

Received April 29, 2020, accepted May 11, 2020, date of publication May 15, 2020, date of current version June 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2994929

Bidirectional Long Short-Term Memory-Based Interactive Motion Prediction of Cut-In Vehicles in Urban Environments

YONGHWAN JEONG¹ AND KYONGSU YI¹, (Member, IEEE)

Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul 08826, South Korea

Corresponding author: Kyongsu Yi (kyi@snu.ac.kr)

This work was supported in part by the Technology Innovation Program through the Development and Evaluation of Automated Driving Systems for Motorways and City Roads funded by the Korean Ministry of Trade, Industry and Energy under Grant 10079730, and in part by the Ministry of Land, Infrastructure, and Transport through the Connected and Automated Public Transport Innovation (National Research and Development Project) under Project 18TLRP-B146733-01.

ABSTRACT This paper presents an interactive motion predictor to infer the intention of cut-in vehicles using a bidirectional long short-term memory (Bi-LSTM) module. The proposed predictor consists of three modules: maneuver recognition, trajectory prediction, and interaction. The driving data for training and validating the Bi-LSTM module were collected by sensors mounted on an autonomous vehicle (AV). In total, 3,828 trajectories of human-driven vehicles around the AV are accumulated in a global coordinate system. After postprocessing the collected trajectories, 83,188 and 35,652 data samples were used to train and validate the Bi-LSTM module, respectively. In the Bi-LSTM module, a maneuver is defined as the desired driving lane of a vehicle, which extend the behavior coverage of the proposed approach. The trajectory prediction step is based on the path-following model with a motion parameter estimator to predict the trajectories for all possible maneuvers. The interaction module considers the likelihood of each maneuver and the collision risk to determine the future trajectories of the surrounding vehicles in terms of the driving scene. The proposed predictor was evaluated in terms of its prediction accuracy and its effects on the motion planner of the AV. It has been shown that the AV benefits from the improved motion prediction of target vehicles provided by the proposed predictor with respect to enhanced safety and reduced control effort in the case of cut-in situations.

INDEX TERMS Autonomous vehicle, interactive motion prediction, machine learning, bidirectional long short-term memory, motion planning.

I. INTRODUCTION

The increasing demand for road safety, driver convenience, and traffic efficiency have led to substantial research on autonomous vehicles (AVs). The research field of automated driving is divided into perception for object detection, scene awareness decision making, and actuation control [1]. Decision made by AVs require the ability to infer other traffic participants' intentions and predict their future motions to understand the driving situation. Human drivers can infer and predict the behaviors of surrounding vehicles based on observed information and their driving experiences. In particular, predicting lane changes is an essential function to increase safety and improve traffic flow by agilely

responding to traffic. Out of all police-reported vehicle collisions in the United States, 9% were two-vehicle lane-change collisions [2]. Lack of attention was the leading collision-contributing factor accounting for 50% of lane-change cases [3]. Researchers have attempted to develop prediction algorithms for lane-change intentions that can realize drivers' perceptions of driving scenes. Prediction of driver behavior has been formulated as a problem in various ways and with numerous approaches. Motion predictors can be classified into one of two categories: (1) model-based and (2) learning-based predictors.

Researchers have utilized physics-based and maneuver-based models to predict the future motion of surrounding vehicles. Physics-based models use kinematic or dynamic models to predict future motion starting from the observed states of the targets. Investigators have employed the constant

The associate editor coordinating the review of this manuscript and approving it for publication was Yan-Jun Liu.

velocity (CV) model, the constant acceleration (CA) model, the constant turn rate (CTR) model, the constant turn rate/constant tangential acceleration (CTRA) model [4], and the path-following model [5] as physics-based prediction models. However, despite the advantages of simplicity and low computational burden, it is difficult to adapt these models to driving situations and interactions between vehicles.

Various maneuver-based prediction models have been studied to overcome the limitations of physics-based models. Maneuver-based models use a predefined set of behaviors, called maneuvers, to classify the driver's intentions. Lane keeping, lane changing, and turning are examples of maneuvers frequently used in the literature. In some studies, a trajectory-level predictor is prepared to represent the actual behavior of each maneuver. In addition, an interaction algorithm is employed to increase the prediction accuracy or the number of vehicles in a multivehicle situation. In terms of the predictor development methodology, the maneuver-based approach can be divided into model-based and learning-based methods. For model-based methods, the basic approach is the comparison between the instantaneous path of the vehicle and the shape of the road and recognizing the intended maneuver [6]. A driver-modeling framework that estimates an empirical reachable set to capture typical lane-changing behaviors was proposed [7]. To develop a probabilistic representation of nonlinear time-dependent behaviors, a dynamic Bayesian network (DBN) was utilized to construct a maneuver and trajectory prediction model [8]. The trajectory model for each maneuver was introduced to the DBN to realize trajectory-level prediction [9], [10]. The integrated approach with physics-based models, such as the CTRA model, was proposed to compensate for the shortcomings of each method [11]. A hidden Markov model (HMM) is used to reduce the computational load and complexity of the DBN [12] for maneuver prediction. A variational Gaussian mixture model (VGMM) was employed to fuse the predicted trajectories with probabilities from the HMM [13]. A similar approach using Gaussian mixture regression with a random decision forest was proposed for rarely occurring lane-change events [14].

Several learning-based approaches have been used to discern driver behaviors from driving data. The neural network is modified to parameterize the VGMM to overcome the limitations of parameter tuning [15]. A support vector machine (SVM) was applied to find a hyperplane, which classifies the observed vehicle states into predefined maneuvers [16], [17]. A nonlinear autoregressive neural network, which predicts the trajectory of the recognized maneuver by the SVM, was proposed to take advantage of machine learning-based classification and regression [17]. In addition, maneuver-based long short-term memory (LSTM) was applied to the probability distribution of the multi-model over future motion [18]. To reduce the computational burden, the occupancy grid map was applied to LSTM [19]. Convolutional social pooling is introduced to LSTM to learn interdependencies in vehicle motion [20].

A careful review of the literature reveals that various methodologies have been employed to develop a motion prediction algorithm. Many studies have defined different maneuvers to represent the behavior of the surrounding vehicles [8]–[14], [16]–[18], [20]. Due to the attempt to predict using specific maneuvers, several studies consider more than ten maneuvers [9], [10], [13], [20]. The increase in the number of maneuvers increases the complexity of the predictor and makes performance verification difficult. Many studies relied on the NGSIM dataset (from the Next Generation Simulation program), which is a public dataset for AV research [7], [15], [18], [20]. However, it is challenging to develop a prediction algorithm for AVs because vehicle information in the NGSIM dataset has been extracted from images collected from an overhead camera installed in the infrastructure.

This study focuses on improving in-lane target recognition and the prediction accuracy by introducing a Bi-LSTM-based interactive motion predictor. The proposed motion predictor is trained using data collected from the surrounding vehicles obtained by sensors on an AV. Real data captured on urban roads in Seoul, South Korea, are used to evaluate the accuracy of the motion predictor. Vehicle tests are conducted to show the improvement in the motion planning performance in multivehicle conditions on urban roads.

The main contributions of this work are as follows:

- 1) A Bi-LSTM-based maneuver recognizer is defined to infer the targeted driving lanes of surrounding vehicles.
- 2) Lateral and longitudinal motion parameter estimation is implemented to improve the trajectory-level prediction accuracy.
- 3) Interactions between vehicles based on the maneuver likelihood and collision probability are considered.

II. ARCHITECTURE OF THE INTERACTIVE MOTION PREDICTOR BASED ON A BI-LSTM MODEL

This study focused on the prediction of lane-changing maneuvers in multi-traffic conditions of urban environments. The interactive motion predictor based on Bi-LSTM has been proposed to realize the trajectory-level motion prediction of the surrounding vehicles. In previous studies, maneuver prediction, which infers the driver's lane-changing intention, was mainly considered in motorways and main roads in urban environments. As mentioned before, these studies have focused on selecting the most appropriate maneuver based on the observed target states. Therefore, intention inference is difficult for cases in which the vehicles behave like they are not included in the predefined maneuver sets. The increasing number of maneuvers, which is mostly used to cover various behaviors, causing the deterioration of the prediction accuracy.

Even though the intention of the surrounding vehicles is predicted appropriately, trajectory-level prediction should be employed to determine the desired motion of the AV based on physical quantities, not indexes. Furthermore, when predicting the motion of multiple vehicles on roads, interactions between vehicles should be considered drivers to predict

the behavior of surrounding vehicles. If the predictions are made separately for individual vehicles, there is a possibility of predicting a situation that cannot occur. An example is predicting a situation where two vehicles attempt to change lanes to the same lane simultaneously. Therefore, we introduce the interactive motion predictor to reduce the number of false intention inferences and increase the prediction accuracy.

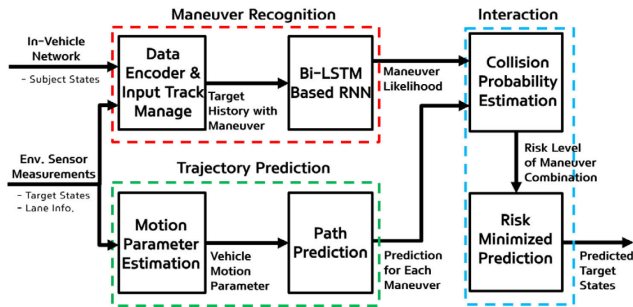


FIGURE 1. Overall architecture of the proposed interactive motion predictor.

The architecture of the Bi-LSTM-based interactive motion predictor is represented in Fig. 1. The proposed motion predictor is composed of three modules: (1) maneuver recognition; (2) trajectory prediction; and (3) interaction. Each module has two submodules. First, the maneuver recognition module consists of a data encoder with input track management and a Bi-LSTM-based RNN. The data encoder with input track management accumulates and standardizes the information from the in-vehicle network and environmental sensors. The in-vehicle network provides subject vehicle states, such as the velocity. The environmental sensors provide target vehicle states and lane information. The accumulated information is used as input features of the Bi-LSTM model, which estimates the likelihood of each maneuver. Second, the trajectory prediction module is composed of a motion parameter estimator and a path predictor. The trajectory prediction module uses target states and lane information to predict future trajectories. The motion parameter estimator estimates the maximum yaw rate of the lane-change behaviors and the desired velocity, which are used as input parameters of the path predictor. The path predictor uses the path-following model with an estimated motion parameter to predict the future trajectories for all possible maneuvers. Finally, the maneuver likelihood from the Bi-LSTM model and the predicted trajectories of each maneuver are considered by the interaction module to estimate the collision probability for all combinations of surrounding vehicle maneuvers. Then, the risk-minimized maneuver combination is used to determine the optimal prediction results.

III. BI-LSTM-BASED MANEUVER RECOGNITION

A. DATASET

We used a data collection vehicle to collect data from various vehicles driving on urban roads. The data collection

vehicle, which was developed as an AV, collected the states of target vehicles and lane marker information by using a laser scanner and vision sensors. To reflect the interactions between the AV and the surrounding vehicles, the states of the data collection vehicle are also accumulated in synchronization with the information from the environmental sensors. In addition, the data collection vehicle drove with real traffic to acquire information obtainable by the environmental sensors of the AV. The predictor based on this dataset can be directly utilized in autonomous driving because the information obtained from infrastructure is excluded. The details will be discussed in the following section.

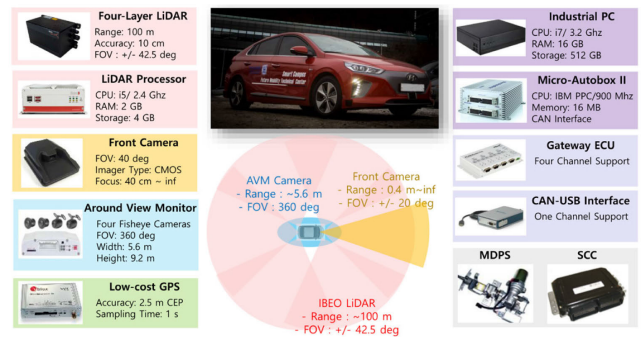


FIGURE 2. Configuration of data collection vehicle.

1) DATA COLLECTION VEHICLE AND THE TARGET ROADS

The configuration of the data collection vehicle is represented in Fig. 2. As mentioned before, the data collection vehicle is designed as an AV for urban environments by covering the 360-degree perception area around the AV. To perceive the surrounding vehicles, we used six ibeo LUX sensors with an ibeo.HAD Feature Fusion system, which detects traffic participants up to 100 m at 25 Hz. This LiDAR system provides the position, heading, and velocity in local coordinates relative to the data collection vehicle with classified information. A front camera, Mobileye Q3, is used to collect the lane information as a second-order polynomial with a recognition quality index. In addition, around-view monitoring (AVM) and a low-cost GPS were employed to acquire road markers and a global position for the localization of the AV in urban environments. A gateway electronic control unit (ECU) is used to interface with the chassis controller area network (CAN) and collect the outputs of the chassis sensors. To acquire highly accurate localization results on urban roads, the outputs of the chassis sensors and low-cost GPS and lane marks from the AVM image were fused to estimate the global position of the AV, which is described in [21]. All the data were stored on an industrial PC. A MicroAutobox II and a motor-driven power steering/smart cruise control module were used to control and operate the AV. The driving data on the surrounding vehicle trajectories were collected on the urban roads of Gwanak-gu, Seoul, South Korea. Fig. 3 presents the driving route of data collection highlighted in the solid red line on a satellite map.

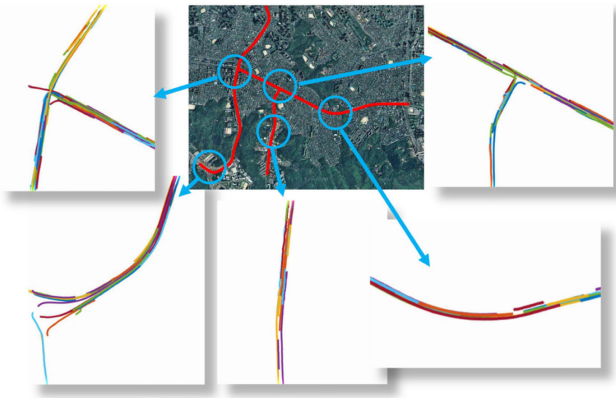


FIGURE 3. Data collection roads and the collected vehicle trajectories in global coordinates.

2) DATA ACCUMULATION AND SELECTION

The collected states of the surrounding vehicles and lane information are defined by the local coordinates of the data collection vehicle with a time index. To create the appropriate input sequence of Bi-LSTM for learning the motion patterns of the vehicles, all the states and lane information should be defined with fixed global coordinates. The estimated global position of the data collection vehicle is used to transform the collected data from local to global coordinates. In Fig. 3, the five enlarged subfigures show examples of the trajectory data accumulated in the fixed global coordinate system.

The collected data contained 3,828 trajectories of human-driven vehicles with lane information. These trajectories were processed to generate data samples to train and validate the Bi-LSTM-based maneuver recognition module. After processing the collected data, 83,188 data samples for training and 35,652 for validation were generated. An example of the generated data is shown in Fig. 4. In the figure, the data collection vehicle and the surrounding vehicles are depicted by black and green vehicles, respectively. The input and output sequences of the Bi-LSTM module are represented in a blue and red-colored vehicle.

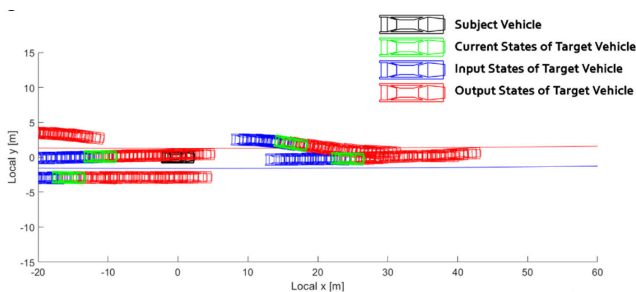


FIGURE 4. Example of data extracted from accumulated vehicle trajectories.

B. MANEUVER RECOGNIZER

The Bi-LSTM-based maneuver recognizer was proposed to estimate the likelihood of each maneuver. Since the vehicle

shows continuous behaviors governed by vehicle dynamics, the time-dependent characteristics should be considered when designing the motion predictor. Rule-based conventional approaches have the advantage that they can be configured with a small amount of data. However, they have the disadvantage that the design is too complex to cope with various driving situations. Therefore, the maneuver recognizer based on the Bi-LSTM-based RNN architecture was proposed based on the information collected from the sensors on the AV.

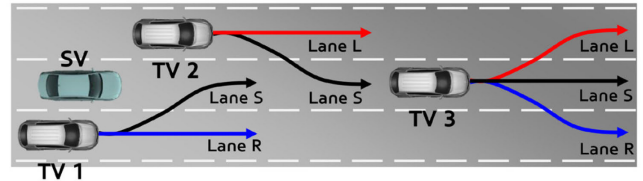


FIGURE 5. Maneuver definition for the road-structure-based interactive motion predictor.

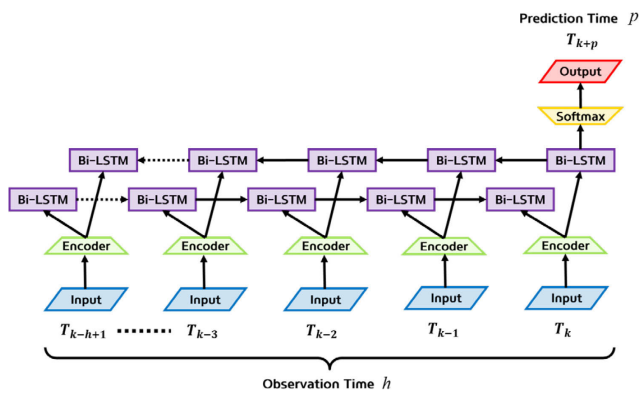
1) MANEUVER DEFINITION

The maneuver of the proposed predictor is defined from the perspective of driving lanes of the subject AV. The graphical representation of the maneuver definition is described in Fig. 5. The maneuvers mean the lanes each vehicle intends to drive in, not a lane-change intention. Therefore, the results of the maneuver recognition module can be directly used to classify the driving lanes of the surrounding vehicles into in-lane, left-lane, and right-lane targets. The other advantage of the proposed maneuver definition reduces the risks of misprediction, such as a double lane change to two lanes away from the subject vehicle. In this case, the subject vehicle might decide to change to a risky lane. Therefore, the proposed maneuver excludes double lane changes.

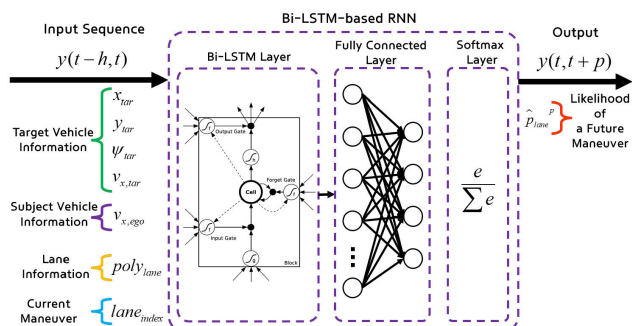
2) NETWORK ARCHITECTURE

This study used a Bi-LSTM-based RNN to construct a maneuver recognizer, which estimates the likelihood of each maneuver. Among the learning-based methodologies, RNNs are suitable for dealing with sequential data. Since the activations in each step are passed to the same network of the next time step and updated with new input data, one set of weights is repeated over the observation horizon. Therefore, when using the RNN, there is an advantage that the parameter is reduced compared to a general neural network in performing the prediction for the same inputs. In addition, Bi-LSTM is introduced to learn long-term dependencies more accurately than simple recurrent architectures. In particular, bidirectional characteristics are useful when the context of the input is essential to increase the prediction accuracy.

Fig. 6 (a) describes the unrolled structure of the Bi-LSTM-based RNN used in this study for an observation horizon h and a prediction horizon of p . The proposed maneuver recognizer outputs maneuver prediction results after p steps.



(a) Diagram representing the proposed RNN for an observation time h and a prediction time p .



(b) Conceptual diagram of the single step of the LSTM-RNN predictor.

FIGURE 6. Diagram of the proposed Bi-LSTM-based maneuver recognition module.

The single step of the prediction using the proposed Bi-LSTM-based RNN is conceptually expressed in Fig. 6 (b). The inputs of the proposed predictor are composed of the target states, lane information, and current maneuver.

The structure of the Bi-LSTM-based RNN has been determined by comparing the prediction accuracies of candidate networks. Fig. 7 presents each layer of the Bi-LSTM-based RNN with the number of cells in each layer. This structure is determined by comparing the recall and precision of 84 RNN candidates. The candidates consist of combinations of fully connected (FC) and Bi-LSTM layers.

3) INPUT AND OUTPUT FEATURES

The objective of maneuver recognition is to estimate the likelihood of each maneuver and driving lane based on the information from the AV sensors. Therefore, the input features of the Bi-LSTM-based RNN are composed of the target vehicle states, subject vehicle states, lane information, and driving lane of the target vehicle. The target vehicle states consist of the x and y positions, heading angle, and velocity of the surrounding target vehicles in local coordinates, which are reconstructed from the dataset, as shown in Fig. 4. The velocity is used as the subject vehicle state. The lane information is composed of the detection quality, curvature, heading angle, and lateral offset of the left and right lanes. Finally, the driving lane of the target vehicle is defined based on the

maneuver definition, which is described in Fig. 5. The output is the predicted driving lane of the target vehicles after the prediction horizon.

4) ENCODER

The neural network input data should be preprocessed to improve stability and performance. In this study, we introduce an encoder to standardize each component of the input data, which rescales the data to a mean of 0 and a standard deviation of 1. Parameters μ and σ were determined using the 83,188 training data samples only and were stored to reuse when validating the proposed algorithm and applying it to the AV. The input to the network is standardized as

$$\bar{x}_{t,n} = \frac{x_{t,n} - \mu_n}{\sigma_n}, \quad (1)$$

where $x_{t,n}$ is the n-th component of the input data such as the position or heading at time t . In addition, $\bar{x}_{t,n}$ is the standardized input of $x_{t,n}$, and μ_n and σ_n are the mean and standard deviation, respectively, of the n-th component. Therefore, 14 μ_n and σ_n values were prepared based on the training dataset.

5) SEQUENCE LENGTH

The sequence length consists of the input and output sequence length. Each length corresponds to the observation horizon and prediction horizon. The observation horizon h is an important factor in improving prediction performance. For this study, we trained the network architecture depicted in Fig. 6 using several candidates in the observation history to find the optimal length. We compared 10, 15, 20, 25, 30, 35, 40, 45, and 50 steps with a sampling time of 100 ms. The results show that an observation horizon of 50 steps yields the most accurate results compared to other horizons. However, the longer the observation horizon is, the longer the delay in performing accurate prediction after recognizing the target. Therefore, we use the second-most accurate observation horizon, 25 steps, as an optimal observation horizon of the interactive motion predictor. For the prediction horizon p , we use 50 steps to manage the risk between the subject and surrounding vehicles considering the driving speed on the urban road. Fifty steps is the prediction horizon before the decrease in the prediction accuracy largely increases when the prediction is performed with 25 input data steps.

IV. TRAJECTORY PREDICTION WITH INTERACTIONS

A. TRAJECTORY PREDICTION

The trajectory prediction module consists of two submodules: motion parameter estimation and path prediction. Path prediction was conducted using the path-following model, which is parametrized as the maximum yaw rate γ_{max} for lateral motion and the desired velocity u for longitudinal motion. The input sequences of each target of the Bi-LSTM module are utilized to estimate γ_{max} and u . Based on the estimated parameters, the trajectory prediction module predicts the trajectories of all possible maneuvers.

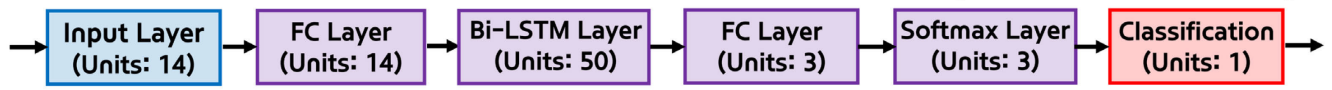


FIGURE 7. Diagram that depicts the individual layers of the Bi-LSTM-based maneuver recognition module.

1) PATH-FOLLOWING MODEL

Predicting the future motion of moving vehicles is a crucial part of autonomous driving to guarantee the safety of the vehicle. An appropriate model is required to predict future motion precisely by using the state estimates of the vehicle as an initial prediction condition. The state vector of the prediction model at time step j is defined as

$$x_{kj} = [p_{x,kj} \ p_{y,kj} \ \theta_{kj} \ v_{kj}]^T, \quad (2)$$

where $p_{x,kj}$, $p_{y,kj}$, and $p_{\theta,kj}$ are the x position, y position, and heading angle, respectively, of the vehicle at prediction step k in relation to the fixed coordinate defined on the digital map; v_{kj} is the absolute velocity of the vehicle at the same prediction step. It is assumed that the slip of the vehicle has been maintained at a negligible level. In other words, the vehicle motion can be modeled as a kinematic model. Based on this assumption, the process update model for motion prediction is defined as

$$\begin{aligned} p_{x,k+1} &= p_{x,k} + v_k \cos \theta_k dt + (a_k \cos \theta_k - v_k \gamma_k \sin \theta_k) dt^2 / 2, \\ p_{y,k+1} &= p_{y,k} + v_k \sin \theta_k dt + (a_k \sin \theta_k + v_k \gamma_k \cos \theta_k) dt^2 / 2, \\ \theta_{k+1} &= \theta_k + \gamma_k dt, \\ v_{k+1} &= v_k + a_k dt. \end{aligned} \quad (3)$$

The key issue of motion prediction is determining how to assume the future behavior of moving vehicles. In this study, moving vehicles are assumed to stay in their lanes, which is termed the path-following model. The possibility of current lane departure will be discussed in the following section. To stay in the lane, virtual inputs $a_{k,input}$ and $\gamma_{k,input}$ for the moving vehicle are defined as

$$\begin{aligned} a_{k,input} &= a_{max} \left[1 - \left(\frac{v_k}{u} \right)^\delta \right], \\ \gamma_{k,input} &= \min \left(-\gamma_{max} \max \left(\gamma_{max} \frac{(1+y_k^2)^{3/2}}{y_k} \right) \right), \end{aligned} \quad (4)$$

where y_k is the approximated centerline in the 4th order polynomial tracked by the target vehicle. In this study, the following lanes have been estimated using the road information of the digital map. Therefore, $\gamma_{k,input}$ is defined by the desired yaw rate calculation based on the curvature of the lane. Then, $a_{k,input}$ is determined based on the velocity tracking term of the intelligent driver model (IDM). The parameters γ_{max} and u will be discussed in the following section.

2) MOTION PARAMETER ESTIMATION

γ_{max} is estimated under the assumption that the lateral acceleration of the lane-changing vehicle is a sinusoidal pattern.

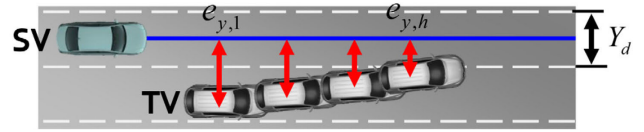


FIGURE 8. Definition of the lateral offset for maximum yaw rate estimation of lateral motion prediction.

The lateral acceleration is assumed to be

$$\begin{aligned} a_y &= \ddot{y} + v_x \cdot \gamma \\ &= v_x \cdot \gamma_{max} \cdot \sin \left(\frac{2\pi}{t_{LC}} t \right), \end{aligned} \quad (5)$$

where t_{LC} is the time it takes for the vehicle to change lanes. The longitudinal velocity v_x of the targets is assumed to be constant while changing lanes. The lateral position p_y is derived by integrating equation (5):

$$p_y = v_x \cdot \gamma_{max} \cdot \frac{t_{LC}}{2\pi} \left[t - \frac{t_{LC}}{2\pi} \sin \left(\frac{2\pi}{t_{LC}} t \right) \right]. \quad (6)$$

From the lane information obtained by the vision sensors, the lane width Y_d is measurable, which is the lateral distance traveled during a lane change. Therefore, p_y should be matched with Y_d after t_{LC} as

$$\begin{aligned} p_y &= v_x \cdot \gamma_{max} \cdot \frac{t_{LC}}{2\pi} \left[t_{LC} - \frac{t_{LC}}{2\pi} \sin \left(\frac{2\pi}{t_{LC}} t_{LC} \right) \right] \\ &= v_x \cdot \gamma_{max} \cdot \frac{t_{LC}^2}{2\pi} \\ &= Y_d. \end{aligned} \quad (7)$$

t_{LC} is estimated based on the trajectory history of the target vehicle. The lateral offset to the target lane at step h is defined as $e_{y,h}$, which is shown in Fig. 8. The first and last offset $e_{y,1}$ and $e_{y,h}$ are used to estimate t_{LC} :

$$\begin{aligned} \hat{v}_y &= (e_{y,h} - e_{y,1}) / t_h, \\ \hat{t}_{LC} &= Y_d / \hat{v}_y. \end{aligned} \quad (8)$$

Then, γ_{max} can be determined by substituting \hat{t}_{LC} into equation (7):

$$\gamma_{max} = \frac{Y_d \cdot 2\pi}{v_x \cdot \hat{t}_{LC}^2}. \quad (9)$$

Second, the desired velocity u of the intelligent driver model (IDM) is estimated to predict future longitudinal motion. The desired velocity candidate u_{candi} is defined by means of linear spacing with an interval of 0.5 m/s. The minimum and maximum values of u_{candi} assume an available

longitudinal acceleration ranges from -5m/s^2 to 1.5m/s^2 within the observation horizon h . u_{candi} is defined as

$$u_{candi} = \left[v_{x,1} + \left(-5\text{m/s}^2 \cdot t_h \right) : 0.5 : v_{x,1} + \left(1.5\text{m/s}^2 \cdot t_h \right) \right], \quad (10)$$

where $v_{x,1}$ is the first velocity of the input sequences. Then, the future longitudinal velocity is predicted by the IDM as

$$v_{k+1} = v_k + a_k \cdot dt, \quad (11)$$

$$a_k = a_{\max} \left[1 - \left(\frac{v_k}{u_{candi}} \right)^\delta \right],$$

where $a_{\max} = 3\text{m/s}^2$ and $\delta = 4$. The cost function for the selection of u_{candi} gives more weight to the observation history near the current state by using an exponential function. \hat{u} is determined by evaluating the error between the observed velocity $v_x(k)$ and the predicted velocity $v_{x,u}(k)$ as

$$\hat{u} = \arg \min_u \sum_{k=1}^{t_h/dt} e^{\lambda k} (v_x(k) - v_{x,u}(k))^2, \quad (12)$$

where λ is the forgetting factor to weigh an observation later.

B. INTERACTIONS

From the maneuver recognition and path prediction modules, we can obtain the likelihood and predicted motion of each maneuver. In this step, we consider all the likelihoods and predicted motions simultaneously to compensate for false predictions from the maneuver recognition module. In other words, the objective of the interaction module is to eliminate the false prediction cases by minimizing the cost function considering the maneuver likelihood and the collision risk between vehicles. The cost function for interactive maneuver prediction consists of three energies: the maneuver likelihood P_{ik}^{RNN} , collision with the subject vehicle P_{ik}^{sub} , and collision between target vehicles P_{ijkl} . Before describing the meaning of each energy, the indexes should be defined. i and j represent the i -th and j -th target vehicles, respectively. k and l are the maneuver indexes, such as lanes L, S, and R. The cost function is defined as

$$\sum_{i=1}^N \left[P_{ik}^{RNN} + \lambda_1 P_{ik}^{sub} \right] + \lambda_2 \cdot \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N P_{ijkl},$$

where

$$P_{ik}^{RNN} = -\log(E_{ik}^{RNN}),$$

$$P_{ik}^{sub} = \left[\sum_{n=1}^p \left((x_{sub}(n) - x_{i,k}(n))^2 - (y_{y,sub}(n) - x_{i,k}(n))^2 \right) \right]^{-1},$$

$$P_{ijkl} = \left[\sum_{n=1}^p \left((x_{i,k}(n) - x_{j,l}(n))^2 - (y_{i,k}(n) - y_{j,l}(n))^2 \right) \right]^{-1}. \quad (13)$$

P_{ik}^{RNN} is defined as a negative log of the maneuver likelihood P_{ik}^{RNN} . P_{ik}^{sub} is the collision risk between the subject

TABLE 1. Integrated cost function evaluation example.

		Maneuver Set								
Target 1	L	L	L	S	S	S	R	R	R	
Target 2	L	S	R	L	S	R	L	S	R	
Cost	132	158	125	895	147	203	659	871	507	

vehicle and i -th target vehicle when performing maneuver k . P_{ijkl} is the collision risk between the i -th target vehicle when it is performing maneuver k and the j -th target vehicle when it is performing maneuver l . The proposed cost function is evaluated for all combinations of the possible maneuvers of the target vehicles. For example, the cost evaluation of the two target scenarios is summarized in Table 1. In this example, the maneuver set of target 1, which is driving in the left lane, and target 2, which is driving in the right lane, is the optimal solution of the nine maneuver sets. The same approach is applied to multiple vehicle target cases. If N target vehicles exist, 3^N maneuver sets are considered to determine the optimal maneuver set.

V. PREDICTION PERFORMANCE ANALYSIS

The proposed interactive motion predictor was evaluated through a driving data-based evaluation in three aspects. First, in Section V. A, the performance of the maneuver recognition module was evaluated based on the 35,652 data samples, which were not used to train the Bi-LSTM module. This analysis includes an analysis of the maneuver recognition accuracy using individual data samples with true labels and an analysis of lane-change detection timing using 486 trajectories of lane-change maneuvers. Second, in Section V. B, the trajectory-level prediction accuracy of the interactive motion predictor is evaluated and compared with the accuracy of conventional algorithms. Finally, in Section V. C, the advantage of using the proposed interaction module is analyzed with three representative scenarios in which it is difficult to make proper predictions.

A. MANEUVER RECOGNITION ANALYSIS

To compare the prediction accuracy of maneuver recognition, the parameters are defined as follows. In Section III. B, a maneuver was defined as the driving lane around the subject vehicle, such as in-lane, left lane, and right lane. Each maneuver is labeled with ‘Lane S,’ ‘Lane L,’ or ‘Lane R,’ as shown in Fig. 5. In other words, maneuver recognition is a classification problem of future driving lanes based on the observed behaviors of the surrounding vehicles. Therefore, confusion matrices have been used to describe the performance of the proposed maneuver recognition module on a set of test data for which the true values are known. However, we defined the confusion matrix based on the lane-change motion because it is important to recognize the lane-changing moment and it is easier to compare our approach with conventional approaches, which try to infer the lane-changing intention. Therefore, the confusion matrix for the

TABLE 2. Confusion matrix of maneuver recognition module.

		True Condition	
		Condition Positive	Condition Negative
Predicted Condition	Predicted Condition Positive	True Positive (TP) → LC is predicted as LC	False Positive (FP) → LK is predicted as LC
	Predicted Condition Negative	False Negative (FN) → LC is predicted as LK	True Negative (TN) → LK is predicted as LK

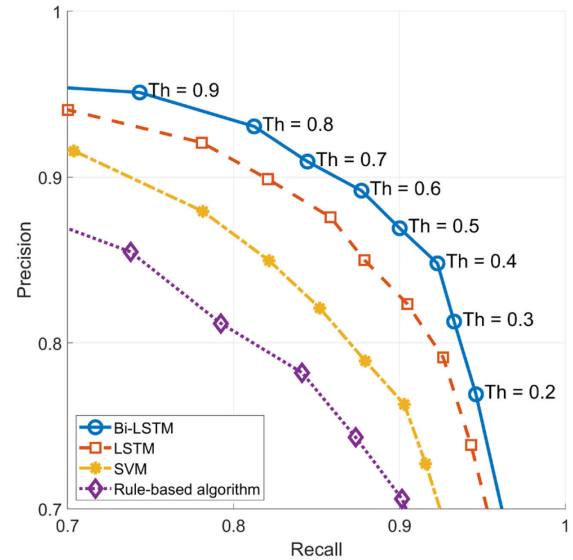
performance analysis of the proposed maneuver recognition module is defined in Table 2. In this case, lane change (LC) means that the driving lane after the prediction horizon is different from the current driving lane. Moreover, lane keeping (LK) means that the driving lane is maintained during the prediction horizon.

As shown in Table 2, the case of predicting the LC condition properly is defined as a true positive (TP). Moreover, the case of correctly predicting the LK condition is defined as a true negative (TN). Based on these definitions, false negatives (FNs) and false positives (FPs) are defined based on false recognitions. To quantify the results of the maneuver recognition module, the true positive rate (TPR)/recall, false positive rate (FPR) and precision are used. The definitions of the recall and precision are

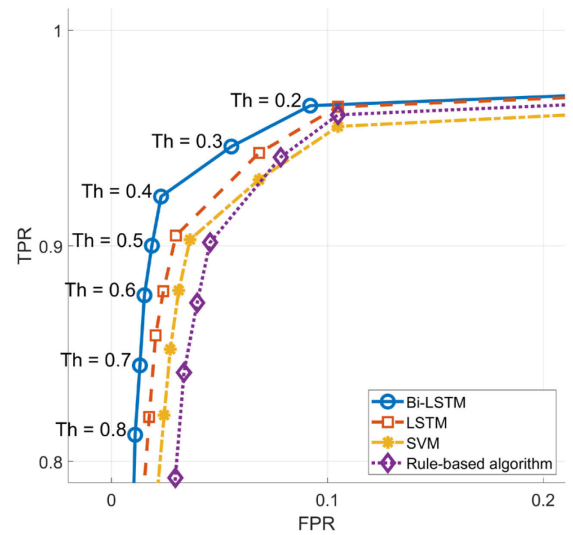
$$\begin{aligned}
 TPR/Recall &= \frac{TP}{TP + FN}, \\
 FPR &= \frac{FP}{TN + FP}, \\
 Precision &= \frac{TP}{TP + FP}.
 \end{aligned} \tag{14}$$

The precision-recall (PR) curve and receiver operating characteristic (ROC) curve of the maneuver recognition with thresholds are depicted in Fig. 9. To analyze the accuracy of the proposed approach, an LSTM network, an SVM, and a rule-based maneuver recognizer were used as a comparison. The rule-based algorithm determines the lane change based on the speed and position of the vehicle approaching the lane. This algorithm is the same approach as the lane departure warning system [22].

The results indicate that the proposed Bi-LSTM-based approach improves the accuracy of maneuver recognition for all the thresholds in terms of the PR curve and ROC curve. As shown in Fig. 9, the PR and ROC curves of the Bi-LSTM-based approach are closer to the upper-right and upper-left corners than the conventional approaches for all the thresholds. The areas under the ROC curves (AUCs) of the Bi-LSTM, LSTM, SVM, and rule-based maneuver recognizer algorithms are 0.971, 0.966, 0.957, and 0.955, respectively. In particular, approximately 95% of the lane-change maneuvers can be detected in a prediction horizon of 5 seconds with an FPR of only 5%. Moreover, when the FPR is the same, the conventional approaches achieve accuracies of approximately 90 to 92%. In particular, the LSTM models outperformed the SVM and rule-based approaches



(a) Precision-recall curve



(b) Receiver operating characteristic curve

FIGURE 9. Analysis of the proposed maneuver recognition module with respect to classification thresholds.

because LSTM models can reflect the time dependencies of the input sequences. Bi-LSTM, which learns long-term dependencies more effectively than LSTM, has a better prediction performance than the conventional approaches.

The recognition timing of lane changes between the proposed maneuver recognizer and the rule-based algorithm are compared. In this comparison, the maneuver recognizer used a threshold of 0.55. This threshold is selected to balance the recall and precision while managing the FPR because both FPs and FNs mean the incorrect prediction of the vehicle behavior between the LK and LC conditions. Therefore, these two errors should be considered equally in terms of driving safety. Therefore, the threshold was determined to be

0.55 based on the PR curve, while the FPR was maintained under a reasonable value by considering the ROC curve. In this case, approximately 89% of lane-change maneuvers can be detected while allowing an FPR of only 2% by the proposed maneuver recognizer.

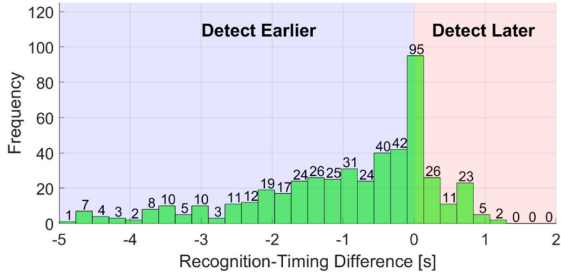


FIGURE 10. Histogram of the lane change recognition-timing difference between the proposed and conventional algorithm.

The lane-change recognition-timing differences between the proposed maneuver recognizer and the rule-based algorithm are presented in Fig. 10. This difference is defined as the lane-change recognition time of the proposed algorithm based on the lane-change recognition time of the rule-based algorithm. Therefore, if the proposed algorithm detects lane changes early, the difference becomes negative. The analysis was performed on 486 cases in which a lane change occurred. Fig. 10 shows that in 324 of the 486 cases, the proposed algorithm recognized the lane-change intention earlier than the rule-based algorithm by up to 5 seconds. Ninety-five cases showed the same recognition timing. However, in 67 cases, the proposed algorithm recognized the lane change later by up to 1.2 seconds. This phenomenon occurs when predictions are made before the newly detected target has observed less than 25 steps, which is the observation horizon of the Bi-LSTM. In other words, the later recognition cases only occurred when the target vehicles first appeared beyond the sensors’ region of interest boundaries, which means that these cases took place sufficiently beyond the safety distance of the subject vehicle. Therefore, these cases had little influence in determining the behavior of the subject vehicle.

B. PREDICTION ACCURACY ANALYSIS

The prediction error was defined to compare the prediction accuracy between the true and predicted states. The x-position error e_{x,T_p} , y-position error e_{y,T_p} , heading error e_{θ,T_p} , and velocity error e_{v,T_p} were defined as

$$\begin{aligned}
 e_{x,T_p} &= p_{x,T_p} - \hat{p}_{x,T_p}, \\
 e_{y,T_p} &= p_{y,T_p} - \hat{p}_{y,T_p}, \\
 e_{\theta,T_p} &= \theta_{T_p} - \hat{\theta}_{T_p}, \\
 e_{v,T_p} &= v_{T_p} - \hat{v}_{T_p}.
 \end{aligned} \tag{15}$$

Among the prediction errors, e_{x,T_p} and e_{v,T_p} were defined in local coordinates that originated from the true state at T_p , as shown in Fig. 11. This error definition prevented the

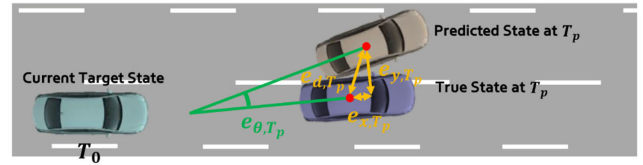
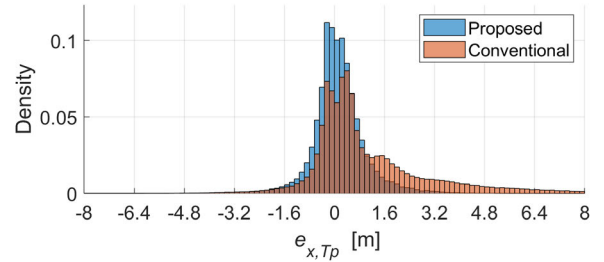
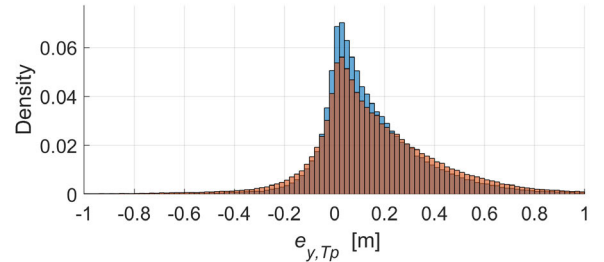


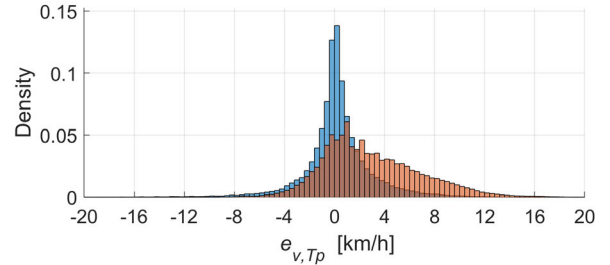
FIGURE 11. Definition of the prediction error at prediction time T_p .



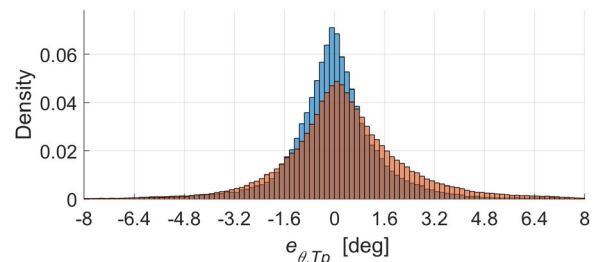
(a) Error distribution of the x-position error e_{x,T_p}



(b) Error distribution of the y-position error e_{y,T_p}



(c) Error distribution of the velocity error e_{v,T_p}



(d) Error distribution of the heading error e_{θ,T_p}

FIGURE 12. Comparison of the prediction error between the proposed and base algorithm.

misinterpretation of the predicted results caused by changing the heading angle that occurs when driving on curved roads or changing lanes.

The motion prediction error of the interactive motion predictor is summarized in Fig. 12 and Table 3. Fig. 12 shows

TABLE 3. Comparison of the prediction errors between the proposed predictor and the CTRV model.

Error	Proposed predictor			CTRV model		
	Mean	STD	RMSE	Mean	STD	RMSE
e_x (m)	0.13	0.87	1.18	1.16	1.99	2.57
e_y (m)	0.14	0.23	0.35	0.15	0.30	0.47
e_θ (°)	0.00	1.68	2.27	0.38	2.35	2.74
e_v (km/h)	0.25	2.81	2.91	3.19	3.90	6.43

the error distributions of the proposed algorithm and the conventional algorithm. In this analysis, the constant turn rate and velocity (CTRV) model, which is frequently used as a prediction model for moving objects in 2-D space, was used as the conventional algorithm. As shown in Fig. 12, the proposed motion predictor shows more accurate results than the CTRV model in all aspects. The proposed algorithm shows significantly reduced prediction errors compared to the base algorithms in terms of the mean, standard deviation (STD), and root mean square error (RMSE).

Parameter estimation for longitudinal motion reduces $e_{x,Tp}$ and $e_{v,Tp}$. In particular, the distributions of $e_{x,Tp}$ and $e_{v,Tp}$ are biased toward the positive side for the CTRV model because the range of the deceleration is larger than the acceleration under normal driving conditions. However, the proposed approach shows a distribution similar to a normal distribution, which means that the biases of $e_{x,Tp}$ and $e_{v,Tp}$ reduce that of the CTRV model. The standard deviations improved to 43.7% for $e_{x,Tp}$ and 72.1% for $e_{v,Tp}$. For lateral motion, the standard deviations of $e_{y,Tp}$ and $e_{\theta,Tp}$ improved to 76.6% and 71.5%, respectively. Since lane changes occur in a tiny portion of the total data, the influence from the statistical analysis is minimal. Therefore, the prediction of lane-keeping vehicles has a significant influence on the error distributions of $e_{y,Tp}$ and $e_{\theta,Tp}$. These improvements are mainly caused by the path-following model and parameter estimation for longitudinal motion. In short, the standard deviations of $e_{x,Tp}$, $e_{y,Tp}$, $e_{v,Tp}$, and $e_{\theta,Tp}$ are bound within a reasonable level, which makes it possible for the AV to perform prediction-based motion planning. Therefore, the motion planning of AVs based on the proposed prediction algorithm can increase safety and passenger acceptance of autonomous driving. The vehicle test results using the AV are discussed in Section VI.

C. INTERACTIVE MOTION PREDICTOR

The effectiveness of the interaction module of the interactive motion predictor is summarized in Fig. 13 with three representative cases. What these three cases have in common is the prediction failure case of the maneuver recognition module. In Fig. 13, the prediction results with environmental sensor outputs are shown in the top view on the left side. All vehicles in the top subfigure show the currently recognized position, the prediction results for 2 seconds at 0.4-second intervals considering the scale of the figure, and the likelihood of each maneuver ('Lane S,' 'Lane L,' and

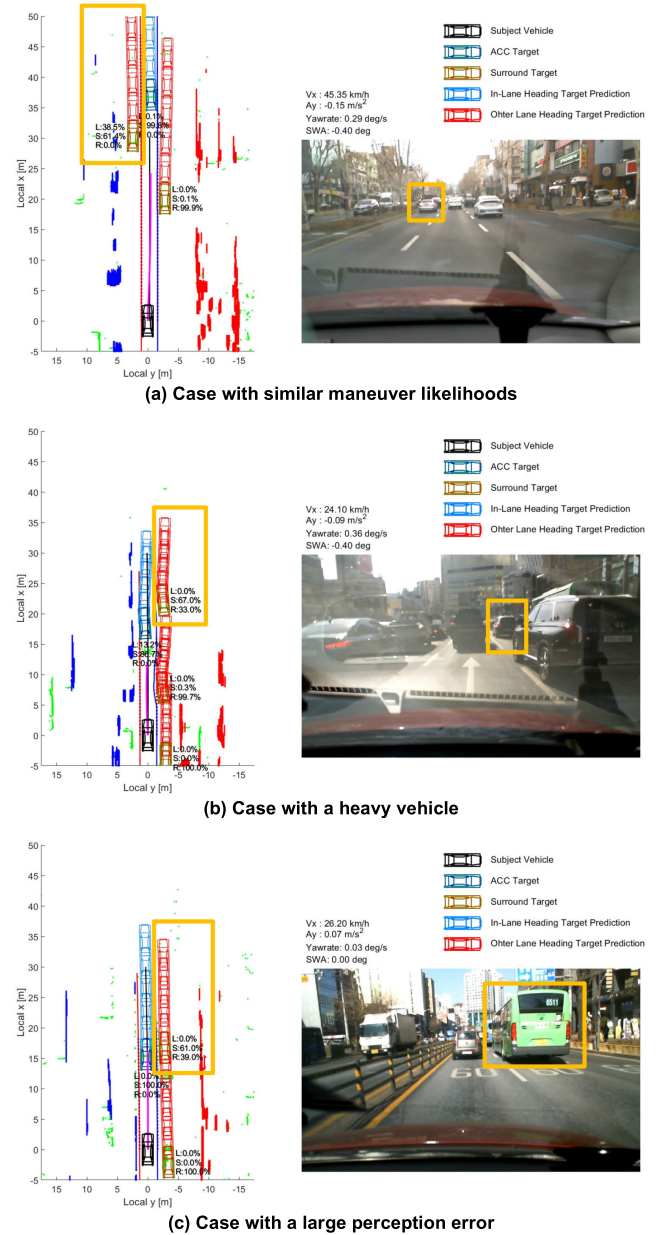


FIGURE 13. Effectiveness of the interaction module of the interactive motion predictor.

'Lane R') The driving scene which was recorded by the dash cam is shown on the right side. Among the surrounding vehicles in the three cases, the vehicle whose result was corrected by the interaction module is indicated by the yellow box even though the prediction results of the maneuver recognition module are incorrect.

The first situation is a case where the likelihoods of 'Lane L' and 'Lane S' of a vehicle driving in the left lane are similar, as shown in Fig. 13 (a). Therefore, it is difficult to determine the future driving lane based on the results of the maneuver recognition module. If the interaction module is not utilized to consider the collision probabilities between

vehicles, it would be determined that a lane change from 'Lane L' to 'Lane S' would be performed by the vehicle in the yellow box. However, since it is certain that the vehicles in 'Lane S' and 'Lane L' will stay in the current lane based on the likelihood from the maneuver recognition module, the interaction module determined that the situation in which all the vehicles remain in the current lane is the most appropriate among the 27 possible maneuver combinations of the three vehicles. In other words, it is judged that the risk caused by a lane change made by the left vehicle is much higher than the 61.4% likelihood for 'Lane S.' Therefore, the predicted maneuver of the left vehicle is 'Lane L', not 'Lane S,' which is the correct prediction as shown in the images taken by the dash cam. If the predictor makes a false prediction that the left vehicle in the yellow box will cut in, it is likely that the subject vehicle will unnecessarily decelerate or attempt to overtake in the left lane.

A similar situation is represented in Fig. 13 (b). The bus in 'Lane R' in the yellow box traveled close to the lane markers, as shown in the top view of the sensor outputs and recorded image of the dash cam, because the width of the bus is wider than that of the passenger cars. In this case, it is common to predict that there is a lane-change intention to the close lane. The proposed maneuver recognition module estimated a similar likelihood for 'Lane S' and 'Lane R' for the bus, even though the measurements from the LiDAR system are relatively accurate because the bus is in a close position. A prediction that the bus will change lanes from 'Lane R' to 'Lane S,' is unrealistic and the preceding vehicle and the bus will collide. However, the interaction module predicted that the bus will stay the current lane based on the estimated collision probability of the candidate maneuver combination.

The last case is that it is difficult to make a proper prediction with only the surrounding vehicle's states from environmental sensors due to a large perception error. As shown in Fig. 13 (c), the heading angle is erroneously perceived as pointing to the left even though the vehicles in the right lane are going straight. In particular, it was estimated that the vehicle in the right lane of the yellow box had a high likelihood, 67.0%, to change lanes to 'Lane S.' However, the interaction module predicted that the vehicle in the yellow box would not cut in and would stay the current lane by considering the four moving vehicles simultaneously.

These cases are failure cases of the maneuver recognition module due to multiple causes, in which a lane-keeping vehicle is predicted to change lanes. However, the interaction module, which considers the likelihood and predicted motion of all the surrounding vehicles simultaneously, determined that these maneuver recognition results are unrealistic. If only individual vehicles were considered without looking at the entire driving situation, the motion plan of AVs based on these maneuver recognition predictions is likely to be risky and uncomfortable for passengers and traffic participants. Therefore, the results of maneuver recognition are rejected, and the interactive motion predictor predicts that the targets will remain in the current lane. Through this finding, it can

be confirmed that the interactive motion predictor prevents the prediction of unrealistic situations and compensates for the incorrect prediction made by the maneuver recognition module.

VI. VEHICLE TEST RESULTS

The application results of the interactive motion predictor to a motion planning algorithm were summarized into results for a case study and the analysis of the entire vehicle test results. The prediction-based distance control algorithm was used to control the distance between the in-lane target and the subject vehicle [23]. Vehicle tests were conducted by implementing the proposed algorithm on an AV, which is described in Fig. 2. The motion predictor and motion planner are implemented in an industrial PC using the LabVIEW/MATLAB-based environment.

A. CASE STUDY OF THE MOTION PLANNING APPLICATION

The representative scenario is successive cut ins of the left- and right-lane vehicles when the AV is accelerating to reduce the clearance for the in-lane target. The prediction results with a dash cam image are depicted in Fig. 14, and the results for the crucial variables of the proposed case are summarized in Fig. 15. Fig. 14 shows the prediction results with the environmental sensor outputs in the top view on the left side. All the vehicles in the left figure show the currently recognized position, the prediction results for 2 seconds at 0.4-second intervals, and the likelihood of 'Lane S,' 'Lane L,' and 'Lane R.' The driving situations recorded by the dash cam are shown on the right side. Three important scenes from this case are presented in Fig. 14 (a) to (c). The details of each scene are discussed in this section.

Fig. 15 shows the longitudinal acceleration, velocity, clearance, lateral offset, time gap, warning index x , and TTC^{-1} . The clearance, lateral offset, time gap, warning index x , and TTC^{-1} are calculated using the closest in-lane target, which is classified by the prediction algorithm. Therefore, when the in-lane target is changed, discontinuous changes in the clearance, lateral offset, time gap, x , and TTC^{-1} appear. Since the vehicle tests were conducted by using the proposed algorithm, the conventional algorithm obtained the results by offline simulation based on the acquired driving data. In this comparison, the CTRV model was used as the conventional algorithm. Therefore, the longitudinal acceleration, clearance, and time gap history, which are depicted in Fig. 15 (a), (c), and (d), respectively, show the actual and desired values calculated in real time during the vehicle tests. In the case of the velocity, x , and TTC^{-1} history, the results between the AV and the in-lane target acquired during the vehicle tests are shown in Fig. 15 (b) and (e). The comparison of the proposed and conventional algorithms is represented in the lateral-offset history of the in-lane target, as shown in Fig. 15 (f). As mentioned before, the lateral-offset history of the conventional algorithm is acquired by offline simulation using the acquired driving data.

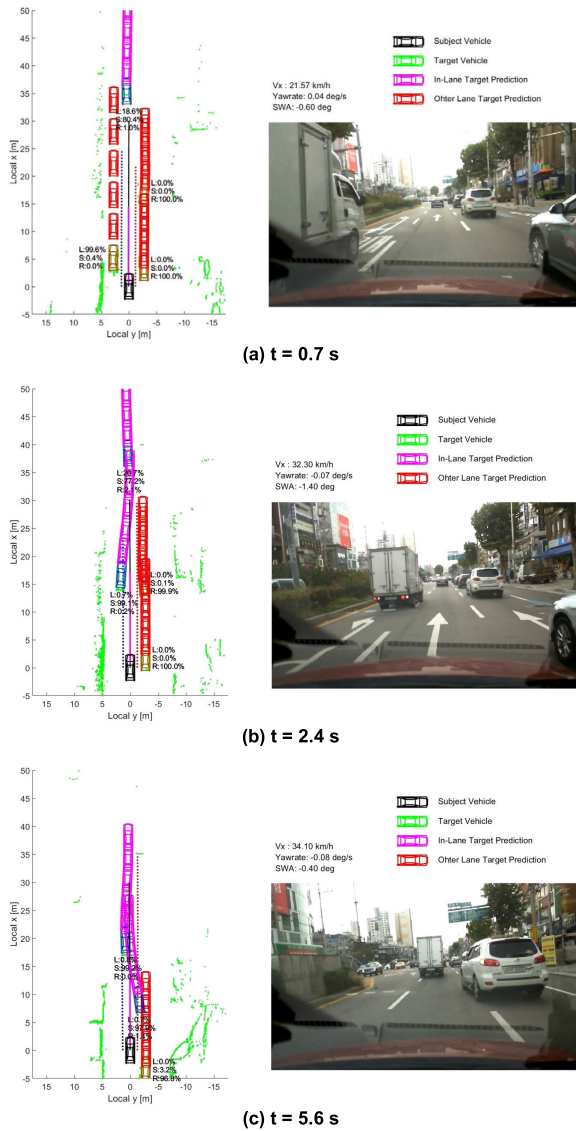


FIGURE 14. Snapshots of the vehicle test results for the two cut-in vehicles case.

The prediction results, which are represented in Fig. 14 (a), show the driving situations of the AV before the surrounding vehicles perform lane changes. At this moment ($t = 0.7s$), four surrounding targets are detected, showing that all vehicles stay in their current driving lanes. As shown in Fig. 14(a), it can be seen that each vehicle has the highest likelihood of the current driving lane, ranging from 80.4% to 100.0%. However, since the amount of longitudinal motion is different for each vehicle, the distances between the prediction results displayed at 0.4-second intervals are different. In this case, the motion planner decides to use maximum acceleration to follow the in-lane target because the surrounding vehicles do not have cut-in intentions. Therefore, the AV used a maximum acceleration of $1.5m/s^2$ to reduce the relative

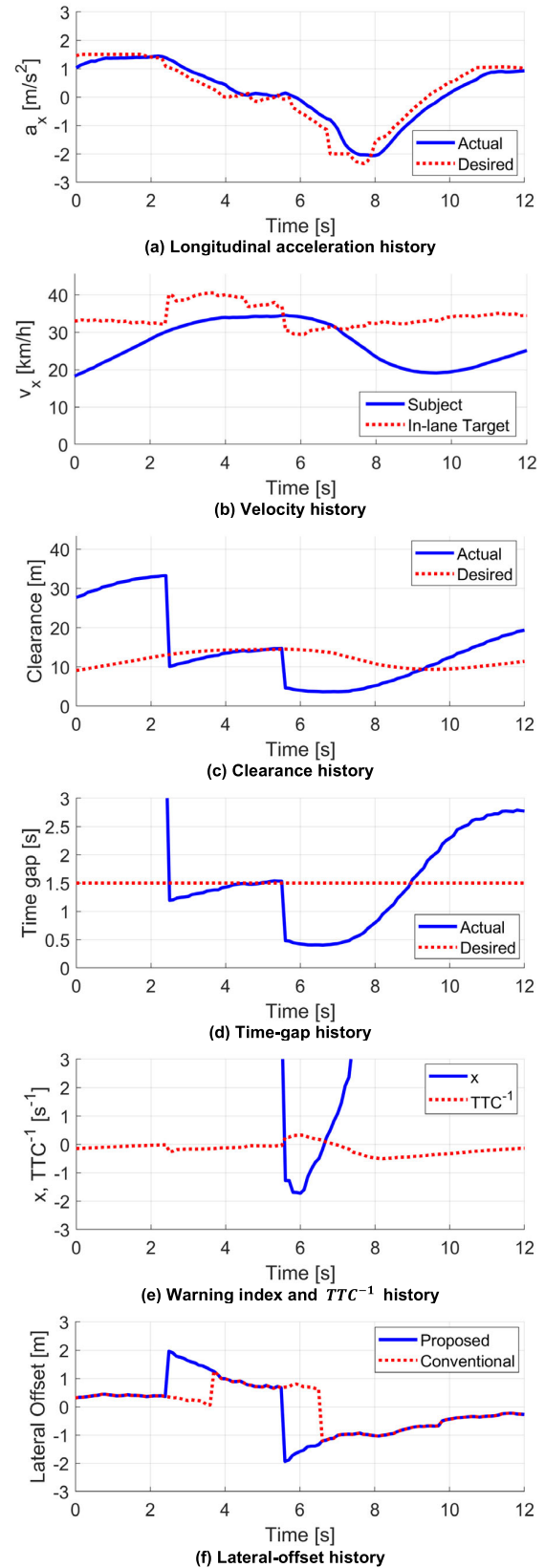


FIGURE 15. Vehicle test results.

velocity, clearance, and time gap and to the in-lane target until 2.4 seconds, as shown in Fig. 15 (a) to (d). Therefore, x and TTC^{-1} are maintained in safe areas, as depicted in Fig. 15(e). At this moment, since the in-lane target is clear, there is no difference in the lateral offset of the in-lane target between the proposed and conventional algorithms, as shown in Fig. 15 (f).

The truck in the left lane, which has the fastest velocity in Fig. 14 (a), cuts in after overtaking the subject vehicle at $t = 2.4s$. At this moment, the proposed predictor recognizes the lane-change intention 1.2 seconds earlier than the conventional predictor, as shown in Fig. 15 (f). The change in the in-lane target is revealed as a discontinuous change in the lateral offset, clearance, time gap, x , and TTC^{-1} at $t = 2.4s$. The lateral offset was approximately 2.0 m when first inferring the cut-in intention, which means that a cut-in intention is predicted before the truck crosses the lane markers, as shown in the dash cam image (Fig. 14 (b)). This early intention inference allows the AV to respond to a new in-lane target before the clearance and time gap decrease to risky levels. As shown in Fig. 15 (a) to (d), the AV smoothly reduces the acceleration and converges to the target clearance, time gap and in-lane target velocity. However, the conventional approach did not recognize the cut-in behavior until the lateral offset was reduced to 1.1 m, as shown in Fig. 15 (f). If the AV had been controlled by the conventional algorithm in the same situation, the AV would have accelerated further until it recognized the in-lane target, which would have caused more deceleration than the proposed algorithm and a risky situation would have occurred.

A similar situation also occurred when a sport utility vehicle (SUV) in the right lane cut in after the truck passed its position at $t = 5.6s$. The proposed predictor classifies the SUV as the in-lane target 1.0 seconds earlier than the conventional predictor does, as shown in Fig. 15. (f). At this moment, the SUV cuts in at a lower speed than the subject vehicle at a distance closer than the previous cut-in truck, which reduces the safety parameter to a dangerous level, as shown in Fig. 15 (c), (d), and (e). The earlier cut-in recognition manages the risk by applying smooth deceleration and slowing down below the velocity of the in-lane target, as shown in Fig. 15 (a) and (b). If the conventional algorithm is used to classify the targets, the risk is expected to increase to a level close to that of a collision. Because of the time gap at the first cut-in recognition, 0.5s is less than the delay of cut-in recognition of 1.0s.

Therefore, the proposed predictor can handle multi-traffic conditions while guaranteeing the safety of AVs by reducing the delay in the reaction to the behaviors of the surrounding vehicles. The clearance, time gap, x , and TTC^{-1} are properly managed even in situations where the surrounding vehicles cut in very close to the AV. In addition, an improvement in the target prediction reduces the use of sudden decelerations and improves ride comfort.

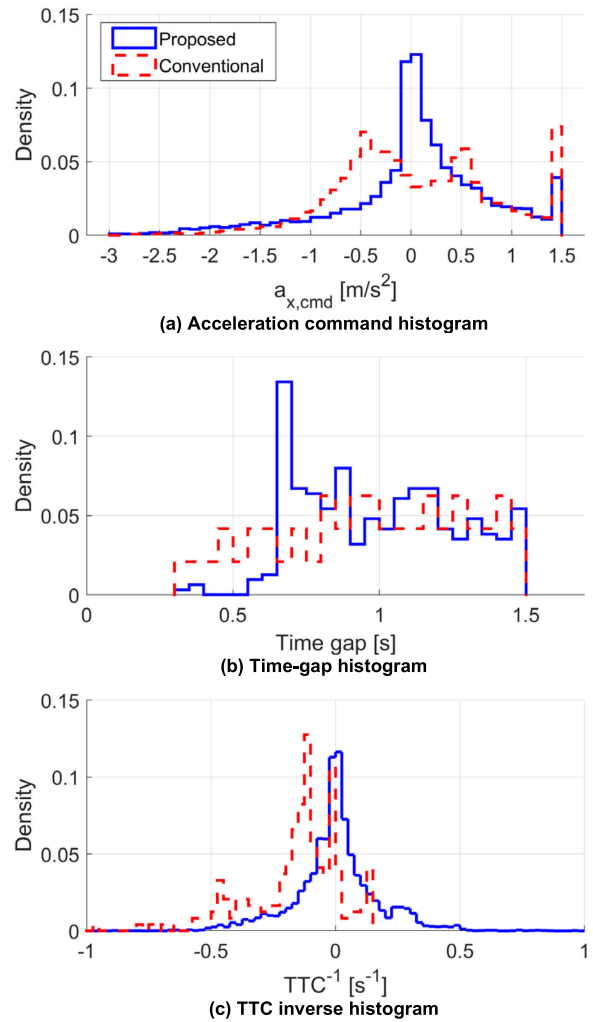


FIGURE 16. Histograms of the longitudinal acceleration command between the proposed and conventional algorithms.

B. STATISTICAL ANALYSIS OF THE MOTION PLANNING APPLICATION

A 53-minute automated vehicle test of the interactive motion predictor-based motion planner was conducted to evaluate the control effort and safety and compare those with those of the CTRV-based motion planner. For the CTRV-based motion planner, a 37-minute vehicle test was conducted. In total, 18,472 samples of in-lane target following and 114 cases of cut-in scenarios are extracted from the postprocessed vehicle test data. The histograms of the desired longitudinal accelerations of both algorithms are depicted in Fig. 16 (a). The desired acceleration of the proposed algorithm shows a bell curve with zero as the origin, which means that the proposed algorithm minimizes the control effort by precisely predicting the future behavior of the in-lane target as a human driver. However, the histogram of the conventional algorithm used more acceleration and deceleration to follow the in-lane target because the conventional motion predictor has a low

accuracy in predicting the acceleration and deceleration of the in-lane target. Therefore, the proposed algorithm reduces the control effort more than the conventional algorithm when the in-lane target exists.

The safety performance of the proposed algorithm based on two parameters is depicted in Fig. 16 (b) and (c): the time gap and TTC^{-1} , respectively. The time gap and TTC^{-1} are maintained in the safe region by the proposed algorithm. The time gap is maintained for more than 0.6 seconds, except for a single case of a dangerous cut-in behavior. In addition, TTC^{-1} shows the bell curve originating at zero, which means that the proposed algorithm properly manages the risks and tracks the desired states. However, the conventional algorithm manages the risk inappropriately, which makes the time gap and TTC^{-1} reduce to the risky level. In particular, the delay in the recognition of the cut-in intentions of the surrounding vehicles frequently caused a time gap of less than 0.6 seconds. In addition, TTC^{-1} is biased to a negative value because the conventional algorithm has difficulty predicting the deceleration, and a deceleration delay occurs accordingly. In short, the proposed algorithm can control the subject vehicle more safely and significantly reduce the control effort.

VII. CONCLUSION

An interactive motion predictor based on bidirectional long short-term memory (Bi-LSTM) was developed and evaluated by implementation in an autonomous vehicle (AV). The proposed predictor consists of three modules: the maneuver recognition, trajectory prediction, and interaction modules. The maneuver recognition module, which has been trained using 83,188 data samples collected by an AV in real traffic, estimates the maneuver likelihood. The trajectory prediction module based on the path-following model with motion parameter estimation predicts all possible trajectories for each maneuver. The interaction module considers the maneuver likelihood and the collision risk between the future trajectories to reduce the false prediction cases. The proposed predictor was evaluated in terms of its accuracy and its effects on the AV by data-based analysis and vehicle tests.

The evaluation results using 35,652 data samples with 486 lane-changing cases showed improved maneuver recognition and prediction accuracy. In particular, cut-in maneuver prediction to classify the in-lane target improved significantly compared to the performance of the constant turn rate and velocity (CTRV) model. The vehicle test results indicated that the proposed predictor can control the subject vehicle more safely than the CTRV model and reduce the control effort significantly. The time gap is maintained for more than 0.6 seconds, and TTC^{-1} shows a bell curve originating at zero, which means that the proposed algorithm properly manages the risks caused by cut-in targets. The desired acceleration from the proposed algorithm shows a bell curve with zero as the origin based on the precise prediction of the future behavior of the in-lane target. Future works in predicting motion of surrounding vehicles can be summarized in two aspects. The first aspect is the coverage extension of

the motion prediction algorithm. The coverage road will be extended from urban roads to highways. In addition, recognizable maneuvers will be increased to cope with various urban road conditions. For example, U-turning or turning at intersections will be covered in the future. The second aspect is learning the behavior of individual target vehicles in real time when AVs drive in urban environments with real traffic. Exploration of these topics is expected to substantially increase the safety and acceptance of autonomous vehicles by traffic participants on urban roads.

REFERENCES

- [1] R. Hoogendoorn, B. van Arem, and S. Hoogendoorn, "Automated driving, traffic flow efficiency, and human factors: Literature review," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2422, no. 1, pp. 113–120, Jan. 2014.
- [2] B. Sen, J. D. Smith, and W. G. Najm, "Analysis of lane change crashes," United States. Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. HS-809 571, Mar. 2003.
- [3] B. Sen, J. D. Smith, and W. G. Najm, "Examination of crash contributing factors using national crash databases," United States. Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. FHWA-JPO-05009, Oct. 2003.
- [4] A. Polychronopoulos, M. Tsogas, A. J. Amditis, and L. Andreone, "Sensor fusion for predicting vehicles' path for collision avoidance systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 549–562, Sep. 2007.
- [5] B. Kim and K. Yi, "Probabilistic and holistic prediction of vehicle states using sensor fusion for application to integrated vehicle safety systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2178–2190, Oct. 2014.
- [6] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Tokyo, Japan, Nov. 2013, pp. 4363–4369.
- [7] K. Driggs-Campbell, V. Govindarajan, and R. Bajcsy, "Integrating intuitive driver models in autonomous planning for interactive maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3461–3472, Dec. 2017.
- [8] T. Gindele, S. Brechtel, and R. Dillmann, "A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments," in *Proc. 13th Int. IEEE Conf. Intell. Transp. Syst.*, Funchal, Portugal, Sep. 2010, pp. 1625–1631.
- [9] M. Schreier, V. Willert, and J. Adamy, "An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2751–2766, Oct. 2016.
- [10] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Qingdao, China, Oct. 2014, pp. 334–341.
- [11] G. Xie, H. Gao, L. Qian, B. Huang, K. Li, and J. Wang, "Vehicle trajectory prediction by integrating physics-and maneuver-based approaches using interactive multiple models," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5999–6008, Feb. 2017.
- [12] P. Liu, A. Kurt, and U. Ozguner, "Trajectory prediction of a lane changing vehicle based on driver behavior estimation and classification," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Qingdao, China, Oct. 2014, pp. 942–947.
- [13] N. Deo, A. Rangesh, and M. M. Trivedi, "How would surround vehicles move? A unified framework for maneuver classification and motion prediction," *IEEE Trans. Intell. Vehicles*, vol. 3, no. 2, pp. 129–140, Jun. 2018.
- [14] J. Schlechtriemen, F. Wirthmueller, A. Wedel, G. Breuel, and K.-D. Kuhnert, "When will it change the lane? A probabilistic regression approach for rarely occurring events," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Seoul, South Korea, Jun. 2015, pp. 1373–1379.
- [15] D. Lenz, F. Diehl, M. T. Le, and A. Knoll, "Deep neural networks for Markovian interactive scene prediction in highway scenarios," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Los Angeles, CA, USA, Jun. 2017, pp. 685–692.
- [16] H. Woo, Y. Ji, H. Kono, Y. Tamura, Y. Kuroda, T. Sugano, Y. Yamamoto, A. Yamashita, and H. Asama, "Lane-change detection based on vehicle-trajectory prediction," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1109–1116, Apr. 2017.

- [17] R. Izquierdo, I. Parra, J. Munoz-Bulnes, D. Fernandez-Llorca, and M. A. Sotelo, "Vehicle trajectory and lane change prediction using ANN and SVM classifiers," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Yokohama, Japan, Oct. 2017, pp. 1–6.
- [18] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Changshu, China, Jun. 2018, pp. 1179–1184.
- [19] B. Kim, C. M. Kang, J. Kim, S. H. Lee, C. C. Chung, and J. W. Choi, "Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Yokohama, Japan, Oct. 2017, pp. 399–404.
- [20] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1468–1476.
- [21] H. Lee, S. Kim, S. Park, Y. Jeong, H. Lee, and K. Yi, "AVM/LiDAR sensor based lane marking detection method for automated driving on complex urban roads," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Los Angeles, CA, USA, Jun. 2017, pp. 1434–1439.
- [22] K. Suzuki, "An analysis of driver's steering behaviour during auditory or haptic warnings for the designing of lane departure warning system," *JSAE Rev.*, vol. 24, no. 1, pp. 65–70, Jan. 2003.
- [23] B. Kim, D. Kim, S. Park, Y. Jung, and K. Yi, "Automated complex urban driving based on enhanced environment representation with GPS/map, radar, lidar and vision," *IFAC-PapersOnLine*, vol. 49, no. 11, pp. 190–195, 2016.



YONGHWAN JEONG received the B.S. and Ph.D. degrees in mechanical engineering from Seoul National University, South Korea, in 2014. His research interests include sensor fusion, risk assessment, automated vehicle motion planning on urban roads, automated vehicle control, intersection assistance systems, driver intention inference, and state prediction.



KYONGSU YI (Member, IEEE) received the B.S. and M.S. degrees in mechanical engineering from Seoul National University, South Korea, in 1985 and 1987, respectively, and the Ph.D. degree in mechanical engineering from the University of California, Berkeley, in 1992. He is currently a Professor with the School of Mechanical and Aerospace Engineering, Seoul National University. His research interests include control systems, driver assistant systems, active safety systems, and automated driving of ground vehicles. He is the Chair of the KSME IT Fusion Technology Division. He serves as a member of the Editorial Board for *Mechatronics*.

...