Transportation Research Part C 141 (2022) 103738

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

An integrated lane change prediction model incorporating traffic context based on trajectory data

Qingwen Xue, Yingying Xing *, Jian Lu

The Key Laboratory of Road and Traffic Engineering, Ministry of Education, Tongji University, 4800 Cao' an Road, Shanghai, PR China

ARTICLE INFO

Keywords:
Lane change prediction
Traffic context
Naturalistic trajectory
Machine learning algorithms

ABSTRACT

Predicting lane change maneuvers is critical for autonomous vehicles and traffic management as lane change may cause conflict in traffic flow. Most existing studies do not consider the effect of traffic context (i.e., traffic level and vehicle type) on lane change maneuvers. Therefore, these models cannot adapt to different traffic environments. This study aims to address this problem and establish an integrated lane change prediction model incorporating traffic context using machine learning algorithms. In addition, lane change decisions and lane change trajectories are both predicted to capture the whole process, which have been less studied. The framework of the proposed model contains two parts: the traffic context classification model, which is used to predict traffic level and vehicle type, and the integrated lane change prediction model, which is used to predict lane change decision with XGBoost and lane change trajectories with LSTM incorporating context information. Instead of considering lane change, we establish trajectory prediction models for left lane change and right lane change, further improving the prediction accuracy. The naturalistic trajectories of the highD dataset are used to train and validate the model. The results show that the proposed model improves the accuracy from 97.02% to 98.20% when predicting lane change decision that incorporate traffic context. In addition, the MSE decreases from 11.21 to 6.62 when predicting trajectories. The proposed models are also validated on NGSIM dataset, proving the adaptability of the model. The proposed model can be applied to different environments to reduce collision risks caused by lane change maneuvers and improve traffic management and driving safety.

1. Introduction

Drivers usually exhibit lane change behaviors when they intend to improve their driving conditions, merge, or diverge across multilane traffic streams (Chen et al., 2019). Lane change behavior requires the interaction between the subject vehicle and surrounding vehicles and can easily cause disturbances in traffic flow and increase the probability of collisions. It has been reported that lane change crashes account for 4 to 10% of all crashes (Barr and Najm, 2001). Lane change behavior is also considered to be responsible for most traffic instabilities, such as capacity and speed decreases (Chen and Ahn, 2018). In addition, lane change behavior plays an essential role in the development of advanced driver-assistance systems (ADASs) and connected and autonomous vehicles (CAVs) (Yang et al., 2019; Norouzi et al., 2019; Xiong et al., 2021). Accurate and robust lane change prediction can promote active traffic safety by warning a driver of potential collisions and actively intervening in the lateral control of autonomous vehicles.

* Corresponding author.
E-mail address: yingying199004@tongji.edu.cn (Y. Xing).

https://doi.org/10.1016/j.trc.2022.103738
Received 4 May 2021; Received in revised form 19 May 2022; Accepted 21 May 2022
Available online 31 May 2022

Therefore, analyzing lane change behavior is of great importance for improving traffic safety and modeling roadway capacity.

With the realization that lane change prediction has a significant influence on road safety and traffic congestion, research on modeling lane change, especially on lane change decisions/intention predictions, has rapidly increased over the last decade. The lane change decision (LCD) model can be classified into two methods: model-driven methods and data-driven methods. For model-driven methods, it is important to establish an LCD model that can infer the driver's intention based on their driving characteristics (Gipps, 1986). Cellular automata (CA) models have been extended to LCDs by adding lane change rules to the CA-based car-following (CF) model (Chowdhury et al., 2000). Similar to traditional lane change models, CA-based lane change models concentrate on the desirability, necessity and gap acceptance of a lane change. Singh and Li (2011) established a state space model incorporating a Markov chain to estimate the probability of each vehicle changing to another lane or staying in their current lane. Mathematical models based on kinematic features (i.e., velocity and steering wheel movement) have also been established to estimate the lane change probability (Schmidt et al. 2014; Butakov and Ioannou, 2014; Zhang et al., 2020). The model-driven method has parameter calibration problems and has been proven to reflect general driving decision rules.

In recent years, with the rapid development of data collection and communication technologies, the data-driven method for predicting LCDs has greatly increased (Wissing et al., 2017; Kumar et al., 2013). Compared with the traditional model-driven method, the data-driven method concentrates on learning driving lane change behavior from vehicle-related data, such as vehicle dynamic data and vehicle trajectories. Some researchers have adopted the HMM combined with the feature extraction method to recognize a driver's lane change intention (Li et al., 2015; Li et al., 2016; Klitzke et al., 2020). Neural network-based algorithms have also been applied to recognize lane change decisions (Xie et al., 2019; Shou et al., 2020; Rákos et al., 2020). Leonhardt and Wanielik (2018) established an artificial neural network that fused features extracted from environmental situations, gaze behavior and vehicle movement to detect an LCD. Shi et al. (2019) proposed a hierarchical reinforcement learning-based structure for lane change decision-making and control, which was applied to decide when to conduct a lane change maneuver. Xing et al. (2020) adopted the RNN-LSTM method to model time-series driving behavior and infer lane change intention based on vision-based signals. Shou et al. (2020) proposed a longer-time ($5 \sim 10$ s) lane change prediction model without any lateral or angle information by MLP, RNN and a logistic regression algorithm. In addition to trajectory data, the driver's psychology and driving style are also considered in LCD modeling (Li et al., 2020b). Most research has concentrated only on lane change intention/decision interference, not trajectory prediction. Predicting lane change trajectories can provide an important reference for surrounding vehicles to reduce collision risk and improve traffic efficiency. LSTM (long short-term memory) is the most applicable method to predict the trajectory of lane change (Messaoud et al., 2019; Xie et al. 2019). Xie et al. (2019) used the trajectory information of a subject vehicle and its surrounding vehicles as the inputs of the model to predict the two-dimensional trajectory of a subject vehicle with an LSTM-based algorithm. However, they did not consider the impact of traffic environment.

The traffic context has been proven to play an important role in supporting automated driving and intelligent transportation systems (Li et al., 2021). Some researchers considered the traffic context, which was defined as the relative relationships between vehicles in their model, such as gaps, relative speeds or the existence of surrounding vehicles (Wissing et al., 2017; Li et al., 2018; Wang et al., 2021). However, traffic environments and vehicle features, such as traffic flow levels and vehicle types, are also traffic contextual information related to driving behavior (Bejani and Ghatee, 2018), which are rarely studied in lane change prediction. The traffic context is different in various driving environments and will lead to different maneuvers. For example, drivers tend to change lanes to obtain higher speeds when encountering a low-speed vehicle in front of them in free flow conditions. In addition, some drivers prefer to stay in the same lane under congested flow conditions. Establishing an adaptive lane change prediction model that can be applicable in different traffic environments is an important problem that has rarely been considered when modeling lane change. Furthermore, most models consider left lane change and right lane change as the same lane change maneuvers. However, they have different trajectories, and right lane change have a more negative impact on traffic flow and crash risk (Li et al., 2020a).

The research gap for the present lane change prediction model can be summarized in two aspects. First, most research only concentrates on modeling lane change intentions/decisions while ignoring the prediction of lane change trajectories. Second, traffic environments and vehicle features are rarely considered in the lane change prediction model, which has a significant impact on lane change maneuvers. To address the above research gaps, an integrated lane change prediction model incorporating traffic context with trajectory data is proposed. The proposed framework consists of two parts: a traffic context classification model and an integrated lane change prediction model. The traffic context classification model is established to recognize the traffic level and vehicle type of lane change maneuvers, as they both have a significant impact on driving behaviors. The integrated lane change prediction model predicts the lane change decisions and lane change trajectories based on historical trajectory data and traffic context. The contributions of the paper are threefold:

(1) An integrated lane change prediction model based on trajectory data is proposed to capture the whole lane change process, that is, lane change decisions and lane change trajectories.
(2) The traffic context (i.e., traffic flow and vehicle type) is integrated into the lane change prediction model. In addition, left lane change and right lane change are investigated, as there is a difference between the two maneuvers.
(3) Advanced machine learning algorithms, i.e., XGBoost and LSTM, are adopted to predict lane change decisions and lane change trajectories, respectively. In particular, the LSTM-based lane change trajectory model accounts for historical trajectory sequences rather than instantaneous data, which can better describe and capture the lane change process.

The rest of this paper is organized as follows. Section 2 presents the details of the integrated lane change prediction model incorporating traffic context classification. Section 3 gives a brief description of the data used in this study and carefully evaluates the

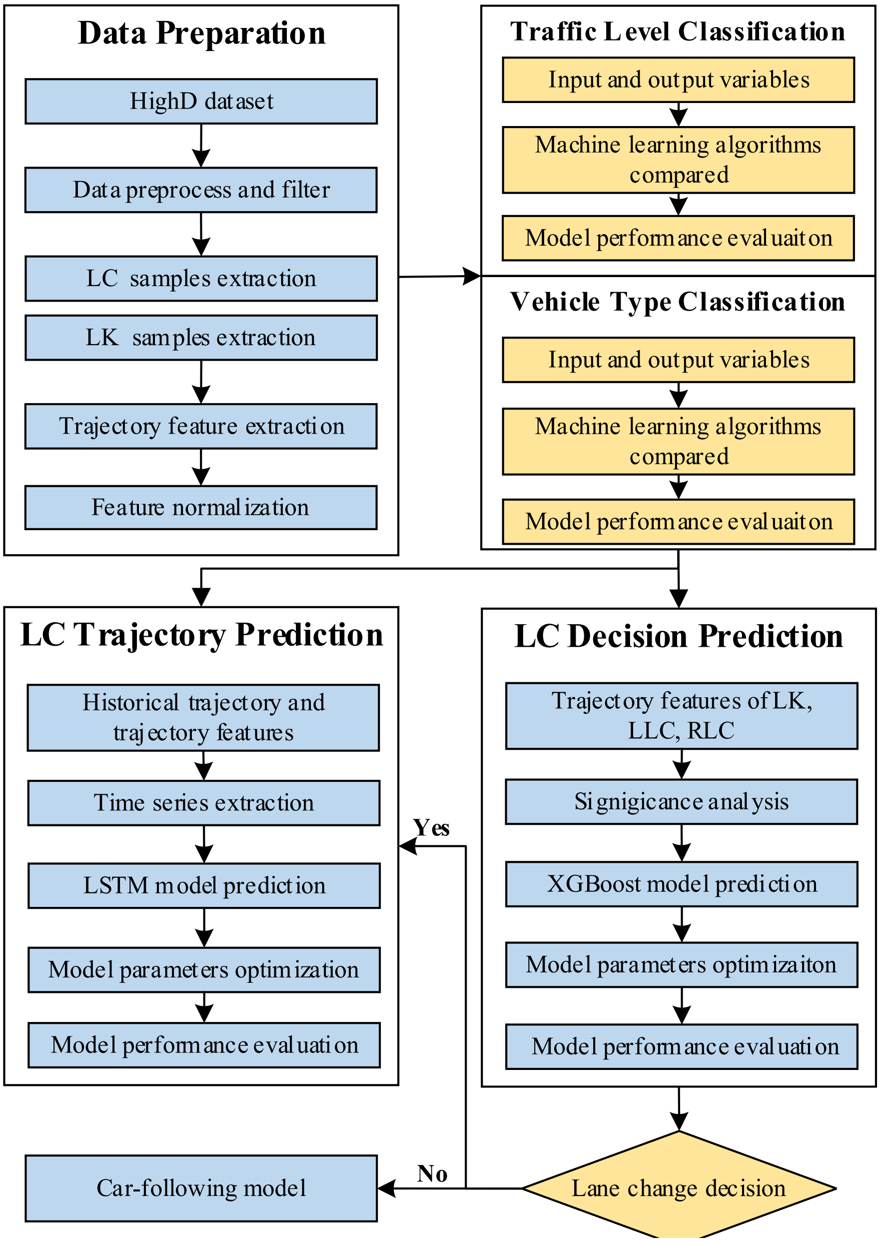


Fig. 1. The framework of the integrated lane change prediction model.

proposed models by using empirical lane change data. Finally, conclusions are made in Section 4.

2. Methodology

The framework of the integrated lane change prediction model incorporating traffic context is introduced in Fig. 1. It mainly contains four phases, namely, data preparation, traffic level and vehicle type classification, lane change decision prediction, and lane change trajectory prediction. The purpose of data preparation is to extract lane-changing and lane-keeping trajectories from the dataset after the data are preprocessed. The features are extracted from the trajectories as independent variables to establish the model, and they are normalized to eliminate differences in magnitude. The traffic level and vehicle type classification models are established with trajectory features, as they have an essential impact on lane change behaviors. Five machine learning algorithms are compared in terms of classification performance to select the most suitable one. Then, the integrated lane change prediction model is proposed, incorporating the classification results of the traffic context. The lane change prediction model contains lane change decision prediction and lane change trajectory prediction. An XGBoost-based method is applied to predict the lane change decision as left lane

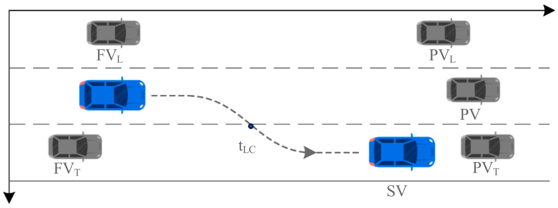


Fig. 2. The lane change process of the vehicle.

Table 1
Description of the variables during the lane change maneuver.

	Variables	Description
Single-vehicle variables	v	The longitudinal velocity of *SV* during the lane change (m/s)
	a	The longitudinal acceleration of *SV* during the lane change (m/s^2)
	v_{lat}	The lateral velocity of *SV* during the lane change (m/s)
	a_{lat}	The lateral acceleration of *SV* during the lane change (m/s^2)
	$\varphi(t)$	The degree between vehicle trajectory and road vertical line
Multivehicle variables	v_{r1}	The relative speeds of *SV* and *PV* (m/s)
	v_{r2}	The relative speeds of PV_T and *SV* (m/s)
	v_{r3}	The relative speeds of FV_T and *SV* (m/s)
	v_{r4}	The relative speeds of PV_L and *SV* (m/s)
	v_{r5}	The relative speeds of FV_L and *SV* (m/s)
	d_1	The space headway between *SV* and *PV* (m)
	d_2	The space headway between PV_T and *SV* (m)
	d_3	The space headway between FV_T and *SV* (m)
	d_4	The space headway between PV_L and *SV* (m)
	d_5	The space headway between FV_L and *SV* (m)

change (LLC), right lane change (RLC) and lane-keeping (LK) while incorporating traffic context. A significance analysis is conducted to confirm the necessity of considering the traffic context in the prediction model. The model parameters are optimized, and the performance of the mode is evaluated. Finally, the lane change trajectory model is proposed incorporating the results of the first three models with the LSTM algorithm. The historical trajectories of lane change are clustered into time series as the input of the LSTM algorithm to predict the trajectory in the future.

The trajectory data are the only data source applied in the integrated lane change prediction model, which can be collected by naturalistic driving experiments, video or driving simulators. The various collection methods of trajectory data make the lane change prediction system applicable in many traffic scenes. In this work, the trajectory data extracted from UAV video are used in the model.

2.1. Notations

A lane change process, which is shown in Fig. 2, usually lasts for several seconds. The description of the variables is detailed in Table 1. The notations are listed as follows.

SV, the subject vehicle, which changes lanes.
PV, the preceding vehicle of *SV* in the original lane.
PV_T, the preceding vehicle of *SV* in the target lane.
FV_T, the following vehicle of *SV* in the target lane.
PV_L, the preceding vehicle of *SV* in the lane on the other side of the original lane.
FV_L, the following vehicle of *SV* in the lane on the other side of the original lane.

The longitudinal velocities of vehicles *SV*, *PV*, PV_T, FV_T, PV_L, and FV_L are defined as v, v_1, v_2, v_3, v_4, and v_5, respectively. The relative speed between vehicles is defined as

$$\begin{cases} v_{r1} = v_1 - v \\ v_{r2} = v_2 - v \\ v_{r3} = v - v_3 \\ v_{r4} = v_4 - v \\ v_{r5} = v - v_5 \end{cases} \tag{1}$$

2.2. The traffic context label

To provide the dependent variables of the traffic context classification model, the traffic flow and vehicle type needs to be labeled first. The vehicle type can be labeled according to the vehicle images, while the traffic flow cannot be directly determined by the images.

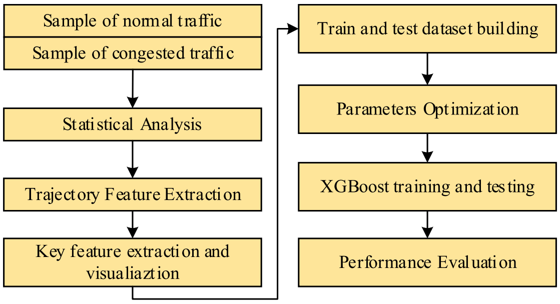


Fig. 3. XGBoost-based traffic level classification model process.

Traffic density is the primary determinate of the traffic flow level (Highway Capacity Manual, 2010). Therefore, we adopt this parameter to label the traffic level (Esfahani et al., 2018). The traffic density is calculated for each sample vehicle. As for a vehicle, the start and end time of the studied vehicle are t_1 and t_2, respectively. The cover range of the UAV video is the length of the road section to analyze the traffic state. The vehicle density is defined as the number of vehicles on the road section divided by the length of the road section at each time step t, which is given as

$$k(t) = \frac{n(t)}{L} \tag{2}$$

where $n(t)$ is the number of the vehicles on the road at time t, L is the length of the road section. The average density of the LC maneuver during t_1 and t_2 is calculated as

$$k = \sum_{t_1}^{t_2} k(t)/(t_2 - t_1) \tag{3}$$

The average density of each lane is the overall density of the road section divided by the number of lanes x, defined as

$$\bar{k} = k/x \tag{4}$$

Then we adopt k-means method to label the traffic flow level based on the traffic density. As the traffic density is different in clustered groups, the traffic levels are different.

2.3. The traffic context classification

2.3.1. The traffic level classification model

The characteristics of lane change trajectory vary with different traffic levels. Therefore, the traffic flow should be classified into different levels to facilitate lane change prediction, improving the accuracy of the lane change prediction model.

The traffic flow levels are the dependent variable of the traffic level classification model. The independent variables of the model are the features of single-vehicle and multi-vehicle variables. As the longitudinal speed $v(t)$ and acceleration $a(t)$ of the vehicle are most easily influenced by the traffic level, they should be considered in the model. Surrounding vehicles on both sides of the original lane should also be taken into account.

The traffic level classification model is proposed as

$$y_t = \{v(t), a(t), ST_R(t)\} \tag{5}$$

where $t = k \cdot \Delta t$, which indicates that the trajectory data is extracted every Δt time to classify the traffic level. Instead of the trajectory of surrounding vehicles, the model uses the relative trajectory between SV and five vehicles to predict the model given by $ST_R(t)$

$$ST_R(t) = \{v_{r1}(t), v_{r2}(t), v_{r3}(t), v_{r4}(t), v_{r5}(t), \quad d_1(t), d_2(t), d_3(t), d_4(t), d_5(t)\} \tag{6}$$

The dependent variable of the traffic level classification model is y_t, respectively defined as 0, 1, 2… for different traffic flow levels.

We adopt five machine learning algorithms (i.e., XGBoost, AdaBoost, naive Bayes, SVM, and decision tree) to train and test the proposed model. Taking the XGBoost algorithm as an example, the XGBoost-based traffic level classification model is shown in Fig. 3.

There are n sets of training data, and each set consists of multidimensional features extracted from $\{v(t), a(t), ST_R(t)\}$ denoting x_i, which belongs to class y_i. Therefore, x_i and y_i indicate the inputs and output of the classification model, respectively, which is given by

$$x_i \in \{v(t), a(t), ST_R(t)\}, \qquad y_i \in \{-1, \quad 1\}, i = 1, 2, ..., n \tag{7}$$

$$\theta = f(x_i, y_i) \qquad i = 1, 2, ..., n \tag{8}$$

where θ denotes the parameter set of the learning algorithm, including the learning rate, max depth, and others. These parameters are optimized based on the training data by differential evolution (DE) as function f to obtain optimal parameters θ (Sun et al., 2005). DE, which is also applied in other models, can improve the accuracy of the proposed model.

2.3.2. The vehicle type classification model

Previous studies have shown that car and truck drivers' driving preferences are significantly different (Ossen and Hoogendoorn, 2011), so the vehicle type should be considered when predicting the lane change process. The vehicle type classification model is proposed to recognize the vehicle type to facilitate the lane change prediction model. Similar to the traffic level classification model, the features extracted from the single-vehicle and multi-vehicle variables are used as independent variables to train the vehicle type recognition model.

In this work, the recognition model is given by

$$y_v = \{\varphi(t), v(t), a(t), ST_R(t)\} \tag{9}$$

The degree $\varphi(t)$ is one of the inputs in this model because the operation sensitivity is different for cars and trucks. The dependent variable of the vehicle type classification model is y_v, defined as -1 for cars and 1 for trucks. We also compare five machine learning algorithms to select the most suitable one to classify the vehicle type.

Notably, the highD dataset can provide vehicle type information. Therefore, we do not need to know the vehicle type from the proposed model. However, the proposed model is established for other cases where the vehicle type cannot be recognized directly. Then, the vehicle type classification model with only trajectory data can be applied to more datasets or scenarios.

2.4. The integrated lane change prediction model

The integrated lane change prediction model contains two steps: the lane change decision prediction and lane change trajectory prediction. The lane change decision is predicted as LLC, RLC or LK based on the trajectory data before the vehicle crosses the pavement markings. The lane change trajectory is predicted based on historic trajectories. The results of the integrated lane change prediction model help with the risk evaluation of lane change behaviors.

2.4.1. The lane change decision prediction model

The lane change process usually lasts for several seconds. Drivers perform lane change behaviors once the decision to change lanes is made. The lane change decision prediction model is proposed to predict whether the lane change decision will be conducted in the next few seconds. The surrounding vehicles in the original and target lanes are considered in the prediction model.

The independent variables of the lane change decision prediction model include the features of the trajectory data of SV and vehicles PV, PV_T, FV_T, PV_L, and FV_L. In particular, the lateral velocity $v_{lat}(t)$ and acceleration $a_{lat}(t)$ should be considered in the model, as a lane change is a lateral movement of a vehicle. In addition, the outputs of the traffic level classification model and vehicle type recognition model are also considered because the impact of traffic context on the lane change process should not be ignored. The lane change decision prediction model is defined as

$$y_d = \{v(t), a(t), \varphi(t), v_{lat}(t), a_{lat}(t), ST_R(t), y_t, y_v\} \tag{10}$$

The output y_d is the lane change decision, which is labeled as 1, 2, and 3 for LLC, RLC, and LK, respectively. The trajectory features were extracted to predict whether drivers would exhibit lane change behaviors a few seconds later.

The LLC, RLC and LK samples are extracted from the dataset and integrated to train the prediction model. In particular, the LLC and RLC maneuvers should be distinguished in the model, as there is a significant difference between LLC and RLC trajectories. This would help to improve the lane change trajectory prediction accuracy and improve the safety evaluation of driving behavior in the future.

2.4.2. The lane change trajectory prediction model

Once the lane change decision is predicted, the lane change trajectory is also predicted based on the subject vehicle's status and its surrounding vehicles in the target lane. A vehicle trajectory sequence is a kind of time-series data. The LSTM algorithm is one of the most applicable methods to predict the vehicle trajectory (Messaoud et al., 2019; Xie et al., 2019), as LSTM networks are well suited for classifying, processing, and making predictions based on time-series data. As a result, LSTM is also used to predict lane change trajectories in this study.

The function of the trajectory prediction model is defined as

$$(x(t+\Delta t), \quad y(t+\Delta t)) = \{v(t), a(t), \varphi(t), v_{lat}(t), a_{lat}(t), ST_R^{'}(t), y_v, y_t, y_d, X_t, Y_t\} \tag{11}$$

The inputs of the model are the trajectory data of SV and the relative trajectory $ST_R^{'}(t)$ between SV and PV, PV_T, and FV_T, where $ST_R^{'}(t)$ is defined as

$$ST_R^{'} = \{v_{r1}(t), v_{r2}(t), v_{r3}(t), d_1(t), d_2(t), d_3(t)\} \tag{12}$$

The results of traffic context and lane change decision y_v, y_t, y_d are also taken as inputs of the model. The historical trajectory of SV, including the lateral position X_t and the longitudinal position Y_t, is also adopted to train and predict the position Δt later and is given by

$$X_t = \{x(t), x(t-\Delta t), x(t-2\Delta t), x(t-3\Delta t)...x(t-k\Delta t)\} \tag{13}$$

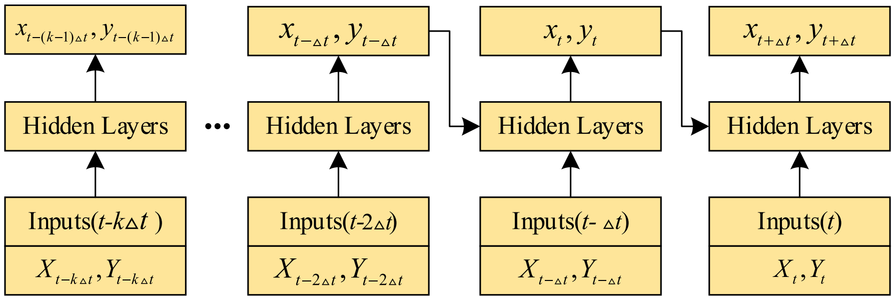


Fig. 4. Flow chart of the prediction model based on the LSTM algorithm.

$$Y_t = \{y(t), y(t-\Delta t), y(t-2\Delta t), y(t-3\Delta t)...y(t-k\Delta t)\} \tag{14}$$

The outputs of the model are the location of the *SV*, i.e., the lateral position $x(t+\Delta t)$ and longitudinal position $y(t+\Delta t)$. All trajectories are collected k preceding time steps before time $t+\Delta t$ to predict the position of the *SV* at time $t+\Delta t$. The LSTM algorithm is adopted to predict the lane change trajectory. The flow chart of the prediction is shown in Fig. 4.

In particular, the results of traffic context classification should be considered when predicting a trajectory, as they have a great impact on driving behavior. In addition, the lane change decision y_d should be added, as there is a great difference between LLC and RCL. The results with and without the consideration of traffic context and lane change decisions are compared in this study.

To evaluate the performance of the traffic context classification model and lane change decision prediction model, the recall rate, precision rate, accuracy rate and F1-score are adopted in this study. The mean squared error (MSE) is also applied to evaluate the results of the model. The MSE is defined as

$$MSE = \frac{1}{N}\sum_{n=1}^{N}(y_n - y_n^{'})^2 \tag{15}$$

where y_n and $y_n^{'}$ are the real and predicted values of the test samples, respectively, and N is the number of testing samples in the model.

2.5. Key feature extraction and visualization

2.5.1. Key feature extraction

To establish the proposed model, the features of trajectory variables need to be extracted as independent variables. This study compares two types of feature extraction methods in the lane change prediction model, i.e., time-domain features and frequency-domain features. They are detailed below.

(1) Time-domain features. The statistical method (SM) can capture time-domain features, i.e., the maximum, minimum, mean and variance of variables, which can extract most of the distribution information of the data. Additionally, time-domain features were commonly used in past research and were proven to be efficient (Chen et al., 2017).

(2) Frequency-domain features. Discrete wavelet transform (DWT) and discrete Fourier transform (DFT) are applied to extract the frequency-domain features. These two methods convert the time series of model input variables into signal amplitudes in the frequency domain. The details of these two methods can be found in Olkkonen (2011) and Zhang et al. (2010).

Three methods, SM, DWT and DFT, are compared in this study to select the most suitable method to facilitate the model.

2.5.2. Key features visualization

Time-domain features and frequency-domain features are adopted in the four models to facilitate lane change prediction. To better compare the difference in lane change maneuvers under different contexts, the t-SNE algorithm is adopted to visualize the features (Laurens and Hinton, 2008). Based on the t-SNE algorithm, high-dimensional features are projected onto two-dimensional space, which can easily distinguish the differences between contexts.

2.6. Machine learning algorithms

In machine learning applications using integrated lane change prediction models, it is generally not easy to select an appropriate model. Therefore, five types of algorithms are evaluated and compared to determine the most suitable one for the lane change prediction task. The algorithms used to predict the lane change decision include XGBoost, AdaBoost, naive Bayes (NB), support vector machine (SVM) and decision tree (DT). The details are shown below. LSTM is used to predict the trajectory, which is described in Section 3.4.

2.6.1. XGBoost algorithm

The XGBoost algorithm proposed by Chen and Guestrin (2016) is a novel implementation method of the gradient boosting

Table 2
Duration of lane change implementation.

Duration of lane change implementation	Number of vehicles	Proportion
≤2 s	4	0.0107
2–3 s	17	0.0456
3–4 s	177	0.4745
4–5 s	118	0.3164
5–6 s	41	0.1099
6–8 s	16	0.0429
Total	373	1.0000

framework. It includes an efficient linear model solver and regression tree algorithm, which can be applied in the regression, classification and ranking of objective functions. This algorithm combines the prediction results of "weak" learners to develop a "strong" learner based on an additive training strategy. XGBoost combines regularization terms with a cost function to control the model complexity. It also sorts the features before training data and executes parallel calculations to reduce the computation.

2.6.2. AdaBoost algorithm

AdaBoost and XGBoost are both integrated algorithms that combine "weak" learners to train the dataset. However, AdaBoost is an adaptive boosting algorithm (Schapire, 2013). Incorrectly classified samples will be weighted to create new samples to train the basic classifier in the next step. A new weak classifier will be added during each training until the maximum number of iterations or the expected error rate is reached.

2.6.3. Naive Bayes algorithm

The naive Bayesian (NB) classifier is a relatively simple probability classifier based on Bayesian theory that calculates the posterior probability by using the prior probability of samples (Leung, 2007). The NB classifier assumes that each feature in the model has strong independence and does not take the correlation between features into account. The Gaussian NB algorithm is a type of naive Bayes that follows a Gaussian distribution, which is suitable for continuous variables. In this study, the Gaussian NB from the scikit-learn toolkit is adopted to train the samples.

2.6.4. SVM algorithm

The SVM is a machine learning algorithm proposed by Cortes and Vapnik (1995). The maximum margin boundary between positive data and negative data is discovered in the SVM, which is also adopted to classify different classes of samples. The training vector, namely, the data closest to the classification boundary, is extracted to obtain the margin boundary. There are several kernel functions used in the SVM, such as linear, Gaussian, and cubic. In this study, the linear function is applied in the SVM model.

2.6.5. Decision tree algorithm

Decision tree is a nonparametric method that does not need to presume the given dataset's distribution (Safavian and Landgrebe, 1991). A decision tree is built based on the training data, selecting the appropriate test attribute for the decision node and defining each leaf's labeling. Starting with the root of the decision tree, the node's attributes move down the tree branch until they meet certain conditions.

3. Results and discussions

3.1. Data description

The data employed in this study are provided by the highD dataset. This dataset contains postprocessed trajectories of vehicles on six different German highways around Cologne during 2017 and 2018. Data is collected by a drone, and the collection range is 420 m. Trajectories are automatically extracted by computer vision algorithms with 25 Hz. Compared with NGSIM, which has been widely used in transportation research, the data from highD have a wider variety with more recording sites and longer recording times. HighD offers more trajectories of truck drivers and a much broader range of mean speed, facilitating the exploration of the impact of traffic context on lane change prediction (Krajewski et al., 2018).

To maintain a similar driving environment (i.e., road and light conditions), we select the trajectory data of 373 vehicles (316 cars and 57 trucks) with lane change maneuvers at one location from 9:30 to 11:00 am to study in this work. The segment where the data are collected contains six straight lanes without intersection or ramp, and the speed limit is 120 km/h. The preceding and following vehicles of subject vehicles are also extracted within the collection range of 420 m. The extraction information of the trajectory includes the frame ID, vehicle ID, position, width and length of the vehicle, velocity and acceleration, lane ID, and surrounding vehicles' IDs. By matching the frame, the preceding and following vehicles of the subject vehicle can be determined. However, preceding or following vehicles may not exist in some cases, and the relative velocity and gap between the SV and surrounding vehicles are defined as a random max value. When assigned the max value, missing surrounding vehicles will not influence the operation of the SV.

The lane change process consists of three stages: preparation, implementation, and recovery. Lane change implementation is defined as the continuous lateral movement of the subject vehicle to the target lane (Xie et al., 2019). The starting point of a lane

Table 3
The statistics of congested and normal traffic.

Variables	Normal	Congested
Density (pc/mile/ln)	28.84	41.46
LOS	C	E

Table 4
LOS criteria for basic freeway segments.

LOS	Density (pc/mile/ln)
A	11
B	18
C	26
D	35
E	45

Table 5
Sample distribution under different traffic contexts.

Context	**LLC**	**RLC**	**LK**	**All**
Car-normal	59	69	80	208
Car-congested	68	120	80	268
Truck-normal	27	30	80	137
All	154	219	240	613

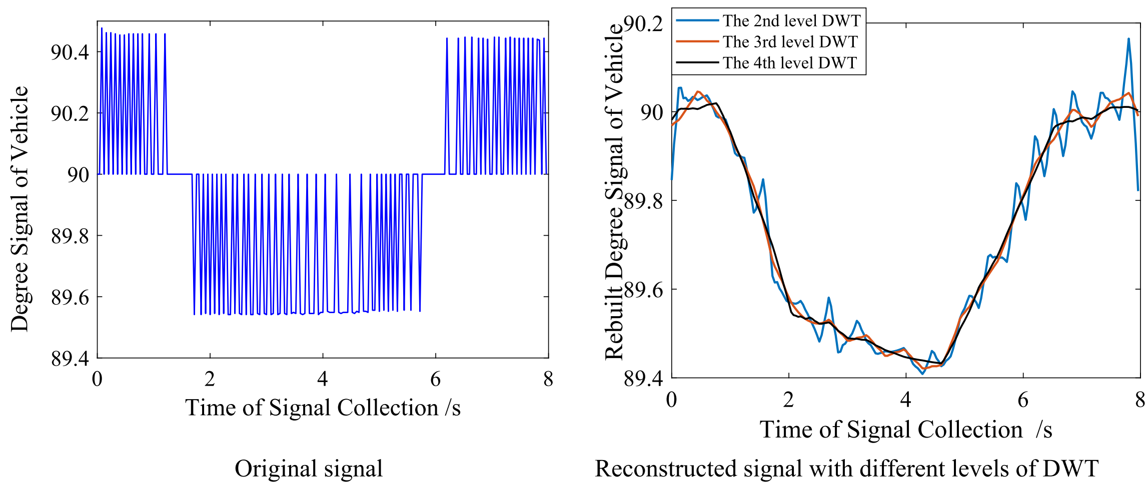


Fig. 5. The reconstructed signal using the DWT method.

change implementation is the point at which a vehicle starts to continuously move toward the target lane and the end point is the point at which the vehicle starts to move in the opposite direction. The lane change preparation and recovery processes occur before and after lane change implementation, respectively. Vehicles in these two stages may not remain in the same direction as in the lane change implementation.

Table 2 summarizes the statistical results of the lane change implementation of selected vehicles. The lane change implementation duration is 5–8 s for trucks and no more than 5 s for cars. An implementation process time of 2–6 s accounts for nearly 95% of the vehicles. While the lane change preparation and recovery processes take approximately 2 s each, a time window of 10 s covers the whole lane change process (Bejani and Ghatee, 2018). Therefore, the duration of lane change samples is defined as 10 s for cars and trucks in this study. Cases whose duration is less than 6 s or more than 10 s are removed from the dataset to avoid unnecessary interference (Wang et al., 2021). It is sufficient to calibrate and validate the integrated lane change prediction model.

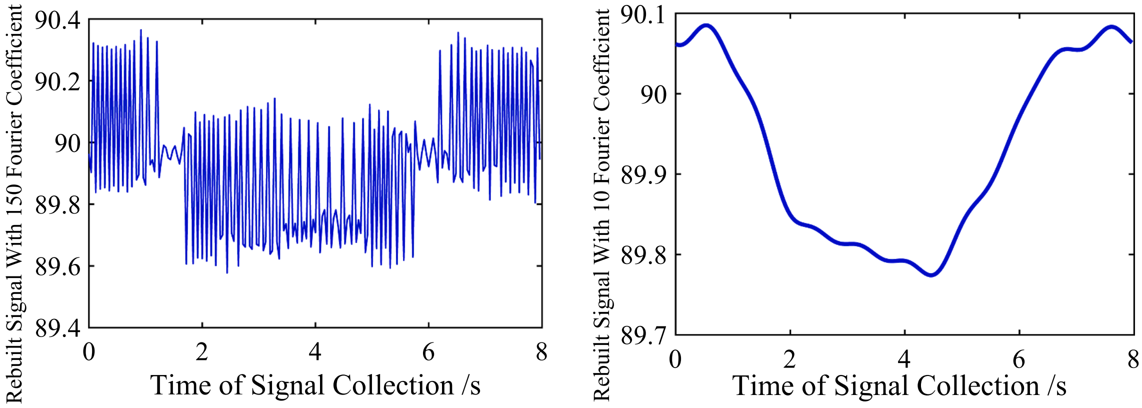
In this study, 373 LC samples and 240 LK samples are extracted from the highD dataset to train and test the prediction model. In order to train the model, the vehicle type and traffic flow level need to be labeled first. The vehicle type of samples is labeled according to the dataset. The traffic flow level is labeled with k-means method. The silhouette value of the k-means reaches highest when the traffic flow is classified into two levels. Then the traffic flow is classified into two groups in the paper. The descriptive statistics of two

Table 6
The MSE comparison with different decomposition layers.

DWT	1-layer	2-layer	3-layer	4-layer	5-layer
Db1	**0.033**	0.039	0.041	0.041	0.045
Db2	0.034	0.041	0.041	0.041	0.043
Db3	0.034	0.040	0.041	0.041	0.043
Db4	0.035	0.041	0.041	0.041	0.042

Table 7
The MSE comparison with 12 different wavelet functions.

DWT	1-layer	DWT	1-layer	DWT	1-layer
Db1	**0.033**	**Sym1**	**0.033**	**Coif1**	0.036
Db2	0.034	**Sym2**	0.034	**Coif2**	0.036
Db3	0.034	**Sym3**	0.034	**Coif3**	0.037
Db4	0.035	**Sym4**	0.036	**Coif4**	0.037



Reconstructed signal with 150 Fourier coefficients Reconstructed signal with 10 Fourier coefficients

Fig. 6. Reconstructed signals with the DFT method.

groups is depicted in Table 3. The LOS of two traffic levels is determined by the LOS criteria for basic freeway segments in the Highway Capacity Manual (2010) in Table 4, which are respectively LOS C and LOS E. Then the traffic level is labeled as normal (LOS is C) and congested (LOS is E). Based on this distinction, the 208 samples under the normal traffic level and 268 samples under the congested traffic level are extracted from the dataset. It should be noted that since the speed difference between congested and normal traffic is not obvious, the proposed model is very sensitive to speed for classifying traffic levels. More specifically, if the proposed model can achieve good performance on the non-sensitive data, then its performance would be better on sensitive data.

The sample distribution under different contexts is shown in Table 5. Because of the limited samples for the truck-congested situation, only the car-normal, car-congested and truck-normal samples are extracted. The comparison between the three types of samples reveals the impact of traffic context on lane change decisions and trajectories.

3.2. Key feature extraction and visualization

3.2.1. Features extraction comparison

The DWT and DFT methods are compared in frequency-domain feature extraction from trajectory variables to facilitate the classification and prediction models. In the DWT method, 12 wavelet basis functions and 5 decomposition layers are applied to decompose the signal, and then the signals are reconstructed to evaluate the performance of the method. The reconstructed signal based on different decomposition layers is shown in Fig. 5. With the increase in the number of decomposition layers, the signal will lose more details and become smoother. This means that the features extracted from DWT higher layers perform worse in traffic context classification and lane change prediction models.

To evaluate the reconstructed signal based on different decomposition layers and wavelet functions, the MSE is used to compare the difference in the original signal and reconstructed signal, as shown in Tables 6 and 7. The MSE decreases with higher decomposition layers, while the 1-layer coefficient achieves the lowest MSE with the Db1 function. In addition, coefficients extracted by 12 functions are adopted to reconstruct the signals and indicate that the Db1 and Sym1 functions outperform other functions in Table 7.

The DFT method is adopted to reconstruct the signal based on the Fourier coefficient, as shown in Fig. 6. The MSE between the

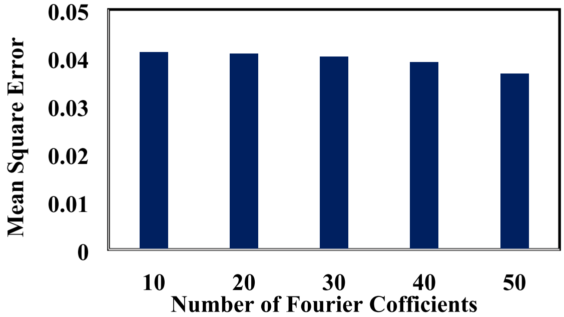


Fig. 7. The MSE comparison of different Fourier coefficients.

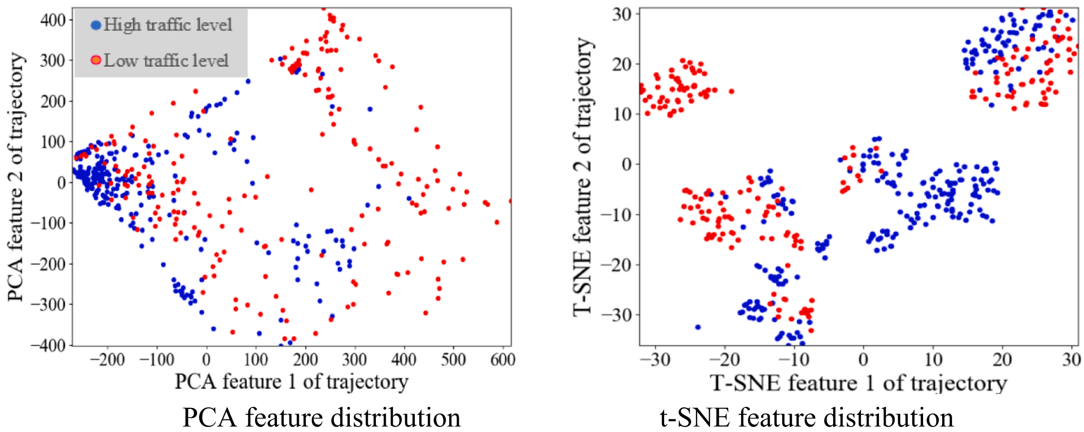


Fig. 8. Features distribution of PCA and t-SNE methods in the traffic level classification model.

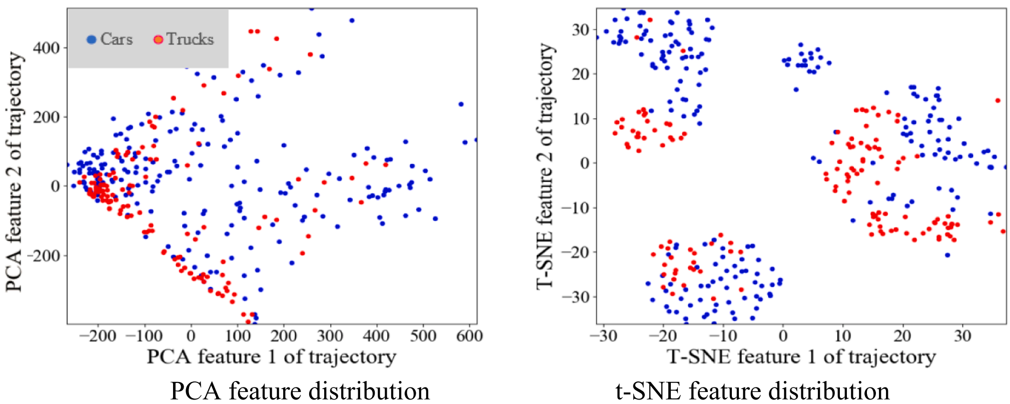


Fig. 9. Features distribution of PCA and t-SNE methods in the vehicle type classification model.

reconstructed signal and original signal is shown in Fig. 7. It can be seen that the signal is better reconstructed with more Fourier coefficients. The MSE of the reconstructed signal with the first 50 Fourier coefficients is 0.037, which is higher than that of the DWT coefficient. The performance of the DWT method is similar to the DFT coefficients, while the features of DWT are much less than those of DFT, reducing the computational complexity. Therefore, the DWT with a 1-layer Db1 function is applied to extract frequency-domain features in this study. The time- and frequency-domain features are compared in prediction performance later to select the most suitable features.

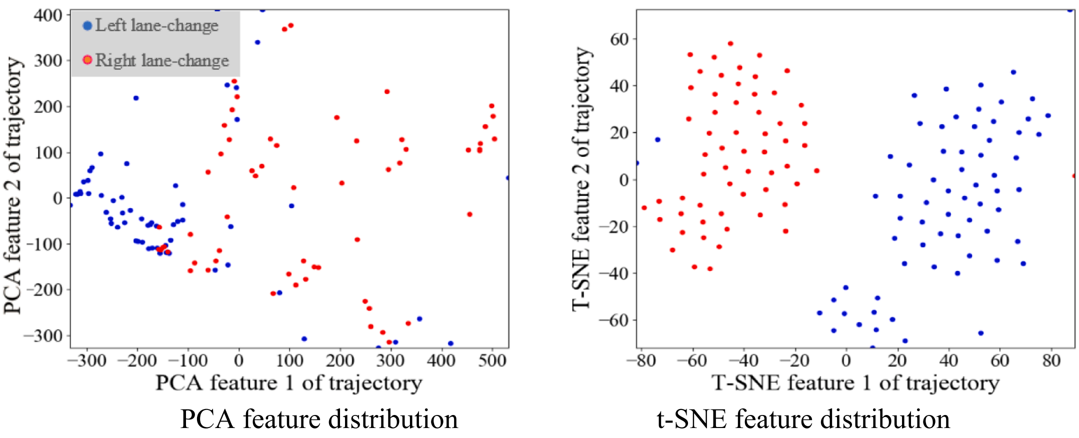


Fig. 10. Features distribution of PCA and t-SNE methods in the lane change decision prediction.

Table 8
Significance analysis between trajectory features for normal and congested traffic.

Variables	Congested		Normal		F-value
	Mean	Std	Mean	Std	
v	25.235	26.975	31.173	8.159	**0.000****
a	0.122	0.186	−0.002	0.138	**0.000****
v_{lat}	0.019	0.262	0.304	0.119	**0.000****
a_{lat}	0.015	0.040	0.108	0.018	**0.000****
v_{r1}	−0.759	5.930	−0.510	17.621	**0.000****
v_{r2}	−0.349	17.787	1.087	25.272	**0.000****
v_r	1.081	38.711	5.690	121.966	**0.000****
d_1	43.225	1728.473	82.877	6031.493	**0.000****
d_2	46.256	3748.281	104.806	9307.097	**0.000****
d_3	60.032	8681.355	103.975	13794.015	**0.000****

** Significance correlation at the 0.01 level (bilateral).

3.2.2. *Key features visualization*

To analyze the feature distribution difference under different traffic contexts, the advanced dimension reduction method t-SNE is adopted to achieve the task. The t-SNE projects the high-dimensional features onto two-dimensional features, which makes it much easier to visualize the features representing different traffic levels, vehicle types and driving maneuvers. To indicate the advantages of t-SNE, another commonly used algorithm, PCA, is adopted for comparison with t-SNE.

The dimension reduction results with trajectory features of the traffic context classification model and lane change prediction model are shown in Figs. 8, 9 and 10. As we can see, the features of high and low traffic levels are nonlinearly separable in Fig. 8. Similar to the traffic level, the features of the vehicle type classification model have large complexity in Fig. 9, which requires a powerful classification algorithm to classify different vehicle types. The trajectory features of LLC and RLC maneuvers are linearly separable in Fig. 10. Compared with PCA, the feature distribution with the t-SNE method is significantly more separable, implying the better performance of t-SNE. The t-SNE algorithm can better separate different classes of data, which further explains the difference in behaviors between right lane change and left lane change. This indicates the necessity of integrating the context into the lane change prediction model.

3.3. *Evaluation of the traffic context classification model*

3.3.1. *The traffic level classification model evaluation*

To explain the significance of traffic level classification, the difference in lane change trajectory variables under different traffic levels is explored by using an independent samples F-test, as shown in Table 8. There is a significant difference between the trajectory variables of congested and normal traffic. The longitudinal and lateral speeds of lane change maneuvers under congested traffic are 25.235 m/s and 0.019 m/s, respectively, which are much slower than those under normal traffic. Additionally, the space headway and relative speed between SV and PV, PV_T, and FV_T are much smaller under congested traffic, indicating that drivers perform lane change maneuvers more aggressively.

The preferences of drivers are different under the two traffic levels according to Table 8, revealing the impact of traffic level on driving behavior. Therefore, it is necessary to classify the traffic level as one of the inputs in the lane change prediction model.

Table 9
Traffic level classification results of frequency-domain features.

Algorithms	Class name	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
XGBoost	Congested	91.24	93.63	**91.35**	**91.36**	**0.086**
	Normal	91.54	88.46			
AdaBoost	Congested	91.51	92.88	91.14	91.16	0.088
	Normal	90.68	88.94			
NB	Congested	88.97	84.64	85.51	85.47	0.145
	Normal	81.44	86.53			
SVM (linear)	Congested	90.29	90.63	89.26	89.26	0.107
	Normal	87.92	87.50			
DT	Congested	88.89	92.88	89.43	89.47	0.105
	Normal	90.30	85.09			

Table 10
Comparison of classification results.

Algorithms	Frequency-based features		Time-based features	
	Accuracy (%)	MSE	Accuracy (%)	MSE
XGBoost	91.36	0.086	90.32	0.096
AdaBoost	91.16	0.088	91.78	0.082
NB	85.47	0.145	71.15	0.288
SVM (linear)	89.26	0.107	81.89	0.181
DT	89.47	0.105	86.52	0.134
Average	**89.34**	**0.106**	**84.33**	**0.156**

Table 11
Significance analysis between trajectory variables for car and truck drivers.

Variables	Car		Truck		F-value
	Mean	Std	Mean	Std	
v	31.702	8.563	25.830	6.832	**0.000****
a	0.048	0.100	0.012	0.044	**0.000****
v_{lat}	0.490	0.105	0.491	0.101	0.967
a_{lat}	0.180	0.015	0.178	0.015	0.432
v_{r1}	−0.606	25.964	2.202	22.304	**0.000****
v_{r2}	0.627	17.861	2.937	17.483	**0.000****
v_r	4.141	28.374	0.245	33.147	**0.000****
d_1	101.720	8669.554	88.231	6294.753	**0.000****
d_2	94.570	6346.226	84.236	6614.721	**0.000****
d_3	94.648	12188.109	87.021	9132.504	**0.000****

** Significance correlation at the 0.01 level (bilateral).

By 10-fold cross-validation, the classification results of training frequency-domain features based on 5 machine learning algorithms are shown in Table 9. XGBoost is most suitable for classification at the normal and congested traffic levels, achieving an accuracy of 91.36%. The accuracy of the AdaBoost algorithm reaches 91.16%, which is higher than the remaining methods. XGBoost and AdaBoost are both integrated machine learning algorithms, so they outperform other methods.

To compare the performance of time-domain features and frequency-domain features, the classification results based on time-domain features are shown in Table 10. The classification accuracy based on frequency-domain features is higher than that based on time-domain features, except for AdaBoost. The average accuracy based on frequency-domain features for the 5 algorithms is 89.34%, while the average accuracy based on time-domain features is 84.33%. These results indicate that the frequency-domain features perform better than time-domain features. Additionally, the MSE for frequency- and time-domain features are 0.106 and 0.156, respectively, confirming the outperformance of frequency-domain features.

3.3.2. Vehicle type classification model evaluation

The dataset used for training and testing the vehicle type classification model consists of 137 truck samples and 208 car samples under the normal traffic level. The significance analysis based on the F-test is conducted on the samples shown in Table 11. Due to the operation sensitivity and vehicle performance, truck drivers' longitudinal speed v and acceleration a are smaller than those of car drivers. Truck drivers tend to perform lane change with larger headway distances to preceding vehicles. The driving trajectories of truck drivers are greatly different from those of car drivers. This result also proves the necessity of considering vehicle type when predicting lane change maneuvers.

The training and testing of the vehicle type classification model are conducted by applying the 5 algorithms and using DWT features

Table 12
The vehicle type classification results of frequency-domain features.

Algorithms	Class name	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
XGBoost	Car	97.56	96.61	**96.51**	**96.51**	**0.035**
	Truck	94.96	96.35			
AdaBoost	Car	95.63	95.16	94.48	94.47	0.055
	Truck	92.75	93.43			
NB	Car	91.25	70.53	78.39	78.19	0.218
	Truck	66.84	89.78			
SVM (linear)	Car	95.16	95.16	94.18	94.18	0.058
	Truck	92.70	92.70			
DT	Car	94.73	95.65	94.17	94.18	0.058
	Truck	93.33	91.97			

Table 13
Significance analysis between trajectory variables for LLC and RLC.

Variables	RLC		LLC		F-value
	Mean	SE	Mean	SE	
v	32.562	5.912	30.692	9.793	**0.000****
a	−0.027	0.052	0.135	0.143	**0.000****
v_{lat}	0.514	0.099	0.462	0.111	**0.000****
a_{lat}	0.178	0.013	0.182	0.017	0.068
v_{r1}	2.801	13.914	−4.612	10.430	**0.000****
v_{r2}	−1.117	16.745	2.677	11.397	**0.000****
v_r	6.443	8.247	1.436	38.492	**0.000****
d_1	146.593	11216.042	48.975	528.090	**0.000****
d_2	114.061	6201.082	71.659	5546.351	**0.000****
d_3	59.073	7193.516	136.464	14824.403	**0.000****

** Significance correlation at the 0.01 level (bilateral).

Table 14
Optimal parameters of XGBoost using the DE algorithm.

Parameters	Range	Optimization results
Num_leaves	(10 50)	22.644
Min_data_in_leaf	(10 50)	23.607
Max_depth	(5 20)	15.334
Bagging_fraction	(0.5 1)	0.8545
Feature_fraction	(0.5 1)	0.5018
Lambda_l1	(0 10)	3.21108
Lambda_l2	(0 10)	3.9605
Min_gain_to_split	(0.001 0.1)	0.06662
Min_child_weight	(0.001 100)	14.7948

extracted from trajectory variables. The vehicle type classification result is shown in Table 12. The XGBoost algorithm performs best on vehicle recognition with the highest accuracy of 96.51%. The accuracies of AdaBoost, SVM with a linear kernel function, and DT are similar to one another, while the accuracy of the NB algorithm is lowest.

3.4. Evaluation of the integrated lane change prediction model

3.4.1. The lane change decision prediction model evaluation

The lane change decision prediction is integrated with the results of traffic level and vehicle type classification models. To evaluate the lane change decision prediction model, all 613 samples in Table 5 are adopted in the model. The 5-fold cross-validation is adopted to train and test the model. The frequency-domain features extracted from the variables are applied to facilitate lane change decisions. In the lane change decision prediction model, each sample's trajectory before crossing pavement markings is extracted every 0.04 s (corresponding to the data extraction frequency 25 Hz) to predict whether the driver chooses a lane change a few seconds later.

Difference analysis is applied to explore the driving preference for LLC and RLC maneuvers, as shown in Table 13. It can be seen that drivers perform LLC maneuvers when the speed of PV_T in the target lane is faster than that of the PV in the original lane. Additionally, the headway in the target lane is higher than that in the original lane, providing a safer environment for vehicles to accelerate. This is because the passing lane is on the left, so drivers change lanes to the left to obtain higher speed. Compared with RLC, LLC involve more aggressive acceleration and place the *SV* much closer to preceding vehicles. However, the headway between the *SV* and *FV* for RLC is longer. This may result from the limited view of drivers as they sit at the left of the car.

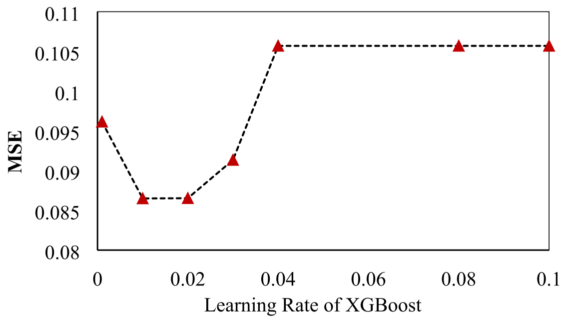


Fig. 11. Evaluation of prediction results by different learning rates.

Table 15
Lane change decision prediction performance based on the XGBoost algorithm.

Time	Class name	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
0.5 s	LK	97.91	97.50	**98.20**	**98.20**	**0.043**
	RLC	99.07	98.61			
	LLC	97.44	98.70			
1.0 s	LK	96.25	96.25	97.10	97.10	0.087
	RLC	97.05	97.05			
	LLC	98.30	98.30			
1.5 s	LK	96.20	95.00	96.61	96.61	0.106
	RLC	95.65	97.05			
	LLC	98.30	98.30			
2.0 s	LK	91.66	96.25	95.19	96.17	0.120
	RLC	98.46	94.11			
	LLC	96.55	94.91			

Table 16
MSE of lane change decision prediction when one input is removed.

Removed variables	MSE	Removed variables	MSE
v_{r1}	**0.116**	d_1	**0.097**
v_{r2}	**0.116**	d_2	0.077
v_{r3}	0.092	d_3	0.092

This model utilizes the XGBoost algorithm to predict lane change decisions. The differential evolution algorithm (DE) method is applied to optimize the parameters of XGBoost, as shown in Table 14. These parameters are directly related to the structure of the algorithm, influencing the training complexity and test accuracy. DE is a heuristic random search algorithm based on group difference. It can search for the optimal parameters of XGBoost by group initialization, variation, crossing and choice. For details on DE, refer to Sun et al. (2005).

The learning rate of XGBoost is an important parameter for the prediction model, which needs to be determined for the training dataset. Larger or smaller learning rates both influence the convergence process of the model. A smaller learning rate may increase the calculation process of the model, resulting in a longer training time. When the learning rate is set too large, the gradient may oscillate back and forth near the minimum value and even fail to converge. Therefore, an appropriate learning rate would facilitate the model convergence and achieve the optimal accuracy. Seven learning rates (0.001, 0.01, 0.02, 0.03, 0.04, 0.08, and 0.1) are adopted in the training model to evaluate the MSE of the prediction result in Fig. 11. The minimum MSE is obtained when the learning rate is 0.01. Therefore, the learning rate for the XGBoost-based lane change decision prediction model is set as 0.01.

Only 5 s can be used to collect the trajectory before vehicles cross lanes (each lane change sample is 10 s). The trajectory data are extracted 0.5 s, 1.0 s, 1.5 s, and 2.0 s before the vehicle crosses the pavement markings to predict whether the vehicle would perform the lane change maneuver a few seconds later. The lane change decision prediction based on XGBoost is shown in Table 15. The accuracy of the lane change decision is 95.19% approximately 2 s before vehicles cross lanes. It can be seen that the prediction accuracy is higher when it approaches the lane change time. The average prediction precision for RLC, LLC and LK are 97.06%, 97.86% and 95.39%, respectively. The precision rate of LLC is higher than that of RLC and LK, indicating that the system is more sensitive to LLC.

It is known that drivers mainly rely on the conditions of surrounding vehicles to determine whether to change lanes. However, which of the surrounding vehicles plays the most essential role in lane change decisions is still unclear. Sensitivity analysis is conducted to explore the efficiency and rationality of variable selection from surrounding vehicles in the lane change decision prediction model.

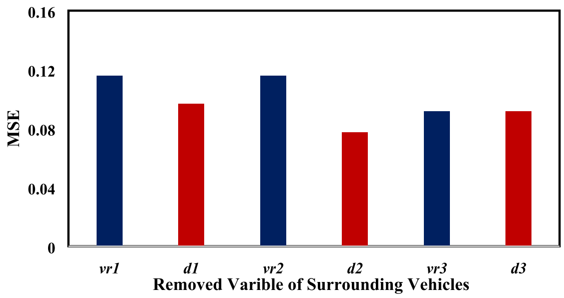


Fig. 12. The MSE of lane change decision prediction when one input is removed.

Table 17
The prediction result comparison with and without traffic context.

Comparison	Class name	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
Traffic context	LK	97.91	97.50	**98.20**	**98.20**	**0.043**
	RLC	99.07	98.61			
	LLC	97.44	98.70			
Without considering traffic context	LK	97.41	94.17	97.00	97.02	0.065
	RLC	96.87	100.00			
	LLC	96.73	96.73			

The process of sensitivity analysis is presented as follows. The trajectory data of subject vehicle *SV* and surrounding vehicles *PV*, PV_T, and FV_T are applied in the model. To carry out the sensitivity analysis, each trajectory variable (i.e., model input) is removed to train the lane change prediction model based on the same samples (0.5 s before crossing lane). The MSE of XGBoost-based lane change decision prediction with fewer inputs is presented in Table 16 and Fig. 12. It can be seen that the MSE increases greatly compared with the MSE resulting from the original inputs. In particular, the MSE increases more than one time when v_{r1} or v_{r2} is removed. This indicates that the relative speed between the *SV*, *PV* and PV_T are important to predict the lane change decision. The average MSE removed variables of the *PV*, PV_T, and FV_T are 0.106, 0.097 and 0.092, respectively, indicating that the *PV* is the most important vehicle for lane change decisions. In addition, removing input variables regarding speed causes more incremental MSE compared with those regarding space headway.

The vehicle type and traffic level both impact vehicle acceleration and mobility. To show the influence of the traffic context, two sample sets are built with and without consideration of traffic context. Both samples consist of 613 samples, while one sample is set further to consider the traffic level and vehicle type. The lane change decision prediction comparison is shown in Table 17. It can be seen that the prediction of lane change decision decreases the accuracy if the context is not considered. Compared with other research (Wang et al., 2021; Kumar et al., 2013), the proposed lane change decision model performs better.

3.4.2. The lane change trajectory prediction model evaluation

All 373 lane change samples are applied in the lane change trajectory prediction model. The “leave-one-method” is adopted to predict the lane change trajectory of each vehicle by LSTM. Each sample in the trajectory prediction model consists of 10 s trajectory data with 25 Hz, which means that the trajectory is composed of 250 data records. Accordingly, 93,000 data records are obtained for training and 250 records for testing. The max–min method is used to normalize the data before training and testing the model.

The network structure of the trajectory prediction model is “14–50-100-Dense-2”, which includes 3 LSTM layers with 0.2 dropout. The two hidden layers have 50 and 100 neurons, respectively. Dense is the fully connected layer, transforming the output vector into the two-dimensional vector corresponding to the position variables. The input layer has 14 neurons corresponding to the input variables in function (10); the output layer has 2 neurons corresponding to the lateral and longitudinal positions of the subject vehicle, respectively.

To train the LSTM model, the parameters of the model must be optimized first.

(1) The Adam optimizer is an adaptive moment estimation method (Zhang, 2018) that can optimize the parameters to obtain optimal results. The learning rate is set as 0.005 after optimization by Adam.
(2) The batch size affects the convergence rate of the model as well as the GPU memory usage. The mini batch-size is set as 32 in the study.
(3) The time step is also an important parameter in the LSTM-based trajectory prediction model. This model adopts the trajectory data within the last time steps to consider their effect on the next samples. An appropriate time step could improve the prediction accuracy. Six kinds of time steps (5, 10, 15, 20, 30 and 40) are used to predict the trajectory of vehicle 4 in the model. MSEs of the prediction results and true results are shown in Fig. 13. It can be seen that the minimum MSE is obtained when the time step is set as 30.

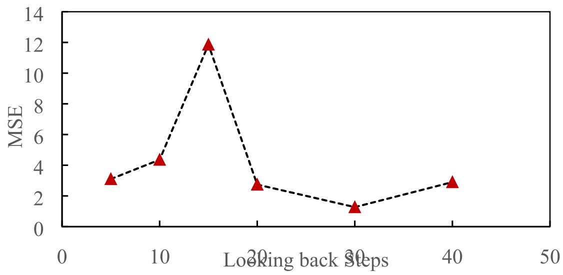


Fig. 13. The MSE of trajectory prediction with different time steps.

Table 18
The MSE of the trajectory predicted by the LSTM algorithm.

ID	MSE	ID	MSE
1	7.59	10	7.6
2	6.27	11	11.17
3	4.18	12	9.02
4	3.5	13	7.78
5	3.87	14	11.02
6	7.14	15	6.38
7	3.2	16	5.65
8	5.96	17	7.12
9	7.09	18	4.58

To evaluate the performance of the LSTM-based lane change trajectory prediction model, 373 lane change samples are used to train and test the lane change trajectory prediction model. For the consideration of traffic context, six types of sample sets are built under car-normal, car-congested and truck-normal situations for LLC and RLC. The MSEs of the predicted trajectory for 18 randomly selected vehicles are shown in Table 18. The average MSE of all predicted trajectories is 6.62. The average MSEs for LLC and RLC are 6.60 and 6.62, respectively, indicating that the LLC and RLC trajectories can be predicted with similar similarity. The predicted trajectories of vehicles 1, 3, 4 and 5 are shown in Fig. 14. Such small MSEs indicate that the LSTM-based lane change trajectory prediction model can capture the whole lane change process well.

To reflect the significance of context awareness, two sample sets are built for LLC and RLC, without classification of context awareness, to train and test the LSTM prediction model. The influences of traffic level and vehicle type are ignored in these samples. To further emphasize the importance of distinguishing LLC and RLC, a sample set of confusing LLC and RLC is built. The predicted results are compared and shown in Table 19. The predicted trajectories of vehicles 1, 3, 4 and 5 without considering context awareness and without considering LLC and RLC are shown in Figs. 15 and 16, respectively.

When the traffic context is ignored in the sample set, the average MSE between the ground-truth and predicted trajectories is much larger than before, increasing from 6.62 to 10.42. It is obvious that the predicted lane change trajectory has a larger deviation from ground-truth data when traffic context is not considered compared with considering traffic context. This indicates that the traffic context should be considered when predicting the lane change decisions and trajectories. The traffic context is a significant factor influencing lane change maneuvers. In addition, the comparison in Table 16 indicates that the prediction accuracy of trajectories decreases if the RLC and LLC samples are not separately considered. Therefore, lane change samples should be divided into RLC and LLC sample sets to train the LSTM-based prediction model. Moreover, this study also addresses the necessity of the lane change decision prediction model for RLC and LLC.

To clarify which surrounding vehicle has the most essential impact on the trajectory prediction of the subject vehicle, one variable is removed each time for vehicle 4 to train the prediction model again. The MSE of the remaining variables in the prediction model is shown in Fig. 17. It can be seen that the *PV* (i.e., the preceding vehicle in the original lane) has the most important influence on the trajectory prediction because the MSE greatly increases when d_1 and vr_1 are removed. The average MSE when removing PV variables increases more than 3 and 1.5 times when PV_T and FV_T variables are removed, respectively. In addition, drivers pay more attention to the space headway between the *SV* and *PV*, PV_T, FV_T rather than the relative speed because the MSE greatly increases when headway is removed from the model.

3.5. Validation of the proposed model

To validate the proposed model, we use the NGSIM dataset to predict the lane change maneuvers incorporating traffic context. The NGSIM dataset (Kovvali et al., 2007) is widely applied in traffic microsimulation research and development, which is especially important for understanding and researching driver behavior at microscopic levels. The I-80 subset of NGSIM is extracted to validate the model. The trajectory data was collected on a segment of I-80 freeway (from 4:00p.m. to 4:15p.m.) in Emeryville, California in 2005. Trajectories are automatically extracted by computer vision algorithms with 10 Hz. The variables of I-80 subset are similar as

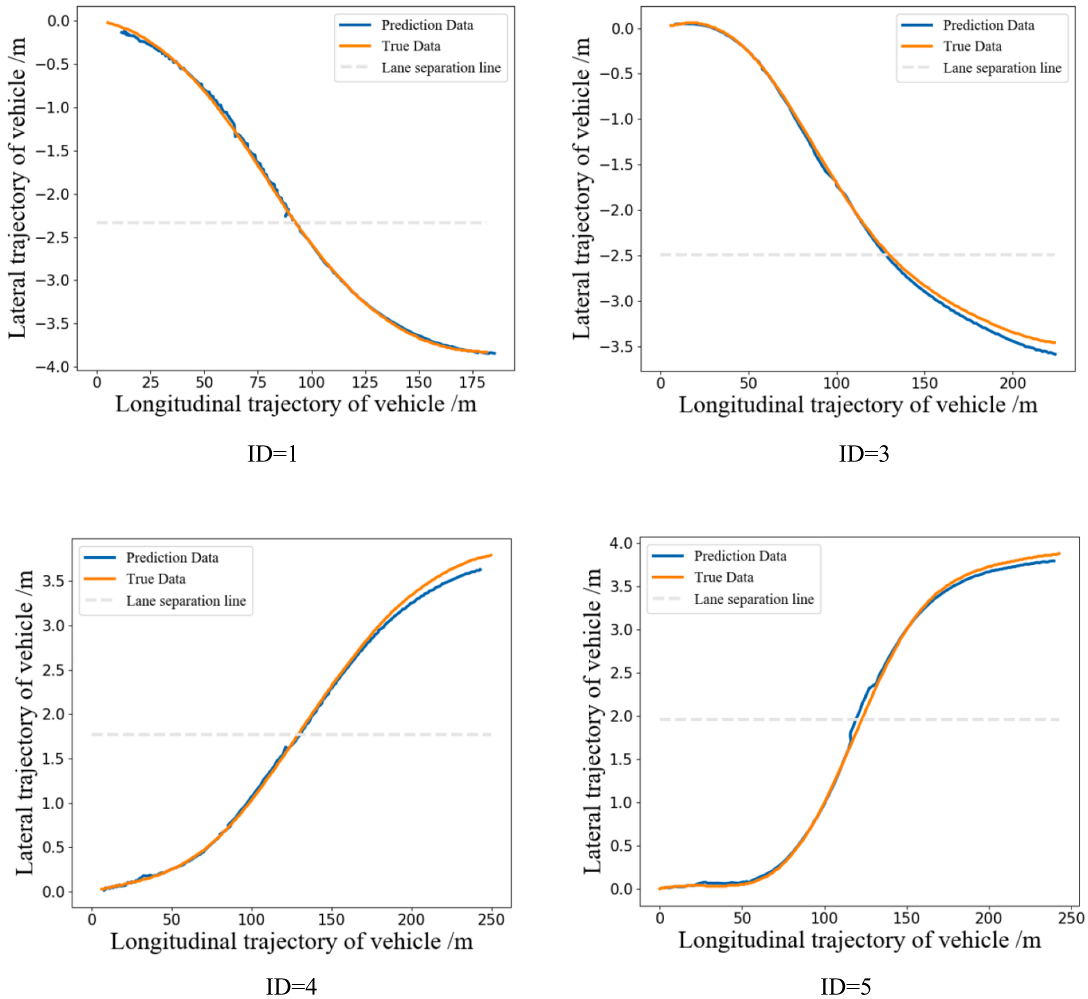


Fig. 14. The true trajectory and predicted trajectory using the LSTM algorithm.

Table 19
The MSE comparison of considering traffic context using the LSTM-based model.

Situation	Total MSE	Average MSE	Lateral MSE	Longitudinal MSE
Considering traffic context and distinguishing LLC and RLC	119.12	6.62	0.01	13.23
Without considering traffic context and distinguishing LLC and RLC	187.68	10.42	0.01	20.85
Without considering traffic context and confusing LLC and RLC	201.80	11.21	0.01	22.42

highD, which can refer to section 3.1. We extract 450 LK samples and 509 LC samples from the subset in the paper.

In order to train the model, the vehicle type and traffic flow level need to be labeled first. The vehicle type of samples is labeled according to the dataset. The traffic flow is labeled based on the k-means method in section 2.2. The silhouette values of k-means are respectively 0.58, 0.64, and 0.57 when the traffic flow is classified into two, three, and four levels. Then we label the traffic flow of samples into three levels, as level 1, level 2, level 3. The descriptive statistics of three groups is depicted in Table 20. Refer to the LOS criteria in Table 4, the LOS of three groups are respectively B, D, and E. As the density of three classes are different, we can label them as three levels of traffic flow. The sample distribution under different contexts is shown in Table 21. The comparison between the different groups of samples reveals the impact of traffic context on lane change decisions and trajectories.

3.5.1. The traffic context classification

It has been indicated that the frequency-domain features outperform time-domain features. Thus, we extracted frequency-domain features of 959 samples as the inputs of the traffic context classification models. The outputs of the traffic level classification model are

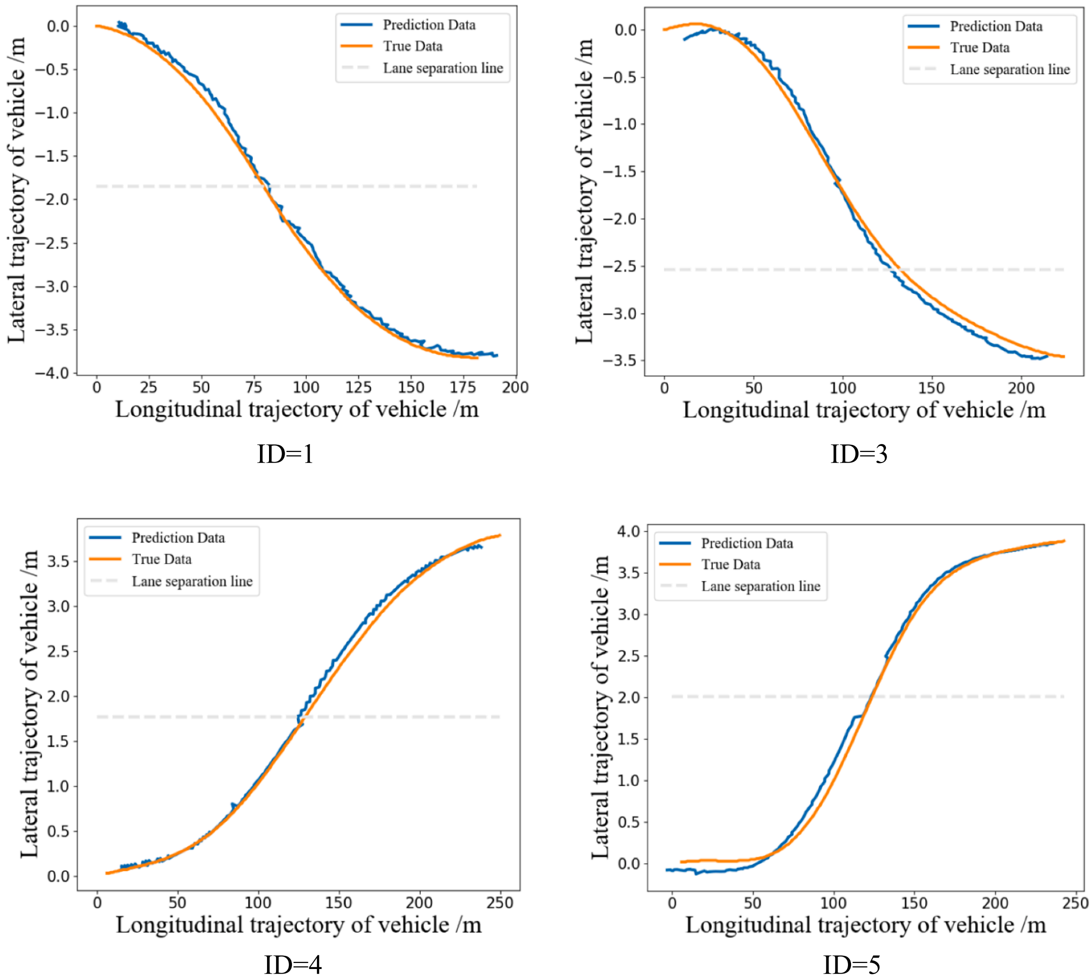


Fig. 15. Trajectory prediction based on the LSTM-based model without considering traffic context.

three levels of traffic. The outputs of the vehicle type classification model are car or truck. We choose three algorithms to conduct the classification, that is XGBoost, AdaBoost and SVM.

The traffic level and vehicle type classification results are respectively shown in Table 22 and 23. The accuracy of traffic level classification achieves the highest 84.95%. The accuracy of vehicle type classification achieves the highest 96.15%. The XGBoost algorithm outperforms other two algorithms.

3.5.2. The lane change decision and trajectory prediction

(1) The lane change decision prediction

We adopt 450 LK, 109 RLC, and 400 LLC samples to train the lane change decision prediction model. The trajectory data are extracted 0.5 s before the vehicle crosses the pavement markings to predict whether the vehicle would perform the lane change maneuver a few seconds later. The lane change decision prediction based on XGBoost with frequency-domain features is shown in Table 24. To compare the influence of traffic context, the decision is predicted without considering contexts. It indicates that with context information, the accuracy of decision is 95.13%. Without context information, the prediction accuracy decreases from 95.13% to 94.79%. The MSE of prediction results increases from 0.15 to 0.16.

(2) The lane change trajectory prediction

All 509 lane change samples are applied in the lane change trajectory prediction model. We select 10 samples to show the MSE of the predicted trajectory and true trajectory in Table 25. In addition, the MSE comparison of considering traffic context using the LSTM-

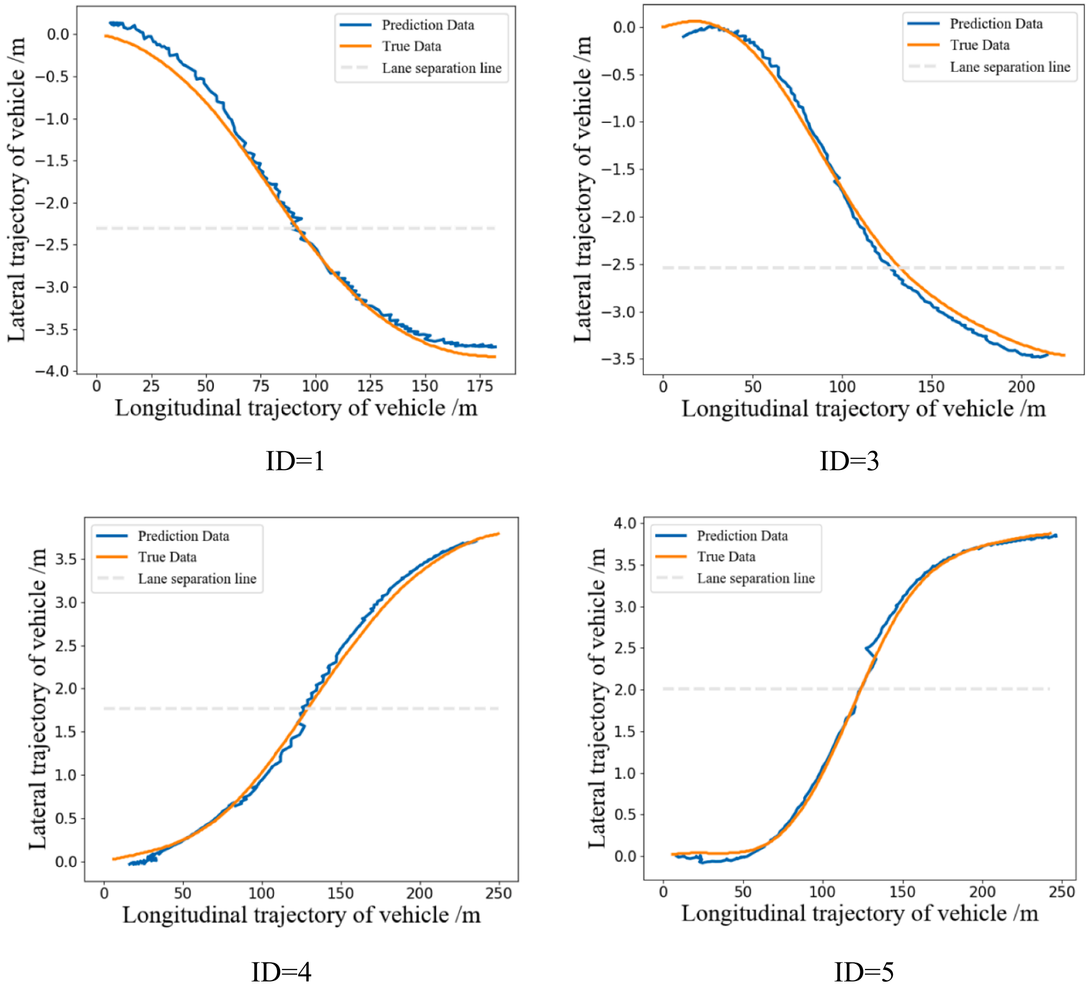


Fig. 16. LSTM-based trajectory prediction without considering traffic context but distinguishing RLC and LLC.

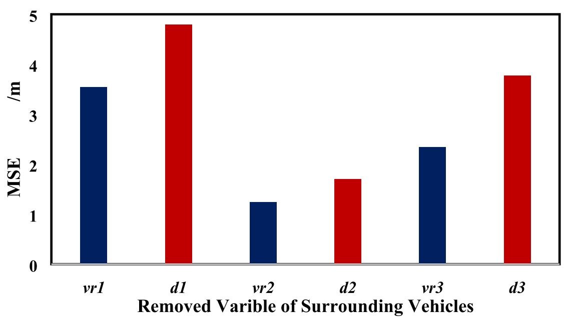


Fig. 17. The MSE comparison of removing the variables of surrounding vehicles.

Table 20
The statistics of three traffic flow levels.

Variables	Density (pc/mile/ln)	LOS
Level 1	20.84	B
Level 2	40.35	D
Level 3	49.39	E

Table 21
The sample distribution under different contexts.

Context	Car	Truck	All
Level 1	112	54	116
Level 2	316	110	426
Level 3	307	60	367
All	735	224	959

Table 22
Traffic level classification results.

Algorithm	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
XGBoost	85.28	84.95	83.76	**84.95**	**0.15**
AdaBoost	81.83	82.52	82.12	82.52	0.19
SVM (linear)	73.24	73.95	72.73	73.91	0.25

Table 23
Vehicle type classification results.

Algorithm	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
XGBoost	92.46	96.05	94.27	**96.15**	**0.08**
AdaBoost	92.42	95.03	93.72	95.06	0.09
SVM (linear)	90.46	91.12	90.59	91.37	0.11

Table 24
The prediction result comparison with and without traffic context.

Comparison	Class name	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	MSE
Traffic context	LK	93.43	96.24	94.81	95.13	0.15
	RLC	91.30	91.30	91.30		
	LLC	97.65	94.69	96.15		
Without considering traffic context	LK	94.03	94.73	94.38	94.79	0.16
	RLC	88.00	95.65	91.66		
	LLC	96.89	94.69	95.78		

Table 25
The MSE of the trajectory predicted by the LSTM algorithm.

ID	MSE	ID	MSE
1	0.17	6	0.65
2	0.15	7	0.24
3	0.12	8	0.15
4	0.06	9	0.28
5	0.24	10	0.11

Table 26
The MSE comparison of considering traffic context using the LSTM-based model.

Situation	Average MSE	Lateral MSE	Longitudinal MSE
Considering traffic context and distinguishing LLC and RLC	**0.22**	0.01	0.43
Without considering traffic context and distinguishing LLC and RLC	0.25	0.01	0.48
Without considering traffic context and confusing LLC and RLC	0.41	0.01	0.81

based model in Table 26. It can be seen that the average MSE of predicted trajectory is the least when considering traffic context and lane change type, which is 0.22. If we do not consider the impact of context, the average MSE will increase to 0.25, and it will get to 0.41 if both context information and lane change type are ignored. Each sample of LLC and RLC predicted trajectory are plotted, shown in Fig. 18. As a comparison, the predicted trajectories are plotted when ignoring the context and confusing lane change type in Fig. 19. It indicates that the proposed model incorporating traffic context performs well on trajectory prediction.

The lane change prediction results above show that the prediction accuracy of lane change intention achieve 95.13%, and the average MSE of predicted trajectory is 0.22. And the model incorporating traffic context performs better in lane change prediction

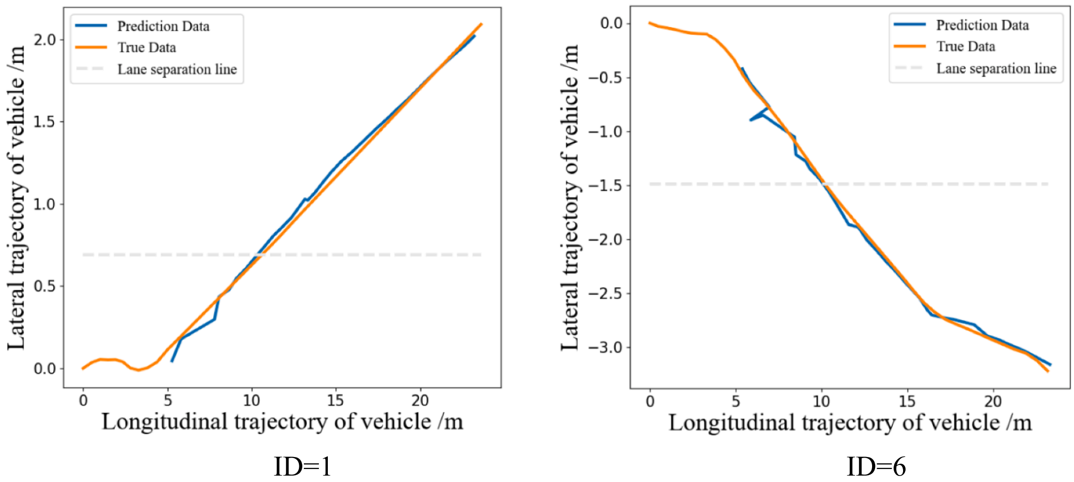


Fig. 18. LSTM-based trajectory prediction considering traffic context and distinguishing RLC and LLC.

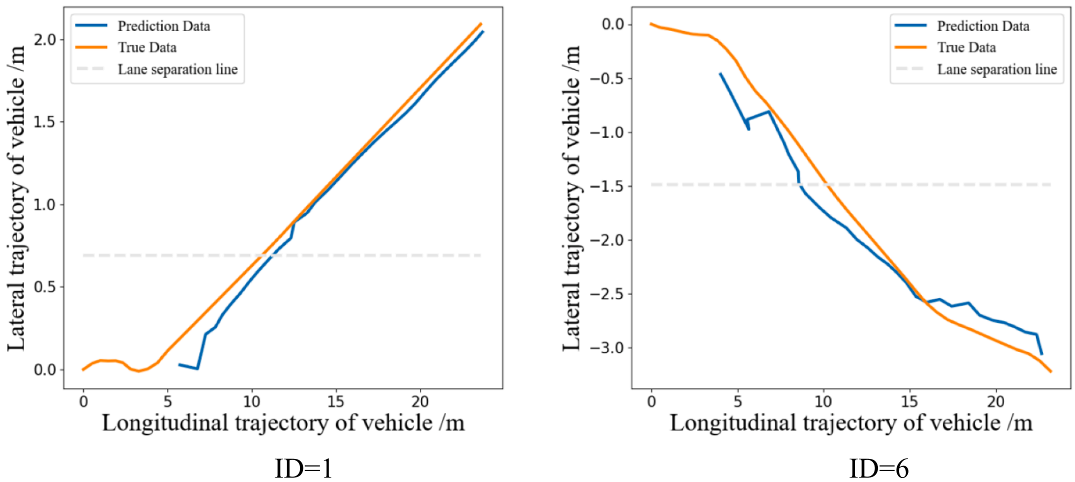


Fig. 19. LSTM-based trajectory prediction without considering traffic context and confusing RLC and LLC.

compared with model without contexts. It indicates that the proposed model can transfer to other datasets.

4. Conclusions

We establish an integrated lane change prediction model incorporating traffic context by machine learning algorithms on trajectory data from the highD dataset. The traffic context is taken into consideration when predicting lane change maneuvers. The impact of traffic level and vehicle type on lane change maneuvers is investigated in this study. The outputs of the traffic context classification model are transformed into the lane change prediction model. The integrated lane change prediction model consists of the XGBoost-based lane change decision prediction model and LSTM-based lane change trajectory prediction model, achieving the accurate prediction of the whole lane change process. The proposed models are also implemented on lane change samples extracted from NGSIM dataset, proving the adaptability of the models.

In this study, the frequency-domain features are extracted by DWT and DFT algorithms. Then, time-domain features are compared with frequency-domain features in the prediction results, which show more advantages. The t-SNE algorithm is applied to excavate the key features from high-dimensional data to show the difference in lane change behavior between contexts. This may be the first time that the frequency-domain features have been applied in lane change prediction. The results show that the integrated lane change prediction model can accurately predict the lane change process of a vehicle. By using the integrated lane change prediction model, lane change decisions could be predicted with an accuracy of 98.20%, and lane change trajectories could be predicted with an average

MSE of 6.62. The prediction accuracy of lane change decisions decreases to 97.02% when not considering traffic context, while the MSE of lane change trajectory predictions increases to 11.21 when not considering traffic context and trajectory differences in LLC and RLC. It also should be noted that even the accuracy of lane change decisions only increases from 97.02% to 98.20% incorporating traffic context, it may lead to great improvement in the real world. For example, there are 10 000 lane change maneuvers on the freeway during a specific time. The integrated lane change model can accurately predict 118 more lane change decisions, which will definitely have great impact on the traffic flow. It can also facilitate the development of the autonomous vehicles (AV), improving the safety level and the application of the AVs.

These above results confirm the influence of traffic context on lane change maneuvers, and that the difference in LLC and RLC should be considered when predicting lane change trajectories. Moreover, the proposed model reveals that the most important factor associated with lane change decisions is the relative speed between the subject vehicle and preceding vehicles in the original and target lanes, while the relative position of the preceding vehicle in the original lane has the most important influence on lane change trajectory predictions.

There are still some limitations in this study. First, vehicle trajectories can only be used to recognize the on-going lane change decision instead of predicting the lane change decision before any actions are taken. Driver behavioral features, such as the head pose and eye gaze signals, can provide earlier clues about the driver's intention. Therefore, both driver behavior and vehicle dynamics should be jointly used to predict lane change decisions in future work. Second, some other traffic contexts, such as weather conditions and road alignment, may also affect the results. These factors can be included in our modeling framework when more trajectory data are available.

CRediT authorship contribution statement

Qingwen Xue: Methodology, Data curation, Software, Writing – original draft, Writing – review & editing, Visualization. **Yingying Xing:** Conceptualization, Writing – original draft, Writing – review & editing, Supervision. **Jian Lu:** Writing – review & editing, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Part of the code is shared as a link in the revised paper.

Acknowledgments

This study has been funded by the National Nature Science Foundation of China (NSFC71871165) and Natural Science Foundation of Shanghai (22ZR1465500).

References

Barr, L., Najm W.G., 2001. Crash problem characteristics for the intelligent vehicle initiative. TRB 80th Annual Meeting, Washington, DC.
Bejani, M.M., Ghatee, M., 2018. A context aware system for driving style evaluation by an ensemble learning on smartphone sensors data. Transport. Res. Part C: Emerg. Technol. 89, 303–320.
Butakov, V., Ioannou, P., 2014. Personalized driver/vehicle lane change models for ADAS. IEEE Trans. Veh. Technol. 64 (10), 4422–4431.
Chen T., Guestrin C. Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.
Chen, D., Ahn, S., 2018. Capacity-drop at extended bottlenecks: merge, diverge, and weave. Transport. Res. Part B: Methodol. 108, 1–20.
Chen, T., Shi, X., Wong, Y.D., 2019. Key feature selection and risk prediction for lane-changing behaviors based on vehicles' trajectory data. Accid. Anal. Prev. 129, 156–169.
Chen, Z., Wu, C., Huang, Z., Lyu, N., Hu, Z., Zhong, M., Cheng, Y., Ran, B., 2017. Dangerous driving behavior detection using video-extracted vehicle trajectory histograms. J. Intell. Transport. Syst. 21 (5), 409–421.
Chowdhury, D., Santen, L., Schadschneider, A., 2000. Statistical physics of vehicular traffic and some related systems. Phys. Rep. 329 (4–6), 199–329.
Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine Learning 20 (3), 273–297.
Esfahani, R.K., Shahbazi, F., Akbarzadeh, M., 2018. Three-phase classification of an uninterrupted traffic flow: a k-means clustering study. Transportmetrica B: Transport Dynamics 2018, 1–13.
Gipps, P.G., 1986. A model for the structure of lane-changing decisions. Transport. Res. Part B Methodol. 20 (5), 403–414.
Klitzke, L., Koch, C., Koster, F., 2020. Identification of Lane-Change Maneuvers in Real-World Drivings with Hidden Markov Model and Dynamic Time Warping. In: IEEE Conference on Intelligent Transportation Systems, Proceedings 2020.
Krajewski, R., Bock, J., Kloeker, L., Eckstein, L., 2018. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2018: 2118-2125.
Kumar, P., Perrollaz, M., Lefevre, S., Laugier, C., 2013. Learning-based approach for online lane change intention prediction. In: 2013 IEEE Intelligent Vehicles Symposium (IV). Presented at the 2013 IEEE Intelligent Vehicles Symposium (IV), IEEE, Gold Coast City, Australia, pp. 797–802.
Kovvali, V., Alexiadis, V., Zhang, P.E., 2007. Video-Based Vehicle Trajectory Data Collection. Presented at Proceedings of the Transportation Research Board 86th Annual Meeting.
Laurens, V.D.M., Hinton, G.E., 2008. Visualizing High-Dimensional Data Using t-SNE. J. Machine Learn. Res. 9, 2579–2605.

Leonhardt, V., Wanielik, G., 2018. Recognition of lane change intentions fusing features of driving situation, driver behavior, and vehicle movement by means of neural networks. In: Advanced Microsystems for Automotive Applications. Springer, Cham, pp. 59–69.
Leung, K.M., 2007. Naive bayesian classifier. Polytechnic University Department of Computer Science/Finance and Risk Engineering, 2007: 123-156.
Li, G., Li, S.E., Liao, Y., Wang, W.J., Cheng, B., Chen, F., 2015. Lane change maneuver recognition via vehicle state and driver operation signals—Results from naturalistic driving data2015. In: IEEE Intelligent Vehicles Symposium (IV). IEEE, pp. 865–870.
Li, K.P., Wang, X, Xu, Y.C., W, J.Q., 2016. Lane changing intention recognition based on speech recognition models. Transport. Res. Part C: Emerg. Technol. 69, 497-514.
Li, M., Li, Z., Xu, C., Liu, T., 2020a. Short-term prediction of safety and operation impacts of lane changes in oscillations with empirical vehicle trajectories. Accid. Anal. Prev. 135, 105345. https://doi.org/10.1016/j.aap.2019.105345.
Li, X., Wang, W., Roetting, M., 2018. Estimating Driver's Lane-Change Intent Considering Driving Style and Contextual Traffic. IEEE Trans. Intell. Transp. Syst. 20 (9), 3258–3271.
Li, Z., Wu, C., Tao, P., Tian, J., Ma, L., 2020b. DP and DS-LCD: A New Lane Change Decision Model Coupling Driver's Psychology and Driving Style. IEEE Access 8, 132614–132624.
Messaoud, K., Yahiaoui, I., Verroust-Blondet, A., Fawzi, N., 2019. Relational recurrent neural networks for vehicle trajectory prediction. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, pp. 1813–1818.
Manual, H.C. HCM2010. Transportation Research Board, National Research Council, Washington, DC, 2010.
Norouzi, A., Masoumi, M., Barari, A., Farrokhpour Sani, S., 2019. Lateral control of an autonomous vehicle using integrated backstepping and sliding mode controller. Proc. Instit. Mech. Eng. Part K: J. Multi-body Dyn. 233 (1), 141–151. https://doi.org/10.1177/1464419318797051.
Olkkonen, H., 2011. Discrete Wavelet Transforms: Biomedical Applications. BoD–Books on Demand.
Ossen, S., Hoogendoorn, S.P., 2011. Heterogeneity in car-following behavior: Theory and empirics. Transport. Res. Part C: Emerg. Technol. 19 (2), 182–195.
Rákos, O., Aradi, S., Bécsi, T., 2020. Lane Change Prediction Using Gaussian Classification, Support Vector Classification and Neural Network Classifiers. Periodica Polytechnica Transport. Eng. 48 (4), 327–333.
Safavian, S.R., Landgrebe, D., 1991. A survey of decision tree classifier methodology. IEEE Trans. Syst. Man Cybernet. 21 (3), 660–674.
Schapire, R.E., 2013. Explaining adaboost. Empirical inference. In: Schölkopf, B., Luo, Z., Vovk, V. (Eds.), Empirical Inference. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 37–52. https://doi.org/10.1007/978-3-642-41136-6_5.
Schmidt, K., Beggiato, M., Hoffmann, K.H., Josef, F.K., 2014. A mathematical model for predicting lane changes using the steering wheel angle. J. Saf. Res. 49 (85), e1–90.
Shi, T., Wang, P., Cheng, X., Chan, C.Y., Huang, D., 2019. Driving decision and control for automated lane change behavior based on deep reinforcement learning. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Preceding 2895-2900.
Shou, Z., Wang, Z., Han, K., Liu, Y.K., Prashant, T., Di, X., 2020. Long-Term Prediction of Lane Change Maneuver Through a Multilayer Perceptron. In: IEEE Intelligent Vehicles Symposium, Preceding, 2020.
Singh, K., Li, B., 2011. Estimation of traffic densities for multilane roadways using a markov model approach. IEEE Trans. Ind. Electron. 59 (11), 4369–4376.
Sun, J., Zhang, Q., Tsang, E., 2005. DE/EDA: A new evolutionary algorithm for global optimization. Inf. Sci. 169 (3-4), 249–262.
Wang, J., Zhang, Z., Lu, G., 2021. A Bayesian inference based adaptive lane change prediction model. Transport. Res. Part C: Emerg. Technol. 132, 103363. https://doi.org/10.1016/j.trc.2021.103363.
Wissing, C., Nattermann, T., Glander, K.-H., Hass, C., Bertram, T., 2017. Lane Change Prediction by Combining Movement and Situation based Probabilities. IFAC-PapersOnLine, 20th IFAC World Congress 50 (1), 3554–3559. https://doi.org/10.1016/j.ifacol.2017.08.960.
Xie, D.-F., Fang, Z.-Z., Jia, B., He, Z., 2019. A data-driven lane-changing model based on deep learning. Transport. Res. Part C: Emerg. Technol. 106, 41–60.
Xing, Y., Lv, C., Wang, H., Cao, D., Velenis, E., 2020. An ensemble deep learning approach for driver lane change intention inference. Transport. Res. Part C: Emerg. Technol. 115, 102615. https://doi.org/10.1016/j.trc.2020.102615.
Xiong, L.u., Yang, X., Leng, B.o., Zhang, R., Fu, Z., Zhuo, G., 2021. Integrated longitudinal and lateral control for autonomous vehicles with active load transfer strategy at the handling limits. Proc. Instit. Mech. Eng. Part D: J. Automobile Eng. 235 (4), 961–974.
Yang, M., Wang, X., Quddus, M., 2019. Examining lane change gap acceptance, duration and impact using naturalistic driving data. Transport. Res. Part C: Emerg. Technol. 104, 317–331.
Zhang, B., Ding, Z., Zhou, M.A., 2020. Lane change prediction algorithm based on probabilistic modeling. In: Proceedings of the 2020 International Conference on Computers, Information Processing and Advanced Education 211-216.
Zhang, Y., Lin, W.C., Chin, Y.-K., 2010. A pattern-recognition approach for driving skill characterization. IEEE Trans. Intell. Transp. Syst. 11 (4), 905–916.
Zhang, Z., 2018. Improved Adam Optimizer for Deep Neural Networks, in: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). In: Presented at the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), pp. 1–2.