

Design and verification of driver interfaces for adaptive cruise control systems[†]

Sang Hun Lee* and Dae Ryong Ahn

Graduate School of Automotive Engineering, Kookmin University, Seoul, 136-702, Korea

(Manuscript Received November 12, 2014; Revised December 28, 2014; Accepted January 7, 2015)

Abstract

In a highly automated system, mode confusion is a significant human error factor that contributes to accidents. To suppress mode confusion in adaptive cruise control (ACC) systems of vehicles, we developed a new driver interface based on a formal approach to analyze and verify human-automation interaction. To enhance the driver's mode awareness, we developed a new ACC interface that eliminates inconsistent mode transitions by reconfiguring the modes. Then, a human-in-loop experiment was conducted in a simulated environment where a driving simulator was used to evaluate the state and mode awareness of drivers with the old and new interfaces. The experimental results showed that the proposed interface model, which was verified a formal method, was very effective in reducing mode confusion compared with the traditional interface model.

Keywords: Adaptive cruise control; Formal verification; Intelligent vehicle; Mode confusion; User interface

1. Introduction

Recently, intelligent vehicles have been equipped with adaptive cruise control (ACC) systems to enhance the safety of drivers and passengers. ACC systems enhance conventional cruise control systems and they are capable of controlling the speed of a vehicle, which is set by the driver. This allows the host vehicle to follow a forward vehicle at an appropriate distance by controlling the engine and/or power train, and potentially the brake. According to the test specifications of ISO 15622:2010, three operational modes are found in ACC systems, i.e., off, standby, and active, while the active mode also has sub-modes such as speed control and following control [1]. Therefore, it is necessary to consider how to reduce the possibility of mode confusion during the design of driver interfaces for ACC systems.

There has been a lot of research on mode confusion and its resulting automation surprise of highly automated systems, in particular, aircrafts [2-7]. Mode confusion in ACC systems has been investigated by several researchers recently. Horiguchi et al. [8, 9] showed that if different modes exhibit a similar response, users have difficulty distinguishing them. Thus, they proposed a new method that uses mode vectors to estimate the possibility of mode confusion. Furukawa et al. [10] conducted an experimental study of mode awareness using a dual-mode ACC system with high-speed and low-speed range modes.

They determined the information that could be effective for supporting the mode awareness of driver in complex situations if some direct information on the system states is concealed. Heymann and Degani [11] described a hierarchy of automated driving aids and their functionalities, beginning with standard cruise control (CC), followed by ACC with the option of full speed range functionality to allow stopping and starting, before adding automatic lane centering, and the possible future incorporation of navigation functions. Ahn et al. [12] studied possible mode confusions in a simulated environment when vehicles are equipped with an adaptive cruise control system. A set of human-in-loop experiment was conducted to observe possible mode confusions and redesign the user interface to reduce. Lee et al. [13] also developed a new driver interface for the ACC system to suppress mode confusion, based on a formal method.

In order to reduce mode confusion, it is necessary to provide the driver with a transparent display of the automation state, as well as correct and succinct information via the user interface [14, 15]. To satisfy these requirements, in this study, we analyzed a conventional driver interface for ACC systems in a formal method and we examined the occurrence of mode confusion based on human-in-the-loop experiments. In addition, a new driver interface was developed, and it was verified for its improvement in terms of mode confusion via formal verification and human-in-the-loop experiments. The design and verification process ensured that the user's mental model was correct and succinct, while the information display was sufficiently clear to prevent mode confusion.

*Corresponding author. Tel.: +82 2 910 4835, Fax.: +82 2 910 4718
E-mail address: shlee@kookmin.ac.kr

[†]Recommended by Associate Editor Ki-Hoon Shin

© KSME & Springer 2015

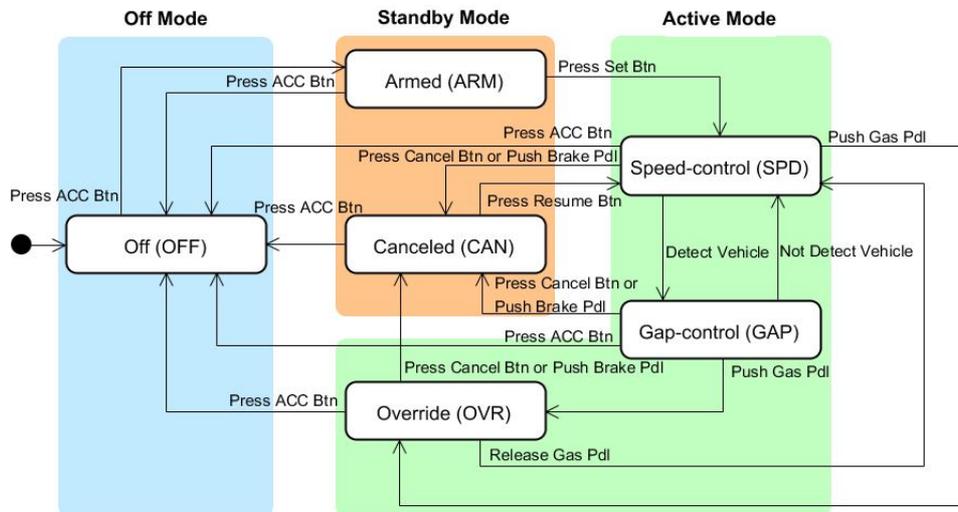


Fig. 1. Machine model and interface model of the ACC system.

This paper is organized as follows. Sec. 1 introduces the background and objectives of this study, including a survey of previous research related to mode confusion and driver interfaces in ACC systems. In Sec. 2, we present the results of a test of a conventional driver interface for an ACC system, where we used a formal analysis method to determine whether its cognitive model of users might cause mode confusion. We also analyzed whether the interface model was consistent with the standardized operational modes. In Sec. 3, based on the results, we developed a new driver interface as an alternative that is more robust to mode confusion, where we followed the rules for clear user interface design. In Sec. 4, we describe the results of driver mode awareness tests in the existing and new driver interfaces based on human-in-the-loop experiments in a simulated environment. In Sec. 5, we summarize the experimental results and discuss them in detail. In Sec. 6, we conclude the study and suggest future research.

2. Machine and user models for the ACC system

2.1 Conventional machine model for the ACC system

As illustrated in Fig. 1, the machine model for the existing ACC systems implemented in most current vehicles comprises six states: off, armed, canceled, override, speed-control, and gap-control. The user-triggered events include pushing and releasing the brake and gas pedals, and pushing various buttons to activate, cancel, and resume the ACC, as well as increasing and decreasing the set speed and set distance. If the driver turns on the ACC system by pushing the ACC button, the ACC enters the armed state, which is a type of ready state that waits for specific activations by the driver. When the set button is pushed, the current speed is set as the set speed and the state is switched to speed-control. A constant vehicle speed is maintained using the set speed in the speed-control state. However, if a vehicle is found in front within the set distance, the state is changed automatically to gap-control,

where the host vehicle follows the target vehicle at a constant distance. Of course, the state returns to the speed-control when the target vehicle exceeds the gap-distance range. In the gap-control or speed-control states, the driver can take control of the vehicle from the ACC by pushing the gas pedal. This is called the override state. The state returns to speed-control as soon as the gas pedal is released in the override state. The canceled state is a standby state in ACC, which occurs when the brake is applied or the cancel button is pushed in the gap-control or speed-control states. The resume button should be pushed by the driver in order to recover the speed-control state from the canceled state.

2.2 Conventional interface model for the ACC system

The interface model for the ACC system, which was suggested in the test specification of ISO 15622:2010 [1] and is used most widely in existing automobiles, provides three modes, i.e., active, standby, and off, where transitions occur between modes via user-triggered events, as shown in Fig. 1. A single mode in an interface model actually includes one or multiple states in a machine model. In the existing interface model, as shown in Fig. 1, the gap-control, speed-control, and override states are located in the active mode, the canceled and armed states are in the standby mode, and the off state is in the off mode.

2.3 Conventional user interface for the ACC system

In this study, a user interface was implemented based on the interface of the ACC system used by the major automotive manufacturers including Hyundai [16]. In Fig. 2, “ACC” and “Set 70 Km/h” are parts that represent a mode in the ACC system, while the figure in the center of the gauge cluster represents the internal states of the active mode. As shown in Table 1, “ACC” distinguishes the off mode from the other

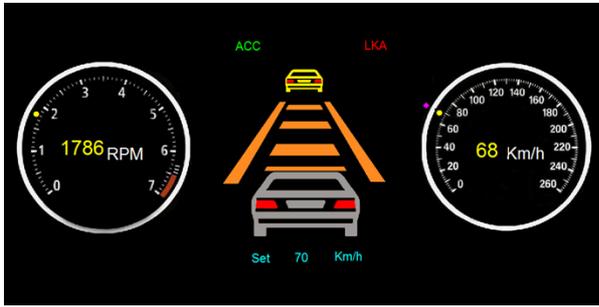


Fig. 2. User interface of the ACC system implemented based on that of the Hyundai Equus [13].

Table 1. User interface for each mode in the existing ACC system [13].

State	Mode	Existing Interface
Off	Off	
Armed	Standby	ACC
Canceled		
Override	Active	ACC
Speed-control		Set 50 km/h
Gap-control		

modes. “Set_km/h” distinguishes the active and standby modes; thus, if “Set_km/h” is turned off, this indicates the standby mode, whereas, if “Set_km/h” is turned on, this indicates the active mode.

2.4 Conventional user interface for the ACC system

The state transitions triggered by the driver’s direct operations in the ACC system can be summarized by a state transition table, as shown in Table 2. The same user-triggered operation is applied to the states that belong to the same mode, but they may make transitions into states that belong to the same or different modes. From the user’s viewpoint, this

should not be a problem because they make transitions into states in the same mode. However, if they make transitions into different modes, the users may be confused and surprised by incompatible mode changes. These cases are indicated by “X” in the final column: “Incompatible.” Incompatible transitions may occur in the standby and active modes. When the resume button is pushed, the canceled (CAN) and armed (ARM) states in the standby mode can make transitions into the speed-control state in the active mode, or remain in the armed state in the standby mode, thereby resulting in transitions into different modes. Because the armed and canceled states in the same mode described above have the same display, a driver cannot distinguish them. Moreover, in the active mode, the override state is incompatible with the gap- and speed-control states because the brake pedal is not available in the override state whereas it is available in the other states. Therefore, the driver becomes confused and surprised by the unexpected incompatible modes produced by the same user input. Furthermore, although mode confusion does not occur when the override state belongs to the active mode, it is not acceptable for users if the override state is grouped into the active mode because the ACC system loses control of the vehicle in the override state.

3. Development of a new user interface for the ACC system

3.1 New interface model for the ACC system

To solve the problems with the conventional interface model described in the previous section, a new interface model and its user interface had to be designed. In this study, we assumed that the machine states could not be changed. Therefore, the standby and active modes were partitioned for each state to remove the incompatible mode transitions. As a result, as shown in Fig. 3, the standby mode was separated into the canceled and armed modes, and the active mode was separated into the override and active modes. The state and mode transition table of the new user interface, as shown in Table 3, indicates that the incompatible mode transitions were clearly eliminated.

Table 2. Transition among states and modes caused by user-triggered operations in the conventional user interface.

Mode		Off	Standby		Active		
State		OFF	ARM	CAN	OVR	SPD	GAP
Driver’s operation	Push gas pedal					OVR	OVR
	Release gas pedal				SPD		
	Push brake pedal					CAN	CAN
	Press cancel button				CAN	CAN	CAN
	Press resume button			SPD			
	Press set button		SPD	SPD			
	Press ACC button	ARM	OFF	OFF	OFF	OFF	OFF
State's compatibility	Compatible	X				X	X
	Incompatible		X	X	X		

Table 3. Transitions of states and modes caused by user-triggered events in the final interface model.

Mode		Off	Armed	Canceled	Override	Active	
State		OFF	ARM	CAN	OVR	SPD	GAP
Driver's operation	Push gas pedal					OVR	OVR
	Release gas pedal				SPD		
	Push brake pedal					CAN	CAN
	Press cancel button				CAN	CAN	CAN
	Press resume button			SPD			
	Press set button		SPD	SPD			
	Press ACC button	ARM	OFF	OFF	OFF	OFF	OFF
State's compatibility	Compatible	X	X	X	X	X	X
	Incompatible						

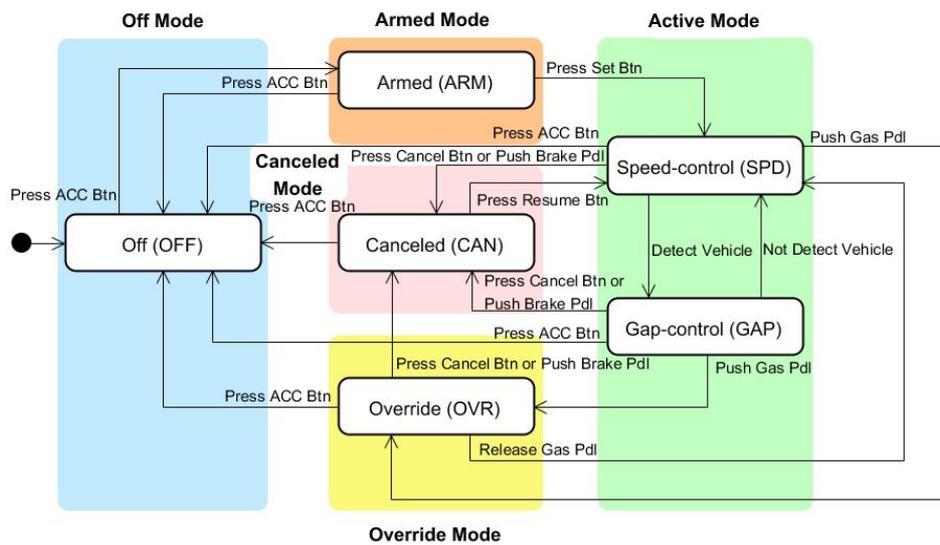


Fig. 3. Final mode design of the interface model for the ACC system.

3.2 New user interface for the ACC system

On the basis of the new interface model proposed in the previous section, we designed a new user interface for the ACC system, as shown in Table 4. The texts and colors of “ACC” and “ACC standby,” as well as the color of “Set ** km/h” are used to identify and distinguish the modes and states more clearly than the previous user interface. This claim was verified in the experiments described in Sec. 4.

4. Human-in-the-loop experiments

4.1 Participants

Ten participants, i.e., two females and eight males aged between 24 and 28 years (Mean = 25.2 years, SD = 1.23 years), were selected from Kookmin University. All had a valid driver’s license with driving experience of one or more years, but little experience of the ACC system. They had normal or corrected to normal vision.

4.2 Procedure

Before the experiment, the participants completed a consent form and were informed about the goal and experimental method. In particular, the experimenter explained the ACC system in detail, including its operational modes and states. Next, each participant obtained hands-on experience of the ACC system in the driving simulator.

In the experiment, a driver started the vehicle and turned on the ACC system. According to the design scenario, a specific event occurred after the host vehicle arrived at a certain location. The typical events were a sudden maneuver by an adjacent vehicle or the appearance of a road construction sign. During each event, the driver tried to control the vehicle in a safe manner by applying the brake or turning the steering wheel in response to the given situation. The experimenter observed the subject’s actions and the resulting mode and state changes in the system. After each event was complete, the experimenter interrupted the driving simulation to ask the

subject about the mode and state changes, and their reason for giving the specific answer. The experimenter noted the answers given in the experiment. The participants were expected to use the ACC system while driving. Therefore, if the active mode was canceled by applying the brake, the subject had to resume the system as soon as the emergency situation was over. The subjects were given no clues about the correct answers. An experimental session lasted 30 minutes from start to finish.

4.3 Driving simulator

The experiments were conducted in a fixed-base driving simulator with TNO PreScan software [17]. Although there

Table 4. New user interface based on the final interface model for the ACC system [13].

State	Mode	Proposed interface
Off	Off	
Armed	Armed	ACC Standby Set ___ km/h
Canceled	Canceled	ACC Standby Set 50 km/h
Override	Override	ACC Set 50 km/h
Speed-control	Active	ACC Set 50 km/h
Gap-control		

was a driving simulator equipped with a motion platform in our university [18], it was not used in this work because its development environment did not allow easy implementation of different graphic user interfaces for instrument clusters. The simulator had three 42-inch widescreen displays, thereby creating a 130° horizontal and 25° vertical field of view, as well as a 10-inch display for the gauge cluster. The input device was a Logitech G27 racing wheel with gas and brake pedals. The buttons on the wheel were defined to produce various operations during ACC mode transitions. In the driving simulator, the ACC system was implemented based on PreScan using MathWorks Matlab and Simulink. PreScan is a physics-based simulation platform that provides various sensor emulation functions, which facilitate the rapid development of various active safety systems or advanced driver assistance systems.

4.4 Scenario

To perform tests while driving in the simulator, a road with six lanes (Three lanes in each direction) was modeled using PreScan. The road was based on the F-1 race circuit of the

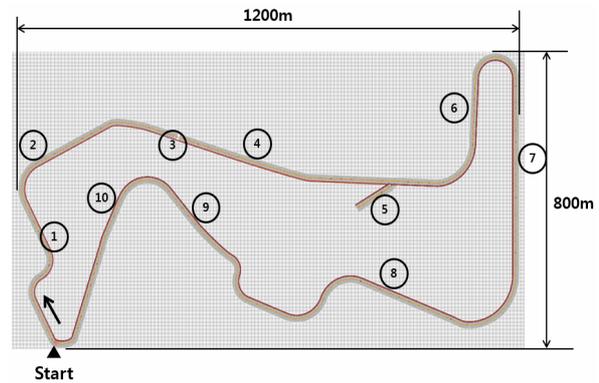


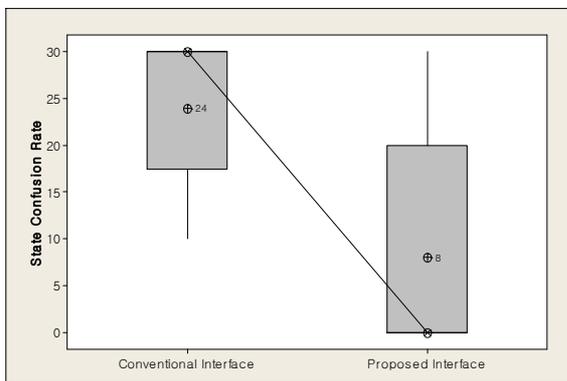
Fig. 4. Test road and the locations where the events occurred.

Table 5. Events and their expected state transitions.

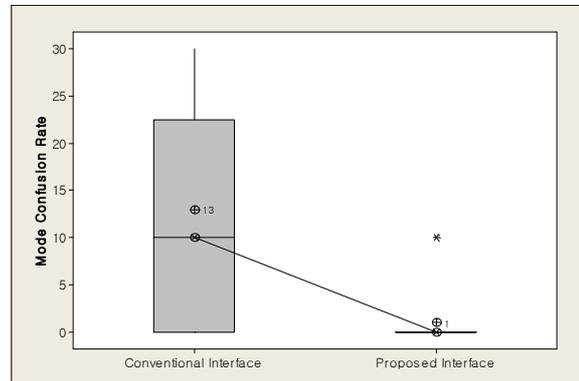
Event No.	Designed traffic situation		Expected driver operation	Expected state transitions	
	Actor	Action		Before	After
1	V ₄	Sudden cutting in front of V ₁	Brake / None	SPD	CAN/GAP
2	V ₄	Returning to the 2 nd lane	None	GAP	SPD
3	V ₂	Smooth braking	Brake / None	GAP	CAN/GAP
4	V ₂	Sudden braking	Brake	GAP	CAN
5	V ₅	Sudden cutting in front of V ₁ from an on-ramp	Brake / None	SPD	CAN/GAP
6	V ₅	Lane change to the 2 nd lane when V ₁ is in the 1 st lane (or V ₁ in the 2 nd lane)	None	GAP (SPD)	SPD (GAP)
7	V ₁	Speeding up	Pressing and releasing the gas pedal	OVR	SPD/GAP
8	V ₁	Turning off the ACC System	Pressing the ACC switch	SPD	OFF
9	V ₁	Turning on the ACC System	Pressing the ACC switch	OFF	ARM
10	V ₁	Changing the gap distance	Pressing the distance switch	GAP	GAP/SPD

Table 6. State and mode confusion rates with the conventional ACC driver interface in the experiments [13].

		States/Modes Recognized by Drivers						State confusion	Mode confusion	Total
Actual States/Modes	Mode	Off	Standby		Active					
	State	OFF	ARM	CAN	OVR	SPD	GAP			
Off	OFF	10 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	10 (100%)
	ARM	0 (0%)	5 (50%)	1 (10%)	0 (0%)	1 (10%)	3 (30%)	5 (50%)	4 (40%)	10 (100%)
	CAN	5 (23%)	2 (9%)	8 (36%)	0 (0%)	0 (0%)	7 (32%)	14 (64%)	12 (55%)	22 (100%)
Active	OVR	0 (0%)	0 (0%)	0 (0%)	10 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	10 (100%)
	SPD	0 (0%)	0 (0%)	0 (0%)	0 (0%)	16 (80%)	4 (20%)	4 (20%)	0 (0%)	20 (100%)
	GAP	0 (0%)	0 (0%)	0 (0%)	0 (0%)	4 (14%)	24 (86%)	4 (14%)	0 (0%)	28 (100%)
Total		15	7	9	10	21	38	27	16	100



(a)



(b)

Fig. 5. Box plots for state and mode confusion rates for the conventional and proposed ACC interfaces: (a) A box plot for state confusion rates; (b) A box plot for mode confusion rates.

Singapore Grand Prix. As shown in Fig. 4, the road had straight and curved sections, and its total length was about 5 km. The vehicle started from the lower left corner of the road. Ten events were designed that occurred at the specific regions numbered 1–10 in Fig. 4. Table 5 shows the traffic situation, the expected driver operation, and the ACC state change during each event.

5. Results and discussion

The following data were collected for each event in the experiment: the state and mode before the event, the subject’s operation, the actual state and mode after the operation, the state and mode that the subject recognized after the operation, and the reason why the subject thought they were in a specific state and mode. We examined the state and mode confusion rates for the conventional and proposed ACC interfaces. Their

medians and means are shown in the box plots in Fig. 5. We performed the Wilcoxon signed-rank tests to analyze the effect of interface type because the distribution of the rates are not normal ($p = 0.005 < 0.05$ in the Anderson-Darling normality test) and the sample size is only 10. The medians of the state confusion rates of the conventional and proposed interfaces are 30% and 0%, respectively, and the medians of the mode confusion rates are 10% and 0%, respectively. The means of the state confusion rates of the conventional and proposed interfaces are 24% (SD = 8.43%) and 8% (SD = 11.35%), respectively, and the means of the mode confusion rates are 13% (SD = 11.60%) and 1% (SD = 3.16%), respectively. The test results show that $p = 0.018 < 0.05$ in the difference of the state confusion rates of two different interfaces, and $p = 0.022 < 0.05$ in the difference of the mode confusion rates. Therefore, it can be concluded that there is a statistically significant difference in the state and mode confusion rates

Table 7. State and mode confusion rates with the new ACC driver interface in the experiments [13].

Mode		States/modes recognized by drivers						State confusion	Mode confusion	Total	
		Off	Armed	Canceled	Override	Active					
State	State	OFF	ARM	CAN	OVR	SPD	GAP				
Actual States/Modes	Off	OFF	10 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	10 (100%)
	Armed	ARM	0 (0%)	10 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	10 (100%)
	Canceled	CAN	0 (0%)	1 (5%)	18 (95%)	0 (0%)	0 (0%)	0 (0%)	1 (5%)	1 (5%)	19 (100%)
	Override	OVR	0 (0%)	0 (0%)	0 (0%)	10 (100%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	10 (100%)
	Active	SPD	0 (0%)	0 (0%)	0 (0%)	0 (0%)	16 (80%)	4 (20%)	4 (20%)	0 (0%)	20 (100%)
		GAP	0 (0%)	0 (0%)	0 (0%)	0 (0%)	3 (10%)	28 (90%)	3 (10%)	0 (0%)	31 (100%)
Total			10	11	18	10	19	32	8	1	100

between the conventional and proposed interfaces.

Tables 6 and 7 show confusion tables for the conventional and proposed user interfaces. The confusion tables allow us to investigate how the participants recognized the changes in the state and mode of the ACC system during the experiment because they compare the actual states with the states recognized by the subjects. The data in the shaded diagonal cells indicate correctly recognized states. As shown in Table 6, 73/100 state changes were recognized correctly using the conventional user interface. Among the 27 cases where state confusion occurred, 16 cases were related to mode confusion. The highest mode confusion rate occurred in the standby mode. The standby mode was confused most often with the active mode. The armed state in the standby mode was confused with the gap-control or speed-control states in the active mode at a rate of approximately 40%. These cases of confusion occurred in event 8 when the ACC system was turned off and then on. In the current ACC system, after the ACC button was pushed, the state changed into the armed state in the standby mode. When the set button was pushed again, the state changed into speed-control in the active mode. However, some of the participants were confused and they believed that the state changed immediately into speed-control or gap-control in the active mode after the ACC button was pushed. This type of mode confusion was experienced by most of the participants who were not familiar with the ACC system. They had no experience of a standby mode in most electronic appliances, which usually execute immediately after they are switched on. Thus, an appropriate ACC system should have an improved operational flow where the set speed is set and the speed-control state activates immediately after the ACC system is turned on by pushing the ACC button.

Table 6 shows that another high rate of mode confusion occurred in the canceled state in the standby mode, which was

confused with the off mode at a rate of 23% and with the gap-control state in the active mode at a rate of 32%. These cases occurred in events 3, 4, and 5 when the subjects pushed the brake pedal because an adjacent vehicle cut in suddenly, or when a vehicle in front stopped suddenly. In the cases where the canceled state was confused with the off state, the subjects thought that the state became the off state in the off mode because the system was not working anymore after they pushed the brake pedal. However, the actual state was the canceled state in the standby mode. In the cases where the canceled state was confused with the gap-control state in the active mode, the subjects thought that even if they pushed a brake, the state would return to the gap-control state in the active mode after they released the brake so the ACC system would continue to work. This mode confusion was also related to the graphic user interface in the ACC system. When the state changed from active to canceled, the ACC indicator light was still turned on and it was the same color. Although the “Set *** km/h” display showing the set speed disappeared in this case, the subjects perceived that the speed information was not related to the operational mode, and thus they believed that the ACC function was still working. Therefore, in the graphic user interface of the ACC system, a change in modes needs to be displayed clearly in the screen region related to the mode. Another cause related to this confusion was that the subjects expected the brake pedal to work in a similar manner to the gas pedal, but it actually worked differently. When the gas pedal was pushed in the active mode, the state changed temporarily to the override state but returned back to the previous active mode as soon as the gas pedal was released. However, if the brake pedal was pushed and released in the active mode, the state did not return to the active mode and it remained in the canceled state in the standby mode. Therefore, in an ACC system, it is necessary to reduce the mode confu-

sion among users by ensuring that the mode transition works in the same manner in both cases for the gas pedal and brake pedal applications.

Table 7 shows a confusion table for the proposed user interface in the ACC system. The table shows that 92/100 state changes were recognized correctly whereas eight changes were recognized incorrectly. One of these errors involved mode confusion where the canceled mode was recognized incorrectly as the armed mode. A relatively high confusion rate occurred between speed-control and gap-control states in the same active mode. The cases where speed-control was misrecognized as gap-control occurred mainly in event 2, where the host vehicle was driving in the gap-control state and the two vehicles in front were in the same lane. Next, a target vehicle in front left the lane and thus the ACC state was changed automatically to speed-control by the system. However, 3/10 subjects recognized the state incorrectly as gap-control because another vehicle was present in front in the same lane. By contrast, in event 6, only one driver recognized the state incorrectly because there was no vehicle in front when the target vehicle left the lane. In all cases, although the speed-control and gap-control states were confused, the driver was not surprised because the two states belonged to the same mode and any operation caused the same mode transition.

These low mode confusion rates demonstrate that the proposed driver interface reduces mode confusion because the same user input operation always leads to the same mode transition and the mode information is displayed clearly in the mode indication area on the gauge cluster screen.

6. Conclusions

In this study, we analyzed and verified mode confusion using the conventional and our proposed interface model and user interface in the ACC system based on a formal method, followed by human-in-the-loop experiments in driving simulators. The results and contributions of this study can be summarized as follows.

The conventional driver interface for an ACC system was tested using a formal analysis method to determine whether its cognitive model of users might cause mode confusion. We also analyzed whether the interface model was consistent with the standardized operational modes.

Based on the analysis results of the conventional interface, we developed a new driver interface as an alternative that is more robust to mode confusion, where we followed the rules for clear user interface design.

The experimental results showed that the proposed interface model, which was verified a formal method, was very effective in reducing mode confusion compared with the conventional interface model. The total rate of mode confusion with the conventional user interface was 16%, whereas that with the proposed user interface was only 1%. The results also showed that the formal method is an effective and useful tool

for designing a human–automation interface.

To reduce mode confusion, it is necessary to provide the driver with a transparent display of the automation state, as well as correct and succinct information via the user interface. To prevent mode confusion from occurring, it is necessary to apply a simple and accurate interface model, as well as providing a clear interface display and internal information in a transparent manner.

In future research, the following issues need to be addressed.

The ACC user interface proposed in this study prevents state and mode confusion very effectively, but it has many modes and they are excessively detailed, which makes it difficult for users to learn and to gain familiarity. This is because we accepted the conventional machine state model and then designed the interface model without modifying the machine model. This strategy restricted the freedom of the interface model design and prevented us from developing a user-oriented system design. Therefore, to obtain a more user-friendly product design, it will be necessary to design the interface model first, before the ACC states are designed and implemented. The use of fewer modes and consistent mode transitions would increase the usability of the system.

There is an urgent need to develop optimal user interfaces for automated systems because of the recent advent of autonomous vehicles such as Google cars. The National Highway Traffic Safety Administration (NHTSA) has defined five levels of vehicle automation. As the level of automation increases, the role of the driver shifts from primary control to supervisory control. At level 1 of automation, one or more specific control functions operate independently of each other. At level 2 of automation, at least two primary control functions work in unison to relieve the driver of the control of these functions. At level 3 of automation, the driver can give up full control of all safety-critical functions in certain traffic or environmental conditions. Future vehicles will set the level automatically for a driver or a system, depending on the state of the driver, vehicle, and environment. Thus, drivers should not be surprised if they are confused by the levels or modes. Based on the current NHTSA criteria, the present study only addressed level 1 of automation. Therefore, it is necessary to extend this research to level 2 and 3 automated vehicles [19, 20], which include automatic lane centering and lane changing systems, as well as ACC, and to develop mental models for the users and driver interfaces that minimize mode confusion.

Recently multi-modal interaction technology has been introduced in driver-vehicle interface [21–25]. In order to reduce not only driver distraction but also mode confusion and automation surprise, this new technology needs to be investigated for design of driver-vehicle interface.

Acknowledgment

This research was supported by National Research Foundation of Korea Grant funded by Korea Ministry of Science, ICT & Future Planning (MSIP) (Grant NRF-2013R1A2A2A0

1068766). It was also supported by Research Program 2012 of Kookmin University in Korea. The authors express their gratitude to Prof. Ji Hyun Yang for providing precious advices and to Hwisoo Eom for drawing figures and formatting the manuscript.

References

- [1] ISO 15622:2010, *Intelligent transport systems-Adaptive cruise control systems - Performance requirements and test procedures* (2010).
- [2] D. A. Norman, The problem with automation: Inappropriate feedback and interaction, not over-automation, *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 327 (1241) (1990) 585-593.
- [3] G. A. Jamieson and K. J. Vicente, Designing effective human-automation-plant interfaces: A control-theoretic perspective, *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 47 (1) (2005) 12-34.
- [4] N. B. Sarter and D. D. Woods, How in the world did we ever get into that mode mode error and awareness in supervisory control, *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37 (1) (1995) 5-19.
- [5] N. B. Sarter, D. D. Woods and C. E. Billings, *Automation surprise, Handbook of Human Factors and Ergonomics*, Second Ed., Wiley Publishing Company, New York, USA (1997) 1926-1943.
- [6] M. L. Bolton, F. J. Bass and R. I. Siminiceanu, Using formal verification to evaluate human-automation interaction: A review, *IEEE Transactions on Systems, Man and Cybernetics: Systems*, 43 (3) (2013) 488-503.
- [7] R. W. Butler, S. P. Miller, J. N. Potts and V. A. Carreno, A formal methods approach to the analysis of mode confusion, *Proc. of AIAA/IEEE 17th Digital Avionics Systems Conference*, Bellevue, WA, USA (1998) C41-1~C41-8.
- [8] Y. Horiguchi, R. Fukuju and T. Sawaragi, An estimation method of possible mode confusion in human work with automated control systems, *Proc. of SICE-ICASE International Joint Conference*, Busan, South Korea (2006) 943-948.
- [9] Y. Horiguchi, R. Fukuju and T. Sawaragi, Differentiation of input-output relations to facilitate user to input-E international joint conference, with automated control, *Proc. of 2007 IEEE International Conference on Systems, Man and Cybernetics*, Montreal, Quebec, Canada (2007) 2570-2575.
- [10] H. Furukawa, T. Inagaki, Y. Shiraishi and T. Watanabe, Mode awareness of a dual-mode adaptive cruise control system, *Proc. of 2003 IEEE International Conference on Systems, Man and Cybernetics*, Washington, DC, USA (2003) 832-837.
- [11] M. Heymann and A. Degani, Automated driving aids: Modeling, analysis, and interface design considerations, *Proc. of the 5th International Conference on Automotive User Interfaces and Interactive Vehicular Application*, Eindhoven, Netherlands (2013) 142-149.
- [12] D. R. Ahn, J. H. Yang and S. H. Lee, A study on mode confusions in adaptive cruise control systems, *Trans. Korean Soc. Mech. Eng. A.* (2015) (Submitted).
- [13] S. H. Lee, D. R. Ahn and J. H. Yang, Mode confusion in driver interfaces for adaptive cruise control systems, *Proc. of the 2014 IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, CA, USA (2014) 4131-4132.
- [14] A. Degani and M. Heymann, Formal verification of human-automation interaction, *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 44 (1) (2002) 28-43.
- [15] M. Heymann and A. Degani, Formal analysis and automatic generation of user interfaces: Approach, methodology, and an algorithm, *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49 (2) (2007) 311-330.
- [16] Hyundai Motor Co., <http://www.hyundai.com/kr/blu/select/DIExpdList.do#none>.
- [17] TASS International, PreScan, <http://www.tassininternational.com/prescan>.
- [18] J.-Y. Lee and W.-S. Lee, A symbolic computation method for automatic generation of a full vehicle model simulation code for a driving simulator, *Journal of Mechanical Science and Technology*, 19 (1) (2005) 395-402.
- [19] T. E. Trimble, R. Bishop, J. F. Morgan and M. Blanco, Human factors evaluation of level 2 and level 3 automated driving concepts: Past research, state of automation technology, and emerging system concepts, *Report No. DOT HS 812 043*, Washington, DC, National Highway Traffic Safety Administration (2014).
- [20] A. Marinik, R. Bishop, V. Fitchett, J. F. Morgan, T. E. Trimble and M., Blanco, Human factors evaluation of level 2 and level 3 automated driving concepts: Concepts of operation, *Report No. DOT HS 812 044*, Washington, DC, National Highway Traffic Safety Administration (2014).
- [21] C. Pickering, K. Burnham and M. Richardson, A review of automotive human machine interface technologies and techniques to reduce driver distraction, *Proc. of the 2nd IET International Conference on System Safety*, London, UK (2007) 223-228.
- [22] B. Pflöging, S. Schneegass and A. Schmidt, Multimodal interaction in the car: Combining speech and gestures on the steering wheel, *Proc. of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Portsmouth, NH (2012) 155-162.
- [23] H. Park, H.-K. Jung and S.-J. Park, Tangible AR interaction based on fingertip touch using small-sized nonsquare markers, *Journal of Computational Design and Engineering*, 1 (4) (2014) 289-297.
- [24] Y. Endo, M. Tada and M. Mochimaru, Reconstructing individual hand models from motion capture data, *Journal of Computational Design and Engineering*, 1 (1) (2014) 1-12.
- [25] H. Ha and K. Ko, A method for image-based shadow interaction with virtual objects, *Journal of Computational Design and Engineering*, 2 (1) (2015) <http://dx.doi.org/10.1016/j.jcde.2014.11.003> (in press).



Sang Hun Lee is a professor in Automotive Engineering at Kookmin University in Seoul, Korea. He received his BE, ME, and Ph.D. degrees in Mechanical Design and Production Engineering from Seoul National University in 1986, 1988, and 1993, respectively. His research interests include computer-aided

design, human-machine interaction, ergonomics, and artificial intelligence for automotive industry.



Dae Ryong Ahn is a researcher at Korea Automotive Technology Institute in Cheonan, Korea. He received his MS degree in Automotive Engineering from Kookmin University in 2014. His research interests include advanced driver assistance system, intelligent control system and human-machine interaction

for automotive industry.