which means the target vehicle in the adjacent lane changes lanes dangerously. In this case, the primary goal is to maintain the safety of the subject vehicle. Thus, the target vehicle in the adjacent lane is directly selected as the target vehicle; that is,

$$d_{x,main} = d_{x,adjacent\ lane}, \quad (16)$$

$$v_{x,main} = v_{x,adjacent\ lane}. \quad (17)$$

As shown in Figure 17, in certain cases, the preceding vehicle will cease changing lanes and return to its original lane. When this cancellation is detected, if the target vehicle is directly changed back to the target vehicle in the current lane, the longitudinal acceleration of the subject vehicle will inevitably fluctuate due to the jump of the target vehicle.

When the lane-changing intention of the target vehicle in the adjacent vehicle changes from 1 to zero, and $d_{y,adjacent\ lane}$ exceeds 0.875 m, it can be determined that the target vehicle in the adjacent vehicle has cancelled the lane change. Under lane-changing cancellation conditions, the target vehicle state is calculated as

$$d_{x,main} = \beta d_{x,inlane} + (1 - \beta) d_{x,adjacent\ lane}, \quad (18)$$

$$v_{x,main} = \beta v_{x,inlane} + (1 - \beta) v_{x,adjacent\ lane}, \quad (19)$$

$$\beta = \alpha_{cancel} \cdot \max \left\{ \frac{2.875 - |d_{y,adjacent\ lane}|}{|2.875 - |d_{y,cancel}|}, 0 \right\}, \quad (20)$$

where $d_{y,cancel}$ is the lateral relative distance of the target vehicle in the adjacent lane relative to the center line of the lane in which the subject vehicle is located, and α_{cancel} is the value of α when the target vehicle in the adjacent lane is first detected as canceling the lane-changing intention.

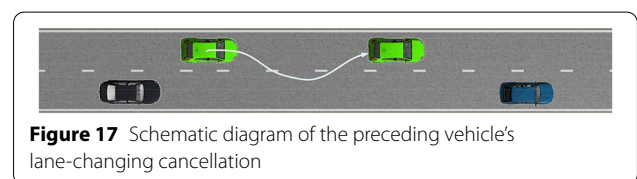


Figure 17 Schematic diagram of the preceding vehicle's lane-changing cancellation

During the lane-changing cancellation process of the target vehicle in the adjacent lane, $d_{y,adjacentlane}$ varies from $d_{y,cancel}$ to 2.875 m (when the lateral relative distance of the target in the current lane exceeds 2.875 m, this target can be considered as the target in the adjacent lane), and β smoothly transfers from α_{cancel} to zero.

5 Longitudinal Motion Control Algorithm

Depending on whether a target vehicle is ahead, the longitudinal motion control can be divided into speed control and following control. When no target vehicle is in front of the subject vehicle, only speed control is applied. For speed control, only the subject vehicle's speed $v_{subject}$ must be kept at the set speed v_{set} . Therefore, the control target in this mode is $\Delta v \rightarrow 0$ and the position error can be directly set to zero:

$$\Delta v = v_{set} - v_{subject}, \tag{21}$$

$$\Delta d = 0. \tag{22}$$

When a target vehicle is in front of the subject vehicle, the control is that of following control, which controls the speed of the subject vehicle to match that of the target vehicle, to thereby maintain a safe distance between the two. The constant time-gap safe distance is selected as the safe distance in this work [27]; it is calculated as

$$d_{des} = v_{self} \tau_h + d_0, \tag{23}$$

where τ_h is the time gap constant, generally set to 1.2–2 s. d_0 is the distance constant, generally set to 2–3 m. In this study, τ_h was set to 2 s, and d_0 was set to 3 m.

In the following control, the subject vehicle speed must be kept the same as that of the target vehicle, and the distance d_x between the subject and target vehicles must be controlled as the safe distance d_{des} ; thus, the control target in this mode is $\Delta v \rightarrow 0, \Delta d \rightarrow 0$, where

$$\Delta v = v_x, \tag{24}$$

$$\Delta d = d_x - d_{des}. \tag{25}$$

A linear-quadratic regulator (LQR) controller was chosen to calculate the desired acceleration of the subject vehicle in this study. The balance state in the longitudinal motion control is $\Delta v \rightarrow 0, \Delta d \rightarrow 0$; thus, it is very suitable to use the LQR controller to calculate the desired acceleration of the subject vehicle. Meanwhile, the LQR controller can consider the weight of the input and state variables to ensure ride comfort during longitudinal motion control.

Time delays can arise between the actual acceleration a_{actual} and inputted desired acceleration a_{des} ; these can

be approximately represented by a one-order inertia element, as

$$a_{actual} = \frac{1}{\tau_d s + 1} a_{des}, \tag{26}$$

where τ_d is the time delay between the actual acceleration a_{actual} and the inputted desired acceleration a_{des} , which was here set to 0.5 s.

Selecting the state variable as $x = [\Delta d, \Delta v, a_{actual}]^T$ and the input variable as a_{des} , we can obtain the continuous state space equation for longitudinal acceleration control as

$$\begin{bmatrix} \Delta \dot{d} \\ \Delta \dot{v} \\ \dot{a}_{actual} \end{bmatrix} = \begin{bmatrix} 0 & 1 & -\tau_h \\ 0 & 0 & -1 \\ 0 & 0 & -1/\tau_d \end{bmatrix} \begin{bmatrix} \Delta d \\ \Delta v \\ a_{actual} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/\tau_d \end{bmatrix} a_{des} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} a_{tar}, \tag{27}$$

where a_{tar} is the acceleration of the target vehicle, which represents an interference term.

Discretizing the above continuous state space equation, we obtain

$$\begin{bmatrix} \Delta d \\ \Delta v \\ a_{actual} \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & T & -\tau_h T \\ 0 & 1 & -T \\ 0 & 0 & 1 - T/\tau_d \end{bmatrix} \begin{bmatrix} \Delta d \\ \Delta v \\ a_{actual} \end{bmatrix}_k + \begin{bmatrix} 0 \\ 0 \\ T/\tau_d \end{bmatrix} a_{des} + \begin{bmatrix} 0 \\ T \\ 0 \end{bmatrix} a_{tar}, \tag{28}$$

where T is the control cycle.

Because the ride comfort is sizably affected by the jerk (the derivative of the acceleration), the above-mentioned state space equation cannot take into account the weight of the jerk. Therefore, we expanded the discrete state space equation to an incremental form, and took the desired acceleration increment Δa_{des} as an input to incorporate the weight of the jerk. The expanded state space equation is as follows:

$$\begin{bmatrix} \Delta d \\ \Delta v \\ a_{actual} \\ a_{des} \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & T & -\tau_h T & 0 \\ 0 & 1 & -T & 0 \\ 0 & 0 & 1 - T/\tau_d & T/\tau_d \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta d \\ \Delta v \\ a_{actual} \\ a_{des} \end{bmatrix}_k + \begin{bmatrix} 0 \\ 0 \\ T/\tau_d \\ 1 \end{bmatrix} \Delta a_{des} + \begin{bmatrix} 0 \\ T \\ 0 \\ 0 \end{bmatrix} a_{tar}. \tag{29}$$

The objective function of the LQR controller is:

$$J = \frac{1}{2} \int_0^\infty [q_{l1} \cdot \Delta d^2 + q_{l2} \cdot \Delta v^2 + q_{l3} \cdot a_{actual}^2 + q_{l4} \cdot a_{des}^2 + r_l \cdot u^2] dt, \tag{30}$$

where u is the desired acceleration increment Δa_{des} ; q_{l1} , q_{l2} , q_{l3} , q_{l4} , and r_l represent the weight of Δv , Δd , a_{actual} , a_{des} , and Δa_{des} , respectively. Here, $q_{l1} = 2$, $q_{l2} = 1$, $q_{l3} = 0$, $q_{l4} = 3$, and $r_l = 3$.

After the desired acceleration of the subject vehicle was calculated by the LQR controller, it was necessary to control the actuator of the subject vehicle (i.e., the throttle opening and brake master cylinder pressure) to ensure that the actual acceleration of the subject vehicle approached the calculated desired acceleration. This paper first established the inverse dynamics model for the subject vehicle. Through this model, the feedforward control quantity of the actuator could be obtained. Owing to the deviation of the subject vehicle's inverse dynamics model parameters and the presence of interference, it was difficult to make the actual acceleration approach the desired one stably via open-loop control alone. A large static error was produced. Therefore, to improve the accuracy and robustness of the longitudinal acceleration control, we took the deviation value between the actual vehicle acceleration and desired acceleration as the input, and we used the proportional-integral-derivative (PID) controller to calculate the feedback control quantity of the actuator.

6 Simulation and Discussion

Next, a co-simulation platform was built using Matlab/Simulink, CarSim, and Prescan software, to verify the proposed algorithm. The scenario and sensor models were established in Prescan. The measurement data of the millimeter wave radar model in Prescan contain noise, which can simulate radar measurement data in the real world to a certain extent. The high-precision

vehicle dynamics model was established in CarSim, and the simulation environment integration and control algorithm was established in Matlab/Simulink, as shown in Figure 18. Simulations were conducted under three different conditions: safe lane-changing, dangerous lane-changing, and lane-changing cancellation.

6.1 Simulation Results under Safe Lane-Changing Condition

To verify the effectiveness of the target vehicle selection algorithm proposed in this paper under safe lane-changing conditions, the following simulation conditions were designed in the co-simulation platform: Initially, the subject vehicle followed the preceding vehicle in the current lane at the set speed (25 m/s), and the longitudinal relative distance between the subject and preceding

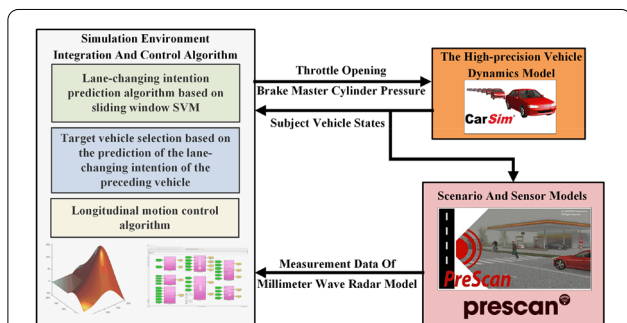


Figure 18 Closed-loop block diagram of Matlab/Simulink, CarSim, and Prescan in the co-simulation platform

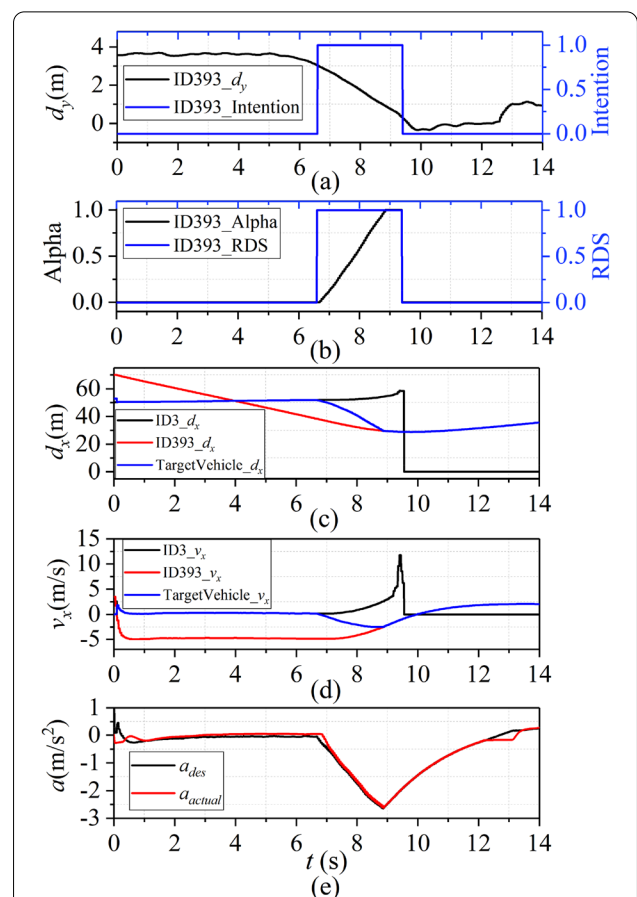
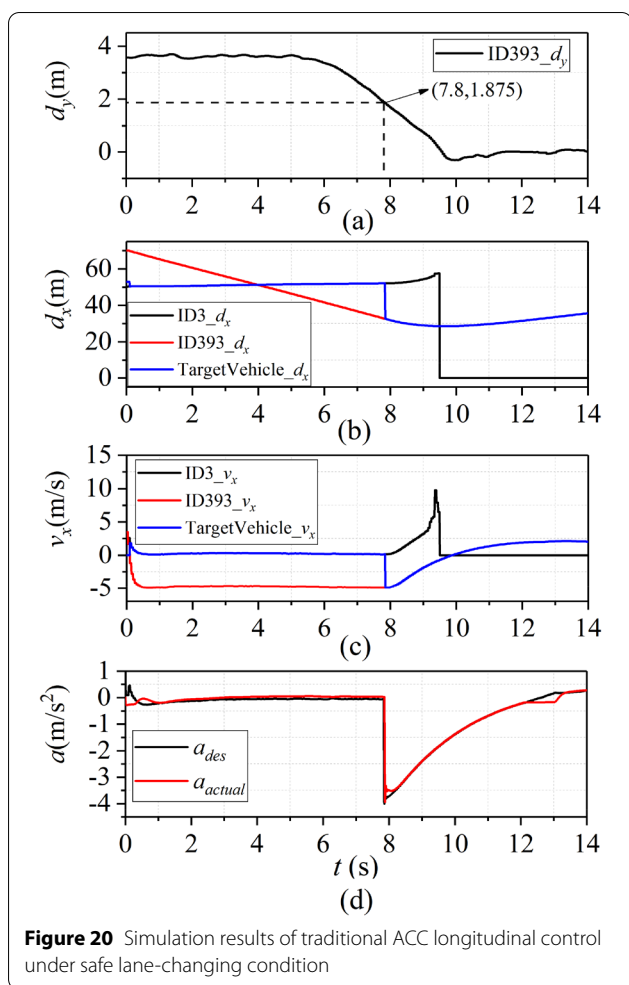


Figure 19 Simulation results of longitudinal control based on lane-changing intention prediction of preceding vehicle under safe lane-changing condition



vehicle in the current lane was 50 m. A preceding vehicle was present in the left adjacent lane. The driving speed of the preceding vehicle in the adjacent lane at the start of the simulation was 18 m/s, and the longitudinal relative distance to the subject vehicle was 70 m. The preceding vehicle began to change lanes 5 s after the start of the simulation. The simulation results are shown in Figures 19 and 20.

At 6.6 s after the start of the simulation, the lane-changing intention prediction algorithm based on the sliding window SVM detected that the preceding vehicle in the adjacent lane intended to change lanes; here, the *RDS* was 1, which meant that the target vehicle in the adjacent lane had a lane-changing intention and there was no risk of collision, as shown in Figure 19(a)–(b). Therefore, the target vehicle selection algorithm fused the target vehicle in the current lane with that in the adjacent one. The result of the fusion is shown in Figure 19(c)–(d). The target vehicle smoothly transitioned from the target vehicle in the current lane (ID3) to that in the adjacent one (ID393). As shown in Figure 20(b)–(c), when using the

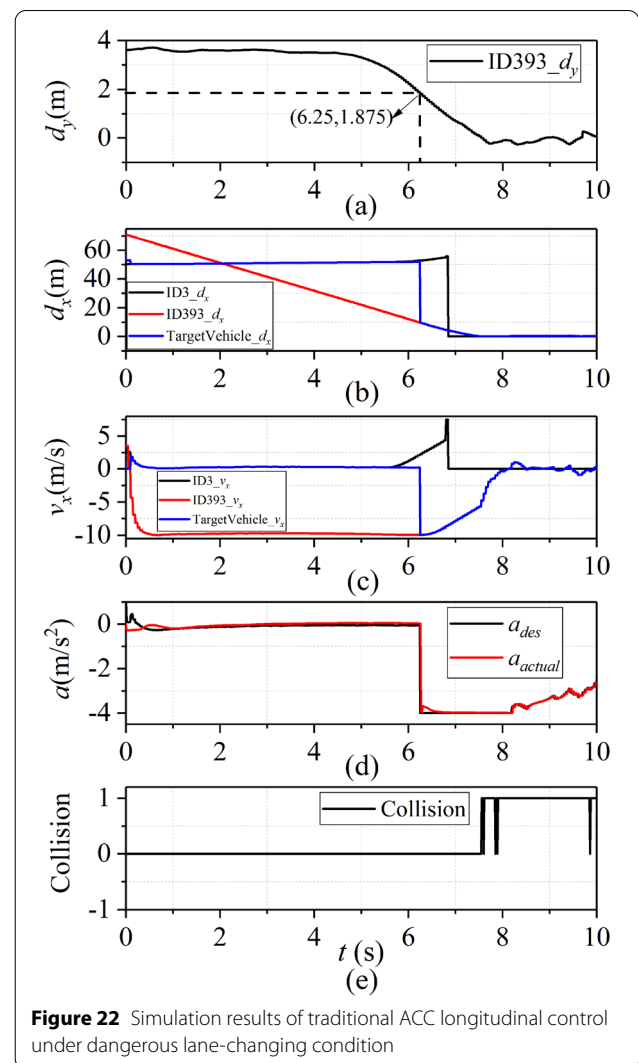
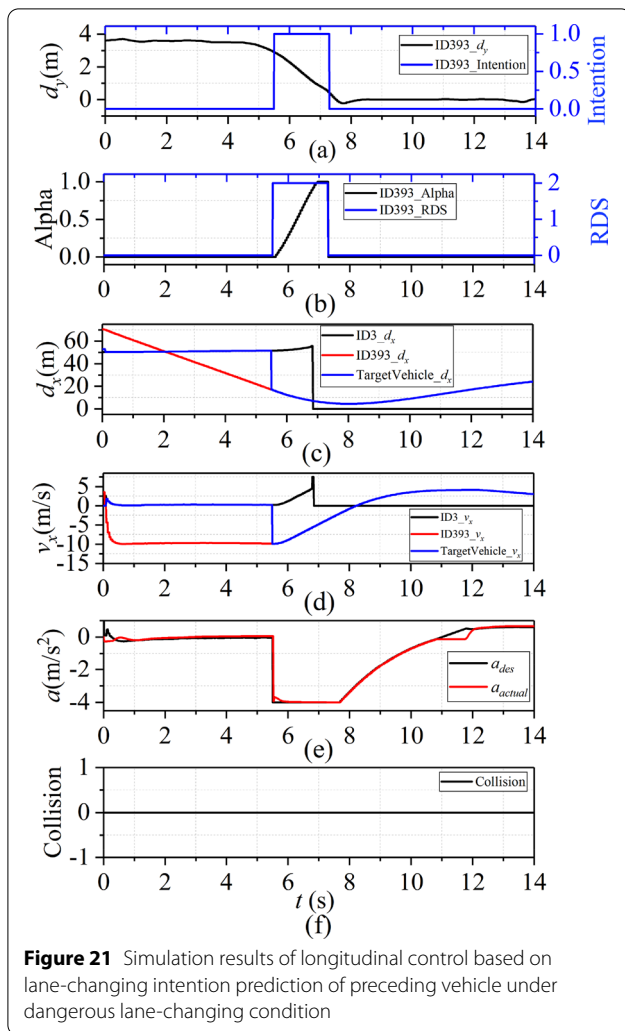
target vehicle selection method of the traditional ACC system, the target vehicle jumped directly from the target vehicle in the current lane to that in the adjacent lane at 7.8 s. The lane-changing intention prediction algorithm based on the sliding window SVM here designed identified the lane-changing intention of the preceding vehicle 1.2 s earlier than the traditional target vehicle selection method. In addition, the state of the target vehicle in the current lane changed suddenly at 9.4 s, as shown in Figure 19(c)–(d). This was because the target vehicle in the current lane was blocked and could not be detected by the subject vehicle sensors when the lane change of the target vehicle in the adjacent lane was completed.

Figures 19(e) and 20(d) show the longitudinal acceleration curve of the subject vehicle under safe lane-changing conditions. It can be seen from the simulation results that the maximum longitudinal deceleration of the subject vehicle was 2.62 m/s² during the entire control process, when using the proposed target vehicle selection algorithm here. When using the traditional target vehicle selection method, the maximum longitudinal deceleration of the subject vehicle was 3.90 m/s². The target vehicle selection algorithm here proposed responded faster (1.2 s earlier) to the lane change of the preceding vehicle in the adjacent lane. Furthermore, the corresponding maximum longitudinal deceleration was reduced by 1.28 m/s². This can effectively reduce the subject vehicle's longitudinal acceleration fluctuations caused by the safe lane change of a preceding vehicle in the adjacent lane, and it thereby improves the riding comfort.

6.2 Simulation Results under Dangerous Lane-changing Condition

Initially, the subject vehicle followed the preceding vehicle in the current lane at the set speed, the driving speed was 25 m/s, and the longitudinal relative distance between the subject and preceding vehicle in the current lane was 50 m. The driving speed of the preceding vehicle in the adjacent lane at the start of the simulation was 15 m/s, and the longitudinal relative distance to the subject vehicle was 70 m. The preceding vehicle began to change lanes 4.5 s after the start of the simulation. The simulation results are shown in Figures 21 and 22.

At 5.5 s after the start of the simulation, the lane-changing intention prediction algorithm based on sliding window SVM detected that the preceding vehicle in the adjacent lane had a lane-changing intention; here, the *RDS* was 2, which means that the target vehicle in the adjacent lane had a lane-changing intention and there was a risk of collision, as shown in Figure 21(a)–(b). In this case, priority should be given to the target vehicle in the adjacent lane. The result is shown in Figure 21(c)–(d).



The target vehicle selection algorithm here proposed jumped directly from the target vehicle in the current lane to that in the adjacent one when the lane-changing intention was detected. According to Figure 22(b)–(c), when using the target vehicle selection method of the traditional ACC algorithm, the target vehicle jumped directly from the target vehicle in the current lane to that in the adjacent lane at 6.25 s when the target vehicle in the adjacent lane crossed the lane line. The lane-changing intention prediction algorithm based on the sliding window SVM identified the lane-changing intention of the preceding vehicle 0.75 s earlier than the traditional ACC target vehicle selection algorithm.

Figures 21(e) and 22(d) show the longitudinal acceleration curve of the subject vehicle under dangerous lane-changing conditions. Figures 21(f) and 22(e) show the collision signal between the subject vehicle and the surrounding traffic vehicles. The simulation results show

that the maximum longitudinal deceleration of the subject vehicle under the target vehicle selection algorithm here proposed and the target vehicle selection method of the traditional ACC system both reached the maximum value of 4 m/s². However, the proposed target vehicle selection algorithm responded faster (0.75 s earlier) to the lane change of the preceding vehicle in the adjacent lane. In Figures 21(f) and 22(e), the subject vehicle can be seen to collide with the target vehicle in the adjacent lane at 7.56 s when using the target vehicle selection method of the traditional ACC system. Because of the proposed target vehicle selection method, the subject vehicle could decelerate 0.75 s in advance, and the minimum longitudinal relative distance between the subject and target vehicle in the adjacent lane was 4.5 m, effectively avoiding any collision.

6.3 Simulation Results under Lane-changing Cancellation Condition

Initially, the subject vehicle followed the preceding vehicle in the current lane at the set speed, the driving speed was 25 m/s, and the longitudinal relative distance between the subject and preceding vehicle in the current lane was 50 m. The driving speed of the preceding vehicle in the adjacent lane at the start of the simulation was 20 m/s, and the longitudinal relative distance to the subject vehicle was 70 m. The preceding vehicle began to change lanes 4.5 s after the start of the simulation. The simulation results are shown in Figures 23 and 24.

At 5.7 s after the start of the simulation, the lane-changing intention prediction algorithm based on the sliding window SVM detected that the preceding vehicle in the adjacent lane had a lane-changing intention; here, the RDS was 1, which means the target vehicle in the adjacent lane had a lane-changing intention and there was no risk of collision, as shown in Figure 23(a)–(b). Therefore, the target vehicle selection algorithm fused the target

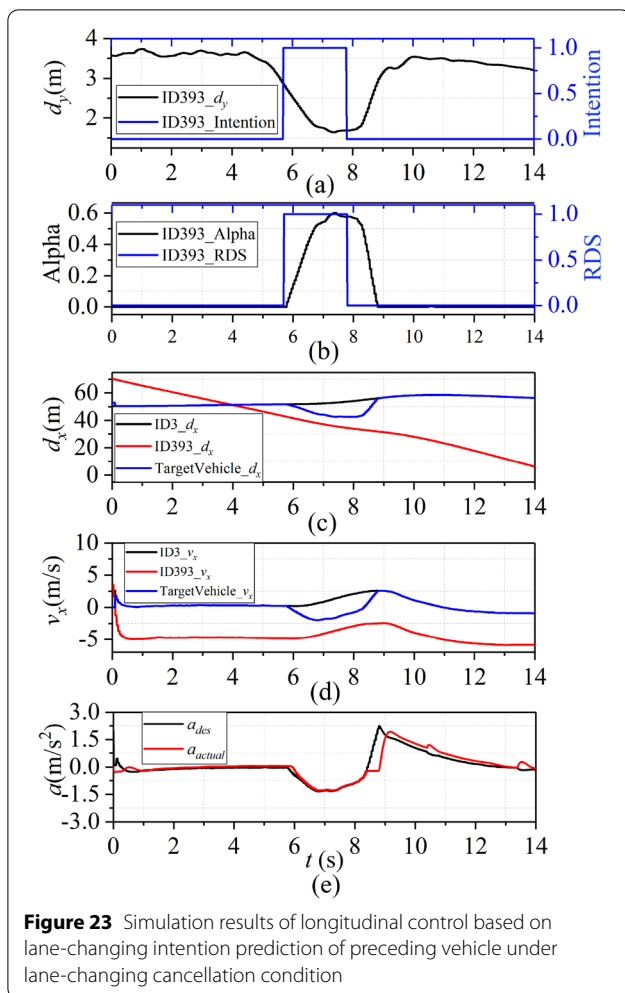


Figure 23 Simulation results of longitudinal control based on lane-changing intention prediction of preceding vehicle under lane-changing cancellation condition

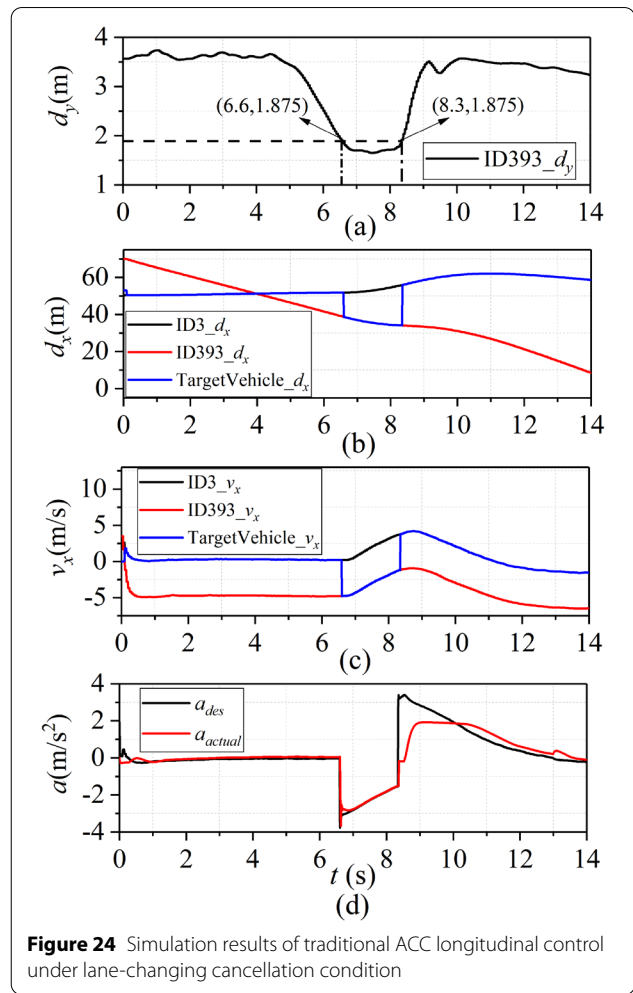


Figure 24 Simulation results of traditional ACC longitudinal control under lane-changing cancellation condition

vehicle in the current lane with that in the adjacent lane. The result of the fusion is shown in Figure 23(c)–(d). The target vehicle smoothly transitioned from the target vehicle in the current lane (ID3) to that in the adjacent lane (ID393).

Before the lane change was completed, the proposed lane-changing intention prediction algorithm detected that the target vehicle in the adjacent lane cancelled the lane change maneuver at 7.8 s, and α was 0.58 at this time, as shown in Figure 23(a)–(b). If the target vehicle directly switches back to the target vehicle in the current lane, it will inevitably lead to the jump of the target vehicle. Therefore, it is necessary to select the target vehicle according to the target vehicle selection algorithm under the lane-changing cancellation condition, so that the target vehicle can smoothly transition back to the target vehicle in the current lane. The target vehicle information is shown in Figure 23(c)–(d).

When using the target vehicle selection method of the traditional ACC system, the target vehicle jumped

directly from the target vehicle in the current lane to that in the adjacent lane, because the target vehicle in the adjacent lane crossed the left lane line at 6.6 s. Furthermore, at 8.3 s, because the target vehicle in the adjacent lane crossed the left lane line and returned to its original lane, the target vehicle switched from the target vehicle in the adjacent lane back to that in the current lane. Figures 23(e) and 24(d) show the longitudinal acceleration of the subject vehicle under the lane-changing cancellation condition. It can be seen from the simulation results that the maximum longitudinal deceleration of the subject vehicle was 1.94 m/s^2 during the entire control process, when using the target vehicle selection algorithm proposed in this paper. When using the target vehicle selection method of the traditional ACC system, the maximum longitudinal deceleration of the subject vehicle was 3.70 m/s^2 . The maximum longitudinal deceleration was reduced by 1.28 m/s^2 . However, the maximum acceleration was almost identical. This is because at the current speed, the acceleration of the subject vehicle was limited, which means that, within 8.5–10 s of the start of the simulation, the throttle opening of the subject vehicle had reached 100%. However, from the desired acceleration curve, it can also be seen that through the smooth transition of the target vehicle, the maximum desired acceleration was reduced by 1.14 m/s^2 when using the proposed target vehicle selection algorithm (the maximum desired acceleration was 2.24 m/s^2) compared with the target vehicle selection method of the traditional ACC system (the maximum desired acceleration was 3.38 m/s^2).

7 Conclusions

In this paper, a target vehicle selection algorithm based on the prediction of the preceding vehicle's lane-changing intention was proposed. This lane-changing intention was identified by the lane-changing intention prediction algorithm based on the sliding window SVM, as trained on the NGSIM dataset. The lane-changing intention prediction algorithm proposed in this paper was applicable to the preceding vehicle both in the current lane and in the adjacent one. Through comparisons with the target vehicle selection method of the traditional ACC system, the simulation results indicate that the target vehicle selection algorithm proposed in this paper can respond to the lane change of the preceding vehicle in advance, thereby effectively reducing the longitudinal acceleration fluctuation and avoiding collisions under dangerous lane-changing conditions.

As future work, the trajectory of the preceding vehicle will be predicted, to further improve the driving safety of the subject vehicle. Meanwhile, the proposed algorithm

will be verified on a real vehicle platform, to verify the real-time ability of the algorithm and its robustness to interference in real road environments.

Acknowledgements

Not applicable.

Authors' contributions

JY and GC were in charge of the whole trial; JY wrote the manuscript; ZG reviewed the manuscript. All authors read and approved the final manuscript.

Authors' Information

Jun Yao, born in 1995, is currently a Ph.D student at *College of Automotive Engineering, Jilin University, Changchun, China*. He received his bachelor degree from *Jilin University, China*, in July 2020. His research interests include vehicle dynamics and control, and autonomous vehicles.

Guoying Chen, born in 1984, is currently an associate professor at *Jilin University, China*. He received his PhD degree from *Jilin University, China*, in 2012. His research interests include control-by-wire vehicle control and autonomous vehicles.

Zhenhai Gao, is currently a professor at in *Jilin University, China*. Dr. Gao received his Ph.D. degree in automotive engineering from *Jilin University, 2000*. He was a foreign researcher of the *University of Tokyo*. In recent years, his research has been focused on connected & automated vehicles, driver behavior analysis, advanced driver assistance system, and human-machine interface.

Funding

Supported by National Key Research and Development Program (Grant No. 2017YFB0102601), and National Natural Science Foundation of China (Grant Nos. 51775236, U1564214).

Competing interests

The authors declare no competing financial interests.

Received: 7 October 2020 Revised: 1 September 2021 Accepted: 11 November 2021

Published online: 11 December 2021

References

- [1] Shah, Puneet. (2019). Adaptive Cruise Control (ACC) Market to Display Significant Growth During Forecast Period. <https://doi.org/10.13140/RG.2.2.27676.44163>.
- [2] J Ziegler, P Bender, M Schreiber, et al. Making bertha drive—an autonomous journey on a historic route. *IEEE Intelligent Transportation Systems Magazine*, 2014, 6(2): 8–20.
- [3] J Kim, J Kim, G J Jang, et al. Fast learning method for convolutional neural networks using extreme learning machine and its application to lane detection. *Neural Netw*, 2017, 87: 109–121.
- [4] D Umberto, D Shilp, F Saber, et al. Towards connected autonomous driving: review of use-cases. *Vehicle System Dynamics*, 2018: 1–36.
- [5] X Chen, J Zhang, Y Liu. Research on the intelligent control and simulation of automobile cruise system based on fuzzy system. *Mathematical Problems in Engineering*, 2016, 6: 1–12.
- [6] J Martinez, C Canudas-De-Wit. A safe longitudinal control for adaptive cruise control and stop-and-go scenarios. *IEEE Transactions on Control Systems Technology*, 2007, 15:246–258.
- [7] S Moon, H Kang, K Yi. Multi-vehicle target selection for adaptive cruise control. *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, 2010, 48(11): 1325–1343.
- [8] D Kim, S Moon, H Kim, et al. Design of an adaptive cruise control / collision avoidance with lane change support for vehicle autonomous driving. *ICCAS-SICE*. IEEE, 2009.
- [9] G Ma, Z Liu, X Pei, et al. Identification of cut-in maneuver of side lane vehicles based on fuzzy support vector machines. *Automotive Engineering*, 2014, 36(3): 316–320. (in Chinese)

- [10] Y Zou, S Wei, G Ma, et al. A research on the side-lane vehicle cut-in control in adaptive cruise control system. *Automotive Engineering*, 2016, 38(3):323–329. (in Chinese)
- [11] H Woo, Y Ji, H Kono, et al. Lane-change detection based on vehicle-trajectory prediction. *IEEE Robotics and Automation Letters*, 2017: 1. <https://doi.org/10.1109/LRA.2017.2660543>.
- [12] S Ma. *Research on intelligent vehicle lane changing behavior judgment and collision risk estimation*. Harbin: Harbin Institute of Technology, 2016. (in Chinese)
- [13] K Zhang. *Recognition method of lane-changing behaviour and danger level for intelligent vehicle*. Harbin: Harbin Institute of Technology, 2017. (in Chinese)
- [14] D Mitrovic. Reliable method for driving events recognition. *IEEE Transactions on Intelligent Transportation Systems*, 2005, 6(2): 198–205.
- [15] H Zhang, R Fu. Driving behavior recognition and intention prediction of adjacent preceding vehicle in highway scene. *Journal of Transportation Systems Engineering and Information Technology*, 2020, 20(1): 40–46. (in Chinese)
- [16] S Yoon, H Jeon, D Kum. Predictive cruise control using radial basis function network-based vehicle motion prediction and chance constrained model predictive control. *IEEE Transactions on Intelligent Transportation Systems*, 2019(99): 1–12.
- [17] D Lee, Y Kwon, S McMains, et al. Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC. *IEEE 20th International Conference on Intelligent Transportation Systems*. IEEE, 2017.
- [18] FHWA, US Department of Transportation. NGSIM–Next Generation Simulation. (2020–3–11). <http://ops.fhwa.dot.gov/trafficanalysisstools/ngsim.htm>. Accessed July 30, 2012.
- [19] P Kumar, M Perrollaz, S Lefevre, et al. Learning based approach for online lane change intention prediction. *Proceedings of the IEEE International Conference on Intelligent Vehicle Symposium*, 2013: 797–802.
- [20] K P Bennett, E J Bredensteiner. Duality and geometry in SVM classifiers. *Proceedings of the 17th International Conference on Machine Learning*, 2000: 57–64.
- [21] S Amsalu, A Homaifar, F Afghah, et al. Driver behavior modeling near intersections using support vector machines based on statistical feature extraction// *Intelligent Vehicles Symposium*. IEEE, 2015.
- [22] M Werling, J Ziegler, S Kammel, et al. Optimal trajectory generation for dynamic street scenarios in a frenet frame. *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010.
- [23] M P Do Carmo. *Differential geometry of curves and surfaces*. Prentice-Hall, 1976.
- [24] V N Vapnik. *The nature of statistical learning theory*. Springer, 1995.
- [25] P Fancher, Z Bareket, R Ervin. Human-centered design of an acc-with-braking and forward-crash-warning system. 2001, 36(2–3): 203–223.
- [26] Manjiang Hu, Yuan Liao, Wenjun Wang, et al. Decision tree-based maneuver prediction for driver rear-end risk-avoidance behaviors in cut-in scenarios. *Journal of Advanced Transportation*, 2017: 7170358.
- [27] J Wang, R Rajamani. Should adaptive cruise-control systems be designed to maintain a constant time gap between vehicles? *IEEE Transactions on Vehicular Technology*, 2004, 53(5): 1480–1490.

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
