Integrating OpenFace 2.0 Toolkit for Driver Attention Estimation in Challenging Accidental Scenarios

Javier Araluce, Luis M. Bergasa, Carlos Gómez-Huélamo, Rafael Barea, Elena López-Guillén, Felipe Arango, Oscar Pérez

Electronics Department, University of Alcalá (UAH), Spain,
{javier.araluce, luism.bergasa, rafael.barea,
elena.lopezg}@uah.es, {carlos.gomezh, juanfelipe.arango ,
oscar.perez}@edu.uah.es

Abstract. Gaze estimation is a typical approach to monitor the driver attention on the road scene. This indicator is of great importance in safe driving and in the design of the takeover control strategy for a Level 3 and Level 4 automation system. Nowadays, most of eye gaze tracking techniques are intrusive and costly, which limits their applicability over real vehicles. On the other hand, current databases used for gaze validation face the driver attention task focused on critical situations in simulation but they do not encounter actual accidents. This paper presents a low-cost and non-intrusive camera-based gaze mapping system integrating the open-source state-of-the art OpenFace 2.0 Toolkit [1] to visualise the driver attention simulation on prerecorded real traffic scenes through a heat map. The proposal has been validated by using the recent and challenging public dataset DADA2000 [2] which has 2000 video sequences with annotated driving scenarios based on real accidents. We compare our system with an expensive desktop-mounted eye-tracker, obtaining on par results and showing it is a good tool for driver attention monitoring able to be used in the design of take over systems and driving scenarios awareness systems for automated vehicles.

Keywords: Driver attention, accidental scenarios, gaze estimation, heat map, computer vision.

1 INTRODUCTION

In last years, important advances have been made in autonomous driving field from an academic and industrial point of view. According to SAE (J3016), five Levels of Automation can be applied, achieving the full automation in the Level 5. Nowadays, we have technologies that can be used to take over the functions normally reserved for the driver. In Level 1 and Level 2 the driver is still supposed to be fully engaged in supervising the actions of the vehicle under all circumstances. In Level 3 and Level 4, the driver is freed from supervision, either in limited situations or during the entire trip. Problems arise at the above levels, because the driver is in-the-loop and not always aware of what is happening (Level 1 and Level 2) or is out of the loop and needs quickly to be brought back into the loop for some unexpected reason (Level 3 and Level 4) [3].

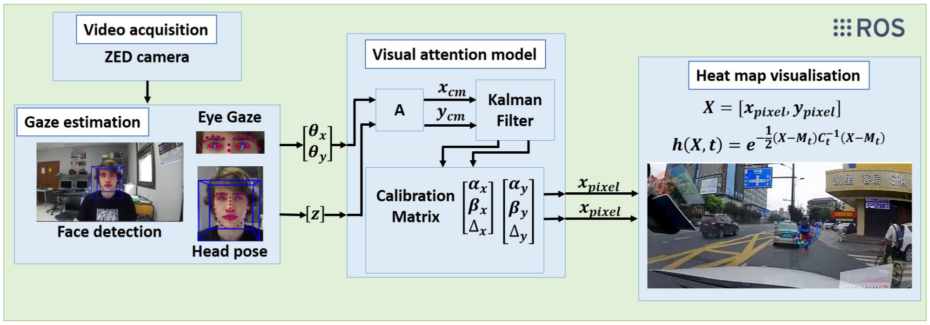

ROS
Video acquisition
ZED camera
Gaze estimation
Eye Gaze
Face detection
Head pose
$\begin{bmatrix}\theta_x\\ \theta_y\end{bmatrix}$
$[z]$
Visual attention model
A
x_{cm}
y_{cm}
Kalman Filter
Calibration Matrix
$\begin{bmatrix}\alpha_x\\ \beta_x\\ \Delta_x\end{bmatrix}$ $\begin{bmatrix}\alpha_y\\ \beta_y\\ \Delta_y\end{bmatrix}$
x_{pixel}
x_{pixel}
Heat map visualisation
$X = [x_{pixel}, y_{pixel}]$
$h(X,t) = e^{-\frac{1}{2}(X-M_t)C_t^{-1}(X-M_t)}$

Fig. 1: General Architecture.

In this context of shared control between human and machine, driving in a safety way requires being ensured that the state of driver is suitable for driving. This is particularly important when the vehicle requests the driver intervenes in case the driving scenario is complex. In addition, it is crucial to assess driver' awareness of driving scenario (e.g. surrounding vehicles or pedestrians) right before the take-over. In conclusion, evaluating driver's visual attention is a key task in the development of automated vehicles. Gaze tracking estimation is the common way for evaluating the driver's visual attention [4]. In the literature, there are different approaches mainly based on head-mounted eye trackers [5], [6], [7], active desktop-mounted eye trackers [8] and passive desktop-mounted cameras [4], [9], [10], [11], [12]. The former provide accurate gaze information but they are intrusive and costly. The latter are low-cost and non-intrusive but less accurate. Most of the existing visual attention systems have been validated in simulation with simple scenes. However, the simulation of driver attention in complex driving scenarios is rather challenging and highly subjective [2]. The goal of these systems is to automatically estimate the regions of interest where the driver is looking. In this sense, acquisition systems and computer vision techniques have gained importance regarding the traditional human factor approaches focused on driving behavior understanding using manual tools.

There are different challenging databases to validate driver visual attention in the state of the art. Recently, the project DR(eye)VE [13] collected 555,000 frames with driver attention maps. However, only few critical scenes were captured for only one driver. Berkeley DeepDrive Attention (BDDA) dataset [8] includes 1232 videos with critical situations in-lab designed. However, the critical situations do not cause true accidents and in consequence the attention simulation does not explore the dynamic process from critical situation to actual accidents. DADA2000 dataset [2] is a larger and more diverse video benchmark with driver attention and driving accident annotation simultaneously. It contains 2000 video clips with fairly complex accidental scenarios in diverse weather and lighting conditions.

This paper presents a low-cost and non-intrusive camera-based gaze mapping system able to estimate the driver attention simulation on challenging prerecorded traffic scenes through a heat map. Gaze direction is calculated by using the open-source state-of-the-art OpenFace 2.0 Toolkit [1]. After a slight calibration process and using a simple projection model, a heat attention map is obtained. Our goal is to implement a cheap and user-friendly measurement system able to work as most sophisticated systems in the localization of accidents over complex scenarios. Our proposal has been validated by using the recent and challenging public dataset DADA2000 which collects annotated driving scenarios based on real accidents providing the accident position and the attention map generated by some users who have watched the scene. We present some performance results for our attention model and we compare our numbers with the obtained with an expensive and active desktop-mounted eye-tracker [2] in similar conditions that the reported by the authors, reaching on par results. From our knowledge, this is the first time a camera-based vehicle-mounted driver attention system is evaluated in the challenging DADA2000 dataset.

2 SYSTEM ARCHITECTURE

The system architecture is divided in four steps to estimate where the driver is looking at. These steps are: video acquisition, gaze estimation, visual attention model and heat map visualization. A general overview of this architecture is depicted in Figure 1.

The complete system is connected using Robot Operating System (ROS). ROS is a framework that gives communication, standardisation and modularity facilities to the developed modules, offering an easy connectivity of each module in the general architecture of our autonomous vehicle.

2.1 Video acquisition

The measurement system consists on a ZED camera. This is an stereo camera, but for this application only the left lens is used. After a positioning study, the camera is located on the middle-bottom of the screen in front of the user where the videos are shown, as we depict in Figure 2. This position provides an optimum working with OpenFace because the face is frontal and all the facial landmarks, including the eyes ones that are vital for this application, are correctly acquired.

ZED camera was chosen due to its ROS compatibility. The manufacturer provides a wrapper to communicate both systems. Another reason was its high working frame rate (100 fps at VGA), higher that the obtained with an standard web-cam.

In order to emulate the perspective that a driver would have in the real vehicle and to have a good gaze dynamic behavior, an study about the user's position with respect to the screen, as well as a study about the optimal screen resolution and system frame rate have been carried out in next sections.

2.2 Gaze estimation

We are interested in non-intrusive systems that do not generate rejection nor fatigue in users and that are able to be used on a vehicle. Therefore, head-mounted eye trackers and desktop-mounted eye trackers are refused. Among vehicle-mounted cameras approaches, several gaze tracking techniques have been proposed in the literature [9], [10], [11], [12], [4]. OpenFace is one of the most popular open-source facial analysis tools due to its fine performance and robustness. We propose to use OpenFace 2.0

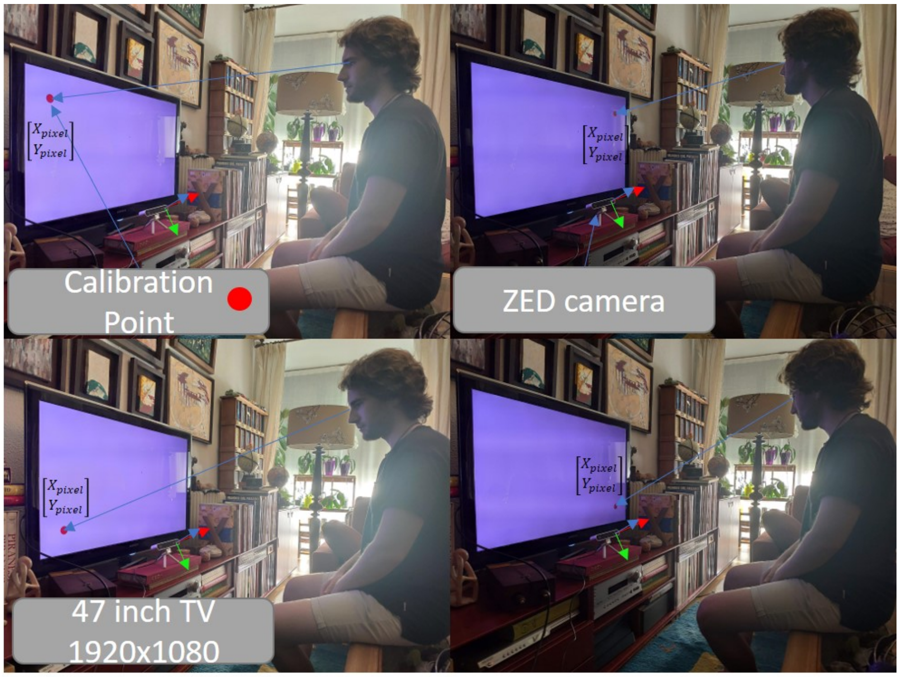


Fig. 2: Calibration step. The user under test has to watch during 8 seconds 4 points displayed on the screen

toolkit [1] because it provides facial behavior analysis algorithms including eye gaze. The tool begins detecting the face. Then, it extracts 68 facial landmarks and estimates the eyes gaze direction from these landmarks, as shown on Figure 1.

The process uses a Conditional Local Neural Field (CLNF) for detecting the facial landmark, including eyelids, iris and pupil. Once the eye and the pupils are located, eye-region data are used to compute the gaze vector for each eye. OpenFace provides three different eye vectors, one for each eye and a third one for the fusion of both eyes. In this project the fusion vector is applied. Eye gaze estimation was evaluated on the challenging MPIIGaze dataset [14] obtaining a mean absolute error of 9.1 degrees per frame, which can be enough for our application assuming this measurement uncertainty.

2.3 Visual attention model

In our application, gaze vector is used to know where the person under the test is looking. The vector is given as two angles, $[\theta_x, \theta_y]$, regarding the (X,Y) axis in the camera reference system. This vector is projected on the screen in order to get an attention pixel through a visual model. To minimize gaze vector uncertainty, a Kalman Filter experimentally adjusted, is implemented over the projection. To emulate driver position in the real vehicle, and following the same strategy that the authors in [2], the user is placed in front of a 47” screen, where traffic videos are shown, at a distance of 125 centimetres.

In this way, his visual field varies between ±24 degrees on the X axis and 27 degrees on the Y axis. In these ranges the projection model is quite linear. This is the reason because we propose to use a simple projection model learned in a previous calibration step in the same way that the done in a touch screen calibration process [15]. We apply a slight calibration method at the beginning of each experiment that consist on looking at four points of the screen to get the projection matrix and the limits of the screen where the test is run.

Calibration method translates the gaze vector into coordinates that accurately represent the projection of this vector on the screen in pixels. For each calibration point, the gaze vector ($[\theta_x, \theta_y]$) is projected on the screen ($[X'_k, Y'_k]$) using the trigonometric equations shown in equation 2. z is the average distance of the user's eyes regarding the camera reference. This parameter is also provided by OpenFace.

$$z = \frac{z_{right\_eye} + z_{left\_eye}}{2} \tag{1}$$

$$\begin{pmatrix} X'_k \\ Y'_k \end{pmatrix} = \begin{pmatrix} -z \cdot sin(\theta_x) \\ z \cdot sin(\theta_y) \end{pmatrix} \tag{2}$$

The matrix A is formed by the projected points (in cm) associated to the 4 screen points that have been projected during the calibration step (red points in Figure 2) on camera reference. In this step, these points have to be matched with the known positions of the calibration points on the screen in pixels ($[X_k, Y_k]$) regarding the screen reference (top-left corner), which are: $[X_1, Y_1] = [200, 200]$, $[X_2, Y_2] = [200, 900]$, $[X_3, Y_3] = [1700, 900]$, $[X_1, Y_1] = [1700, 200]$.

$$A = \begin{pmatrix} X'_1 & Y'_1 & 1 \\ X'_2 & Y'_2 & 1 \\ X'_3 & Y'_3 & 1 \\ X'_4 & Y'_4 & 1 \end{pmatrix} \tag{3}$$

To match these points we use a parametric model composed of the coefficients $[\alpha_X, \beta_x, \Delta_x]$ and $[\alpha_y, \beta_y, \Delta_y]$, which transform the spatial space from centimetres to pixel and change the reference from the camera to the left upper corner of the screen, according to the equation 4.

$$\begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{pmatrix} = A \begin{pmatrix} \alpha_X \\ \beta_X \\ \Delta_X \end{pmatrix} \; and \; \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{pmatrix} = A \begin{pmatrix} \alpha_Y \\ \beta_Y \\ \Delta_Y \end{pmatrix} \tag{4}$$

Equation 5 shows how the coefficients are calculated from equation 4.

$$\begin{pmatrix} \alpha_X \\ \beta_X \\ \Delta X \end{pmatrix} = A^{-1} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{pmatrix} \; and \; \begin{pmatrix} \alpha_Y \\ \beta_Y \\ \Delta Y \end{pmatrix} = A^{-1} \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{pmatrix} \tag{5}$$

After calibration, each attention point is obtained applying the trigonometric projection of its corresponding vector gaze provided by OpenFace, where $X_{cm} = -z \cdot sin(\theta_x)$ and $Y_{cm} = z \cdot sin(\theta_y)$, and the parametric transformation carried out by equation 6.

$$\begin{aligned} x_{pixel} &= \alpha_X X_{cm} + \beta_X Y_{cm} + \Delta_X \\ y_{pixel} &= \alpha_Y X_{cm} + \beta_Y Y_{cm} + \Delta_Y \end{aligned} \tag{6}$$

2.4 Heat map visualisation

To represent the spatial distribution of the gaze fixations on the screen, we use the common visualization approach known as heat map. To take into account the trajectory of the gaze, we integrate the spatial distribution of the fixations within a temporal window [t-w,t], where t is the current frame and w the number of last frames. This temporal window has been set to 1 second as done by [16]. In this way, we have 60 samples for frame in order to generate the heat map. With these samples a Gaussian model is built to represent the different areas of colour of the heat map, where ($M_t = [m_x(t), m_y(t)]$) is the centre of the Gaussian, C_t is the covariance matrix and $h(X,t)$ is the intensity of the heat map for the pixel (X=[x,y]). In a practical way, we draw ellipses in the heat map until two times the standard deviation of the Gaussian model.

$$h(X,t) = e^{-\frac{1}{2}(X-M_t)^T C_t^{-1}(X-M_t)} \tag{7}$$

3 Experimental Results

This section presents experimental results obtained for our architecture proposal. Evaluation is divided in two different subsections in order to compare our results with the obtained with other state-of-the-art systems.

3.1 Visual attention model evaluation

To evaluate the precision of our driver attention model we have applied the following testing procedure. After the calibration step, the user has to look at eight points, placed on different positions on the screen, during eight seconds for each one. To get a more stable view, the first two second of each point are despised. This procedure has been done to test the camera parameters, the camera position, the loss of precision by using glasses and finally the performance with different users. The metric used to evaluate the accuracy of the tool is the root mean square error (RMSE) of the gaze projection on the screen and its ground-truth. Accuracy is shown in total and regarding the (X,Y) axis and is evaluated in percentage, pixels and millimeters.

Camera parameters. To reach optimum results, a previous study about camera parameters was carried out. In this aspect, resolution and camera frame rate were changed to test precision for the different options. ZED camera allows four different resolutions with different frame rates, as we show on Table 1. Performance shows the ability of

OpenFace to process frames. This test was done only by one user, but to achieve more robust conclusions each test was repeated 5 times.

Results show that, on the X axis, best performance is get by VGA resolution at 100 frames per seconds. On the Y axis, best performance is get by HD720. In total, best accuracy is obtained for HD720. In conclusion, this last configuration is taken for the following tests. Also performance shows that OpenFace is not able to work at 100 Hz with an VGA resolution.

Table 1: Testing accuracy vs camera parameters on a 1920x1080 screen. Test done by one person looking at eight points on a screen. Test was done 5 times.

Resolution	Frame rate	Acc_x			Acc_y			Acc_total			Performance (Hz)
		%	pix	mm	%	pix	mm	%	pix	mm	
HD2K	15	1,9	36,5	11,4	5,4	57,9	18,1	4,0	77,2	24,1	14,761
HD1080	30	1,4	27,5	8,6	5,7	61,0	19,1	4,1	79,2	24,7	29,378
HD720	60	1,9	36,1	11,3	**4,0**	43,7	13,6	**3,2**	60,5	18,9	50,74
VGA	100	**0,6**	12,3	3,8	4,7	50,2	15,7	3,3	63,7	19,9	74,63

Camera position. Camera position is vital for the system working. Three different positions compatible with a real car were tested, all of them aligned with the middle of the screen but at different heights (10, 48 and 78 centimetres over the table where the screen is placed). Results are shown on Table 2 for the same conditions that the explained in the before section. The optimal position for the camera is just in front of the user, but this is not possible in a real car because hides the road scene. Top position is not allowed because face landmarks are occluded most of the time and the model does not work properly. Then, bottom position was chosen.

Table 2: Testing accuracy vs camera position on a 1920x1080 screen. Test done by one person looking at eight points on a screen. Test was done 5 times.

Position	Acc_x			Acc_y			Acc_total		
	%	pix	mm	%	pix	mm	%	pix	mm
Top	-	-	-	-	-	-	-	-	-
Mid	**1,9**	36,1	11,3	**4,0**	43,7	13,6	**3,2**	60,5	18,9
Bot	**2,5**	48,0	15,0	**10,6**	114,5	35,8	**7,7**	147,9	46,2

Glasses Some drivers wear glasses while driving and some other drivers prefer to drive with sun-glasses. Performance of every gaze monitoring system must be analyzed under these common situations. Commercial applications like Tobii Pro Glasses [17]do not work properly because their sensors are on some special glasses and two different glasses cannot be used at the same time.

In our case, precision will get worse for sure due to the glasses hide some face landmarks. To evaluate impaired performance three different test were done: no glasses, with glasses and with sun glasses, in the same conditions that in the before case. Results are depicted on Table 3. For the sunglasses case, our system does not work because eyes landmarks are hidden and gaze can not be calculated. For the glasses case, accurate numbers decrease a little but can be used without problems.

Table 3: Testing accuracy vs glasses on a 1920x1080 screen. Test done by one person looking at eight points on a screen. Test was done 5 times

Eyes-glasses	Acc_x			Acc_y			Acc_total		
	%	pix	mm	%	pix	mm	%	pix	mm
Free	**1,9**	36,1	11,3	**4,0**	43,7	13,6	**3,2**	60,5	18,9
Glasses	**2,5**	48,7	15,2	**4,6**	50,0	15,6	**3,7**	71,7	22,4
Sunglasses	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**

Precision test Once the best camera parameters and camera position have been found, it is time to test the accuracy of our model with different users. 25 users were requested for the tests. Firstly, they were asked to calibrate the tool looking at 4 points as shown on Figure 2. Age of the users were between 17 and 55 years old and 3 of them wore glasses during the test.

Results for all of the users can be seen on Figure 3. Red circles indicate the goal points where the person has to look at. The heat map represents the spatial distribution of the data, divided in this case in eight different clusters, modelled by a Gaussian Mixture Model (GMM) of eight Gaussian. The heat map for each Gaussian indicate the probability of a pixel to belong to each cluster. In theory, the centre of each Gaussian should be placed over the corresponding calibration point and its covariance indicates the measurement uncertainty.

Our method achieves about 1.9 % error on the X axis and 4 % error on Y axis, being bellow the reported by OpenFace for the MPIIGaze dataset (9.1 degrees equivalent in our case to 19 % on X axis and 34 % on Y axis) [14]. This comparison should be taken with caution since in our case gaze movements are limited and the number of users who undergo the test is less.

Results on Y axis are worse due to the number of pixels for the vertical field of view range is smaller that the corresponding to the horizontal one. Despite these errors are considerable, they can be enough for our application. Figure 5 shows the error overprinted on an image with a Crash Object in order to compare scales. As we can see, in spite of the measurement uncertainty the object is perfectly detected in the frame, which demonstrate our method can be used to localize accidents in complex scenarios.

Table 4: Testing accuracy on a 1920x1080 screen. Test done by one person looking at eight points on a screen. Test done by 25 users looking at eight points on a screen.

	Acc_x			Acc_y			Acc_total		
	%	pix	mm	%	pix	mm	%	pix	mm
25 people	**1,9**	36,1	11,3	**4,0**	43,7	13,6	**3,2**	60,5	18,9

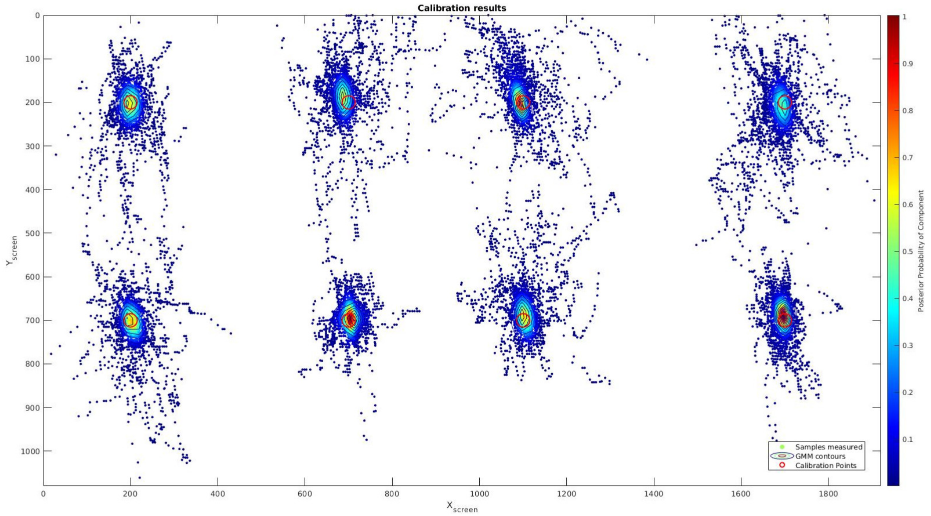


Fig. 3: Calibration results. 8 points are displayed on the screen (Red circle). The results from our output creates 8 Gaussian around the testing points.

3.2 DADA2000 evaluation

Once shown the potential of our visual attention method to localize objects of a certain size in an image, we are ready to test it on a challenging video benchmark with driver attention and driving accidents annotated to evaluate the performance of our proposal to estimate accidents in video sequences. To have a clear idea of the quality of our method we will compare results with the obtained using an expensive and active desktop-mounted eye-tracker watching the same videos in similar conditions. Different options were raised, such as MIT300 [18], CAT2000 [19], DR(eye)VE [16] and DADA2000 [2]. The two first ones were discarded because they are not focused on drivers. Between the two lasts, authors decided to use DADA2000 because it is focused on real traffic accidents, which are the critical moments to evaluate the driving attention.

DADA2000 was done collecting accident videos from different websites, and it is composed of 658,476 frames in 2000 videos with a resolution of 1584x660 pixels. At the time of this work, only half of data is public. Then, we use 1000 videos for the experiments, which are classified in three groups: training (598 videos), validation (198 videos) and testing (222 videos). The videos are divided into 54 accident categories classified into two large groups, ego-car involved and ego-car uninvolved. Videos are recorded in different environment such as illumination conditions (day and night), weather (sunny, rainy and snowy) and occasion (urban, rural, highway and tunnel). The dataset provides the following information: annotated crash-objects position in pixels per frame, attention map for each frame and temporal window for each accident indicating the involved frames. Videos are partitioned in three main clips: before the accident, accident and after the accident, as it is depicted in Figure 4. Also, the accident clip is divided in three sub-sections for a better analysis (Start, Mid and End).

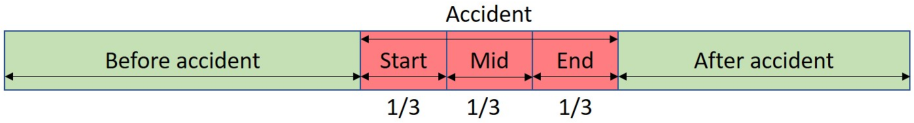


Fig. 4: Temporal partition of the video. Dividing the accident part in three pieces of same length to evaluate the results by sections.

Crash Object detection in DADA2000 benchmark To test Crash Object detection, we employed 4 volunteers who were invited to watch the testing section of the dataset, composed by 222 videos. For each visualization, the output of our visual attention model was recorded in a synchronized way with the video. In order to wide the working field of view, dataset was resized from 1584x660 to 1920x1080 to match it with the screen resolution used in the experiment. The clips were played at 30 fps to compare our results with the obtained by the authors of DADA2000 [2] in the same terms. We recorded attention map without temporal aggregation, obtaining two measures per frame, because our system run at 60 Hz.

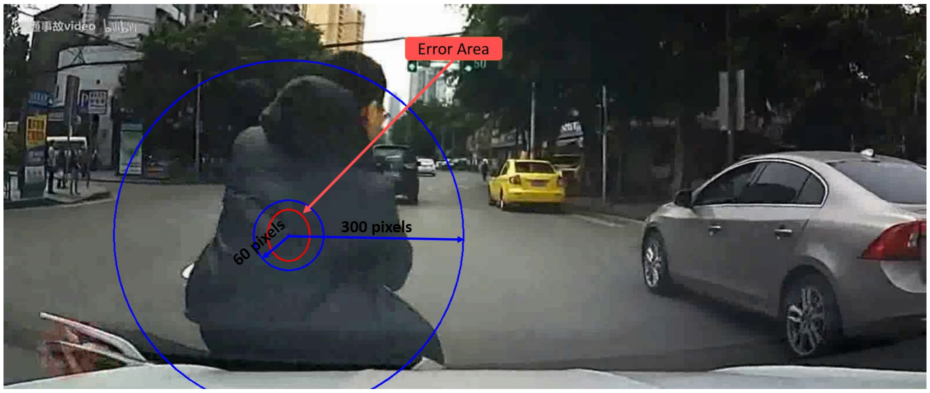


Fig. 5: DADA2000 frame. Crash object circle for 60 and 300 pixels (blue). Error area (red).

The goal of this test is to localize driver attention on a screen. We use the same metrics as the authors of the dataset to evaluate this parameter. We measure the distance between the Crash Object center (ground-truth manually annotated) and the attention points obtained for each frame. If this distance is below a certain threshold (Th) in pixels the detection is consider as true and false if it is higher. Figure 5 shows a frame with the Crash Object an its true detection area, represented by a circle, for a Th equals to 60 and 300 pixels. Performance detection is solved as a binary problem: True, if the attention point falls inside the circle, and False, if it falls outside. For this work, we only have tested experiments with the priori information option (with-priori), that means users are told that they have to find crash-objects in the sequences. With these premises we obtain two indicators:

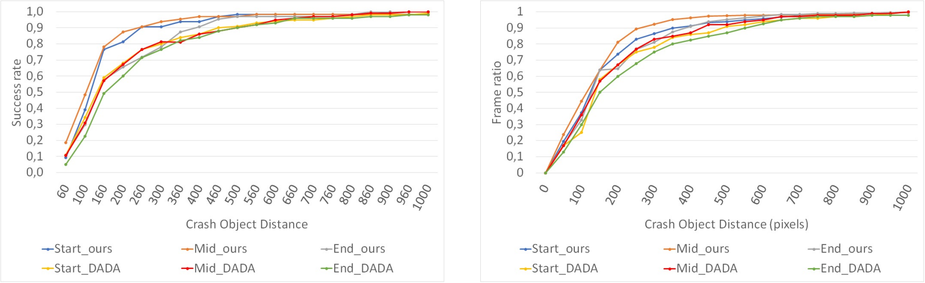
- Frame ratio: percentage of frames in each clip section where the attention point is inside the correct detection area. This is measured frame by frame because the crash object change with it.
- Success rate: percentage of clips which frame ratio for the entire clip is over 50%. Clips refers different video sub-sections defined for the accident.

Results for different detection area sizes and a comparison with the numbers presented by the DADA2000 authors using a Senso Motoric Instrument (SMI) RED250 desktop-mounted infrared eye tracker are shown on Table 5. Our success rate is higher than the obtained by the DADA200 authors for all the partitions except for the start one with a threshold of 60 pixels. However, our numbers are obtained for 4 users per video instead of the 5 they argue to use, even though they also argue every video was watched by at least two people, indicating that not all of were watched by the 5 users. For a 300 pixels threshold we have obtained more than 90 % of success rate for start and mid sections of the accident event. The end section always get less success rate because, at this part of the video, users normally fix their attention in other part of the scene. Results are or par or even better than the claimed by the authors of DADA2000 but it must be remarked that our experiment has been recorded on a 47 inch screen instead on a 24 inch screen. Results shows that early accident detection is worse than for the middle partition because users take some time to react. In the last part of the accident some users under test keep their eyes out the accident and the late accident detection gets worse.

Figure 6(a) shows the success rate and frame ratio curves, obtained for a threshold between 60 and 1000 pixels and for the three sub-sections presented in the accident partition, in the same way that the shown by the dataset authors in [2]. Our results outperform the obtained for our baseline for all the analyzed sub-sections.

Table 5: Success rate: percentage of number of clips that frame ratio for the entire clip is more than half of the frames of the entire accident scene.

System	Ours (Based on OpenFace)			DADA2000		
Th (pixel)	Start	Mid	End	Start	Mid	End
60	9,4%	18,8%	10,9%	10,8%	10,7%	5,1%
100	39,1%	48,4%	29,7%	34,4%	30,8%	22,6%
160	76,6%	78,1%	57,8%	59,0%	57,2%	49,1%
200	81,3%	87,5%	65,6%	68,0%	67,2%	60,1%
260	90,6%	90,6%	71,9%	76,6%	76,6%	71,5%
300	90,6%	93,8%	78,1%	80,0%	81,3%	76,6%



(a) Success rate obtained by the system proposed.

(b) Frame ratio obtained by the system proposed.

Fig. 6: Results obtained on the testing

Heat attention map on DADA2000 Heat attention map is a good visualization tool to represent where a user is looking when he is driving. To represent the trajectory of the gaze, a temporal window is used. For the presented experiments the temporal window has been set to 1 second, like in the DR(eye)VE project.

Figure 7 shows the way this experiment has been done. The user is looking at the accident while the camera is recording him from a lower position. The attention map

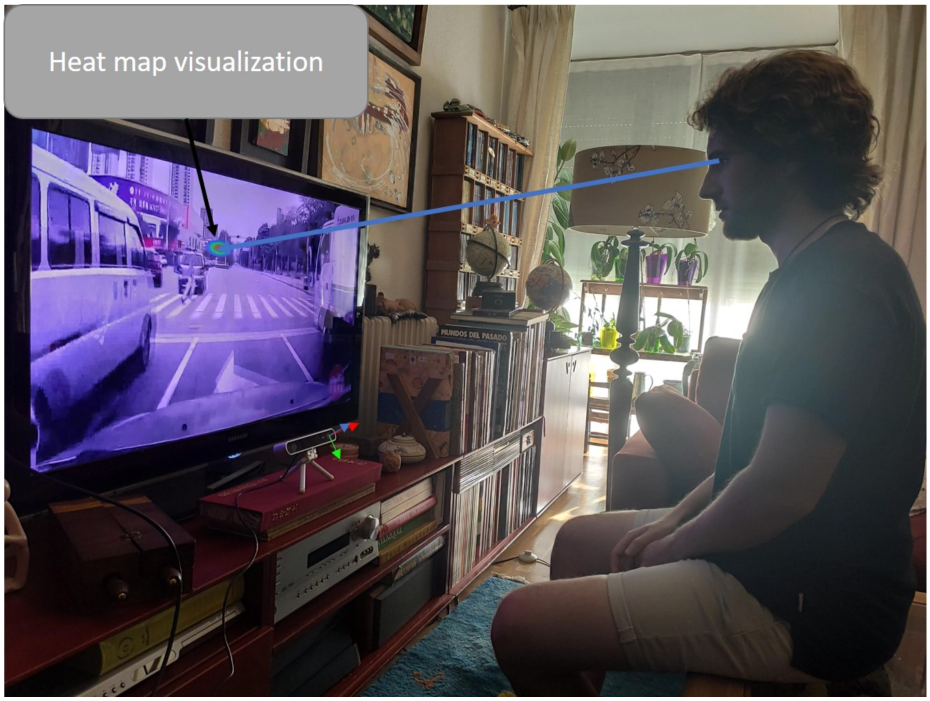


Fig. 7: Heat attention map. First column shows the DADA2000 attention map. Second column shows our heat attention map.

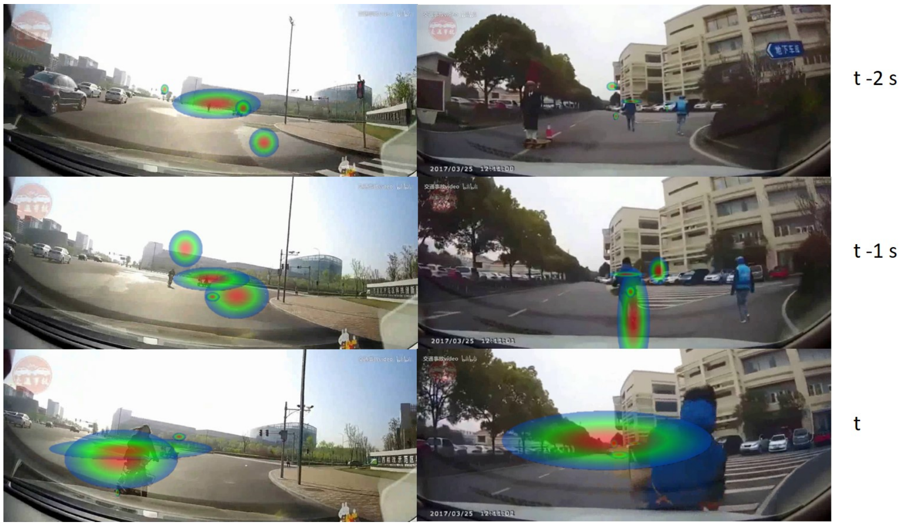


Fig. 8: Attention map with temporal aggregation for the frame t, t-1 s and t-2 s.

generated by the samples is shown. As we can see, our attention map is included in the baseline map, showing the usability of our tool but using a cheaper and non-intrusive sensor.

Figure 8 shows how temporal aggregation on heat attention map can help to predict where the accident will happen. The figure shows 2 accident situations with their corresponding maps in the image where the accident takes place as a heat map. Images displayed are with a temporal difference of 1 second. As we can see, the heat map is able to correctly predict all the crash objects.

4 CONCLUSIONS AND FUTURE WORKS

This paper presented a new low-cost and not intrusive method to get visual attention maps, based on a passive vehicle-mounted camera, as an alternative to the head-mounted eye-tracker and the active desktop-mounted eye trackers (our baseline), which are intrusive and costly. Our proposal has been validated in the challenging public dataset DADA2000, showing on par results with our base line in similar conditions. This fact confirms our technique is a good tool for driver attention monitoring able to be used in the design of take over systems and driving environment awareness systems for automated vehicles.

As future works we plan to implement a more sophisticated attention model, to obtain saliency maps with temporal windows on challenging driving scenarios using deep-learning techniques and to test our proposals in our simulator based on CARLA [20].

ACKNOWLEDGMENT

This work has been funded in part from the Spanish MICINN/FEDER through the Techs4AgeCar project (RTI2018-099263-B-C21) and from the RoboCity2030-DIH-CM project (P2018/NMT- 4331), funded by Programas de actividades I+D (CAM) and cofunded by EU Structural Funds.

References

1. T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, “Openface 2.0: Facial behavior analysis toolkit,” in *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pp. 59–66, IEEE, 2018.
2. J. Fang, D. Yan, J. Qiao, J. Xue, H. Wang, and S. Li, “Dada-2000: Can driving accident be predicted by driver attention$f$ analyzed by a benchmark,” in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 4303–4309, IEEE, 2019.
3. F. Jimenez, *Intelligent Vehicles: Enabling technologies and future developments*. Butterworth-Heinemann, 2017.
4. L. Yang, K. Dong, A. J. Dmitruk, J. Brighton, and Y. Zhao, “A dual-cameras-based driver gaze mapping system with an application on non-driving activities monitoring,” *IEEE Transactions on Intelligent Transportation Systems*, 2019.

5. E. Dalmaijer, S. Mathôt, and S. Stigchel, “Pygaze: An open-source, cross-platform toolbox for minimal-effort programming of eyetracking experiments,” *Behavior Research Methods*, vol. 46, 11 2013.
6. M. Cognolato, M. Atzori, and H. Müller, “Head-mounted eye gaze tracking devices: An overview of modern devices and recent advances,” *Journal of rehabilitation and assistive technologies engineering*, vol. 5, p. 2055668318773991, 2018.
7. J. Shen, S. Zafeiriou, G. G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic, “The first facial landmark tracking in-the-wild challenge: Benchmark and results,” in *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 1003–1011, Dec 2015.
8. Y. Xia, D. Zhang, J. Kim, K. Nakayama, K. Zipser, and D. Whitney, “Predicting driver attention in critical situations,” in *Asian conference on computer vision*, pp. 658–674, Springer, 2018.
9. N. Mizuno, A. Yoshizawa, A. Hayashi, and T. Ishikawa, “Detecting driver’s visual attention area by using vehicle-mounted device,” in *2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, pp. 346–352, IEEE, 2017.
10. F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, “Driver gaze tracking and eyes off the road detection system,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2014–2027, 2015.
11. R. A. Naqvi, M. Arsalan, G. Batchuluun, H. S. Yoon, and K. R. Park, “Deep learning-based gaze detection system for automobile drivers using a nir camera sensor,” *Sensors*, vol. 18, no. 2, p. 456, 2018.
12. P. Jiménez, L. M. Bergasa, J. Nuevo, N. Hernández, and I. G. Daza, “Gaze fixation system for the evaluation of driver distractions induced by ivis,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1167–1178, 2012.
13. A. Palazzi, D. Abati, F. Solera, R. Cucchiara, *et al.*, “Predicting the driver’s focus of attention: the dr (eye) ve project,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 7, pp. 1720–1733, 2018.
14. X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, “Mpiigaze: Real-world dataset and deep appearance-based gaze estimation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 1, pp. 162–175, 2017.
15. W. Fang and T. Chang, “Calibration in touch-screen systems,” *Texas Instruments Incorporated*, vol. 10, 2007.
16. S. Alletto, A. Palazzi, F. Solera, S. Calderara, and R. Cucchiara, “Dr (eye) ve: a dataset for attention-based tasks with applications to autonomous and assisted driving,” in *Proceedings of the ieee conference on computer vision and pattern recognition workshops*, pp. 54–60, 2016.
17. “Tobii pro glasses 2,” 2020. https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/.
18. Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba, “Mit saliency benchmark.”
19. A. Borji and L. Itti, “Cat2000: A large scale fixation dataset for boosting saliency research,” *arXiv preprint arXiv:1505.03581*, 2015.
20. A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *Proceedings of the 1st Annual Conference on Robot Learning*, pp. 1–16, 2017.