Statistical Model for Driver Drowsiness Detection based on blinking eyes and yawn behavior

Wanxin Xu

1. Introduction

Driving involves executing a series of actions, perceiving the situation, and making quick and accurate decisions. Monitoring attention state is considered as one of the most important parameters of safe driving. [1] Fatigue and distraction can slow a person's reaction time and prevent them from driving effectively. Some progress has been made in the field of driver monitoring, especially in driver drowsiness detection. Drowsiness symptoms include yawning [2], eyelid closure [3] and so on. Various symptoms and degree of fatigue may occur in humans, so a single symbol may not be used for drowsiness detection independently and accurately.

This report focuses on machine learning models for driver drowsiness detection based on physical features such as blink frequency, eye closure duration and whether the tester is yawning.

Currently, there have been many technologies employed to address this drowsiness detection problem for automobile drivers. Those technologies could be categorized into three groups which are the pattern of car driving, the biological and psychological features of drivers and computer vision techniques for driver monitoring.

In the first group of techniques, some of the recent research focused on calculating the degree to which the car departed from the lanes, the movement of steering wheel, the acceleration of the car [4][5], while the second group of techniques worked on electrical biological signals such as Electroencephalography (EEG), Electrocardiography (ECG) and Electrooculogram (EOG) [6]. However, these two kinds of methods have many limitations. The first group’s solution only applies to limited driving conditions and the robustness is not proved, while for the second group’s solution, it is difficult to build up real-world applications since it is inconvenient and uncomfortable to make the driver wear various nonportable sensors for data collection during the time whenever they are driving.

Therefore, computer vision and pattern recognition-based method is becoming more and more important on this problem. The computer vision techniques mainly focus on detecting the status of eyes (open or closed), the yawning pattern and the overall movement of head.

2. Related Work

In order to better highlight the strengths of my insights, comparison between the existing methods and new idea are necessary. As mentioned in the introduction section, all technologies utilized to detect driver's drowsiness can be classified into the following three groups.

2.1 Features on the Pattern of Driving

As stated in section 1, the first category of driver drowsiness detection methods is based on the measurement of movement of steering wheel, lane departure degree and lateral position. However, the driving pattern-based method depends highly on the driving technique of the specific driver, the road condition and characteristics of the specific automobile. First, based on the research of Krajewski et al. [7], we find that there is extremely high correlation between the fatigue status and the micro adjustments, which are common practice for drivers to drive within the lane. Nevertheless, this conclusion is only applied to major straight roads with normal lanes. In rural area, we cannot expect any driver either to always adjust the steering wheel with the same degree or drive straight along a fixed lane. Second, some research involves the monitoring technology on detecting the lane deviation. Similarly, this technology is based on the assumption that the driver is well trained and road condition is good to recognize.

2.2 Psychophysiological Characteristics of Drivers

Another existing technology necessitates the physical sensors to connect to the driver, such as EEG, ECG and EOG. For instance, EEG includes three signal metrics, alpha, delta and theta signals, which are used to measure a driver's brain activity. When a driver is in fatigue state, delta and theta signals spike up and alpha signal increases slightly. Meanwhile, ECG is used to measure the driver's heart beating while EMG

reflects whether a driver's muscle is tight or slack. The working principles of EEG, ECG and EOG are quite similar, and they perform much higher accuracy than first group of technologies. Unfortunately, few drivers are willing to attach these kinds of sensors to the body, because they are uncomfortable to wear and might even disturb drivers.

2.3 Facial Feature Extraction Using Computer Vision

Over the past few years, computer vision technology has been increasingly popular. To detect the fatigue status of the driver, the critical steps are which facial features should we extract and how to extract them to the server. Previous research involved the recognition algorithms on eye closure, head movement, gaze or facial expression. For example, we could calculate the blink frequency of driver's eyes to determine whether the driver is actually sober. Moreover, yawning is another important measurement for drowsiness. The opening degree of driver's mouth and the duration should be calculated to recognize the yawning behavior, and Viola-Jones Object Detection Algorithm is often performed [10] to detect the mouth and yawn behaviors.

2.4 Drowsiness Detection Using Deep Learning

Deep learning is widely used for complicated problems that traditional technologies cannot solve. For example, Convolutional Neural Networks (CNN) is an excellent algorithm to classify the image, detect the object, recognize the emotion and segment the scene. Based on the research from Bhakti Baheti, Suhas Gajre and Sanjay Talbar, a CNN-based system was proposed that not only detects the distracted driver but also identifies the cause of distraction. The research involving CNN manifestly enhance both the detection accuracy and speed. However, deep learning also encountered a big bottleneck currently, since these algorithms perform limitedly on embedded systems, which come with relatively low computational complexity.

3. Drowsiness-related Dataset

As is stated in section 2, the state-of-art computer vision solution for drowsiness detection includes deep learning methods like fast RCNN. However, the deep learning method requires large amount of video data and high-performance processor, which are not available in this project. Therefore, only statistical learning models were trained in this project rather than neural network models. There are some open-source datasets for drowsiness detection, such as the DROZY database [8].

DROZY, is a database containing various types of drowsiness-related data (signals, images, etc.) In order to get the biological signal features, sensors were attached to the subjects' faces and that could affect accurate recognition for computer vision techniques, as shown in Figure 1.

Figure 1. Sample Data in DROZY Database

Therefore, custom collected data, shown in Figure 2, was used in this project. Details will be discussed in Section 5.

Figure 2. Sample Custom Data: Non-drowsy vs Drowsy

4. Material and Methods

4.1 Software and Packages

This project is accomplished by using OpenCV and Dlib library [9]. OpenCV was used to input the video file and to preprocess the raw data, while Dlib library was installed in order to get facial landmarks from the subject.

The Haar cascade classifiers provided by OpenCV (i.e. the Viola-Jones detectors), as discussed in Section 2.3, are widely used in computer vision techniques for object detection. However, when detecting faces in OpenCV, it is difficult to tune the parameters called cv2.detectMultiScale and these parameters for different images may vary, which causes the performance to drop.

Instead, the Dlib library uses a pre-trained face detector which is based on a modification to the Histogram of Oriented Gradients + Linear SVM method for object detection.

The facial landmark detector included in the Dlib library is an implementation of the One Millisecond Face Alignment with an Ensemble of Regression Trees paper by Kazemi and Sullivan. [11] This method

uses a set of facial landmarks marked on training images. These images are manually labeled, specifying specific (x, y) coordinates around each facial structure. This method also uses the probability of the distance between the input pixel pairs. Based on these training data, an ensemble of regression trees is trained, directly from the pixel intensity itself. Therefore, no feature extraction is performed. As a result, this method could detect facial landmarks in real-time with high accuracy of prediction.

In this project, this Dlib model was pre-trained on the iBUG 300-W face landmark dataset (https://ibug.doc.ic.ac.uk/resources/facial-point-annotations/[12], and the pre-trained model was obtained from http://dlib.net/files/shape_predictor_68_face_landmarks.dat.bz2. This pre-trained model could find frontal human faces in an image and estimate their pose. The pose takes the form of 68 landmarks. These are points on the face such as the corners of the mouth, along the eyebrows, on the eyes, and so forth. The visualization of 68 facial landmarks is shown in Figure 3 and the result of facial landmark detection on the custom data is shown in Figure 4.

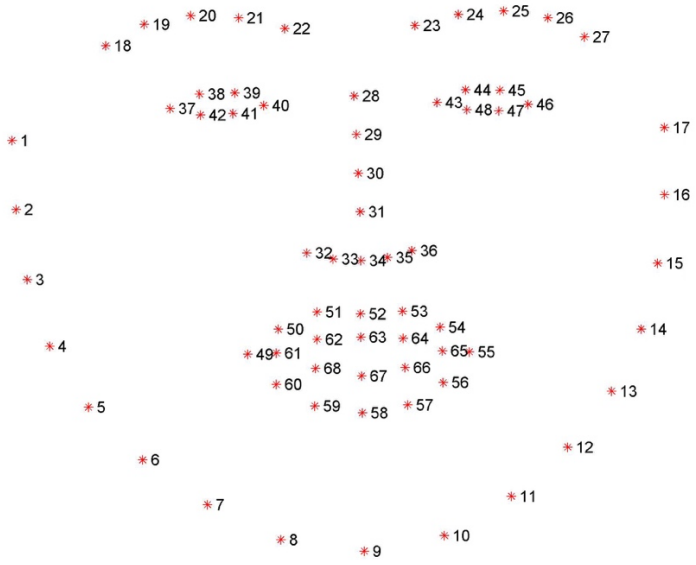

Figure 3. Visualization of 68 facial landmark coordinates from the iBUG 300-W dataset

Figure 4. Live Visualization of 68 facial landmark coordinates on Custom Data

In summary, to detect drowsiness using facial landmarks first need two steps:

1) Localize face in the image or video frame. Haar cascades and classic HOG + Linear detectors are both fine for this step.
2) Utilize the shape detector (facial landmark detector) to extract the coordinates of specific structures (eyes, mouth, nose and so on) in the face region of interest (ROI) which is a bounding box we can obtain in step 1.

4.2 Baseline method

Key Insights. The baseline method in this project is a threshold-based approach which basically detects drowsiness by measuring the duration of eye closure. The simplest way to detect if someone is going to nap is to see if the person's eyes have been closed for a while. If so, we assume the person will be likely to nod off and become drowsy. To determine if a person's eyes are closed or not, we computed the metric

called the eye aspect ratio (EAR), introduced in the paper called Real-Time Eye Blink Detection Using Facial Landmarks. [13]

$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

Figure 5. Eye aspect ratio equation.

As stated in section 4.1, we could get the coordinates of all facial landmarks using the facial detector and shape detector. In the 68-facial-landmark representation, each eye has six (x,y) coordinates, shown in Figure 6. Figure 5 shows the EAR equation introduced in the paper, where p1, p2, p3, p4, p5, p6 are all 2-dimensional facial landmark locations, visualized in Figure 6.

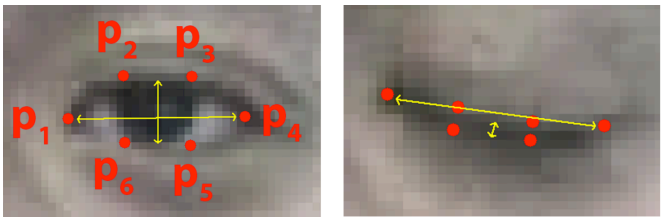

Figure 6. Coordinates of eyes

The numerator of this equation calculates the distance between the vertical eye markers (the height of the eye), and the denominator calculates the distance between the horizontal eye markers (the width of the eye). Since there is only one set of horizontal points and two sets of vertical points, the denominator is weighted by 2.

As stated in the paper [13], experiments show that when eyes open, the eye-to-eye ratio is almost constant, but when the eyes blink or close, it quickly drops to around zero. Therefore, the ratio of these distances may help us identify if someone is closing his eyes.

As shown in the right-hand side of Figure 6, once the eye is closed, the vertical distance of landmarks divided by the horizontal distance is close to 0.

The advantage of the EAR feature is that it is almost insensitive to different people and different head poses. Although the aspect ratio of the open eyes has a small variance in even the same individual, it is

completely invariant to the uniform scaling of the image and the in-plane rotation of the face. Since the closure of eyes is synchronized between the two eyes, the EAR of the two eyes is averaged.

Details and Implementation. The baseline method is implemented in the Baseline.py file in the project. First of all, frontal facial detector was loaded using Dlib library and facial landmark detector was created from the pre-trained model 'shape_predictor_68_face_landmarks.dat'. Different facial regions are mapped to different indexing. For example, the right eye uses index from 36 to 42, left eye uses from 42 to 48 and the mouth could be accessed through points 48 to 68. The raw video data is preprocessed and resized to a width of 450 pixels and converted to greyscale.

Second, we iterate through the video, run the HOG-based face detector on each frame, which will return all of the face ROIs. Then we extract the facial coordinates by applying the created facial landmark detector.

Last, we calculate the Euclidean distance of vertical and horizontal eye landmarks and compute the EAR value, compare it to a reasonable EAR threshold. If the total number of consecutive frames where the person has his eyes closed exceeds a certain number of frames, the person is considered to be drowsy.

4.3 Statistical learning method

Key Insights. The baseline method sets a threshold for the number of consecutive frames where the EAR value is close to 0 which represents how long the person has closed his eyes for. However, this simple threshold-based method performs detection based on one specific feature. In the experiment, this approach sometimes encountered false-alarms and sometimes showed a non-drowsy result in the case where there were some other visual cues indicating that the subject was drowsy. For example, the driver was yawning or blinking more frequently to prevent himself from falling asleep.

Therefore, statistical modeling was used here to find the relation between drowsiness and the various expression of facial landmarks. The features used in the model include the blink rate, the average eye closure degree, the maximum frames for eye closure, the duration when human face exists, the total

blinks, whether the tester yawns. Instead of setting a hard threshold for the frames for eye closure, the statistical method uses this as one of the features.

Blink rate. Some research shows that not only episodes of slow eye closure occur in response to increased drowsiness, but changes in the frequency, amplitude, and duration of blinks may also relate to the fatigue status. [14] Therefore, we include the blink rate feature in the model. The blinks are counted by how many times the tester have the eyes closed, which is represented by the EAR value mentioned above. We just calculated how many times the EAR value drops to around zero. However, in the experiment, if the driver is turning over and looking at the rearview mirror, he will turn the face to the side, at this time there will not be any frontal faces detected and the EAR scheme won't work as well. Therefore, the equation for blink rate is $\frac{\text{totalBlinks}}{\text{validDuration}}$, where validDuration means that the total time when there is a valid frontal human face detected.

The average eye closure degree. When a person is becoming more and more drowsy, it is possibly that his blink duration becomes longer even if the maximum blink duration does not exceed a certain length. Therefore, here, the average eye closure degree is computed to evaluate if the driver is sober. The average eye closure degree is calculated by $\frac{\sum_{validFrames} EAR}{validFrames}$, where EAR means the EAR value of each valid frame, a valid frame means that there is a valid frontal face detected in this frame.

The maximum frames for eye closure. This feature is similar to the metric used in the baseline. Instead of using a rule-based method to determine drowsiness or not, the frames for eye closure is used as just one of the features to train the classifier here.

The duration when human face exists. As stated above, it is possible to not detect any faces or eye landmarks when the driver is moving or turning around. The time for no face detected should be subtracted from the total duration. To some degree, no frontal face detected often means the driver is moving, not in the asleep status, which indicates non-drowsiness.

The total blinks. The total blinks are computed based on the times of changes in EAR values, which could be then used to compute the blink rate/frequency.

Whether the tester yawns. In addition to all the eye features listed above, mouth feature is another important way to identify drowsy cases. In this project, an approach, which is similar to blink detection, is utilized to detect yawns.

First, all landmark coordinates related to mouth could be accessed by the shape detector.

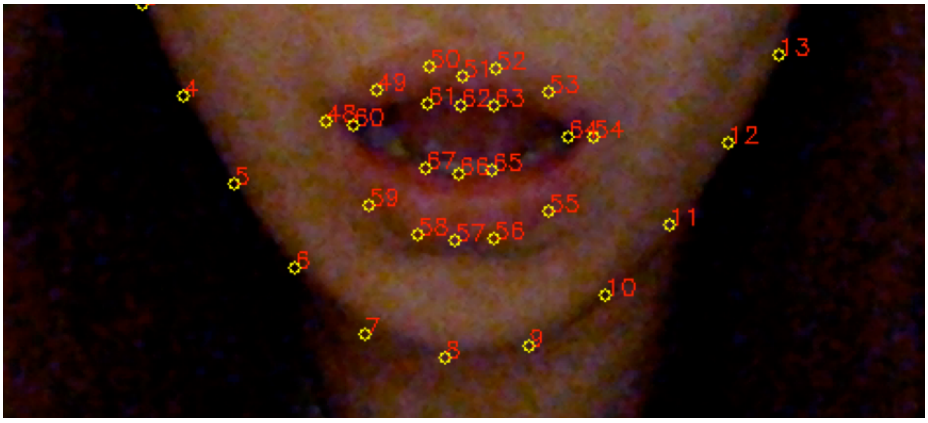

Figure 7. Landmark points for mouth

Second, we calculated the center coordinates for both top lip and bottom lip by respectively averaging the coordinates of points 50, 51, 52 and points 56, 57, 58 so that we could computed the vertical lip distance as the EuclideanDist($topLipCenter, BottomLipCenter$). Similarly, the length of lips was calculated as EuclideanDist($Point\ 60, Point\ 64$). The yawn behavior is defined as when:

$$\text{EuclideanDist}(topLipCenter, BottomLipCenter) > 0.5 \times \text{EuclideanDist}(Point\ 60, Point\ 64)$$

$$and \qquad yawnFrameCount >= YAWN\_FRAMES,$$

Where yawnFrameCount represents the consecutive frames where the ratio of height and width of mouth exceeds 0.5 and YAWN_FREAMES is a threshold empirically set to 48.

Summary. The first five features are related to eye/eyelids status, which are computed in eyeFeature.py file, while the last feature is based on mouth features, which is computed in mouthFeature.py. The eye features are all numeric features, while the mouth feature is Boolean.

5. Experiment Setup

One subject participated in the experiment and he was asked to drive in different conditions: on a straight road, on freeway, take a turn, reverse, pull over, etc. and drive at different time during a day under different light condition. Different forms of drowsiness are included in the custom data, such as slow eye movement, head tilting, eye closure, yawning, etc.

Figure 8. Intra-class variance: yawn with eyes open, eye closure, lower head, head tilt, yawn with eyes closed

The videos were taken by front camera of iPhone 6 installed at the car dash and the videos were recorded at 30 frames per second of size 720 x 1280. In order to make the dataset more uniform, the original videos were preprocessed into video clips, each of which has a length of 4 seconds, a total of 120 frames and a size of 450 x 800. This step is accomplished in videoClip.py using OpenCV library.

After preprocessing the original data, the dataset contains a total of 122 samples, 27 of which are drowsy cases and 95 of which are non-drowsy cases. If a ZeroR classifier is applied, the accuracy would be 77.87%.

6. Evaluation

6.1 Metrics and Experimental Results

Our baseline (rule-based) method was directly tested on the whole dataset, while the statistical model was validated by using 10-fold cross validation. To evaluate the performance, a set of standard performance measures were used including the accuracy, the F1 score, area under ROC curve (AUC).

$$F_1 = \left(\frac{\mathrm{recall}^{-1} + \mathrm{precision}^{-1}}{2} \right)^{-1} = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$

F1-Score is the harmonic mean of precision and recall. It is an overall measure of a model's accuracy that combines precision and recall. A good F1 score means that the model has low false positives and low false negatives.

The area under the curve (often referred to as simply the AUC) is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one. AUC provides an aggregate measure of performance across all possible classification thresholds. It is scale invariant and classification threshold invariant. [15]

Methods	Accuracy	F1 – Score	AUC
Baseline – Threshold Method	85.2459%	84.2610%	73.2943%
SVM + Eye feature + Mouth Feature	90.1640%	89.5073%	80.4288%

Table 1. Evaluation Results

The SVM classifier with an RBF kernel and penalty parameters C = 1000, γ = 0.001 obtained the best result. Those parameters are determined by Cross Validation Grid Search.

6.2 Discuss

As is shown in Table 1 in Section 6.1, the statistical model using SVM and hand-crafted features including both eye features and mouth features overperformed the threshold-based baseline. The total accuracy reached 90.1640%, compared with an accuracy of 85.2459% with baseline. This experiment verifies features including blinking frequency, eye closure degree, eye closure time and yawns would contribute much to the drive drowsiness detection. This model could also be employed for real-time drowsiness detect as we could create a temporal window to calculate these statistical features and get the prediction from the classifier.

However, the size of custom collected dataset is relatively small, and it is quite imbalanced as the non-drowsy cases are much more than the drowsy cases. For future study, a larger and more comprehensive dataset would be needed. In addition, only limited number of subjects participated in the experiment. It is a personalized model, so for different individuals, the results may vary.

Computer Vision techniques encountered many difficulties when dealing with this problem. It is hard to ensure the robustness of system against subjects from different ethnicities, races, genders, various illumination conditions and partial occlusion (glasses, hair, etc.)

6.3 Further Ideas

Head Posture. During the experiment, we found that head tilt might be related to drowsiness and fatigue. That is to say, we could track the subject's head inclination. When head's inclination exceeds a certain time T and a predefined angle X, that means, his gaze direction is detached from the wheel and he is probably drowsy and may need a nap.

Real-time Alertness. This experiment was conducted on a series of video clips with a length of 4 seconds, so for real-time alertness, we could apply a sliding time window and keep updating the statistical features.

Robustness. Besides, there are still many other features which may help detection such as the ratio of the amplitude to velocity of eyelid closure. More data will be needed in order to increase the robustness of the

system. Also, deep learning methods may show strengthens in extracting deeper features and enhance the overall robustness.

7. Conclusion

Driver drowsiness is one of the major causes of traffic accidents. This report surveys the literature on driver drowsiness detection and classifies the existing methods into three groups: the driving pattern-based method, the biological signal feature-based methods and vision-based methods. However, monitoring the driving pattern of the vehicle only applied to limited number of cases where the road condition and driver quality is great, and biological signal feature-based methods cannot be easily applied for real-world applications. Computer vision methods become popular in this field and both statistical models and deep neural network models work well on this drowsiness detection problem.

There are few open-source datasets specifically targeted on driver drowsiness and the existing dataset for driver drowsiness contains not only video data but also the ECG/ECG biological signals which are not useful in this project. Therefore, I used my custom collected data in this project. The state-of-art computer vision solution for drowsiness detection includes deep learning method like fast RCNN. Because of the relatively small size of my dataset, only light-weighted statistical learning models are tried in this project.

My baseline is a rule-based method in which I set a threshold for the number of consecutive frames where the driver's eye aspect ratio is below a constant value. Based on the rule-based method, I extracted features of blinking rate, the closure degree of eyes, the maximum time of eye closure, yawns in the collected videos and manually labeled the video clips with ‘Drowsy’ and ‘Conscious’. Those features were used to train an SVM with a radial basis function kernel and the model achieved an accuracy of 90.1640% and an F1-Score of 89.5073% which outperformed the baseline.

Those eye features and mouth features were obtained based on the HOG + Linear SVM face detector and the facial landmark detector created from an ensemble of regression trees trained from the pixel intensity.

In order to make the model more robust, more data is needed to make the dataset more balanced and more features could be tested. We could verify the performance of model in different situations (e.g. different lighting conditions) and have more subjects conduct the experiment.

8. Code and Instruction

The whole project and custom dataset are accessible via github.com/WanxinXu27/DrowsinessDetect.

Raw videos are in the /videos directory and the video clips after processing is in /data directory. Manually annotated data is in the /goundtruth directory. The feature files: eyeFeatures.csv and mouthFeatures.csv are both located in /output folder. Run SVM_model.py then the results of this model will automatically show.

9. Reference

[1] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," Hum. Factors, J. Hum. Factors Ergonom. Soc., vol. 37, no. 1, pp. 32–64, 1995.

[2] M. Kamienska-Zyła and K. Prync-Skotniczny, "Subjective fatigue symptoms among computer systems operators in Poland," Appl. Ergonom., vol. 27, no. 3, pp. 217–220, 1996.

[3] R. Schleicher, N. Galley, S. Briest, and L. Galley, "Blinks and saccades as indicators of fatigue in sleepiness warnings: Looking tired?" Ergonomics, vol. 51, no. 7, pp. 982–1010, 2008.

[4] K. Mattsson, "In-vehicle prediction of truck driver sleepiness. Lane positions variables," M.S. thesis, Division of Media Technology, Dept. of Computer Science and Electrical Engineering, Lulea Univ. of Technology, Sodertalje, Sweden, 2007.

[5] H. Malik, F. Naeem, Z. Zuberi, and R. ul Haq, "Vision based driving simulation," in Proc. 2004 Int. Conf. Cyberworlds, 18–20 Nov. 2004, pp. 255–259

[6] Z. Mardi, S. N. Ashtiani, and M. Mikaili, "EEG-based drowsiness detection for safe driving using chaotic features and statistical tests," J. Med. Signals Sens., vol. 1, pp. 130–137, 2011.

[7] J. Krajewski, D. Sommer, U. Trutschel, D. Edwards and M. Golz, "Steering Wheel Behavior Based Estimation of Fatigue", in Proceedings of the Fifth International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design, pp. 118-124

[8] Q. Massoz, T. Langohr, C. Francois, J. G. Verly, “The ULg Multimodality Drowsiness Database (called DROZY) and Examples of Use, WACV 2016

[9] Dlib Library Documentation: http://dlib.net/optimization.html

[10] B. Hariri, S. Abtahi, S. Shirmohammadi, and L. Martel, “A yawning measurement method to detect driver drowsiness,” Distrib. Collab. Virtual Environ. Res. Lab., Univ. Ottawa, Ottawa, ON, Canada, 2011

[11] Vahid Kazemi and Josephine Sullivan, One Millisecond Face Alignment with an Ensemble of Regression Trees, CVPR 2014

[12] C. Sagonas, E. Antonakos, G, Tzimiropoulos, S. Zafeiriou, M. Pantic. 300 faces In-the-wild challenge: Database and results, Image and Vision Computing (IMAVIS), Special Issue on Facial Landmark Localisation "In-The-Wild". 2016.

[13] Tereza Soukupova and Jan Cech, Real-Time Eye Blink Detection using Facial Landmarks, 21st Computer Vision Winter Workshop, Luka Cehovin, Rok Mandeljc, Vitomir Struc (eds.), Rimske Toplice, Slovenia, February 3–5, 2016

[14] Wilkinson VE, et al. (2013) The accuracy of eyelid movement parameters for drowsiness detection. J Clin Sleep Med 9(12):1315–1324

[15] CX Ling, J Huang, H Zhang AUC: a statistically consistent and more discriminating measure than accuracy. - Ijcai, 2003 - cling.csd.uwo.ca