Driver Capability Monitoring in Highly Automated Driving: from state to capability monitoring

Joel Gonçalves, Klaus Bengler
Institute of Ergonomics
Technical University Munich
Munich, Germany
{goncalves,bengler}@lfe.mw.tum.de

Cristina Olaverri-Monreal
Innovation Systems Department
AIT Austrian Institute of Technology
Vienna, Austria
cristina.olaverri@ait.ac.at

***Abstract*— A collision probability estimator in the advent of an emergency Take Over Request (TOR) that considers the driver reaction time and the driver state is an essential tool for developing driver assistance systems for Highly Automated Driving (HAD). In this paper we present an architecture for capturing the driver state and behavior inside the vehicle. This system is then used to predict the collision probability in the situation where drivers resolve the TOR doing a keep lane maneuver (KLM) and brake to avoid the collision. Since this maneuver can be executed safely even under fast reactions, we use it as a reference to determine if is it safe for transferring the vehicle control to the driver.**

Keywords—HAD, Collision Prediction, Driver Monitoring System)

I. Introduction

Major automotive manufactures have announced the introduction of Highly Automated Driving (HAD) within commercial vehicles with the projected date of 2020. As a matter of fact, the first fleet of autonomous vehicles on roadways are expected in the near future, whose capabilities include safety-critical driving functions and monitoring roadway conditions without human intervention [1].

According to the BASt-Expert-Group definitions of vehicle automation-degrees; in full automation, the driver is not required to monitor the road [2]. As a consequence, drivers are free to engage in non-driving tasks [3].

In HAD, however, the driver does not need to permanently monitor the road conditions but in the event of a take-over request, the driver must take-over control within a certain time span [4]. In accordance with this, Take Over Request (TOR) is the process for vehicular control change, posing an abundance of challenges for the Human Machine Interface (HMI) [5] and Human Factors communities [6].

In Great Britain, the Annual Report of the Department of Transport stated that 13% of all accidents in 2013 involved human error in the form of lack of awareness or due to some form of impairment or distraction [7]. HAD promotes out-of-the-loop states [8] during which the human driver is not aware of the driving functions and roadway conditions. Introducing a new driving paradigm in HAD that enables the transference of vehicle control back to the driver, without knowing his/her grade of awareness of the road and driving situation [9] and with sufficient time to ensure an user friendly, efficient interaction and road safety, represents a considerable challenge.

A promising approach is using Driver State Monitoring Systems (DSMS) for monitoring online the driver status, enabling HMI components to support the driver. Traditional DSMS focused on having a set of normative driving performance and physiological measures, which are then compared with the current values. If these values do not fit within the expected values, then the driver is considered to be under some non-ideal condition that compromises safety. Applications range from issuing warnings to automatic emergency braking [10], but always with the goal of keeping the driver involved with the driving task.

In this paper we advance the research on DSMS focusing by proposing DSMS that meets HAD requirements. With the introduction of an automated driver and an out-of-the-loop driving experience, traditional DSMS approaches will be outdated. A new paradigm is needed to meet the expectations of HAD by enabling the human driver to focus on non driving tasks and permitting a broader set of driver states that would not be safe in manual driving conditions.

We aim to improve driving experience and safety in HAD by: 1) regulating the control transition between human and automation driver and 2) monitoring TOR process until its resolution. The first part deals with the following question: Will the driver receiving control be capable of dealing with the driving situation? We use a time-based approach for predicting, under the driver's condition, if there is enough time to safely re-engage with the driving task. An interesting application of such a model is that it can be used to adjust the automation driver behavior to the human driver status, hence meeting the HAD requirements for a less restrictive DSMS.

The second part addresses the theoretical situation in which a TOR process is ongoing, and there is a need to assess the collision risk. We monitor the driver's eye-behavior to assess if s/he identifies the other vehicles in the driving scene, and use driving performance for checking if collisions are avoided during the process. HMI can profit from this information for informing the driver of unaware vehicles.

The remainder of this paper is organized accordingly: The following section presents related work in the area of driver state monitoring systems. Section III introduces the system's architecture. Section IV presents a description of the estimation model we implemented and exemplifies the approach. Finally, Section V concludes the paper.

II. Related Work

In this section we describe state of the art DSMS. These systems are advance automotive safety systems that monitor the driver, infer its psychophysiological state, and assess the risk in the current driving scene. The system's aim is to prevent accidents.

A. Driver State

Driver states target by DSMS are sub-optimal psychophysiological states the driver is experiencing, that undermine driving performance. On the other hand, under normative states, drivers are assumed to be able to perform driving tasks with an ideal performance. DSMS are then designed to capture online measurements, compare with normative values, and assess if the driver is under a non-ideal state.

From the DSMS overview in [10], two main impairment categories are targeted for diagnosis: distraction and fatigue. Distraction encompasses inattention situations where the driver is distracted away from the driving task while engaging with "secondary" tasks [11]. This is usually associated with the driver's gaze focused away from the road center as stated in [12]. An example of a situation would be a driver interacting with an in-vehicle information system.

Fatigue [13] is a broad concept and is a natural consequence of human engagement with tasks (driving or non-driving) that over time cause the driver to show signs of drowsiness and slowly being less motivated to pursue the driving task goals. This can easily occur in situations such as during long driving and/or monotonous driving scenes.

Both driver states are very relevant for HAD research because HAD has the potential to create the conditions to provoke these impairments. By taking the focus of control from the driver, s/he will engage in either 1) monitoring the automation and the driving scenario, or 2) engaging in non-driving tasks. In the first use case, there is evidence that humans are bad as monitors for long time periods[14], and prone to be bored if there aren't many stimulants from these tasks. The second use case will also be expected as the key distinction between the HAD and the predecessor levels is precisely the ability to be distracted and truly experience a novel driving experience.

B. DSMS in Highly Automated Driving

Albeit large amounts of research has been done with DSMS, most were done in a manual driving context and do not consider HAD specifies.

HAVEit was a European Commission (EC) funded project aiming to develop and explore several levels of automation in a single integrated driving concept. The project summarizes metrics available for each automation level, highlighting the lack of driving performance metrics in HAD level [15] table 5. One of the project modules is the DSA (Driver State Assessment) module which is responsible to monitor the driver. DSA objective is to decide which level of automation is more adequate for the driver. As part of the assessment process, two driver states are addressed alertness and attention, which were assessed by drowsiness and distraction detection systems respectively. Drowsiness is detected using blink patterns, standard deviation lateral position (SDLP), and steering behavior. In this project they address distraction, particularly caused by on-board systems usage. Due to the scope of a multi-levels of automation project, no human specific factors in HAD were considered, e.g. take-over requests outcome prediction.

In [16] Zeeb et al. conducted an experiment where drivers distracted by texting and Internet searches in on-board systems, suddenly had to perform an emergency take-over request. The authors summarized their findings by validating the takeover model, and also identified the value of accurately defining and computing an adequate time for a safe TOR. However, the authors claimed that the TOR process was influenced mainly by cognitive processes and not by motor processes which can be derived by the decision of doing an on-board system interaction. It is however unlikely that if the driver is using its own device, instead of interacting with on-board systems, the TOR process would have a big motor sub-process related with the device disposal and the steering grab. Furthermore, the model proposed lacks formality to define the time boundaries, like maximum time needed for performing the TOR without colliding.

III. Architecture

Although we focus in this paper on one of the two modes we intend to implement for this framework, in this section we describe the main building blocks of our work. For each mode, there are different objectives and behavior associated to the respective situation.

The system operates in two modes: Prediction and TOR. *Prediction mode* is the default option, fulfilling the purpose of constant monitoring and predicting the driver safety if an object enters the automation sensing zone. In *TOR mode*, the system is executing the TOR process, so the objective is to track if the process is proceeding in a manner fit for a change in control. This later mode presents itself in moments when the vehicle control is shifted to the human driver and the system checks online if the driver is actually resolving the situation. The system transitions between modes when a TOR is issued, then returns back to Predict mode when the TOR is resolved. Only highway scenarios are considered.

A. Overview

The system's architecture proposed in this paper is depicted Fig. 1. The Sensor Layer, the HMI, the Automation and the Supervisor, compose the system's borders. Sensor Layer provides data input streams for the all system. For driving simulator studies we use SILAB, for eye-tracking the Facelab, and for motor responsiveness the body motion tracker Kinect.

The HMI module provides information regarding driver alertness, so appropriate feedback can be delivered to the human driver. When the human driver is not capable of handling or fails to resolve a TOR, then the Supervisor is warned. In this final scenario, the Supervisor is then responsible for forwarding the warning, in order for the automation to perform a safe maneuver. The last interface is between the Automation and the TOR Handling subsystem, this enables the system to change mode in order to handle the dynamic of a TOR.

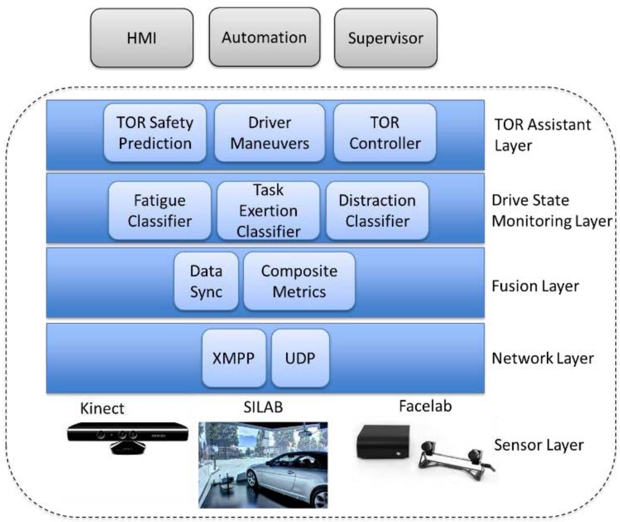

Fig. 1. System's architecture overview.

B. *Network Layer*

The Network Layer is a mix between library and infrastructure. Associated with this layer will be a library for easing the integration and easy access to the data streams in a uniform and simplified way. Regarding the infrastructure, a XMPP server is installed and it is used mainly for session data (such as login/session), while data with more real-time requirements will be send by DPU.

C. *Fusion Layer*

It is in this subsystem where the data is being actually processed by the framework. The main aim of this layer is to compute composite variables which would not be available by the default systems. For instance, a prime example of this would be the counting of micro sleeps [15] or has the driver gazed a vehicle. Data is then shared as events, as opposing as streams. This not only conveys information to that is important but also allows it to be more human understandable. For example, explaining why a fatigue detection system is issue warning could be traced back to a small number of events like: micro sleep detected, yawning detected, nodding detected.

D. *Driver State Monitoring Layer*

The DSM layer relies on the Fusion Layer to receive only their information needs. This layer acts as a current DSMS, hence focusing on diagnosing any form of perceived impairment of the driver.

The fatigue classifier is expected to be able to rely on metrics such as PERCLOS[17] or blink frequency, but also to improve the reliability with the incorporation of the Kinect ability to track posture. There are several works where actions like nodding, face scratching, etc could be very good metrics.

The distraction classifier will focus on assessing if the driver is aware of the current driving situation. Therefore, it will not only evaluate eye-tracking measurements but also head rotation and posture to assess if it is possible for the driver to pay attention to the situation.

Task exertion module is a module that will address the challenges of understanding the activities the driver is participating in. This is absolutely essential in checking if the driver is distracted in the short term, but also for predicting the effects exertion caused by engagement in a prolonged task.

E. *TOR Assistant Layer*

This layer is what distinguishes this work from the current SOTA. The intention of this layer is to create a set of services that work together to help overcome situations in HAD, namely emergency TOR.

The “driver maneuvers” is the module that assesses the driving situation and decides which maneuvers may or may not be allowed. The identification of such different decision paths is critical for understanding and presenting information to users.

Once the maneuvers have been identified, then that data is forwarded to the prediction module. This is the module that will be running most of the time. The module tries to answer questions such as if the automation would encounter an obstacle, how likely would the driver be able to handle the situation. The estimation can be particularly useful to the automation behavior in adapting the driving style to the current status of the driver.

Lastly is the TOR Controller, that is only activated when an actual TOR requests is issued. The purpose is to monitor the progress of the driver during a maneuver execution and evaluate if s/he will be successful.

IV. KLM Estimation System Model

In this section we propose definitions in developing our reasoning, and ensure a coherent taxonomy. Let’s consider the scenario represented in Fig. 2, which recreates a situation where an emergency TOR is issued by the automation and the driver has to regain control of the vehicle. Note that emergency TOR is interpreted as a TOR that is not predictable, as opposed to a TOR when leaving the highway, for instance.

Once faced with the obstacle situation, in this case a lead vehicle, the driver can perform 3 actions at the maneuver level. Within the lane change options, the driver can resolve the situation by lane change left (LCL), or lane change right (LCR). The third option is the keep lane maneuver (KLM), where the driver simply remains in the same lane.

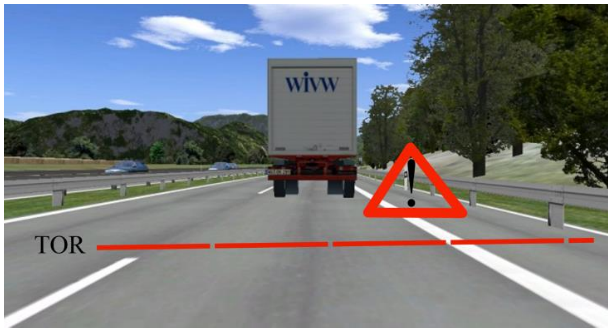

Fig. 2. Emergency TOR in a highway due to potential frontal collision with stopped object. The red line is a simbol for where the TOR will be fired. From that point on, the dirver must resolve the situation by either keep lane or changing to the left adjacent lane.

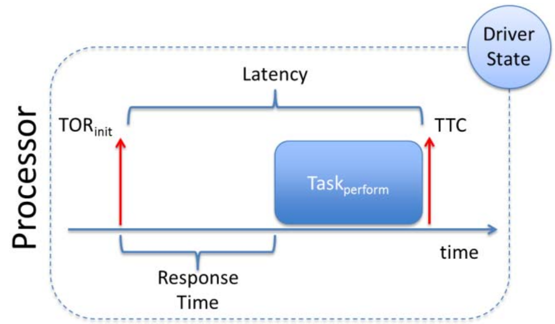

Fig. 3. Schematic representation on how a driver interacts with a specific vehicle's interface.

A. Scope

The purpose of the system here described is to perform estimations of the time required and the collision risk for and only for a KLM. The estimations are considered relevant for improving HMI during the TOR process and automation driving behavior before the TOR, respectively. These estimations are supposed to be done as soon as an obstacle is detected by the automation or a system boundary is reached in a specific position (e.g. construction site), as long as it happens in the same lane the vehicle is occupying. It is assumed that the system boundaries that trigger the system are stationary, i.e. events detected can be associated with a specific location that does not change.

Outside the scope of this system are DSMS technologies, which are assumed to exist and provide information to our system. Also, the automation driving algorithm itself is not considered; we provide assessment which can eventually influence the automation behavior, but we do not directly influence it. Furthermore, how HMI specifically interacts with the driver is also outside of the scope of this work.

B. Driver-Vehicle Interaction model

In order to estimate the outcome of a TOR, we must first formalize how the driver interacts with the vehicle in order to complete all tasks associated with a maneuver.Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

Fig. 3 shows the modeling approach for the driver-vehicle interaction. Consider a set of processors $P = \{\rho_1, \dots, \rho_n\}$, which represent the human driver's means of interacting with the vehicle. Example of such processors would be the feet or the hands. Each processor can execute a maximum of one task per time.

Let $T = \{\tau_1, \dots, \tau_m\}$ a set containing all relevant tasks related to driving the vehicle. Example of tasks are steering or pressing the brake pedal. A processor can queue tasks to process individually. Each task is associated with one and only one processor.

A task is characterized by $\tau = (T_{resp}, T_{perf})$. T_{resp} is associated with the Response Time, and T_{perf} to the Task Perform represented in Fig. 3. Response time models the time once the TOR was issued until the processor actually is performing the task. Situations like being distracted and then correct posture before actually engaging with the steering wheel and pedals, etc.… The specific definition of the Response Time element is the tuple $T_{resp} = (\Sigma, (\mu_{resp}, \sigma^2_{resp}))$. This corresponds to a map function that given a driver state, it returns a normal distribution for this task's response time. $\Sigma \subset S$, set of all driver states observable by the DSMS. The T_{perf} is equivalent but for the actual performance time. The final task characteristic is the latency, as depicted in Fig. 3, this being the addition of the $T_{resp}, and\ T_{perf}$. Note that the distribution on dependent from the driver state.

C. Output

Once a system boundary situation is detected, the most important prediction for KLM is:

- Does the driver has the time required to resolve the TOR such that the collision probability is acceptable?

We formulate this problem in the following format:

$$\begin{cases} \mathrm{T}_{completion} = \max(\tau_{latency}) + brake_{latency} \\ P(\mathrm{T}_{completion} < TTC_0) \geq safe\_threshold \end{cases}$$

The resolution has two main steps. In the first part, we consider the most time consuming task ($\max_{i=1..n}(\tau^i_{latency})$) and the brake time ($brake_{latency}$) needed to stop the car in order to obtain a probabilistic model $\mathrm{T}_{completion}$ of the amount of time the driver needs to stop the vehicle since the TOR.

In the rest of this chapter, we detail how to compute each component of those equations.

D. KLM Specification

For the purpose of this paper, a maneuver is initiated with a set of tasks that once all executed, the maneuver is complete. The KLM is an interesting maneuver because on the one hand it eases the calculations as one must only consider the reference in front (usually an obstacle) for the calculations. On the other hand, all other maneuvers, when performed adequately, they require a significant more tasks to perform (e.g. check mirror, evaluate gap). So KLM can be seen as a quick reactive action that leads to a safe resolution, even if not the most ideal resolution.

With that in mind, we consider that in order to safely perform the KLM, drivers must: 1) use the steering wheel for lateral control, 2) brake pedal for longitudinal control, and 3) look to the windshield for spotting the reference point.

All tasks considered can be performed in parallel, and each processor only executes one task. In this situation, the maximum task latency is given by the longest task of all. Once we choose the normal distribution for representing the response time, then the last parameter for computing the $\mathrm{T}_{completion}$ is the $brake_{latency}$. This is vehicle dependent, however an approximation can be done by:

$$brake_{latency} = {}^{-v_0}/_{a_{max}}$$

Where v_0 is the initial velocity and the a_{max} the maximum deceleration possible when full pedal press.

E. Collision Probability

The final step to complete our prediction is to calculate the probability of stopping the car under the driver's reaction time, before the time-to-collision then the TOR is issued (TTC_0).

The collision probability is given by the cumulative distribution function:

$$P(\mathrm{T}_{completion} < TTC_0) = \frac{1}{\sigma_c\sqrt{2\pi}} \int_0^{\mathrm{TTC}_0} e^{-\frac{1}{2}\left(\frac{t-\mu_c}{\sigma_c}\right)^2} ts$$

Where $\mathrm{T}_{completion} = N(\mu_c, \sigma_c^2)$.

V. CONCLUSION AND FUTURE WORK

Emergency TOR will compose many challenges for researchers in order to ensure the driver safety in these complex situations. We consider the KLM as the reference of the most feasible maneuver that can be performed, such that if a TOR is issued and the KLM is no longer feasible then the vehicle control should not be transferred to the driver.

Future work will be divided in two directions. First collect empirical data for synthesize the normal distributions where the reaction time depends on different driver states. The second direction is to move from the prediction, to the real-time tracking and control of the TOR process.

ACKNOWLEDGMENT

This work was supported by the HF-AUTO: Human Factors of Highly automated Driving (PITN–GA–2013–605817), project funded by the European Commission within the Innovative Training Networks (ITN), a funding scheme under Horizon 2020.

REFERENCES

[1] T. E. Trimble, R. Bishop, J. F. Morgan, and M. Blancho, "Human factors evaluation of level 2 and level 3 automated driving concepts: Past research, state of automation technology, and emerging system concepts," 2014.

[2] T. M. Gasser and D. Westhoff, "BASt-study: Definitions of Automation and Legal Issues in Germany," 2012.

[3] N. Merat, a. H. Jamson, F. C. H. Lai, and O. Carsten, "Highly Automated Driving, Secondary Task Performance, and Driver State," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 54, no. 5, pp. 762–771, 2012.

[4] C. Gold, D. Damböck, L. Lorenz, and K. Bengler, "'Take over!' How long does it take to get the driver back into the loop?," in *Proceedings of the Human Factors and Ergonomics Society 57th Annual Meeting*, 2013, pp. 1938–1942.

[5] F. Flemisch, A. Schieben, N. Schoemig, M. Strauss, S. Lueke, and A. Heyden, "Design of Human Computer Interfaces for Highly Automated Vehicles in the EU-Project HAVEit," in *Universal Access in Human-Computer Interaction. Context Diversity*, Springer Berlin Heidelberg, 2011, pp. 270–279.

[6] J. C. F. de Winter, R. Happee, M. H. Martens, and N. A. Stanton, "Effects of adaptive cruise control and highly automated driving on workload and situation awareness: A review of the empirical evidence," *Transp. Res. Part F Traffic Psychol. Behav.*, Aug. 2014.

[7] F. Graves, D. Lloyd, D. Wilson, D. Mais, W. Deda, and A. Bhagat, "Reported Road Casualties Great Britain : 2013 Annual Report," 2014.

[8] N. Merat and J. D. Lee, "Preface to the Special Section on Human Factors and Automation in Vehicles: Designing Highly Automated Vehicles With the Driver in Mind," *Hum. Factors J. Hum. Factors Ergon. Soc.*, vol. 54, no. 5, pp. 681–686, Oct. 2012.

[9] M. Saffarian, J. C. F. de Winter, and R. Happee, "Automated Driving: Human-Factors Issues and Design Solutions," *Proc. Hum. Factors Ergon. Soc. Annu. Meet.*, vol. 56, no. 1, pp. 2296–2300, Oct. 2012.

[10] Y. Dong, Z. Hu, K. Uchimura, and N. Murayama, "Driver inattention monitoring system for intelligent vehicles: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 596–614, Jun. 2011.

[11] T. W. Victor, J. L. Harbluk, and J. a. Engström, "Sensitivity of eye-movement measures to in-vehicle task difficulty," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 8, no. 2 SPEC. ISS., pp. 167–190, 2005.

[12] C. Olaverri-Monreal, A. E. Hasan, J. Bulut, M. Korber, and K. Bengler, "Impact of In-Vehicle Displays Location Preferences on Drivers' Performance and Gaze.," *IEEE Trans. Intell. Transp. Syst. (IEEE T-ITS)*, vol. 15, no. 4, pp. 1770–1780, 2014.

[13] R. O. Phillips, "A review of definitions of fatigue – And a step towards a whole definition," *Transp. Res. Part F Traffic Psychol. Behav.*, vol. 29, pp. 48–56, Feb. 2015.

[14] J. Rogé, T. Pebayle, and A. Muzet, "Variations of the level of vigilance and of behavioural activities during simulated automobile driving," *Accid. Anal. Prev.*, vol. 33, no. 2, pp. 181–186, Mar. 2001.

[15] R. Hoeger, H. Zeng, A. Hoess, T. Kranz, S. Boverie, M. Strauss, E. Jakobsson, A. Beutner, A. Bartels, T.-B. To, H. Stratil, K. Fürstenberg, F. Ahlers, E. Frey, A. Schieben, H. Mosebach, F. Flemisch, A. Dufaux, D. Manetti, A. Amditis, I. Mantzouranis, H. Lepke, Z. Szalay, B. Szabo, P. Luithardt, M. Gutknecht, N. Schoemig, A. Kaussner, F. Nashashibi,

P. Resende, B. Vanholme, S. Glaser, P. Allemann, F. Seglö, and A. Nilsson, "HAVEit Final Report," 2011.

[16] K. Zeeb, A. Buchner, and M. Schrauf, "What determines the take-over time? An integrated model approach of driver take-over after automated driving," *Accid. Anal. Prev.*, vol. 78, no. May 2015, pp. 212–221, 2015.

[17] D. Sommer and M. Golz, "Evaluation of PERCLOS based current fatigue monitoring technologies," *2010 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC'10*, pp. 4456–4459, 2010.