SAFE STRIP: Hardware in the Loop motorcycle simulator experiment for C-ITS applications

Ioannis Symeonidis[1], Nikos Dimokas[1], Ioannis Gragkopoulos[1], Ioannis Tsetsinas[1], Evangelia Chrysochoou[1], Katerina Touliou[1], Giammarco Valenti[2], Francesco Biral[2], Maria Gkemou[1], Evangelos Bekiaris[1]
[1]Hellenic Institute of Transport, Centre for Research and Technology Hellas, Greece
[2]Department of Industrial Engineering, University of Trento, Italy

Abstract

The SAFE STRIP project (SAFE and green Sensor Technologies for self-explaining and forgiving Road Interactive aPplications) introduces an electronic road strip technology that can provide C-ITS functionality to existing road infrastructure. The SAFE STRIP technology among other vehicles has application also to motorcycles. SAFE STRIP can detect critical events such as vehicles traveling in the wrong direction on the highway, adverse weather conditions, deteriorated road surface and inform the rider with intuitive personalized messages in his language. Experiments took place before field tests in order to verify the correct functionality of the V2I communications (Road Side Unit), the decision logic (Co-driver and DSS) and to assess the HMI (smartphone app) of the SAFE STRIP prototype technology. The CERTH-HIT motorcycle simulator was used to test the system in Hardware in the Loop configuration. Communication, decision logic, HMI and simulator deficiencies were tuned and corrected during the experiment. No virtual accident happened while the system received mainly positive feedback from the riders.

1. Introduction

The SAFE STRIP project (SAFE and green Sensor Technologies for self-explaining and forgiving Road Interactive aPplications) introduces a road solution, the so-called “strip”, that can provide C-ITS functionality exploiting the existing road infrastructure aiming at a more consistent and in-time warning with regard static and dynamic environmental, traffic and road conditions in order to prevent critical traffic incidents. The SAFE STRIP technology among other vehicles has application also to motorcycles. SAFE STRIP, through its multisensorial platform and built driver and rider applications, can detect various critical events such as vehicles traveling in the wrong direction on the highway, adverse weather conditions, deteriorated road surface and inform/warn the driver/rider with intuitive personalized messages in their language. The relevant information arrives far earlier before visual contact so the rider has enough time to adapt his/her speed and overall driving behaviour.

Experiments took place prior the field tests in order to verify the correct functionality of the Infrastructure-to-Vehicle (I2V) and V2I communications enabled through a custom Road Side Unit (called Road Side Bridge - RSB), the decision logic (co-driver and Decision Support System (DSS)) and secondly, to assess the Human Machine Interface (HMI) of the SAFE STRIP prototype solution. Users experienced the solution through a smartphone application. Hardware-in-the-Loop (HiL) methodology was used for the experiments.

In HiL simulation a part of the real hardware is included in the simulation loop during system development. Rather than testing the control algorithm on a purely mathematical model of the system the real hardware is used in a simulated environment that can provide the appropriate signals. HiL simulation is used for the design of anti-lock braking systems (ABS), traction control systems (TCS), suspension systems and others (Bacic 2005). In our case the real hardware was the cloud infrastructure and the smartphone while the simulated environment was the motorcycle simulator the road strips and the RSB. The virtual implementations communicated with the cloud implementation, while warning messages were sent to the smartphone from the cloud.

The objectives of the experiments were:

2. to test and tune the function and communication of the SAFE STRIP components,
3. to study the reaction of the volunteers to the warning messages, and
4. to collect feedback from the volunteers concerning the system and its application for motorcycles.

All the above objectives were addressed before the field experiments.

5. Materials and Methods

The experiments were performed with the CERTH motorcycle simulator (Fig 1) (Nehaoua 2007, Symeonidis 2018). The motorcycle simulator represented a non-equipped with V2x and IVIS (In Vehicle Information System) vehicle.

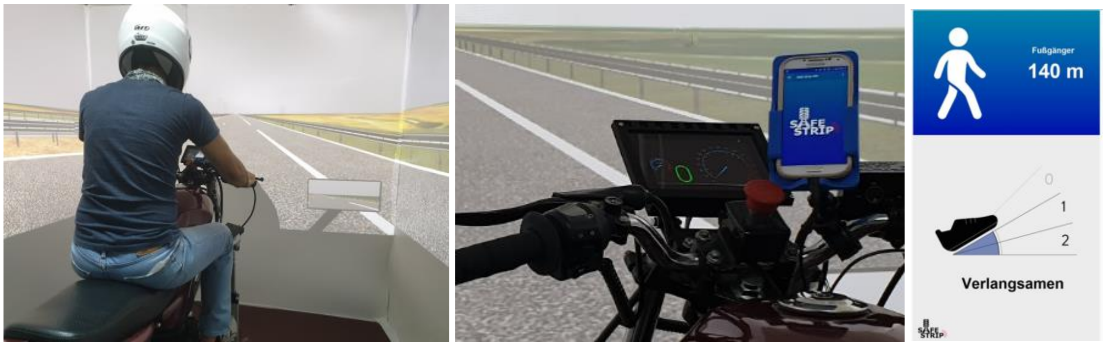

Figure 1: CERTH Motorcycle simulator, the app running on the smartphone and a warning message.

The volunteer rode the simulator and during the ride warning messages (Fig 1) appeared on the smartphone attached with a grip to the handlebar while audio warnings were sent to the rider's headset to prepare the volunteers for a critical event in their course.

Two simulator setups and two scenarios were tested (Table 1).

Table 1 Motorcycle simulator scenarios

	Setup 1		**Setup 2**	
Road condition	Dry road		Wet road	
Scenario	Pedestrian	Wrong way driving	Pedestrian	Wrong way driving

The difference between the two setups concerned the road traction available for braking and acceleration. In Setup 1, the maximum traction allowed a braking deceleration of 7 m/s², while in the wet road conditions the maximum deceleration was 3.6 m/s².
The two different scenarios concerned the critical event. One was a vehicle entering the highway from a highway exit (Fig. 2) and the other a pedestrian crossing the road (Fig. 3). Virtual road strips were simulated on the simulator on the road of the highway exit and on the pedestrian crossing.

Figure 2: The wrong way driving scenario with the car entering the highway from a highway exit (view from the motorcycle rider and zoomed view)

Figure 3: A pedestrian crossing the road on a crossing (view from the motorcycle rider and zoomed view)

For the realization of the scenarios apart from the motorcycle simulator, the Hardware, in the Loop, was a cloud server and a smartphone. The cloud server was running an MQTT broker, the critical event detection logic (Co-driver) and a warning message prioritization system (DSS). The smartphone functioned also as the virtual RSB for the road strips.

In more detail the Co-Driver module implements a virtual agent used to understand what the driver is doing with respect to the potential incoming danger (i.e. pedestrian on cross walk or wrong way driving vehicle) and issuing a warning if a safe manoeuvre is available to achieve the goal required by the driving scenario (e.g. stopping at a cross-walk). The Co-Driver module receives the potential danger via DENM message it calculates all the feasible and human like (smooth and optimal) manoeuvres to stop the vehicle before the danger given the actual vehicle state (i.e. velocity, acceleration and distance to the danger) (Da Lio 2018). The manoeuvres also consider the potential traction available in the specific portion of road.

The Co-Driver rates all the computed manoeuvres based on the effort necessary to change the longitudinal dynamics (i.e. acceleration and speed) in order to enter in the safe state (i.e. stop before the danger).

The following paradigm explains the line of communication. The motorcycle emits its position with the smartphone application while information from the road strips is sent to the cloud server through the RSB. The cloud server detects critical events with the Co- driver component and prioritizes warning messages with the DSS before emitting them. The warning messages are received from the app preparing the rider for the event.

The communication protocols, implemented via MQTT, followed the ETSI ITS- G5 standard (Table 2 and 3).

Table 2 Virtual actors and SAFE STRIP components running in different computing systems

Simulator	Smartphone	Cloud server
Ego vehicle	RSB	MQTT broker
Road Strip (for vehicles and pedestrians)	SAFE STRIP app	Co-driver
Other vehicle		DSS
Pedestrian		

Table 3 Wireless communications between the different computing systems

Communication direction	Type of information	Communication technology standard
From ego vehicle to SAFE STRIP app	Ego vehicle speed, acceleration, geodetic coordinates	Bluetooth
From SAFE STRIP app to MQTT broker	Vehicle CAM	WiFi
From virtual road strip to RSB	Strip id when activated	Bluetooth
From RSB to MQTT broker	DENM	WiFi
From DSS to SAFE STRIP app	HMI Input by App Active message	WiFi

Two log files were used in order to collect the simulation data. Since the motorcycle rider did not have direct interaction with the cloud server the traffic of the messages in the Cloud server was not relevant, only the time duration between sending the strip id from the smartphone and receiving the warning message to the smartphone was logged for synchronization purposes.

Table 4 Log files

Simulator	Smartphone
Simulator controls	DENM message
Ego vehicle, other vehicle and pedestrian kinematics and vehicle dynamics parameters	HMI Input by App Active message
Strip id when activated	Strip id when activated

In order to synchronise the two log files, the message sent from the simulator to the smartphone with the strip identifier, when the strip was activated, was recorded in both files.

The motorcycle simulator vehicle dynamics, motorcycle controls and actuator motor control are running on a Real Time Operating System (RTOS) with a frequency of 100Hz while the simulator's visual environment the smartphone and the cloud server are running in General Purpose Operating Systems (GPOS). Most of the SAFE STRIP components had very small delays in the order of a couple milliseconds. Only the communications to the cloud and the smartphone display of the audio and visual warning had significant delays. These delays were subtracted from the measurement in order to have the reaction time of the volunteer from the moment they receive the warning.

Since smartphones are not following real time constraints and Android and iOS are GPOSs they have variable response times based on the process scheduling. Because these delays were not recorded in the smartphone or the simulator log file, an ad-hoc experiment was performed to measure these app delays (delay for visual message and delay for audio message).

The app was running on an Android mobile phone (Samsung S9+, SM-G965F, Android v.9). A Light Dependent Resistor (LDR) was attached on the screen of the smartphone and headphones were plugged on the 3.5 mm headphone jack of the smartphone. An oscilloscope (Rohde & Schwarz RTB2004) with four channels was used for the measurement. At one channel of the oscilloscope the device that sends the "Strip identifier when activated" message through Bluetooth from the simulator to the smartphone was connected and at the two other channels the headphones and the LDR sensor were attached. The delay between the simulator message and the reception of the warning message was also recorded (Fig 4).

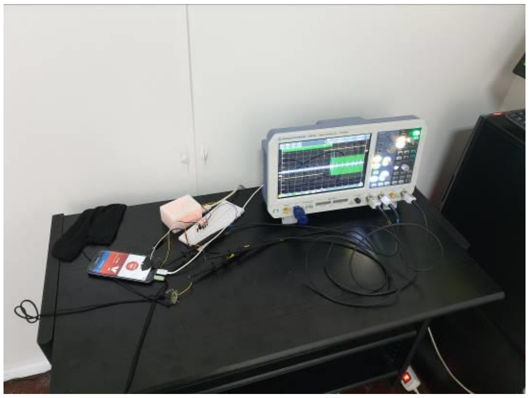

Figure 4: Smartphone delays measurement with an oscilloscope the timing delays of the audio and visual warning

From this experiment an average delay of around 400ms was measured for the audio message and additionally around 100ms for the visual display. The delay was different if different smartphones were used. During the experiment the specific smartphone was used.

The recruited volunteers were informed about the purpose of the experiment and signed a GDPR compliant informed consent form. Acceptance and usability questionnaires were completed post-test.

6. Results

Ten male volunteers, experienced motorcycle riders participated in the experiment. The volunteers that were not familiar with the CERTH motorcycle simulator had a free ride for 5 min before the experiment to familiarise themselves with the simulator. All volunteers completed the experiments.

After synchronization and subtraction of the communication and processing delays the first and second reaction of the volunteers were calculated as depicted in the figure below (red vertical line is the HMI stimuli, green line is the first reaction and blue the second reaction).

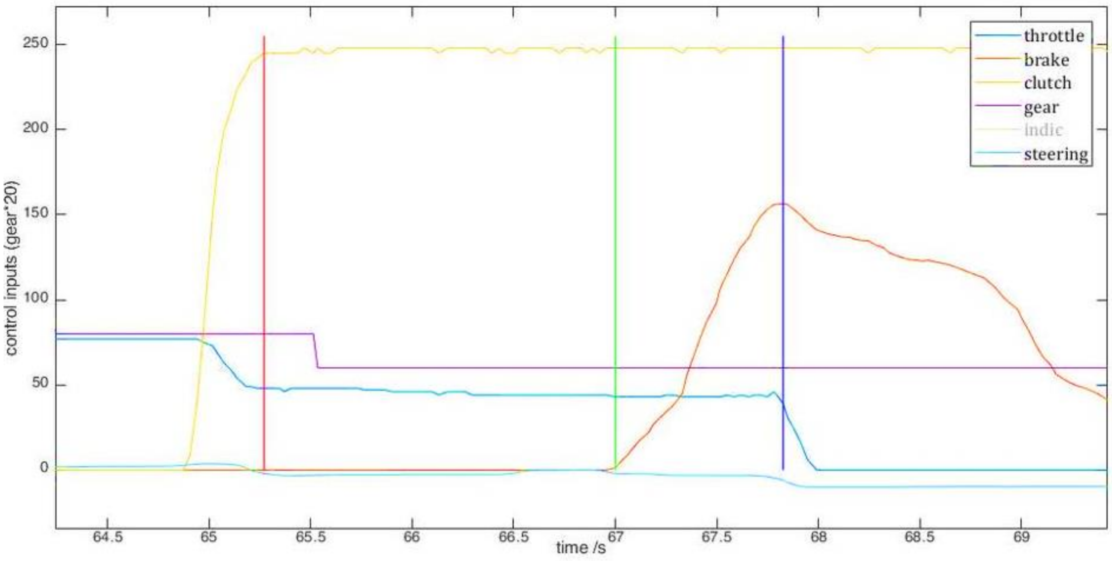

Figure 5 HMI stimulus (red vertical bar) and first (green) and second (blue) reaction of the volunteers with the motorcycle controls

The type of reaction type for each volunteer was more or less the same across the tests with most common being the throttle reduction.

In the table below the measured reaction times are presented. Reaction times above 3s were not considered as reaction to HMI input (Table 5).

Table 5 Reaction times (mean and standard deviation)

	Setup 1		**Setup 2**	
Road condition	Dry road		Wet road	
Scenario	Pedestrian	Wrong way driving	Pedestrian	Wrong way driving
Average reaction time/ s (with SD)	1.560 (0.758)	1.453(0.712)	1.654 (0.675)	1.269 (0.785)

The speed adaptation 5s after the HMI warning was issued, was analysed for all scenarios and all volunteers. The highest speed reduction was recorded at the Dry road. The steering angle after the warning had no important variation mainly due to being in a straight road scenario and not having need to avoid the obstacle/pedestrian due to early reaction.

Table 5 Speed adaptation per scenario

	Setup 1		Setup 2	
Road condition	Dry road		Wet road	
Scenario	Pedestrian	Wrong way driving	Pedestrian	Wrong way driving
Speed adaptation/ km/h (with SD)	-22.29 (13.43)	-22.89 (7.63)	-16.20 (12.36)	-21.67 (8.76)

7. Discussion and Conclusion

Several parameters of the decision logic were tuned mainly concerning the timing of the warnings and the persistence of the message on the screen. The DSS was better integrated to the cloud server and the smartphone delays were optimized. The volunteers adapted their speed in time and no virtual accident happened during the experiments. The system received mainly positive feedback from the riders. The volunteers found the audio warning more useful than the visual warning on the screen of the smartphone, and they mainly used the screen to better understand the situation.

8. Acknowledgement

The SAFE STRIP project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723211.

9. References

Bacic, M. (2005), On hardware-in-the-loop simulation. In Proceedings of the 44th IEEE Conference on Decision and Control (pp. 3194-3198). IEEE.

Gkemou, M., Spiliotis, A., & Bekiaris, E. (2018), Needs, restrictions and priorities for the development and installation of a multifunctional C-ITS solution on existing road pavement. Transportation research procedia, 36, 185-192.

Nehaoua, L., Hima, S., Arioui, H., Seguy, N., & Espié, S. (2007), Design and modelling of a new motorcycle riding simulator. In 2007 American Control Conference (pp. 176-181). IEEE.

Symeonidis, I., Nikolaou, S., Touliou, K., Gaitatzi, O., Chrysochoou, E., Bekiaris, E., Xochelli, A, Manuzzi, M, Guseo, T, Zanovello, L, Georgoulas, G. (2018), ADAS&ME: experiments for the development of a rider condition monitoring system. Forschungshefte Zweiradsicherheit, (18).

Da Lio, M., A. Mazzalai, A., Darin, M., (2018), 'Cooperative Intersection Support System Based on Mirroring Mechanisms Enacted by Bio-Inspired Layered Control Architecture', IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, pp. 1415–1429, doi: 10.1109/TITS.2017.2731901.