IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 10, 2021, accepted April 13, 2021, date of publication April 26, 2021, date of current version May 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3075677

Interaction-Aware Probabilistic Trajectory Prediction of Cut-In Vehicles Using Gaussian Process for Proactive Control of Autonomous Vehicles

YOUNGMIN YOON[1], CHANGHEE KIM[1], JONGMIN LEE[1,2], AND KYONGSU YI[1], (Member, IEEE)

[1]Department of Mechanical and Aerospace Engineering, Seoul National University, Seoul 08826, South Korea
[2]Smart Mobility Lab, SML International Company Ltd., Seoul 08826, South Korea

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

This work was supported in part by the Technology Innovation Program (Development and Evaluation of Automated Driving Systems for Motorways and City Roads) funded by the Korean Ministry of Trade, Industry and Energy, South Korea, under Grant 10079730, 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, in part by the Technology Innovation Program (Development of AI-Based Autonomous Computing Modules and Demonstration of Services) funded by the Ministry of Trade, Industry and Energy, South Korea, under Grant 20005705, in part by the Institute of Engineering Research, Seoul National University, South Korea, and in part by the Institute of Advanced Machinery and Design, Seoul National University (SNU-IAMD), South Korea.

ABSTRACT This paper presents a probabilistic trajectory prediction of cut-in vehicles exploiting the information of interacting vehicles. First, a probability distribution of behavioral parameters, which represents the characteristics of lane-change motion, is obtained via Gaussian Process Regression (GPR). For this purpose, Gaussian Process (GP) models are trained using real-world trajectories of lane-changing vehicles and adjacent vehicles. Subsequently, the future states of the lane-change vehicle are probabilistically estimated using a path-following model, which introduces virtual measurements based on the information of behavioral parameters. The proposed predictor is applied to the motion planning and control of autonomous vehicles. A Model Predictive Control (MPC) is designed to achieve predictive maneuvering of autonomous vehicles against cut-in preceding vehicles. The proposed predictor has been evaluated in terms of its prediction accuracy. Also, the performance of the proposed predictor-based control has been validated via computer simulations and autonomous driving vehicle tests. Compared to conventional prediction methods, it is shown that the interaction-aware proposed predictor provides improved prediction of cut-in vehicles' motion in multi-vehicle scenarios. Furthermore, the control results indicate that the proposed predictor helps the autonomous vehicle to reduce the control effort and improve ride quality for passengers in cut-in scenarios, while guaranteeing safety.

INDEX TERMS Autonomous vehicle, autonomous driving, Gaussian process (GP), machine learning, vehicle trajectory prediction, interaction-aware motion prediction, model predictive control.

I. INTRODUCTION

Autonomous driving technology has been rapidly developed as the performance of electronic and mechanical devices and computer software improves [1]. Taking advantage of the development of the sub-modules in autonomous vehicles, researchers have tried to develop algorithms that enable autonomous vehicles to perform various tasks. Autonomous vehicles are expected to provide convenience to drivers and ensure safety through these technological advances. Moreover, the role of autonomous driving performance is important in situations in which safety is greatly affected by the driver's attention and maneuverability. In particular, response to lane-changing vehicles is one of the situations in which many accidents occur due to the driver's insufficient judgment or lack of attention [2]. Transportation statistics from

The associate editor coordinating the review of this manuscript and approving it for publication was Jason Gu.

the United States reveal that accidents related to lane change account for between 4% and 10% of total accidents in the country [3], [4]. Analysis of traffic accident data in the Netherlands shows that the proportion of accidents caused by lane changes was 12.6% of the total [5]. Accidents from lane changes affect traffic flow on roads by causing 10% of latencies [6]. Therefore, resolving the cut-in vehicle response problem through autonomous driving is expected to contribute significantly to improving safety and reducing passengers' anxiety.

Autonomous driving algorithms can be divided into environment perception, decision-making, and control in functional units [7]. In the field of decision-making and control, it is essential to predict the motion of surrounding vehicles in order to evaluate the risk of collision and establish a behavioral plan. Many studies have dealt with the control methodology of autonomous driving by reflecting the prediction results of surrounding vehicles [8]–[11]. These studies have reported that the application of the prediction algorithm to the control helps to improve safety and establish natural behavior. Because the accuracy of the prediction result has an influence on the control strategy, accurate motion prediction is required to prevent threats to safety or sudden changes in an automated vehicle's motion.

Many researchers have devised motion prediction algorithms using a variety of methods. Several works have used physics-based motion models which assume constant physical variables within the prediction horizon: Constant Velocity (CV) model [12], Constant Acceleration (CA) model [13], [14], Constant Turn Rate and Velocity (CTRV) model [15], and Constant Turn Rate and Acceleration (CTRA) model [16]. Although these approaches allow for simplification of the model and low computational cost, the prediction results are not suitable for long-term prediction due to an inability to capture motion pattern changes within the wide prediction horizon. To consider the variability of physical parameters, researchers have mixed the simple physics-based models using the Interactive Multiple Model (IMM) approach while assuming the transition probability matrix with respect to the motion modes [17].

A lot of research has defined a set of maneuvers that represents the driving situations semantically. To derive the prediction results at the trajectory level, the motion model is defined separately for each maneuver. Lane-keeping and lane-changing maneuvers have been mainly treated to construct the maneuver-based prediction model [18]–[20]. In the case of lane-change maneuvers, assumption of the behavioral rule has been widely applied to several works. To model the predictive trajectory of a lane change, a sinusoidal function [21], [22] and a quintic function [23] have been suggested with the constraints related to the current states and final states. A feedback control approach has been used to imitate the drivers' tracking motion of the desired centerline [24]. A potential field-based approach has been applied to generate the predictive path of surrounding vehicles considering the adjacent obstacles [25]. In the rule-based prediction model, however, fine assumption of the behavioral rule is required to secure the reliable prediction performance, which requires many parameters related to the motion model. Furthermore, advancement of the behavioral rule that describes the general drivers' motion requires complex motion assumptions, which increases the difficulty of parameter tuning.

Based on the requirement of the human-driven model, machine learning has been widely applied to the prediction of driving motion. Trajectory prediction methods based on soft clustering have been suggested to probabilistically model the conditional distribution of future trajectories given past trajectories utilizing a Gaussian Mixture Model (GMM) or a Variational Gaussian Mixture Model (VGMM) [26], [27]. A Gaussian process has been introduced to predict the probability distribution of vehicles' future positions based on velocity field modeling on spatial space [28], [29]. A Radial Basis Function Network (RBFN) has been used to derive a quintic polynomial approximation of predicted positions [30]. A Variational Autoencoder (VAE) has been applied to learn the distribution of a latent variable that is used to generate the future trajectories of vehicles [31]. However, the aforementioned methods have limitations in terms of considering the interaction among surrounding vehicles in the trajectory-level prediction because only the information about individual vehicles is reflected to model the corresponding drivers' pattern. Many researchers have tried to model the interaction in the traffic environment. Information about nearby vehicles has been concatenated in input features of the learning framework [32]–[34]. Convolutional social pooling has been applied to Long Short-Term Memory (LSTM) for learning interdependencies between vehicles [35]. However, these approaches require the observation of a large number of surrounding vehicles, which is usually not available in automated vehicles based on local sensors due to occlusion and the inherent limit of perception range.

This study focuses on improving the trajectory-level prediction accuracy of lane-changing vehicles and performing suitable control against cut-in maneuvers of the side lane vehicle. The proposed predictor applies a machine learning-based approach to estimate the crucial parameters of lane-change motion. The future states of cut-in vehicles are probabilistically predicted by applying a path-following model. The probability distribution of future states is applied to the control of an autonomous vehicle to proactively react to the cut-in vehicle. Vehicle tests have been conducted to evaluate the improvement of control performance allowed by accurate prediction results.

The main contributions of this work are summarized as follows:

1) Lane-change motion of cut-in vehicles is predicted with consideration of the interaction among surrounding vehicles.
2) Gaussian Process Regression (GPR) is introduced to probabilistically estimate the dominant parameters that represent lane-change behavior.

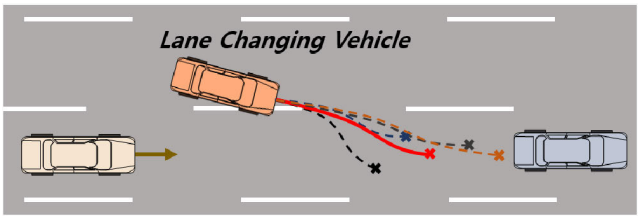


FIGURE 1. Multiple possible trajectories of a lane-changing vehicle. The most probable trajectory is depicted in red.

3) Impact of predictive performance improvement on control performance is validated by control simulation tests.
4) Applicability of the proposed predictor-based proactive control is validated by vehicle tests based on autonomous driving in the real world.

II. OVERVIEW OF THE PROPOSED PREDICTOR

This study has focused on trajectory-level prediction of cut-in vehicles in multi-vehicle traffic, as shown in Fig. 1. For long-term trajectory prediction, a model that reflects the characteristics of cut-in behavior should be designed rather than simple physics-based models. An important factor in determining the characteristics of cut-in motion is the drivers' behavioral aggression to complete the lane-change maneuver [36]. Because the lane-change behavior of cut-in vehicles is determined by the influence of interaction with surrounding vehicles, the reference information for the predictor should include information about interacting vehicles. If the prediction is performed without awareness of interaction, unreliable prediction results can be derived. For example, in dense traffic, a predictor without consideration of interaction may provide the prediction result that a cut-in vehicle collides with a front or rear vehicle in the target lane. Thus, we introduce the cut-in vehicle trajectory predictor in which the inter-vehicle interaction is implicitly modeled to enhance the reliability and accuracy of the prediction results.

The cut-in vehicle trajectory predictor is subdivided into two main parts: behavioral parameter estimation and vehicle state prediction, as shown in Fig. 2. The behavioral parameter estimation module provides a probability distribution of behavioral parameters of the lane-changing vehicle utilizing a Gaussian Process Regression (GPR). To reflect the interaction with surrounding vehicles, GP models are trained using information about the lane-changing vehicle as well as the nearby vehicles in the target lane. The estimated behavioral parameters implicitly represent the driving style of the lane-changing vehicle. The vehicle state prediction module estimates the sequence of future states using Extended Kalman Filter (EKF) with the path-following model. The behavioral parameters estimated from GPR are utilized to generate virtual measurements in EKF. Therefore, the proposed predictor provides the outputs consisting of time series data of future states and future uncertainties of the cut-in vehicle, using the inputs composed of the current states of the cut-in vehicle and adjacent vehicles in the target lane. The future information of the cut-in vehicle is utilized in the predictive control of autonomous vehicles, which is described in Section V.

III. BEHAVIORAL PARAMETER ESTIMATION

The behavioral parameters of cut-in motion have been estimated using the GPR method, which is a data-driven regression method in a supervised manner. The features of the GP model have been selected considering the representativeness of the cut-in driving style and surrounding traffic. The mapping function between the input and output features has been learned by training the GP model. The details are described in the following sections.

A. FEATURE DEFINITION

The selection of features for supervised regression is an important factor in understanding the situation and properly extracting the desired property. Therefore, the input features should include all the information that is necessary to recognize the situation at the time of the query. The output features must contain other properties of the situation that can be properly utilized to run the following module.

The description of the input features for behavioral parameter estimation is shown in Fig. 3. The input features have been selected with two considerations: lane-changing progress of the cut-in vehicle and relative configuration of the interacting vehicles. Both are represented by the physical properties of the cut-in vehicle and nearby vehicles in the target lane, respectively. The vector of the input features is defined as follows:

$$\mathbf{x}_{input} = \left[e_{y,\text{target}} \;\; e_{\theta,\text{target}} \;\; v_x \;\; p_{x,rel,\text{ft}} \;\; v_{x,\text{ft}} \;\; p_{x,rel,\text{rt}} \;\; v_{x,\text{rt}} \right]^T, \tag{1}$$

where subscripts ft and rt mean the front vehicle and the rear vehicle in the target lane, respectively; $e_{y,\text{target}}$ and $e_{\theta,\text{target}}$ are the lateral offset and heading offset of the cut-in vehicle with respect to the target centerline, respectively; v_x is the longitudinal velocity of the cut-in vehicle; $p_{x,rel,\text{ft}}$ and $p_{x,rel,\text{rt}}$ are the relative position of the interacting vehicles with respect to the cut-in vehicle; and $v_{x,\text{ft}}$ and $v_{x,\text{rt}}$ are the longitudinal velocity of the interacting vehicles. $e_{y,\text{target}}, e_{\theta,\text{target}}$, and v_x represent the lane-changing progress of the cut-in vehicle. $p_{x,rel,\text{ft}}$, $p_{x,rel,\text{rt}}$, $v_{x,\text{ft}}$, and $v_{x,\text{rt}}$ represent the relative configuration of the interacting vehicles.

The output features, which are the behavioral parameters of the cut-in vehicle to be estimated, have been selected as dominant physical properties to represent the lane-change aggressiveness, as depicted in Fig. 4. The vector of the output features is determined as follows:

$$\mathbf{y}_{output} = f(\mathbf{x}_{input}) = \left[s_{LC} \;\; e_{y,f,LC} \;\; t_{LC} \right]^T, \tag{2}$$

where f is the mapping function between the input and output features; s_{LC} is the longitudinal distance remaining to complete the lane-change maneuver; $e_{y,f,LC}$ is the lateral offset with respect to the target centerline at the completion of the

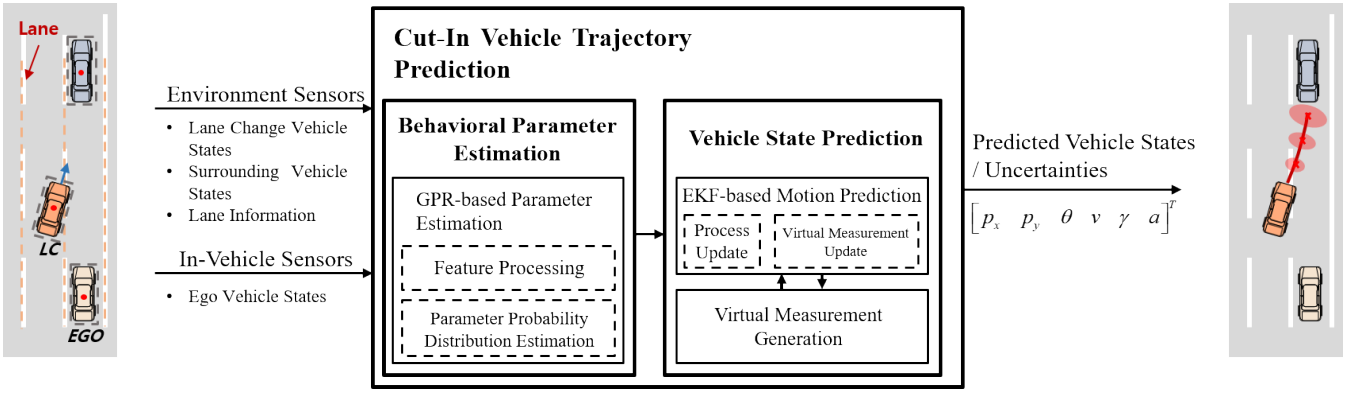


FIGURE 2. Overall architecture of the proposed cut-in vehicle trajectory prediction algorithm.

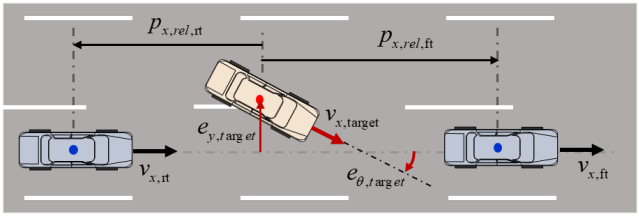


FIGURE 3. Description of input features for behavioral parameter estimation.

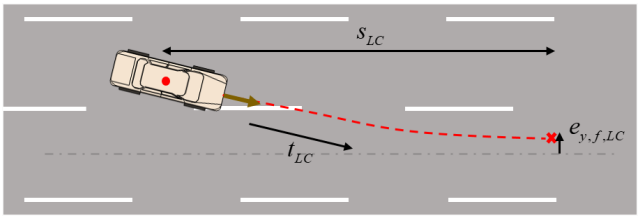


FIGURE 4. Description of behavioral parameters representing the lane-change behavior.

lane change; and t_{LC} is the time left to complete the lane change. s_{LC} and $e_{y,f,LC}$ represent the final position of the lane-change maneuver.

B. GPR-BASED PARAMETER ESTIMATION

A GP is a collection of random variables that has a multivariate Gaussian distribution [37]. The distribution of function f is modeled by GP, which considers a set of function values $[f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_N)]$ as the random variables. The formulation of GP is written as follows:

$$f \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{3}$$

where $\mathbf{x}$ is a feature vector; and $m(\mathbf{x})$ and $k(\mathbf{x}, \mathbf{x}\prime)$ are a mean function and a kernel function, respectively. The correlation between the function values is approximately modeled by the kernel function, which measures similarity between the corresponding input features.

A GPR, a GP-based regression method, is a non-parametric supervised learning method utilizing the Bayesian approach. A GPR efficiently models the predictive probability distribution of the function value $f(\mathbf{x}_*)$ given a query input $\mathbf{x}_*$ and training dataset, which is defined as follows:

$$\mathcal{D} = \{(\mathbf{x}_i, f(\mathbf{x}_i))|\ i = 1, 2, \cdots, N\}, \tag{4}$$

where N denotes the number of the training samples.

A joint distribution of $\mathbf{f}$ and $f(\mathbf{x}_*)$ is determined as follows:

$$\begin{bmatrix} \mathbf{f} \\ f(\mathbf{x}_*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\mathbf{X}) \\ m(\mathbf{x}_*) \end{bmatrix}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I} & K(\mathbf{X}, \mathbf{x}_*) \\ K(\mathbf{x}_*, \mathbf{X}) & k(\mathbf{x}_*, \mathbf{x}_*) \end{bmatrix}\right), \tag{5}$$

where σ_n is the standard deviation of the observation noise; $\mathbf{X}$ is the collection of the training input dataset and $\mathbf{f}$ is the collection of the training output dataset; $K(\mathbf{X}, \mathbf{X})$ is the $N \times N$ matrix where $[K(\mathbf{X}, \mathbf{X})]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$; $K(\mathbf{X}, \mathbf{x}_*)$ is the $N \times 1$ matrix where $[K(\mathbf{X}, \mathbf{x}_*)]_i = k(\mathbf{x}_i, \mathbf{x}_*)$; and $K(\mathbf{x}_*, \mathbf{X})$ is the transpose matrix of $K(\mathbf{X}, \mathbf{x}_*)$. The predictive distribution of the function value $f(\mathbf{x}_*)$ is derived as the following Gaussian distribution:

$$f(\mathbf{x}_*)|\mathcal{D}, \quad \mathbf{x}_* \sim \mathcal{N}\left(\mu_*, \sigma_*^2\right), \tag{6}$$

where

$$\begin{aligned} \mu_* &= m(\mathbf{x}_*) + K(\mathbf{x}_*, \mathbf{X})\left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1}(\mathbf{f} - m(\mathbf{X})), \\ \sigma_*^2 &= k(\mathbf{x}_*, \mathbf{x}_*) - K(\mathbf{x}_*, \mathbf{X})\left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} K(\mathbf{X}, \mathbf{x}_*). \end{aligned} \tag{7}$$

In this paper, three GP models are designed and trained to estimate the distribution of the behavioral parameters, s_{LC}, $e_{y,f,LC}$, and t_{LC}. The models are represented as follows:

$$\begin{aligned} f_{s_{LC}} &\sim \mathcal{GP}_{s_{LC}}(m_{s_{LC}}(\mathbf{x}), k_{s_{LC}}(\mathbf{x}, \mathbf{x}')), \\ f_{e_{y,f,LC}} &\sim \mathcal{GP}_{e_{y,f,LC}}(m_{e_{y,f,LC}}(\mathbf{x}), k_{e_{y,f,LC}}(\mathbf{x}, \mathbf{x}')), \\ f_{t_{LC}} &\sim \mathcal{GP}_{t_{LC}}(m_{t_{LC}}(\mathbf{x}), k_{t_{LC}}(\mathbf{x}, \mathbf{x}')). \end{aligned} \tag{8}$$

For all GP models, this work utilizes the affine mean function and Matern 3/2 kernel function with an automatic relevance determination (ARD) structure as shown below:

$$m(\mathbf{x}) = \alpha \mathbf{x} + \beta,$$

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \left(1 + \sqrt{3\,\|\mathbf{x} - \mathbf{x}'\|^2_{\mathbf{M}^{-1}}}\right) \times \exp\left(-\sqrt{3\,\|\mathbf{x} - \mathbf{x}'\|^2_{\mathbf{M}^{-1}}}\right), \quad (9)$$

where α and β are the coefficients of the affine mean function; σ_f is the standard deviation of the signal; and $\mathbf{M}$ is the positive-definite diagonal matrix that implicitly determines the relevance of each dimension of the input feature vector. Constructing the GP models, the predictive distribution of each parameter is obtained as follows:

$$\begin{aligned} s_{LC}|\mathcal{D}_{s_{LC}}, \quad \mathbf{x}_* &\sim \mathcal{N}\left(\hat{s}_{LC}, \hat{\sigma}^2_{s_{LC}}\right), \\ e_{y,f,LC}|\mathcal{D}_{e_{y,f,LC}}, \quad \mathbf{x}_* &\sim \mathcal{N}\left(\hat{e}_{y,f,LC}, \hat{\sigma}^2_{e_{y,f,LC}}\right), \\ t_{LC}|\mathcal{D}_{t_{LC}}, \quad \mathbf{x}_* &\sim \mathcal{N}\left(\hat{t}_{LC}, \hat{\sigma}^2_{t_{LC}}\right). \end{aligned} \quad (10)$$

The parameters $\boldsymbol{\theta}_h = \{\alpha, \beta, \sigma_f, \sigma_n, \mathbf{M}\}$ are regarded as hyperparameters of the GP model for definition of the mean function and kernel function. Optimizing the hyperparameters is required to enhance the estimation performance of GPR. The optimal parameters are obtained by maximizing the log likelihood in the GP model, as given below:

$$\begin{aligned} \hat{\boldsymbol{\theta}}_h &= \arg\max_{\theta_h} \log p(\mathbf{f}|\mathbf{X}, \theta_h) \\ &= \arg\max_{\theta_h} -\frac{1}{2}\log\left|2\pi(K(\mathbf{X}, \mathbf{X}) + \sigma_n^2\mathbf{I})\right| \\ &\quad -\frac{1}{2}(\mathbf{f} - m(\mathbf{X}))^T(K(\mathbf{X}, \mathbf{X}) + \sigma_n^2\mathbf{I})^{-1}(\mathbf{f} - m(\mathbf{X})). \end{aligned} \quad (11)$$

This optimization problem is solved offline by using an L-BFGS algorithm, which efficiently approximates the inverse Hessian matrix in the quasi-Newton method [38].

C. TRAINING GP MODELS

The GP models for the behavioral parameters have been constructed using the dataset consisting of the lane-changing trajectories in the real world. The Next Generation Simulation (NGSIM) database, a public dataset established by the U.S. Federal Highway Administration, has frequently been used to analyze the traffic characteristics and microscopic behavior of the participants [39]. In this paper, the I-80 and US-101 dataset from the NGSIM have been utilized to generate the training and testing dataset. Both datasets contain 45 minutes of traffic data, respectively.

The trajectory sets, including 537 events of cut-in maneuvers, have been extracted from the NGSIM database and pre-processed using a Kalman filter [40]. The trajectories of the cut-in vehicles as well as the front and rear vehicles in the target lanes have been used to prepare the dataset. The target lane for each lane-change maneuver has been determined as the lane to which the cut-in vehicle finally completes the lane change. The output features of the datasets have been determined by marking the final points of the cut-in maneuvers. To determine the final points, we have obtained the point when the cut-in vehicle is in the target lane and heading offset with respect to the target centerline reaches below a threshold θ_{bound} [33]. In this study, θ_{bound} of 2 deg is used.

TABLE 1. Statistics of estimation error of the behavioral parameters.

Error	Left Lane Change		Right Lane Change	
	MAE	RMSE	MAE	RMSE
s_{LC} (m)	1.5182	3.2954	0.3255	1.0358
$e_{y,f,LC}$ (m)	0.0604	0.1261	0.0085	0.0267
t_{LC} (s)	0.1980	0.4251	0.0294	0.0807

To construct the GP models, 4000 input-output pairs for left lane change and right lane change have been randomly selected. The two selected datasets have been used to train the GP models for the two lane changes, respectively. In Table 1, the behavioral parameter estimation performance of the trained GP models is shown with error statistics in terms of Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) derived by the testing data pairs. The larger errors of the left lane-change models compared to the right lane-change models are due to the larger portion of overtaking cases in the dataset, which make the lane-change behavior dynamic. Meanwhile, the overall errors of both models are bounded within the reliable level. Therefore, the error statistics show that the feature configuration of the GP models is suitable to derive the approximation of the behavioral parameters.

IV. VEHICLE STATE PREDICTION

The future states of the cut-in vehicle are predicted based on the EKF approach. The path-following model is utilized to generate virtual measurements of the EKF. The probability distribution of the behavioral parameters is considered to determine the values and noise properties of the virtual measurements. Therefore, the vehicle state prediction module probabilistically predicts the time series data of the future states by repeating the EKF.

A. EKF-BASED MOTION PREDICTION

An EKF is designed to probabilistically predict the sequence of the future motion of the cut-in vehicle by modeling the vehicle behavior. The vehicle state vector and measurement vector of the EKF at predictive time step k are defined as:

$$\begin{aligned} \mathbf{x}_k &= \begin{bmatrix} p_{x,k} & p_{y,k} & \theta_k & v_k & \gamma_k & a_k \end{bmatrix}^T, \\ \mathbf{z}_k &= \begin{bmatrix} \gamma_{k,virtual} & a_{k,virtual} \end{bmatrix}^T, \end{aligned} \quad (12)$$

where p_x and p_y are the x position and y position, respectively; θ is the heading angle; v is the absolute velocity; γ is the yaw rate; a is the longitudinal acceleration; and subscript *virtual* implies the virtual measurements derived by the path-following model described in Section IV. B. The recursive update of the future vehicle states is derived as the following process update model:

$$p_{x,k+1} = p_{x,k} + v_k \cos\theta_k \cdot \Delta t$$

$$+(a_{x,k}\cos\theta_k - \gamma_k v_k \sin\theta_k)\cdot\frac{(\Delta t)^2}{2} + w_{k,1},$$

$$p_{y,k+1} = p_{y,k} + v_k \sin\theta_k \cdot \Delta t + (a_{x,k}\sin\theta_k + \gamma_k v_k \cos\theta_k)\cdot\frac{(\Delta t)^2}{2} + w_{k,2},$$

$$\theta_{k+1} = \theta_k + \gamma_k \cdot \Delta t + w_{k,3},$$

$$v_{k+1} = v_k + a_k \cdot \Delta t + w_{k,4},$$

$$\gamma_{k+1} = \gamma_k - (k_\gamma \gamma_k)\cdot\Delta t + (k_\gamma^2\gamma_k)\cdot\frac{(\Delta t)^2}{2} + w_{k,5},$$

$$a_{k+1} = a_k - (k_a a_k)\cdot\Delta t + (k_a^2 a_k)\cdot\frac{(\Delta t)^2}{2} + w_{k,6}, \quad (13)$$

$$\begin{bmatrix} w_{k,1} & w_{k,2} & \dots & w_{k,6} \end{bmatrix}^T \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k), \quad (14)$$

where Δt is the sampling time of 0.1s; k_γ and k_a are the decaying rates of the yaw rate and longitudinal acceleration, respectively; $w_1, w_2, \dots, w_6$ denote the process noise terms of each element of the state vector; and $\mathbf{Q}$ is the covariance matrix of the process noise vector. The assumption of decaying yaw rate and longitudinal acceleration is used to represent the stabilizing behavior of the vehicle and enhance the prediction accuracy. $\mathbf{Q}$ is derived by the results of the statistical analysis of the sensor noise [24]. The process update model described in equation (14) could be summarized as follows:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{w}_k, \quad (15)$$

where $\mathbf{f}$ is the function of the process update model; and $\mathbf{w}$ is the process noise vector including $w_1, w_2, \dots, w_6$. The measurement update model is given as follows:

$$\gamma_{k,virtual} = \gamma_k + n_{k,1},$$

$$a_{k,virtual} = a_k + n_{k,2}, \quad (16)$$

$$\begin{bmatrix} n_{k,1} & n_{k,2} \end{bmatrix}^T \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k), \quad (17)$$

where n_1 and n_2 denote the measurement noise terms of the virtual yaw rate and longitudinal acceleration, respectively; and $\mathbf{R}$ is the covariance matrix of the measurement noise vector. The estimated state vector $\hat{\mathbf{x}}$ and its error covariance $\mathbf{P}$ are derived by following the procedures of the EKF, which obtains the optimal solution in the minimum mean-square-error sense [40]. The recursive estimation of the EKF is briefly described as follows:

$$\bar{\mathbf{x}}_k = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}),$$

$$\mathbf{M}_k = \mathbf{F}_{k-1}\mathbf{P}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1},$$

$$\mathbf{F}_{k-1} = \left.\frac{\partial \mathbf{f}_{k-1}(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1}}, \quad (18)$$

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}_k\bar{\mathbf{x}}_k),$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{M}_k,$$

$$\mathbf{H}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{K}_k = \mathbf{M}_k\mathbf{H}_k^T(\mathbf{H}_k\mathbf{M}_k\mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \quad (19)$$

where the initial conditions, $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$, are defined as the probability distribution of the perceived vehicle at the current time step. Therefore, the EKF obtains the sequence of the vehicle state vector $\{\hat{\mathbf{x}}_k | \ k = 1, 2, \dots, N_p\}$ and error covariance $\{\mathbf{P}_k | k = 1, 2, \dots, N_p\}$, where N_p is the prediction horizon.

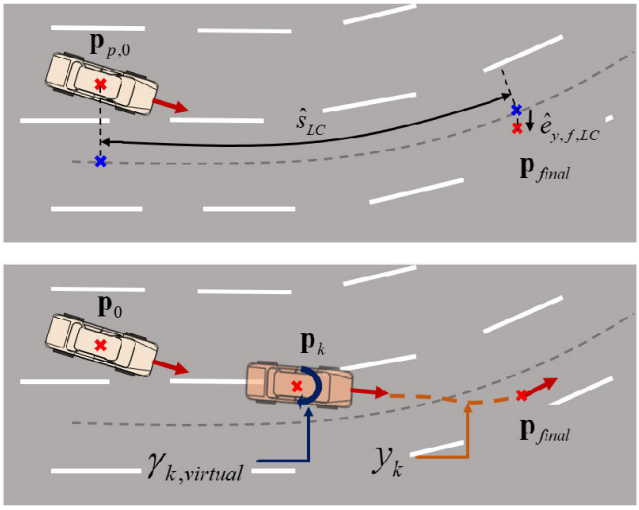


FIGURE 5. Derivation of virtual yaw rate of cut-in vehicle using behavioral parameters.

B. VIRTUAL MEASUREMENT GENERATION

Determining the virtual measurements has a major influence on the prediction performance of the proposed structure. The path-following model, which assumes that the driver may follow the target lane, is utilized to generate the virtual measurements. In this study, the behavioral parameters, s_{LC}, $e_{y,f,LC}$, and t_{LC}, are applied to the path-following model and used to derive the values and noise properties of the virtual measurements: yaw rate and longitudinal acceleration.

As depicted in Fig. 5, the virtual yaw rate is derived based on the design of a curve that reaches the final point of the lane change. First, the final point of the lane change is determined in the following sense:

1) The target vehicle travels from the current position to the lane-change final position with an arc length of s_{LC} based on the road coordinate.
2) The final point deviates from the centerline by the lateral offset of $e_{y,f,LC}$.

The derivation of the final position point $\mathbf{p}_{final} = [p_{x,final}, p_{y,final}]^T$ is formulated as given:

$$\int_{p_{x,0}}^{p_{x,final}} \sqrt{1 + (2a_2 \cdot x + a_1)^2}\,dx = \hat{s}_{LC},$$

$$p_{y,final} = \hat{e}_{y,f,LC} + \left\{a_2 \cdot p_{x,final}^2 + a_1 \cdot p_{x,final} + a_0 + W_{road} \cdot N_{lane}\right\}, \quad (20)$$

where $p_{x,0}$ is the current longitudinal position; a_0, a_1, and a_2 are the zeroth, first, and quadratic coefficients of the quadratic polynomial that models the geometry of the road centerline; W_{road} is the width of the road lane; and N_{lane} is the integer lane index of the lane-change target lane. Subsequently,

the trajectory y_k is modeled at the predictive time step k using a cubic polynomial curve which is a twice-differentiable function. The polynomial curve has the constraint that it passes through points $\mathbf{p}_k$ and $\mathbf{p}_{final}$ while the tangents at $\mathbf{p}_k$ and $\mathbf{p}_{final}$ are parallel to the predictive orientation of the target vehicle at the predictive time step k and the road tangent at the final point, respectively, where $\mathbf{p}_k$ is the predictive position of the target vehicle at the time step k defined as $\mathbf{p}_k = [p_{x,k}, p_{y,k}]^T$. The boundary condition of the curve has the sense that the target vehicle adjusts its heading angle parallel to the lane direction at the end of the lane change. Finally, the virtual yaw rate $\gamma_{virtual}$ at the predictive time step k is derived based on the curvature and orientation of the trajectory as follows:

$$\gamma_{k,virtual} = v_k \frac{y''_k}{\left(1 + y'^2_k\right)^{3/2}}, \tag{21}$$

where v_k is the predictive velocity of the target vehicle at the time step k.

The virtual longitudinal acceleration is determined as the acceleration input for velocity tracking. The Optimal Velocity Model (OVM) is utilized to design the acceleration input, which tracks the desired velocity as follows [41]:

$$\begin{aligned} a &= \dot{v} \\ &= \kappa \cdot (v_{des} - v), \end{aligned} \tag{22}$$

where a and v are the acceleration and velocity: v_{des} is the desired velocity; and κ is the gain constant for the velocity difference. In this study, κ of 0.16s^{-1} is used. The desired velocity v_{des} is estimated in the sense that the target vehicle travels the arc length of s_{LC} during the time of t_{LC}. Assuming that the vehicle generates the longitudinal motion according to the OVM, the velocity v is derived as a function of time t by solving the linear differential equation in equation (22), and the desired velocity v_{des} is obtained by matching the coefficients of the terms in the solution v with the following conditions:

$$\begin{aligned} v(t = t_0) &= v_0, \\ \int_{t_0}^{t_0 + \hat{t}_{LC}} v \cdot dt &= \hat{s}_{LC}, \end{aligned} \tag{23}$$

where t_0 and v_0 are the current time and velocity, respectively. Therefore, the virtual longitudinal acceleration at each predictive time step is derived using the OVM utilizing the estimated desired velocity v_{des}.

The proper assumption of the measurement noise covariance $\mathbf{R}$ is important to improve the prediction accuracy of the EKF because $\mathbf{R}$ affects the determining procedure of the Kalman gain $\mathbf{K}$. $\mathbf{R}$ is designed through the intuitive that the noise properties of the virtual measurements are dependent on the reliability of the estimated behavioral parameters. The noises of the virtual measurements can be derived as functions of the nominal values and errors of the predictive vehicle states and the behavioral parameters. Finally, the standard deviation of the virtual measurement noises and the measurement noise covariance $\mathbf{R}$ at the predictive time step k are expressed as follows:

$$\begin{aligned} \sigma_{\gamma_{k,virtual}} &= g_{\gamma_{k,virtual}}(\bar{\mathbf{x}}_k, \mathbf{M}_k, \hat{\boldsymbol{\xi}}_{GP}), \\ \sigma_{a_{k,virtual}} &= g_{a_{k,virtual}}(\bar{\mathbf{x}}_k, \mathbf{M}_k, \hat{\boldsymbol{\xi}}_{GP}), \\ \mathbf{R}_k &= \text{diag}(\sigma^2_{\gamma_{k,virtual}}, \sigma^2_{a_{k,virtual}}), \end{aligned} \tag{24}$$

where $\boldsymbol{\xi}_{GP}$ is defined as a set of the nominal values and standard deviations of the behavioral parameters derived by the GP models; and g denotes the function that maps the *a priori* state $\bar{\mathbf{x}}$, *a priori* error covariance $\mathbf{M}$, and $\boldsymbol{\xi}_{GP}$ to the standard deviations of the behavioral parameters.

V. MPC-BASED MOTION PLANNING AND CONTROL

The proposed predictor is applied to the motion planning and control of the autonomous vehicle that deals with the situation that a side front vehicle in the side lane changes its lane to the driving lane of the subject vehicle. In this paper, the longitudinal vehicle controller is designed to maintain clearance and adjust the speed with the preceding vehicle. The control of acceleration and deceleration is considered in this work. To carry out an appropriate response to the cut-in vehicle, predictive control is required to consider the potential risk and behavior.

An MPC problem is designed to deal with the requirements mentioned above. The concept in [42] is utilized in the predictive control against the cut-in vehicle. To reflect the system dynamics to the MPC, a point mass model is used as a plant model as defined below:

$$\begin{aligned} &\mathbf{q}_{k+1} = \mathbf{A}\mathbf{q}_k + \mathbf{B}u_k \\ &\text{s.t. } \mathbf{A} = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 - \Delta t/\tau \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \Delta t/\tau \end{bmatrix}, \end{aligned} \tag{25}$$

where k is the predictive time step; Δt is the sampling time of 0.1s; $\mathbf{q}$ is the state vector consisting of the travel distance p, longitudinal velocity v, and longitudinal acceleration a; u is the control input; and τ is the time constant of the first-order response system of the actuator. In this study, longitudinal acceleration command is used as the control input u; and τ of 1s is used. The MPC derives the time series of the solution u within the predictive horizon N_p by minimizing the following cost function:

$$\begin{aligned} J = \sum_{k=1}^{N_p} \left\{ (\mathbf{q}_k - \mathbf{q}_{k,ref})^T \cdot \mathbf{W}_{q,k} \cdot (\mathbf{q}_k - \mathbf{q}_{k,ref}) \right\} \\ + \sum_{k=0}^{N_p - 1} \left\{ W_{u,k} \cdot u_k^2 \right\}, \end{aligned} \tag{26}$$

where $\mathbf{q}_{ref}$ denotes the reference states to be tracked; and $\mathbf{W}_q$ and W_u are the weighting factors for reference tracking error and input magnitude, respectively. The constraints related to the MPC are determined as follows:

$$\mathbf{q}_{k+1} = \mathbf{A}\mathbf{q}_k + \mathbf{B}u_k,$$

$$\mathbf{G}^T \mathbf{q}_k \leq \mathbf{q}_{k,bound}, \quad \mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}^T,$$
$$u_{\min} \leq u_k \leq u_{\max}, \tag{27}$$

where $\mathbf{q}_{bound}$ is the position boundary of the vehicle derived by the configuration of the surrounding vehicles; and $u_{\min}$ and $u_{\max}$ denote the minimum and maximum control input, respectively. The reference states are determined in the sense that the vehicle adjusts the velocity to the predictive velocity of the preceding vehicle and maintains a desired clearance c_{des} designed as shown below:

$$c_{des} = \tau_{gap} \cdot v_{ref} + c_{\min}, \tag{28}$$

where τ_{gap} is the desired time gap of 1.3s; $c_{\min}$ is the minimum clearance of 3m; and v_{ref} is the reference velocity of the subject vehicle. τ_{gap} and $c_{\min}$ are determined considering the typical characteristics of human drivers in car following scenarios by referring [45]. A longitudinal safe distance between the center of gravity of the subject vehicle and the preceding vehicle is defined as follows:

$$sd = c_{des} + L_{\text{vehicle}} + \sigma_{p,\text{target}}, \tag{29}$$

where L_{vehicle} is the length of the vehicle; and $\sigma_{p,\text{target}}$ is the standard deviation of the predictive longitudinal position of the preceding vehicle. The reference states are designed using the predictive information of the preceding vehicle as follows:

$$\begin{aligned} \mathbf{q}_{k,ref} &= \begin{bmatrix} p_{k,ref} & v_{k,ref} & 0 \end{bmatrix}^T \\ \text{s.t. } p_{k,ref} &= \begin{cases} p_{k,\text{target}} - sd_k & \text{if } p_{k,\text{target}} < sd_k \\ p_{k,\text{host}} & \text{otherwise,} \end{cases} \\ v_{k,ref} &= \\ & \begin{cases} v_{k,\text{target}} & \text{if } p_{k,\text{target}} < sd_k \\ W_k \cdot v_{set} + (1 - W_k) \cdot v_{k,\text{target}} & \text{otherwise,} \end{cases} \\ W_k &= \frac{p_{k,\text{target}} - sd_k}{p_{k,\text{target}}}, \end{aligned} \tag{30}$$

where p_{target} and v_{target} are the predictive longitudinal position and velocity of the preceding vehicle, respectively; v_{set} is the set speed of the subject vehicle; and $p_{k+1,\text{host}} = p_{k,\text{host}} + v_{k,ref} \cdot \Delta t$ for the time step k with the initial condition $p_{0,\text{host}} = 0$. If the cut-in vehicle is predicted to invade the driving lane of the subject vehicle, the information about the cut-in vehicle is reflected in the decision of the preceding vehicle's predictive states. Fig. 6 depicts an exemplary cut-in case in which a side lane vehicle is predicted to cross the lane at the predictive time step $N_{p,\text{cross}}$ within the horizon of N_p. In this case, the cut-in vehicle is regarded as the preceding vehicle in deriving the reference states at $k = N_{p,\text{cross}}, N_{p,\text{cross}} + 1, \ldots, N_p$. Therefore, the controller allows the vehicle to react to the cut-in vehicle proactively before the lane-changing vehicle actually invades the driving lane of the subject vehicle.

VI. PREDICTION PERFORMANCE ANALYSIS

The prediction performance of the proposed trajectory-level predictor has been evaluated through the data-based evaluation. The dataset for the evaluation is obtained from 11,678 samples of the cut-in driving data from the NGSIM database. Section VI. A reports the prediction accuracy of the proposed algorithm with prediction error analysis. Section VI. B describes the prediction results for two cases from the evaluation dataset.

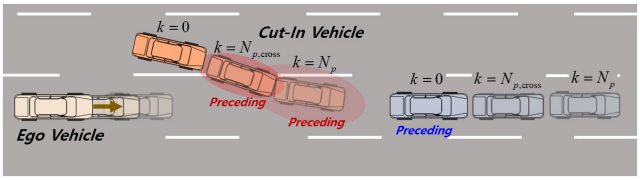


FIGURE 6. Exemplary driving scene in which a vehicle in the side lane cuts in. Predictive states of the cut-in vehicle are reflected in reference states of MPC at $k = N_{p,\text{cross}}, N_{p,\text{cross}} + 1, \ldots, N_p$.

The prediction results have been compared to the results from the existing prediction methods that have been widely used in the prediction of surrounding vehicles' motion: the Constant Turn Rate and Velocity (CTRV) model and the path-following model with Constant Velocity (CV) [8], [12]–[15], [24], [42]. Two base algorithms utilize states of the prediction vehicles without the awareness of inter-vehicle interaction. In particular, the path-following model with CV utilizes lane information and derives the prediction results based on the virtual yaw rate by a rule-based approach referring [8].

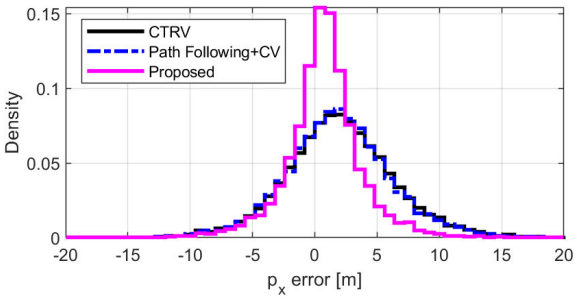
A. PREDICTION ERROR ANALYSIS

The accuracy of the prediction has been evaluated by analyzing the characteristics of the prediction errors with respect to ground truth. The prediction errors of position and velocity have been obtained by processing the samples from the NGSIM database. The prediction accuracy of time-to-cross, i.e., the time required to cross lanes, has also been discussed based on the dataset, which is composed of the lane-change trajectories earlier than 0.5 seconds before crossing lanes. To evaluate the robustness of the prediction algorithms against the noise generated by the perception module, white Gaussian noises have been added to the states of the vehicles in the current time step. By referring to the performance of LiDAR-based perception algorithms discussed in [43], [44], the additive Gaussian noises for the vehicle states have been designed to have zero means and standard deviations of 0.3m, 0.3m, 0.05rad, 0.3m/s, 0.06rad/s, and 0.3m/s^2 for x position, y position, heading angle, absolute velocity, yaw rate, and longitudinal acceleration, respectively.

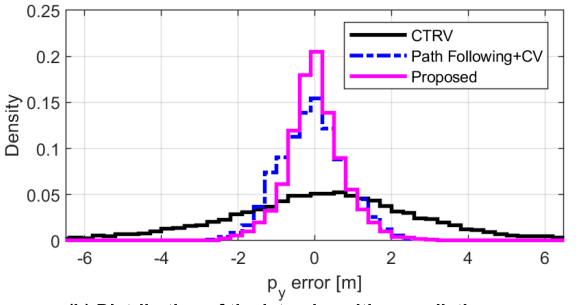
The prediction errors including comparison with base algorithms are described in Table 2 and Fig. 7. Table 2 shows that the proposed prediction method provides a lower level of the prediction error compared to the base algorithms in terms of the following error metrics: Mean Absolute Error (MAE), Standard Deviation (STD), and Root Mean Square Error (RMSE). The error histograms depicted in Fig. 7 imply that the errors from the proposed predictor are more distributed near zero, which signifies the reduced proportion of the

TABLE 2. Comparison of statistics of prediction errors between the proposed algorithm and the base algorithms.

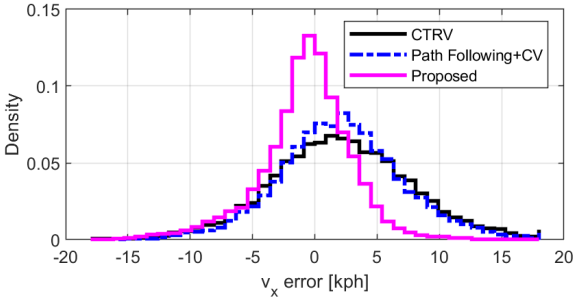
Error	Prediction Horizon	CTRV			Path Following Model + CV			Proposed		
		MAE	STD	RMSE	MAE	STD	RMSE	MAE	STD	RMSE
Longitudinal Position (m)	1 s	1.00	1.54	1.61	0.99	1.54	1.61	0.86	1.44	1.48
	2 s	2.24	3.05	3.24	2.22	3.03	3.21	1.58	2.49	2.56
	3 s	3.63	4.53	4.83	3.54	4.45	4.71	2.18	3.18	3.20
Lateral Position (m)	1 s	0.58	0.80	0.80	0.41	0.58	0.58	0.39	0.56	0.56
	2 s	1.27	1.75	1.75	0.59	0.83	0.84	0.50	0.78	0.79
	3 s	2.16	2.99	2.99	0.73	1.11	1.13	0.61	1.07	1.07
Longitudinal Velocity (kph)	1 s	1.97	2.47	2.58	1.90	2.39	2.50	1.56	2.07	2.07
	2 s	3.44	4.27	4.48	3.17	3.95	4.17	2.34	3.11	3.17
	3 s	4.66	5.77	6.01	4.12	5.11	5.39	2.84	3.74	3.92
Time-to-cross (s)	-	0.76	0.99	1.07	0.73	0.93	1.07	0.67	0.89	0.98



(a) Distribution of the longitudinal position prediction error



(b) Distribution of the lateral position prediction error



(c) Distribution of the longitudinal velocity prediction error

FIGURE 7. Prediction error histograms at the prediction horizon of 3s with comparison between the proposed algorithm and the base algorithms.

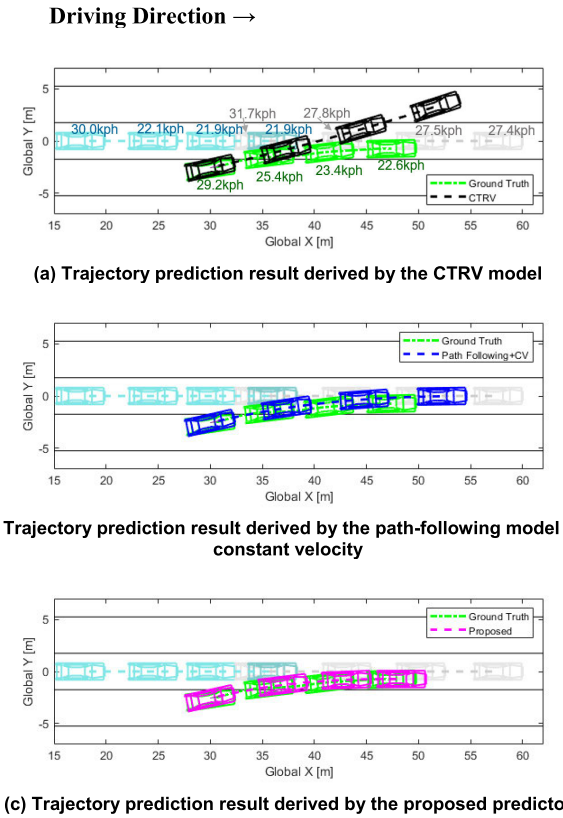
samples with large errors. In particular, for a prediction time of 3 seconds, the samples in which the lateral position error exceeds the threshold of 1.5m account for 53.1% in the CTRV model, 9.0% in the path-following model with CV, and 6.3% in the proposed method. The prediction results with different prediction horizons show that the improvement of the prediction accuracy by the proposed method is more noticeable in long-term prediction than in short-term prediction. The advanced long-term prediction efficiently captures the cut-in aggressiveness, which improves the estimation results of the time-to-cross.

B. CUT-IN CASE STUDY

Fig. 8 and Fig. 9 present two cases of the cut-in motion prediction performance in a multi-vehicle situation with the proposed predictor and the base prediction algorithms. The true trajectories of the cut-in vehicle, the side front vehicle, and the side rear vehicle are shown with the true speed values in green, gray, and light blue, respectively, for 3 seconds at 1-second intervals. The prediction results of the CTRV model, the path-following model with CV, and the proposed predictor are represented in black, blue, and magenta, respectively, for 0, 1, 2, and 3 seconds of prediction time.

In Fig. 8 (a), the CTRV model provides the erroneous prediction that the cut-in vehicle changes its lane by two lanes. Fig. 8 (b) shows the unreliable result derived by the path-following model with CV that the vehicle takes on an extremely high risk of collision with the side front vehicle. On the other hand, as shown in Fig. 8 (c), the proposed algorithm manages to predict that the cut-in vehicle decelerates and reaches a reliable position in the lane-change target space. Note that the lateral motion is more accurately predicted in the proposed predictor than the path-following model with CV. In another case, as depicted in Fig. 9 (a) and (b), the two base algorithms provide the unrealistic prediction result that the cut-in vehicle collides with the side rear vehicle at the predictive time step of 3s. However, in Fig. 9 (c), the proposed algorithm predicts that the vehicle moves to the adjacent lane with acceleration to avoid the collision.

The comparison results of the base and proposed algorithms imply three interpretations. First, the road-aware

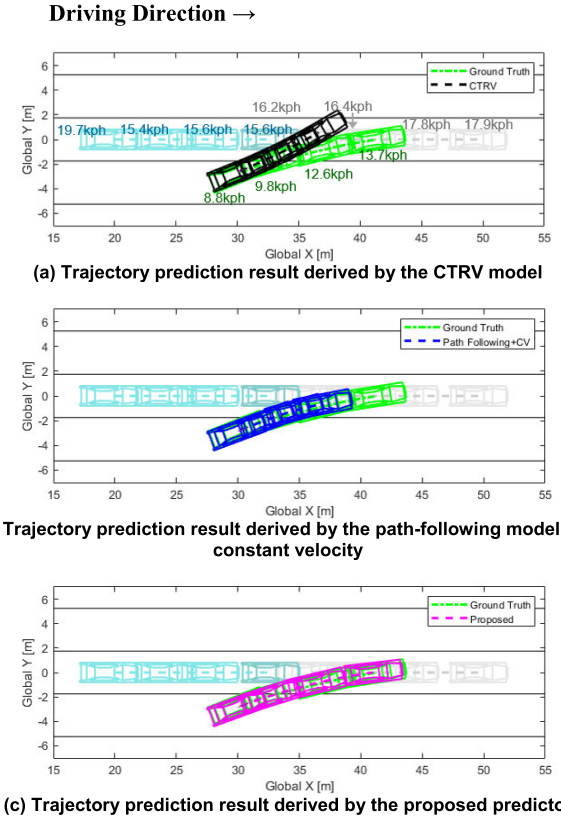


(a) Trajectory prediction result derived by the CTRV model

(b) Trajectory prediction result derived by the path-following model with constant velocity

(c) Trajectory prediction result derived by the proposed predictor

FIGURE 8. Examples of trajectory prediction results derived by the proposed algorithm and the base algorithms for normal traffic environment.



(a) Trajectory prediction result derived by the CTRV model

(b) Trajectory prediction result derived by the path-following model with constant velocity

(c) Trajectory prediction result derived by the proposed predictor

FIGURE 9. Examples of trajectory prediction results derived by the proposed algorithm and the base algorithms for dense traffic environment.

prediction improves the accuracy of lateral motion prediction. Compared to the CTRV model, which assumes a fixed yaw rate, the path-following rule allows the yaw rate to change according to the road geometry and reduces the effect of the perception error of the yaw rate into the prediction performance. Second, the interaction-aware approach enhances the prediction performance of the longitudinal motion prediction. Unlike the base algorithms, the proposed predictor properly captures the traffic context of the interacting vehicles and prevents the unrealistic prediction of the cut-in vehicle's behavior such as extremely high collision risk with the side lane vehicles. Finally, the data-driven approach improves the trajectory prediction accuracy. The improved ability of driving style modeling based on the learning-based behavioral parameters reduces the prediction errors in a long time interval. Therefore, these improvements aid the subject vehicle in accurately estimating the potential risk of the cut-in vehicle in a long time horizon.

VII. CONTROL PERFORMANCE ANALYSIS

The performance of the control strategy has been evaluated via computer simulations and real vehicle tests. The evaluation environment is the multi-vehicle traffic scenario with the side front vehicle's cut-in maneuver. The MPC-based controller has been used to achieve clearance keeping with the preceding vehicle and predictive reaction against the cut-in vehicle. To implement the predictor-based control algorithm within an integrated system, the lane-change intention has been determined based on a method in [25] that classifies the maneuver of surrounding vehicles as ''lane keeping'', ''lane changing'', ''arrival'', and ''adjustment'' based on a multi-class Support Vector Machine (SVM). The control results have been analyzed in terms of ride quality and safety performance. In Section VII. A, the simulation results based on the data-based simulation are described and discussed. To compare the impact of prediction performance to control, the base and proposed prediction algorithms have been applied to the MPC-based controller. In Section VII. B, the vehicle test results are described to evaluate the control performance in the real world.

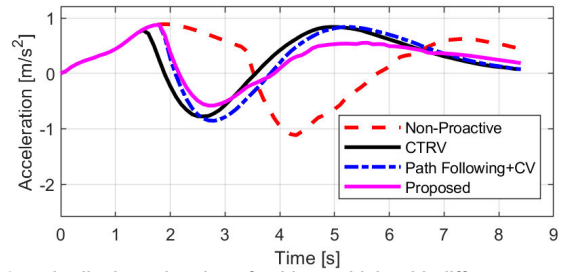
A. SIMULATION RESULTS

The simulation of the control system combined with prediction has been conducted in the lane-keeping scenario. The simulation environment is a two-lane road on which the subject vehicle and two surrounding vehicles are placed. In the simulation scenario, a side lane vehicle changes to the driving lane of the subject vehicle and moves into the space between the subject vehicle and a vehicle directly in front of the subject vehicle. For each scenario, the initial location and motion conditions of all vehicles are defined based on the traffic data from the NGSIM database. The

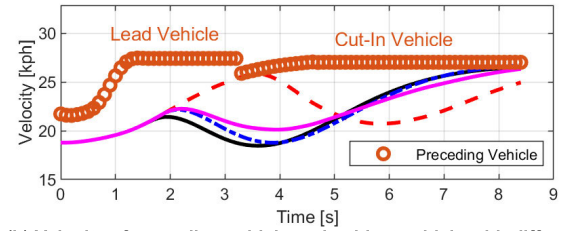
whole motions of the surrounding vehicles are simulated using the NGSIM database. The movement of the subject vehicle is simulated based on the MPC-based controller combined with prediction. To evaluate the effectiveness of the proactive response against the cut-in vehicle, the control result of the proposed proactive control method is compared to that of a non-proactive controller that responds to the cut-in vehicle only if the vehicle invades the driving lane. Also, to validate the improvement of the proposed predictor in proactive control, the controllers based on the proposed prediction method and the base prediction methods discussed in Section VI are compared based on the performance results. The controllers with different approaches are simulated in identical conditions.

The simulation results have been evaluated based on the controller's ability to provide ride quality and maintain safety with respect to the preceding vehicle. Fig. 10 (a) to (d) show the simulation results of the controller with the four different approaches, including longitudinal acceleration command, velocity, clearance, and inverse Time To Collision (TTC). In Fig. 10 (a), the legend of 'Non-Proactive' denotes the result of the non-proactive controller, while the legends of the rest denote the proactive control results with the three different cut-in motion predictors. In Fig. 10 (b), brown markers denote the velocity history of the preceding vehicle. In Fig. 10 (c), the desired clearance defined in equation (28) is represented as green markers. Fig. 10 (d) depicts the inverse TTC between the subject vehicle and the preceding vehicle. Fig. 10 (e) represents the lateral position history of the cut-in vehicle with respect to the subject vehicle. In this scenario, a side lane vehicle actually crosses the lane at t = 3.3s. Fig. 10 (f) depicts the simulation scenario at t = 2.7s, in which the subject, cut-in, and lead vehicles are represented in blue, red, and black, respectively. The prediction result of the cut-in vehicle with the proposed predictor is shown for 2 seconds at the interval of 0.4 seconds.

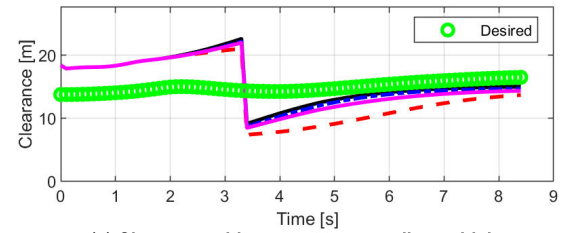
As shown in Fig. 10 (a), the controllers except for the non-proactive control proactively slow down the vehicle in reaction to the cut-in 1.6 seconds before the side lane vehicle crosses the lane. The early deceleration of the proactive controllers improves the desired clearance tracking performance compared to the non-proactive approach, as shown in Fig. 10 (c). As indicated in Fig. 10 (b), the velocity change caused by the reaction against the cut-in vehicle is relatively small in the proactive control methods, which leads to the fast adjustment of the subject vehicle velocity to the preceding vehicle velocity. Accordingly, the inverse TTC reaches the region near zero more rapidly in the proactive control compared to the non-proactive control, as shown in Fig. 10 (d). This implies that the proactive deceleration in response to the cut-in improves ride quality and adaptation of the subject vehicle's motion to the preceding vehicle's motion. Among the proactive controllers with the three different prediction approaches, the controller with the proposed predictor uses the smallest amount of deceleration in response to the cut-in, as depicted in Fig. 10 (a). This leads to the relatively small



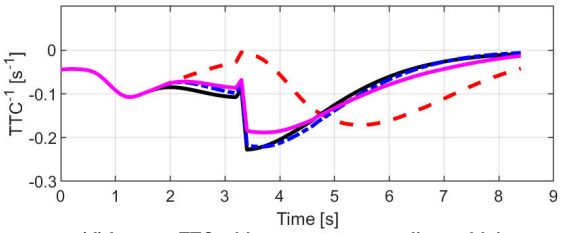
(a) Longitudinal acceleration of subject vehicle with different controllers



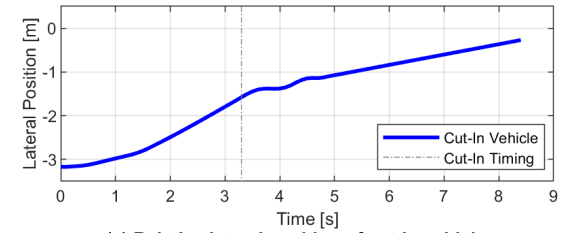
(b) Velocity of preceding vehicle and subject vehicle with different controllers



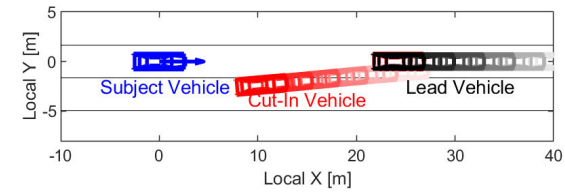
(c) Clearance with respect to preceding vehicle



(d) Inverse TTC with respect to preceding vehicle



(e) Relative lateral position of cut-in vehicle



(f) Simulation scenario at t = 2.7s

FIGURE 10. Simulation results of predictive control in cut-in scenario with comparison between the proposed algorithm and the base algorithms.

velocity variation from t = 2 to 4s in the proposed method. Accordingly, the usage of the acceleration for velocity recov-

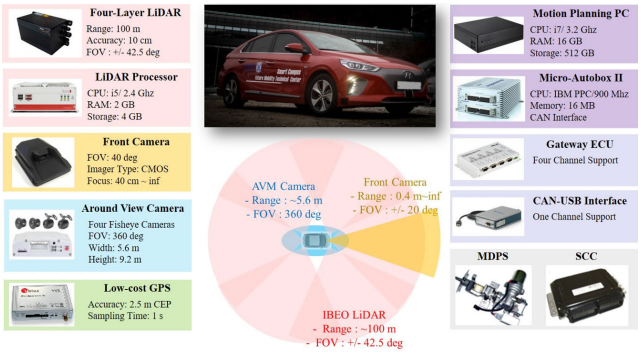


FIGURE 11. Configuration of the test vehicle for implementation of the proposed algorithm.

TABLE 3. Statistics of control performances in the simulation test in terms of control effort and inverse TTC.

Variable	Metric	Non-Proactive Control	Proactive Control		
			CTRV	Path Following Model + CV	Proposed Predictor
Control Effort (m/s^2)	Mean	0.62	0.59	0.59	**0.57**
	Mean Peak	1.11	1.08	1.08	**1.03**
Inverse TTC (s^{-1})	Mean	**-0.005**	-0.059	-0.054	**-0.052**
	Mean Peak	0.278	0.024	0.033	0.036

ery from t = 4 to 6.5s is the lowest in the proposed method. Meanwhile, as shown in Fig. 10 (c), the desired clearance tracking performance of the proposed method is comparable to that of the controllers with other predictors. The drop of the inverse TTC at t = 3.3s is smaller in the controller with the proposed predictor than with the other predictors, which implies the less conservative response of the proposed method. This indicates that the proactive controller with the proposed predictor reacts to the cut-in vehicle with less control effort while maintaining safe clearance similarly to that of other proactive methods.

The simulation results of 509 lane-keeping scenarios with 21,964 total samples have been acquired and analyzed statistically. Table 3 describes the statistical results including the control effort and the inverse TTC in terms of the mean and the mean peak value. The ''mean peak'' implies the average value of the peak values in each simulation scenario. In this section, the control effort is defined as the absolute value of the longitudinal acceleration command. The results show that the proactive control method uses less control effort compared to the non-proactive method. In particular, the controller based on the proposed predictor reduces the average control effort usage by 7.8% compared to the non-proactive controller. Also, the proposed predictor leads to the lowest usage of the control effort compared to the other predictors in terms of both the mean and the mean peak value. The mean of the inverse TTC is closest to zero in the non-proactive control than other methods. However, the positive mean peak value of the inverse TTC larger than $0.2s^{-1}$ reveals that the non-proactive controller may cause a risky situation when reacting to the cut-in, which leads to anxiety among the passengers. In fact, the work in [45] evaluates a situation with the inverse TTC larger than $0.2s^{-1}$ as uncomfortable. On the other hand, the proactive controllers manage the risk properly and make the inverse TTC bounded within safe region in terms of both the mean and mean peak. The proposed predictor-based controller shows the mean value of the inverse TTC that is nearest to zero among the three proactive controllers. This means that the proposed predictor leads to better performance of the adaptive cruise control.

TABLE 4. The number of occurrences of the critical cases in the simulation test in terms of control effort and inverse TTC.

Variable	Case	Non-Proactive Control	Proactive Control		
			CTRV	Path Following Model + CV	Proposed Predictor
Control Effort (m/s^2)	> 1.5	768	711	732	**640**
	> 2.0	265	264	258	**223**
Inverse TTC (s^{-1})	> 0.2	113	13	14	14
	> 0.4	32	0	0	0

To evaluate the frequency of undesirable situations, the occurrence of critical cases has been counted for each controller. In this paragraph, the critical situation is defined as excessive control usage and high collision risk. The number of occurrences of critical cases out of the 21,964 samples is represented in Table 4 in terms of the control effort and the inverse TTC with different boundaries of the criticality. The controller combined with the proposed predictor reduces the cases with the control effort larger than $1.5m/s^2$ by 16.7%, and the cases larger than $2.0m/s^2$ by 15.9% compared to the non-proactive controller. Compared to other proactive controllers, the proposed method reduces the rate of the event with a large control effort by over 9.9%. In the case of the inverse TTC, the number of events in which the inverse TTC exceeds $0.2s^{-1}$ significantly falls by over 88.5% in the proactive control compared to the non-proactive control. Moreover, the proactive controllers achieve zero occurrences of the inverse TTC exceeding $0.4s^{-1}$, while the non-proactive method does not. Meanwhile, the frequency of the high collision risk situation is comparable within the proactive controllers with the different predictors. Therefore, the proactive response to the cut-in reduces the control effort and enhances collision safety. Furthermore, the improvement of the accuracy of the trajectory-level prediction affects the control efficiency and the adaptation ability to the cut-in vehicle's motion.

B. VEHICLE TEST RESULTS BASED ON AUTONOMOUS DRIVING

The application of the proposed predictor to the MPC-based controller has been validated via real vehicle tests based on

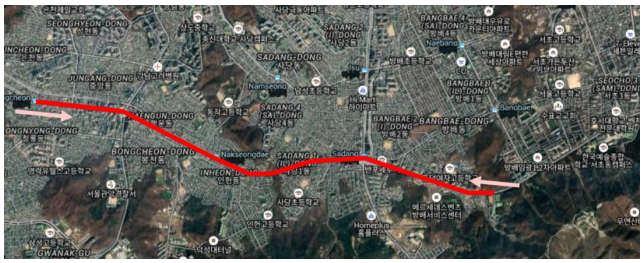


FIGURE 12. Vehicle test route.

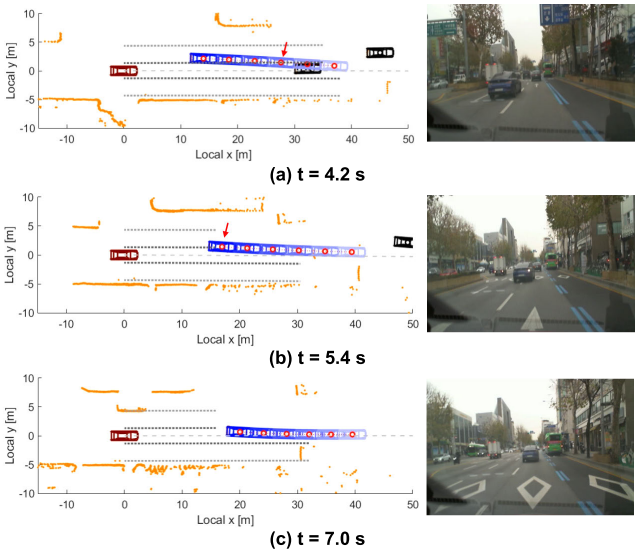


FIGURE 13. Snapshots of the vehicle test results in cut-in scenario.

autonomous driving. The proactive control has been realized by implementing the proposed algorithm on the test vehicle. The test vehicle is equipped with sensors, processors, and control modules as shown in Fig. 11. Surrounding obstacle information is provided by the LiDAR system, which consists of six 2D-LiDAR and LiDAR processors. The relative position, orientation, velocity, and type of obstacles are obtained by the LiDAR system. Lane information is provided as a quadratic polynomial by a front camera system. The environmental information acquired by the sensors is processed by an industrial PC. The controller combined with the proposed predictor has been implemented in the industrial PC. Micro-Autobox II is used to apply control command to the vehicle actuator system through the gateway Electronic Control Unit (ECU).

Vehicle tests have been conducted on urban driving roads in Gwanak-gu, Seoul, South Korea, as shown in Fig. 12. In South Korea, a permit system for autonomous driving in the public road environment is in operation. The permit is granted through autonomous driving test, fail-safe operation test, and driver override test. The test vehicle in Fig. 11 has been approved for the permission, and can be tested for autonomous driving in real world. The automated driving data have been collected for 82 minutes in multi-vehicle traffic environment. In total, 4,454 samples of data have been

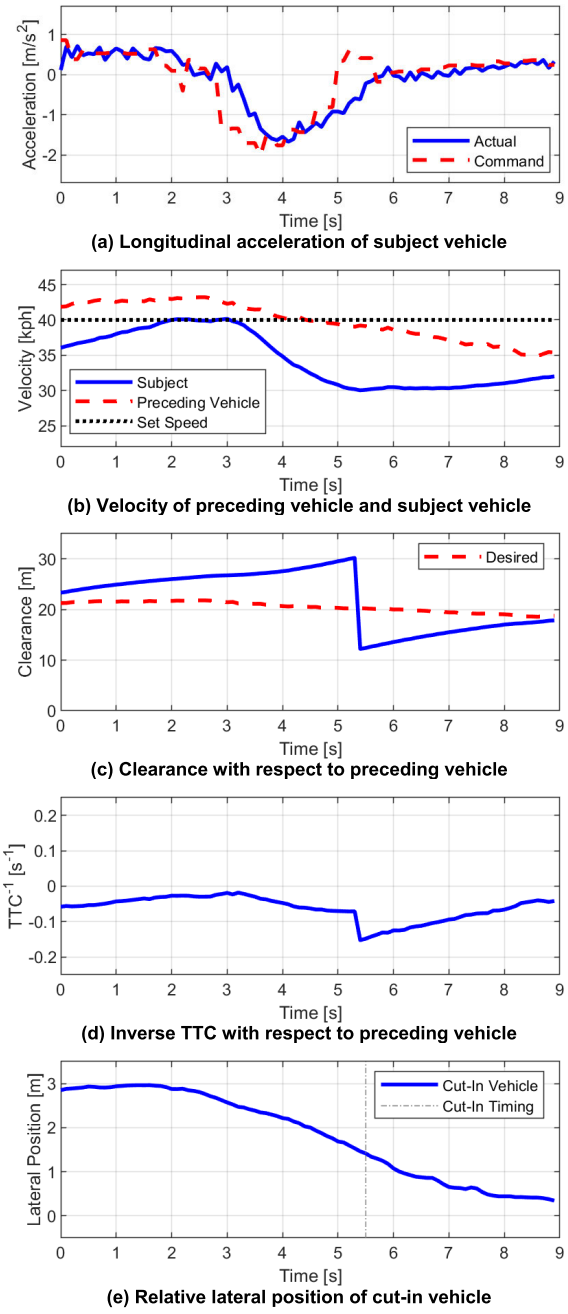


FIGURE 14. Vehicle test results in cut-in scenario.

obtained from 69 cut-in scenarios and processed to evaluate the control performance. Fig. 13 and Fig. 14 depict the vehicle test results in a cut-in situation in which a sedan with blue color changes lanes to the subject lane. Fig. 13 (a) to (c) show the snapshots of the test results with a dashboard camera image and trajectory prediction results of the cut-in vehicle. The trajectory prediction results are shown for the predictive horizon of 2 seconds with the interval of 0.4 seconds, which are represented as red markers and blue lines. The subject and lead vehicles are visualized with brown and black lines, respectively. The point clouds measured by the LiDAR are represented as orange points. The dotted gray

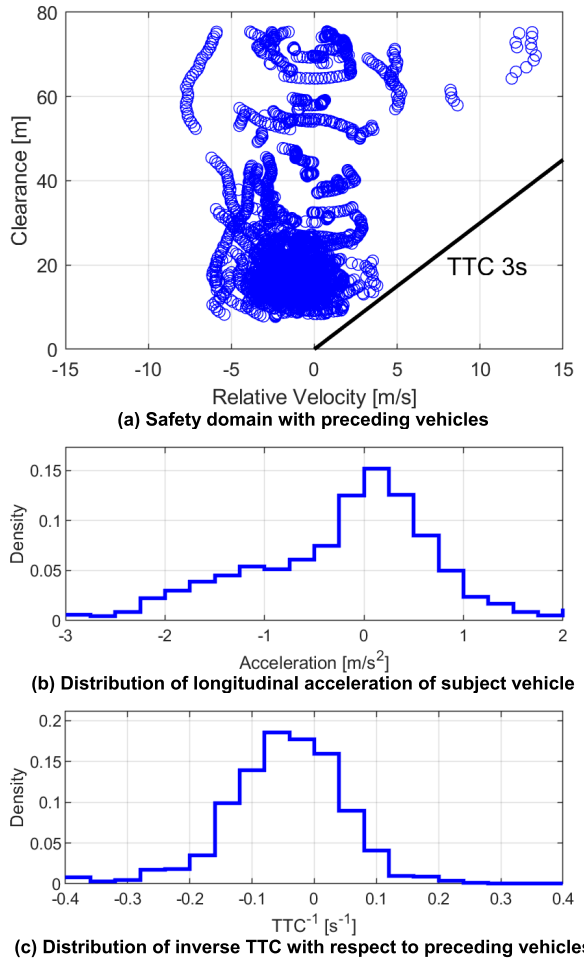


FIGURE 15. Summary of vehicle test results in cut-in scenario.

lines indicate the lane information perceived by the front camera system. The results of the test case in Fig. 13 are represented in Fig. 14 with the longitudinal acceleration, velocity, clearance, inverse TTC, and lateral position of the cut-in vehicle with respect to the subject vehicle.

As shown in Fig. 13 (a), the geometric center of the blue sedan is predicted to cross the lane at the prediction time of 1.2s from t = 4.2s. The center of the cut-in vehicle actually crosses lanes at t = 5.4s, as shown in Fig. 13 (b). Fig. 14 (a) and (b) show that the test vehicle starts to react to the cut-in behavior at t = 2.9s and decelerates before the blue sedan crosses lanes. The clearance is dropped when the blue sedan becomes a new preceding vehicle at t = 5.4s, as depicted in Fig. 14 (c) and (e). The proactive deceleration makes the clearance smoothly increase to reach the desired clearance after the clearance drop occurs. As depicted in Fig. 14 (b), the velocity of the subject vehicle smoothly adapts to the preceding vehicle's velocity after t = 5.4s. Fig. 14 (d) shows that the adaptation of the velocity induces the inverse TTC to reach the region near zero. During the response to the cut-in, the test vehicle uses the mild level of the longitudinal acceleration within the range of -1.7m/s^2 to 0.8m/s^2, as shown in Fig. 14 (a). Fig. 14 (c) shows that adequate clearance is maintained with a minimum clearance of 12.2m. Also, Fig. 14 (d) shows that the inverse TTC is maintained within the safe region.

As depicted in Fig. 15, the vehicle test results of the 69 cut-in scenarios are summarized as the following visualizations: a safety domain of the relative velocity and the clearance, and histograms of the longitudinal acceleration and the inverse TTC. Fig. 15 (a) shows that the clearance and the TTC are maintained within the safe region. The clearance is maintained above 7.71m, and the minimum value of positive TTC is 3.83s. In Fig. 15 (b) and (c), the distributions of the longitudinal acceleration and the inverse TTC show a bell-shaped curve centered near zero, respectively. The histograms in Fig. 15 (b) and (c) imply that the proposed algorithm manages to use small control effort and to rapidly adapt motion to the cut-in vehicle's motion.

Therefore, the proactive control based on the proposed predictor can deal with the cut-in situation caused by the side lane vehicle in the real world. The safety performance with respect to the cut-in vehicle is properly maintained while a moderate level of longitudinal acceleration is used. The improvement of the trajectory prediction helps the vehicle to reduce the severe change in the predictive motion strategy. Accordingly, this leads to the low frequency of abrupt braking and the smooth adaptation to the cut-in vehicle's motion, which improves ride quality.

VIII. CONCLUSION

A probabilistic trajectory prediction algorithm of cut-in vehicles has been developed, and its application to the control of the autonomous vehicle has been evaluated via simulation studies and autonomous driving vehicle tests. The behavioral parameters of the cut-in motion are estimated through Gaussian Process Regression (GPR) trained with the Next Generation Simulation (NGSIM) database. The interaction with the surrounding vehicles is implicitly considered in the Gaussian Process (GP) model to accurately determine the cut-in behavior by using the surrounding traffic information. Subsequently, the time series information of the future states of the cut-in vehicle is recursively obtained using the Extended Kalman Filter (EKF) approach combined with the path-following model. In addition, a Model Predictive Control (MPC) has been designed to calculate the longitudinal acceleration leading to the proactive reaction to the cut-in preceding vehicle.

The proposed algorithm has been evaluated in terms of prediction accuracy and proactive control performance of the automated vehicle applied by the predictor in a simulation study. The evaluation results have been compared with two base prediction algorithms: the Constant Turn Rate and Velocity (CTRV) model and the path-following model with Constant Velocity (CV). Compared to the base algorithms, the proposed predictor reduces the unrealistic prediction and improves the prediction accuracy by properly capturing the motion characteristics with the awareness of

interaction. The control simulation results show that the proposed prediction-based proactive control properly reacts to the cut-in vehicle with reduced control effort and enhanced ride quality while guaranteeing safety performance. The proactive controller with the proposed predictor has been implemented in the automated vehicle to evaluate the applicability of the proposed algorithm in the real world. The vehicle test results based on autonomous driving indicate that the proactive control combined with the proposed predictor uses mild acceleration while maintaining an appropriate level of clearance and Time To Collision (TTC) with respect to the preceding vehicle.

In the future, motion prediction of surrounding vehicles must be advanced in three aspects. First, intention inference of lane-change behavior must be developed and advanced to construct the integrated framework of the motion prediction algorithm. Second, temporal dependencies of the sequences of vehicle states should be considered to more efficiently capture the dynamic feature of vehicles. This may be achieved by applying a recurrent neural network (RNN). Finally, the interaction between the surrounding vehicles must be more delicately considered. To achieve this, spatial relation and dynamic states of the neighboring vehicles will be comprehensively processed to explicitly extract social cues. The improvement of motion prediction in these aspects will enhance the accuracy of prediction and the ride quality of control in the automated vehicle.

REFERENCES

[1] K. Bimbraw, "Autonomous cars: Past, present and future—A review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology," in *Proc. 12th Int. Conf. Informat. Control, Autom. Robot. (ICINCO)*, vol. 1, 2015, pp. 191–198.

[2] L. Barr and W. Najm, "Crash problem characteristics for the intelligent vehicle initiative," in *Proc. Transp. Res. Board 80th Annu. Meeting*, 2001, pp. 1–30.

[3] S. E. Lee, C. B. Olsen, and W. W. Wierwille, "A comprehensive examination of naturalistic lane changes," United States Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. FHWA-JPO-04-092, 2004.

[4] C. Bax, P. Leroy, and M. P. Hagenzieker, "Road safety knowledge and policy: A historical institutional analysis of The Netherlands," *Transp. Res. F, Traffic Psychol. Behav.*, vol. 25, pp. 127–136, Jul. 2014.

[5] J. D. Chovan, L. Tijerina, G. Alexander, and D. L. Hendricks, "Examination of lane change crashes and potential IVHS countermeasures. Final report," Nat. Highway Traffic Saf. Admin., Washington, DC, USA, Tech. Rep. HS-808, 1994, vol. 71.

[6] B. N. Campbell, J. D. Smith, and W. 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, 2003.

[7] N. Suganuma and T. Uozumi, "Development of an autonomous vehicle-system overview of test ride vehicle in the Tokyo motor show 2011," in *Proc. SICE Annu. Conf. (SICE)*, 2012, pp. 215–218.

[8] J. Suh, H. Chae, and K. Yi, "Stochastic model-predictive control for lane change decision of automated driving vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 4771–4782, Jun. 2018.

[9] S. Lefevre, A. Carvalho, and F. Borrelli, "A learning-based framework for velocity control in autonomous driving," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 1, pp. 32–42, Jan. 2016.

[10] G. Cesari, G. Schildbach, A. Carvalho, and F. Borrelli, "Scenario model predictive control for lane change assistance and autonomous driving on highways," *IEEE Intell. Transp. Syst. Mag.*, vol. 9, no. 3, pp. 23–35, Jul. 2017.

[11] W. Xu, J. Pan, J. Wei, and J. M. Dolan, "Motion planning under uncertainty for on-road autonomous driving," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2507–2512.

[12] S. Ammoun and F. Nashashibi, "Real time trajectory prediction for collision risk estimation between vehicles," in *Proc. IEEE 5th Int. Conf. Intell. Comput. Commun. Process.*, Aug. 2009, pp. 417–422.

[13] 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.

[14] H.-S. Tan and J. Huang, "DGPS-based vehicle-to-vehicle cooperative collision warning: Engineering feasibility viewpoints," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 415–428, Dec. 2006.

[15] C. G. Prevost, A. Desbiens, and E. Gagnon, "Extended Kalman filter for state estimation and trajectory prediction of a moving object detected by an unmanned aerial vehicle," in *Proc. Amer. Control Conf.*, Jul. 2007, pp. 1805–1810.

[16] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *Proc. 11th Int. Conf. Inf. Fusion*, 2008, pp. 1–6.

[17] R. Toledo-Moreo and M. A. Zamora-Izquierdo, "IMM-based lane-change prediction in highways with low-cost GPS/INS," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 180–185, Mar. 2009.

[18] C. Laugier, I. E. Paromtchik, M. Perrollaz, M. Y. Yong, J. Yoder, C. Tay, K. Mekhnacha, and A. Negre, "Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety," *IEEE Intell. Transp. Syst. Mag.*, vol. 3, no. 4, pp. 4–19, Oct. 2011.

[19] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 797–802.

[20] D. Lee, Y. P. Kwon, S. Mcmains, and J. K. Hedrick, "Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[21] 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, Jul. 2018.

[22] 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.

[23] J.-H. Kim and D.-S. Kum, "Threat prediction algorithm based on local path candidates and surrounding vehicle trajectory predictions for automated driving vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2015, pp. 1220–1225.

[24] 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.

[25] 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.

[26] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction with Gaussian mixture models," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 141–146.

[27] 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.

[28] K. Kim, D. Lee, and I. Essa, "Gaussian process regression flow for analysis of motion trajectories," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1164–1171.

[29] M. Tiger and F. Heintz, "Gaussian process based motion pattern recognition with sequential local models," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1143–1149.

[30] S. Yoon, H. Jeon, and D. Kum, "Predictive cruise control using radial basis function network-based vehicle motion prediction and chance constrained model predictive control," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3832–3843, Oct. 2019.

[31] Y. Yoon, T. Kim, H. Lee, and J. Park, "Road-aware trajectory prediction for autonomous driving on highways," *Sensors*, vol. 20, no. 17, p. 4703, Aug. 2020.

[32] C. Dong, Y. Zhang, and J. M. Dolan, "Lane-change social behavior generator for autonomous driving car by non-parametric regression in reproducing kernel Hilbert space," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 4489–4494.
[33] S. Su, K. Muelling, J. Dolan, P. Palanisamy, and P. Mudalige, "Learning vehicle surrounding-aware lane-changing behavior from observed trajectories," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1412–1417.
[34] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1179–1184.
[35] 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.
[36] Q. Zhang, R. Langari, H. E. Tseng, D. Filev, S. Szwabowski, and S. Coskun, "A game theoretic model predictive controller with aggressiveness estimation for mandatory lane change," *IEEE Trans. Intell. Vehicles*, vol. 5, no. 1, pp. 75–89, Mar. 2020.
[37] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, vol. 2, no. 3. Cambridge, MA, USA: MIT Press, 2006.
[38] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Math. Program.*, vol. 45, nos. 1–3, pp. 503–528, Aug. 1989.
[39] FHWA. (2006). *Next Generation Simulation (NGSIM) Program*. Accessed: Apr. 27, 2020. [Online]. Available: http://ngsim-community.org/
[40] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
[41] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, "Dynamical model of traffic congestion and numerical simulation," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 51, no. 2, pp. 1035–1042, Feb. 1995.
[42] H. Chae, Y. Jeong, H. Lee, J. Park, and K. Yi, "Design and implementation of human driving data–based active lane change control for autonomous vehicles," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, vol. 235, no. 1, pp. 55–77, Jan. 2021.
[43] H. Lee, H. Chae, and K. Yi, "A geometric model based 2D LiDAR/radar sensor fusion for tracking surrounding vehicles," *IFAC-PapersOnLine*, vol. 52, no. 8, pp. 130–135, 2019.
[44] H. Lee, J. Yoon, Y. Jeong, and K. Yi, "Moving object detection and tracking based on interaction of static obstacle map and geometric model-free approach for urban autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 26, 2020, doi: 10.1109/TITS.2020.2981938.
[45] S. Moon and K. Yi, "Human driving data-based design of a vehicle adaptive cruise control algorithm," *Vehicle Syst. Dyn.*, vol. 46, no. 8, pp. 661–690, Aug. 2008.

YOUNGMIN YOON received the B.S. degree in mechanical engineering from Seoul National University, South Korea, in 2018, where he is currently pursuing the Ph.D. degree in mechanical engineering. His research interests include motion prediction of surrounding vehicles, predictive motion planning, and automated driving vehicle control.

CHANGHEE KIM received the B.S. degree in mechanical engineering from Seoul National University, South Korea, in 2019, where he is currently pursuing the Ph.D. degree in mechanical engineering. His research interests include motion planning and control, lane change decision making, and urban autonomous driving.

JONGMIN LEE received the B.S. degree from the Department of Automotive Engineering, Tsinghua University, China, in 2015, and the M.S. degree in mechanical engineering from Seoul National University, South Korea, in 2018, where he is currently pursuing the Ph.D. degree in mechanical engineering. His research interests include fail-safe system for autonomous vehicle, and autonomous vehicle environmental sensor degradation.

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 Engineering, Seoul National University. His research interests include control systems, driver assistant systems, active safety systems, and automated driving of ground vehicles. He also serves as a Member of the Editorial Board for *Mechatronics* and the Chair for the KSME IT Fusion Technology Division.