Design of an Airbag Deployment Algorithm Based on Precrash Information

Kwanghyun Cho, *Student Member, IEEE*, Seibum B. Choi, *Member, IEEE*, and Hyeongcheol Lee, *Member, IEEE*

***Abstract*—Airbag systems have become an essential safety device for guaranteeing the physical well-being of drivers and their passengers. Unlike other safety devices, airbags are used as the last resort in a collision, and because they are directly linked to the life of the driver and passengers, the proper functioning of the system is an issue of paramount importance. Hence, to ensure the precision and reliability of airbag operation, it is necessary to design a robust crash algorithm. Currently, several companies are working to achieve an optimal deployment time for airbags in collisions by diversifying the type and location of crash-related sensors. Nevertheless, several problems must still be confronted. For instance, when a vehicle operates off road or when the sensor inside the airbag control unit (ACU) receives a powerful shock, the vehicle's airbags may inadvertently deploy, although no collision has occurred, because a crashlike signal is delivered to the ACU. Alternately, in a collision situation that requires airbag deployment, the crash algorithm may make an erroneous judgment with regard to the collision configuration and miss the time frame for airbag deployment or fail to deploy the airbags altogether. Such problems can be attributed to the following two major causes: 1) Only signals produced through crash tests are used in the design of crash algorithms, and 2) the algorithms themselves only utilize postcrash input from the relevant sensors. To resolve these issues, this paper proposes a precrash algorithm that generates information about the crash scenarios before a collision has occurred. The purpose of the precrash algorithm is to make judgments about the impending collision configuration prior to impact by estimating the behavior of frontal objects and to communicate this information to the crash algorithm to enable correct recognition of the crash scenarios. This paper also proposes a crash algorithm based on crash-related sensors for the verification of and interfacing with the proposed precrash algorithm. The limitations of existing crash algorithms, which deploy airbags using only postcrash signals, were resolved through the development of an integrative crash algorithm that combines the precrash and crash algorithms to reflect precrash information. The developed algorithm was verified by running a wide range of simulations using CarSim, based on data from real crash tests. The results showed that, compared with independently using the crash algorithm, adding precrash information estimation significantly improved the reliability of airbag deployment.**

***Index Terms*—Advanced airbag system, airbag crash algorithm, precrash, sensor fusion, vehicle dynamics.**

Manuscript received August 30, 2010; revised December 20, 2010; accepted February 13, 2011. Date of publication March 14, 2011; date of current version May 16, 2011. The review of this paper was coordinated by Dr. S. Anwar.

K. Cho and S. B. Choi are with the Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea (e-mail: khcho08@kaist.ac.kr; sbchoi@kaist.ac.kr).

H. Lee is with the Division of Electrical and Biomedical Engineering, Hanyang University, Seoul 133-791, Korea (e-mail: hclee@hanyang.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2011.2126614

I. Introduction

AIRBAG systems have become an essential safety device for guaranteeing the physical well-being of drivers and their passengers. Unlike other safety devices, airbags are used as the last resort in a collision, and because they are directly linked to the life of the driver and passengers, the proper functioning of the system is an issue of paramount importance. Hence, to ensure the precision and reliability of airbag operation, it is necessary to design a robust crash algorithm.

Currently, several companies are working to achieve optimal deployment time for airbags in collisions by diversifying the type and location of crash-related sensors. The reality, however, is that numerous limitations remain, to a degree that accidents have been caused by the malfunctioning of airbag systems [1]–[3]. For instance, when a vehicle operates off road or when the sensor inside the airbag control unit (ACU) receives a powerful shock, the vehicle's airbags may inadvertently deploy, although no collision has occurred, because a crashlike signal is delivered to the ACU. Alternately, in a collision situation that requires airbag deployment, the crash algorithm may make an erroneous judgment with regard to the collision configuration and miss the timeframe for airbag deployment or fail to deploy the airbags altogether [4]. Such situations may largely be attributed to the following two defining causes.

First, crash algorithms are designed based solely on the data obtained through crash testing. However, to maintain cost effectiveness in vehicle manufacturing, crash tests are performed only in accordance with a number of standardized scenarios [5]. This condition inevitably limits the number of cases that can be used to design a crash algorithm. Consequently, if a crash scenario that is dissimilar to a crash test scenario occurs, the crash algorithm may not properly recognize the configuration and ultimately deploy the airbags in error. Second, crash algorithms use only postcollision input from crash-related accelerometers. Therefore, if these sensors are broken, rotated, or moved by impact, the resulting error signals are reflected without adjustment. This factor is a key that can cause a crash algorithm to commit errors in evaluating crash scenarios and other situations [6].

To overcome the limitations that result from the exclusive use of postcrash signals, this paper proposes a precrash algorithm that generates information about crash scenarios before a collision takes place. The purpose of the precrash algorithm is to make judgments about the impending collision configuration prior to impact by estimating the behavior of frontal objects and to communicate this information to the crash algorithm to enable correct recognition of the crash scenario. This paper also proposes a crash algorithm to verify the performance of the precrash algorithm as a means of improving the deployment of airbags by the crash algorithm. By interfacing the two algorithms, this paper proposes a conceptually new crash algorithm that incorporates precrash information. The proposed algorithm resolves problems that affect crash algorithms that exclusively rely on postcrash signals for airbag deployment by complexly utilizing precrash and postcrash data, thus allowing the deployment of airbags at the required time. The robustness of the proposed algorithms is verified by applying actual crash test data to a variety of road conditions and crash scenarios simulated in CarSim.

This paper is organized as follows. Section II discusses the precrash algorithm, which is designed to estimate the behavior of frontal objects and make judgments about the impending situation prior to collision. Existing precrash algorithms, which are singly and independently used, are then analyzed to propose an algorithm design that is oriented toward improving the performance of the crash algorithm. Section III deals with the crash algorithm, designed to deploy a vehicle's airbags upon collision. From an analysis of the problems of previously developed crash algorithms, this section proposes a crash algorithm with enhanced effectiveness. In Section IV, the algorithms designed in Sections II and III are combined to produce a new crash algorithm that incorporates precrash information. In Section V, a diverse range of simulations are run to verify the efficacy of the proposed algorithms, and the performance of these algorithms is compared and analyzed according to the various conditions of integration. Finally, Section VI presents the overall conclusions and brings this paper to a close.

II. Precrash Algorithm

In this section, existing precrash algorithms are analyzed, and a new precrash algorithm that can be interfaced with an actual airbag deployment algorithm is examined. The primary purpose of the precrash algorithm is to generate crash-related information about frontal objects so that the crash algorithm may recognize a crash situation before a collision takes place. A precrash algorithm builds information that is most useful to the crash algorithm, thus boosting the crash algorithm's performance. Fig. 1 schematizes the precrash algorithm proposed in this paper [7].

A. Previous Research Analysis

Existing precrash systems have been developed in designing active safety systems for vehicles. For example, the active safety system known as collision mitigation system (CMS) uses a variety of high-technology sensors to assess the possibility of collision with frontal objects and sounds an alert to the driver if the possibility is substantial. Furthermore, if the collision cannot be avoided by the driver, it prevents or mitigates the impending crash by controlling the vehicle's steering or brake system [8]–[11]. Although such active safety systems can provide much information about the precrash situation by employing high-performance sensors, they are virtually unused

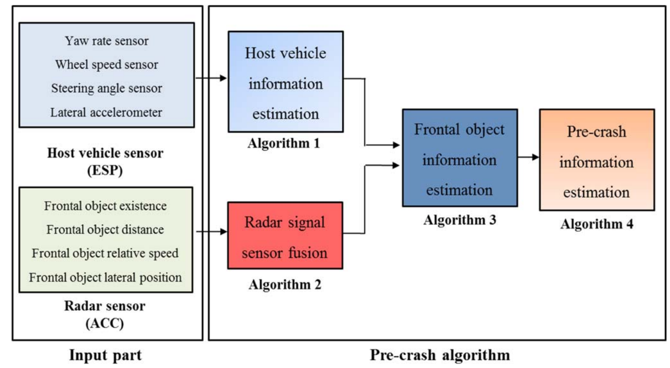

Fig. 1. Precrash algorithm block diagram.

once a crash has occurred. However, if an active safety system is extended to an airbag system, it can be used to provide highly useful information. Moritz introduced a method for applying radar sensors, used individually or in combinations of three or four, to a crash algorithm. His method defines a possible crash zone and lowers the crash algorithm threshold if a vehicle is present within that zone to trigger early functioning of the airbag system. In particular, the start threshold of the crash algorithm is lowered in advance so that the algorithm may detect an imminent crash faster. In addition, the strength of impact is defined prior to collision based on the precrash collision speed so that the airbag deployment threshold may accordingly be lowered [12]. Building on the precrash algorithm designed by Moritz, Bunse introduced improvements to the way crash type and crash severity are estimated. Using radar sensors, the stiffness of frontal objects can be determined by the size of the ACU-X sensor signals during the time that leads up to the estimated moment of impact. His method is distinctive, because it uses only radar sensors and the ACU-X sensor, without recourse to the front impact sensor (FIS) [13].

Such attempts are aimed at accelerating the crash algorithm's recognition of imminent crashes to allow the crash algorithm to make faster judgments about crash scenarios. However, because most crash algorithms are triggered only after a certain threshold has been reached, lowering the starting value can lead to airbag misfire in cases where no collision has occurred or the impact of the collision is low. Furthermore, if crash severity is determined prior to collision based on crash velocity, airbag deployment becomes inconsistent if the stiffness of the frontal object is low despite a high collision speed, or *vice versa*. This condition can have the opposite of the intended effect by resulting in a failure to comply with the required time to fire (RTTF). In addition, the use of information obtained only from simple radar sensors does not allow for the estimation of the relative movement of frontal objects. This condition, in turn, makes it impossible to predict whether a frontal object will disappear from the defined crash zone or what form of crash will occur. To address such problems, this paper estimated frontal objects through the simultaneous use of radar sensors and host vehicle information sensors. In addition, in this paper, a precrash algorithm that can enhance the airbag deployment performance of existing crash algorithms without altering their threshold was developed.

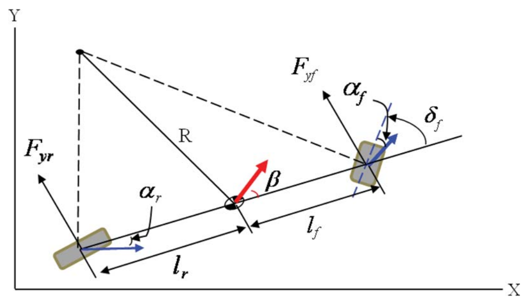

Fig. 2. Bicycle model for a front steering vehicle.

B. Host Vehicle Information Estimation

Host vehicle information estimation consists of estimating the vehicle's longitudinal and lateral velocity by employing sensors used for the vehicle stability control system. The information thus obtained serves as data that represent the movement of the origin in a rectangular coordinate system that was intended to estimate the position of the frontal object. For the algorithm input, yaw rate, wheel speed, steering angle, and longitudinal/lateral acceleration sensors are used.

1) Longitudinal Velocity Estimation: Because an actual longitudinal accelerometer is not installed on a vehicle, the speed of the four wheels is used to estimate the vehicle's longitudinal velocity. In this paper, the estimation of the longitudinal velocity is carried out through signal processing based on undriven wheel speed, according to whether the various vehicle controllers, including the antilock braking system (ABS), traction control system (TCS), dynamic rear proportioning (DRP), and vehicle stability control (VSC), are functioning.

2) Lateral Velocity Estimation: Typically, a vehicle's lateral motion can be explained in terms of the bicycle model. The bicycle model shown in Fig. 2 has linearly been simplified upon the premise that the vehicle's longitudinal velocity is constant and that tire-cornering stiffness is equal in both the left and right directions. Although limited by its high sensitivity to vehicle parameters and road surface conditions, this model has a small uncomplicated computational load while offering relatively superior performance. When estimating the side-slip angle, it uses the comparatively more reliable yaw rate sensor rather than directly resorting to a lateral accelerometer, which makes it robust to sensor influence.

To formulate a dynamics model based on the bicycle model, the following state-space equations can be derived from the force and moment balance equations:

$$\dot{x} = Ax + Bu, \qquad A \in R^{2\times 2}, \quad B \in R^{2\times 1} \tag{1}$$

$$y = Cx, \qquad C \in R^{1\times 2} \tag{2}$$

where

$$A = \begin{bmatrix} -\frac{2(C_f+C_r)}{mv_x} & \frac{2(C_r l_r - C_f l_f)}{m{v_x}^2} - 1 \\ \frac{2(C_r l_r - C_f l_f)}{I_z} & -\frac{2\left(C_f l_f^2 + C_r l_r^2\right)}{I_z v_x} \end{bmatrix}, \quad B = \begin{bmatrix} \frac{2C_f}{mv_x} \\ \frac{2C_f l_f}{I_z} \end{bmatrix}$$

$$C = [0 \quad 1], \quad x = \begin{bmatrix} \beta \\ r \end{bmatrix}, \quad y = r.$$

In this paper, lateral velocity is estimated using a bicycle model that has been modified from the aforementioned bicycle model. Considering that most vehicles are equipped with not only yaw rate sensors but lateral accelerometers as well, this model is derived using the mechanical relation between existing state variables and measured lateral acceleration. This approach is intended to account for the large change in vehicle movement that occurs upon collision in an ordinary driving configuration. Expressing the side-slip angle and yaw rate relative to lateral acceleration yields the following equation:

$$a_y = \dot{v}_y + rv_x \approx v_x(\dot{\beta} + r). \tag{3}$$

If (3) is expressed using the matrix elements from (1) and (2), the following output state equation is derived, which is modified from (2):

$$\begin{bmatrix} r \\ a_y \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{2(C_f+C_r)}{m} & -\frac{2(C_f l_f - C_r l_r)}{mv_x} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2C_f}{m} \end{bmatrix} \delta_f. \tag{4}$$

Therefore, a_y can be expressed as follows:

$$a_y = -\frac{2(C_f + C_r)}{m}\beta - \frac{2(C_f l_f - C_r l_r)}{mv_x} r + \frac{2C_f}{m}\delta_f. \tag{5}$$

To estimate the lateral velocity, the Luenberger observer is designed using the aforementioned model equation as

$$\dot{\hat{x}} = A\hat{x} + B\delta_f + K(y - \hat{y}) \tag{6}$$

where

$$\hat{x} = \begin{bmatrix} \hat{\beta} \\ \hat{r} \end{bmatrix}, \quad y = \begin{bmatrix} r \\ a_{ym} \end{bmatrix}, \quad \hat{y} = \begin{bmatrix} \hat{r} \\ \hat{a}_y \end{bmatrix}.$$

$\hat{\beta}$ is the estimated side-slip angle, $\hat{r}$ is the estimated yaw rate, r is the measured yaw rate, δ_f is the front steering wheel angle, a_{ym} is the measured lateral acceleration, and K is the observer gain.

Using (1) and (6), the following error dynamics can be derived:

$$\dot{\tilde{\beta}} = A_{11}(1 - K_{12}v_x)\tilde{\beta} + (A_{12}(1 - K_{12}v_x) - K_{11} - K_{12}v_x)\,\tilde{r} \tag{7}$$

where $\tilde{\beta} = \beta - \hat{\beta}$, and $\tilde{r} = r - \hat{r}$.

Simplifying the formula by defining the observer gain as $K_{12} = 1/v_x$ yields the following equation:

$$\dot{\tilde{\beta}} = -(1 + K_{11})\tilde{r}. \tag{8}$$

Finally, applying the negative pole-placement method to the remaining observer gain results in the following observer gain, with p signifying the tuning parameter [14]:

$$K = \begin{bmatrix} \frac{I_z(l_f C_f - l_r C_r)p^2}{2C_f C_r (l_f + l_r)^2} - 1 & \frac{1}{v_{xpm}} \\ -2p & \frac{m\left(l_f^2 C_f + l_r^2 C_r\right)}{I_z(l_f C_f - l_r C_r)} \end{bmatrix}. \tag{9}$$

TABLE I
RADAR SENSOR'S SPECIFICATION

Contents	Range	Unit	Tolerance
ObjStatus	0 - 1	-	-
ObjDist	1-200	0.1m	0.25m
ObjLatPos	-20-20	0.1m	0.1m
ObjRelSpd	-255-88	1m/s	0.5km/h
ObjAng	-30-30	1deg	-
Sampling time	0.02sec		
Phase lag	0.066sec		

C. Radar Modeling and Signal Processing

Radar sensors that are attached to the vehicle's ACC system are used to obtain summary information about frontal objects. In actual crash testing, the use of radar sensors incurs cost constraints and environmental limitations. Therefore, in this paper, radar sensors have been substituted with a simulation using CarSim, and the modeling of the radar sensors is carried out according to the specifications and CAN data for real radar sensors in current use.

Information from ACC radars has various resolutions and phase lags. In this paper, a sensor fusion observer is developed for the processing of radar signals. The proposed sensor fusion observer improves signal reliability by utilizing the mechanical relationship between two sets of data with disparate methods of measurement. In terms of radar information, the sensor fusion observer uses distance information based on the signal echo time and speed information obtained through the Doppler effect. For the purposes of this paper, reliability is verified based on the resolution and tolerance of the two signals, as provided in Table I. The results show that the speed signal has higher reliability than the distance signal. The developed sensor fusion observer is given as follows:

$$\dot{\hat{x}} = Lv_m + K(x_m - \hat{x}) \tag{10}$$

where x_m, v_m are the measured distance and velocity, respectively, $\hat{x}$ is the estimated distance, and L, K are the tuning parameters.

Using radar information, the time to crash with a frontal object t_{TTC} and the angle of impact $\angle a$ are acquired as follows:

$$t_{TTC} = dL/v_m \tag{11}$$

$$\angle a = \arccos(dy/dL) \tag{12}$$

where dL is the distance from a frontal object, v_m is the relative speed, and dy is the lateral offset.

D. Frontal-Object Information Estimation

The purpose of frontal-object information estimation is to predict the frontal object's relative position, heading angle, and position at the crash moment based on the host vehicle using the estimated information about the host vehicle and the radar-acquired information about the frontal object. Fig. 3 is a block diagram that illustrates the process of obtaining the frontal object's heading angle and its longitudinal and lateral positions at the crash moment, where $v_x(k)$ and $v_y(k)$ are the longitudinal and lateral velocity at the current time step, respectively, $\triangle x_h(k)$, $\triangle y_h(k)$ is the moving distance for the sample time period, $x_t(k)$ and $y_t(k)$ are the longitudinal and lateral positions at the current time step, $x_t(k-1)$ and $y_t(k-1)$ are the longitudinal and lateral positions at the previous time step, and $x_t(t_{TTC})$ and $y_t(t_{TTC})$ are the estimated positions at crash time.

To calculate the frontal object's position and heading angle relative to the host vehicle, a rectangular coordinate system is proposed. This system takes as its origin the installation location of the host vehicle's ACC radar sensor, as shown in Fig. 4. In addition, to express the movement of the rectangular coordinate system centered at the host vehicle, the latter's estimated longitudinal/lateral velocity is integrated at each sampling time, and its movement in either direction is calculated as shown in (13) and (14), shown below, where T_r signifies the CAN data sampling time of the radar

$$\triangle x_h(k) = x_h(k) - x_h(k-1) = v_x T_r \tag{13}$$

$$\triangle y_h(k) = y_h(k) - y_h(k-1) = v_y T_r. \tag{14}$$

Equations (13) and (14) represent the migration of the coordinate system itself. The frontal object's distance and polar form information, acquired by signal-processing radar information, is converted into rectangular coordinate system information that signifies the longitudinal and lateral positions and is reflected in the host-vehicle-centric coordinate system as

$$(x_t, y_t) = (\sqrt{dL^2 - dy^2}, dy). \tag{15}$$

The frontal-object position from the previous sample time is then converted into its position at the current sample time as

$$x_t(k-1)' = x_t(k-1) - \triangle x_h(k) \tag{16}$$

$$y_t(k-1)' = y_t(k-1) - \triangle y_h(k). \tag{17}$$

Using the change in each direction between the frontal-object positions at the previous and current sample times, the frontal object's heading angle is calculated as follows:

$$\angle a_{heading} = \arctan\left(\frac{y_t(k) - (y_t(k-1) - \triangle y_h(k))}{x_t(k) - (x_t(k-1) - \triangle x_h(k))}\right). \tag{18}$$

Next, the estimation method for the frontal object's longitudinal/lateral position at the crash moment is discussed. The following equations represent the coordinate system's longitudinal and lateral relative positions at the previous and current sample times at the crash moment, as well as the distance traveled by the host vehicle until the crash moment:

$$x_t(k-1)|_{t_{TTC}} = x_t(k-1) - [\triangle x_h(k) + \triangle x_h(t_{TTC})] \tag{19}$$

$$y_t(k-1)|_{t_{TTC}} = y_t(k-1) - [\triangle y_h(k) + \triangle y_h(t_{TTC})] \tag{20}$$

$$x_t(k)|_{t_{TTC}} = x_t(k-1) - \triangle x_h(t_{TTC}) \tag{21}$$

$$y_t(k)|_{t_{TTC}} = y_t(k-1) - \triangle y_h(t_{TTC}) \tag{22}$$

$$\triangle x_h(t_{TTC}) = \frac{\triangle x_h(k)}{T_r} \cdot t_{TTC} \tag{23}$$

$$\triangle y_h(t_{TTC}) = \frac{\triangle y_h(k)}{T_r} \cdot t_{TTC}. \tag{24}$$

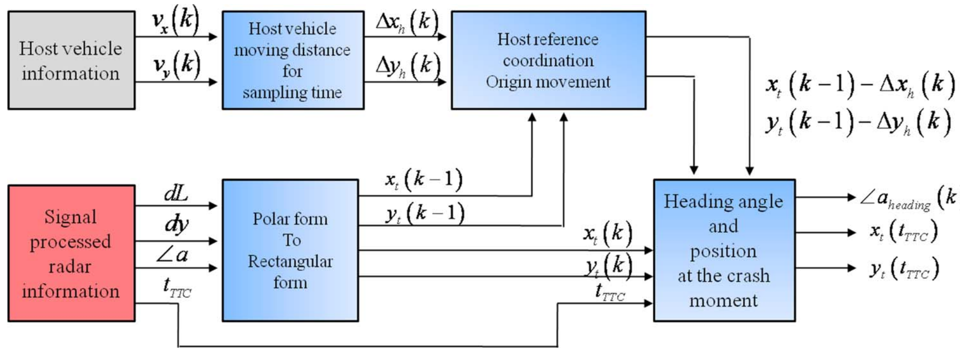

Fig. 3. Frontal object estimation block diagram.

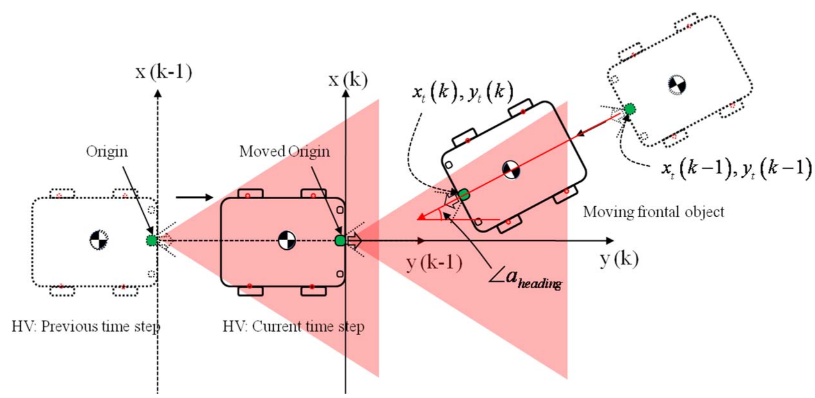

Fig. 4. Frontal object movement.

Using the aforementioned equations, the frontal-object position at the crash moment is derived as follows:

$$x_t(t_{TTC}) = -\frac{x_t(k-1) - x_t(k) - \triangle x_h(k)}{T_r} \cdot t_{TTC} + [x_t(k-1) - (\triangle x_h(k) + \triangle x_h(k-1))] \tag{25}$$

$$y_t(t_{TTC}) = -\frac{y_t(k-1) - y_t(k) - \triangle y_h(k)}{T_r} \cdot t_{TTC} + [y_t(k-1) - (\triangle y_h(k) + \triangle y_h(k-1))]. \tag{26}$$

E. Precrash Information Estimation

The main purpose of precrash information estimation is to generate information that can maximize the performance of the crash algorithm for optimal airbag deployment. Hence, it is necessary to consider how this information may be interlinked with the crash algorithm. Crash algorithms are divided into the following two kinds: 1) algorithms that deploy airbags regardless of the crash type [15] and 2) algorithms that deploy airbags after identifying a specific crash type [16]–[18]. Between the two types, the latter type is more capable of robust and enhanced performance, provided that the reliability of the crash-type information can be guaranteed.

In this paper, precrash information is adopted by considering crash algorithms that discriminate among various crash types. Crash-type-discriminating algorithms can take advantage of radar sensors that provide information about a frontal object's orientation, time to crash, and relative velocity. The information used in this paper is given as follows.

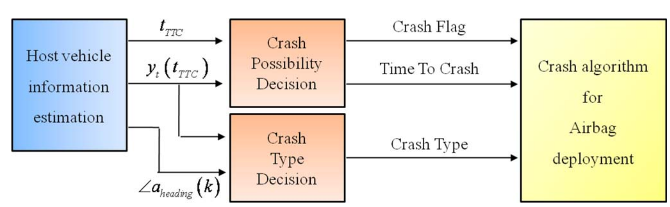

Fig. 5. Precrash information estimation.

- The *crash possibility* is predicted by estimating the position of the frontal object.
- The *time to crash* is calculated using the relative speed and distance information provided by the radar.
- The *crash type* is discriminated based on the frontal object's heading angle and position.

Fig. 5 is a diagram that represents the algorithm for generating precrash information.

1) Crash Probability: Because the precrash algorithm is directly linked to the crash algorithm, conditions for the precrash assessment of crash possibility must be appropriately established. This condition requires considering the actions that a driver may take to avoid collision. In this paper, the following methods were utilized to consider the time that it takes for brake pressure to be communicated when a driver who has recognized an impending crash decelerates and applies the brake pedal,

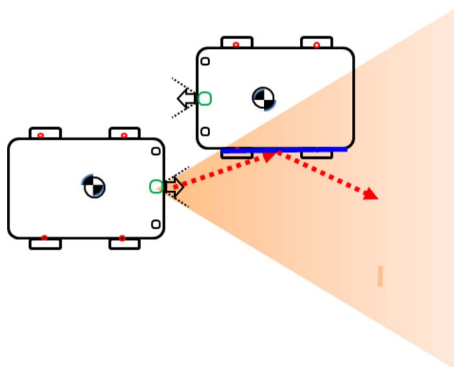

Fig. 6. Radar signal reflection.

as well as the time that it takes for the vehicle to respond when a driver applies step steering to avoid a collision.

- The brake manipulation response time is calculated as

$$\begin{aligned} &\text{Gas pedal off} + \text{brake pedal on} + \text{brake pressure on} \\ &\quad = (0.3 \sim 0.4) + (0.2 \sim 0.3) + (0.1 \sim 0.2) \\ &\quad = 0.6 \sim 0.9 \text{ s.} \end{aligned}$$

- The steering manipulation response time is calculated as

$$\begin{aligned} &\text{Initial vehicle response time after step steering} \\ &\quad + \text{vehicle dynamics completion time} \\ &\quad = (0.1 \sim 0.4) + (0.2 \sim 0.3) = 0.3 \sim 0.7 \text{ s.} \end{aligned}$$

The brake manipulation response time is a statistical figure for the period measured from the time that a driver recognizes a crash to the time that brake pressure is communicated. The steering manipulation response time represents actual data for the vehicle yaw rate response produced for step steering similar to crash situations. When the possible actions for avoiding collision are factored in, it is shown that steering manipulation requires relatively less crash avoidance time than brake manipulation. Based on this result, it can be deduced that, when the available time is less than the 0.1 s needed for the initial vehicle response following steering input, there is no input that can achieve crash avoidance. Therefore, if the time to collision with the frontal object is less than 0.1 s and the estimated lateral position of the frontal object at crash falls within the width of the host vehicle, collision becomes inevitable. Based on this condition, the first definition for the possible crash zone is formulated as follows:

$$t_{TTC} \le 0.1 \text{ s} \;\&\; |y_t(t_{TTC})| \le 0.5 \cdot W \tag{27}$$

where W is the width of the host vehicle.

Meanwhile, even when the frontal object exists within the possible crash zone, no collision is judged to occur if the radar signal is completely reflected off the side surface of the vehicle, as illustrated in Fig. 6 To correct this case, a secondary condition that addresses the crash possibility of a frontal object that has disappeared off the radar within a certain timeframe is needed. Considering that the maximum time required to avoid collision through brake manipulation is 0.9 s, the secondary condition for the possible crash zone is formulated as follows:

$$ObjStatus = 0 \;\&\; t_{TTC} \le 1 \text{ s} \;\&\; |y_t(t_{TTC})| \le 0.5 \cdot W. \tag{28}$$

If either (27) or (28) is satisfied, the crash flag that signifies crash possibility is set to 1, and the step preparatory to delivering precrash information to the crash algorithm is reached.

2) Crash-Type Discrimination: The crash type is discriminated using information about the frontal object's heading angle and predicted lateral position at the crash moment, obtained through frontal-object information estimation. The heading angle is used to differentiate between frontal or offset crashes, which occur at an angle near 0°, and oblique crashes, which occur at a wider angle. The estimated lateral position at the crash moment is used to differentiate between frontal and offset crashes, which are difficult to distinguish based on the heading angle alone. Table II shows the conditions for discriminating crash types based on the frontal object's heading angle and estimated lateral position at the crash moment. Each threshold is defined by simulating a variety of crash scenarios.

TABLE II
CRASH-TYPE DISCRIMINATION CONDITIONS

	$\lvert y_t(t_{TTC})\rvert$		
$\lvert\angle a_{heading}\rvert$	$0 \sim 0.25 \cdot w$	$0.25 \cdot w \sim 0.45 \cdot w$	$0.45 \cdot w \sim 0.5 \cdot w$
$0 \sim 10$ deg	frontal	offset	oblique
$10 \sim 90$ deg	oblique	oblique	oblique

III. CRASH ALGORITHM

This section focuses on the crash algorithm, which deploys the vehicle's airbags using signals received by the accelerometer upon collision. A crash algorithm of the class that discriminates among crash types is proposed so that the enhancement of airbag deployment performance enabled by the precrash algorithm proposed in Section II may be maximized.

A. Previous Research Analysis

A diverse array of crash algorithms, which have been developed by a correspondingly large number of companies, are currently available. These crash algorithms can largely be divided into two categories based on their characteristics. The first class deploys the airbags once the speed and acceleration from collision impact have reached a certain threshold, without discriminating among crash types [15]. The second class discriminates among crash types and applies different thresholds for airbag deployment based on the identified crash type [16]–[18].

The former type of crash algorithm is more robust to crash scenarios that fall outside crash test conditions, because the algorithm itself does not discriminate among crash types. However, it is less able to satisfy the various time-to-fire requirements, which are differently applied according to individual crash test conditions. On the other hand, the latter type of crash algorithms is more robust when it comes to deploying airbags according to each crash type, because it specifically identifies the crash scenario at hand and accordingly sets the threshold for

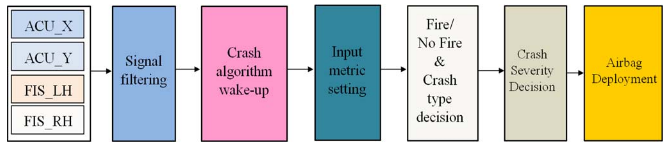

Fig. 7. Crash algorithm block diagram.

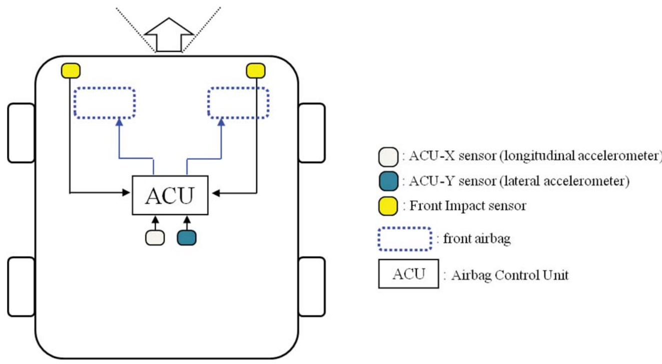

Fig. 8. Sensor used for the crash algorithm.

airbag deployment. However, it can commit substantial error in deployment time if an actual crash scenario falls outside the purview of crash test configurations or lies on the boundary between different sets of crash test configurations. If the reliability of crash-type discrimination can be guaranteed, the latter type of crash algorithms can offer more robust enhanced performance. Therefore, this paper proposes a crash algorithm that can differentiate among crash types so that the precrash algorithm proposed in Section II may be used to effectively enhance the airbag deployment performance of the crash algorithm. The proposed crash algorithm may run in conjunction with the precrash algorithm or as a stand-alone algorithm. Its performance is verified using real vehicle crash test data in a variety of crash scenarios.

B. Crash Algorithm

The operation of the crash algorithm follows the sequence schematized in Fig. 7. The acceleration signals received from the sensors are processed by the algorithm's hardware filter, and if the starting conditions for the algorithm are met, the crash algorithm is triggered. The processed acceleration signals are then calculated in the manner predetermined by the algorithm and used to judge the crash type, crash severity, and whether to deploy the airbag. If the results of these judgments satisfy the various thresholds assigned based on the crash test data, the vehicle's airbags are deployed. The sensors used in this process are the longitudinal and lateral accelerometers inside the ACU, as well as the longitudinal accelerometers installed on each side of the vehicle near the front bumper, as shown in Fig. 8.

The running of the crash algorithm using the aforementioned sensors follows the flowchart in Fig. 9. Each part of the flowchart is explained in detail in the following sections.

1) Crash Algorithm Wake Up: Crash algorithms are not in constant operation. They are triggered when the minimum signal size interpretable as a crash is entered. In this paper, the start and end of the crash algorithm are triggered by the following inequality, which uses the longitudinal accelerometer inside the ACU:

$$a_x \leq -1g \quad \text{or} \quad avg\left(|a_x|\right) \geq 0.4g. \tag{29}$$

Table III shows the average wake-up time by crash type for the proposed crash algorithm, which is measured from the moment that a collision has occurred to the moment that the crash algorithm is triggered. Because the typical average wake-up time for crash algorithms ranges between 4 and 5 ms, it is shown, based on the test results in Table III, where the average is 4.29 ms, that the applied conditions are valid.

2) Metric Setting: The acceleration signals delivered as output from the various sensors after impact can be divided into the following three categories: 1) input variables related to collision force; 2) input variables related to collision energy; and 3) input variables related to the combination of force and energy. In this paper, the following variables are used, because they are deemed to be best suited to express crash characteristics [19]–[23]:

- acceleration: $a(t)$;
- sum of absolute acceleration: $\sum |a(t)|$;
- velocity: $v(t) = \sum a(t)$;
- rate of velocity change: $a(t)|_{4samples} = (v(t) - v(t-4)/4 \cdot Ts)$;
- rate of change of the velocity change rate: $da(t)_{4samples}/dt$;
- acceleration differential: $j(t) \approx (\triangle a(t)/\triangle t)$;
- sum of acceleration signal length: $\sum \sqrt{(da(t)/dt)^2 + 1}$.

The sum of absolute acceleration, velocity, and the rate of velocity change are used as signals for discerning the crash type, whereas velocity and the rate of velocity change are used as signals for airbag deployment. The sum of acceleration length is used to determine whether the airbag will be deployed.

3) Airbag Fire/No-Fire (Deployment) Decision: It is not necessary for airbags to be deployed whenever a collision occurs. In fact, in a low-velocity collision, the deployment of airbags may injure the driver. To prevent such injuries, airbag deployment must occur only when the crash velocity is above a certain level. Therefore, it is vital to use the severity of the crash as a basis for determining whether airbags will be deployed. In the case of frontal crashes, airbags must not deploy if the velocity corresponds to crash test category FRT#1; they must only deploy if the crash speed corresponds to FRT#2 or higher. If velocity or energy signals are used to discriminate between such cases, the proportional relationship between collision speed and signal size allows airbag fire/no fire situations to clearly be distinguished across frontal, offset, and oblique crash categories. However, in the case of Car To Pole (CTP) or Bumper Override (OVRD) crashes, velocity signals are smaller than FRT#1, as shown in Fig. 10, which makes it difficult to base fire/no-fire decisions of velocity input variables.

To make it distinguishable among the aforementioned types of signals, this paper uses the following formula for the sum of acceleration length:

$$\int \sqrt{\left(\frac{da}{dt}\right)^2 + 1\, dt}. \tag{30}$$

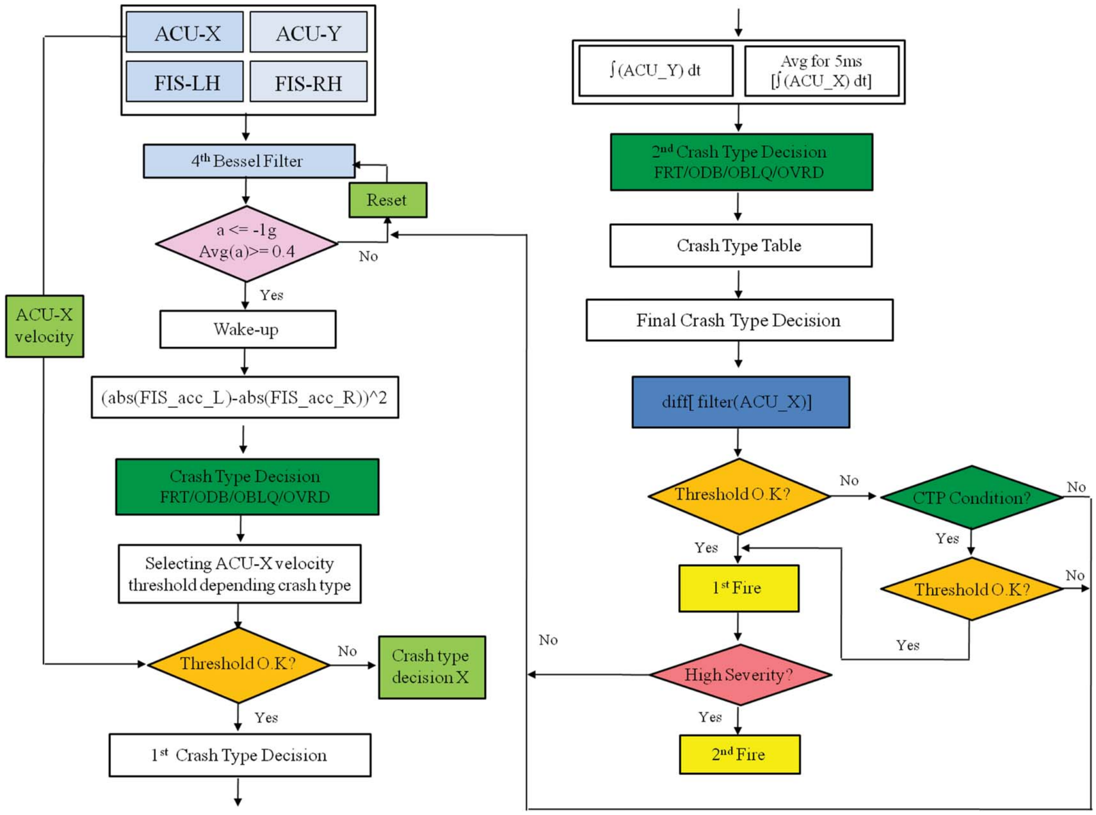

Fig. 9. Crash algorithm flowchart.

TABLE III
AVERAGE WAKE-UP TIME, DEPENDING ON THE CRASH TYPE

Crash Type	Average Wake-Up Time
Front	4.2426 msec
Oblique	3.5083 msec
Offset	4.0278 msec
CarToPole	4.3222 msec
Override	8.2111 msec
Average	4.2931 msse

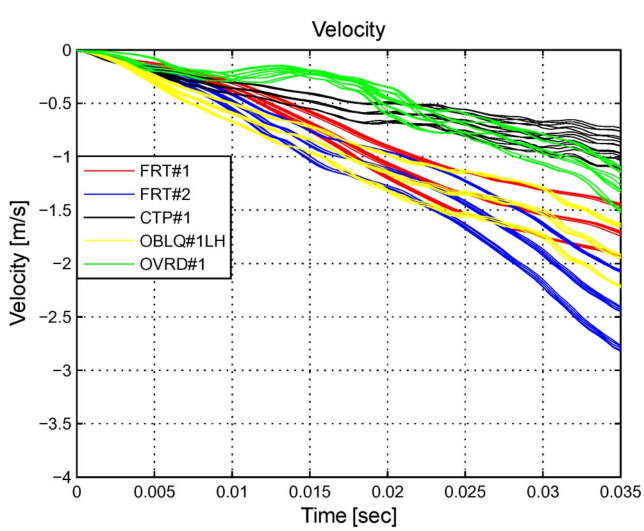

Fig. 10. Fire/No-fire decision using the velocity signal.

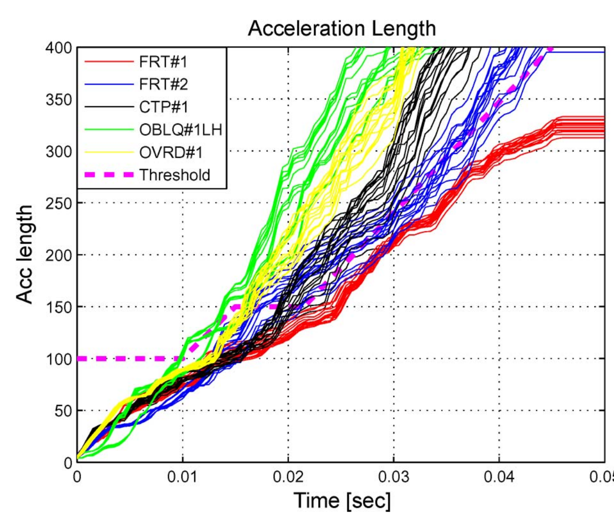

Fig. 11. Fire/No-fire decision using the sum of acceleration signal.

ACU-X acceleration signals are used as input for (30). Fig. 11 demonstrates the validity of the sum of acceleration signal in deciding on the firing and nonfiring of airbags.

4) Crash-Type Decision: Vehicle crash tests are aimed at ensuring the driver's safety by using simulations of actual collisions to determine the optimal timeframe for airbag deployment. To approximate real situations as closely as possible, crash testing covers a wide variety of crash types, including Frontal Crash (FRT), Offset Deformable Crash (ODB), Oblique Crash (OBLQ), CTP, and OVRD. This paper proposes a dual-stage crash-type decision method that differentiates FRT,

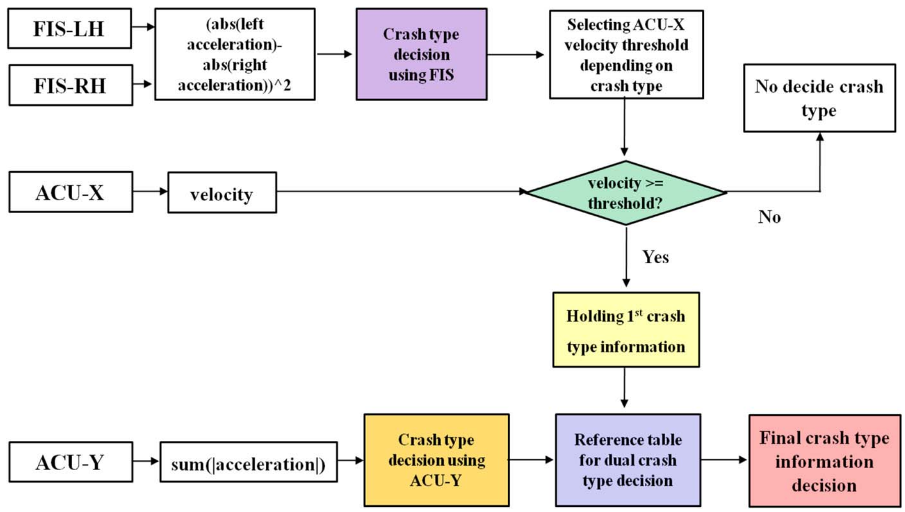

Fig. 12. Dual-stage crash-type decision.

ODB, and OBLQ crashes through the simultaneous use of the ACU-Y sensor and the FIS. To differentiate CTP and OVRD crashes from other types, this paper proposes a separate method based on the ACU-X sensor [24].

ACU-Y sensors generally produce signals with higher reliability, because the impact of collision does not directly reach them. However, they are subject to time delays in identifying the crash type, which raises the possibility of airbag misfiring due to erroneous decision making with regard to the crash scenario. By contrast, FISs offer the advantage of more rapid crash-type identification, but they are also vulnerable to erroneous decision making, because the sensor signal is sensitive to shock. This paper proposes an original crash-type decision method that utilizes the relationship between these two types of sensors, assuming that the FIS's more rapid decision making and the ACU-Y sensor's higher reliability may be exploited in a mutually complementary manner. Fig. 12 shows a block diagram of the new crash-type decision method proposed in this paper.

Fig. 12 details the dual-stage crash-type decision method based on the complementary use of the FIS and the ACU-Y. For the initial crash-type discrimination, the results from the FIS, which produces a crash-type decision at an early point, are used. In this stage, the type of crash is identified using the following formula based on the difference between the left and right FIS signals:

$$\left(\sum |a(t)|_{\mathrm{FIS-LH}} - \sum |a(t)|_{\mathrm{FIS-RH}}\right)^2. \qquad (31)$$

The aforementioned expression utilizes the fact that the left and right signals will differ in size, depending on the direction of the crash. According to the thresholds determined based on the crash test data, FRT, ODB, and OBLQ crashes are differentiated. However, crash-type information acquired using the FIS quickly changes from FRT to ODB to OBLQ as time progresses. Therefore, it is necessary to introduce a measure for arresting the crash-type information at a certain point in time. This point must be the moment at which the crash-type information most suited to the crash scenario at hand is obtained. In the majority of cases, this approach is considerably achieved sooner than the RTTF. Accordingly, as shown in Fig. 13, thresholds are set for the ACU-X sensor's velocity signal by crash type, and the current crash-type information is maintained without further change once the relevant threshold has been exceeded. Fig. 14 illustrates how the first crash-type information is generated and maintained using this process. To make the secondary decision with regard to the crash type, the first crash-type information acquired through the FIS and the additional crash-type information provided by the ACU-Y sensor are compared. Crash-type discrimination based on the ACU-Y sensor is performed using the following formula:

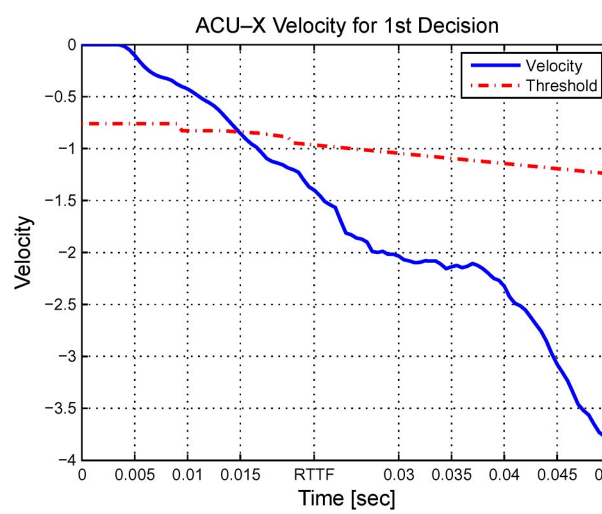

Fig. 13. Velocity signal to fix the first decision.

$$\sum |a(t)|_{\mathrm{ACU-Y}}. \qquad (32)$$

By employing (32), the ACU-Y sensor-based crash-type discrimination method differentiates FRT, ODB, and OBLQ crashes according to the thresholds determined by the crash test data.

The results of running simulations using the crash test data show that, for FRT and ODB crashes, the majority of first

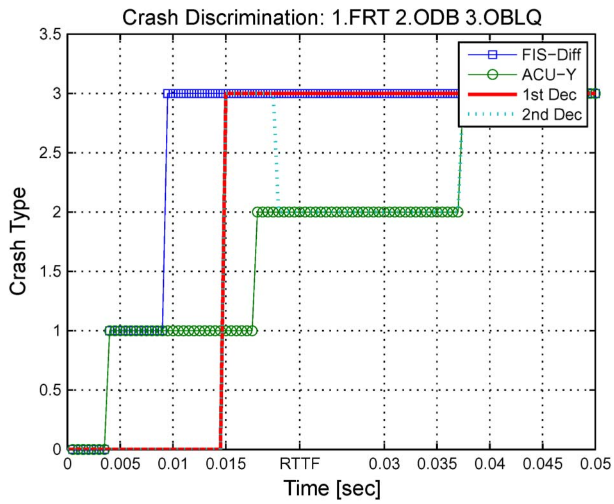

Fig. 14. ODB #2: Crash-type decision results.

TABLE IV
DISCRIMINATION CONDITION FOR THE DUAL CRASH-TYPE DECISION

1st Decision \ 2nd Decision	Frontal	Offset	Oblique
Frontal	Frontal	Offset	Oblique
Offset	Frontal	Offset	Offset/Oblique
Oblique	Frontal	Offset	Oblique

crash-type decisions coincide with the actual crash type. However, when the first crash-type information indicates an OBLQ crash, there are instances when the actual crash is either ODB or OBLQ. To resolve this issue, a correction process using the ACU-Y sensor is introduced. Table IV is a reference table that compares the information obtained from the first and second crash-type decisions.

Figs. 13 and 14 show the results obtained to verify the aforementioned method using real vehicle crash test data for ODB crashes.

When only the FIS is used, the crash type is identified as OBLQ before the RTTF, but when the proposed method is applied, the decision is corrected to ODB at the RTTF. The ACU-X-sensor-based crash-type discrimination is used to identify CTP and OVRD crashes, because these crash types are difficult to distinguish using the FIS or the ACU-Y sensor, which discriminate among crash types through orientation. In CTP crashes, collision impact is delivered only to a localized part of the vehicle body, thus resulting in buckling or bending. Hence, the absolute size of the acceleration signal is smaller than as generated in FRT, ODB, or OBLQ crashes at the same velocity. Based on this characteristic, CTP crashes are identified using jerk signals, obtained by acquiring velocity signals at every fourth acceleration sample and double differentiating the results. In addition, the severity of vehicle body vibration due to localized impact is ascertained using the number of times that the jerk signal crosses zero. Accordingly, the zero-crossing number for each crash type immediately prior to the RTTF is counted, as shown in Table V. The results show that CTP crashes can be differentiated from other crash types by their significantly higher zero-crossing number at the RTTF.

TABLE V
ZERO-CROSSING NUMBER AT THE RTTF FOR EACH CRASH TYPE

Crash	Zero crossing #	Crash	Zero crossing #
FRT#1	8∼12 at 50ms	OBLQ#1LH	8
FRT#2	4∼8	OBLQ#1RH	8
FRT#3	4	OBLQ#2LH	5
FRT#4	2	OBLQ#2RH	6
FRT#5	4	ODB#1	9
FRT#6	2	ODB#2	6
OVRD#1	7∼9	CTP#1	16∼23

TABLE VI
AVERAGE ACCELERATION (IN SQUARE METERS PER SECOND) FOR THE EARLY 5 ms

Crash	Min	Max	Crash	Min	Max
FRT#1	-19.00	-15.00	OBLQ#1LH	-33.50	-24.00
FRT#2	-26.00	-18.50	OBLQ#1RH	-37.10	-27.30
FRT#3	-39.10	-29.60	OBLQ#2LH	-41.00	-36.00
FRT#4	-44.20	-32.50	OBLQ#2RH	-52.00	39.00
FRT#5	-56.50	-42.00	ODB#1	-34.80	-25.30
FRT#6	-73.40	-55.00	ODB#2	-51.60	38.50
OVRD#1	-13.50	-05.00	CTP#1	-28.20	-21.00

Table III shows that OVRD crashes have a substantially slower wake-up time than other crash types, i.e., OVRD values for the initial average acceleration are relatively lower (based on absolute values) than in the case of other crash types. This result is confirmed by the average acceleration for the first 5 ms by crash type, as listed in Table VI.

5) Airbag Deployment Decision: Several airbag deployment algorithms base their thresholds for airbag firing on the velocity, distance, and energy signals conveyed by the ACU-X sensor. The drawback of such signals, which take the form of simple acceleration integration, is that even a small change in the signal can lead to a significant difference in airbag deployment time. This paper uses parameters that are less sensitive to signal change yet are highly expressive of crash characteristics to deploy the airbags. In cases where the vehicle absorbs the impact of the crash, collision velocity can sometimes remain constant or decrease because of the vehicle's internal structure, bumper, or crash box. This condition creates points at which the slope that results from the collision velocity becomes 0 or reaches its maximum value. To identify such points, the points at which the velocity slope for every fourth velocity sample is differentiated, i.e., at which the inflection point for the velocity signal becomes 0, are detected. Using this method makes it easier to detect absorption points than using a simple differentiation of velocity. At the same time, the airbag deployment algorithm is designed with different threshold settings for each crash type. Such characteristics show strong distinguishability in the case of FRT, ODB, and OBLQ crashes. By contrast, in the case of CTP crashes, unlike other crash types, the airbag deployment condition is based on the fact that the acceleration signal suddenly increases just before the RTTF, leading to the creation of points where the acceleration signal takes a positive value. Because this phenomenon does not occur with other crash types, the airbag deployment algorithm in this instance is designed to reflect this distinguishing feature.

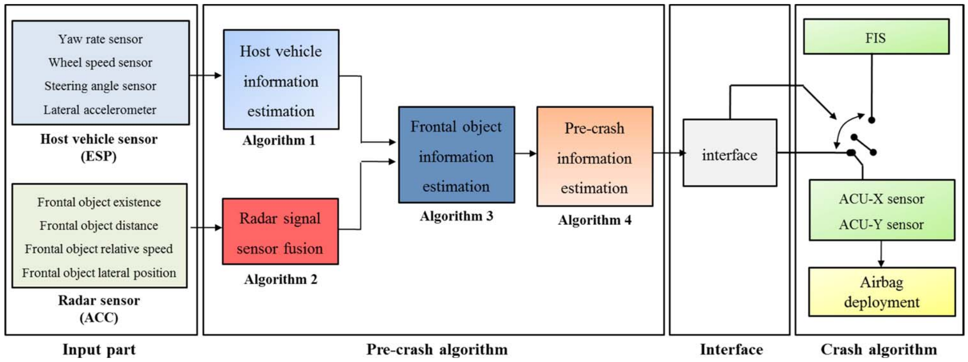

Fig. 15. New crash algorithm based on precrash information.

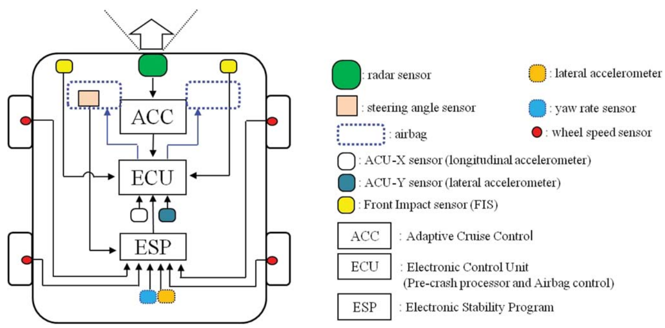

Fig. 16. Hardware configuration.

IV. Crash Algorithm Using Precrash Information

This section proposes a new airbag deployment algorithm (crash algorithm) based on the interface between the precrash algorithm and the crash algorithm proposed in Sections II and III, respectively. Fig. 15 is a block diagram that represents the overall crash algorithm that incorporates precrash information estimation, whereas Fig. 16 shows the hardware configuration for the algorithm [25].

In the crash algorithm based on precrash information, the ECU, which is independently used in the crash and precrash algorithms, is utilized for the interfacing of the two algorithms. For the ECU input, the various sensors installed for the host vehicle's ESP, the radar sensors used for the ACC, and the accelerometers related to the crash algorithm are used.

A. Algorithm Overview

The purpose of the crash algorithm based on precrash information is to generate crash possibility, time to crash, and crash-type information using various radars and host vehicle sensors; communicate the resulting crash-type information about possible crash scenarios; and, thereby, deploy the vehicle's airbags in a manner befitting the configuration at hand. The flow chart for the proposed crash algorithm based on precrash information is shown in Fig. 17.

B. Interface Between Crash and Precrash Information

The interfacing of the crash and precrash algorithms is an important issue. If the precrash algorithm malfunctions, the overall performance may become worse than when only the crash algorithm is used for airbag deployment. In Fig. 18, the interface between the two algorithms is designed.

The AND condition is used for the start and crash flags to prevent meaningless information, e.g., when crash possibility information is generated through radar sensor malfunction, although there is no frontal object present, from being delivered to the crash algorithm and triggering erroneous airbag deployment. It is also intended to allow the crash algorithm to function as a stand-alone process when the radar sensors do not operate, by making independent decisions about a crash situation. For precrash information to be conveyed, the crash flag that signifies the crash possibility must be set to 1. At the same time, the crash algorithm must also detect the crash and have the start flag set to 1. Because precrash information is not continually retained, the duration of information retention is set to a maximum of TTC+0.02 s, considering the time to crash obtained through precrash information estimation and the sampling time for the radar information.

V. Simulation

The robustness of the aforementioned algorithms is verified through simulation in various crash scenarios and actual situations. To recreate a wide variety of crash scenarios, the CarSim software is used, whereas to verify airbag deployment, actual crash test data that cover a diverse range of conditions are used. Individual crash scenarios consisted of crash test conditions employed by automakers and simulated environments that are similar to crash test conditions. The various crash test simulation scenarios are listed in Table VII.

To determine how the integration of the various subordinate algorithms that comprise the precrash algorithm affects the crash algorithm, the precrash algorithm, and the unified algorithm, simulations are performed using the algorithm combination methods detailed in Tables VIII and IX.

Among the methods proposed in Section III, only crash-type discrimination using the X–Y axial accelerometers inside the ACU, and not the FIS, is used for the crash algorithm shown in Table IX. This approach is aimed at analyzing whether crash-type decision making based on the precrash algorithms frontal-object information estimation could replace the FIS, which suffers from low crash-type decision reliability.

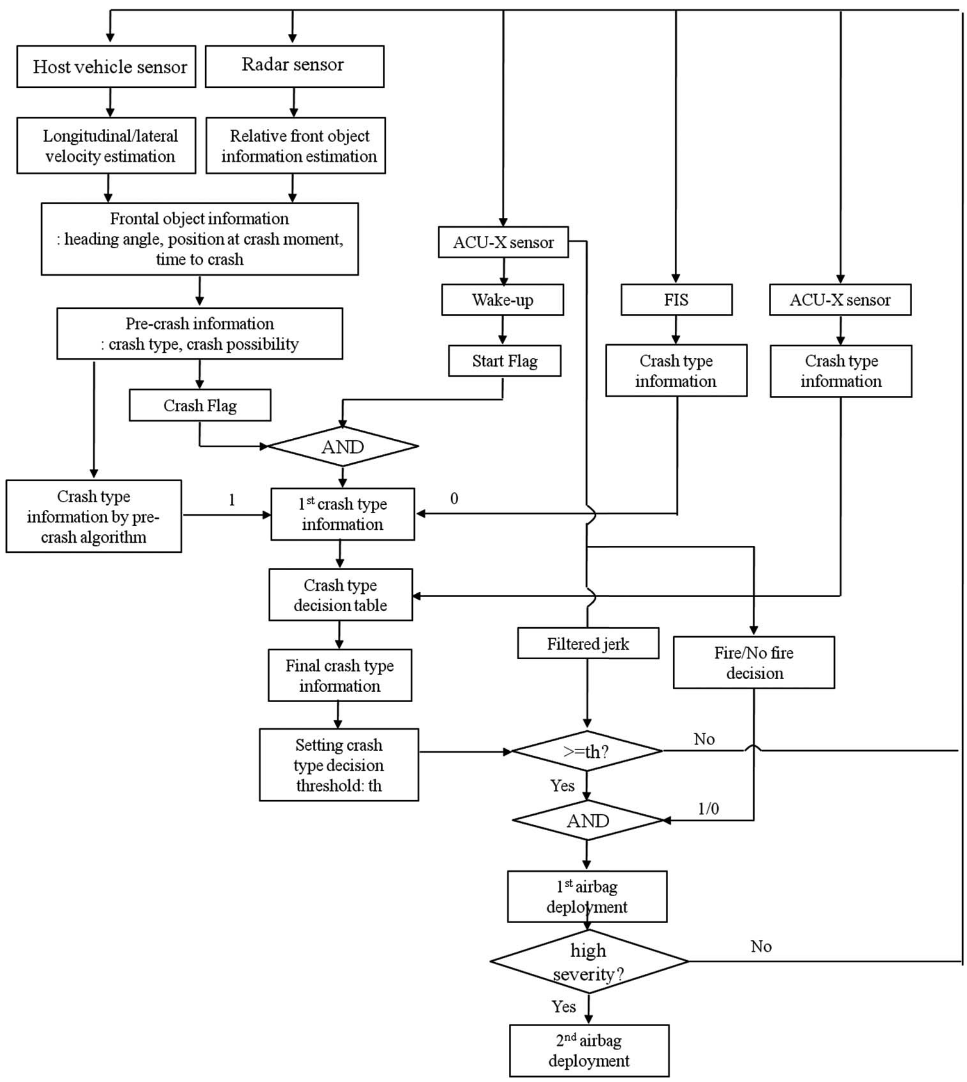

Fig. 17. Flowchart for the crash algorithm based on precrash information.

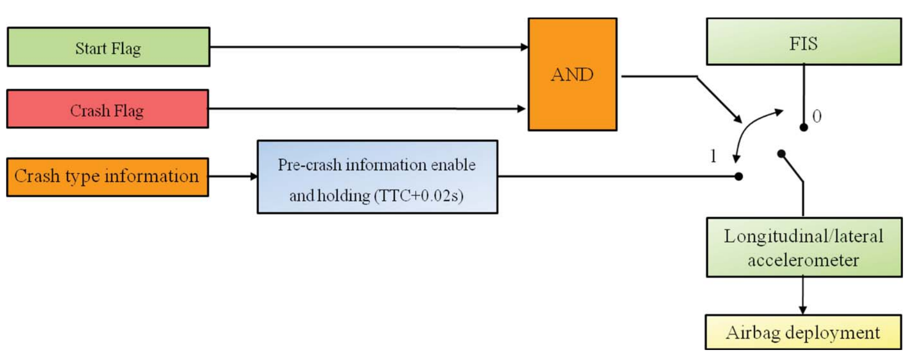

Fig. 18. Interface between the crash and precrash algorithms.

A. Precrash Algorithm

Evaluating the robustness of the precrash algorithm consists of ascertaining how well it can identify crash types in crash environments. In this paper, the scenarios listed in Tables VII and VIII are applied to the precrash algorithm. The crash-type decisions made by the algorithm according to these scenarios are given in Table X.

It is shown Tables X–XII that the accuracy of crash-type discrimination is improved when the host vehicle information and radar information are used in tandem (method 3) than when the latter information is exclusively used (method 1). This case is because using only radars to estimate frontal objects leads to incongruous crash-type decisions in several cases due to insufficient information about the frontal object's

TABLE VII
SIMULATION SCENARIO

Standard Crash Test Mode	
Frontal Crash (FRT)	#1/#2/#3/#4/#5/#6 mph (← Low Velocity High →)
Offset Crash (ODB)	#1/#2 mph (High velocity→)
Oblique Crash (OBLQ LH,RH)	#1/#2 mph (High velocity→)
CarToPole Crash (CTP)	#1 mph
Bumper-Override Crash (OVRD)	#1 mph
Crash Test Emulation Mode	
Offset Crash (ODB)	#2 mph: -10% $\sim$ 100%
Oblique Crash (OBLQ)	#2 mph: 17 $\sim$ 26 deg

TABLE VIII
SIMULATION METHOD: PRECRASH ALGORITHM

	Radar information	Sensor Fusion	Pre-crash Information
Method1	·	·	·
Method2	○	·	·
Method3	○	○	·
Method4	○	·	○
Method5	○	○	○

TABLE IX
SIMULATION METHOD: UNIFIED ALGORITHM

	Radar information	Sensor Fusion	Pre-crash Information	Crash Algorithm
Method1	·	·	·	○
Method2	○	·	·	○
Method3	○	○	·	○
Method4	○	·	○	○
Method5	○	○	○	○

TABLE X
CRASH-TYPE DECISION: STANDARD MODE

	Method1	Method2	Method3	Method4
FRT#1	1	1	1	1
FRT#2	1	1	1	1
FRT#3	1	1	1	1
FRT#4	1	1	1	1
FRT#5	1	1	1	1
FRT#6	1	1	1	1
ODB#1,#2	3	3	2	2
OBLQ#1LH,RH	3	3	3	3
OBLQ#2LH,RH	1	2	3	3
CTP#1,OVRD#1	1	1	1	1

TABLE XI
CRASH-TYPE DECISION: ODB EMULATION MODE

ODB#2	Method1	Method2	Method3	Method4
-10%	0	0	0	0
0%	0	0	2	0
20%	0	0	2	3
30%	3	3	2	2
40%	3	3	1	2
50%	3	3	2	2
60%	3	3	1	2
70%	2	2	1	1
80%	1	2	1	1
90%	1	2	1	1
100%	1	1	1	1

TABLE XII
CRASH-TYPE DECISION: OBLQ EMULATION MODE

OBLQ#2	Method1	Method2	Method3	Method4
#1	1	1	3	3
#2	1	1	3	3
#3	2	3	3	3
#4	3	3	3	3
#5	3	3	3	3
#6	3	3	3	3
#7	0	0	3	3
#8	0	0	3	0

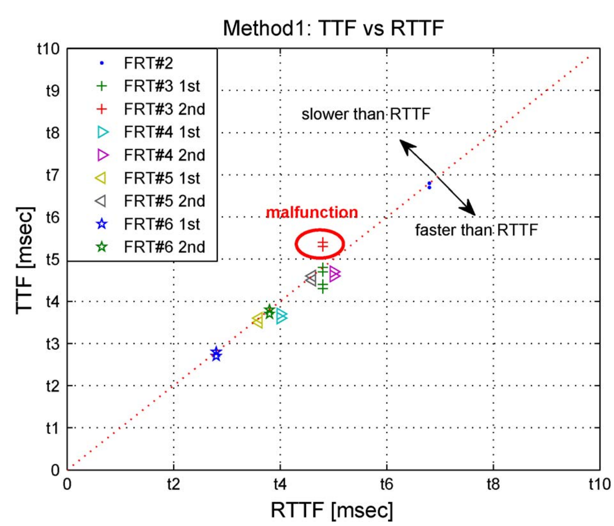

Fig. 19. Method1: FRT crash airbag deployment time.

orientation and crash possibility. With regard to the effect of applying sensor fusion to radar signals, the analysis result shows that the accuracy of crash-type decisions is improved when sensor fusion is used in signal processing, as shown in methods 2 and 4, than when sensor fusion is not used, as shown in methods 1 and 3. This difference in performance stems from the varied effects that problems that result from radar signal phase lag and resolution have on the assessment of crash possibility and estimated crash position. Based on these findings, it can be concluded that frontal-object information received from radar sensors must be integrated with the host vehicle information such that the precrash algorithm achieves improved performance and that applying sensor fusion to radar signals is the optimal method for enhancing the accuracy of crash-type decisions.

B. Crash Algorithm Using Precrash Information

The robustness of the unified algorithm that incorporates precrash information is verified using the method shown in Table IX. The following figures represent airbag deployment performance based on the crash algorithm alone (method 1) and on the unified algorithm that includes the precrash algorithm (method 5).

Figs. 19 and 20 show that using only the sensors inside the ACU results in nondeployment or significantly premature deployment. These problems are caused by erroneous crash-type discrimination and can be resolved by introducing precrash information about the crash type. In Figs. 21 and 22, the aforementioned problems have been resolved through the use of

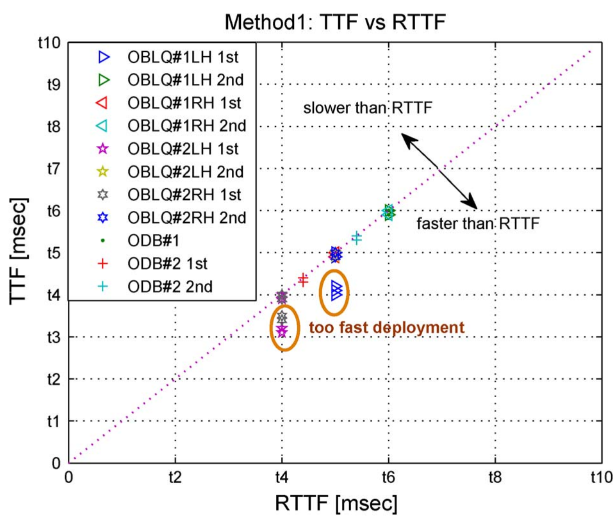

Fig. 20. Method1: OBLQ and ODB crash airbag deployment times.

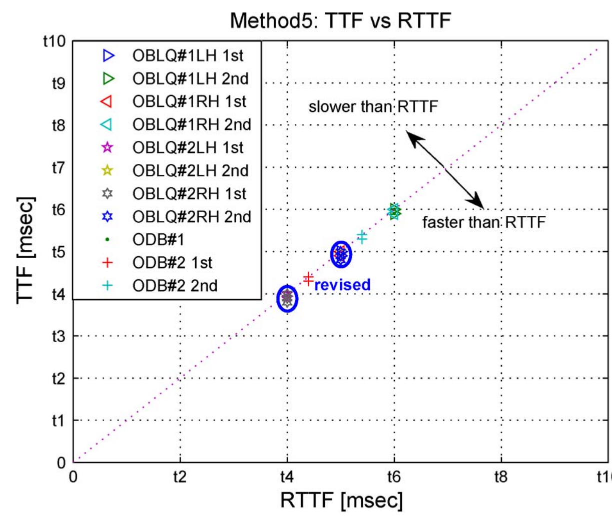

Fig. 22. Method5: OBLQ and ODB crash airbag deployment times.

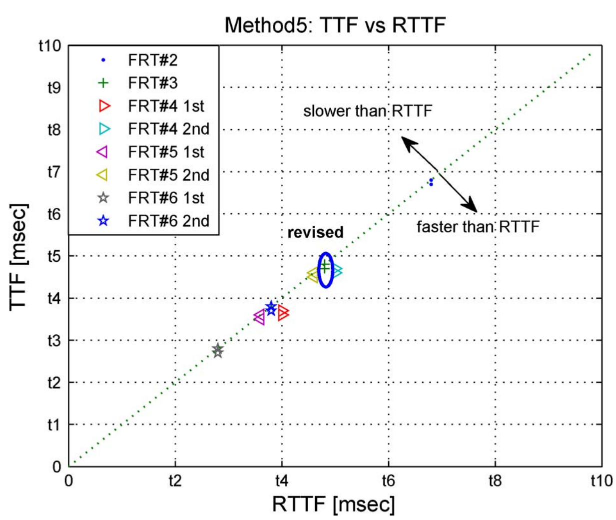

Fig. 21. Method5: FRT crash airbag deployment time.

precrash information, and airbag deployment has been achieved at or near the RTTF in most cases.

VI. Conclusion

This paper has proposed a precrash algorithm to resolve airbag malfunctions caused by the limitations of existing crash algorithms. To obtain input for designing the precrash algorithm, the sensors installed for the vehicles ESP system, including the wheel speed sensor, steering angle sensor, yaw rate sensor, and lateral acceleration sensor, are utilized, in addition to radar sensors. This approach allows for the generation of more reliable precrash information through the addition of estimated information about the frontal-object position and behavior to information based on the host vehicle itself. This paper has also proposed an original method for crash-type decision making by the crash algorithm. By using the FIS and the ACU-Y sensors in combination, problems in crash-type discrimination that result from the independent use of either type of sensor are resolved. Performance enhancement is confirmed by measuring airbag deployment times based on real crash test data. For the final step, a new crash algorithm that interfaces the proposed precrash and crash algorithms is developed. By combining the two algorithms in various ways, this paper has verified that the effectiveness of the crash algorithm significantly improves when precrash information is utilized.

The sensors used for the proposed algorithms consist of combinations of existing sensors used in commercial automotives. Hence, the proposed methods can be implemented at no extra cost. Moreover, because the precrash algorithm is inserted into the proposed crash algorithm as a supplemental function, it can directly be applied to existing crash-type decision algorithms. In conclusion, the methods proposed in this paper offer several benefits in terms of cost and applicability to existing algorithms, and their applicability and performance can further be improved if they are integrated with cameras or ultrasonic sensors.

References

[1] F. J. Stuetzler and K. Century, “Advanced frontal crash sensing with peripheral sensors,” presented at the Society Automotive Engineers Int., Detroit, MI, 2000, 2000-01-2683.
[2] B. Pipkorn, “Crash sensing and algorithm development for frontal airbag systems using CAE methods and mechanical tests,” presented at the Society Automotive Engineers Int., Detroit, MI, 2004, 2004-01-1633.
[3] J. M. Lim, H.-W. Park, S.-H. Hong, B.-J. Kim, and K.-H. Park, “The application of CAE in the development of airbag restraint system performance for a certain vehicle,” *Threshold*, vol. 1501, pp. 1–6, 2007.
[4] T. W. Park, H. Y. Jeong, and S. W. Park, “A crash severity algorithm for all frontal crash modes using compensation factors,” *Proc. ImechE—Part D: J. Automobile Eng.*, vol. 220, no. 5, pp. 531–541, 2006.
[5] T. W. Kim and H. Y. Jeong, “Stochastic analysis of the variation in injury numbers of automobile frontal crash tests,” *Int. J. Autom. Technol.*, vol. 11, no. 4, pp. 481–488, Aug. 2010.
[6] T. Gioutsos and D. Tabar, “Determination for crush zone severit using a ball-in tube and accelerometer sensing system (BASS),” presented at the Society Automotive Engineers Int., Warrendale, PA, 1999, 1999-01-326.
[7] K. H. Cho, “A precrash discrimination system for an airbag crash algorithm,” in *Proc. Amer. Control Conf.*, 2010, pp. 6949–6954.
[8] R. Labayrade, C. Royere, and D. Aubert, “A precrash system based on radar for automotive applications,” in *Proc. IEEE Veh. Symp.*, 2005, pp. 441–446.
[9] M. Skutek, D. T. Linzmeier, N. Appenrodt, and G. Wanielik, “A precrash system based on sensor data fusion of laser scanner and short range radars,” in *Proc. 8th Int. Conf. Inf. Fusion*, 2005, vol. 2, pp. 1–8.
[10] J. Jansson, F. Gustafsson, and J. Ekmark, “Decision making for collision avoidance systems,” in *Proc. Society Autom. Eng. Int.*, 2002, pp. 1–8.

[11] P. M. Knoll, "Predictive safety systems: convenience-collision mitigation-collision avoidance," presented at the Society Automotive Engineers Int., Detroit, MI, 2006, 2006-21-0082.
[12] R. Moritz, "Precrash sensing its functional evolution based on a platform radar sensor," presented at the Society Automotive Engineers Int., Detroit, MI, Oct. 2000, 2000-01-2718.
[13] M. Bunse, A. Kuttenberger, M. Theisen, T. Sohnke, J. Sans Sangorrin, J. Hoetzel, and P. Knoll, "System architecture and algorithm for advanced passive safety by integration of surround sensing information," presented at the Society Automotive Engineers Int., Detroit, MI, 2005, 2005-01-1233.
[14] H. W. Son and S. B. Choi, "A study on design of a robust vehicle side-slip angle observer using an integration of kinematic and bicycle model," in *Proc. KSAE Annu. Conf.*, 2008, pp. 1200–1207.
[15] B. R. Ham, S. G. Hong, and H. Y. Jeong, "A study on crash severity algorithm for advanced airbag system," in *Proc. Autumn Conf. Korean Soc. Autom. Eng.*, 2000, pp. 1041–1046.
[16] S. W. Park and M. C. Jeon, "Crash discrimination algorithm using electronic dual-axis accelerometer," in *Proc. Autumn Conf. Korean Soc. Autom. Eng.*, 2001, pp. 1249–1254.
[17] S. W. Park, J. H. Lee, and M. C. Jeon, "A study on improvement of crash discrimination performance for offset and angular crash events using electronic X–Y 2-axis accelerometer," *Trans. Korean Soc. Autom. Eng.*, vol. 11, no. 1, pp. 128–136, 2003.
[18] S. W. Park, J. H. Lee, and M. C. Jeon, "Crash discrimination algorithm for airbag electronic single-point sensor," *Trans. Korean Soc. Autom. Eng.*, vol. 7, no. 3, pp. 301–309, 1999.
[19] J. Kanamoto, M. Miyamori, K. Jinno, and M. Hirao, "Electronic crash sensing unit for airbag," in *Proc. SAE Paper*, 1994, p. 940 624.
[20] J. L. Allen, "Power-rate crash sensing method for safety device actuation," in *Proc. SAE Paper*, 1992, p. 920 478.
[21] T. Gioutsos, "A predictive-based algorithm for actuation of an airbag," in *Proc. SAE Paper*, 1992, p. 920 479.
[22] S. W. Park and M. C. Jeon, "Crash discrimination algorithm for airbag electronic single-point sensor," in *Proc. KSAE Paper*, 1998, p. 98 134 (In Korean).
[23] H. Y. Jeong and Y. H. Kim, "New algorithm and accelerometer locations for frontal crash discrimination," *Proc. ImechE—Part D: J. Automobile Eng.*, vol. 215, no. 11, pp. 1171–1178, 2001.
[24] K. H. Cho *et al.*, "Design of a crash algorithm using dual discrimination for crash type," in *Proc. Annu. Conf. KSAE*, 2009.
[25] K. H. Cho *et al.*, "Design of an airbag deployment algorithm using a radar sensor," in *Proc. IFAC Symp. Adv. Automot. Control*, 2010.

Kwanghyun Cho (S'11) received the B.S. degree in electrical engineering and computer science from Kyungpook National University, Daegu, Korea, in 2008 and the M.S. degree in mechanical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2010. Since February 2010, he has been working toward the Ph.D. degree with the Department of Mechanical Engineering, KAIST.

His research interests include vehicle dynamics control and active safety.

Seibum B. Choi (M'09) received the B.S. degree in mechanical engineering from Seoul National University, Seoul, Korea, in 1985, the M.S. degree in mechanical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Seoul, in 1987, and the Ph.D. degree in controls from the University of California, Berkeley, in 1993.

From 1993 to 1997, he worked on the development of automated vehicle control systems with the Institute of Transportation Studies, University of California, Berkeley. From 1997 to 2006, he was with TRW, Livonia, MI, where he worked on the development of advanced vehicle braking-control systems. He is currently with the faculty of the Department of Mechanical Engineering, KAIST. His research interests include control systems, driver-assistant systems, camless engines, dual-clutch transmission, and active safety systems of ground vehicles.

Dr. Choi is a member of the Korean Society of Automotive Engineers.

Hyeongcheol Lee (M'10) received the B.S. and M.S. degrees from Seoul National University, Seoul, Korea, in 1988 and 1990, respectively, and the Ph.D. degree from the University of California at Berkeley, in 1997.

Currently, he is a Professor with the Division of Electrical and Biomedical Engineering of Hanyang University, Seoul. His research interests include adaptive and nonlinear control, embedded systems, applications to vehicle controls, and vehicle dynamics.