IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 22, 2021, accepted April 5, 2021, date of publication April 12, 2021, date of current version April 20, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3072413

Lane Change Intention Classification of Surrounding Vehicles Utilizing Open Set Recognition

DAE JUNG KIM[1], (Graduate Student Member, IEEE), JIN SUNG KIM[1], (Graduate Student Member, IEEE), JIN HO YANG[1], (Member, IEEE), SEOK CHEOL KEE[2], (Member, IEEE), AND CHUNG CHOO CHUNG[3], (Member, IEEE)

[1]Department of Electrical Engineering, Hanyang University, Seoul 04763, South Korea
[2]Smart Car Research Center, Chungbuk National University, Cheongju 28644, South Korea
[3]Division of Electrical and Biomedical Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Chung Choo Chung (cchung@hanyang.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Ministry of Science and ICT (MSIT) (Data-Driven Optimized Autonomous Driving Technology Using Open Set Classification Method) under Grant 2021R1A2C2009908.

ABSTRACT This paper proposes a classification algorithm utilizing an open set recognition concept to conservatively detect lane change intention of surrounding vehicles. Conservatively predicting the lane change intention of the surrounding vehicles is needed to improve adaptive cruise control (ACC) performance and avoid possible accidents. However, existing machine learning can make incorrect decisions due to information not included in the training data set or confused data even with probability. To cope with this problem, we present a classification algorithm using a multi-class support vector machine applying an open set recognition concept to detect the surrounding vehicles' lane change intentions. Feature vectors are constructed from lateral information obtained by a Kalman filter using only radar and in-vehicle sensors. The open set recognition concept is adapted using Meta-Recognition based on binary classifiers scores. Furthermore, we analyze lateral information where an object vehicle changes lanes. From experimental results, we observe that the proposed system conservatively deals with wrong decisions and detects and cancels detecting the closest in-path vehicle (CIPV) earlier with average times of 1.4 sec and 0.4 sec compared with a commercial radar system, respectively.

INDEX TERMS Advanced driver assistance system, lane change intention, surrounding vehicles, machine learning, open set recognition, classification algorithm, multi-class support vector machine.

I. INTRODUCTION

Adaptive cruise control (ACC) that implements longitudinal speed control has been commercially available and is widely used in autonomous vehicles (AVs) beyond advanced driver assistance systems (ADASs). An essential function of ACC is to detect the closest in-path vehicle (CIPV) in front of the ego vehicle [1]. A front radar is mainly used because of the accuracy of its longitudinal information obtained from the Doppler effect to detect the CIPV [2]. On the other hand, the conventional radar system has low-resolution limits for lateral information. Moreover, the intrinsic latency time of a radar system causes late detection and, in a situation where the ego vehicle's speed should be rapidly reduced if the object vehicle cut-in at a close distance to the ego vehicle, may cause a collision. A rapid change in longitudinal speed could make passengers of the ego vehicle feel uncomfortable. Furthermore, when passengers notice an object trying to cut-in, but the ego vehicle does not respond appropriately, they may feel anxiety until the ego vehicle starts to reduce its speed. Therefore, detection of the CIPV is in significant need of improvement for optimal ACC performance.

Detecting the CIPV is commonly based on motion prediction of surrounding vehicles to decide whether one of the surrounding vehicles is changing lanes to the ego vehicle's lane or not. Motion prediction of surrounding vehicles is a core technology for preventing collisions in ADASs and AVs. The primary method for predicting the driving motion of

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson.

the surrounding vehicles is to use a constant velocity, acceleration, or turn model [3], [4]. However, the model cannot reflect motion, such as lane changes of the object vehicles or mutual driving motion between vehicles, correctly. For this reason, artificial intelligent models, machine learning, and deep learning methods have recently drawn attention to predicting the driving intention of surrounding vehicles, such as graphical modeling and unsupervised learning techniques [5], a multi-class support vector machine (MSVM) [6], a convolution neural network (CNN) [7], a multilayer perceptron [8], a recurrent neural network (RNN) [9], a long short term memory (LSTM) [10], and a trajectory proposal network (TPNet) [11]. These methods utilize real-world driving data collected on public roads with standard production sensors.

Detecting the CIPV is one of the most important factors to conservatively predict the lane change intentions of surrounding vehicles to improve ACC performance and avoid possible accidents. It is well known that existing learning-based researches can be easily scaled, and its model can be adapted using data [12]. The classification model may be preferred over the regression model to predict the lane change intention since the ACC system requires a deterministic decision about the lane change intention, e.g., whether the CIPV exists or not. However, machine learning, such as deep learning, cannot make conservative detection, though it is based on the probabilistic decision with the SoftMax layer. For example, since deep learning requires full handcrafted data details, it can make incorrect decisions for information not included by the training data set or confused data even with probability [13]–[15]. Additionally, a massive data set is necessary to catch the corner cases, but it has not been proven how to cope with such corner cases [16]. Therefore, an algorithm detecting lane change intention of the surrounding vehicles overcoming these limitations should be developed to be deterministic but conservative for ACC performance.

This paper first presents a feature extraction method to enhance the classification performance of the prediction of lane change intention of surrounding vehicles. In our previous work [6], we presented an MSVM classification method that can detect the lane change intention of the object vehicle. First, we divide an object vehicle's driving motion into seven types and then detect the CIPV according to each motion. Classifying the lane change intention of an object vehicle is made using feature vectors that consist of defined relative lateral distances and velocities based on the circular motion estimation obtained by only a commercial front radar and in-vehicle signal's time-window. However, the previous work [6] could not correctly deal with a low-resolution problem for lateral information of the commercial radar. Thus, this paper presents a feature vector extraction method, including a Kalman filter, to overcome the radar's lateral low-resolution limits and obtain a newly defined relative lateral velocity not accessible from the radar system. This paper shows improved classification performance using the presented feature vector. However, classification algorithms, including the MSVM, have a limitation that can make mistaken decisions on decision boundaries for the classification problem. In that case, conservatively predicting the lane change intention of surrounding vehicles is needed to improve ACC performance and avoid possible real driving accidents. To this end, we propose an MSVM classification method applying the open set recognition concept to conservatively detect lane change intentions of surrounding vehicles. The open set recognition concept is adapted using Meta-Recognition and allows for confusing classification [14]. To develop the MSVM classification containing the open set, we utilize binary SVM classifiers scores if the input is far from known training data. The proposed algorithm is validated with a data set not included in the training data set. We observe that the proposed system copes conservatively with wrong decisions. A comparative study with a commercial radar is quantitatively made to show the effectiveness of the proposed method. From the experimental results, the proposed system could detect and cancel detecting the CIPV earlier with average times of 1.4 sec and 0.4 sec, respectively. Furthermore, we constructed a confusion matrix to evaluate the proposed method's accuracy and achieved an accuracy of 92.2%.

In summary, the main contributions of this paper are as follows:

1) We present a feature extraction method based on only radar and in-vehicle sensors, including a KF to extract the velocity, not accessible from the radar, and filter out noise in the raw data. The feature extraction method facilitates the classification by considering the ego vehicle's yaw rate motion.
2) We propose an MSVM classification algorithm utilizing an open set recognition concept to conservatively detect lane change intention of surrounding vehicles. Introducing the open set recognition for CIPV enables the ACC system to cope conservatively with wrong decisions.
3) We analyze defined feature vectors where an object vehicle changes lanes and experimental results of the proposed method in the case of confusing situations for lane change intention of the object vehicle.

II. FEATURE VECTOR EXTRACTION

To begin with, we explain the overall structure of the proposed system for lane change intention classification of the object vehicle described in Fig. 1. In this study, we focus on using only radar and in-vehicle sensors. A feature vector is extracted with time-windowing and a Kalman filter based on newly defined lateral information. It is used for training a classification model and predicting the object vehicle's lane change intention. In this paper, we utilize the MSVM for the classification model. However, almost all machine learning, including the MSVM, can predict the object vehicle's lane change intention, but it might be difficult to make a conservative decision. To resolve this problem, we propose a method applying the open set recognition concept in the MSVM. Conclusively, the proposed system classifies the open set data

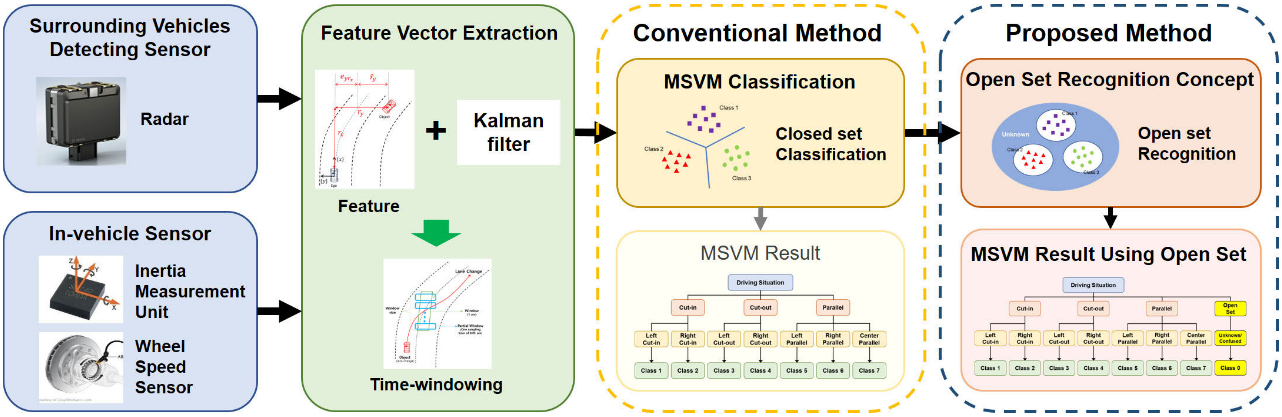
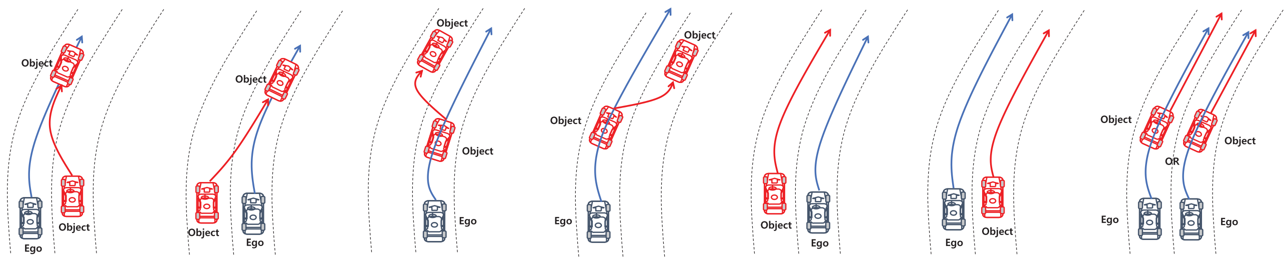


FIGURE 1. The overall structure of the proposed system: This paper focuses on using only radar and in-vehicle sensors. A feature vector is extracted with time-windowing and a Kalman filter based on newly defined lateral information. It is used for training a classification model and predicting the object vehicle's lane change intention. In this paper, we utilize the MSVM for the classification model. We also propose a method applying the open set recognition concept in the MSVM. Conclusively, the proposed method's prediction results classify the open set data such as unknown or confusing data.



(a) Class 1 (LCI) (b) Class 2 (RCI) (c) Class 3 (LCO) (d) Class 4 (RCO) (e) Class 5 (LP) (f) Class 6 (RP) (g) Class 7 (CP)

FIGURE 2. Description of each class over driving situation of: (a) left cut-in (LCI), (b) right cut-in (RCI), (c) left cut-out (LCO), (d) right cut-out (RCO), (e) left parallel (LP), (f) right parallel (RP), and (g) center parallel (CP) [6].

such as unknown or confusing data. To this end, we first introduce driving motion types of the object vehicles' lane change and a method for extracting the feature vector in this section.

A. DRIVING MOTIONS OF OBJECT VEHICLE

For lane change intention classification of an object vehicle, possible motions are presented in Fig. 2 [6]. We divided the classes into seven types of driving motions of the object vehicle based on the ego vehicle. Assuming that the ego vehicle is keeping the lane, the driving situation with an object vehicle can be divided into three cases: 1) The object vehicle cuts-in toward the ego vehicle, 2) The object vehicle cuts-out away from the front of the ego vehicle, and 3) The object vehicle and the ego vehicle drive parallel. Then, the cut-in case can be thought of as two situations where an object vehicle cuts-in from the left or right direction. Likewise, the cut-out case considers two situations where the object vehicle cuts-out to the left or right direction in front of the ego vehicle. Finally, the parallel case takes into account a situation in which an object vehicle runs parallel to the left or right of the ego vehicle, and one in which the object vehicle runs parallel to the front of the ego vehicle in the same lane. As an application using these seven driving motions of the object vehicle, whether the object vehicle is in the ego vehicle's path or not can be determined. In other words, the CIPV is detected as the closest object vehicle in the path of the ego vehicle when 'Left Cut-in,' 'Right Cut-in,' and 'Center Parallel' corresponding to 'Class 1,' 'Class 2,' and 'Class 7,' respectively; otherwise, it is not selected.

B. FEATURE EXTRACTION

Most machine learning, such as the MSVM, uses a feature vector to classify each class using the salient feature. The feature vector is a set of numerical values of the elements showing different properties for each class. When predicting the object vehicle's motion described previously, the feature vector can use information such as relative lateral distance and velocity measured from a radar system. However, it might not be useful to use raw data of the relative lateral distance and velocity given by the radar system due to its low lateral resolution. Furthermore, it is necessary to consider the road curvature to effectively predict the object vehicle's driving motion on a curved road.

A commercial radar system gives a relative longitudinal distance r_x and a relative lateral distance r_y as depicted

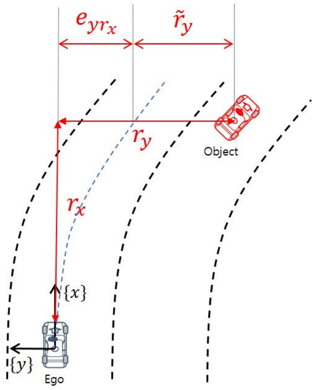


FIGURE 3. Illustration of the four variables r_x, r_y, $\tilde{r}_y$, e_{yr_x} in the vehicle coordinate system $\{xy\}$.

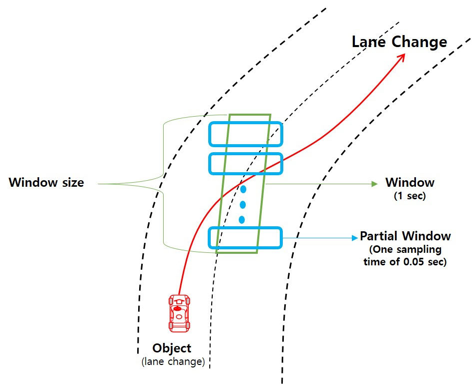


FIGURE 4. Definition of time-window and time-window size.

TABLE 1. Tendencies to relative lateral distance and velocity.

Class	1	2	3	4	5	6	7
r_y	$-$	$+$	$+$	$-$	$+$	$-$	ϵ
$\dot{r}_y$	$+$	$-$	$+$	$-$	ϵ	ϵ	ϵ

Notes: $+$ and $-$ are the positive and the negative sign where the values in each class are larger and lower than $\bar{\epsilon}$ for $\bar{\epsilon} > 0$, respectively. Also, ϵ satisfies $|\epsilon| < \bar{\epsilon}$.

in Fig. 3 and their velocities in the vehicle coordinate frame. In this paper, we define the lateral offset at the relative longitudinal distance to effectively estimate its relative distance from the ego vehicle's lane on a curved road. With the assumption that the ego vehicle is driving with a circular motion [17], the lateral offset at the relative longitudinal distance e_{yr_x} is obtained by

$$e_{yr_x} = \frac{\rho_v}{2} {r_x}^2 = \frac{\dot{\psi}}{2V_x} {r_x}^2 \tag{1}$$

where ρ_v is the circular curvature of the lane drawn by the path of the ego vehicle, and $\dot{\psi}$ and V_x are the yaw rate and the longitudinal velocity of the ego vehicle, respectively. After that, we introduce two new variables, the relative lateral distance at the relative longitudinal distance $\tilde{r}_y$ by

$$\tilde{r}_y = r_y - e_{yr_x} \tag{2}$$

where the variables $\tilde{r}_y$, r_y, e_{yr_x} are described in Fig. 3, and its velocity $\dot{\tilde{r}}_y$. However, $\dot{\tilde{r}}_y$ does not directly come from the vehicle sensor, and it is obtained through the estimator described in section II-C. Table 1 shows how the variables distinguish the presented driving motions of the object vehicle. In practice, when the object vehicle cuts-in or cuts-out, the relative lateral distance and velocity have a certain arbitrary value with a sign. If the ego vehicle and the object vehicle drive parallel, the relative lateral distance also has a certain arbitrary value with a '+' or '−' sign, but the relative lateral velocity has a value close to zero. Therefore, using two variables of $\tilde{r}_y$ and $\dot{\tilde{r}}_y$ as the features is helpful to classify driving motions of the object vehicle. More exact values for training and these analyses will be researched in section IV-A.

The object vehicle's driving behavior quite depends on the driver's experience and environmental situation such as traffic congestion and road curvature during cut-in or cut-out. It is not simple to classify the driving motion based on the relative lateral distance and velocity relationship at each sample time. Furthermore, it is very challenging to model human factors over the lane change. To this end, a time-window method is used for establishing the feature vectors as shown in Fig. 4 [6], [18]. In discrete-time with a sampling period T_r of the radar system at time $t = kT_r$, the feature vector x_s is defined as follows

$$x_s(k) = \left[\tilde{r}_y(k)^T,\ \dot{\tilde{r}}_y(k)^T\right]^T \in R^{2N_r} \tag{3}$$

where

$$\tilde{r}_y(k) = [\tilde{r}_y(k-(N_r-1)), \cdots, \tilde{r}_y(k-1), \tilde{r}_y(k)]^T$$
$$\dot{\tilde{r}}_y(k) = [\dot{\tilde{r}}_y(k-(N_r-1)), \cdots, \dot{\tilde{r}}_y(k-1), \dot{\tilde{r}}_y(k)]^T$$

and N_r is the number according to a time-window size. In this paper, N_r is 20 samples for a time-window over 1 second with a radar sampling time T_r of 0.05 second.

C. KALMAN FILTERING

This section describes how feature vectors based on the circular motion estimation are made using a KF to obtain $\dot{\tilde{r}}_y$ not available from commercial radar. Also, a low-cost radar system has poor resolution of lateral motion, e.g., 0.5 degree, due to its cost and algorithm characteristics dealing with multi-targets based on electromagnetic waves and using the doppler effect. Methods using high-resolution radar and fusing multi-radars to improve the lateral motion accuracy are researched in [19], [20], respectively, but it is not yet easy to be applied to a commercial radar system due to its cost.

Therefore, we estimate $\dot{\tilde{r}}_y$ and filter out $\tilde{r}_y$ using the KF by considering the constant velocity model. Without the loss of generality, let us define the state $x_f = \left[\tilde{r}_y\ \dot{\tilde{r}}_y\right]^{\mathrm{T}}$ and the measurement output $y_f = \tilde{r}_y$. In addition, we can assume that the two noise sources, system and measurement noises, ω_k

and υ_k are independent ($E[\omega_i \upsilon_j^T] = 0$). Then we can model the motion of object vehicle assuming constant velocity as

$$\begin{aligned} x_f(k+1) &= \Phi x_f(k) + \omega(k) \\ y_f(k) &= C x_f(k) + \upsilon(k) \end{aligned} \tag{4}$$

where

$$\Phi = \begin{bmatrix} 1 & T_r \\ 0 & 1 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 0 \end{bmatrix},$$
$$\omega_k \sim N(0, Q), \; \upsilon_k \sim N(0, R).$$

The optimal state estimator minimizing the variances of the estimation error will then be the KF as

$$\begin{aligned} \bar{x}_f(k+1) &= \Phi \hat{x}_f(k) \\ \hat{x}_f(k) &= \bar{x}_f(k) + L(y_f(k) - C\bar{x}_f(k)). \end{aligned} \tag{5}$$

Observer gain L is solved by using the discrete-time algebraic Riccati equation [21] as

$$Y = \Phi Y \Phi^T - \Phi Y C^T (CYC^T + R)^{-1} CY\Phi^T + Q \tag{6}$$

where $L = YC^T(CYC^T + R^{-1})$. Using the discrete-time model (4) and the KF (5), $\hat{r}_y$ and $\hat{\dot{r}}_y$ can be estimated. The Q and R are the covariance matrices of the corresponding ω_k and υ_k, respectively, which are the system noise and measurement noise under the assumption that both are white and Gaussian with known covariances. The detailed analysis of the radar system is presented in [22].

III. MULTI-CLASS SUPPORT VECTOR MACHINE BASED ON OPEN SET RECOGNITION CONCEPT

For the classification problem, since almost all machine learning needs full handcrafted data, it can make wrong decisions for information not included in the training data set or by confused data even with probability [13], [14]. Furthermore, a massive data set is necessary to catch the corner cases, but it is not proven how to cope with such corner cases [16]. To conservatively detect the CIPV, we introduce an MSVM classification method utilizing the open set recognition concept in this section.

A. BINARY-CLASS SUPPORT VECTOR MACHINE

The SVM is a type of machine learning used for data classification when the data has two classes [23], [24]. It has been used in various industrial applications because it is more practical than other deep learning methods due to its low computational power requirements and high performance. Let us consider a binary classification problem. From now on, we will refer to SVM binary-class SVM (BSVM) to avoid confusion with MSVM. A data set is given the training vectors $x_i \in \mathbb{R}^m$ and the label vector $y_i \in \{-1, 1\}$ in two classes for $i = 1, \ldots, n$. Then, the function $f_s(\cdot) \in \mathbb{R}$ is

$$f_s(x) = \langle \varphi(x), \alpha \rangle + \beta = 0 \tag{7}$$

where $\varphi(x) \in \mathbb{R}^m$ is a nonlinear mapping function of x, $\alpha \in \mathbb{R}^m$ is the weight vector, and $\beta \in \mathbb{R}$ is the bias. Also, $\langle \cdot, \cdot \rangle$ means the inner product. Then, the best separating hyperplane is obtained by solving the problem maximizing the following optimization problem as

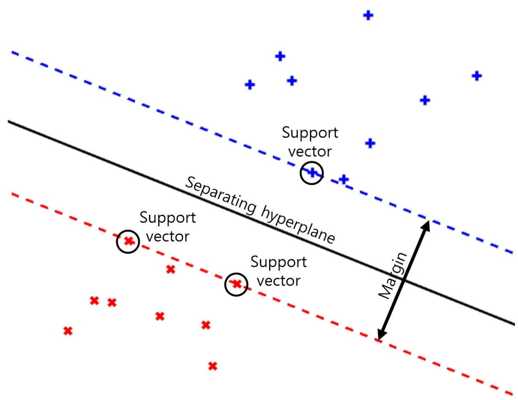


FIGURE 5. **Support vectors and the hyper-plane.**

$$\begin{aligned} &maximize: J = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j \zeta(x_i, x_j) \\ &subject\ to: \sum_{i=1}^{n} \lambda_i y_i = 0 \; (\lambda_1, \cdots, \lambda_n \geq 0) \\ &kernel: \zeta(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle \\ &\qquad\qquad = \exp(-\tfrac{1}{2\sigma^2} \|x_i - x_j\|^2). \end{aligned} \tag{8}$$

In this paper, Gaussian radial basis function is adapted as the kernel function [25] as represented by (8) to accommodate nonlinear behavior of the object vehicle such as a lane change.

B. MULTI-CLASS SUPPORT VECTOR MACHINE

The BSVM classifies binary classes based on their original maximum margins. However, because there are various actual driving situations, it is often necessary to classify three or more categories. MSVM generally splits the problem into several binary problems that can be applied immediately to the BSVM and then reassemble. Let us consider N categories for the MSVM. First, the MSVM requires the design of a classification matrix $M_s \in R^{N \times L}$ where L is the number of binary learners that determines the classes. For the matrix M_s, each element $m_{s,pq}$ consists of $m_{s,pq} \in \{1, 0, -1\}$ for $p = 1, 2, \ldots, N$ and $q = 1, 2, \ldots, L$. Additionally, the number of L complies with the split method such as the One-Against-All method or the One-Against-One method [26], [27]. In this paper, we use the One-Against-One method for a more accurate classification performance, and thus $L = {}_7\mathrm{C}_2$.

Then, a loss-weighted based class prediction method is used to determine the resulting class. It is well known that the method improves the classification accuracy by keeping the loss values for all classes in the same dynamic range [28]. The loss-weighted function aggregates the binary classifiers' results, which produces the minimum average of the binary losses. Using the loss-weighted function, the MSVM scores

Algorithm 1 The MSVM Classification Algorithm Based on the Open Set Recognition Concept

Require: Means μ_p and libMR models $\rho_p = (\tau_p, \lambda_p, \kappa_p)$ about each true BSVM score vector
Require: α, the number of "top" classes to revise
1: Measure $s = (s_1, s_2, \ldots, s_L)$, $z = (z_1, z_2, \ldots, z_N)$
2: Let $a(i) = \text{argsort}(z_p)$; Let $\omega_p = 1$
3: **for** $i = 1, \ldots, \alpha$ **do**
4: $\quad \omega_{a(i)} = 1 - \frac{\alpha - i}{\alpha} e^{-\left(\frac{\|s-\tau_{a(i)}\|}{\lambda_{a(i)}}\right)^{\kappa_{a(i)}}}$
5: **end for**
6: Define $\hat{z}_0 = \Sigma_i z_i (1 - \omega_i)$
7: Update MSVM scores $\hat{z}_p = z_p \cdot \omega_p$
8: Return $\hat{p} = \text{argmin}_p \hat{z}_p$

$z_p(\cdot, \cdot)$ is represented as follows

$$z_p(m_s, s_q) = \frac{\sum_{q=1}^{L} |m_{s,pq}| g(m_{s,pq}, s_q)}{\sum_{q=1}^{L} |m_{s,pq}|}, \quad \text{for } p = 1, 2, \ldots, N, \tag{9}$$

where

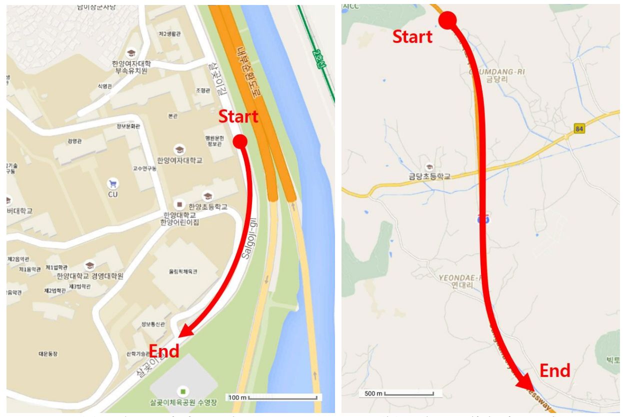
- $m_{s,pq}$: element (p, q) of the classification matrix M_s (i.e., corresponding to class p of binary learner q)
- s_q: BSVM score of binary learner q for an observation (i.e., $s_q = f_{s,q}(x)$)
- $g(\cdot, \cdot)$: binary loss function.

For the binary loss function, we use the hinge loss function as $g(m_{s,pq}, s_q) = \max(0, 1 - m_{s,pq} \cdot s_q)$. Consequently, the predicted class is determined as the class p taking a minimized value z_p of N classes.

C. MULTI-CLASS SUPPORT VECTOR MACHINE BASED ON OPEN SET RECOGNITION CONCEPT

Since the MSVM is a deterministic classification method based on the training data set, it could make wrong decisions due to information not included by the training data set or due to confused data on decision boundary [13]. Therefore, when detecting a lane change intention, classification results near the decision boundary can be confused due to uncertainty, and it might be difficult to make conservative detection. To cope with this problem, we present an MSVM classification method, including an open set recognition concept. The open set recognition concept is adapted using Meta-Recognition and allows for classification of "fooling" and unrelated open set data presented to the system [14]. To apply the open set recognition concept in the MSVM, we utilize the BSVM scores about if the input is "far" from known training data. The proposed method can formally handle unknown or confusing classes during operation.

Algorithm 1 summarizes the steps for the MSVM classification computation. To develop an MSVM classification algorithm including open set, let ρ_p be a vector of meta-recognition models for each class, which includes parameters τ for shifting the data as well as the Weibull



(a) The training place (b) The validation place

FIGURE 6. The experiment places: (a) to obtain the training data set and (b) to validate lane change intention of the object vehicle.

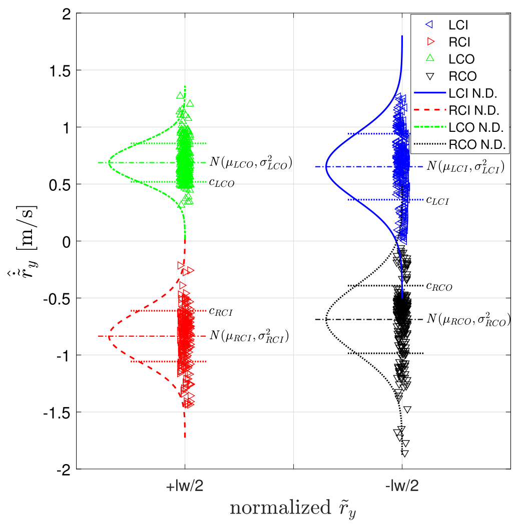
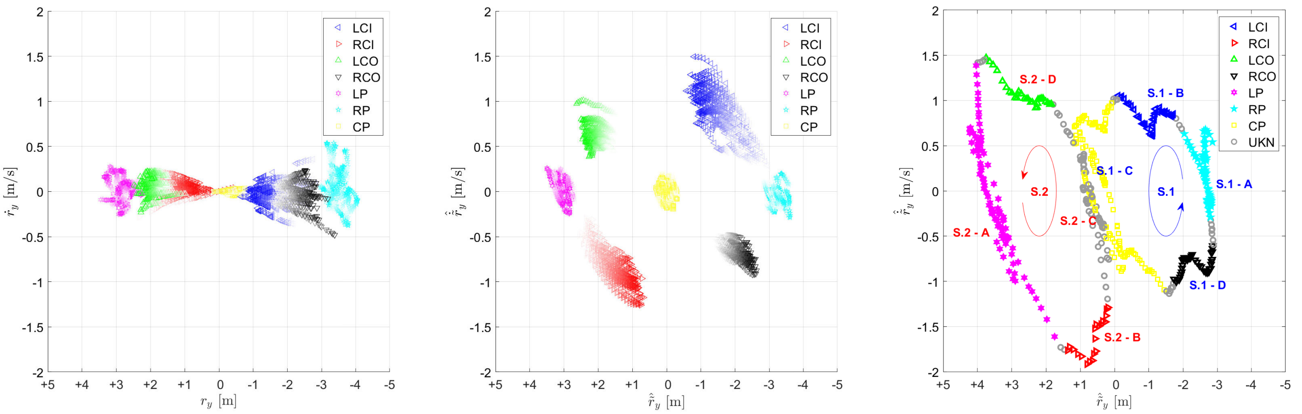


FIGURE 7. $\dot{\tilde{r}}_y$ analysis for the lane change of the object vehicle.

shape and scale parameters, κ and λ, respectively (see [14] for details). For convenience, we define the unknown/confusing class to be at index 0. We use the Weibull cumulative distribution function (CDF) probability (line 4 of **Algorithm 1**) on the distance between s and μ for the score of the uncertain estimation. The model ρ is computed using the BSVM scores associated with class p, namely, scores that were classified correctly during the training process. We expect the extreme value theorem (EVT) function of distance to provide a meaningful revised score only for few top ranks. Thus, in line 3 of **Algorithm 1**, we compute weights for the α largest activation classes and use it to scale the MSVM scores based on the Weibull CDF. We then compute revised MSVM scores with the top scores changed and a pseudo-activation for the unknown/confusing class, keeping the total activation level constant. Thus, the open set recognition concept provides the revised MSVM scores that support uncertain estimation when the unknown/confusing class ($\hat{p}$=0) has the smallest revised score.



(a) r_y-$\dot{r}_y$ measured from the radar directly (b) $\hat{r}_y$-$\hat{\dot{r}}_y$ obtained by the Kalman Filter (c) $\hat{r}_y$-$\hat{\dot{r}}_y$ trajectory and validation results

FIGURE 8. Comparison of relative lateral distance-velocity plot: (a) r_y-$\dot{r}_y$ measured from the radar directly, (b) $\hat{r}_y$-$\hat{\dot{r}}_y$ obtained by the KF, and (c) $\hat{r}_y$-$\hat{\dot{r}}_y$ trajectory and the proposed MSVM classification results at each sample time using the validation data. (a) and (b) used the same training data set according to the time-window that the gradation from a light to a dark color represents the flow of time. The feature extraction method facilitates the classification by considering the ego vehicle's yaw rate motion. (c) shows that classification for unknown class on decision boundaries is needed in a real driving situation.

IV. EXPERIMENTS AND RESULTS

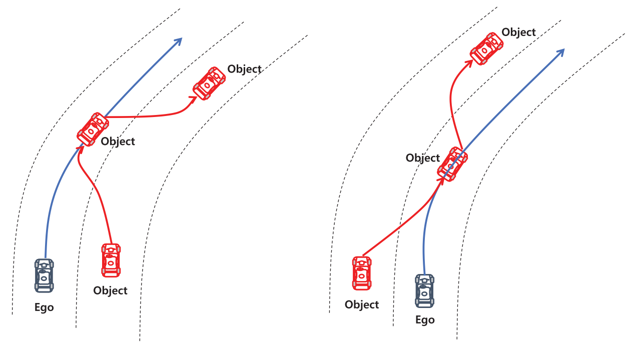
This section introduces experimental data and analyzes them to conservatively detect lane change intention of surrounding vehicles. Then, experiment results show that the proposed system outperforms the detecting performance of commercial radar systems.

A. EXPERIMENTAL TRAINING DATA AND ITS ANALYSIS

The experiment used two test vehicles: one for the ego vehicle and the other for the object vehicle. The ego vehicle is a luxury passenger car, Genesis from Hyundai Motors. The front radar system, in which a long-range radar and a mid-range radar are embedded, mounted on the ego vehicle's grille, and transmitted information of objects through a Controller Area Network (CAN) communication system. The object vehicle used in the experiment is a small Sport Utility Vehicle (SUV), Tucson from Hyundai Motors. The training data set was collected on a local curved road which is a two-lane road at the rear of Hanyang University, as shown in Fig. 6 (a). Regarding seven driving motions, each driving motion was tried about 30 times to obtain 240 data sets. Then, for validation of the presented algorithm, experiments to validate the lane change intention classification of the object vehicle were conducted on a different road, the Jungbunaeryuk-Expressway, which is a two-lane road, as shown in Fig. 6 (b). On two roads, the first lane and the second lane are defined as innermost and outer lanes. The two vehicles' velocities were set from 30 $\sim$ 40 kph for the training data set and from 70 $\sim$ 90 kph for the validation data set. The seven driving situations and the number of experiments for each situation is depicted in Fig. 2 are described as follows:

1) LEFT CUT-IN (LCI), "32 TIMES"

The ego vehicle was moving in the first lane while the object vehicle was riding in the second lane. The object vehicle changed its lane ahead of the ego vehicle. The LCI is labeled as true when $\hat{r}_y$ is negative and become more than a half of the lane width, and $\hat{\dot{r}}_y$ is more than c_{LCI}.



(a) *Scenario 1* (b) *Scenario 2*

FIGURE 9. Two scenarios to be used for algorithm verification.

2) RIGHT CUT-IN (RCI), "31 TIMES"

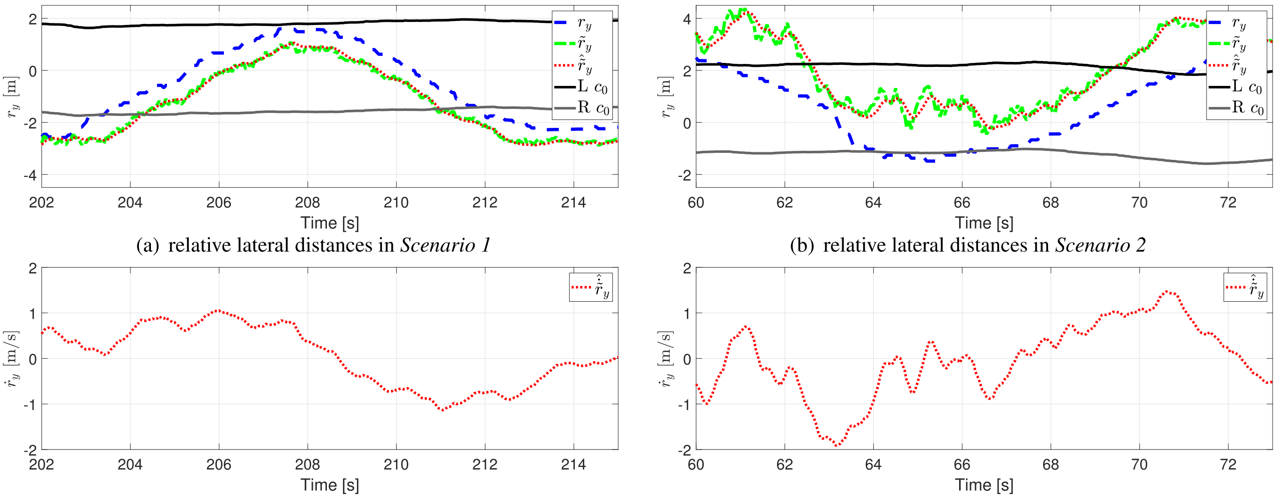
The ego vehicle was in the second lane while the object vehicle was in the first lane. The object vehicle changed its lane ahead of the ego vehicle. The RCI is labeled as true when $\hat{r}_y$ is positive and become less than a half of the lane width, and $\hat{\dot{r}}_y$ is less than c_{RCI}.

3) LEFT CUT-OUT (LCO), "30 TIMES"

Both the ego and the object vehicles were in the same lane, the second lane. The object vehicle in the front of the ego vehicle cut out into the first lane. The LCO is labeled as true when $\hat{r}_y$ is positive and become more than a half of the lane width, and $\hat{\dot{r}}_y$ is more than c_{LCO}.

4) RIGHT CUT-OUT (RCO), "27 TIMES"

Both the ego and object vehicles were in the same lane, the first lane. The object vehicle in the front of the ego vehicle



(a) relative lateral distances in *Scenario 1*
(b) relative lateral distances in *Scenario 2*
(c) relative lateral velocities in *Scenario 1*
(d) relative lateral velocities in *Scenario 2*

FIGURE 10. Plots of the relative lateral distances and velocities in *Scenario 1* and 2, respectively. From (a) and (b), r_y (blue dashed line) is measured from the radar sensor, $\tilde{r}_y$ (green dash-dotted line) is obtained from Eq. (2), and $\hat{\tilde{r}}_y$ (red dotted line) is filtered by Kalman filter. Left and right lane offsets (black and grey solid lines), L and R c_0's, were measured from the external camera sensor for comparison with which the ego vehicle is driving with a circular motion. Further, from (c) and (d), $\dot{\hat{\tilde{r}}}_y$ (red dotted) is estimated by Kalman filter.

cut out into the second lane. The RCO is labeled as true when $\hat{\tilde{r}}_y$ is negative and become more than a half of the lane width, and $\dot{\hat{\tilde{r}}}_y$ is less than c_{RCO}.

5) LEFT PARALLEL (LP), "31 TIMES"

The ego vehicle was in the second lane while the object vehicle was in the first lane. Both vehicles proceeded in each lane without changing lanes. The LP is labeled as true when $\hat{\tilde{r}}_y$ is positive and about the lane width, and the absolute value of $\dot{\hat{\tilde{r}}}_y$ is less than 0.2 m/s.

6) RIGHT PARALLEL (RP), "29 TIMES"

The ego vehicle was in the first lane while the object vehicle was in the second lane. Both vehicles were running in each lane without changing lanes. The RP is labeled as true when $\hat{\tilde{r}}_y$ is negative and about the lane width, and the absolute value of $\dot{\hat{\tilde{r}}}_y$ is less than 0.2 m/s.

7) CENTER PARALLEL (CP), "60 TIMES"

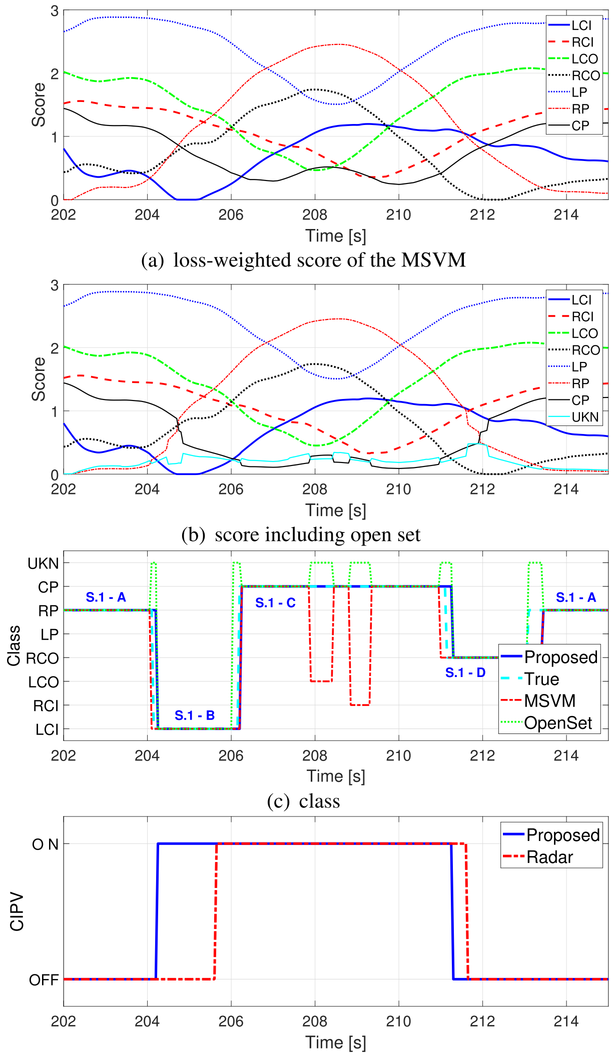
Both vehicles were in the same lane, either the first or second, without changing lanes. The CP is labeled as true when absolute values of $\hat{\tilde{r}}_y$ and $\dot{\hat{\tilde{r}}}_y$ are less than 0.2 m and 0.2 m/s.

Above presented variables c_i for i = {LCI, RCI, LCO, RCO} can be chosen by analyzing tendency of $\dot{\hat{\tilde{r}}}_y$ during lane change of the object vehicle as shown in Fig. 7. This data presents the normal distribution of $\dot{\hat{\tilde{r}}}_y$ using the experimental data when the object vehicle arrives at a half of the lane width measured from a camera sensor, lw, during lane change of the object vehicle. In Fig. 7, μ_i and σ_i denote expectation and standard deviation of the distributions for each lane change situation, respectively. The details of the values are presented in Table. 2. Using the information, the boundaries c_i are designed to cover reasonable $\dot{\hat{\tilde{r}}}_y$ for the experimental training data.

TABLE 2. Normal distribution analysis of $\dot{\hat{\tilde{r}}}_y$ for the lane change of the object vehicle.

Class	μ	σ	c : $\dot{\hat{\tilde{r}}}_y$ boundary for training
1 (LCI)	0.652	0.289	$\mu_{LCI} - \sigma_{LCI} = 0.363$
2 (RCI)	−0.834	0.223	$\mu_{RCI} + \sigma_{RCI} = -0.612$
3 (LCO)	0.688	0.169	$\mu_{LCO} - \sigma_{LCO} = 0.519$
4 (RCO)	−0.688	0.297	$\mu_{RCO} + \sigma_{RCO} = -0.391$

Each feature vector consists of a set of ($\hat{\tilde{r}}_y$, $\dot{\hat{\tilde{r}}}_y$) for a time-window period of one second. In other words, each set of ($\hat{r}_y$, $\dot{\hat{r}}_y$) for one second is labeled. Figure 8 shows the comparison of relative lateral distance-velocity plots that (a) is r_y-$\dot{r}_y$ measured from the radar directly and (b) is $\hat{\tilde{r}}_y$-$\dot{\hat{\tilde{r}}}_y$ obtained by feature extraction using the KF. (a) and (b) used the training data set according to the time-window that the gradation from a light to a dark color represents the flow of time. Figure 8 (c) shows $\hat{\tilde{r}}_y$-$\dot{\hat{\tilde{r}}}_y$ trajectory and the proposed MSVM classification results at each sample time using the validation data. It is obvious that it is not easy to classify the driving mode without using the Kalman filter proposed in this paper. We can see that the proposed feature extraction method can be modeled easily for machine learning such as SVM and improve the performance classifying lane change intention of the object vehicle. If r_y-$\dot{r}_y$, as plotted in Fig. 8 (a), are used as the feature vectors, it is challenging to classify driving motions between LCI and RCO or between RCI and



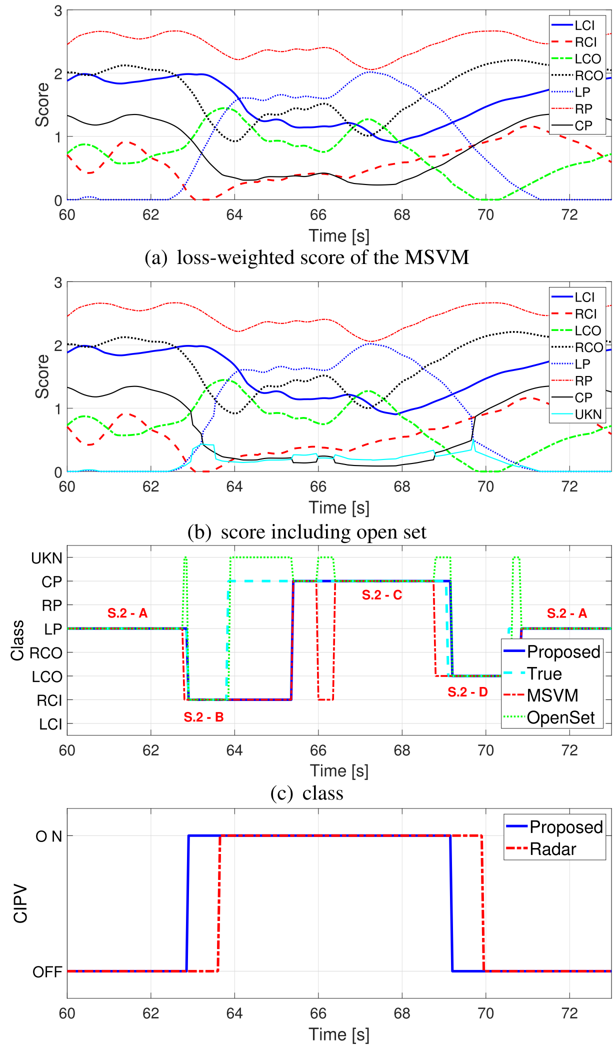
(a) loss-weighted score of the MSVM

(b) score including open set

(c) class

(d) CIPV

FIGURE 11. Results of the proposed method for *Scenario 1*: (a) loss-weighted scores of the MSVM, (b) scores based on the open set, (c) classes including the open set, and (d) results about the CIPV application.



(a) loss-weighted score of the MSVM

(b) score including open set

(c) class

(d) CIPV

FIGURE 12. Results of the proposed method for *Scenario 2*: (a) loss-weighted scores of the MSVM, (b) scores based on the open set, (c) classes including the open set, and (d) results about the CIPV application.

LCO using the MSVM. Therefore, as plotted in Fig. 8 (b), using $\hat{\hat{r}}_y$ and $\dot{\hat{r}}_y$ is more well-suited to classify lane change intention of the object vehicle. However, classification in real driving should also be taken into account on boundaries since the classification results make confusing decisions. To this end, we propose an MSVM classification utilizing an open set recognition concept and showing MSVM classification results as shown in Fig. 8 (c). The details of the classification results will be covered in the following experiment results.

B. EXPERIMENT RESULTS

We validated the proposed algorithm using data measured from a different place not included in the data set used for training. The two scenarios to cover all the driving situations are described below. Each scenario is shown in Fig. 9.

1) SCENARIO 1

- S.1-A (RP, CIPV off): The ego vehicle rides in the first lane, and the object vehicle rides in the second lane.
- S.1-B (LCI, CIPV on): The object vehicle cuts-in from the second lane to the first lane.
- S.1-C (CP, CIPV on): The object vehicle and the ego vehicle ride front and back in parallel in one lane.
- S.1-D (RCO, CIPV off): The object vehicle cuts-out from the first lane to the second lane.
- Then, the situation returns to S.1-A again.

2) SCENARIO 2

- S.2-A (LP, CIPV off): The ego vehicle rides in the second lane, and the object vehicle rides in the first lane.
- S.2-B (RCI, CIPV on): The object vehicle cuts-in from the first lane to the second lane.

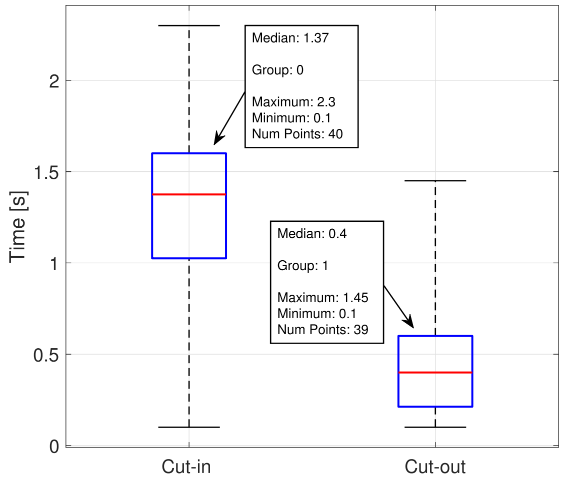


FIGURE 13. A grouped box chart for cut-in/out performance distribution compared with the radar. In the chart, groups 0 and 1 are relative times for detecting and canceling the CIPV, respectively, i.e., detected times of the proposed method minus detected times by the radar. The chart shows that the proposed method detects or cancels the CIPV earlier than the commercial radar system.

- S.2-C (CP, CIPV on): The object vehicle and the ego vehicle ride front and back in parallel in one lane.
- S.2-D (LCO, CIPV off): The object vehicle cuts-out from the second lane to the first lane.
- Then, the situation returns to S.2-A again.

In these scenarios, lane changing was performed at 80 kph on a curved road in the validation place so that the execution time from the starting lane changed to its completion. Both vehicles maintained a relative longitudinal distance r_x of about 50 $\sim$ 70 m. Detailed information is also observed in Figs. 14 and 15. Figure 10 shows plots of the relative lateral distances and velocities in *Scenarios 1* and *2*, respectively. From Fig. 10 (a) and (b), we can see that $\tilde{r}_y$ compensates for the curved road with respect to r_y measured from the radar system and $\hat{\tilde{r}}_y$ filters out $\tilde{r}_y$. Without compensation of the curved road, r_y could go beyond the lane offset even through both vehicles are keeping the lane. Fig. 10 (c) and (d) show $\hat{\dot{\tilde{r}}}_y$ estimated by Kalman filter since it is difficult to measure.

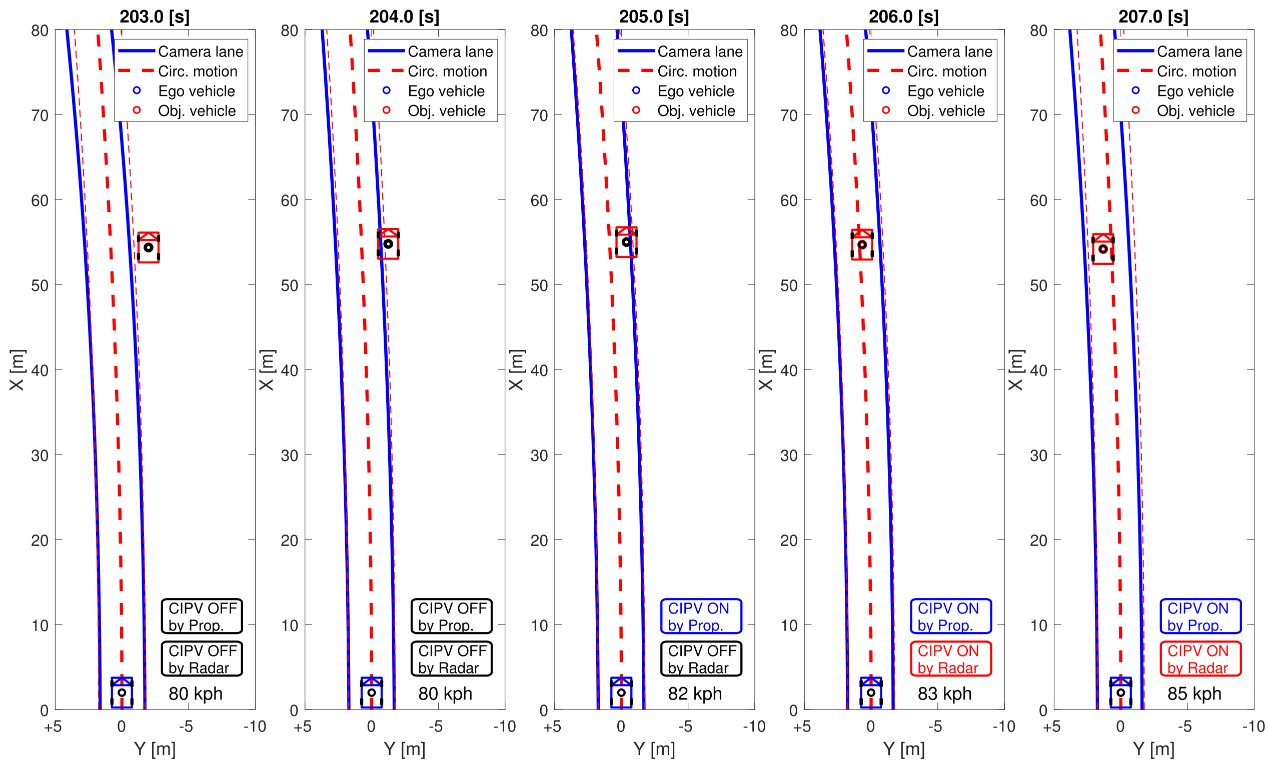
Both Figs. 11 and 12 show the classification results of the lane change intention of the object vehicle using the proposed method. The predicted class and the classification result are determined as the class taking a minimized score in (a) and (b). Without the open set classification, the MSVM can predict faulty results. The proposed method predicts unknown/confusing classes to help control the vehicle conservatively. Furthermore, the classification problem is confusing on boundaries between classes. The results show that the proposed method has confusing classes on these boundaries. The trajectories over two results are also described in Fig. 8 (c). If it is used, it can enable more conservative vehicle control to improve ACC performance and avoid a collision. In Figs. 11 (c) and 12 (c), we applied a method

TABLE 3. The confusion matrix of the performance by MSVM.

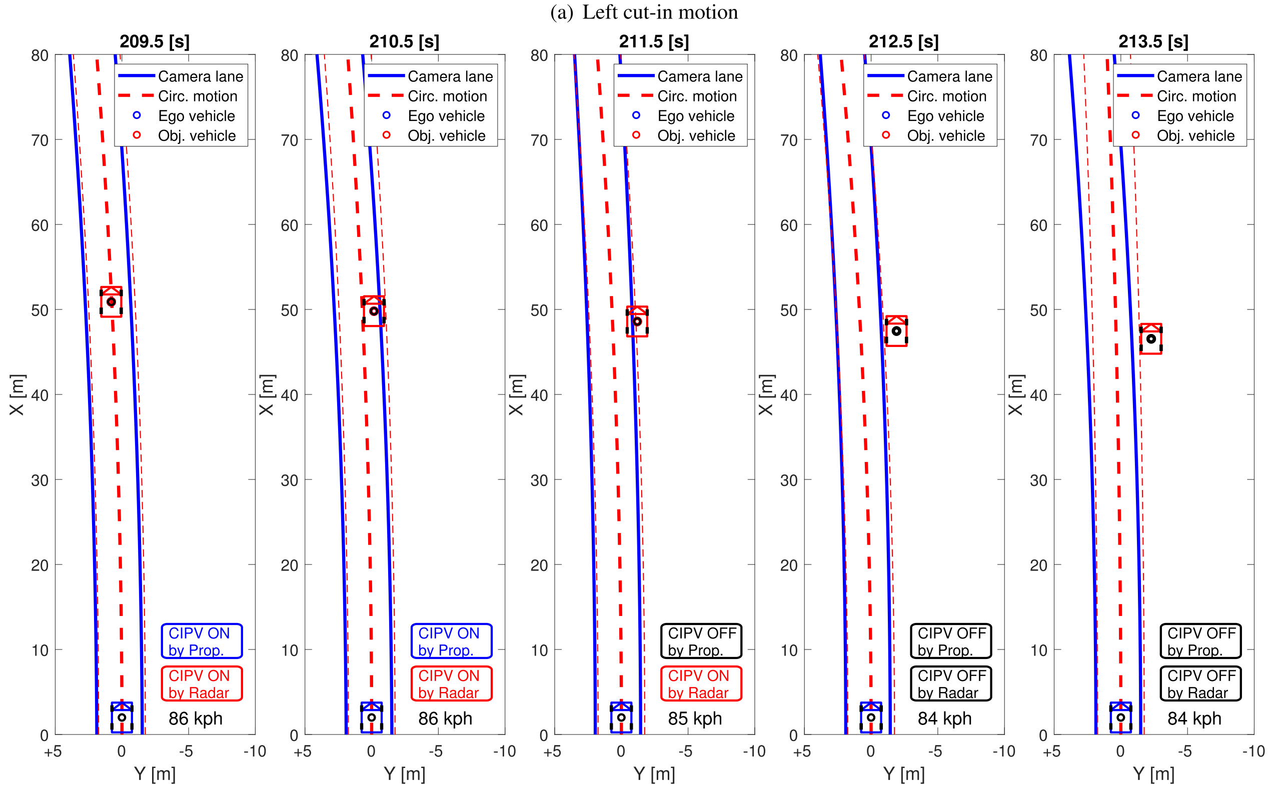
		Actual								
		LCI	RCI	LCO	RCO	LP	RP	CP	Total	Precision (%)
Prediction	LCI	**789**	0	0	0	0	0	55	844	93.5
	RCI	0	**670**	6	0	1	0	219	896	74.8
	LCO	0	1	**581**	0	125	0	24	667	87.1
	RCO	0	0	0	**648**	0	87	0	725	89.4
	LP	0	54	0	0	**1,916**	0	0	2,042	93.8
	RP	45	0	0	3	0	**1,292**	0	1,379	93.7
	CP	0	0	80	74	0	0	**3,469**	3,767	92.1
	Total	834	725	667	725	2,042	1,379	3,767	**10,154**	
	Recall (%)	**94.6**	**92.4**	**87.1**	**89.4**	**93.8**	**93.7**	**92.1**	**Accuracy:** (%)	**92.2**

keeping the "OpenSet" result at the last sample time as one of the applications utilizing open set results to detect the CIPV in Figs. 11 (d) and 12 (d). It is shown with a solid blue line, the "Proposed" result, in Figs. 11 (c) and 12 (c). True data, the dashed cyan line, was handcrafted using perimeter limits as described for the training data set. With the true data, a confusion matrix of the presented algorithm is presented in Table 3. Through the confusion matrix, we obtained an accuracy of 92.2% as well as significant precision and recall results for each situation.

Moreover, compared with the CIPV of a commercial radar system, we also show that the presented algorithm detects and cancels detecting the CIPV faster than radar. Figure 13 shows a grouped box chart for the statistical performance of the CIPV detection compared with commercial radar. In the chart, groups 0 and 1 are relative times for detected and canceling the CIPV, respectively, i.e., detecting times of the proposed method minus detected times by the radar. In the case of the object vehicle cut-in tests performed 40 times for all scenarios in *1* and *2*, the algorithm has a median time of 1.37 s, minimum 0.1 s, and a maximum of 2.3 s. When the object vehicle cuts-out 39 times, a median 0.4 s, minimum 0.1 s, and maximum 1.45 s are shown in Fig. 13. Additionally, the upper and lower points at the blue boxes mean 75% and 25% bounds of each group data, respectively. Furthermore, 4 representative driving motions in *Scenarios 1* and *2* are described in Figs. 14 and 15. Blue and red vehicles are ego and object vehicles measured from the front radar sensor, respectively. The blue dashed line is plotted using lane offset c_0, heading offset c_1, road curvature c_2, and the derivative of road curvature c_3 measured from a camera sensor, and the broken red line is plotted using ρ_v in Eq. 1 and lane width $\pm 1.7m$ is applied in this paper. We can see again that the proposed algorithm detects and cancels the CIPV before the radar sensor, even on a curved road in each motion of the object vehicle.

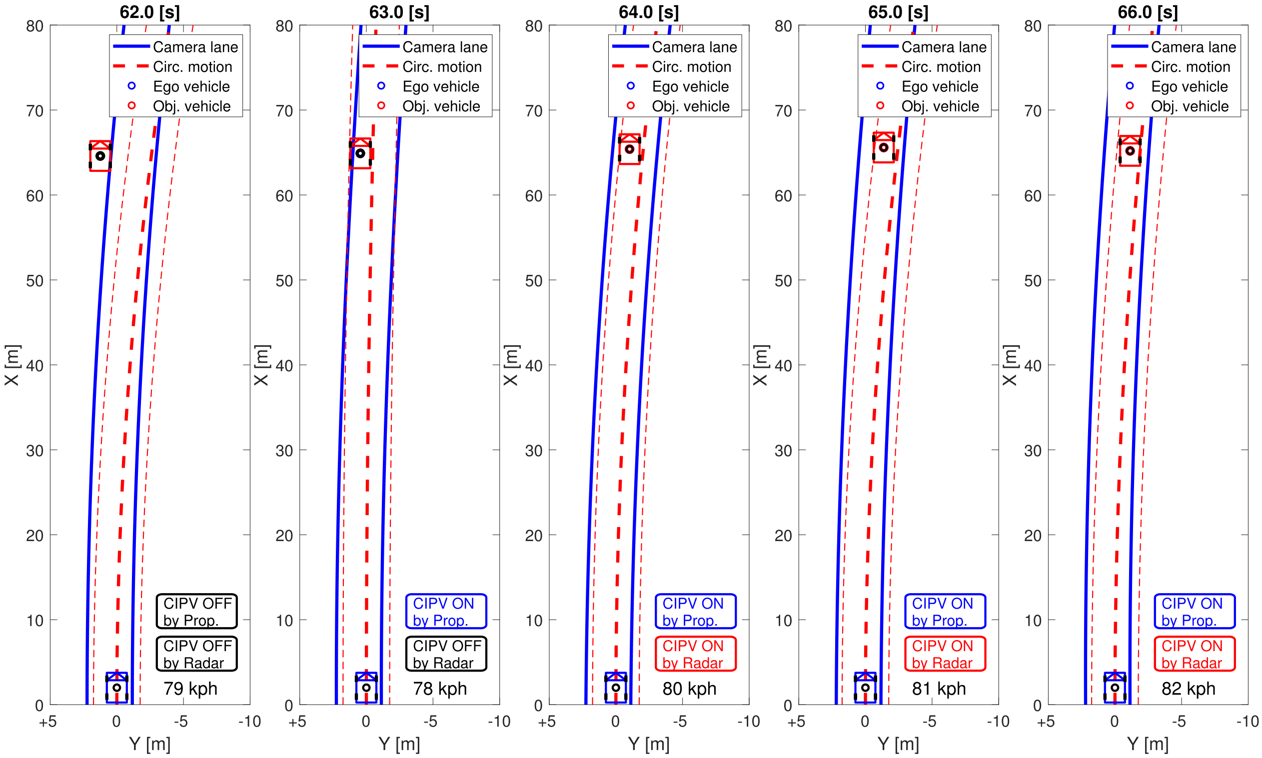


(a) Left cut-in motion

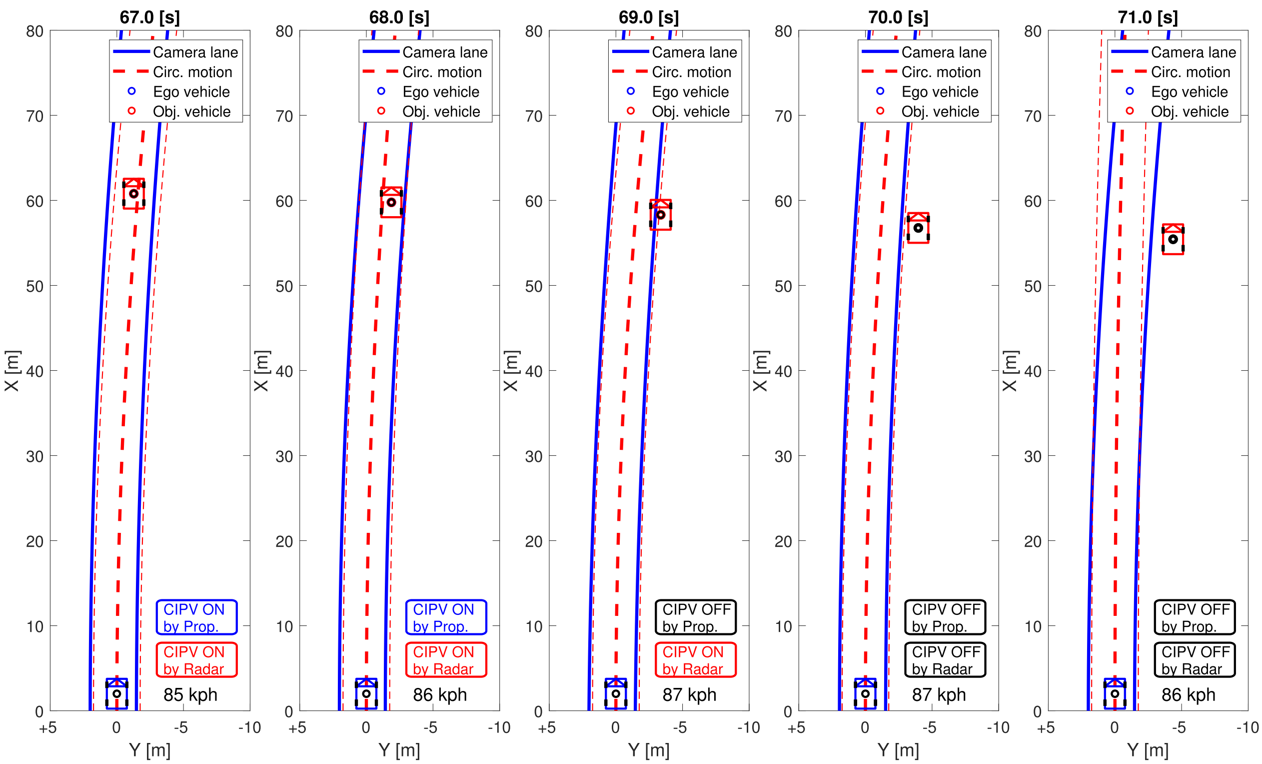


(b) Right cut-out motion

FIGURE 14. Driving motions in *Scenario 1*: (a) left cut-in motion and (b) right cut-out motion. Blue and red vehicles are ego and object vehicles measured from radar sensor, respectively. The blue dashed line is plotted using lane offset c_0, heading offset c_1, road curvature c_2, and the derivative of road curvature c_3 measured from camera sensor and the red broken line is ploted using ρ_v in Eq. 1 and lane width ± 1.7 m is applied in this paper. In each motion situation of the object vehicle, we can see again that the proposed algorithm detects and cancels the CIPV before radar sensor even on a curved road.



(a) Right cut-in motion



(b) Left cut-out motion

FIGURE 15. Driving motions in *Scenario 2*: (a) right cut-in motion and (b) left cut-out motion. Blue and red vehicles are ego and object vehicles measured from radar sensor, respectively. The blue dashed line is plotted using lane offset c_0, heading offset c_1, road curvature c_2, and the derivative of road curvature c_3 measured from camera sensor and the red broken line is ploted using ρ_v in Eq. 1 and lane width ± 1.7 *m* is applied in this paper. In each motion situation of the object vehicle, we can see again that the proposed algorithm detects and cancels the CIPV before radar sensor even on a curved road.

V. CONCLUSION

This paper proposed a classification method utilizing an open set recognition concept to conservatively detect lane change intention of surrounding vehicles. To this end, we first divided the driving motions of an object vehicle into seven types and then detected the CIPV according to each motion. Classifying the lane change intention of an object vehicle was made using salient feature vectors that consist of defined relative lateral distances and velocities based on the circular motion estimation obtained by a commercial front radar and in-vehicle signal's time-window. In that case, we used a KF to overcome the low-resolution limits for lateral information of the radar sensor and to obtain the relative lateral velocity not available from the radar system. Then, we presented the MSVM classification algorithm with the open set recognition concept for the classification algorithm to conservatively detect lane change intention of the surrounding vehicles. The proposed algorithm was validated with a data set not included in the training data set. We observed that the proposed system conservatively copes with wrong decisions. A comparative study with commercial radar was quantitatively made to show the effectiveness of the proposed method. From the experimental results, we also observed that the proposed system could detect and cancel detecting the CIPV earlier with median times of 1.4 sec and 0.4 sec, respectively. Furthermore, we constructed a confusion matrix to evaluate the proposed method's accuracy and achieved an accuracy of 92.2%.

REFERENCES

[1] S.-K. Park, B.-S. Kim, S.-H. Jeong, H.-J. Kang, and P.-J. Yoon, "Lane estimation using lateral histogram in radar based ACC system," in *Proc. 11th Eur. Radar Conf.*, Oct. 2014, pp. 193–196.

[2] S. H. Jeong, J. E. Lee, S. U. Choi, J. N. Oh, and K. H. Lee, "Technology analysis and low-cost design of automotive radar for adaptive cruise control system," *Int. J. Automot. Technol.*, vol. 13, no. 7, pp. 1133–1140, 2012.

[3] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH J.*, vol. 1, no. 1, p. 1, Jul. 2014, doi: 10.1186/s40648-014-0001-z.

[4] W. Kim, C. Mook Kang, Y. Seop Son, S.-H. Lee, and C. Choo Chung, "Vehicle path prediction using yaw acceleration for adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 3818–3829, Dec. 2018.

[5] D. Lee, A. Hansen, and J. Karl Hedrick, "Probabilistic inference of traffic participants' lane change intention for enhancing adaptive cruise control," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 855–860.

[6] H. S. Park, D. J. Kim, C. M. Kang, S. C. Kee, and C. C. Chung, "Object detection in adaptive cruise control using multi-class support vector machine," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1948–1953.

[7] 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. 1277–1282.

[8] S. Yoon and D. Kum, "The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 1307–1312.

[9] 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)*, Oct. 2017, pp. 399–404.

[10] N. Deo and M. M. Trevedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1179–1184.

[11] L. Fang, Q. Jiang, J. Shi, and B. Zhou, "TPNet: Trajectory proposal network for motion prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6796–6805.

[12] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1, no. 2. Cambridge, MA, USA: MIT Press, 2016.

[13] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult, "Toward open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, Jul. 2013.

[14] A. Bendale and T. E. Boult, "Towards open set deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1563–1572.

[15] C. Geng, S. J. Huang, and S. Chen, "Recent advances in open set recognition: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Mar. 18, 2020, doi: 10.1109/TPAMI.2020.2981604.

[16] J.-A. Bolte, A. Bar, D. Lipinski, and T. Fingscheidt, "Towards corner case detection for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 438–445.

[17] Y. S. Son, W. Kim, S.-H. Lee, and C. C. Chung, "Predictive virtual lane method using relative motions between a vehicle and lanes," *Int. J. Control, Automat. Syst.*, vol. 13, no. 3, pp. 471–477, 2015.

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

[19] A. Scheel, C. Knill, S. Reuter, and K. Dietmayer, "Multi-sensor multi-object tracking of vehicles using high-resolution radars," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 558–565.

[20] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Tracking of extended objects with high-resolution Doppler radar," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1341–1353, May 2016.

[21] G. F. Franklin, J. D. Powell, and M. Workman, *Digital Control of Dynamic Systems*. Reading, MA, USA: Addison-Wesley, 1997.

[22] W. Y. Choi, C. M. Kang, S.-H. Lee, and C. C. Chung, "Radar accuracy modeling and its application to object vehicle tracking," *Int. J. Control, Autom. Syst.*, vol. 18, no. 12, pp. 3146–3158, Dec. 2020.

[23] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. Springer, 2009.

[24] N. Christianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[25] B. Scholkopf, K.-K. Sung, C. J. C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik, "Comparing support vector machines with Gaussian kernels to radial basis function classifiers," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2758–2765, Nov. 1997.

[26] J. Milgram, M. Cheriet, and R. Sabourin, "'One against one' or 'one against all': Which one is better for handwriting recognition with SVMs?" in *Proc. 10th Int. Workshop Frontiers Handwriting Recognit.*, G. Lorette, Ed. La Baule, France: Suvisoft, Oct. 2006. [Online]. Available: https://hal.archives-ouvertes.fr/inria-00103955/document

[27] Y. Ma and G. Guo, *Support Vector Machines Applications*. New York, NY, USA: Springer, 2014.

[28] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *J. Mach. Learn. Res.*, vol. 1, pp. 113–141, Sep. 2001.

DAE JUNG KIM (Graduate Student Member, IEEE) received the B.S. degree in electricity and electronic engineering from Anyang University, Anyang, South Korea, in 2014. He is currently pursuing the Ph.D. degree in electrical engineering with the Systems and Control Laboratory, Hanyang University, Seoul, South Korea. His main research interests include control theory, machine learning, application to autonomous driving, and system integration of intelligent vehicles. He is also a Student Member of the IEEE Intelligent Transportation Systems Society (ITSS), IEEE Control System Society (CSS), Society of Automotive Engineers (SAE), the Korean Society of Automotive Engineers (KSAE), and the Institute of Control, Robotics and Systems (ICROS).

JIN SUNG KIM (Graduate Student Member, IEEE) received the B.S. degree in electronic engineering from Kookmin University, Seoul, South Korea, in 2014, and the M.S. degree in electrical engineering from Hanyang University, Seoul, in 2019, where he is currently pursuing the Ph.D. degree in electrical engineering. His main research interests include control theory, autonomous driving vehicles, optimal control, robust control, and artificial intelligence. He is a member of the IEEE Control System Society (CSS), the Society of Automotive Engineers (SAE), the Korean Society of Automotive Engineers (KSAE), and the Institute of Control, Robotics and Systems (ICROS).

JIN HO YANG (Member, IEEE) received the B.S. degree in electronics and communications engineering from Hanyang University, Ansan, South Korea, in 2017, and the M.S. degree in electrical engineering from Hanyang University, Seoul, South Korea, in 2020, respectively. He is currently a Research Engineer with the Autonomous Vehicle Laboratory, Hyundai Mobis Corporation, Yongin, South Korea. His research interests include control theory, deep learning, autonomous driving, and vehicular active safety systems. He is also a member of the Korean Society of Automotive Engineers (KSAE), the Institute of Control, Robotics and Systems (ICROS), and Korean Institute of Communications and Information Sciences (KICS).

SEOK CHEOL KEE (Member, IEEE) received the B.S. and M.S. degrees in control and instrumentation engineering and the Ph.D. degree in electrical engineering from Seoul National University, Seoul, South Korea, in 1987, 1989, and 2002, respectively. From 1989 to 2007, he worked as a Principal Research Staff with the Samsung Advanced Institute of Technology (SAIT). From 2010 to 2015, he worked as the Head of Electronic Research and Development Center in Mando. In 2015, he joined the faculty of Chungbuk National University, where he is currently an Associate Professor of the Smart Car Research Center. His research interests include computer vision, image processing, sensor fusion, and embedded control system for automotive applications.

CHUNG CHOO CHUNG (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Seoul National University, Seoul, South Korea, and the Ph.D. degree in electrical and computer engineering from the University of Southern California, Los Angeles, CA, USA, in 1981, 1983, and 1993, respectively.

Before his Ph.D., he worked with LG Electronics and IBM Korea. From 1994 to 1997, he was with the Samsung Advanced Institute of Technology, South Korea, where he was selected as a Samsung Group 21C Leader. In 1996, he finished the Samsung Advanced Management Program with The Wharton School, The University of Pennsylvania. In 1997, he joined the faculty of Hanyang University, Seoul. He is also a member of the National Academy of Engineering of Korea (NAEK). He was the 2019 President of the Institute of Control, Robotics and Systems (ICROS), South Korea. He was the Program Co-Chair of ICCAS-SICE 2009, Fukuoka, Japan, the 2015 IEEE Intelligent Vehicles Symposium (IV 2015), COEX, South Korea, and an Organizing Chair of the International Conference on Control, Automation and Systems (ICCAS 2011), KINTEX, South Korea. He was the General Chair of ICCAS 2019, Jeju, South Korea and CDC 2020, Jeju, in 2020. He was an Associate Editor of the *Asian Journal of Control*, from 2000 to 2002; the IEEE TRANSACTIONS ON CONTROL SYSTEM TECHNOLOGIES, from 2013 to 2018; and the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, from 2016 to 2018. He was a (Founding) Editor of the *International Journal of Control, Automation and Systems* (IJCAS), from 2003 to 2005. He served as an associate editor and/or a senior editor for various international conferences, such as the IEEE Conference on Decision and Control (CDC), the American Control Conferences (ACC), the IEEE Intelligent Vehicles Symposium (IV), and the Intelligent Transportation Systems Conference (ITSC). He was a Guest Editor for a Special Issue on "Advanced servo control for emerging data storage systems" (TCST, 2012) and the *IEEE Intelligent Transportation Systems Magazine* (2017). He is currently an Associate Editor of the IFAC *Mechatronics*.