Automatic and Manual Driving Paradigms: Cost-Efficient Mobile Application for the Assessment of Driver Inattentiveness and Detection of Road Conditions

Arman Allamehzadeh and Cristina Olaverri-Monreal* *Member, IEEE*

Abstract—**Assessment of the driver's state and the driving environment is essential in promoting road safety in both manual and automatic driving paradigms where the monitoring tasks are either performed by the driver or by the system. Within this work, we present a cost effective mobile application to measure gaze behavior and analyze road conditions for a request to take vehicle's control in case of an automatic driving or to avoid inattentive driving in a manual driving paradigm. We evaluated the application under daylight conditions. Results showed a high rate of detections in a short period of time.**

I. INTRODUCTION

The association for safe international road travel states that road crashes account for 2.2 per cent of all deaths globally and predicts that road injuries will become the fifth leading cause of death by 2030 [1]. As reported in 2013 by the International Organization for Road Accident Prevention [2], human error accounts for 90 percent of the road accidents. Over the years, different measures have been investigated to reduce the rate of accidents due to human errors. The introduction of autonomous vehicles represent an opportunity to continue working for increased road safety, namely because the automation will be in charge of driving sub tasks such as navigation, guidance and control that have been so far performed by the driver and whose intervention will not be required anymore.

According to the international levels of driving automation classification for on road vehicles of SAE [3], in conditional and high automation steering, acceleration, deceleration and the task of monitoring the driving environment are performed by the automated driving system. A response from the driver to a Take Over Request (TOR) in case of an unexpected road situation is expected in conditional automation. In high automation however, the automated driving system maintains the vehicle's control depending on the driver's response to the request.

Several studies have shown that drowsiness and hypo-vigilance while manually operating a vehicle frequently occur during highway driving, and might be responsible for serious road accidents [4]. A long time driving under same road conditions contributes to a decrease of driving workload and a consequent driver situation awareness reduction. This hypo vigilance also occurs in highly automated driving and it has to be taken into account when a vehicle control is expected from the driver.

Driver State Monitoring Systems (DSMS) are automatically triggered when the driver's attention is taken away from the road for a period of time that is deemed unsafe. In high automation driving, DSMS play a crucial role to determine whether or not the driver is prepared to take control of the vehicle.

In this paper, we focus on monitoring the driver's attention to the road through a context aware system relying on the sensor technology that modern smart devices provide. We develop a cost effective application that can be used in a manual and high automation driving paradigm and determines the visual attention of the driver based on the gaze location, according to [5]. The application additionally measures the response to a conversation started by an implemented evaluator function. Moreover, as the current technology used to detect construction zones bases on the online acquisition data from maps and databases related to roads under construction [6], our application is particularly useful for detecting objects that reflect information not available in maps due to their temporary nature (such as construction zones). The cost-effective, ubiquitous use of our approach makes it particularly appropriate to be used in every vehicle independently of the number of automatic features included in it.

The remainder of this paper is organized accordingly: The following section presents related work in the areas of driving assistant systems and cost efficient mobile applications. Section III describes the application requirements and functioning. Section IV describes the driving paradigms addressed. Sections V and VI describe the monitoring processes for the road and the driver. Sections VII presents the evaluation results and section VIII concludes the paper.

II. RELATED WORK

In a vehicular context, some applications intend to decrease the risk of drowsiness in a road context using lights [7] or sounds and vibrations [8] to keep the driver awake. Others demand driver interaction to measure the reaction time to respond to a sound stimuli [9].

Recent approaches take advantage of Advanced Driving Assistance Systems features for obstacle detection and collision avoidance based on radar, camera, and vehicle sensors and include driver monitoring based on eye-tracking and driver

[1] University of Applied Sciences Technikum Wien, Department of Information Engineering and Security, Höchstädtplatz 6, 1200 Vienna, Austria. arman4u@gmail.com, olaverri@technikum-wien.at
* Corresponding author

 DOI: 10.1109/IVS.2016.7535359

performance systems. In this context, an automated driver-assistance system was proposed in [10]. The system responded to driver's actions and at the same time determined the driver's observations and inattentiveness based on driver's eye gaze.

Smart phones have also recently been used to monitor the driver using cameras to estimate the eye gaze direction as an indicator of driver attention. The authors in [11] processed several input video frames and used facial characteristics to build a set of Support Vector Machines (SVMs) to detect and localize the face, and estimate head pose relying on the methodology in [12]. In a further application both road and driver were monitored [13] to detect distracting and driving behavior in real time (i.e. drowsy driving, tailgating). The application used embedded sensors such as inertial sensors to take into account potential blind spots.

The system we propose, scans the road via the back camera to gain knowledge about potential construction zones that are marked through orange traffic cones. As the authors in [13] pointed out in their work, current smart devices are still not capable of processing streams from both the front and rear cameras simultaneously. Therefore, our application also switches between both the front and rear phone camera thanks to a context-aware algorithm focusing on both, manual and automatic driving paradigms.

In a computer vision context, the estimation of the orientation of head pose affects the ability to characterize the gaze direction [14]. To this end, we perform a characteristic analysis of the driver's head orientation and face and combine several approaches in order to: a) monitoring driver behavior and b) avoiding inattentive driving conducts through feedback about unsafe driving actions as proposed in [15]. In the next sections we describe the system architecture of our tool based on the Android operative system (OS).

III. System Requirements and Functioning

The application uses Android SDK API Jellybeans version 4.1.x. It supports the front and back camera of smart devices to acquire the required data. The accelerometer and GPS sensors integrated in these devices are used to make sure that the application is only monitoring the driver when the vehicle is in motion in order to avoid annoying situations in which the system warns the driver when the vehicle is stopped (e.g. waiting for a red traffic light to turn green).

After the extraction of data from the frames and relying on computer vision techniques to process the images detected, the system informs the driver by different output methods (i.e. a beep sound, speech) in the event that a target has been detected on the road. Furthermore, the system allows the user to easily configure the app in order to meet his/her expectations for manual or automatic driving modus by assigning the required variables with the most proper values for each particular case.

The application requires the mobile device to be located in front of the driver with a good holder and right alignment at a distance lower than 55 cm to ensure an accurate calibration in the area where both eyes are going to be tracked. Figure 1 shows the set up in the vehicle with the mounted smartphone

Fig. 1. In-vehicle location of the smart phone for a smooth face characteristics detection

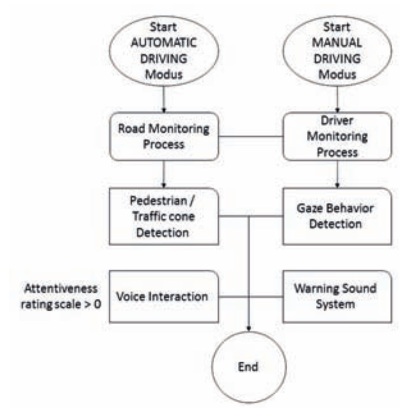

Fig. 2. Flowchart for the automatic and manual driving paradigms

in which the driver's face and eyes have been detected. In a similar way to the work presented in [16] our application did not require high accuracy or robust tracking, so gaze tracking was performed using the iris. For the application development we aimed for a separate layered structure. We monitored the driver through the detection of face and eyes and the road situation through the detection of construction zones and pedestrians. Orange traffic cones were targeted for detecting construction zones because of their extended use (also for marking road sections that require special attention by the driver) and easy placement on roads.

IV. Driving Paradigms

Depending on the selection the driver has made in the user interface regarding an automatic or a manual driving, are the road or the driver monitoring processes activated by default. The flowchart in figure 2 illustrates the process. The options in the submenus enable the activation of additional features.

A. Automatic Driving Paradigm

As previously mentioned, in vehicles with a degree of high autonomy the driver does not need to permanently monitor the road conditions. However, challenges for the Human Machine Interface [17] and Human Factors communities [18], [19]

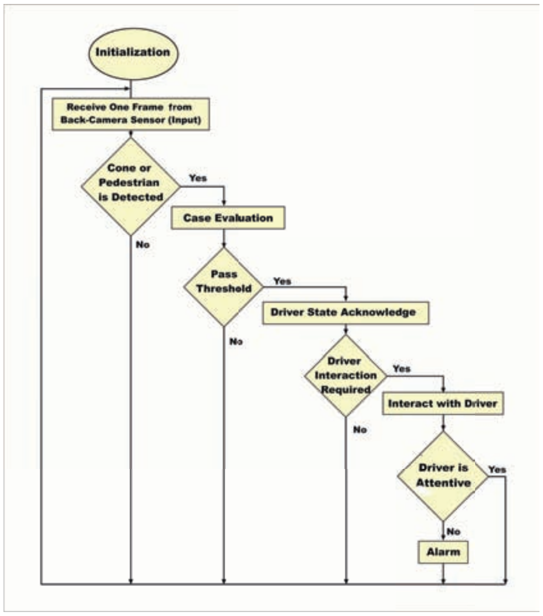

Fig. 3. Flowchart for the Automatic Driving Paradigm

are derived from required vehicular control changes (Take Over Request). In order to contribute to the efficiency of the changing process, our application distinguish two use cases, a road scenario with a construction zone and a pedestrian on the road that require this control change. In case an object has been detected on the road and the system evaluates that the road situation requires the attention of the driver and even it was deemed appropriate to trigger a TOR, the system monitors the driver attentiveness to the road and verifies the driver's status. To assess the road situation the number of detected objects in last 3 seconds was considered to avoid noise or false positives. The followed procedure is described below and illustrated in Figure 3.

- Scanning the road via the back camera to gain knowledge about the road environment. Section V describes the process.
- Determining the visual attention of the driver based on a) his/her gaze location, and b) response to a succession of words uttered by the system's evaluator function. Section VI describes the process in detail.

B. Manual Driving Paradigm

In order to detect the driver's visual attention to the road in a manual driving modus using the mobile application, we applied the computer vision techniques for image processing and extraction of data explained in Section VI and assessed the driver status by the number of detected eyes for each frame. An alarm was triggered if face and eyes were not detected by the system in a time frame that exceeded 2 seconds, according to the eyes-off-road time recommended by the NHTSA for glances away from the roadway [20]. This eye monitoring process corresponded to the last 25 continuous frames. The system enables the change of the configuration to a shorter time from 2 seconds to 1 second if the user wishes a stronger monitoring. In this case, the driver can activate the "Sensitive" button.

V. Road Monitoring Process

The road monitoring process is activated by default as first when the automatic driving option has been selected. To detect pedestrians and/or cones, the back camera of the mobile device obtained the current frame from the road and transmitted it to the processor. This road monitoring process was constantly applied until the aimed object pedestrian or orange cone was detected. Once the object is detected the driver attention to the road will be evaluated. To reduce the noise/false detections, the number of detections occurred in the last 3 seconds was stored and recognized as real detections if this number was higher than 10. The algorithm we used for pedestrian and cone detection based on a classifier with multiple stages of filters (Cascade) working with Haar-Like features. It consisted of a detection window that slided around the whole image to detect the target object. If the input region fails to pass the threshold of a stage, the cascade classifier will reject the region. Otherwise, it will be classified as a potential target object and further processing will be applied.

A. Pedestrians detection

Pedestrians were detected building upon the haarcascade_fullbody classifier [21] confining the search area by removing 15% from the left and right sides (sidewalk) and 20% from up and down (sky and dashboard). The classifier is designed so that it can be easily resized in order to be able to find the objects of interest at different sizes, which is more efficient than resizing the image itself. We set the sliding window with 5% of the frames width and 10% of frames height for the minimum size of any pedestrian in the image. Although this small sliding window requires more detection time and reduces the frame rate to 4 fps, it increases the range of detection. For example, a driver in a vehicle at a speed of 50 km/h that uses our system an detects a person at a distance of 40 meters, will have around 3 seconds to react. Equation 1 shows the calculation of the ROI for road monitoring.

$$\begin{aligned} W_{interest} = W_{total} - W_{reduced} = 100\% - (15\%(from\ left) \\ +(15\%(from\ right)) = 70\% \\ H_{interest} = H_{total} - H_{reduced} = 100\% - (20\%(from\ top)+ \\ (20\%(from\ bottom)) = 60\% \end{aligned} \tag{1}$$

where:
total area= $Width(W) \times Height(H) = 100\%$
interest area= $W_{interest} \times H_{interest} = 42\%$
reduced area= $100\% - 42\% = 58\%$

B. Orange Traffic Cones Detection

We built our own classifier to detect the orange cones for the construction zone. To this end, we first collected samples

Fig. 4. Pedestrian (A) and cone (B) detected by the application.

of images for orange cones by recording several short videos (approx. 120 seconds) of construction zones in Vienna that showed the objects from different perspectives. Afterwards, we developed an OpenCV application (in VC++.NET) to extract images from the videos and then manually verify if the images were appropriate. In case of a similarity of an image with the previous one, it was deleted. We then scaled all the images to the same size to extract the objects features in form of vectors of measurements. The total number of images consisted of 125 positive samples. The same method was applied to classify the negative samples (414 images). Finally, relying on [22] we created a classifier that we trained using the images collection. Figure 4 shows images of pedestrians and traffic orange cones detected in Vienna by the application.

VI. Driver Monitoring Process

The driver monitoring process is activated by default as first when the manual driving option has been selected in the user interface. If the driver's attention is taken away from the road for a period of time that is estimated unsafe a warning is automatically triggered. To monitor the driver attentiveness to the road, we relied on two methods: the characteristic analysis of driver's face to measure gaze behavior and the response to a conversation started by the system based on speech processing techniques. As previously mentioned the driver monitoring is activated in the automatic modus after the system has determined from the road monitoring process that a TOR might be necessary. Details of the functioning in each case are described below.

A. Gaze Behavior Detection

To determine the driver status regarding the gaze location, the application used the front camera of the mobile device to locate the driver's face position. After the frame reception, the application analyses if the driver's face was detected. Detection only occurs when the driver's face is located in front of the camera and his/her gaze is directed to the road.

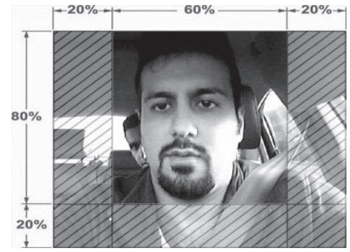

Fig. 5. Original frame area with the sections removed and face detected in the reduced area

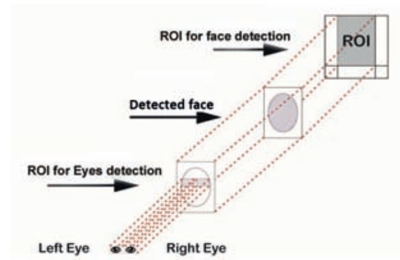

Fig. 6. Calculation of the Region Of Interest for face and eyes detection

To obtain the required classifiers to detect the face and eyes of the driver, we applied the OpenCV library for object detection using Haar Feature-based Cascade Classifiers relying on the original and improved versions of the face and eye detection algorithm by [23], [24], [25].

To detect the face we extracted an approximate area for the face to be located. We then removed 40% of the frame (20% on the right and 20% on the left side) and 20% of its height starting from the bottom part of the original frame. Afterwards we searched for the face in this reduced area. Figure 5 shows the original frame with the sections removed and the face detected in the reduced area.

After the face was detected, a function was applied to detect the right and left eyes. Again, we estimated a Region Of Interest (ROI) within the general face geometry focusing only on this area instead of searching the whole frame in order to reduce the system resource usage. This area reduction enabled us to efficiently detect both eyes and within them, in the reduced area, their iris through a function that distinguished a circular structure in the center of the eye. Figure 6 illustrates the extraction process for the calculation of the ROI for face and eyes detection.

The detection process (Figure 7) occurs for a specific period of time during which the driver's face and eyes are monitored by an attentiveness rating system. The rating scale ranges from 50 to 0 points starting with the maximal value and decreasing depending on the grade of the attentiveness detected. The score decreases one point every time that the front camera detects the face. To rate a detection of both, face and eyes, the score decreases one additional point. In case, neither the face nor the eyes are detected the score is incremented by one. When the maximal attentiveness has been reached (score = 0), the process starts monitoring the road again. If the score is higher

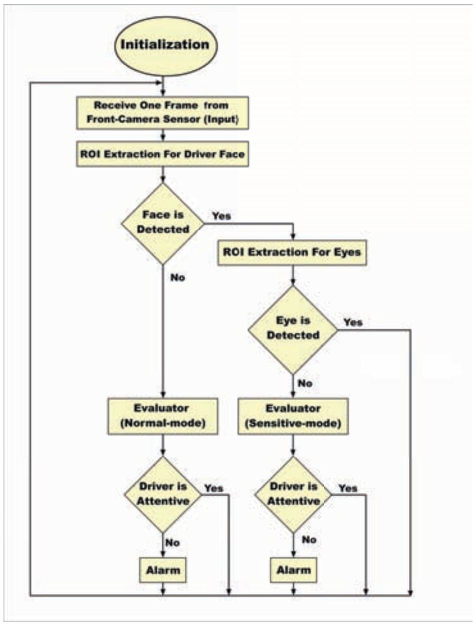

Fig. 7. Flowchart for the driver monitoring process

than 0 the system warns the driver so that his/her attention is being focused in the road and the eyes can be detected again.

B. Voice Interaction

As illustrated in the flowchart for automatic driving in Figure 3 voice interaction is activated when the road monitoring conditions require the driver to take over the control of the vehicle. In this case, to monitor the driver attentiveness to the road, the system started a conversation with the driver by asking aloud: "Are you looking at the road?". At the same time the question was displayed on the smart device screen and the system waited until a voice answer was detected from the driver. The language used was English. The answer was then converted into the text and analyzed via a linguistic function.

VII. System Evaluation Results

To evaluate the performance of our implemented application regarding object detection rate, accuracy and time, we performed several tests with a vehicle and 3 drivers under daylight conditions with two different Android Samsung smartphones: a) one Samsung Galaxy Fame (480p@25fps Primary=5 MP, Secondary=VGA) "a" and b) one Samsung Galaxy A5 with higher image processing performance and better camera resolution (1080p@30fps Primary=13 MP, Secondary=5MP) "b".

A. Cones detection rate

We tested 47 different scenarios that consisted of a variety of cone sizes in a detection range from 5 to 15 meters. The total number of detections with device "b" (154) was higher than with device "a" (146). Device "a" detected 54.1% of all the cones with a rate of 11.3% false detections or system classification of other objects as cones and device "b" detected

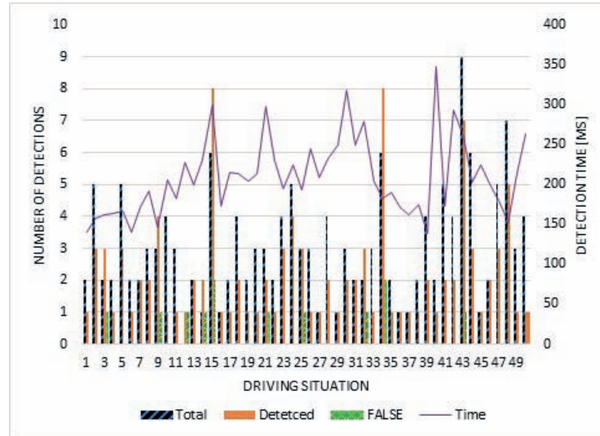

Fig. 8. Orange traffic cones detection rate and time for device "b"

70.7% of the cones total number with a rate of 11% of false detections

Per test case, the device with better image processing performance "b" detected more orange traffic cones in total (max. 8), but device "a" detected at least one cone in all the situations (max. 3) as well. Figure 8 shows the detection time and rate with device "b" for each individual case
The detection time ranged from 138 to 347ms, being the average detection time of 209.2 milliseconds.

B. Pedestrians detection rate

For the evaluation of the pedestrians detection algorithm, we tested 87 different scenarios with a maximum number of 11 pedestrians per scenario in a detection range from 5 to 40 meters. In the previous section we showed that the detection rate was higher when performed with device "b". Therefore, we focus in this section on results obtained with this device. 70.3% of all the pedestrian were detected and the rate of false detections constituted 16.1%. A reduction in the detection time could be appreciated depending on the CPU power. In each round a total of maximum 9 pedestrians were detected. The detection time ranged from 109 to 305 ms, being the average detection time of 185.3 milliseconds.

C. Face and Eyes Detection Rate

The average of the face detection time for drivers 1, 2 and 3 were 6.1ms, 7ms, and 5.6ms respectively. We could detect some outliers in the detection rate that we associated with external processes of the OS. Regarding the detection rate of the eyes in the 3 drivers, results ranged from 4 to 18 milliseconds and almost did not differ among the drivers. Remarkable were the differences between the detection rate depending on the device used. As shown in Figure 9, device b required more time to detect the eyes.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we proposed an application to monitor driver behavior and assess road conditions for an automatic and

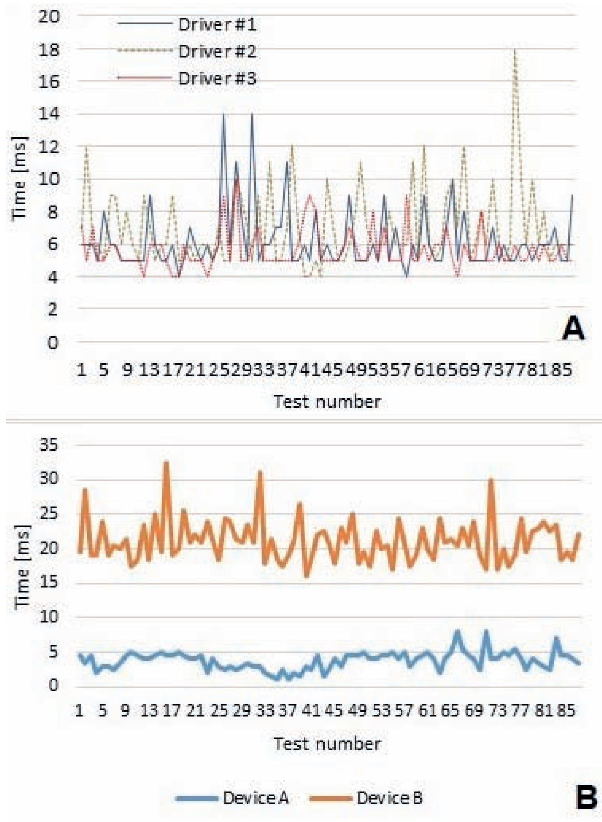

Fig. 9. Face detection time with device "b" (A) and eye detection time for driver 1 (B)

manual driving paradigms using mobile smart devices. Evaluation results showed a high object detection rate within few milliseconds of time. As we previously mentioned the detection time differed depending on the device used. This was due to the higher resolution of device b that needed more time to process the higher number of pixels and detected more objects. A device with a lower camera quality detected the target objects as well. Therefore, a good camera is not required to apply the approach proposed in this paper. The number of false object detections, could be reduced by training the cone classifier with more samples. Future work will include the storage of the detected construction zones in an online database that will elicit geographical information of a given construction zone location in real-time.

ACKNOWLEDGMENTS

This work was supported by the KiTSmart Project - City of Vienna Competence Team for Intelligent Technologies in Smart Cities, funded by national funds through the MA 23, Urban Administration for Economy, Work and Statistics, Vienna, Austria.

REFERENCES

[1] "Annual global road crash statistics," http://asirt.org/initiatives/informing-road-users/road-safety-facts/road-crash-statistics .

[2] NHTSA, "Traffic safety facts, research note."

[3] "International levels of driving automation for on road vehicles," http://www.sae.org/misc/pdfs/automateddriving.pdf/.

[4] P. Thiffault and J. Bergeron, "Monotony of road environment and driver fatigue: a simulator study," *Accident Analysis & Prevention*, vol. 35, no. 3, pp. 381–391, 2003.

[5] H. Deubel and W. X. Schneider, "Saccade target selection and object recognition: Evidence for a common attentional mechanism," *Vision research*, vol. 36, no. 12, pp. 1827–1837, 1996.

[6] "Key:construction," http://wiki.openstreetmap.org/wiki/Key:construction.

[7] "Antisommeil mobile application," http://play.google.com/store/apps/details?id=com.antisommeil .

[8] "Sleep driving alarm," https://play.google.com/store/apps/details?id=com.pumpkintech.sleepdrivingalarm .

[9] "Antisleep," https://play.google.com/store/apps/details?id=em.software.antisleep .

[10] L. Fletcher and A. Zelinsky, "Driver inattention detection based on eye gazeroad event correlation," *The international journal of robotics research*, vol. 28, no. 6, pp. 774–801, 2009.

[11] M.-C. Chuang, R. Bala, E. Bernal, P. Paul, and A. Burry, "Estimating gaze direction of vehicle drivers using a smartphone camera," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 165–170.

[12] J. Ng and S. Gong, "Composite support vector machines for detection of faces across views and pose estimation," *Image and Vision Computing*, vol. 20, no. 5, pp. 359–368, 2002.

[13] C.-W. You, N. D. Lane, F. Chen, R. Wang, Z. Chen, T. J. Bao, M. Montes-de Oca, Y. Cheng, M. Lin, L. Torresani *et al.*, "Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones," in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM, 2013, pp. 13–26.

[14] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation in computer vision: A survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 4, pp. 607–626, 2009.

[15] J. Gonçalves, R. J. Rossetti, and C. Olaverri-Monreal, "IC-DEEP: A serious games based application to assess the ergonomics of in-vehicle information systems," in *15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2012, pp. 1809–1814.

[16] C. Olaverri-Monreal, A. E. Hasan, J. Bulut, M. Körber, and K. Bengler, "Impact of in-vehicle displays location preferences on drivers' performance and gaze," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 4, pp. 1770–1780, 2014.

[17] 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, 2011, pp. 270–279.

[18] F. Naujoks, C. Mai, and A. Neukum, "The effect of urgency of take-over requests during highly automated driving under distraction conditions," *Advances in Human Aspects of Transportation: Part I*, vol. 7, p. 431, 2014.

[19] J. Gonçalves, C. Olaverri-Monreal, and K. Bengler, "Driver capability monitoring in highly automated driving: Evolving from state to capability monitoring," in *18th International IEEE Conference on Intelligent Transportation Systems, Las Palmas de Gran Canaria, Spain. September 2015 (ITSC)*, 2015, pp. 2329–2333.

[20] N. H. T. S. Administration *et al.*, "Visual-manual nhtsa driver distraction guidelines for in-vehicle electronic devices," *Washington, DC: National Highway Traffic Safety Administration (NHTSA), Department of Transportation (DOT)*, 2012.

[21] "Haarcascadefullbody," https://github.com/Itseez/opencv/tree/master/data/haarcascades.

[22] M. Rezaei, "Creating a cascade of haarlike classifiers Step by step," 2014, https://www.cs.auckland.ac.nz/ m.rezaei/ Tutorials/Creating_a_Cascade// _of_HaarLike_Classifiers_Step_by_Step.pdf.

[23] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–511.

[24] Q. Li, U. Niaz, and B. Merialdo, "An improved algorithm on viola-jones object detector," in *Content-Based Multimedia Indexing (CBMI), 2012 10th International Workshop on*. IEEE, 2012, pp. 1–6.

[25] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 1. IEEE, 2002, pp. I–900.