Turning Maneuver Prediction of Connected Vehicles at Signalized Intersections A Dictionary Learning-Based Approach

Zhang, Hailun; Fu, Rui; Wang, Chang; Guo, Yingshi; Yuan, Wei

DOI
10.1109/JIOT.2022.3188312

Publication date
2022

Document Version
Final published version

Published in
IEEE Internet of Things Journal

Turning Maneuver Prediction of Connected Vehicles at Signalized Intersections: A Dictionary Learning-Based Approach

Hailun Zhang, Rui Fu, Chang Wang, Yingshi Guo, and Wei Yuan

***Abstract*—Vehicle-to-Infrastructure (V2I) communication has provided a solution for the improvement of the traffic efficiency of smart city intersections. For example, turning maneuvers prediction at signalized intersections in a connected environment helps traffic command centers time traffic lights and dynamically predict traffic flow. However, the modeling methods used in existing research on this topic have some limitations, such as poor scalability and interpretability of machine learning. Thus, this study proposes a dictionary learning-based approach to predict turning maneuvers before the intersection. The proposed dictionary model estimates the LogDet divergence-based sparse inverse covariance matrix (LDbSICM) of driving behavior samples. The graphical lasso method is used to estimate the sparse inverse covariance matrix of the driving samples to construct a dictionary library of the maneuver behavior. The LogDet divergence is used to calculate the difference between each inverse covariance matrix. A driving simulator is utilized to collect experimental data consisting of turning left (TL), turning right (TR), and going straight (GS) behaviors to establish and evaluate the proposed model. The experimental results demonstrate that the proposed dictionary learning-based turning maneuver prediction model achieves 100% prediction accuracy for TL and GS and 97.2% for TR. The proposed model has substantial advantages over existing methods. The model can predict TL, TR, and GS in a connected environment 270, 280, and 290 m, respectively, before the intersection.**

***Index Terms*—Graphical lasso, LogDet divergence, signalized intersection, turning maneuver prediction, Vehicle-to-Infrastructure (V2I).**

I. Introduction

A. Motivations and Challenges

THE SAFETY and smoothness of intersections are crucial for urban traffic. However, statistics show that intersection-related accidents account for a relatively high rate of total accidents in the United States [1], [2], Europe [3], and China [4]. These statistics reveal that maneuvering at intersections is a highly dangerous driving task due to the complex traffic flow in the area. Therefore, the construction of smart cities and the development of intelligent transportation systems require particular attention to traffic safety and traffic efficiency issues at intersections. The connected environment has received increasing attention [5]–[7] since the U.S. Department of Transportation and the Ministry of Science and Technology of the People's Republic of China launched their respective intelligent connected transportation environment projects, which emphasize traffic safety and traffic efficiency [8], [9]. Technological advances in sensors and communication have promoted the development of a connected traffic environment. Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and Vehicle-to-Everything (V2X) communications are believed to improve driving behavior and solve large-scale traffic problems related to mobility, efficiency, safety, and environmental impact [10], [11].

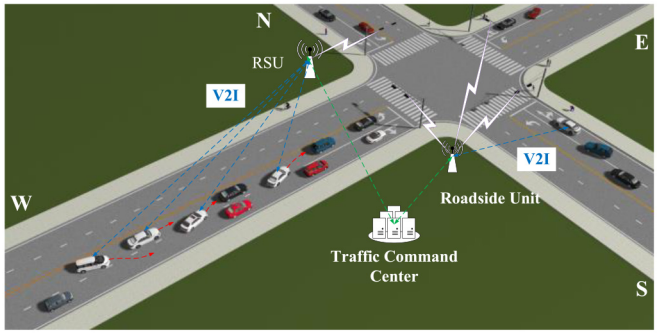

Fig. 1. Scenario in which CVs enter an intersection and interact with the infrastructure.

In the V2I mode, a vehicle can communicate with the infrastructure via onboard sensors and communication networks to obtain basic traffic information. The driver adjusts their driving behavior in time by observing the traffic information disseminated by the human–machine interface (HMI), improving traffic efficiency and traffic safety [10], [12]–[15]. Of course, the roadside units (RSUs) can also obtain the vehicle kinematics pattern in this environment to predict the vehicle's maneuvering behavior at the intersection. Achieving large-scale maneuver prediction provides the basis for the traffic command center (TCC) to dynamically plan traffic. Fig. 1 illustrates a common case of a busy intersection.

Manuscript received 15 March 2022; revised 2 May 2022 and 13 June 2022; accepted 30 June 2022. Date of publication 4 July 2022; date of current version 7 November 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1600501; in part by the China Scholarship Council under Grant 202106560009; and in part by the Fundamental Research Funds for the Central Universities under Grant CHD 300102222709. *(Corresponding author: Rui Fu.)*

Hailun Zhang is with the School of Automobile, Chang'an University, Xi'an 710064, China, and also with the Department of Transport and Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology, 2600 Delft, The Netherlands (e-mail: iszhanghailun@outlook.com).

Rui Fu, Chang Wang, Yingshi Guo, and Wei Yuan are with the School of Automobile, Chang'an University, Xi'an 710064, China (e-mail: furui@chd.edu.cn; wangchang@chd.edu.cn; guoys@chd.edu.cn; yuanwei@chd.edu.cn).

Digital Object Identifier 10.1109/JIOT.2022.3188312

Connected vehicles (CVs) can interact with the traffic infrastructure via the V2I technology. RSUs send traffic light signal information or other traffic information to the vehicles, and vehicle sensors can send vehicle kinematic data to the TCC via the RSUs. The TCC can then dynamically time the signal lights according to the real-time traffic flow, thereby alleviating traffic congestion and improving traffic efficiency. Drivers are a vital component of the transportation system. The connected environment changes the behavior of drivers, affecting the entire transportation system. According to this logic, this work focuses on the prediction of driving maneuvers of CVs at signalized intersections.

Few studies focused on turning prediction at intersections in the Internet of Vehicles environment. Predicting turning intentions in the traditional environment is typically achieved using data-driven machine learning methods [16], [17]. However, extensive research has been performed on trajectory and motion prediction [18]–[20]. It should be noted that trajectory prediction or motion planning refers to estimating the potential driving trajectory of vehicles, and the model outputs a trajectory sequence. This work focuses on predicting the future driving state of the driver at intersections, i.e., turning left (TL), turning right (TR), and going straight (GS).

It is challenging to predict turning maneuvers at a distance from an intersection without a dedicated lane. When a driver receives information disseminated by the HMI, they will inevitably make the most reasonable and safest decision [21]. In this case, the prediction of turning maneuvers is a more challenging problem because the driver's behavior is affected by the connected information. In addition, due to the driver's access to connected information, the prediction model must consider many variables, increasing the difficulty of model training. In this work, a dictionary learning-based model is developed to predict the maneuvers (i.e., turning or GS) of a CV entering an intersection in a connected environment. This model can be used in a traffic control system for predicting and controlling the traffic flow after obtaining the maneuvering intention of the CVs.

B. Related Research

Data-driven and advanced machine learning methods have exhibited advantages over other methods for behavior prediction/anticipation [22], [23]. This type of method mainly uses behavior features extracted from raw data to achieve maneuver detection and prediction [24], [25]. Unlike behavior recognition [26], [27], driving maneuver prediction refers to predicting a potential maneuver before it occurs. Early prediction provides the TCC with more planning time, enabling it to control the traffic.

When assessing the driving situation and deciding on the next maneuver, the driver obtains information from different sources, such as the surrounding traffic, vehicle status, and destination. Therefore, the prediction model must fuse these multisource cues to predict the driver's maneuver state [25], [28]. Before entering an intersection, the driver checks the traffic situation, accelerates or decelerates the vehicle, and moves into the appropriate lane. These behavior patterns can be used as input information for data-driven models [3], [29], [30]. Therefore, machine learning methods are used extensively for driving behavior prediction and have provided encouraging results [30], [31]. It was hypothesized that the vehicle's motion state information could be obtained via V2V and V2I communication [32]. Two classifiers based on a support vector machine integrated with Bayesian filtering (SVM-BF) and a classifier based on the hidden Markov model (HMM) were constructed. Some studies [33], [34] used a Gaussian regression model to predict the maneuver behavior of traffic participants at intersections. The absolute position and speed of the vehicle were used as the model input, regardless of the vehicle's intention and the interaction between the vehicles. Song *et al.* [35] developed a continuous HMM to predict high-level movement intentions and low-level interaction intentions, using the distance to the intersection, the longitudinal speed, the acceleration, and the yaw rate as inputs, and achieved satisfactory prediction performance. Tran *et al.* [36] developed a vision-based method to predict the driver's braking and acceleration behavior using optical flow-based foot tracking and HMM-based techniques to characterize temporal behavior. Gadepally *et al.* [37] proposed a hybrid-state system (HSS) based on the HMM to identify TL, TR, GS, and stopping at an intersection. Yu *et al.* [38] developed a random forest (RF)-based algorithm to predict drivers' reactions and achieved an accurate prediction of the driver's response within 87.4 m of the intersection. Doshi and Trivedi [39] found that head movement, lane position, and vehicle dynamics provided a good combination of indicators for the prediction of drivers' intentions. Gaze information does not improve the prediction performance of lane-changing behavior. However, no research has explored the role of gaze information in predicting driving behavior at intersections. Yi *et al.* [3] proposed a personalized driver intention prediction system for unsignalized T intersections consisting of polynomial regression mixture clustering and classification and regression tree (CART) algorithms. The system achieved good intention prediction performance and could be adapted to different drivers.

Zyner *et al.* [30] used the lateral and longitudinal positions, heading angle, and speed relative to other vehicles as the inputs of a recurrent neural network (RNN) to predict a potential conflict 1.3 s in advance. Similarly, Ou and Karray [25] proposed a driving maneuver prediction system using a deep RNN to fuse driver observation behavior and driving environment information. They achieved high-precision prediction of driving maneuvers due to new data labeling methods and effective sequence modeling. Ou and Karray [25] designed a driver activity prediction framework composed of RNNs and long short-term memory (LSTM) networks to capture long-term dependencies. This method used sequence-to-sequence prediction and achieved a prediction accuracy of 83.8% at 3.78 s before the start of the turning behavior. Similarly, RNNs were also used to predict the behavior of drivers [25], [30], [40]. Olabiyi *et al.* [41] proposed a sliding window bidirectional RNN to extract the feature representation of spatiotemporal information from both ends of the input sequence. In addition, the researchers propose to use

deep learning methods to predict driving maneuvers from images. For example, Xu *et al.* [42] proposed an end-to-end LSTM-based architecture to directly predict driving actions. Du *et al.* [43] compared the prediction performance of transfer learning, convolutional neural network, LSTM, and ResNet on steering angle. Chen *et al.* [44] proposed a deep neural network to predict action transitions and action selection probabilities.

The conclusions drawn from these studies can be summarized as follows. Simple classifiers (i.e., SVMs or the *k*-nearest neighbor (KNN) algorithm [37]) can only model and predict the current observation vector but cannot model and mine hidden states in the time series. The HMM is suitable for time-series modeling, but its state transition probability cannot be updated after training. Thus, the generalization ability of the algorithm is low in complex traffic environments and for different driving styles. Deep learning, e.g., RNNs, has become the mainstream method for predicting driving maneuver behavior at intersections due to its effectiveness. However, the limitation of RNNs is their poor interpretability. These methods require large amounts of time and effort for model training for a large sample size and multiple variables.

Drivers dynamically adjust their driving behavior in the V2I communication mode, indicating that the V2I mode affects driving decisions. Also, maneuvering behaviors before intersections often exhibit randomness due to various driving styles and environmental factors. Therefore, many variables must be included in a prediction model because it is unknown which variables have a high predictive capability. In recent years, sparse representation has become a hotspot in machine learning and computer vision research. The basic assumption is that the image represents a sparse signal, and the input signal is expressed linearly with a set of overcomplete bases. The expansion coefficient provides a good approximation of the original signal under certain sparseness conditions [45]. Studies have shown that the sparse representation model based on signal reconstruction has achieved good pattern recognition performance [46]. Excellent performance has been achieved in different related fields, such as face recognition [45], pedestrian trajectory recognition [47], and speech emotion recognition [48]. The essence of sparse representation is to express the maximum amount of information with the minimum amount of resources to achieve fast calculations. Therefore, the maneuvering behavior time series can be treated as a 2-D matrix for mathematical processing, and a dictionary model can be constructed using sparse representation theory to express turning maneuvers effectively and robustly. The model is characterized by modularity and can incrementally model or update a certain maneuver without affecting other parts of the model. This strategy overcomes the limitation of the poor scalability of traditional methods.

C. Contributions

It is often difficult to predict steering maneuvers at a distance from the intersection and no dedicated turning lane. A CV can send the driver's turning intention to the TCC to support efficient management of intersection traffic. We establish a turning maneuver prediction model for connected environments to overcome the limitations of existing models. To the best of our knowledge, dictionary learning for turning maneuver prediction has not been used in previous studies. The main contributions of this article are listed as follows.

1) The development of a turning maneuver prediction model using dictionary learning by estimating the LogDet divergence-based sparse inverse covariance matrix (LDbSICM), which represents the multidimensional parameter time series of driving behavior.
2) The use of the graphical lasso method to estimate the sparse inverse covariance, which improves the speed and ease of updating the learning samples of the model.
3) The conduct of a driving simulator experiment in a connected environment to collect high-quality vehicle kinematic data, driver observation activity data, and controller area network (CAN) bus data for model establishment and evaluation.

The results indicate that the proposed model achieves higher prediction accuracy than benchmark models.

D. Article Organization

The remainder of this article is organized as follows. The theoretical explanation of the proposed method is given in Section II. Section III provides a detailed description of the driving simulator experiment, data collection, and data processing. The experimental results are discussed in Section IV. Finally, Section V concludes this article and outlines the potential future work.

II. Methodology

This section briefly introduces the behavior of drivers at intersections and formulates the problems addressed in this research. The sparse representation concept is introduced, and the model construction details are presented.

A. Analysis of Driving Behavior at an Intersection and Problem Formulation

Maneuvering behaviors, such as changing lanes, overtaking, and cutting in, are the results of the interaction between the driver and the surrounding environment [27], [49]–[51]. The difference between these behaviors and turning behavior is that the driver's turning intention at an intersection is often not influenced by other road users. While approaching the intersection, drivers must observe and evaluate the surrounding traffic environment and traffic lights, adjust the speed and trajectory in time, and change lanes at specific locations to comply with traffic rules. Analyses have revealed that the indicators that predict the behavior of drivers at intersections are the driver's observations, the acceleration and deceleration behavior, the vehicle trajectory, and the kinematic characteristics.

An urban scene including multiple intersections was designed, and a driving simulator experiment simulating daily driving tasks was carried out (see Section III). The collected data included the driver's head rotation and gaze information, the vehicle kinematic data, the vehicle trajectory data, and the

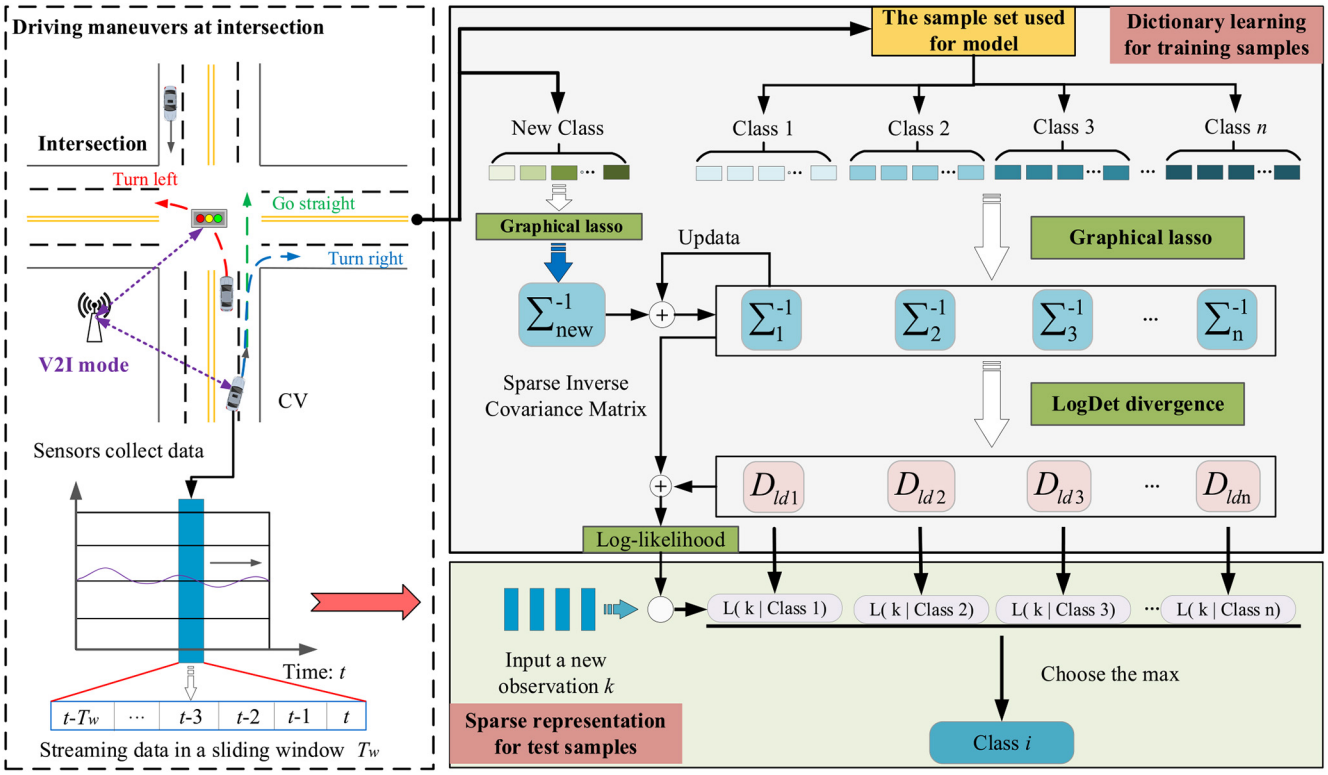

Fig. 2. Architecture of the proposed method.

drivers' behavior data (collected by the CAN bus). The goal was to predict the future maneuvering state of the driver given the historical observation sequence $\boldsymbol{\xi}$ with a length of T using the sliding window technique. The input of the proposed model is the information sequence

$$\boldsymbol{\xi} = [\xi_{t-T}, \xi_{t-T+1}, \ldots, \xi_t]$$

where $\boldsymbol{\xi}$ is an observation sequence of length $\boldsymbol{T}$. ξ_t includes the 3-D coordinates of the Rodrigues head rotation (Rx, Ry, Rz) and gaze direction (Gx, Gy, Gz), head heading, gaze heading, abscissa and ordinate of the trajectory, heading angle and yaw rate, longitudinal and lateral acceleration, longitudinal and lateral velocity, vehicle speed, steering wheel torque, steering wheel angle, steering wheel angular rate, brake pedal displacement and force, and the throttle opening at time instance t. The output is a set of log-likelihood values at t

$$\boldsymbol{L}_t = [L_t^1, L_t^2, \ldots, L_t^{\mathrm{n}}]$$

where L_t^{n} represents the log-likelihood value that a driver will choose a maneuver action n in the near future. Hence, at any time t, the maneuver prediction model models the conditional likelihood

$$L_t^{\mathrm{k}} = \mathrm{L}(\mathrm{k}|[\xi_0, \ldots, \xi_t], \phi), \mathrm{k} \in \{1, \ldots, \mathrm{n}\}$$

where k is the new observation input, L is the calculation formula of the log likelihood, $[\xi_0, \ldots, \xi_t]$ represents the previous input information used to infer future maneuvers, and ϕ represents all parameters in the model.

B. System Architecture and Sparse Representation Theory

The essence of sparse representation is to express the maximum amount of information with the minimum amount of few resources, thereby enabling rapid calculation. The vehicle behavior expressed by multiple parameters can be mathematically treated as a 2-D matrix, and sparse representation theory can be used to construct a dictionary model to express the vehicle behavior effectively and robustly. The model is characterized by modularity and can simultaneously model different turning or GS behaviors and incrementally model a certain behavior without affecting other parts of the model. This strategy overcomes the limitation of the poor scalability of traditional methods. The architecture of the proposed algorithm is illustrated in Fig. 2. Although the method has a similar structure as existing machine learning methods commonly used for classification problems, it has strong extensibility and interpretability. First, unlike traditional pattern recognition models, the key of the proposed algorithm is to learn a sparse inverse covariance dictionary from a large number of samples using the graphical lasso method, thereby simplifying the training process and reducing model complexity. For newly added samples, it is only necessary to update the sparse inverse covariance matrix of the corresponding sample class using the graphical lasso method without retraining the model. Second, for the newly added class, it is only necessary to add the sparse inverse covariance matrix representing the class to the dictionary. Subsequently, the log-likelihood value corresponding to each class of the sample is calculated to recognize a new sample, and the maximum value is selected for sample identification. The function of the LogDet

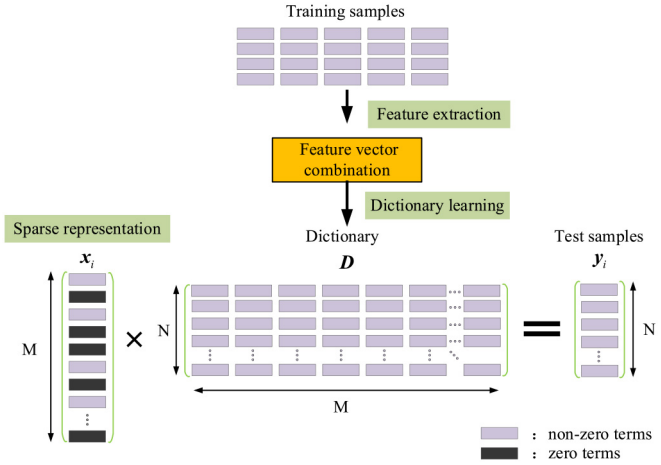

Fig. 3. Schematic framework of the sparse representation theory.

divergence is to calculate the difference between each sparse inverse covariance matrix.

The sparse representation is designed to find the smallest possible number of solutions that contain nonzero terms; thus, it is essential to find a concise representation of the signal. Fig. 3 presents the process of expressing the test sample using sparse representation theory. The basis of sparse linear representation is to find an $N \times M$ dictionary matrix $\boldsymbol{D}$ for the training sample. The given data $\boldsymbol{y}$ are then linearly modeled as $\boldsymbol{D} \cdot \boldsymbol{x}$. In other words, in sparse coding, a given test signal $\boldsymbol{y}_i \in \mathbb{R}^N$ can be accurately represented as a linear combination of the dictionary atoms $\boldsymbol{D}_{ij} \in \mathbb{R}^{N \times M}$ and the sparse representation $\boldsymbol{x}_i \in \mathbb{R}^M$. It is expected that $\boldsymbol{x}_i$ can reconstruct $\boldsymbol{y}_i$ while being as sparse as possible.

When sparse coding is used for classification, the training data can be used as atoms in the dictionary. Assuming that there are k class samples in the database, the class i training sample can be expressed as $\boldsymbol{D}_i = [d_{i,1}, d_{i,2}, \ldots, d_{i,m}] \in \mathbb{R}^{n \times m}$, where $d_{i,j}$ is the jth sample in class i, n is the dimension of the sample, and m is the total number of class samples. Thus, if test sample $\boldsymbol{y}$ belongs to class i, there may be a sparse representation $\boldsymbol{x}$ using the dictionary $\boldsymbol{D}_i$, i.e., $\boldsymbol{y} = x_{i,1}d_{i,1} + x_{i,2}d_{i,2} + \cdots + x_{i,m}d_{i,m}$, where most of the coefficients $x_{i,j}$ are zero, and $j = 1, 2, \ldots, m$.

Each $\boldsymbol{D}_i$ is concatenated to form a dictionary $\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2, \ldots, \boldsymbol{D}_k]$, which serves as a sparse coding dictionary of the test samples, where $i = 1, 2, \ldots, k$, and each $\boldsymbol{D}_i$ describes an individual class. The linear combination approximation of the test sample $\boldsymbol{y}$ can be expressed as

$$\hat{\boldsymbol{x}} = \arg\min_x \|\boldsymbol{y} - \boldsymbol{D} \cdot x\|_2^2; \ \|x\|_0 \leq \rho \tag{1}$$

where ρ is the sparsity constraint. The imposition of ρ on the representation $\boldsymbol{x}_i$ leads to optimization problems. $\hat{\boldsymbol{x}}$ is the sparse representation coefficient of $\boldsymbol{y}$, the ℓ_0-norm in the constraint condition represents the number of nonzero items in the vector $\boldsymbol{x}$, and its physical meaning is the degree of sparsity. Ideally, the coefficients corresponding to the base of all classes should be zero, except for the coefficient corresponding to the base of the class to which sample $\boldsymbol{y}$ belongs, which is nonzero. It should be noted that solving the ℓ_0-norm is a nondeterministic polynomial hard (NP-hard) problem; therefore, the solution typically uses the ℓ_1-norm instead of the ℓ_0-norm. The ℓ_1-norm is defined as

$$\|\boldsymbol{x}\|_1 = \sum_{i=1}^{n} |x_i| \tag{2}$$

where n is the dimension of $\boldsymbol{x}$.

The ℓ_1-norm provides computational advantages because of its convex nature. The use of (1) leads to constrained regression; thus, it is expressed as follows:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \boldsymbol{y} - \boldsymbol{D} \cdot \boldsymbol{x}_2^2 + \lambda \boldsymbol{x}_1 \tag{3}$$

where λ is the coefficient of regularization used to balance the signal reconstruction error and the degree of sparsity. It is often called a lasso in the literature.

C. LogDet Divergence

Divergence is a measure of dissimilarity between variables, e.g., the Euclidean distance, relative entropy, and Bregman divergence. It has been widely used for approximate estimation and information classification.

The LogDet divergence $D_{ld}(\boldsymbol{X}, \boldsymbol{Y})$ is a type of loss function of the Bregman divergence; given two matrices $\boldsymbol{X} \in \mathbb{Z}_+^n$, $\boldsymbol{Y} \in \mathbb{Z}_{++}^n$, Z_+^n is an $n \times n$ symmetric positive semidefinite matrix, and Z_{++}^n is an $n \times n$ symmetric positive-definite matrix. The LogDet divergence is defined as follows [52]:

$$D_{ld}(\boldsymbol{X}, \boldsymbol{Y}) = tr\left(\boldsymbol{X}\boldsymbol{Y}^{-1}\right) - \log\det\left(\boldsymbol{X}\boldsymbol{Y}^{-1}\right) - n. \tag{4}$$

It is asymmetrical, i.e., $D_{ld}(\boldsymbol{X}, \boldsymbol{Y}) \neq D_{ld}(\boldsymbol{Y}, \boldsymbol{X})$. This definition is not the same as the distance definition in the general functional analysis and is convex only in the first argument. In the sparse representation framework, if the matrix $\boldsymbol{\Sigma}$ is a covariance matrix, and $\boldsymbol{C}$ is a given covariance sample, the objective is to approximate $\boldsymbol{\Sigma}$ by a linear sum of a series of covariance matrices. Therefore, the LogDet divergence $D_{ld}(\boldsymbol{\Sigma}, \boldsymbol{C})$ can be selected as the solution formula of the objective function in the sparse representation framework. It should be noted that the LogDet divergence has the same affine invariance as the geodesic distance; thus, for any invertible $\boldsymbol{X}$, there are

$$D_{Id}\left(\boldsymbol{X}\boldsymbol{D}_i\boldsymbol{X}^{\mathrm{T}}, \boldsymbol{X}\boldsymbol{D}_j\boldsymbol{X}^{\mathrm{T}}\right) = D_{Id}(\boldsymbol{D}_i, \boldsymbol{D}_j). \tag{5}$$

This property can be used to simplify the calculations.

D. Graphical Lasso for Inverse Covariance Estimation

The graphical lasso is an algorithm that can quickly estimate the inverse covariance matrix. It uses the ℓ_1 penalty to increase the sparsity of the inverse covariance matrix and the fast coordinate descent method to solve a single lasso problem. When the dimension of the data is high, the calculation speed is very fast [53].

Suppose there are n independent samples with a Gaussian distribution. Each sample has p dimensions with a mean u and a covariance matrix $\boldsymbol{\Sigma}$. Let $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$, and let $\boldsymbol{S}$ be the

empirical covariance matrix. The graphical lasso method is used to maximize the penalized log likelihood as follows:

$$\hat{\boldsymbol{\Theta}} = \max(\log\det\boldsymbol{\Theta} - tr(\boldsymbol{S}\boldsymbol{\Theta}) - \rho\|\boldsymbol{\Theta}\|_1) \tag{6}$$

where $\boldsymbol{\Theta}$ is the inverse covariance matrix, ρ denotes the penalty parameters, which can determine the sparsity of the inverse covariance [54], and $\rho > 0$.

Let $\boldsymbol{W}$ be an estimate of $\boldsymbol{\Sigma}$. As described in a previous study [55], (6) is convex and can be solved by optimizing each row and the corresponding column of $\boldsymbol{W}$ using the block coordinate descent method. $\boldsymbol{W}$, $\boldsymbol{S}$, and $\boldsymbol{\Theta}$ are partitioned as follows:

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{12}^{\mathrm{T}} & \omega_{22} \end{pmatrix}, \boldsymbol{S} = \begin{pmatrix} \boldsymbol{S}_{11} & \boldsymbol{s}_{12} \\ \boldsymbol{s}_{12}^{T} & s_{22} \end{pmatrix}, \boldsymbol{\Theta} = \begin{pmatrix} \boldsymbol{\Theta}_{11} & \theta_{12} \\ \theta_{12}^{T} & \theta_{22} \end{pmatrix} \tag{7}$$

where $\boldsymbol{W}_{11}$, $\boldsymbol{S}_{11}$, and $\boldsymbol{\Theta}_{11}$ are matrices with $p-1 \times p-1$ dimensions, $\boldsymbol{\omega}_{12}$, $\boldsymbol{s}_{12}$, and $\boldsymbol{\theta}_{12}$ are $(p-1)$-dimensional column vectors, and ω_{22}, s_{22}, and θ_{22} are scalars.

$\boldsymbol{\omega}_{12}$ satisfies the following equation:

$$\hat{\boldsymbol{\omega}}_{12} = \arg\min_{\boldsymbol{\omega}_{12}} \left\{ \boldsymbol{\omega}_{12}^{\mathrm{T}} \boldsymbol{W}_{11}^{-1} \boldsymbol{\omega}_{12} : \|\boldsymbol{\omega}_{12} - \boldsymbol{s}_{12}\|_{\infty} \le \rho \right\}. \tag{8}$$

Equation (8) is a box-constrained quadratic programming problem, which can be solved by the interior point method. By exchanging rows and columns, each column is equivalent to solving (8), and the estimate of $\boldsymbol{W}$ is updated after each phase. This process is repeated and updating occurs until $\boldsymbol{W}$ converges.

By using convex duality, solving (8) is equivalent to solving the following dual problem:

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2} \left\| \boldsymbol{W}_{11}^{1/2} \boldsymbol{\beta} - \boldsymbol{W}_{11}^{-1/2} \boldsymbol{s}_{12} \right\|^2 + \rho\|\boldsymbol{\beta}\|_1 \right\} \tag{9}$$

where $\boldsymbol{\omega}_{12} = \boldsymbol{W}_{11}\boldsymbol{\beta}$. If β can solve (9), (8) can be solved by solving $\boldsymbol{\omega}_{12} = \boldsymbol{W}_{11}\boldsymbol{\beta}$. $\hat{\boldsymbol{\beta}}$ is a sparse vector. The speed will be extremely fast when calculating $\boldsymbol{\omega}_{12} = \boldsymbol{W}_{11}\hat{\boldsymbol{\beta}}$. Equation (9) is similar to (3).

Equations (6) and (9) are equivalent (see the Appendix for the derivation process) and are similar to a lasso regression problem; thus, they can be solved using the coordinate descent method [53].

The solution of β is as follows. Let $\boldsymbol{V} = \boldsymbol{W}_{11}$ and $\boldsymbol{l} = \boldsymbol{s}_{12}$. The update is performed using a soft threshold form as follows:

$$\hat{\beta}_j = \Phi\left(l_j - \sum_{k+j} V_{jk}\hat{\beta}_k, \rho \right) \Big/ V_{ij} \tag{10}$$

where $j = 1, 2, \ldots, p, 1, 2, \ldots, p, \ldots$until it converges, and Φ is the soft threshold operator. When the average absolute change in $\boldsymbol{W}$ is less than $t \cdot \text{ave}\left|\boldsymbol{S}^{-\text{diag}}\right|$, the algorithm will stop updating. Here, $\boldsymbol{S}^{-\text{diag}}$ is the off-diagonal element of the covariance matrix $\boldsymbol{S}$, and t is a fixed threshold. In this work, $t = 0.001$.

Based on the preceding derivation, the inverse covariance matrix $\hat{\boldsymbol{\Theta}}$ can be quickly found. The algorithm of the inverse covariance estimation is presented in Algorithm 1.

Algorithm 1 Inverse Covariance Matrix Estimation

Input: Observation sample matrix and covariance matrix $\boldsymbol{S}$
Output: Estimated covariance matrix $\hat{\boldsymbol{\Theta}}$
Initialization: $\boldsymbol{W} = \boldsymbol{S} + \rho\boldsymbol{I}$
For $j = 1, 2, \ldots, p, 1, 2, \ldots, p, \ldots$**do**
  Split $\boldsymbol{W}$ and $\boldsymbol{S}$ into blocks
  Solve Eq. (9) using the fast coordinate descent method, obtain the $(\boldsymbol{p-1})$-dimensional vector solution $\hat{\boldsymbol{\beta}}$, and store it in the $\boldsymbol{j}$-th column of matrix $\hat{\mathbf{B}}$
  Update $\boldsymbol{\omega}_{12} = \boldsymbol{W}_{11}\hat{\boldsymbol{\beta}}$ and $\boldsymbol{W}$ until convergence
End for
Output covariance matrix $\boldsymbol{W}$
Calculate $\hat{\boldsymbol{\Theta}}$ using $\hat{\mathbf{B}}$ and $\boldsymbol{W}$, for $j = 1, 2, \ldots, p$, $\hat{\boldsymbol{\beta}}$ is the element of column j in matrix $\hat{\mathbf{B}}$, and calculate $\hat{\boldsymbol{\theta}}_{12} = -\hat{\boldsymbol{\beta}}\theta_{22}$ and $\hat{\theta}_{22} = 1\big/(\omega_{22} - \boldsymbol{\omega}_{12}^{T}\hat{\boldsymbol{\beta}})$, respectively
Output estimated inverse covariance matrix $\hat{\boldsymbol{\Theta}}$

Algorithm 2 Learning the LDbSICM for Prediction

Input training samples of different classes
Find the mean u of each class
Find the inverse covariance $\hat{\boldsymbol{\Theta}}$ of each class using the graphical lasso (Algorithm 1)
Save u and $\hat{\boldsymbol{\Theta}}$
For i =1 to Total training times **do**
  Use u and $\hat{\boldsymbol{\Theta}}$ and the LogDet divergence to classify the training set
  Compare the classification results and perform further training to iteratively update $\hat{\boldsymbol{\Theta}}$
  Find the optimal $\hat{\boldsymbol{\Theta}}$ with the highest classification accuracy of the training samples
End for
Input the observation k
Calculate the log-likelihood value of each class corresponding to observation k
Choose the maximum log-likelihood value, which is the prediction result corresponding to observation k

The proposed driving behavior prediction method for learning the LDbSICM for prediction is presented in Algorithm 2. Before training the model, we need to smooth the data. Therefore, most of the computational cost of the proposed model is attributed to solving the convex optimization problem (9). According to $\boldsymbol{W} \in \mathbb{Z}^{n\times n}$, the computational complexity of solving (9) is $O(n^3)$.

III. Experiment and Data Collection

A driving scene highly similar to a real intersection in a connected traffic environment was designed using a high-fidelity real-time simulator, V2I communication software, and an HMI to collect the data and ensure the safety of the drivers. The experimental setup and data collection method are detailed in this section.

A. Apparatus and Participants

Fig. 4 displays the driving simulation platform used in this work. The simulator simulated real traffic scenes and the yaw,

Fig. 4. Driving simulator and components.

roll, and pitch motions of the vehicle using a six-degree-of-freedom (DOF) motion platform. Specifically, it also included the following experimental equipment: a real vehicle cabin, a front-view curved screen display system, a two-channel rearview display system to monitor traffic through a rear-view mirror, an E2M 6-DOF motion platform, a control system to collect data, a high-performance industrial computer, and a scene projection system containing five projectors. The data acquisition system collected the yaw angle, yaw rate, lateral and longitudinal speed, lateral and longitudinal acceleration, and vehicle speed and trajectory. Moreover, the steering wheel torque, steering wheel angle, steering wheel angular rate, brake pedal pressure and position, and throttle opening were obtained via the CAN bus.

The driver's observation actions included head rotation and eye gaze. The SmartEye 8.0 eye-tracking system was employed and included four eye motion-tracking cameras and three infrared light sources, as shown in Fig. 5. The data recorded by the eye-tracking system included information on the driver's head rotation (Rodrigues rotation), head

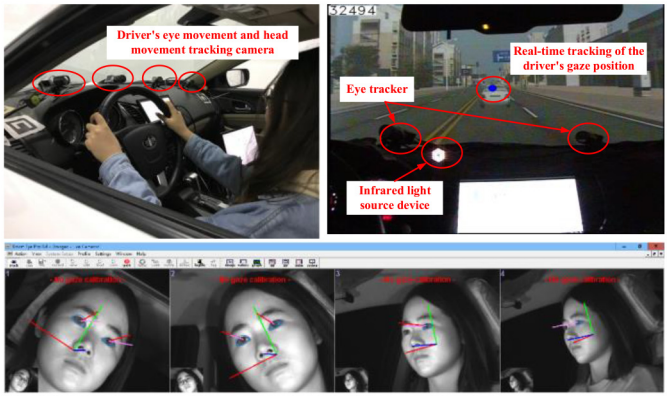

Fig. 5. Eye-tracking system.

orientation, eye fixation point position, and eye orientation. The sampling frequency of the eye tracker was 60 Hz. It should be noted that the sampling frequencies of the simulator, CAN bus, and eye tracker were synchronized to 60 Hz during data collection, and Gaussian smoothing was used to minimize noise interference [9].

A total of 35 experimental participants (26 male and 9 female participants) from different backgrounds were recruited for the experiment. The participants were all healthy and had good eyesight. They were between 24 and 50 years of age, with an average age of 38.4 years. They all held a valid Chinese driving license and had an average driving experience of 16.3 (SD 10.6) years.

B. Experimental Scenario Design and Experimental Process

A layout consistent with a real urban scene was designed in the driving simulator for a realistic driving environment. It included multiple signalized intersections. Each part of the intersection was a two-way, four-lane road with a lane width of 3.75 m. The centerline of the road was a double yellow solid line, the centerline of each lane was a white dashed line, and the edge lines were solid white lines. Fig. 6 illustrates one of the intersection scenes. The upper graphic is a top view of the urban scene, the bottom-left graphic is the driver's perspective when the vehicle arrived at one of the intersections, and the bottom-right graphic is the schematic of a vehicle entering an intersection in the V2I mode.

A channel for outputting the traffic information from the driving simulator to the HMI was created to enable drivers to obtain infrastructure information beyond the visual range. The HMI display platform was designed based on the Python language and was installed on the right side of the dashboard. As shown in Fig. 7, it could display four types of information in the connection mode: 1) the current traffic light status of the intersection; 2) the real-time traffic light timing; 3) navigation function information; and 4) the real-time distance between the vehicle and the intersection.

Before conducting the experiment, the participants were trained the V2I system. In addition, considering the difference between the driving simulator and actual driving, each driver

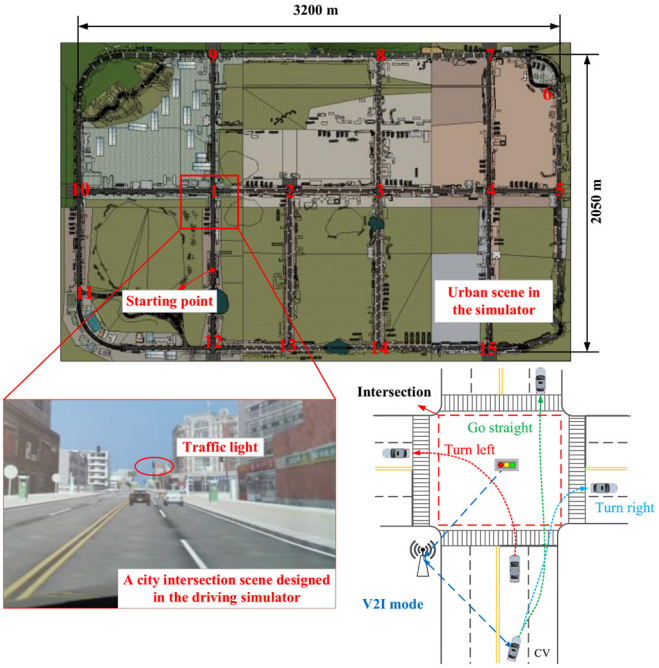

Fig. 6. Intersection scene.

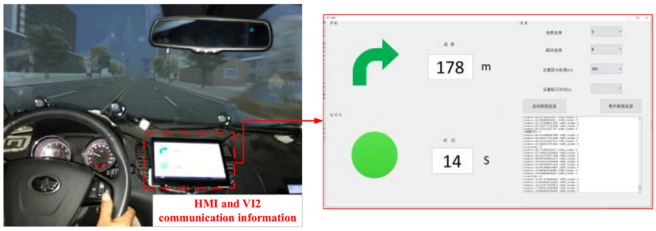

Fig. 7. HMI display.

was required to perform a 20-min test drive before the experiment to become accustomed to the devices and operation. Each participant performed the following driving tasks.

1) *Traditional Drive (Baseline Group):* No connection information was provided, and the driver could only perceive the surrounding traffic vehicles and infrastructure information via vision and hearing, which is consistent with traditional driving. The test was performed three times, and the paths were 1-2-3-14-15-4-7-8-3-2-1, 1-9-10-1-12-13-14-3-8-7-4, and 1-10-9-8-7-4-15-14-3-2-1, as shown in Fig. 6.
2) *Connected Drive (Treatment Group):* The driving process was carried out in a connected environment, and the driver obtained infrastructure information via the HMI. When the vehicle was 300 m from the intersection, the HMI function was activated, and the connection information was displayed in real time. The driver only received a voice prompt once the HMI was activated to avoid interference with the driving decisions. This test was also performed three times, and the driving paths were the same as those in the traditional drive.

The order of the traditional and connected drives was different for each participant to prevent anticipation. Some drivers performed the traditional drive first, followed by the

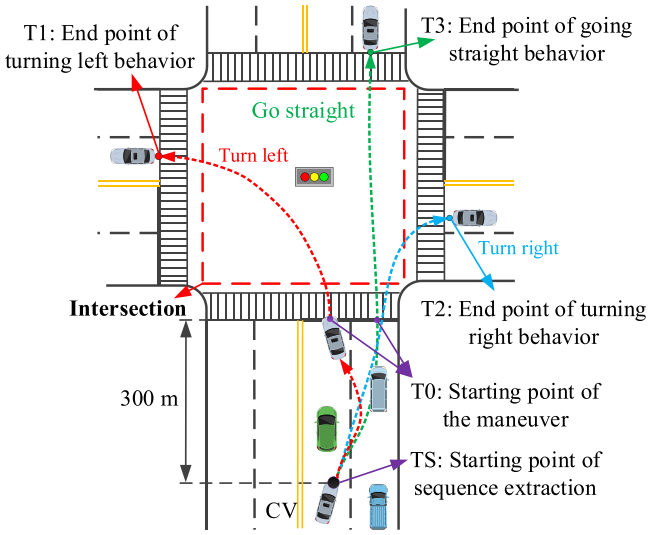

Fig. 8. Illustration of turning maneuvers at an intersection.

connected drive, whereas others did the opposite. There was no communication delay in the connection channel, and the connection information was immediately received when the vehicle arrived at the designated location [38]. Therefore, the length of the drive for each analyzed intersection was approximately 300 m before arriving at the intersection. It should be noted that the activation of the VI2 communication 300 m before the intersection was determined in accordance with the requirements of the People's Republic of China's "Specification for Layout of Urban Road Traffic Signs and Markings" and "Specification for Layout of Highway Traffic Signs and Markings" to implement warning signs at level intersections [56], [57]. In this work, the influences of the temporal and spatial characteristics of the connection information on the driver's decision and behavior were not considered; the possible maneuvers were predicted based solely on the cues of the driver and vehicle. Therefore, the vehicle entering the intersection may have been accelerating, decelerating, or passing at a constant speed, or it may have been stopped.

The traffic density was moderate; as shown in Fig. 8, three vehicles were moving around the subject CV. The drivers were told to drive the vehicle according to their driving style and habits, observe the rearview mirror before changing lanes, pay attention to traffic safety, and avoid collisions with other vehicles. The road in the experimental scene was flat, the tire-road surface adhesion was good (the adhesion coefficient was 0.8), and the weather was sunny to ensure sufficient light for driving.

The experiment lasted two weeks, and a total of 34 drivers completed the experiment. Totals of 600 and 558 intersection-approaching maneuvers were extracted in the connected and traditional conditions, respectively. The summary statistics of the extracted samples in the connected and traditional conditions are listed in Table I.

C. Model Construction

Before executing a lane change, the driver checks the rearview mirror to determine whether it is safe to change lanes.

TABLE I
STATISTICS OF MANEUVER SAMPLES IN THE TRADITIONAL AND CONNECTED DRIVES

	TL	TR	GS	Total
Connected	200	200	200	600
Traditional	186	186	186	558
Total	386	386	386	1158

Therefore, when researchers predict lane-changing behaviors, they often use the first or last glance before the lane-changing behavior as the key parameter [9], [58]–[60], which represents the starting point of the intent time window. However, drivers in front of an intersection tend to pay less attention to the vehicles behind them and are more focused on the safe distance from the vehicles in front of them [61], traffic lights [21], [62], or potential collisions with other road users at the intersection [63]. Although the vehicle may stop when it reaches the intersection because the traffic signal is red, it is still possible to extract the preferences and steering patterns of drivers by observing their actions, the operating characteristics, and the kinematics and trajectory of the vehicle before entering the intersection [3], [64]. Therefore, the effective extraction of behavioral sequences is essential for model construction.

1) Behavior Sequence Extraction for Modeling: Fig. 8 illustrates three different maneuvers at intersections. The start (T0) and endpoints (T1, T2, and T3) of the maneuvers are defined based on the intersection structure. In Fig. 8, T0, T1, T2, and T3 are on the solid line of the crosswalk (not on the side closest to the intersection). The maneuver starts or ends when the vehicle crosses the crosswalk.

It is necessary to include the sequence before the start of the maneuver to construct a maneuver prediction model. Therefore, the location at 300 m before the start of the maneuver was used as the starting point for sequence extraction in this study (i.e., point TS in Fig. 8), and the endpoints were T1 for TL, T2 for TR, and T3 for GS.

2) Feature Selection: As much data as possible was collected because it is unclear which indicator variable has a high predictive capability for turning maneuver prediction in an intersection scene in a connected environment. Subsequently, various combinations of the connected data were input into the model, retraining was conducted, and the test results were obtained to determine which indicator was the most important for the prediction of the turning maneuvers. The characteristics to construct the model included the following four categories: 1) the driver observation parameters, which reflect the driver's lane-change intention; 2) the vehicle trajectory parameters, which reflect the driving characteristics; 3) the vehicle kinematic parameters, which reflect the driver's intention and behavior patterns; and 4) the parameters collected by the CAN bus, which reflect the driver's operating characteristics. There were 23 parameters; Table II lists all the features and their definitions. Fig. 9 illustrates an example of the changes in the heading angles of the TL, TR, and GS behaviors when the vehicles entered the intersection in the connected drive mode.

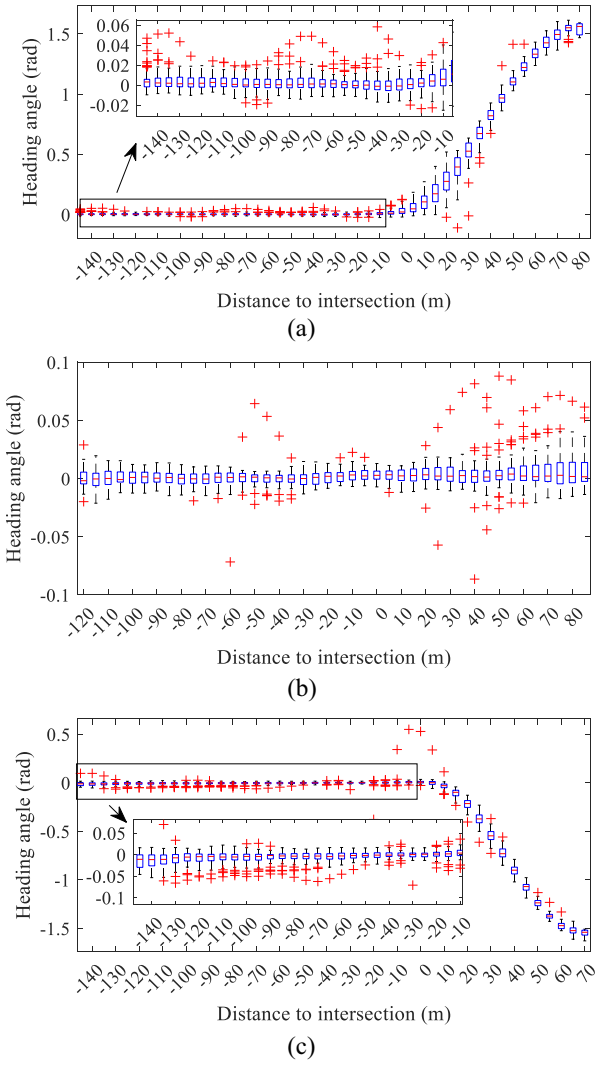

Fig. 9. Examples of the dynamics of the heading angle for the three different maneuvers in the connected drive. (a) TL. (b) TR. (c) GS.

The time series of the heading angles reveal that these parameter dynamics can reflect different maneuvering patterns when approaching an intersection.

3) Model Training: Unlike traditional models, the proposed model based on sparse representation theory has incremental characteristics, i.e., the model can separately estimate the sparse inverse covariance matrix for different turning maneuvers and incrementally create a maneuver prediction dictionary model, as described in Section II. The sparse inverse covariance matrices representing different turning maneuvers in the dictionary do not affect each other.

The maneuvers have different labels. Each data sequence is a 23 $\times$ [TS - Ti] (i = 1, 2, 3) tensor, which represents 23-D input features with a sequence length ranging from TS to Ti. The length of the sample sequence from TS to Ti was selected to determine if the model could accurately identify the maneuver behavior in the early stage of the maneuver. The training procedure was performed on a computer with a 3.2-GHz Intel Core i7 processor and 16-GB installed memory.

TABLE II
FEATURES USED FOR TRAINING

Category	Parameter number	Parameter
Driver observation parameters	1	Rodrigues' head rotation *Rx*
	2	Rodrigues' head rotation *Ry*
	3	Rodrigues' head rotation *Rz*
	4	Head heading
	5	Gaze direction *Gx* (Pitch)
	6	Gaze direction *Gy* (Yaw)
	7	Gaze direction *Gz* (Roll)
	8	Gaze heading
Trajectory parameters	9	The abscissa of the trajectory
	10	The ordinate of the trajectory
Vehicle kinematic parameters	11	Heading angle
	12	Yaw angle rate
	13	Longitudinal acceleration
	14	Lateral acceleration
	15	Longitudinal velocity
	16	Lateral velocity
	17	Vehicle speed
CAN bus parameters	18	Steering wheel torque
	19	Steering wheel angle
	20	Steering wheel angular rate
	21	Brake pedal displacement
	22	Brake pedal force
	23	Throttle opening

The fivefold cross-validation method was used to test the model. All data were normalized to prevent the influence of the variable dimensions. It should be noted that the samples in this work were time-series data; thus, the recognition of the driving maneuver represents a time-series classification. Therefore, the receiver operating characteristic (ROC) curve was used as the performance evaluation index of the proposed method. This method evaluates the performance of a classifier by assessing the true positive rate (TPR) and false positive rate (FPR). The calculation formulas of the TPR and FPR are as follows:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TN} + \mathrm{FN}}$$
$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} \tag{11}$$

where TP, TN, FP, and FN are the true positives, true negatives, false positives, and false negatives, respectively. The function provided in MATLAB was used to calculate the ROC curves of the models, and the TP, TN, FP, and FN values were calculated after obtaining the true class labels and the predicted scores of the test samples.

IV. EXPERIMENTAL RESULTS

The prediction of the possible maneuver state (i.e., the driving intention) of drivers at intersections can facilitate the dynamic management of traffic at intersections for intelligent transportation systems. Therefore, the system requires accurate results as early as possible for turning maneuver prediction based on driver behavior and vehicle kinematics. In this section, the proposed model is evaluated and compared with current state-of-the-art algorithms. All models are evaluated using the prediction accuracy and prediction distance. The prediction distance indicates the distance from the intersection

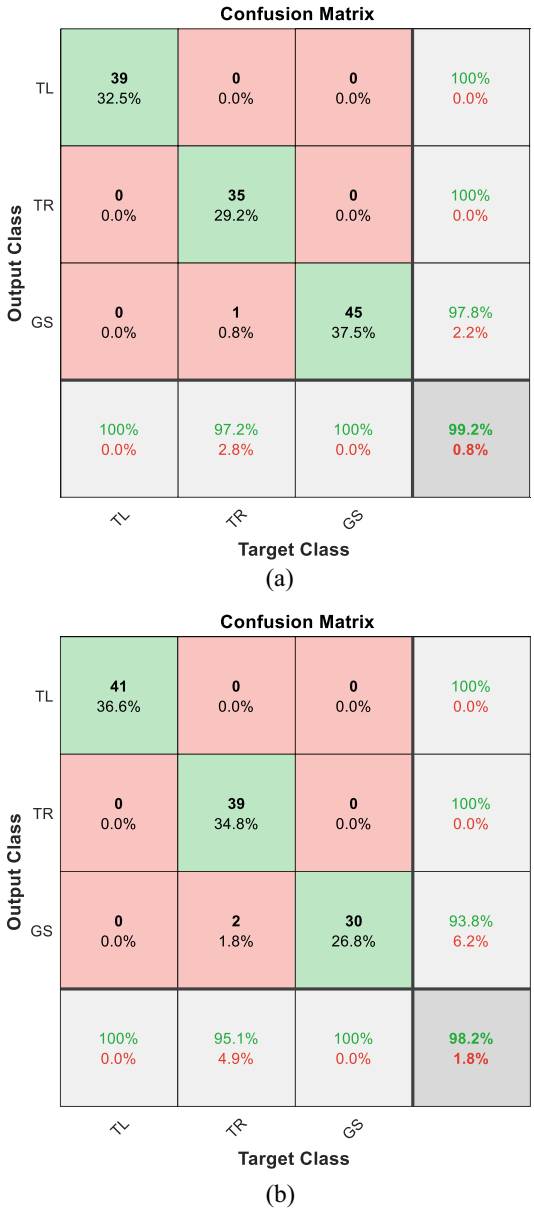

Fig. 10. Confusion matrices for (a) connected drive mode and (b) traditional drive mode.

at which the prediction model makes accurate predictions. Although this evaluation index differs from that in the literature, in which the prediction horizon was typically used to evaluate the performance of lane-change prediction models [25], [58], the two indices have similar meanings because the purpose is to determine if the model can make accurate predictions before the start of the maneuver.

A. Driving Behavior Prediction

1) Model Generalization Performance: Fig. 10 exhibits the confusion matrix of the prediction of the turning maneuvers in the two driving modes. Because the input of the model is a segment of the driving sequence, it can be seen from the confusion matrix result that the proposed model exhibited good performance for the prediction of intersection maneuvers in

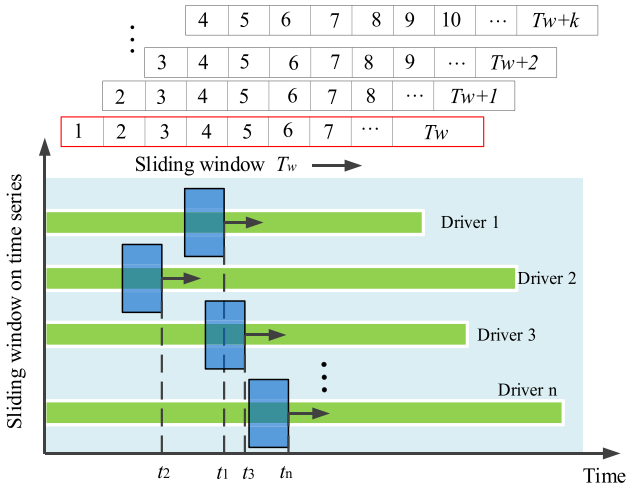

Fig. 11. Time-series identification of driving behavior using a sliding time window.

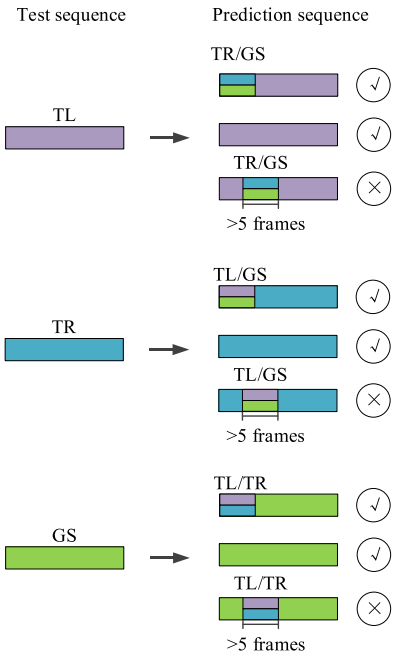

Fig. 12. Rules for evaluating the accuracy of a test sequence prediction result.

different driving modes. The overall accuracies in the connected and traditional drive modes were, respectively, 99.2% and 98.2%. Specifically, the model achieved 100% prediction accuracy for the TL and GS behaviors in both driving modes and 97.2% and 95.1% for the TR behavior in the connected and traditional drive modes, respectively. One TR maneuver in the connected drive mode and two TR maneuvers in the traditional drive mode were incorrectly predicted as GS. The reason may be that the driver did not exhibit clear steering intentions before the start of the TR maneuver, although the driver knew that they were about to make a right turn (i.e., there were no apparent observation actions, no significant acceleration and deceleration characteristics, etc.). Thus, the model was unable to make accurate predictions based on the input sequence. Nevertheless, the predictive ability of the model was very satisfactory. The predictive ability of the model is crucial, i.e., the model must predict the maneuvering intention of the driver as early as possible. Therefore, the index of the prediction distance (i.e., the distance of the vehicle from the intersection at which the model can achieve an accurate prediction) was used to evaluate the model.

2) Statistics of the Prediction Distance: A sliding time window was used to obtain the position of the vehicle when a maneuver was identified. The schematic of the sliding time window is shown in Fig. 11. The sliding time window slid forward on the time series of the driving maneuver, and an observation sample was obtained for each step. The sample was evaluated until the correct recognition result was obtained. Algorithm 3 is the pseudocode of the sliding time window algorithm. It should be noted that all test sequences used for the time-lapse analysis needed to be correctly identified. The rules for the correct prediction of the samples used in a previous study [65] were adopted in this work, as shown in Fig. 12. For a test sequence of TL (correspondingly TR), if an identification result in frame i (i is the arbitrary position of the sequence) was TR or GS (correspondingly TL or GS), and the result remained the same for more than five frames, the prediction was considered incorrect. If an identification

Algorithm 3 Sliding Time Window Algorithm

1: Input the time series sample **R** for testing
2: Calculate the length of **R**, *lenR*
3: Define the initial position *ip* of the sliding time window
4: Define the moving step *ms* of the sliding time window
5: Define the size of the sliding time window, *Tw*
6: **For** i = *ip* to *lenR* - *Tw* + 1 **do**
7: $\mathbf{R}_i$ = **R** (i to i + *Tw* -1) #
8: *ip* = *ip* + *ms*
9: return $\mathbf{R}_i$ for identification
10: **End for**

MATLAB syntax.

result was TR or GS (correspondingly TL or GS) at the starting point of the sequence, the prediction continued. The same rules were applied to the prediction of GS.

The initial position of the sliding time window was the beginning of the time sequence, and the moving step *ms* was 1 frame (i.e., the sliding window moved forward frame by frame). The size of the sliding time window ranged from 0.5 to 2 s to obtain the maximum prediction accuracy; this range was obtained according to previous studies [32], [65], [66], and 1.5 s was selected after the evaluation using the fivefold cross-validation method.

Fig. 13 presents the statistical distribution of the prediction distances in the connected and traditional driving environments. In the connected driving environment [Fig. 13(a)], more than 80% of TL maneuvers were predicted at 270–280 m from the intersection. Similarly, the TR maneuvers were also correctly identified at 280–290 m from the intersection. More than 80% of the GS maneuvers were identified at the beginning of the sequence. In contrast, some differences are evident in Fig. 13(b). Although the prediction was achieved before

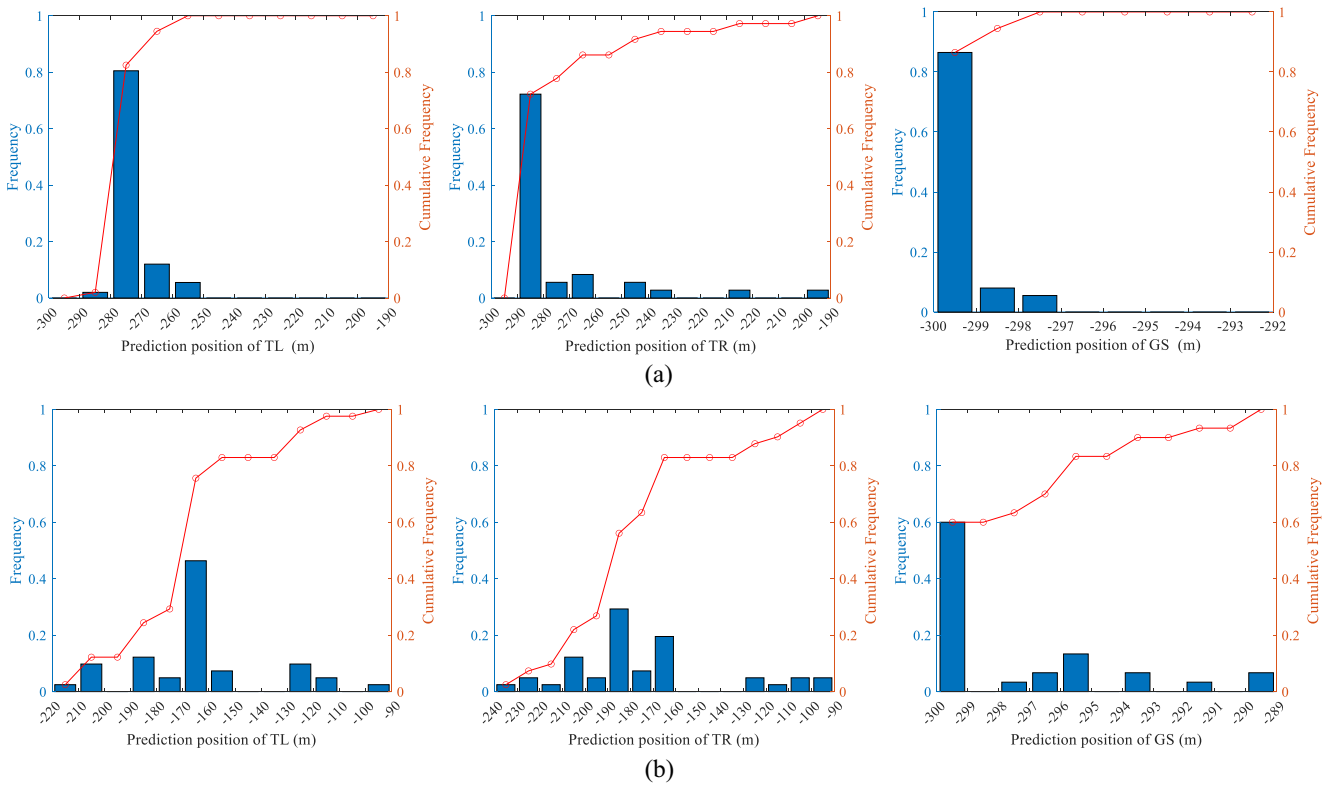

Fig. 13. Statistical distributions of the distance to the intersection of the correct recognition results for different maneuvers in (a) treatment group and (b) baseline group.

the start of the maneuver in the traditional environment, the prediction distances for the three maneuvers lagged behind those in the connected environment. Specifically, in the traditional drive mode, the prediction positions of TL were mostly concentrated in the range of 160–190 m from the intersection, and those of TR occurred in the range of 160–200 m from the intersection. These results demonstrate that the connected driving environment changed the driving behavior; thus, the maneuvering process resulted in distinguishable characteristics to enable the model to achieve early prediction of the turning maneuvers. Early prediction is required for traffic management using intelligent transportation systems.

3) Performance Comparison With Existing Methods: The prediction accuracy of the proposed model was further compared with those of existing methods to demonstrate the superiority of the model. Fivefold cross-validation was used to evaluate each model based on the same experimental data. The following methods were evaluated.

1) The HMM is often used to recognize lane change and turning maneuvers [37], [50], [67]. However, the HMM has achieved a lower performance than the HMM with Bayesian filtering (HMM-BF) [65], [66]. Therefore, HMM-BF was used as one of the benchmark algorithms in the present study. It should be noted that the input of the HMM is a time series, and the output result is a maneuvering state (i.e., a scalar). Thus, BF was used to minimize the fluctuations in the preliminary classification results to improve the prediction accuracy.
2) Several other supervised learning approaches, such as the SVM [32], KNN [68], and RF [38] models were used as benchmark algorithms for comparison. They were implemented in the Statistics and Machine Learning Toolbox provided by MATLAB. It should be noted that the input data forms of the SVM, KNN, and RF algorithms are vectors. The method described in a previous study [69] was adopted to concatenate the observed variables in the time window of each step to obtain the input feature vector of the model. This method only changes the form of the data and does not affect the maneuvering characteristics.
3) In addition, two novel deep learning models based on the RNN [30], i.e., the bidirectional LSTM (Bi-LSTM) [70] and LSTM-RNN [58], have achieved good maneuver prediction performance. These two approaches were also adopted as benchmark models for comparison with the proposed method. The LSTM-RNN model was trained using the MATLAB Deep Learning Toolbox with the Adam optimizer, an initial learning rate of 0.001, and a gradient decay rate of 0.9. The maximum number of epochs was 150, and the mini-batch size was 32. The LSTM layer had 120 hidden units, followed by two fully connected (FC) layers with 100 neurons and an output layer with three neurons.
4) As a very competitive deep learning method, generative adversarial networks (GANs) have shown excellent performance for multivariate time-series

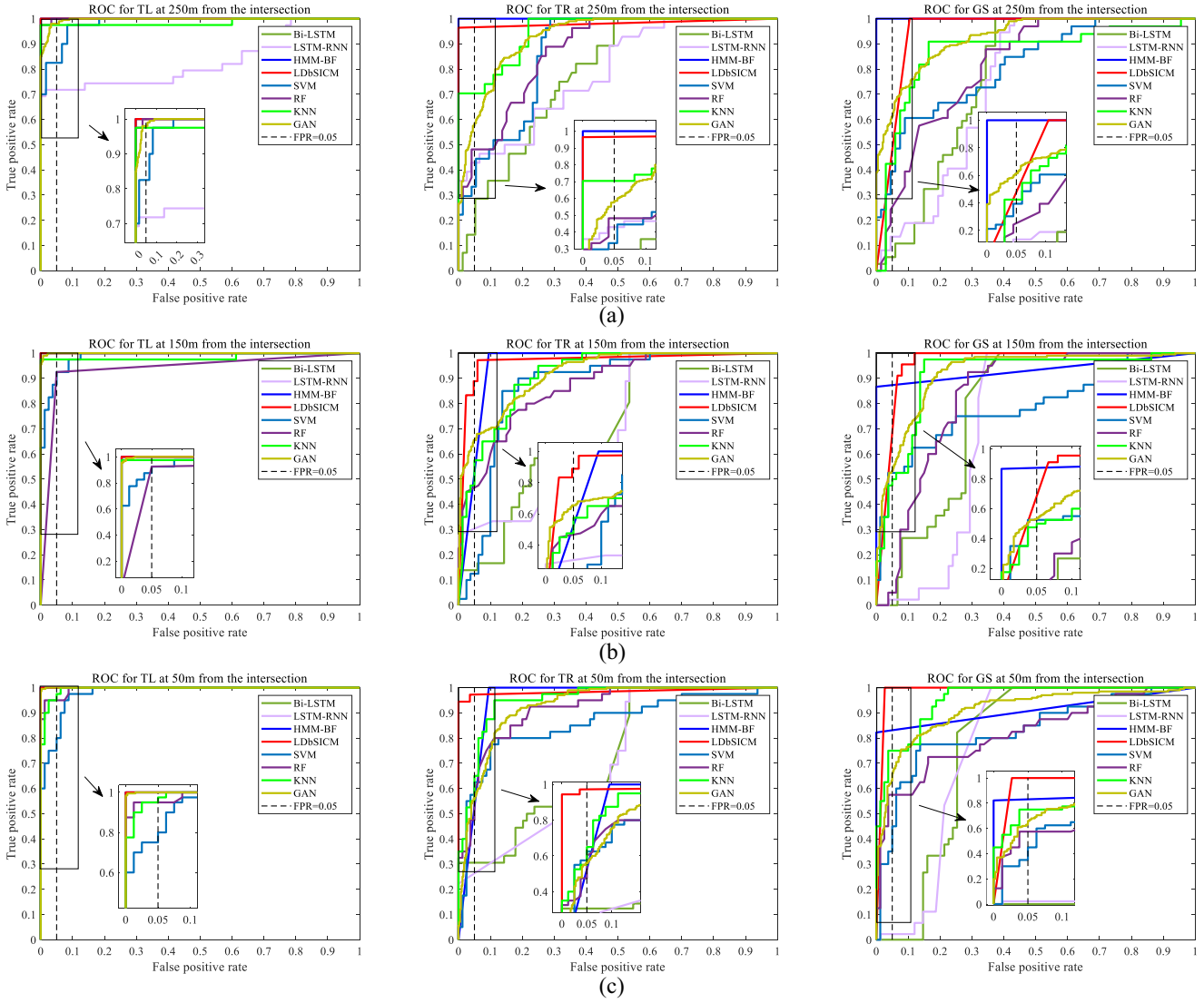

Fig. 14. ROC curves for the prediction of TL, TR, and GS at distances of (a) 250 m, (b) 150 m, and (c) 50 m from the intersection. The dashed line represents the TPR at an FPR of 0.05.

classification [71]. Therefore, this method was used as a benchmark model. It was trained with the MATLAB Deep Learning Toolbox. Refer to [71] for the parameter definitions of the generator and adversarial networks. Multivariate time series were transformed into images by measuring the similarities between the variables for each time step [72].

Fig. 14 presents the ROC curves for maneuver prediction (i.e., TL, TR, and GS) at distances from the intersection of 250, 150, and 50 m. The prediction results indicate the following.

1) The proposed LDbSICM model achieved the best recognition results. Although the performance of HMM-BF was slightly better than that of LDbSICM at 250 m, the prediction results of LDbSICM were better at the other two distances. Overall, the prediction performances of the Bi-LSTM and LSTM-RNN models were not good, and these models could not accurately predict the maneuvers at intersections, especially TR and GS. These findings indicate that although the RNN models achieved good performance for modeling time series, they could not accurately predict the turning maneuver when the vehicle was far from the intersection. The performance of GAN was generally better than that of the RNN-based model. It achieved an average prediction accuracy of more than 80% at three distances; however, its performance was not as good as the LDbSICM model.
2) All models achieved good prediction performances for TL, but the predictions for TR and GS were not good, except for those of LDbSICM and HMM-BF. This result demonstrates the superiority of the LDbSICM model for detecting the hidden state in the data and indicates that the TR and GS maneuvers are often difficult to distinguish before the start of the maneuvers.
3) The similarity of the prediction accuracies at the three distances indicates that predicting turning maneuvers at an intersection is not the same as predicting lane-change intentions. In the latter, the closer to the start of the maneuvering behavior, the higher the prediction

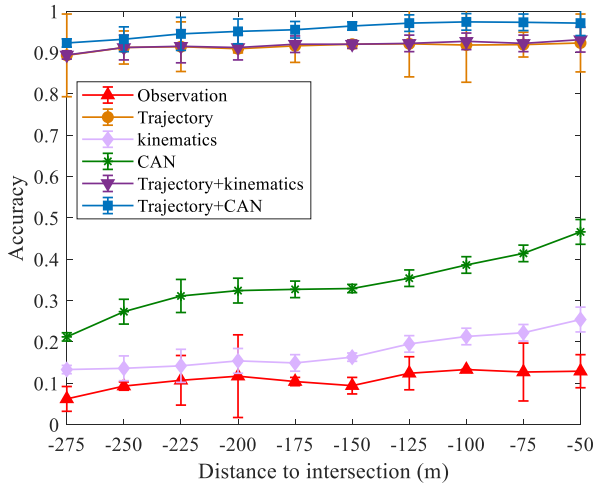

Fig. 15. Relationship between the prediction accuracy and the distance to the intersection.

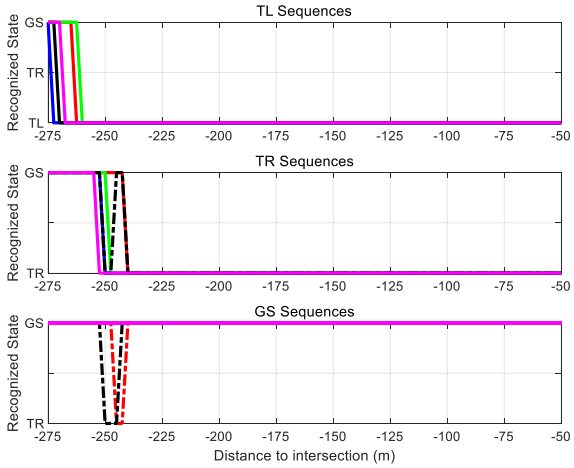

Fig. 16. Prediction results for 15 different maneuvers corresponding to 5 TL, 5 TR, and 5 GS sequences.

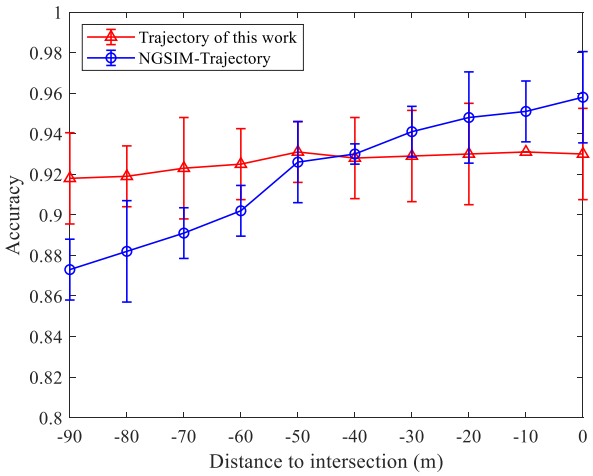

Fig. 17. Relationship between the prediction accuracy and the distance to the intersection using the trajectory in this work and the NGSIM trajectory.

accuracy is, as described in a previous study [58]. When predicting a turning maneuver, the prediction accuracy of the LDbSICM and the other benchmark models did not increase as the vehicle approached the intersection. This result is similar to that obtained in a previous study [73].

4) The other benchmark models exhibited no significant change in prediction accuracy as the distance to the intersection decreased, indicating that the driver's driving pattern (trajectory feature) may be highly useful as an indicator of maneuver prediction because the vehicle showed a distinct deceleration trend when approaching an intersection. The specific behavior characteristics (observation actions, operating behaviors, etc.) before approaching an intersection may not be the best indicators for behavior prediction.

Based on these, the variable importance for turning maneuver prediction is evaluated in the subsequent section.

B. Variable Importance for Maneuver Prediction

This section analyzes the variable importance in the predictive model. All parameters were divided into four categories according to Table II, and the model was retrained to determine which types of parameters were the most important for maneuver prediction.

Fig. 15 presents the impacts of different inputs and their combinations on the prediction performance. The test sequence was shifted back for every 25 m. The "observation" lines indicate that the model was trained without using the vehicle-related features (trajectory, kinematic, and CAN bus data). The "trajectory," "kinematics," and "CAN" lines also have similar meanings. "Trajectory + kinematics" represent that the model was trained using the trajectory and kinematic features, and "Trajectory + CAN" has a similar meaning. It can be seen that it was impossible to obtain accurate maneuver prediction results by using only the driver's observation behavior, vehicle kinematic characteristics, or CAN bus data. However, when the trajectory information was used to train the model, improved prediction performance was achieved. The combination of trajectory and CAN features achieved the highest prediction accuracy. This result demonstrates that trajectory information is an effective indicator of turning patterns before the vehicle begins to maneuver. The CAN bus parameters reflect the operating behavior of the driver, which reflects the driving characteristics and preferences and is crucial for predicting turning maneuvers. Fig. 16 shows the prediction results for five randomly selected sequences (5 TL, 5 TR, and 5 GS samples). The model achieved good prediction performance. Accurate prediction results were obtained quickly for TL but more slowly for TR and GS. After obtaining the correct prediction results, the model maintained a stable prediction performance.

We used next-generation simulation (NGSIM) data for testing [74] to demonstrate the performance of the proposed model using only trajectory information. TL, TR, and GS samples were obtained using the extraction criteria in our previous work [75]. It should be noted that the total length of the four intersections in the NGSIM data of Lankershim Boulevard is 484.02 m (1588 feet). Thus, we conducted the

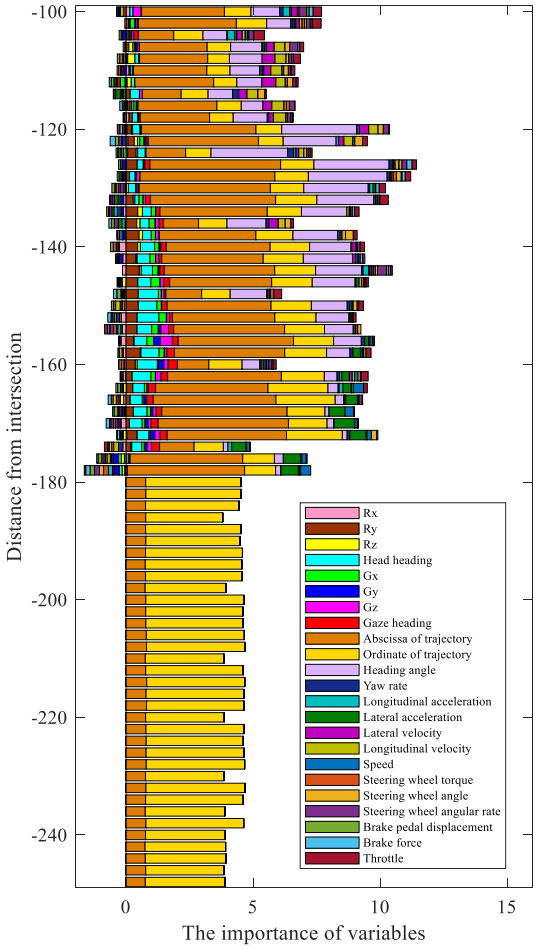

Fig. 18. Variable importance at distances of 250–100 m from the intersection in the connected driving mode.

test 100 m from the intersection [73]. The test sequence length was 10 m. Fig. 17 shows the results of testing the proposed model using our trajectory data and the NGSIM trajectory data. The proposed model also achieved relatively good results using the NGSIM trajectory data. The accuracy was higher for the NGSIM trajectory closer to the intersection, especially within 50 m. The likely reason is that Lankershim Boulevard has many dedicated lanes, which enables more accurate predictions.

The RF algorithm was used to construct a prediction model (a bagged ensemble of 200 regression trees) and estimate the predictor importance values by permuting out-of-bag (OOB) observations among the trees. For the details of the RF algorithm, refer to [38] and [76]. It should be noted that the data required normalization to obtain a consistent data range. Fig. 18 shows the variable importance at distances of 250–100 m from the intersection in the connected environment. The numbers on the abscissa from 1 to 23 refer to the 23 variables listed in Table II.

Fig. 18 reveals that trajectory information was the most important predictor variable at a distance of less than 180 m from the intersection, indicating that the vehicle trajectory can reflect the maneuvering intention when predicting long-term turning maneuvers. These results are similar to those of a previous study [3], in which the trajectory feature of the vehicle at a "high level" was a spatiotemporal process reflecting the driver's long-term behavior information, such as driving preferences and turning patterns. After entering the 180-m range, the importance of the other variables increased, although the importance of the trajectory information remained high. In addition, the importance of the heading angle increased as the vehicle approached the intersection.

The driver's observation behavior had relatively low importance in the predictive model. However, this does not mean that the driver's observation actions in the connected environment are not important. The reason for the relatively high maneuver prediction in the connected environment is that the driver interacts with the HMI and produces a response, which is reflected in the operating behavior.

V. Conclusion

In this research, maneuvers at signalized intersections in a connected environment were predicted using data related to the driver's observations, vehicle trajectory, kinematic characteristics, and driving before the vehicle entered the intersection. Moreover, the variable importance of the proposed model was analyzed.

A connected driving environment in a driving simulator was designed to collect experimental data. The driving data of 34 skilled drivers were collected and analyzed. Subsequently, a novel dictionary learning-based maneuver prediction model, i.e., a dictionary model, was proposed by estimating and learning the sparse inverse covariance matrix based on the LogDet divergence of different turning maneuvers. Existing models do not use incremental learning to update the model. In contrast, the proposed model is characterized by incremental learning and scalability, i.e., the sparse inverse covariance matrix can be updated without updating the entire dictionary. The experimental results demonstrated that the proposed model exhibited excellent predictive ability for turning maneuvers in both the connected and traditional environments. In the connected driving environment, the prediction accuracy of TL and GS was 100%, and that of TR was 97.2%. The ROC curves demonstrated the superiority of the proposed model over existing models. Another advantage of the proposed model is its ability to predict the driving behavior at a sufficient distance before entering the intersection. The statistical distribution results of the prediction distance indicated that the proposed model provided a rapid prediction of the maneuvers in a connected environment. Finally, it was found that the trajectory and heading angle of the vehicle were the most important variables in the RF-based prediction model. The trajectory data were the most important information in a connected environment when the distance between the vehicle and the intersection exceeded 180 m. In addition, the driver's operating behavior should also be included in the predictive model.

A. Limitations

Drivers in a connected environment should be able to receive sufficient beyond-the-horizon traffic or advisory information via V2I communication equipment. This

information, such as speed recommendations and information on pedestrians, cyclists, and other road users, cannot be obtained in traditional traffic environments. In one of the connected driving scenarios, the driver received the real-time status of the traffic light and the remaining time of the current status. Due to the complexity of the connected driving environment, it is impossible to simulate this environment accurately. Therefore, it is reasonable to consider only one connected driving scenario. The driving behavior in a specific scene can also reflect the characteristics of the driving behavior in a connected environment because the driver must frequently interact with the HMI to improve their driving behavior. In addition, the reason for setting the connection information prompt at a distance of 300 m from the intersection was the lack of relevant standards and reports for establishing a connected traffic environment. Researchers utilize different experimental variables and conditions based on their understanding of the connected driving environment.

B. Future Work

The driver's interaction with the HMI and the response to connection information will be investigated in future work. In addition, drivers may focus more on the HMI and less on real-time traffic conditions in a connected driving environment; thus, this impact on car-following requires evaluation.

APPENDIX

It can be demonstrated that (6) is equal to (9). From $\boldsymbol{W} = \boldsymbol{\Sigma}$ and $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$, $\boldsymbol{W\Theta}=\mathrm{I}$ can be obtained as the following expression:

$$\begin{pmatrix} \boldsymbol{W}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{12}^{\mathrm{T}} & \omega_{22} \end{pmatrix}\begin{pmatrix} \boldsymbol{\Theta}_{11} & \theta_{12} \\ \theta_{12}^{T} & \theta_{22} \end{pmatrix}=\begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0}^{T} & 1 \end{pmatrix}. \tag{12}$$

According to (10), the following formulas can be obtained:

$$\boldsymbol{W}_{11}\boldsymbol{\theta}_{12} + \boldsymbol{\omega}_{12}\theta_{22} = 0 \tag{13}$$

$$\boldsymbol{\omega}_{12}^{\mathrm{T}}\boldsymbol{\theta}_{12} + \omega_{22}\theta_{22} = 1. \tag{14}$$

The subgradient equation that maximizes the log likelihood (6) is defined as follows:

$$\boldsymbol{W} - \boldsymbol{S} - \rho \cdot \Gamma = 0 \tag{15}$$

where the derivative of $\log\det\boldsymbol{\Theta}$ is $\boldsymbol{\Theta}^{-1}=\boldsymbol{W}$, the derivative of $tr(\boldsymbol{S\Theta})$ to $\boldsymbol{\Theta}$ is $\boldsymbol{S}$, and the subgradient of $|\boldsymbol{\Theta}|$ is Γ. Here, Γ is a $(p \times p)$-dimensional matrix with $\Gamma_{ij} \in \ \mathrm{sign}(\Theta_{ij})$, in which $\Gamma_{ij} = \mathrm{sign}(\Theta_{ij})$ if $\Theta_{ij} \neq 0$; otherwise, $\Gamma_{ij} \in [-1, 1]$ if $\Theta_{ij} = 0$. Moreover, Θ_{ij} refers to the element in row i and column j in $\boldsymbol{\Theta}$, and Γ_{ij} refers to the element in row i and column j in Γ.

Equation (12) is expanded according to the matrix block form as follows:

$$\begin{pmatrix} \boldsymbol{W}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{12}^{\mathrm{T}} & \omega_{22} \end{pmatrix} - \begin{pmatrix} \boldsymbol{S}_{11} & \boldsymbol{s}_{12} \\ \boldsymbol{s}_{12}^{\mathrm{T}} & s_{22} \end{pmatrix} - \rho\begin{pmatrix} \Gamma_{11} & \gamma_{12} \\ \gamma_{12}^{\mathrm{T}} & \gamma_{22} \end{pmatrix} = 0 \tag{16}$$

where $\Gamma_{11} \in \ \mathrm{sign}(\boldsymbol{\Theta}_{11})$ is a matrix of $p-1\times p-1$ dimensions, $\boldsymbol{\gamma}_{12} \in \mathrm{sign}(\boldsymbol{\theta}_{12})$ is a $(p-1)$-dimensional column vector, and $\gamma_{22} \in \ \mathrm{sign}(\boldsymbol{\theta}_{22})$ is a scalar.

The following formula can be obtained from (14):

$$\boldsymbol{\omega}_{12} - \boldsymbol{s}_{12} - \rho \cdot \boldsymbol{\gamma}_{12} = 0. \tag{17}$$

Finding the subgradient of (9) yields the following:

$$\boldsymbol{W}_{11}\boldsymbol{\beta} - \boldsymbol{s}_{12} + \rho \cdot \boldsymbol{\nu} = 0 \tag{18}$$

where $\boldsymbol{\nu} \in \ \mathrm{sign}(\boldsymbol{\beta}).\boldsymbol{\theta}_{12} = -\boldsymbol{W}_{11}^{-1}\boldsymbol{\omega}_{12}\theta_{22}$ can be derived from (11). Because $\theta_{22} > 0$ and $\boldsymbol{\beta} = \boldsymbol{W}_{11}^{-1}\boldsymbol{\omega}_{12}$, $\mathrm{sign}(\boldsymbol{\theta}_{12}) = \mathrm{sign}(-\boldsymbol{W}_{11}^{-1}\boldsymbol{\omega}_{12}\theta_{22}) = -\mathrm{sign}(\boldsymbol{W}_{11}^{-1}\boldsymbol{\omega}_{12}) = -\mathrm{sign}(\boldsymbol{\beta})$ and $\boldsymbol{\nu} = -\boldsymbol{\gamma}_{12}$. Because $\boldsymbol{\omega}_{12} = \boldsymbol{W}_{11}\boldsymbol{\beta}$, i.e., (15) and (16) are equal, (6) and (9) are equal. Therefore, the objective function changes from (6) to (9).

REFERENCES

[1] "Crash factors in intersection-related crashes: An on-scene perspective," Nat. Center Stat. Anal., Nat. Highway Traffic Safety Admin., Washington, DC, USA, Rep. DOT HS 811 366, 2010.

[2] C. Liu and T. J. Ye, "Run-off-road crashes: An on-scene perspective," Nat. Highway Traffic Safety Admin., Washington, DC, USA, Rep. HS-811 500, 2011.

[3] D. Yi, J. Su, C. Liu, and W.-H. Chen, "Trajectory clustering aided personalized driver intention prediction for intelligent vehicles," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3693–3702, Jun. 2019.

[4] *Annual Report of Road Traffic Accident Statistics of the People's Republic of China*, Traffic Manag. Bureau Ministry Public Security, Beijing, China, 2019.

[5] M. Li, X. Wu, X. He, G. Yu, and Y. Wang, "An eco-driving system for electric vehicles with signal control under V2X environment," *Transp. Res. C, Emerg. Technol.*, vol. 93, pp. 335–350, Aug. 2018.

[6] M. Sepulcre and J. Gozalvez, "Context-aware heterogeneous V2X communications for connected vehicles," *Comput. Netw.*, vol. 136, pp. 13–21, May 2018.

[7] M. A. Javed, S. Zeadally, and E. Ben Hamida, "Data analytics for cooperative intelligent transport systems," *Veh. Commun.*, vol. 15, pp. 63–72, Jan. 2019.

[8] X. Zeng, K. Balke, and P. Songchitruksa, "Potential connected vehicle applications to enhance mobility, safety, and environmental security," Dept. Texas Transp. Inst., Texas A M Univ Syst., College Station, TX, USA, Rep. SWUTC/12/161103-1, 2012.

[9] Y. Guo, H. Zhang, C. Wang, Q. Sun, and W. Li, "Driver lane change intention recognition in the connected environment," *Physica A, Stat. Mech. Appl.*, vol. 575, Aug. 2021, Art. no. 126057.

[10] Y. Ali, A. Sharma, M. M. Haque, Z. Zheng, and M. Saifuzzaman, "The impact of the connected environment on driving behavior and safety: A driving simulator study," *Accid. Anal. Prevent.*, vol. 144, Sep. 2020, Art. no. 105643.

[11] O. Grembek, A. Kurzhanskiy, A. Medury, P. Varaiya, and M. Yu, "Making intersections safer with I2V communication," *Transp. Res. C, Emerg. Technol.*, vol. 102, pp. 396–410, May 2019.

[12] X. Li, A. Vaezipour, A. Rakotonirainy, S. Demmel, and O. Oviedo-Trespalacios, "Exploring drivers' mental workload and visual demand while using an in-vehicle HMI for eco-safe driving," *Accid. Anal. Prevent.*, vol. 146, Oct. 2020, Art. no. 105756.

[13] G. Njobelo, T. Sando, S. Sajjadi, E. Mtoi, E. E. Ozguven, and J. Sobanjo, "Safety evaluation of the advanced stop assist system in connected vehicle environment," *Transp. Res. Rec.*, vol. 2672, no. 22, pp. 47–57, Dec. 2018.

[14] H. A. Rakha, H. Chen, M. Almannaa, R. K. Kamalanathsharma, I. El-Shawarby, and A. Loulizi, "Field testing of eco-speed control using V2I communication," Dept. Transp. Res. Innov. Technol. Admin., Virginia Tech Transp. Inst., Blacksburg, VA, USA, Rep. DUNS: 0031370150000, 2016.

[15] S. Doecke, A. Grant, and R. W. G. Anderson, "The real-world safety potential of connected vehicle technology," *Traffic Injury Prevent.*, vol. 16, no. s1, pp. S31–S35, Jun. 2015.

[16] S. Lefèvre, J. Ibañez-Guzmán, and C. Laugier, "Context-based estimation of driver intent at road intersections," in *Proc. IEEE Symp. Comput. Intell. Veh. Transp. Syst. (CIVTS)*, 2011, pp. 67–72.

[17] T. Bo, S. Khokhar, and R. Gupta, "Turn prediction at generalized intersections," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2015, pp. 1399–1404.

[18] W. Wang *et al.*, "Vehicle trajectory clustering based on dynamic representation learning of Internet of Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3567–3576, Jun. 2021.

[19] Y. Jeong and K. Yi, "Target vehicle motion prediction-based motion planning framework for autonomous driving in uncontrolled intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 168–177, Jan. 2021.

[20] Y. Xing, C. Lv, and D. Cao, "Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1341–1352, Feb. 2020.

[21] T.-Q. Tang, Z.-Y. Yi, J. Zhang, and N. Zheng, "Modelling the driving behaviour at a signalised intersection with the information of remaining green time," *IET Intell. Transp. Syst.*, vol. 11, no. 9, pp. 596–603, 2017.

[22] F. Wirthmüller, J. Schlechtriemen, J. Hipp, and M. Reichert, "Teaching vehicles to anticipate: A systematic study on probabilistic behavior prediction using large data sets," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 7129–7144, Nov. 2021.

[23] A. Benterki, M. Boukhnifer, V. Judalet, and C. Maaoui, "Artificial intelligence for vehicle behavior anticipation: Hybrid approach based on maneuver classification and trajectory prediction," *IEEE Access*, vol. 8, pp. 56992–57002, 2020.

[24] G. Li, Y. Chen, D. Cao, X. Qu, B. Cheng, and K. Li, "Extraction of descriptive driving patterns from driving data using unsupervised algorithms," *Mech. Syst. Signal Process.*, vol. 156, Jul. 2021, Art. no. 107589.

[25] C. Ou and F. Karray, "Deep learning-based driving maneuver prediction system," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1328–1340, Feb. 2020.

[26] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, "Driver activity recognition for intelligent vehicles: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5379–5390, Jun. 2019.

[27] C. Lu, H. Wang, C. Lv, J. Gong, J. Xi, and D. Cao, "Learning driver-specific behavior for overtaking: A combined learning framework," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6788–6802, Aug. 2018.

[28] M. Veres and M. Moussa, "Deep learning for intelligent transportation systems: A survey of emerging trends," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3152–3168, Aug. 2020.

[29] H. Zhang and R. Fu, "A hybrid approach for turning intention prediction based on time series forecasting and deep learning," *Sensors*, vol. 20, no. 17, p. 4887, 2020.

[30] A. Zyner, S. Worrall, and E. Nebot, "Naturalistic driver intention and path prediction using recurrent neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1584–1594, Apr. 2020.

[31] S. Lee, M. Q. Khan, and M. N. Husen, "Continuous car driving intent detection using structural pattern recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 1001–1013, Feb. 2021.

[32] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Driver behavior classification at intersections and validation on large naturalistic data set," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 724–736, Jun. 2012.

[33] Q. Tran and J. Firl, "Modelling of traffic situations at urban intersections with probabilistic non-parametric regression," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2013, pp. 334–339.

[34] Q. Tran and J. Firl, "Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression," in *Proc. IEEE Intell. Veh. Symp.*, 2014, pp. 918–923.

[35] W. Song, G. Xiong, and H. Chen, "Intention-aware autonomous driving decision-making in an uncontrolled intersection," *Math. Problems Eng.*, vol. 2016, Apr. 2016, Art. no. 1025349.

[36] C. Tran, A. Doshi, and M. M. Trivedi, "Modeling and prediction of driver behavior by foot gesture analysis," *Comput. Vis. Image Understand.*, vol. 116, no. 3, pp. 435–445, Mar. 2012.

[37] V. Gadepally, A. Krishnamurthy, and U. Ozguner, "A framework for estimating driver decisions near intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 637–646, Apr. 2014.

[38] B. Yu, S. Bao, F. Feng, and J. Sayer, "Examination and prediction of drivers' reaction when provided with V2I communication-based intersection maneuver strategies," *Transp. Res. C, Emerg. Technol.*, vol. 106, pp. 17–28, Sep. 2019.

[39] A. Doshi and M. M. Trivedi, "On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 453–462, Sep. 2009.

[40] A. Jain, H. S. Koppula, S. Soh, B. Raghavan, A. Singh, and A. Saxena, "Brain4Cars: Car that knows before you do via sensory-fusion deep learning architecture," 2016, *arXiv:1601.00740*.

[41] O. Olabiyi, E. Martinson, V. Chintalapudi, and R. Guo, "Driver action prediction using deep (bidirectional) recurrent neural network," 2017, *arXiv:1706.02257*.

[42] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 2174–2182.

[43] S. Du, H. Guo, and A. Simpson, "Self-driving car steering angle prediction based on image recognition," 2019, *arXiv:1912.05440*.

[44] J. Chen, C. Zhang, J. Luo, J. Xie, and Y. Wan, "Driving maneuvers prediction based autonomous driving control by deep Monte Carlo tree search," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7146–7158, Jul. 2020.

[45] J. Wright, A. Y. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[46] J. Yin, Z. Liu, Z. Jin, and W. Yang, "Kernel sparse representation based classification," *Neurocomputing*, vol. 77, no. 1, pp. 120–128, 2012.

[47] Z. Chen *et al.*, "A novel sparse representation model for pedestrian abnormal trajectory understanding," *Expert Syst. Appl.*, vol. 138, Dec. 2019, Art. no. 112753.

[48] D. Torres-Boza, M. C. Oveneke, F. Wang, D. Jiang, W. Verhelst, and H. Sahli, "Hierarchical sparse coding framework for speech emotion recognition," *Speech Commun.*, vol. 99, pp. 80–89, May 2018.

[49] X. Wang, M. Yang, and D. Hurwitz, "Analysis of cut-in behavior based on naturalistic driving data," *Accid. Anal. Prevent.*, vol. 124, pp. 127–137, Mar. 2019.

[50] Z. Yihuan, L. Qin, W. Jun, S. Verwer, and J. M. Dolan, "Lane-change intention estimation for car-following control in autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 276–86, Sep. 2018.

[51] T. Rehder, A. Koenig, M. Goehl, L. Louis, and D. Schramm, "Lane change intention awareness for assisted and automated driving on highways," *IEEE Trans. Intell. Veh.*, vol. 4, no. 2, pp. 265–276, Jun. 2019.

[52] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos, "Tensor sparse coding for positive definite matrices," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 592–605, Mar. 2014.

[53] J. H. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[54] M. Yuan and Y. Lin, "Model selection and estimation in the Gaussian graphical model," *Biometrika*, vol. 94, no. 1, pp. 19–35, 2007.

[55] O. Banerjee, L. E. Ghaoui, and A. Daspremont, "Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data," *J. Mach. Learn. Res.*, vol. 9, pp. 485–516, Jun. 2008.

[56] Ministry of Transport of the People's Republic of China, *Specification for Layout of Highway Traffic Signs and Markings*. China: People's Commun. Press, 2009.

[57] Ministry of Transport of the People's Republic of China, *Specification for Layout of Urban Road Traffic Signs and Markings*. China: China Plan. Press, 2015.

[58] Y. Xing, C. Lv, H. Wang, D. Cao, and E. Velenis, "An ensemble deep learning approach for driver lane change intention inference," *Transp. Res. C, Emerg. Technol.*, vol. 115, Jun. 2020, Art. no. 102615.

[59] J. Peng, Y. Guo, R. Fu, W. Yuan, and C. Wang, "Multi-parameter prediction of drivers' lane-changing behaviour with neural network model," *Appl. Ergon.*, vol. 50, pp. 207–217, Sep. 2015.

[60] X. Li, W. Wang, and M. Roetting, "Estimating driver's lane-change intent considering driving style and contextual traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3258–3271, Sep. 2019.

[61] X. Li, O. Oviedo-Trespalacios, and A. Rakotonirainy, "Drivers' gap acceptance behaviours at intersections: A driving simulator study to understand the impact of mobile phone visual-manual interactions," *Accid. Anal. Prevent.*, vol. 138, Apr. 2020, Art. no. 105486.

[62] B. K. Pathivada and V. Perumal, "Analyzing dilemma driver behavior at signalized intersection under mixed traffic conditions," *Transp. Res. F, Traffic Psychol. Behav.*, vol. 60, pp. 111–120, Jan. 2019.

[63] J. Appiah, F. A. King, M. D. Fontaine, and B. H. Cottrell, "Left turn crash risk analysis: Development of a microsimulation modeling approach," *Accid. Anal. Prevent.*, vol. 144, Sep. 2020, Art. no. 105591.

[64] A. Zyner, S. Worrall, J. Ward, and E. Nebot, "Long short term memory for driver intent prediction," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2017, pp. 1484–1489.

[65] K. Li, X. Wang, Y. Xu, and J. Wang, "Lane changing intention recognition based on speech recognition models," *Transp. Res. C, Emerg. Technol.*, vol. 69, pp. 497–514, Aug. 2016.

[66] M. Zhang, R. Fu, D. D. Morris, and C. Wang, "A framework for turning behavior classification at intersections using 3D LIDAR," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7431–7442, Aug. 2019.

[67] S. Liu, K. Zheng, L. Zhao, and P. Fan, "A driving intention prediction method based on hidden Markov model for autonomous driving," *Comput. Commun.*, vol. 157, pp. 143–149, May 2020.

[68] X. Chen, L. Zhou, and L. Li, "Bayesian network for red-light-running prediction at signalized intersections," *J. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 120–132, 2019.
[69] P. Kumar, M. Perrollaz, S. Lefèvre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2013, pp. 797–802.
[70] Y. Jeong and K. Yi, "Bidirectional long shot-term memory-based interactive motion prediction of cut-in vehicles in urban environments," *IEEE Access*, vol. 8, pp. 106183–106197, 2020.
[71] Y. Choi, H. Lim, H. Choi, and I.-J. Kim, "GAN-based anomaly detection and localization of multivariate time series data for power plant," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, 2020, pp. 71–74.
[72] S. Lele and J. T. Richtsmeier, "Euclidean distance matrix analysis: A coordinate-free approach for comparing biological shapes using landmark data," *Amer. J. Phys. Anthropol.*, vol. 86, no. 3, pp. 415–427, Nov. 1991.
[73] D. J. Phillips, T. A. Wheeler, and M. J. Kochenderfer, "Generalizable intention prediction of human drivers at intersections," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2017, pp. 1665–1670.
[74] V. Alexiadis, J. Colyar, J. Halkias, R. Hranac, and G. McHale, "The next generation simulation program," *Ins. Transp. Eng. J.*, vol. 74, no. 8, pp. 22–26, 2004.
[75] H. Zhang and R. Fu, "An ensemble learning-online semi-supervised approach for vehicle behavior recognition," *IEEE Trans. Intell. Transp. Syst.*, early access, Jul. 16, 2021, doi: 10.1109/TITS.2021.3095053.
[76] S. Yang, W. Wang, Y. Jiang, J. Wu, S. Zhang, and W. Deng, "What contributes to driving behavior prediction at unsignalized intersections?" *Transp. Res. C, Emerg. Technol.*, vol. 108, pp. 100–114, Nov. 2019.

Hailun Zhang received the B.S. and M.S. degrees in vehicle engineering and mechanical engineering from Anhui University of Science and Technology, Huainan, Anhui, China, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree with the Director of Key Laboratory of Automobile Transportation Safety Technology, Chang'an University, Xi'an, China.

He is currently visiting Delft University of Technology, Delft, The Netherlands, as a Visiting Scholar. His research interests include vehicle dynamics, driving behavior modeling, and driving intention recognition in the vehicle–road collaborative environment.

Rui Fu received the B.S. degree in vehicle engineering from Jilin University of Technology, Changchun, China, in 1986, the M.S. degree in vehicle application engineering from Xi'an Highway College, Xi'an, China, in 1989, and the Ph.D. degree in vehicle application engineering from Jilin University of Technology in 1996.

She is currently a Professor with the Department of Transportation Safety, School of Automobile, Chang'an University, Xi'an. Her research interests include human factors, driving behavior, human–machine collaboration control, and vehicle active safety.

Chang Wang received the Ph.D. degree in vehicle application engineering from Chang'an University, Xi'an, China, in 2012.

He is currently a Professor with the Department of Vehicle Safety Engineering, Chang'an University. His research interests include lane-change warning, traffic simulation, driving behavior prediction, and autonomous driving.

Yingshi Guo received the B.Tech., M.S., and Ph.D. degrees in vehicle application engineering from Jilin University of Technology, Changchun, China, in 1985, 2000, and 2009, respectively.

He is currently a Professor with the Department of Vehicle Engineering, School of Automobile, Chang'an University, Xi'an, China. His research interests include automobile test technology, human–machine systems, vehicle control, and vehicle active safety.

Wei Yuan received the B.Tech., M.S., and Ph.D. degrees in transportation from Chang'an University, Xi'an, China, in 1997, 2003, and 2008, respectively.

He is currently a Professor with the Department of Transportation Safety Engineering, School of Automobile, Chang'an University. His research interests include traffic optimization, driving behavior, drivers' vision, and vehicle safety assistance technology.