

S & M 0837

Decision Fusion via Integrated Sensing System for a Smart Airbag Deployment Scheme

Mahammad Abdul Hannan*, Aini Hussain and Salina Abdul Samad

Department of Electrical, Electronic & Systems Engineering,
Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia

(Received April 12, 2010; accepted July 16, 2010)

Key words: sensing system, airbag deployment, decision, intelligent vehicle

In this paper, we address the development of an integrated sensing system for the airbag deployment decision in an intelligent vehicle with focus on passenger and driver airbags in vehicles. The innovation provides a cost-effective system for deploying an airbag whilst maintaining precise, reliable and effective operation; in addition, it can be easily retrofitted into any vehicle with built-in airbag control. A number of sensing systems such as weight, vision and vehicle crash have been developed. The decision of several sensing systems are fused together to provide an exact airbag deployment decision. A LabWindow/CVI in the C interface program is developed for prototype implementation. The sensing systems are developed and integrated into a prototype platform. The performance of the prototype system is evaluated through several test runs. The results prove that the airbag deployment decision is unique, robust and intelligent for vehicle application.

1. Introduction

The term “intelligent vehicle” does not literally mean that intelligence resides inside the vehicle. The term “intelligence” implies the presence of an active essential part of the vehicle that contributes to safety, security and driving comfort. An intelligent vehicle operates to ensure the safety and comfort level of the occupant in the vehicle.⁽¹⁾ However, because of the high expectations for control and safety, an efficient sensing system must be developed to determine the airbag deployment decision in an intelligent vehicle.⁽²⁾ This increased concern for the deployment decision has resulted in the design of an accurate integrated sensing system that involves occupant-weight- and position-sensing technology, detection and classification of an impending crash or unsafe driving conditions, and driver warnings to improve the driver’s ability to control the vehicle and prevent an accident.^(3,4)

Many individual researchers have focused on an airbag system that involves occupant detection, classification and position, vehicle crash detection, crash severity analysis

*Corresponding author: e-mail: hannan@eng.ukm.my

and the airbag deployment decision. A vehicle occupant position detector and an airbag control system are described, in which the controller between the airbag deployment mechanism and the sensors calculates the centre of gravity of the seat occupant, the weight of the occupant and seat inclination angle.⁽⁵⁾ Another method to control passenger side airbag deployment has been developed using a seat cushion weight sensor to determine the mass of the seat occupant to classify the occupant size.⁽⁶⁾ A smart airbag system, including a sensor mechanism that controls deployment to protect an occupant in a crash, has been developed to control the flow of gas into or out of the airbag to optimize the injury protection capability of the airbag. The system also accepts inputs from the occupant position, velocity and weight sensors and/or anticipatory crash sensors, when such are available, which may affect the deployment control signal.^(7,8) However, airbag control deployment for occupant protection is a difficult task. Despite the success of some of these systems, occupant detection and human/nonhuman object detection and classification still pose a number of challenges for real-time implementation and operation as they are costly to apply and require complex systems.^(9,10)

Vehicle crash detection is very helpful for achieving preventative safety, preventing accidents and collisions and minimizing human injury when an accident occurs.⁽¹¹⁾ However, in the past, vehicle crash detection has seldom been discussed or theoretically analyzed in the field of conventional engineering.^(12,13) Therefore, the National Highway Traffic Safety Administration (NHTSA) and other organizations concerned about safety ruled that vehicle crash detection and analysis are mandatory.^(14,15)

In this study, we integrated several sensing systems to deploy an airbag using occupant weight, vision and a vehicle crash-sensing system. The objective of this development is to integrate the sensing system to analyze the roles of occupant detection, classification and position, vehicle crash detection and crash severity analysis in airbag deployment. The experimental results provide a cost-effective, precise and reliable alternative to the current systems, in addition to being easily retrofitted into vehicles, which suggests a more robust airbag deployment decision.

2. Integrated Sensing Systems

The integrated sensing systems for deploying an airbag combine both hardware and software components that identify the execution environment of the system. The development of the system is a complex process and involves many factors, such as the interconnections of the hardware component to support the execution of the software component. The integrated sensing system can be defined from four points of view: the sensing systems, decision algorithm, hardware platform and system interface. A functional diagram of the proposed method and system for deploying an airbag is depicted in Fig. 1. The sensing system addresses the issues of detecting human and nonhuman objects, occupant weight class and position, and vehicle crash occurrence with a webcam, an occupant sensor and a crash sensor, respectively. The decision algorithm fuses the individual decisions through logic combination based on algorithmic rules to provide an executable decision. The hardware platform identifies a set of hardware objects associated with the processors. The system interface provides a high level of

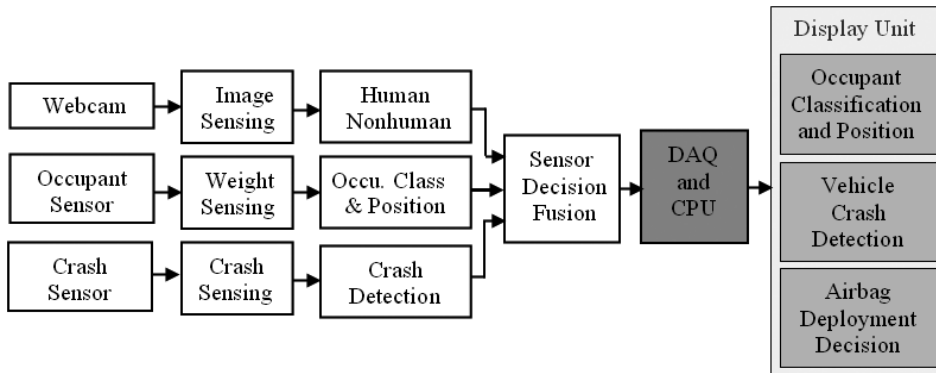


Fig. 1. Functional diagram of integrated sensing system for an airbag deployment decision.

interface between software and hardware objects. The proposed system gathers sensing data through a set of individual sensors. The collected sensing data are eventually processed in the central processing unit (CPU) and fused through logic combination to make the airbag deployment decision and address individual safety issues. The proposed sensing system provides an executable decision from three sensing systems: a weight-sensing system, a vision-sensing system and a crash-sensing system. Details of the individual sensing systems are given below.

2.1 Weight-sensing system

The computational structure of the weight-sensing system involves two calculations, namely, weight force and centroidal distance, for occupant weight classification and position detection, respectively. In the weight-sensing system, an interface algorithm is developed using the LabWindow/CVI program.⁽¹⁶⁾ The program determines whether the seat is empty according to the measurements of four weight sensors. For classification, the weight measurement data are used with logic combination. For example, when an adult occupant is on the seat, the adult logic is true and child and nonhuman object logics are false; thus, the dynamic output is classified as adult and applied to the airbag deployment decision.

Next, the occupant position is determined by the centroidal calculation of weight to check whether the occupant's upper body is in a good position. The centroid distance from the centre of the seat in the y -direction, F_y , is the sum of these forces times the distance divided by the total force. The horizontal position F_x can be calculated similarly. The centroidal distances F_x and F_y , as shown in Fig. 2, are defined as follows:

$$F_x = x \frac{(-F_1 + F_2 - F_3 + F_4)}{(F_1 + F_2 + F_3 + F_4)}, \quad (1)$$

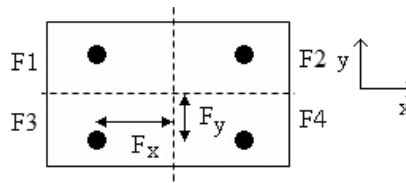


Fig. 2. Centroidal distance for occupant position calculation.

$$Fy = y \frac{(F1+F2-F3-F4)}{(F1+F2+F3+F4)}, \quad (2)$$

where $F1$, $F2$, $F3$, and $F4$ are the weight forces of the four sensors, and x and y are the distances from the center to the sensor in the x - and y -directions, respectively. These calculations of F_x and F_y provide the appropriate position of the occupant with respect to the origin.

2.2 Vision-sensing system

The vision-sensing system structure for detection and classification of an occupant, nonhuman object and nonobject is shown in Fig. 3. The nonobject image defines the images that contain neither a human face nor a specific object. In this study, a webcam is used as a vision sensor to detect either a human face or a nonhuman object for the airbag deployment decision. The webcam is mounted on the dashboard such that it can capture the maximum possible view of the seat and its occupant. The detection of both classes of object in one frame is more challenging than individual detection. To meet these challenges, our vision-sensing system employs neural network classifier architectures by combining the fast neural network (FNN) and the classical neural network (CNN) for detecting and classifying a human face and a nonhuman object. The FNN is used to extract any positive detection, including false detection from the test image by correlating the input image and hidden units. The output of the FNN is then fed to the CNN to verify the region that has been detected.

A bootstrap technique is used to add nonobject images to the database to improve the FNN performance during the training period and automatically clip false detections and to insert them into the current training set. Postprocessing strategies are applied to convert the normalized outputs back into the same units that were used for the original targets using 2D multidetection and 3D multidetection, and elimination of overlapping detection. There is some assumption that the FNN may introduce some false detection due to variation in lighting conditions, which may change its overall appearance. To solve this problem, light intensity is adjusted automatically using histogram equalization or a lighting correction function. However, the histogram equalization and lighting correction cannot be integrated into the Fourier Transform framework because they belong to different processing systems. We go one step further using the CNN as an object verification procedure to reduce the number of false detections and to check for human facial structures and nonhuman objects. This proposed combination network

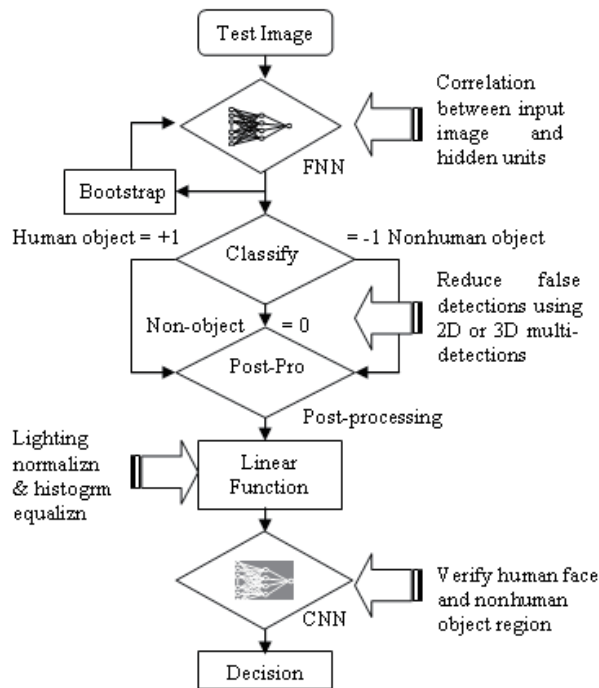


Fig. 3. Neural network structure for occupancy detection.

is robust in terms of detection accuracy and computational efficiency compared with a single network, which is unable to fully eliminate the false detection problem.

2.3 Crash-sensing system

Development of the crash-sensing system is based on mass (m), gravity (g), spring constant (k), and damping constant (b) of the crash-generating device to determine displacement [$x(t)$], velocity [$v(t)$], and acceleration [$a(t)$]. Displacement, acceleration and velocity are used to determine the vehicle crash and its severity.^(17,18) The change in velocity [$\Delta v(t)$] is an essential parameter for crash detection. The change in velocity is obtained by integrating the acceleration signal. Because the integral over the noise component is approximately zero, the parameter $\Delta v(t)$ is essential for the crash detection algorithm and is written as:

$$\Delta v(t) = \int a(t) dt + \int n(t) dt = -A\omega^2 \int \cos(\omega t + \delta) dt. \quad (3)$$

In the crash-sensing system, an interface program is developed, as shown in Fig. 4, that utilizes an accelerometer to acquire data through AX10410A DAQ using the

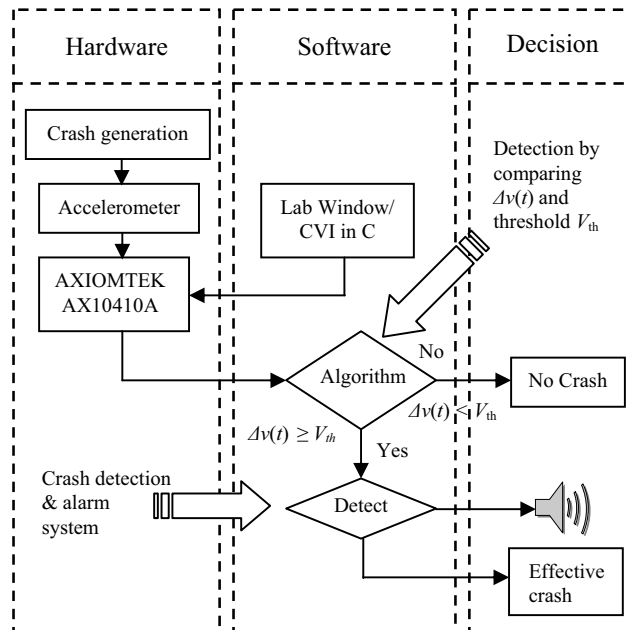


Fig. 4. Crash sensing system interface program flow diagram.

LabWindow/CVI program. The program determines the change in the velocity $\Delta v(t)$ and compares it with the threshold value V_{th} to determine whether a crash has occurred. The threshold value can be easily determined from the lowest speed of an effective crash (i.e., 22.54 km/h) as defined by the NHTSA.⁽¹⁴⁾ To detect a crash, if the change in velocity is greater than or equal to the threshold value (i.e., $\Delta v(t) \geq V_{th}$), then a crash detection decision is made and displayed on the monitor, and an alarm sounds. However, if $\Delta v(t)$ is less than V_{th} , then the decision is that no crash has occurred.⁽¹⁷⁾

These decisions determined from the weight-, vision- and crash-sensing systems are fused on the basis of a decision fusion algorithm. The fused decision is used as input for the airbag deployment decision for occupant safety and comfort level measures.

3. Decision Algorithm

The integrated sensing system for the airbag deployment decision provides a framework for categorizing the various logic decisions to correlate input from various sensing systems and create a statistically based decision algorithm. Each sensing system decision is then combined in an IF/THEN logic fusion engine that makes the final decision to disable or enable the airbag. The decision algorithm is implemented using the Lab Windows/CVI program.

The weight-sensing system classifies the occupant as an adult, child or nonhuman object or as an empty seat, and it involves a decision algorithm based on weight. Once occupancy is detected, the occupant's weight (w) is sampled and then checked as to whether $w \geq m$, $m > w \geq n$ or $0 < w < n$, where m and n are the minimum weight of an adult and child, respectively. The following rules are used in the decision algorithm:

- i) If $w \geq m$, then the program classifies the occupant as an adult.
- ii) If $w \geq n$ but $w < m$, then the program classifies the occupant as a child.
- iii) If $w < n$ and $w > 0$, then the program classifies the occupant as a nonhuman object.
- iv) If none of the above conditions are satisfied, then the program reports that the seat is empty.

The vision-sensing system detects the occupant's face as human or a nonhuman object, as well as the absence of any object. The output of the system is a neural network matrix of values of +1, -1, or 0. Positive, negative and zero training images denote human and nonhuman objects and the absence of an object, respectively. The following rules are used in the decision algorithm:

- v) If the neural network matrix yields 1, the program classifies the occupant as human.
- vi) If the neural network matrix yields -1, the program classifies the occupant as a nonhuman object.
- vii) If the neural network matrix yields 0, the program reports the absence of an object; which means the seat is empty.

The weight- and vision-sensing decisions are then fused to determine the precise occupant class: human as adult or child, nonhuman object or no object, i.e., an empty seat. Once occupancy is determined, the occupant's centroidal position is calculated and identified as either good or out-of-position. Listed below are the decision rules for occupant position:

- viii) If the occupant is in a good position, the program dictates that the airbag be deployed during a vehicle crash.
- ix) If the occupant is out of position, the program dictates that the airbag is not deployed, even if a crash occurs.

The crash-sensing system is responsible for detecting a vehicle crash and its severity. In crash-sensing systems, the change in the velocity $\Delta v(t)$ is typically compared with a threshold value v_{th} to determine whether a vehicle crash has occurred. The following rules are used for the crash detection algorithm of the present system:

- x) If $\Delta v(t) \geq v_{th}$, then the program reports that a vehicle crash has occurred.
- xi) If $\Delta v(t) < v_{th}$, then the output sent to the program is that no vehicle crash has occurred.

For weight rules (iii) and (iv), vision rules (vi) and (vii), position rule (ix) and crash detection rule (xi), the airbag will not be deployed. However, any violation of occupancy conditions (i), (ii), and (v), position rule (viii) or crash detection rule (x) also dictates that the airbag will not be deployed under normal circumstances, even though the occupant has been classified or is positioned properly.

4. Hardware Objects and System Interface

The system implementation is developed through physical interconnections between hardware objects and an interface program. The entire developed system structure consists of the following hardware objects: a load cell weight sensor, a Logitech webcam and Crossbow accelerometer crash sensor, a data acquisition board for analogue to digital conversion, a CPU card, a touch screen to display the result and an ATX switch mode power supply. Wheatstone bridge configuration load cell weight sensors are placed inside the vehicle seat for occupant detection, classification and position. Logitech webcam is used to detect whether a human or nonhuman object is on the seat. Crossbow MEMS accelerometers are used for vehicle crash detection and severity analysis. The AX10410A high-speed data acquisition board is used to interface directly with the featured-pack CPU SBC84600 through the PC/104 bus. The CPU, SBC84600 is incorporated with a thin-film transistor (TFT) colour LCD monitor to display the execution behaviour of the specified target event and real-time target of the prototype system. Multilevel output ATX is used to provide regulated power to hardware objects.

Data are acquired through interface connections between the sensors, DAQ board and the CPU using the interface program. The interface program between the software and hardware objects is developed on the basis of Lab Window/CVI in the C programming language named UKM.dll. The low-level driver called "c:\cvinterface\UKM.dll" is written as a Win32 DLL file. In this DLL file, the function *Func1* processes the analogue signal received by CH0 to CH6 of the A/D converter of the DAQ board from the weight sensors and crash sensors. The function *WeightSense* provides a decision based on the weight of the occupant. The function *ImageProcess* classifies the object as a human, a nonhuman, or a nonobject. The results of this function are fused with the logic combination of the weight sensor to detect whether the occupant is an adult, a child, or a nonhuman object, or whether the vehicle seat is empty. The function *CrashSense* is used to determine whether or not a crash has occurred. The position detection function, *PositionDetection*, calculates the centroidal distance of the *x*- and *y*-axes to provide the occupant position. Finally, the function *ABagDecision* provides the airbag deployment decision upon fusing the logic combination of the occupant classification, position and vehicle crash detection decisions.

5. Experimental Results and Discussion

The experimental results, both image and signal processing, are discussed in terms of occupant detection, classification and position, vehicle crash detection and crash severity analysis towards airbag deployment decision. Usually, the real-time prototype operation varies up to 1 min. However, in our prototype hardware, the execution vectors of the entire system are derived from the experimental measurements within 50 ms. Details of the experimental results and analysis are given below.

In the interface system algorithm, an adult, a child, a nonhuman object and an empty seat are identified as 1, 0.5, 0.25, and 0, respectively, on the basis of weight. Next, the categorization task is performed by introducing the conditions of with an adult, a child, a

nonhuman object, or without an occupant. The results are shown in Fig. 5, which depicts the chronological sequence of testing. Therefore, it can be concluded that the developed prototype system is robust for occupant classification.

Figure 6 shows the centroidal position of the y -centroid versus x -centroid of the vehicle front passenger seat with a size of $50 \times 50 \text{ cm}^2$, which indicates various positions of the occupant: standard, good, and unsafe positions are indicated with solid, dash dotted and dash line markings, respectively. Figure 6(a) shows that the seated occupant is in a good position with his/her back properly aligned to the back of the seat. Figure 6(b) shows that the occupant is in an unsafe position (i.e., the occupant is aligned very much to the right of the seat). Similarly, Fig. 6(c) indicates that the occupant is in an unsafe position and in very close proximity to the airbag. In both Figs. 6(b) and 6(c), the proposed safety device provides a nondeployment decision for the airbag unit. Figure 6(d) also indicates that the occupant is in the extreme left of the vehicle seat, which signifies an unsafe position, and therefore, a nondeployment decision is issued. Figure 6 also shows that the occupant is around $25 \times 25 \text{ cm}^2$ and aligned to the back of the seat, which is the true position of the occupant. On the basis of results, it is reasonable to conclude that the occupant position monitoring of the embedded system is robust and effective.

A sample from the test image database is shown in Fig. 7 and demonstrates the detection capability of the proposed human face and nonhuman object detection scheme. It detects a human face in the car environment under various lighting conditions and backgrounds. It also demonstrates the results of the system application to the detection of a human face with various contrasts and backgrounds in groups or singularly. Similarly, three nonhuman objects in the car seat are also detected from many samples of the test dataset. In the proposed vision-sensing system, the detection rate is about 97.63% with very few false alarms, and the computation speed is 2.3 s, which is much better than that reported previously.⁽¹⁹⁾

In image processing, the vision-sensing system uses two data sets of images in the experiments to test the performance of detection of human faces and nonhuman objects, which are distinct from the training sets. The first set consists of 253 test images, which have a wide variety of complex backgrounds in various environments and scale changes for the object of interest along with some occlusions and variations in lighting. Twenty-

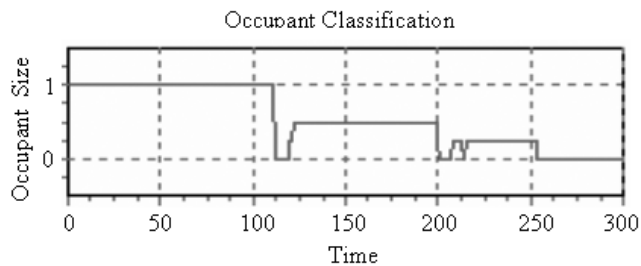


Fig. 5. Occupant classification based on weight sensing.

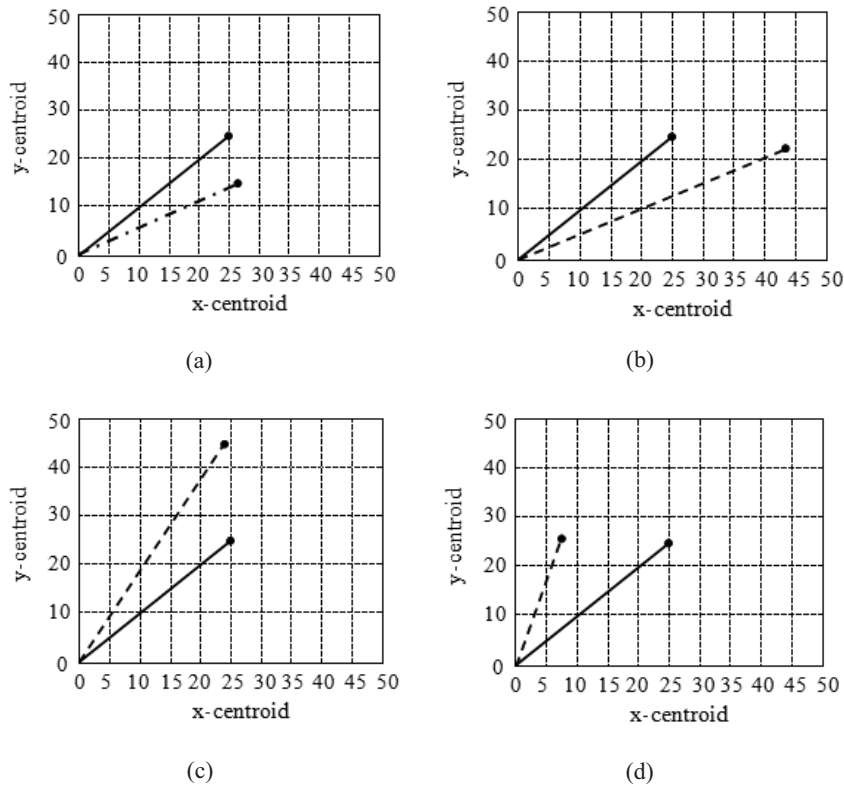


Fig. 6. Occupant centroidal position calculation.

five human face images of interest are included for a total of 253 test images. The second data set contains 112 test images that were collected from seven nonhuman objects of interest. The systems undergo the bootstrapping cycle, which results in 4,500 to 9,500 zero samples, to evaluate the performance of the true detection of the test images and the rate of false detection from the images of natural scenes that do not contain human faces or nonhuman objects.

Table 1 shows the results of human face detection using various methods on test set 1 and compares the results with those of systems in terms of the number of detected faces, missed faces, false detections and computation time. The success rate of the proposed method is 97.6%, with 6 false alarms. It should be noted that the number of false alarms is smaller than those generated by the methods proposed by Ben-Yacoub *et al.*⁽²⁰⁾ and Fasel *et al.*⁽¹⁹⁾ This may show that the combination of two networks can successfully separate human faces from nonobject examples. The improved performance reported by Rowley *et al.*⁽²¹⁾ is likely due to the size of the training data. We used 7,344 human face images and 8,000 nonobject examples, whereas Rowley *et al.*⁽²¹⁾ trained using 16,000 face images and 9,000 nonface images. However, the technique is less efficient than the proposed technique in terms of false detection and response time. On the other hand,

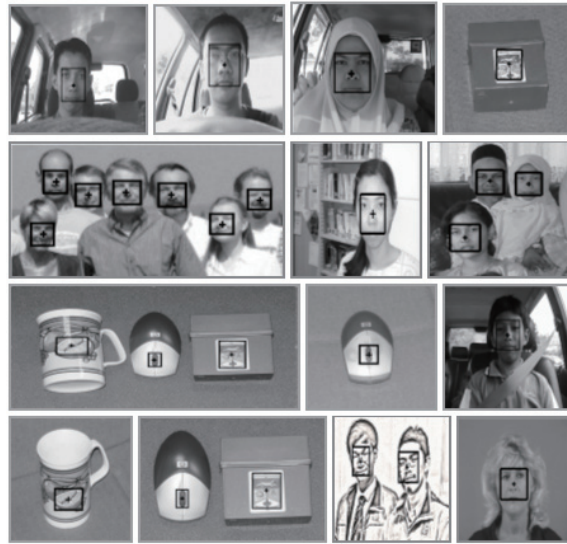


Fig. 7. Human face and nonhuman object detection.

Table 1
Detection rate of set 1 for different methods.

Method	Human detection (%)	Miss human detection (%)	No. of false detection	Process time
FNN+CNN	97.63%	2.37%	6	2.3 s
Rowley <i>et al.</i>	97.86%	2.14%	13	0.013 M
Yacoub <i>et al.</i>	84.31%	15.69%	347	0.7 s
Fasel <i>et al.</i>	96.8%	3.2%	278	3.1 s

Ben-Yacoub *et al.*⁽²⁰⁾ showed a very fast processing but more false alarms.

Similarly, Table 2 shows the summarized results for nonhuman objects from test set 2 compared with those from other systems. We found that the nonhuman object detection rate is 96.42%, which means 108 out of 112 nonhuman objects are detected. The false detection rate is 3.58%, which is lower than that reported by Agarwal *et al.* (2004) and that of other methods.^(23,24) However, the average process time is faster than that of other methods providing additional calculation on CNN except for that obtained by Viola and Jones (2001). The process time obtained by Viola and Jones (2001) is faster owing to the size of training data set, feature resolution and test images. On the basis of the results shown in Tables 1 and 2, we conclude that the human face and nonhuman object detection system has acceptable tradeoffs between the number of false detections and detection rate. Once the image processing part is completed, the *ImageProcess* function outputs 1 for a human and 0 for a nonhuman object. This 1 or 0 is fused with the output of the weight sensor situated inside the vehicle seat to provide an accurate occupant

Table 2
Detection rates of set 2 for different methods.

Method	Nonhuman object detection (%)	Miss nonhuman object detection (%)	No. of false detection	Process time
FNN+CNN	96.42%	3.58%	4	2.9 s
Agarwal <i>et al.</i>	94%	6%	30	3.6 s
Mahmud & Hebert	82%	18%	187	4.0 s
Viola & Jones	95%	5%	71	0.7 s

detection and classification for the integrated intelligent safety system. To illustrate the performance, some sample results obtained from the prototype system are included for various cases, such as an occupied or empty seat. If the seat is occupied, the occupant is classified as an adult, a child or a nonhuman object. Usually, a human, and therefore his/her weight in the seat, takes various positions on the seat. However, a nonhuman object, such as a grocery bag, is static and its position does not vary. It also demonstrates that the position of the occupant can be determined to take safety issues into account in airbag application or to measure the comfort level.

Next, we implemented experiments that study a frontal static crash using a crash generating device and the interface program. The experimental results of the crash reaction force applied to generate the crash are shown in Fig. 8. We tried to obtain the reaction force during repeated crash conduction between 51 to 80 s. It is seen that the repeated crash gains a huge force between approximately 1,000 – 5,800 N/m. Moreover, the velocity immediately before the crash exceeds 22.54 km/h. The reaction force depends on the crash velocity of the system. As velocity increases, the reaction force also increases, which in turn increases the crash severity. This is a situation that puts the occupant at a higher risk.

The experimental results of the developed system for occupant detection, classification, and position, vehicle crash detection, and crash severity analysis contribute to providing the airbag deployment decision. The decisions of the addressed safety issues are fused using a logic combination. Several cases of the event were simulated, and the decision results of the airbag deployment system are shown in Table 3, where ND, AAD, CAD, NOB and NOH represent no deployment, adult airbag deployed, child airbag deployed, nonobject and nonhuman object, respectively. Information on the occupant class, position, and vehicle crash status is used to make the initial decision before a final decision is made as to whether or not to deploy the airbag. The final decision is made by fusing individual logic decisions on occupant weight class, position and vehicle crash detection. Therefore, the fused decision leads to the optimal and reliable performance of the developed airbag deployment system. However, individual decisions only detect a specific safety feature.

6. System Verification and Validation

When developing a prototype system, it is important to determine if the system meets certain specifications and if its outputs are correct. Therefore, extreme care should be

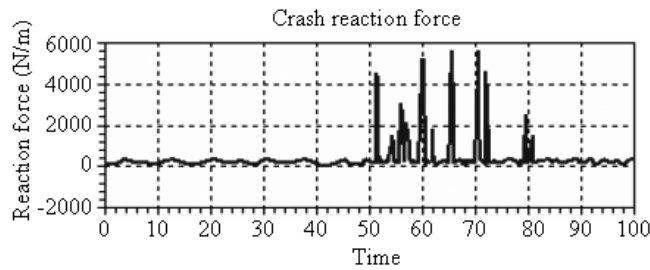


Fig. 8. Vehicle crash reaction force.

Table 3
Airbag deployment decision for several cases of events.

Case	Image detected	Class detected	Position detected	Crash detected	Deployment decision
A	Face	Adult	Good	None	ND
B	Face	Adult	Good	Yes	AAD
C	Face	Adult	Bad	Yes	ND
D	Face	Child	Good	Yes	CAD
E	Face	Child	Bad	Yes	ND
F	NOB	Empty	NA	Yes	ND
G	NOB	Empty	NA	Yes	ND
H	NOB	Object	NA	Yes	ND

taken in the development of the prototype systems to ensure that a sufficient time is spent on verification and validation to prove that a system is correct.

We have verified the implemented prototype intelligent safety system through dynamic and static testing. Dynamic testing involves the execution of the proposed safety system, in which all functions that are defined in the program for the system requirement are identified and tested; furthermore, information from the internal structure of the system to devise tests to check the operation of the individual safety features is used. Functional and structural testing investigates a particular characteristic of the integrated system. In static testing, we analyze the consistency and measurement of the program property, in which correct syntax, correct parameter matching between procedures, correct typing, requirements and specification translation are ensured, and properties such as error likelihood, understandability, and structure are measured. We adopted formal validation techniques: fault injection and dependability analysis. In addition, we used mathematical and logical techniques to express, investigate, and analyze the specification, design, documentation and behaviour of both hardware and software. We intentionally injected faults into the physical hardware and errors into the software program to observe the system operation under faulty conditions. However, we found that the implemented prototype vehicle intelligent system worked well under

fault conditions. We performed dependability analysis to identify hazards and their root causes to develop possible countermeasures and for risk analysis of hazards and their probability of occurring.

7. Conclusions

This study is original, in which the integration of various sensing systems and a rule-based decision algorithm into a platform that provides an airbag deployment decision in an intelligent vehicle is developed. A LabWindow/CVI in the C interface is used for real-time prototype implementation. Weight, vision and crash-sensing systems are integrated by developing the rule-based algorithm into the hardware platform and system interface program. The prototyped intelligent system has led to a successful fabrication of a real-time working device, and its performance has been evaluated through several test runs. Application of the prototype intelligent system has resulted in a successful real-time working device, and it has been proven that the airbag deployment decision is unique, maintains a high precision, is reliable, effective, and robust, and is easily retrofitted into any vehicle with a built-in airbag control. The contributions of the proposed prototype provide management, performance characterization, problem determination, real-time data monitoring and a safety warning for vehicles.

Acknowledgement

The authors would like to thank the Malaysian Ministry of Science, Technology and Innovation (MOSTI) for funding this work through IRPA research grant 03-02-02-0017-SR0003/07-03.

References

- 1 D. S. Timothy and M. M. Trivedi: International Conference on Intelligent Vehicles 2003 (IEEE, Parma, 2003) p. 570.
- 2 M. Devy, A. Giralt, and A. Marin-Hernandez: Intelligent Vehicle Symposium 2000 (IEEE, Dearborn, 2000) p. 714.
- 3 J. W. Weber, C. A. Mullins, R.W. Schumacher and C. D. Wright: Vehicle Navigation & Information System 1994 (IEEE, Yokohama, 1994) p. 431.
- 4 M. Shiraishi, H. Sumiya and Y. Ysuchiya: Intelligent Transportation System 2002 (IEEE, Singapore, 2002) p. 210.
- 5 W. W. Osmer, M. Blakesley and B. Patrick: Vehicle occupant position detector and airbag control system, US Patent, 6250671, June 26, 2001.
- 6 D. S. Breed: Smart airbag system, US Patent, 6532408, March 11, 2003.
- 7 C. H. Stephan, M. A. Cuddihy and S. Y. Schondorf: Method for controlling airbag deployment, US Patent, 6764095, July 20, 2004.
- 8 D. S. Breed, W. E. DuVall, W. C. Johnson, K. A. Lukin and V. M. Konovalov: Communication method and arrangement, US Patent, 7110880. September 19, 2006.
- 9 N. Shigeyuki: Mitsubishi Motors Technical Review **16** (2004) 61.
- 10 A. Leykin, R. Yang and R. Hammoud: Computer Vision and Pattern Recognition 2007 (IEEE, New York, 2007) p. 1.

- 11 T. Marunaka, T. Kimura, M. Taguchi, T. Yoshikawa, H. Kumamoto and K. Kishida: Railroad Conference 2001(IEEE/ASME, Ontario, 2001) p. 251.
- 12 A. Rovid and G. Melegh: Intelligent Signal Processing 2003 (IEEE, Budapest, 2003) p. 149.
- 13 A. R. Varkonyi-Koczy, A. Rovid and R. M. da-Graca: IEEE Trans. Instrum. Meas. **55** (2006) 2304.
- 14 NHTSA: Proposed new federal Motor vehicle safety standard for motor vehicle side impact protection (FMVSS 214) (2001), <http://www.ita.doc.gov/td/auto/domestic/staffreports/NHTSA%20Side%20Impact%203.pdf> (July 23, 2006).
- 15 National Transportation Safety Board (NTSB): Safety recommendations H-97-19 through -21, http://www.nts.gov/recs/letters/1997/H97_19_21.pdf, Washington, DC 20594 (January 10, 2005).
- 16 M. A. Hannan, A. Hussain, S. A. Samad, A. Mohamed, D. A. Wahab and A. K. Arrifin: Int. J. Automot. Technol. **7** (2006) 827.
- 17 A. Hussain, M. A. Hannan, A. Mohamed, H. Sanusi and A. K. Arrifin: Int. J. Automot. Technol. **7** (2006) 179.
- 18 M. A. Hannan, A. Hussain, A. Mohamed and S. A. Samad: Int. J. Crashworthiness **13** (2008) 579.
- 19 B. Fasel, S. Ben-Yacoub and J. Luetttin: Fast multi-scale face detection, IDIAP-Com 98-04, (1998) 1-87.
- 20 S. Ben-Yacoub, B. Fasel and J. Luetttin: Proceedings Second International Conf. on Audio and Video-based Biometric Person Authentication 1999 (Washington DC, 1999) p. 135.
- 21 H. A. Rowley, S. Baluja and T. Kanade: IEEE Trans. Pattern Anal. Mach. Intell. **20** (1998) 23.
- 22 S. Agarwal, A. Awan and D. Roth: IEEE Trans. Pattern Anal. Mach. Intell. **26** (2004) 1475.
- 23 P. Viola and M. Jones: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2001 (IEEE, Kauai, 2001) p. 511.
- 24 S. Mahamud and M. Hebert: Computer Society Conference on Computer Vision and Pattern Recognition 2003 (IEEE, Wisconsin, 2003) p. 248.