Technologies for Autonomous Vehicles
Assignment 1 - Report

1rd Emanuele Giuseppe Siani
s330980@studenti.polito.it

***Abstract*—This report presents an AI-driven Driver Monitoring System (DMS) leveraging MediaPipe, which monitors driver drowsiness and distraction. The system computes the Eye Aspect Ratio (EAR) and PERCLOS using a dynamic calibration and timer-based approach, while head and eye orientations are estimated via a PnP method. Experimental results suggest that the integrated framework effectively detects driver inattention.**

***Index Terms*—AI, Driver Monitoring System, Eye Aspect Ratio, PERCLOS**

I. Introduction

This report provides a comprehensive commentary on the code implementing an AI-driven Driver Monitoring System (DMS) using MediaPipe.

To realize the driver monitoring features, the system leverages the calculation of the Eye Aspect Ratio (EAR) and the Roll-Pitch-Yaw angles of both the face and the eyes.

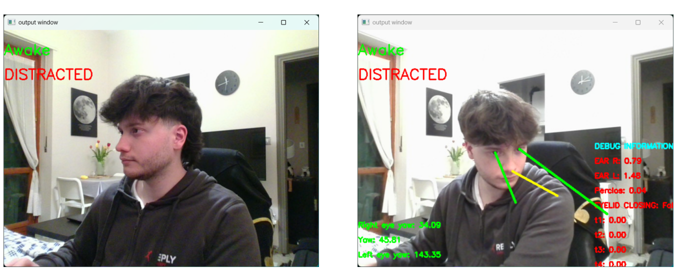
Specifically, the DMS monitors the following conditions:

1) **Drowsiness**: The driver is considered drowsy if the computed EAR remains above 0.8 for more than 10 consecutive seconds.
2) **Distraction**: The driver is considered distracted if the absolute value of the yaw angle of both the face and the eyes exceeds 30 degrees.

In addition, the implemented DMS computes the PERCLOS 80 metric by adopting the algorithm proposed in [1].

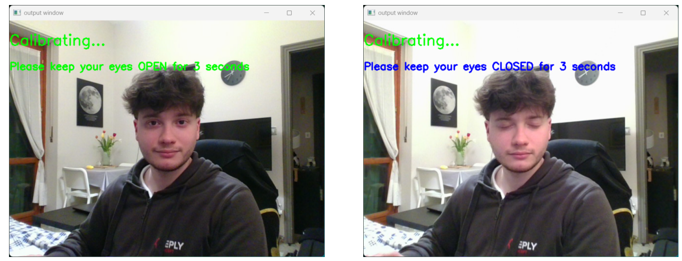
II. Code Comment

The following subsections will serve as guidance for the implementation of EAR and PERCLOS implementation. **N.B.**: Set the DEBUG flag to True to show the computed values of EAR, the PERCLOS and its needed times. Morover, guiding lines for face and eyes orientation will be drawn.



(a) Regular View (b) Debug view

Fig. 1: Regular vs Debug view



(a) Calibration phase: Open eyes threshold (b) Calibration phase: Closed eyes threshold

Fig. 2: Calibration phase

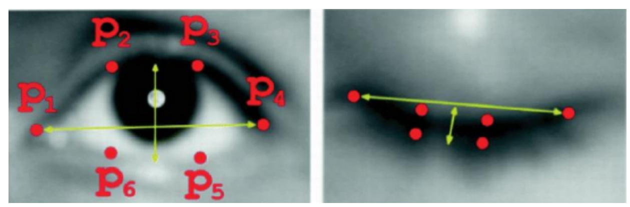


Fig. 3: Eye Aspect Ratio (EAR)

A. Calibration Phase

The Eye Aspect Ratio (EAR) is computed using the following equation:

$$EAR = \frac{|Y_2 - Y_6| + |Y_3 - Y_5|}{2 \cdot |X_1 - X_4|} \quad (1)$$

The numerator represents the sum of the vertical (Y-axis) distances between the points P_2, P_6 and P_3, P_5, while the denominator corresponds to twice the horizontal (X-axis) distance between the points P_1 and P_4.

When a person blinks or closes their eyes, the horizontal distance between P_1 and P_4 remains nearly constant, while the vertical distances between P_2 and P_6, and between P_3 and P_5, decrease significantly.

Since the absolute value of the EAR may vary across individuals due to differences in eye geometry, a dynamic calibration step is performed to personalize the thresholds. The user is asked to keep their eyes open for the first 3 seconds to collect samples for the open-eye condition, followed by 3 seconds with eyes closed to gather data for the closed-eye condition. During these intervals, the system stores the computed raw EAR values for both eyes.

Note: For the open eyes calibration phase, the system requires that the eyes are *naturally open*. Excessively wide apertures will lead to incorrect calibrations.

The thresholds for open and closed states are computed separately for each eye. Specifically, the *open-eye threshold* is defined as the mean of the 100 largest EAR values recorded during calibration phase, while the *closed-eye threshold* is defined as the mean of the 100 smallest EAR values during calibration phase.

Let $\{EAR_L^{(i)}\}$ and $\{EAR_R^{(i)}\}$ be the sets of recorded EAR values for the left and right eyes during calibration. Define: - O_L and O_R as the open-eye thresholds for the left and right eyes; - C_L and C_R as the closed-eye thresholds.

These thresholds are computed as follows:

$$O_L = \frac{1}{100}\sum_{i=1}^{100} EAR_L^{(i)\uparrow} \tag{2}$$

$$O_R = \frac{1}{100}\sum_{i=1}^{100} EAR_R^{(i)\uparrow} \tag{3}$$

$$C_L = \frac{1}{100}\sum_{i=1}^{100} EAR_L^{(i)\downarrow} \tag{4}$$

$$C_R = \frac{1}{100}\sum_{i=1}^{100} EAR_R^{(i)\downarrow} \tag{5}$$

Here, $EAR^{(i)\uparrow}$ and $EAR^{(i)\downarrow}$ denote the i-th largest and smallest values, respectively, from the sorted calibration set.

At each timestep, the raw EAR values are normalized to fall within the range $[0, 1]$ using the following equations:

$$EAR_L^{\text{norm}} = \frac{EAR_L - C_L}{O_L - C_L} \tag{6}$$

$$EAR_R^{\text{norm}} = \frac{EAR_R - C_R}{O_R - C_R} \tag{7}$$

Note: Thresholds are computed separately for each eye to accommodate potential asymmetries, such as in individuals with ptosis (drooping eyelid).

B. PERCLOS 80 Calculation

The PERCLOS 80 metric (Percentage of Eye Closure) is computed by monitoring the normalized Eye Aspect Ratio (EAR) over time using a timer-based method that marks distinct phases of the eye closure event.

First, the algorithm continuously appends the current time and the corresponding normalized EAR values of both eyes to their respective histories. Once 10 consecutive samples have been collected, an approximation of the EAR derivative is computed for each eye over the interval $\Delta t = t_{10} - t_1$:

$$D_R = \frac{EAR_R^{(10)} - EAR_R^{(1)}}{\Delta t}, \tag{8}$$

$$D_L = \frac{EAR_L^{(10)} - EAR_L^{(1)}}{\Delta t}. \tag{9}$$

A sufficiently negative derivative (empirically, $D_R < -0.5$ and $D_L < -0.5$) indicates the onset of an eye closure. At this point, an `EYELID_CLOSING` flag is set and timer t_1 is started to mark the beginning of the event.

After detecting the closing motion, the mean normalized EAR is computed as:

$$\overline{EAR} = \frac{EAR_R^{\text{norm}} + EAR_L^{\text{norm}}}{2}. \tag{10}$$

The algorithm then controls further timers based on the value of $\overline{EAR}$:

- When $0.2 < \overline{EAR} < 0.8$, timer t_2 is started, while timer t_1 is stopped, to capture the transition phase.
- When $\overline{EAR} < 0.2$, indicating that the eyes are in a deeper closure phase, timer t_3 (marking the period of maximum closure) is initiated and simultaneously timer t_4.
- If $\overline{EAR}$ raises again above 0.2, timer t_3 is stopped.
- When $\overline{EAR}$ raises above 0.8, signaling that the eyes are almost fully open, timer t_4 is stopped, the `EYELID_CLOSING` flag is reset, and the PERCLOS value is calculated.

The PERCLOS is then computed, as suggested in [1], as the ratio of the effective eye closure duration to the total duration of the eye closure event:

$$\text{PERCLOS} = \frac{t_3 - t_2}{t_4 - t_1}, \tag{11}$$

provided that the elapsed time values are all greater than zero. In the event that the computed PERCLOS is negative (when the eye closure is too fast for the script), its value is reset to zero. Finally, all timers (t_1, t_2, t_3, and t_4) are reset to prepare for the next measurement cycle.

This dynamic timer-based approach provides a real-time estimate of the percentage of eye closure, which is a critical indicator for detecting driver drowsiness.

C. Driver Distraction Detection

The system detects driver distraction by estimating the 3D head and eye orientations using a PnP (Perspective-n-Point) approach. The key steps are as follows:

1) **Camera Calibration:** The camera matrix is defined by:

$$K = \begin{bmatrix} f & 0 & \frac{h}{2} \\ 0 & f & \frac{w}{2} \\ 0 & 0 & 1 \end{bmatrix}, \tag{12}$$

   where f is the focal length, and w and h are the width and height of the image, respectively.
2) **Pose Estimation:** The 3D facial and eye landmark coordinates are used with `cv2.solvePnP` to obtain rotation vectors, which are converted to rotation matrices via `cv2.Rodrigues`. The matrices are then decomposed to extract the Euler angles (pitch, yaw, and roll) for both the face and the eyes.

3) **Distraction Determination:** The driver is flagged as distracted if the absolute yaw angles for the face, left eye, and right eye all exceed $30°$:

$$|\text{yaw}| > 30° \quad \wedge \quad |\text{yaw}_{\text{left}}| > 30° \quad \wedge \quad |\text{yaw}_{\text{right}}| > 30°. \tag{13}$$

III. Conclusion

In conclusion, the presented Driver Monitoring System effectively integrates real-time EAR calibration, PERCLOS computation, and 3D head pose estimation to assess driver drowsiness and distraction. The system utilizes a dynamic calibration approach to personalize eye closure thresholds, ensuring that variations in individual eye geometry are appropriately managed. By combining timer-based analysis with the precise calculation of facial and ocular orientation through PnP methods, the system reliably distinguishes between states of focus and distraction..

References

[1] Zulkarnanie., Enhancements to PERCLOS Algorithm for Determining Eye Closures, 2022 International Conference on Future Trends in Smart Communities (ICFTSC), pp.76–81