System for detecting driver's drowsiness, fatigue and inattention

Aleksa Arsić, Velibor Ilić, Bogdan Pavković

Abstract – ***One of the main challenges and requirements of today's automotive industry is, besides functional safety, to assure and secure the safety of the driver, passengers, other traffic participants, and environment. As many external factors cannot be controlled by the driver, a high percentage of road accidents with even fatal outcomes are caused by the driver's drowsiness and other factors that can be controlled by the driver himself. Thus, for modern Advanced Driving Assistance Systems (ADAS) it would be of great use to implement a reliable system for detecting driver's drowsiness, fatigue, and inattention. One way this could be achieved is by using convolutional neural networks (CNN) and machine learning (ML) principles. In this paper, we present academic research on the topic which is based on three CNN's used for monitoring the driver with a possibility to dispatch notification when concluded that his state of attention is not suitable for operating a motorized vehicle. Each of the three CNN's processes different parts of the image from the inside of the vehicle and are connected in a series in a way that the outputs of the previous are used as the inputs for the next CNN model in the series. State of drowsiness, fatigue, and inattention is assessed using data extracted from the driver's eyes and based on the angle of the driver's face.***

***Keywords* — machine learning, attention, deep neural networks, convolutional neural networks (CNN), advanced driver assistance systems (ADAS)**

I. INTRODUCTION

In the Republic of Serbia in 2019, 25% of road accidents with fatal outcomes were caused by driver's drowsiness, fatigue, inattention, and other similar psychophysical conditions [1]. Such accidents make the third largest cause of road accidents with fatal outcomes in the country. Many European countries record similar numbers [2]. Hence, the Driver Monitoring System would be of great use in modern vehicles since it can lower these numbers and increase passengers', other traffic participants, and environmental safety.

In this paper, we propose a driver monitoring system based on three Deep Convolutional Neural Networks (CNN). The first CNN, face detection network (FDN), as input uses images captured from the camera, this network is trained to determine the frame that contains the driver's face. The second CNN network for selecting the region of eyes (REN), as input uses extracted image with face, and this network is trained to determine the frame that contains the eyes of the driver. The third CNN network, a network for monitoring single eye (SEN), as input uses extracted image with eyes, and this network monitors individual eyes. After processing the image through these three neural networks, it is assessed whether the driver lacks attention. By determining a lack of attention, an option of giving a sound warning is integrated to avoid potentially dangerous or even fatal situations. The paper is organized as follows. Section II shows related work on the topic and the main difference between those approaches and our proposed approach. Section III contains a description of used CNN models and the structure of software for monitoring the driver's attention. Section IV contains a description of generating needed datasets for the training and testing of CNNs. After that, section V represents the achieved result, and finally, the last section contains conclusions and we summarize the presented work and give directions for the future work. We included a web link to the short video demonstration of the proposed system at the end of this paper.

Aleksa Arsić, RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia, (e-mail: aleksa.arsic@rt-rk.com).

Velibor Ilić, RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia, (e-mail: velibor.ilic@rt-rk.com).

Bogdan Pavković, RT-RK, Institute for Computer Based Systems, Novi Sad, Serbia, (e-mail: bogdan.pavkovic@rt-rk.com).

II. RELATED WORK

Systems that monitor driver's attention levels are one of the main components of Advanced Driver Assistance Systems (ADAS) and there have been numerous research papers on this subject. Some of them, older ones, are based on some of the classic, more traditional approaches in computer vision (e.g. Kalman filter, RANSAC, POSIT) [3][4], while some of them took a different approach using modern approaches with Machine Learning (ML) principals and Deep Neural Networks (DNN). [5][6][7][8]

S. Park et al [9] proposed a method for detecting driver's drowsiness state using three CNN's where they have adopted pre-trained AlexNet [10], VGG-FaceNet [11], and FlowImageNet [12]. Where AlexNet was used for extracting image features, VGG-FaceNet was used for extracting facial feature representation and FlowImageNet was used for behavior feature representation. After which outputs of the three networks were used and combined to create a single prediction on the drowsiness state. On the other hand, R. Jabbar et al [6] presented a method for detecting driver's drowsiness in real-time based on Multilayer Perceptron Classifier (MPC) with three hidden layers. In their proposition 68 landmark coordinates are extracted from the images using Dlib [13] to map the facial structures of the face and which are used for training the used MPC model.

Our proposition does not include any pre-trained CNN model architectures, thus, we have designed our own CNN architectures and trained them with our own generated training datasets. These models are used in a similar way, where each model is capable of extracting coordinates of

978-1-6654-2585-8/21/$31.00 ©2021 IEEE

interest starting from larger image elements (drivers face) to smaller ones (points of interest on drivers eyes) and where outputs of the previous are used as the inputs for the next CNN model in the series. Thus, partially mocking the humans brain cognitive behavior of focused and selective attention. [14] I.e. having the ability to respond to specific stimuli and avoid distracting ones.

III. Driver attention software

Convolutional neural networks represent the specific architecture of artificial neural networks mainly used in the field of computer vision. One of their main characteristics is that they are shift-invariant. In other words, they are capable to detect the same objects on different images regardless of their position inside the image.

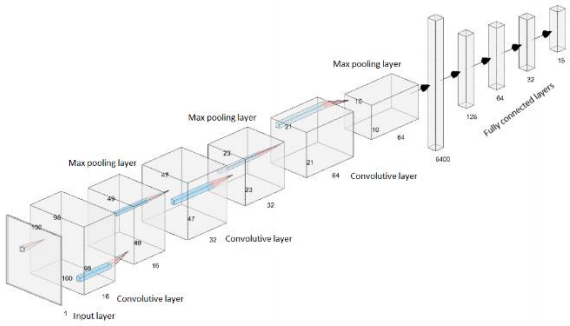

Fig. 1 Architecture of used CNN's

As mentioned, in this paper, we used three CNNs whose architecture is similar to one another (Fig. 1). The inputs of each CNN are grey-scaled images that have 100×100 pixels dimension. Those images can be captured from a video-source live camera recording or pre-recorded video.

The main difference between used CNNs is in the output layer. The dimensions of the final fully connected layer are 12, 10, and 15 fully connected nodes for the FDN, REN, and the SEN model respectfully.

The idea for the algorithm that monitors the driver's attention starts with having a fixed video source directed towards the driver and forwards individual frames for further processing. After extraction of a single frame, it is necessary to process it into the appropriate input format supported by the FDN model.

In this case, the source was a simple webcam that can capture RGB frames with a resolution of 640×480 pixels. Firstly, the captured frame is preprocessed in a way that the RGB image is converted to a grey scaled image and resized to dimensions of 100×100 pixels, where every pixel is normalized to the value range [0, 1]. The preprocessed frame is propagated through the FDN model and if the predictions confirm that the driver's face is detected, denormalized predictions of the center of the face are used to cut the face from the whole frame with dimensions of 300×200. Again, the face frame is preprocessed, dimensions of the face frame are rescaled to dimensions of 100×100 and every pixel is normalized to the value range [0, 1]. After preprocessing the face frame is propagated through the REN model.

REN model is capable to locate frame with the driver's eyes. As several papers suggest, the potential drowsy state of the person can be determined with great confidence solely from analyzing his eyes. [15] [16] [17] Whereas one of the most commonly used methods for drowsiness assessment is based on the monitoring person eyes, monitoring of percentage eye openness tracking, (PERCLOS). [18]

If at least one eye is found in the predictions of the REN network, the eye frame is extracted from the face frame in a dimension of 100×100 pixels, where all values of individual pixels are normalized to the range of [0, 1]. Such manipulated eye frame is propagated through the SEN model.

SEN model is trained to monitor individual eyes. This network monitor left and right outer eye points, upper and lower outer center points, and the pupil center. With those in mind, the SEN model can predict if the observed eye is open or closed, and where the gaze is directed.

It can be observed that while we designed our own CNN's input size of all three models are the same (100x100 pixels) and FDN and REN models do not preserve the aspect ratio of height and width of the input frame. In other words, images that were to be fed to FDN and REN models are first rescaled to the size of their input layers. It was possible to preserve the aspect ratio of the input images which would result in three completely different architectures of our CNNs. The described approach is mainly taken because of common practice when designing CNNs to have a square-shaped input layer. [19]

The data flow of Driver Attention Software is presented in *Fig. 2*.

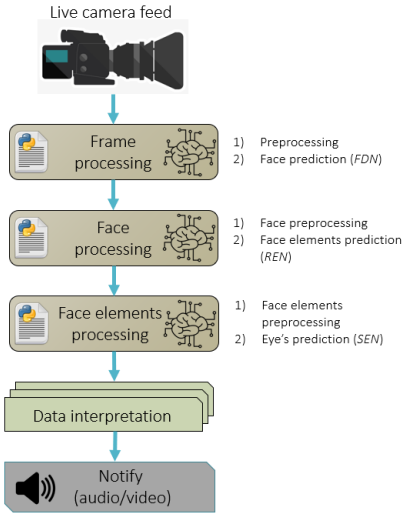

Fig. 2 Driver attention software dataflow

In Fig.3, we present the points of interest that every model can predict.

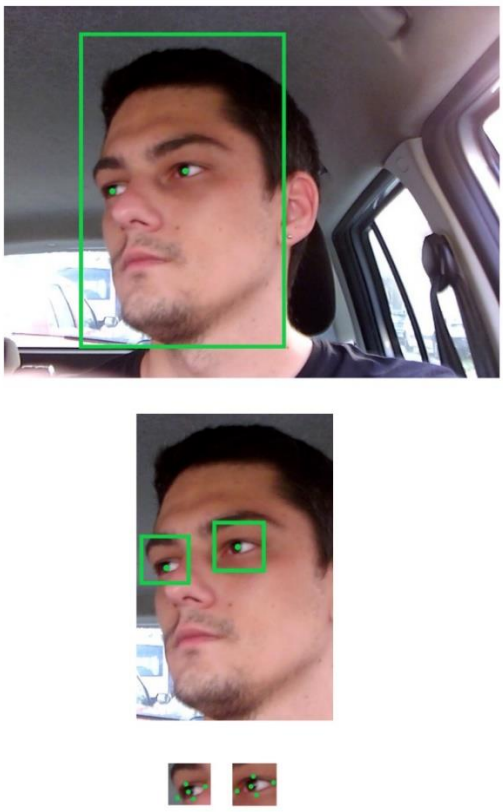

Fig. 3 Points of interest predicted by all three CNN models

Alongside points of interest, every CNN model can predict several other states of the driver's face and face elements.

$$[\boldsymbol{p_f}\ \boldsymbol{f_x}\ \boldsymbol{f_y}\ \boldsymbol{e_{lx}}\ \boldsymbol{e_{ly}}\ \boldsymbol{e_{rx}}\ \boldsymbol{e_{ry}}\ \boldsymbol{p_l}\ \boldsymbol{p_r}\ \boldsymbol{p_u}\ \boldsymbol{p_d}\ \boldsymbol{f_w}]$$

Output tensor of the FDN model consists of twelve values where $\boldsymbol{p_f}$ represents the probability that the driver's face is found on the current frame. $(\boldsymbol{f_x}, \boldsymbol{f_y})$ represents x and y coordinates of the center of the driver's face. $(\boldsymbol{e_{lx}}, \boldsymbol{e_{ly}})$ represents x and y coordinates of the driver's left eye, $(\boldsymbol{e_{rx}}, \boldsymbol{e_{ry}})$ represent the x and y coordinates of the driver's right eye. Values $(\boldsymbol{p_l}, \boldsymbol{p_r}, \boldsymbol{p_u}, \boldsymbol{p_d})$ represent the probability that the driver's face is oriented to the left, right, up, or down, respectively. Finally, value $\boldsymbol{f_w}$ represents width estimate of the driver's face.

$$[\boldsymbol{p_{le}}\ \boldsymbol{p_{re}}\ \boldsymbol{e_{lx}}\ \boldsymbol{e_{ly}}\ \boldsymbol{e_{rx}}\ \boldsymbol{e_{ry}}\ \boldsymbol{p_l}\ \boldsymbol{p_r}\ \boldsymbol{p_u}\ \boldsymbol{p_d}]$$

Output tensor of the REN model consists of ten values where $(\boldsymbol{p_{le}}, \boldsymbol{p_{re}})$ represents the probability that the left and right eye, respectively, are present on the current face frame. $(\boldsymbol{e_{lx}}, \boldsymbol{e_{ly}})$ represents x and y coordinates of the driver's left eye, $(\boldsymbol{e_{rx}}, \boldsymbol{e_{ry}})$ represent the x and y coordinates of the driver's right eye. $(\boldsymbol{p_l}, \boldsymbol{p_r}, \boldsymbol{p_u}, \boldsymbol{p_d})$ represent the probability that the driver's gaze is oriented left, right, up or down, respectively.

$$[\boldsymbol{p_{ec}}\ \boldsymbol{c_{ux}}\ \boldsymbol{c_{uy}}\ \boldsymbol{c_x}\ \boldsymbol{c_y}\ \boldsymbol{c_{dx}}\ \boldsymbol{c_{dy}}\ \boldsymbol{l_x}\ \boldsymbol{l_y}\ \boldsymbol{r_x}\ \boldsymbol{r_y}\ \boldsymbol{p_l}\ \boldsymbol{p_r}\ \boldsymbol{p_u}\ \boldsymbol{p_d}]$$

Output tensor of the SEN model consists of fifteen values where $\boldsymbol{p_{ec}}$ represents the probability that the observed eye is closed. $(\boldsymbol{c_{ux}}, \boldsymbol{c_{uy}})$ represents the x and y coordinates of the upper center point of the eye. $(\boldsymbol{c_x}, \boldsymbol{c_y})$ represents the x and y coordinates of the pupil of the eye. $(\boldsymbol{c_{dx}}, \boldsymbol{c_{dy}})$ represents the x and y coordinates of the lower center point of the eye. $(\boldsymbol{l_x}, \boldsymbol{l_y})$ represents the x and y coordinates of the left point of the observed eye. $(\boldsymbol{r_x}, \boldsymbol{r_y})$ represents the x and y coordinates of the right point of the observed eye. $(\boldsymbol{p_l}, \boldsymbol{p_r}, \boldsymbol{p_u}, \boldsymbol{p_d})$ represent the probability that the driver's gaze is oriented left, right, up or down, respectively.

Based on CNN's outputs which represent the probability that the driver's head has an angle and the possibility that the eye is closed, the decision of alerting the driver about the absence of attention is made. The decision to notify the driver is made throughout the time interval of 2.5 seconds if any of the following criteria are met:

- FDN model predicts 40 frames for which the driver's face has an angle,
- SEN model predicts 35 frames in which the observed eye is closed (where in the case where both eyes are found and observed, it is sufficient that just one meets these criteria).

If described criteria are met the driver is notified using sound signals thus preventing potentially dangerous and fatal outcomes.

Values, 40 for the FDN model and 35 for the SEN model are experimentally taken for the criteria as for those values system gave the best results of the several other values tested.

The video notifications and representations of CNN's outputs are displayed because of their practical and experimental nature as they help visualize described processes and show useful information about the state of the entire system. The predictions of CNNs are outlined in a form of boundary rectangles which represent the face region prediction and points of interest in the eyes.

IV. Generating Datasets

Each dataset contains images of eight people, five males, and three females, who voluntarily participated in the research. Every participant was asked to shoot a short video of themselves in a simulated environment. The datasets included videos of different distances from the camera, as well as videos from both the inside and outside of a parked vehicle. They were instructed to behave naturally as if they were in a real-life driving situation.

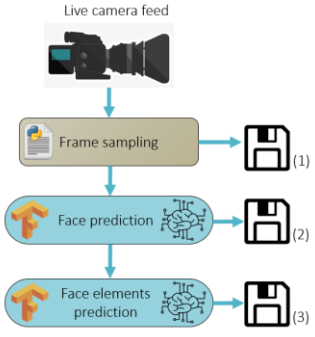

Fig. 4 Process of generating individual datasets

The process of generating our own datasets is shown in Fig. 4 and can be divided into several sections:

1. Individual frames are captured and saved from the video source. These frames are manually labeled and used in the training of the FDN model. This dataset contains numerous images with drivers' face being inside captured frame (partially or whole) and images without the driver in them.
2. Using FDN model predictions from whole frames, face frames are cut, manually labeled, and used to train the REN model.
3. Using predictions of the REN model, face elements (eyes) are cut from face frames, manually labeled, and used to train the SEN model.

Final datasets consist of around ten thousand images for the FDN model and about five thousand images for both REN and SEN models. Around twenty thousand images were individually labeled in the process of generating the datasets using self-implemented labeling software.

V. Results

The testing accuracy of all three CNN models can be seen in Table 1.

CNN model	Test accuracy [%]
FDN	90.79
REN	86.68
SEN	72.65

Table 1 Testing accuracy of the CNN models

The train/test split is performed before training each of the CNN models, where the final evaluation of the CNN model is performed over the test data set. Test datasets contain ~ 10% of the total datasets used to train each of the CNN models. The test set for the FDN model consists of around 1000 images while test sets for the REN and SEN models consist of about 500 images each. It is important to mention that each test dataset is chosen randomly from each training dataset. The predictions are considered correct if the percentage difference between the expected value and the prediction value is less than 10% when it comes to the parameters that represent the coordinates of the points of interest. In other cases, when the prediction of the CNN model represents the probability, we assume that the prediction is correct if the prediction value exceeds the threshold of 0.5 when the expected value is equal to 1 and less than 0.5 when the expected value is equal to 0.

When used in the proposed system for detecting drivers' drowsiness, fatigue, and inattention, implementation of Simple Moving Average (SMA) with a window of fifteen past frames is used after every CNN model before predictions of the model are used for further processing. The use of SMA achieves greater error tolerance of the whole system. Even if the CNN models make a mistake in a few frames, it will not impact the result of the overall assessment of the driver's attention state.

It was mentioned earlier that the FDN model and REN model were both trained to predict the same points of interest, the central points of the driver's eyes. Based on the obtained results we consider the REN model exclusion possibility, i.e., observation is performed to see if the eyes are too small elements within the whole frame to be found with satisfactory precision by the FDN model.

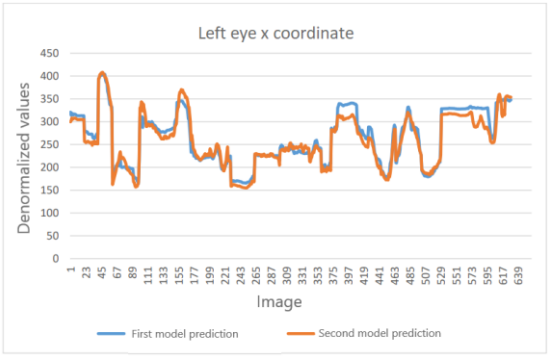

Fig. 5 Left eye x coordinate predictions of the FDN and REN model

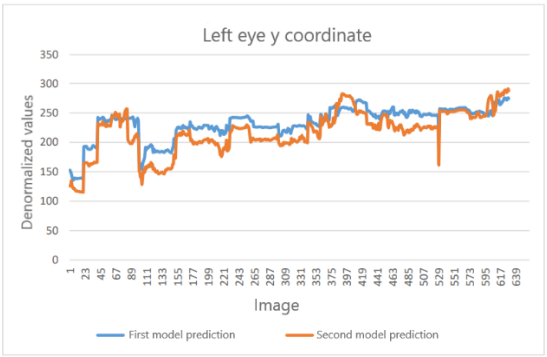

Fig. 6 Left eye y coordinate predictions of the FDN and REN model

In Fig. 5 and Fig. 6, we present predictions of the FDN model and REN model of the driver's left eye x and y positions, respectively.

At first glance, it would seem there was no need to introduce another CNN model into the system. However, attention should be paid to the fact that figures show denormalized predictions, thus the values are presented in pixels. The REN model is integrated as one of the system parts, and based on the empirical results, it is known that these results are satisfactory accurate.

The difference between predictions of the FDN and the REN model are in the range from 20 to 40 pixels, which is an unacceptable and extremely large error. A conclusion is reached that the FDN model is not capable of reliably finding the central points of the left eye, i.e., it made sense to introduce the second, REN model into the system. Similar results were obtained when analyzing the right eye predictions of the mentioned models.

For experimental purposes, we have trained another CNN with the intent to confirm discussed problem. Train dataset for this new CNN was around 500 images from whom around 50 images were extracted for the test data set. This new CNN could detect coordinates of the points

of interest on the driver's eyes along with the position of the driver's face on the whole frame, considering the exclusion of the REN and the SEN model.

With the current position of the camera and in respect of possible minimal and maximal distance of the driver from the camera, the eyes will not represent more than ~9% area of the whole frame as opposed to the driver's face which takes a much greater area. Those dimensions of the driver's eyes should be considered too small for one CNN to accurately predict points of interest and decide if they are closed or not. Given that the SEN model worked with the images of just the eye, where the eye is taking almost 100% area of the frame, the error margin is greater than it would be for the model that works with the whole frame. i.e., we can tolerate greater error from the output predictions of the SEN model. To support our hypothesis, we present a graphical representation of the different error margins in Fig. 7 and Fig. 8.

In Fig. 7, the green point represents the accurate, labeled, center of the eye, the blue point represents the prediction of the experimental CNN with an error margin of 3% and the red point represents the prediction of the experimental CNN with the error margin of 6%. Even with the error margin as small as 3%, the predicted center of the eye is greatly displaced from the real center, thus not accurate.

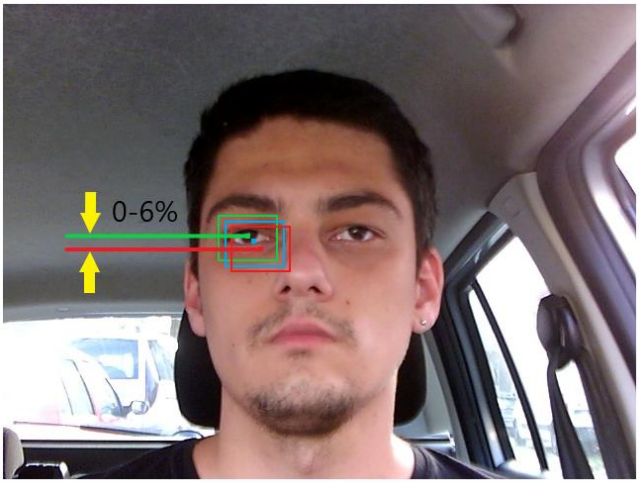

Fig. 7 Different error margins of the new experimental CNN model

In Fig. 8, the green point represents the accurate, labeled, upper center point of the eye, the blue point represents prediction of the SEN model used in our system with the error margin of 5% and the yellow point represents the prediction of the SEN model with the error margin of 10%. Even when the error margin is high as 10% prediction of the upper center point of the eye is borderline accurate.

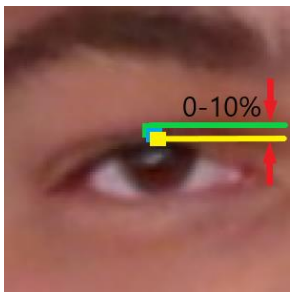

Fig. 8 Different error margins of the SEN model

Consistent with Fig. 5 and Fig. 6 analysis and Fig. 7 and Fig 8. hypothesis, we confirmed our logic. Results of the predictions over test dataset for both experimental and three-model approach are presented in Table 2, where the accuracy of predictions vary in the range from ~41% to ~86% for the experimental CNN model. With an error margin of 3%, this CNN model is not capable of reliably predicting points of interest on driver's eyes from the whole frame and thus if the eyes are open or closed. From the other perspective, a 3% error margin is too strict of a requirement for predicting a driver's face as it takes a much greater area of the whole frame as opposed to the area taken by the driver's eyes. This was one of the main reasons why we took the approach with three CNN's where each CNN model has greater error tolerance. The three-model approach we took is more complex, however, it gives more reliable and better results than the one CNN approach we discussed.

Table 2 presents the accuracy of the experimental and the three CNN model approach over the test dataset. For the simplicity of the paper only selected data is shown. Similar results were obtained for the rest of the outputs of the neural networks.

Face element	**Parameter**	**One network Accuracy** [%]	**Three network Accuracy** [%]
Face	*x* coordinate	73.91	91.19
	y coordinate	80.43	98.72
	Width	52.17	82.98
Left Eye	Center up *y* coord.	67.39	93.45
	Center *y* coord.	73.91	96.42
	Center down *y* coord.	71.74	92.26
	Left point *y* coord.	76.09	85.71
	Right point *y* coord.	86.96	90.47
Right eye	Center up *y* coord.	69.57	89.28
	Center *y* coord.	67.39	91.07
	Center down *y* coord.	71.74	90.47
	Left point *y* coord.	71.74	85.12
	Right point *y* coord.	76.09	86.91

Table 2 Accuracy of the experimental and three CNN models over test dataset

VI. Conclusion

In this paper, we proposed an application which as its building blocks relies on principles of deep learning with a set of three CNNs for real-time monitoring of driver's attention and issues a sound warning if it concludes that the driver lacks attention while driving.

All three CNNs were designed from scratch, and in response to that, we generate our datasets which were used in training. As proposed in various researches [5][6][7][8][9], deep CNNs have been proven efficient and elegant solution to the initial problem, where a lot can be achieved with bigger and more diverse datasets. We showed that they are exceptionally sensitive and with skillful handling can provide results with great precision.

We conclude and point out that this is only academic research and by no means this can be a commercial solution integrated into modern ADAS systems, however, it could be a good starting reference. The factors used to assess the attention levels are far from enough for a complete assessment and there are many more factors that are not considered. For example, driving at night can be another problem in which there should be an adequate solution for a video source that would assure undisturbed processing done by CNN's.

Proposed system demonstration

A short demonstration of the proposed system for detecting driver's drowsiness, fatigue, and inattention can be found on the following weblink:

youtu (.) be (/) ha-RSszCpQQ

Video contains a demonstration in real-life driving situations, as well as in a controlled environment.

Acknowledgments

This work was partially supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, under grants number: III44009-6 and TR32034.

References

[1] А. за безбедност саобраћаја Републички завод за статистку Републике Србије, "Статистички избештај о стању безбедности саобраћаја у Републицци Србији у 2019. години," 2019.

[2] E. Commission, "MOBILITY AND TRANSPORT, Road Safety." https://ec.europa.eu/transport/road_safety/specialist/knowledge/fatique/fatigue_and_road_crashes/conclusions_en.

[3] S. Abtahi, B. Hariri, and S. Shirmohammadi, "Driver drowsiness monitoring based on yawning detection," *Conf. Rec. - IEEE Instrum. Meas. Technol. Conf.*, no. July, pp. 1606–1610, 2011, doi: 10.1109/IMTC.2011.5944101.

[4] L. M. Bergasa, J. M. Buenaposada, J. Nuevo, P. Jimenez, and L. Baumela, "Analysing driver's attention level using computer vision," *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, pp. 1149–1154, 2008, doi: 10.1109/ITSC.2008.4732544.

[5] J. Yu, S. Park, S. Lee, and M. Jeon, "Driver Drowsiness Detection Using Condition-Adaptive Representation Learning Framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 11, pp. 4206–4218, 2019, doi: 10.1109/TITS.2018.2883823.

[6] R. Jabbar, K. Al-Khalifa, M. Kharbeche, W. Alhajyaseen, M. Jafari, and S. Jiang, "ScienceDirect The 9th International Conference on Ambient Systems, Networks, and Technologies (ANT 2018) Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques," *Procedia Comput. Sci.*, vol. 00, pp. 0–000, 2018, [Online]..

[7] B. Reddy, Y. H. Kim, S. Yun, C. Seo, and J. Jang, "Real-Time Driver Drowsiness Detection for Embedded System Using Model Compression of Deep Neural Networks," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2017-July, pp. 438–445, 2017, doi: 10.1109/CVPRW.2017.59.

[8] A. Tran and L. F. Cheong, "Two-Stream Flow-Guided Convolutional Attention Networks for Action Recognition," *Proc. - 2017 IEEE Int. Conf. Comput. Vis. Work. ICCVW 2017*, vol. 2018-Janua, pp. 3110–3119, 2017, doi: 10.1109/ICCVW.2017.368.

[9] S. K. and C. D. Y. S. Park, F. Pan, "Driver drowsiness detection system based on feature representation learning using various deep networks," no. The ACCV Workshop on Driver Drowsiness Detection from Video 2016, Taipei, Taiwan, ROC, 2016.

[10] Y. Zhang, J. Gao, and H. Zhou, "Breeds Classification with Deep Convolutional Neural Network," *ACM Int. Conf. Proceeding Ser.*, pp. 145–151, 2020, doi: 10.1145/3383972.3383975.

[11] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," no. Section 3, pp. 41.1-41.12, 2015, doi: 10.5244/c.29.41.

[12] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, no. June, pp. 2625–2634, 2015, doi: 10.1109/CVPR.2015.7298878.

[13] "Dlib C++ tollkit.", dlib.net.

[14] E. Commodari, "Novice readers: The role of focused, selective, distributed and alternating attention at the first year of the academic curriculum," *Iperception.*, vol. 8, no. 4, 2017, doi: 10.1177/2041669517718557.

[15] B. N. Manu, "Facial features monitoring for real time drowsiness detection," *Proc. 2016 12th Int. Conf. Innov. Inf. Technol. IIT 2016*, no. November 2016, 2017, doi: 10.1109/INNOVATIONS.2016.7880030.

[16] J. Xu, J. Min, and J. Hu, "Real-time eye tracking for the assessment of driver fatigue," *Healthc. Technol. Lett.*, vol. 5, no. 2, pp. 54–58, 2018, doi: 10.1049/htl.2017.0020.

[17] Ó. Cobos, J. Munilla, A. M. Barbancho, I. Barbancho, and L. J. Tardón, "Facial activity detection to monitor attention and fatigue," *Proc. Sound Music Comput. Conf.*, pp. 295–296, 2019.

[18] O. Of and M. Carriers, "PERCLOS : A Valid Psychophysiological Measure of Alertness As Assessed by Psychomotor Vigilance," *October*, vol. 31, no. 5, pp. 1237–1252, 1998, [Online].

[19] Raimi Karim, "Illustrated: 10 CNN Architectures, A compiled visualisation of the common convolutional neural networks," *Towards Data Science*. https://towardsdatascience.com/illustrated-10-cnn-architectures-95d78ace614d#bca5.