



DRIVER DROWSINESS DETECTION SYSTEM USING OPEN COMPUTER VISION

R. Manikandan*, S. Abilash, C. Agilakalanchian*** & P. Tamilselvan******

Rajiv Gandhi College of Engineering and Technology, Puducherry

Cite This Article: R. Manikandan, S. Abilash, C. Agilakalanchian & P. Tamilselvan,

“Driver Drowsiness Detection System Using Open Computer Vision”, International Journal of Current Research and Modern Education, Volume 3, Issue 1, Page Number 410-414, 2018.

Copy Right: © IJCRME, 2018 (All Rights Reserved). This is an Open Access Article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract:

In recent years driver fatigue is one of the major causes of vehicle accidents in the world. A direct way of measuring driver fatigue is measuring the state of the driver i.e. drowsiness. So it is very important to detect the drowsiness of the driver to save life and property. This project is aimed towards developing a prototype of drowsiness detection system. This system is a real time system which captures image continuously and measures the state of the eye according to the specified algorithm and gives warning if required. Though there are several methods for measuring the drowsiness but this approach is completely non-intrusive which does not affect the driver in any way, hence giving the exact condition of the driver. For detection of drowsiness the per closure value of eye is considered. So when the closure of eye exceeds a certain amount then the driver is identified to be sleepy. For implementing this system several OpenCv libraries are used including Haar-cascade. The entire system is implemented using microcontroller.

Key Words: Drowsiness Detection, ADAS, Face Detection and Tracking, Eyes Detection and Tracking, Eye State & PERCLOS

1. Introduction:

Currently, transport systems are an essential part of human activities. We all can be victim of drowsiness while driving, simply after too short night sleep, altered physical condition or during long journeys. The sensation of sleep reduces the driver's level of vigilance producing dangerous situations and increases the probability of an occurrence of accidents. Driver drowsiness and fatigue are among the important causes of road accidents. Every year, they increase the number of deaths and fatalities injuries globally. In this context, it is important to use new technologies to design and build systems that are able to monitor drivers and to measure their level of attention during the entire process of driving. In this paper, a module for ADAS (Advanced driver assistance System) is presented in order to reduce the number of accidents caused by driver fatigue and thus improve road safety. This system treats the automatic detection of driver drowsiness based on visual information and artificial intelligence. We propose an algorithm to locate, track and analyze both the driver face and eyes to measure PER- CLOS (percentage of eye closure). The remainder of this paper is organized as follows, Section 2 presents the related works, Section 3 presents the proposed system and the implementation of each block of the system, the experimental results are shown in section 4 and in the last section conclusions and perspectives are presented.

2. Related Works:

Some efforts have been reported in the literature on the development of the not-intrusive monitoring drowsiness systems based on the vision. Malla et al. [1] develop a light-insensitive system. They used the Haar algorithm to detect objects [2] and face classifier implemented by [3] in OpenCV [4] libraries. Eye regions are derived from the facial region with anthropometric factors. Then, they detect the eyelid to measure the level of eye closure. Vitabile et al. [5] implement a system to detect symptoms of driver drowsiness based on an infrared camera. By exploiting the phenomenon of bright pupils, an algorithm for detecting and tracking the driver's eyes has been developed. When drowsiness is detected, the system warns the driver with an alarm message. Bhowmick et Kumar [6] use the Otsu thresholding [7] to extract face region. The localization of the eye is done by locating facial landmarks such as eyebrow and possible face center. Morphological operation and K-means issued for accurate eye segmentation. Then a set of shape features are calculated and trained using non-linear SVM to get the status of the eye.

Hong et al. [8] define a system for detecting the eye states in real time to identify the driver drowsiness state. The face region is detected based on the optimized Jones and Viola method [2]. The eye area is obtained by an horizontal projection. Finally, a new complexity function with a dynamic threshold to identify the eye state. Tian et Qin [9] build a system that checks the driver eye states. Their system uses the Cb and Cr components of the YCbCr color space. This system locates the face with a vertical projection function, and the eyes with a horizontal projection function. Once the eyes are located the system calculates the eyes states using a function of complexity. Under the light of what has been mentioned above, the identification of the driver drowsy state given by the PERCLOS is generally passed by the following stages:

- ✓ Face detection,
- ✓ Eyes Location,
- ✓ Face and eyes tracking,
- ✓ Identification of the eyes states,
- ✓ Calculation of PERCLOS and identification of driver state.

3. The Proposed System:

In this section, we discuss our presented system which detects driver drowsiness. The overall flowchart of our system is shown in Figure 1.

3.1 Face Detection: The symmetry is one of the most important facial features.

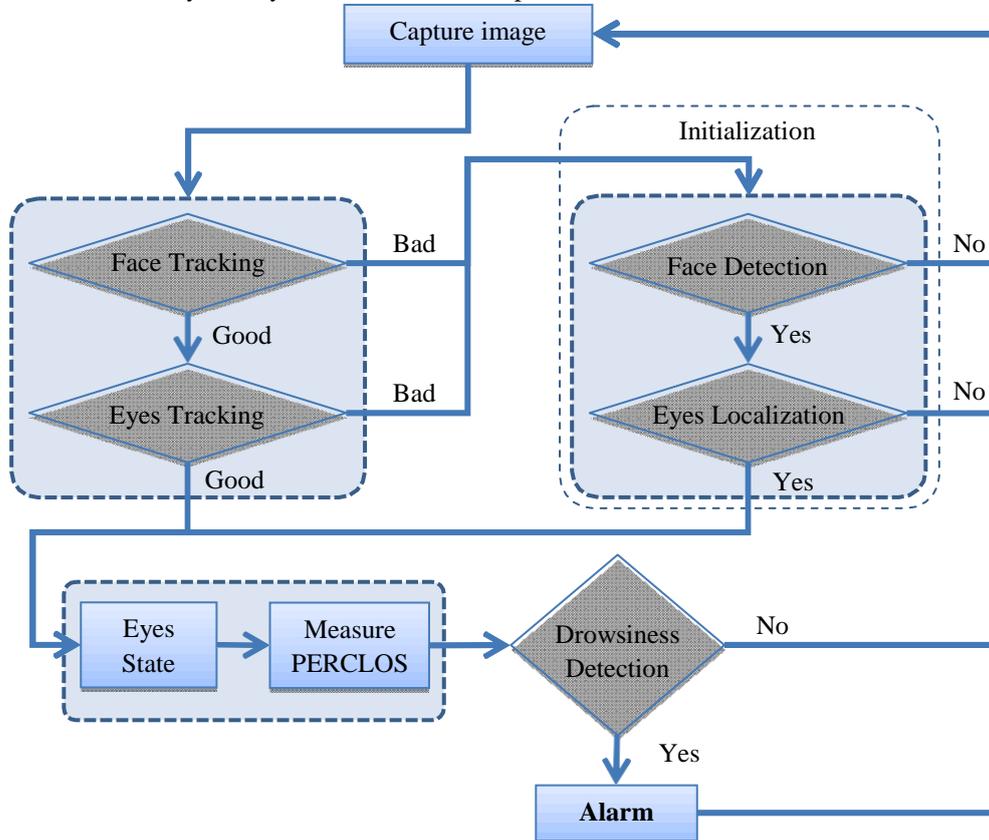


Figure 1: Flowchart of the proposed system.

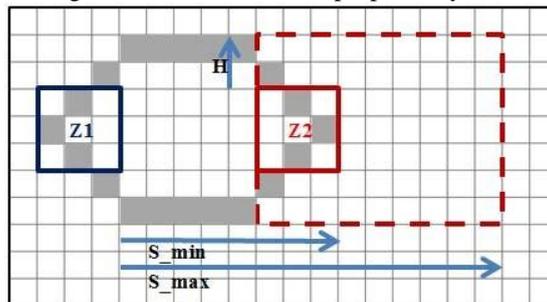


Figure 2: The new method to improve the calculation of the symmetry in the image.

We modeled the symmetry in a digital image by a one-dimensional signal (accumulator vector) with a size equal the width of the image, which gives us the value corresponding to the position of the vertical axis of symmetry of objects in the image. The traditional principle to calculate the signal of symmetry is for each two white pixels which are on the same line we increment the value in the medium between these two pixels in the accumulator vector. (The algorithm is applied on an edge image, we called a white pixel: the pixel with value1).

We introduce improvements on the calculation algorithm of symmetry into an image to adapt it to the detection of face, by applying a set of rules to provide a better calculation of symmetry of the face. Instead of computing the symmetry between two white pixels in the image, it is calculated between two windows (Z1 and Z2) (Figure2).

For each window Z1, we sweep the window Z2 in the area determined by the parameters S_{min} , S_{max} , and H . We increment the signal of symmetry between these two windows if the sum of white pixels is located between two thresholds $S1$ (maximum) and $S2$ (mini- mum).

Then we extract the vertical region of the image contours (Region of Interest ROI) corresponding to the maximum index of the obtained signal of symmetry. Next, we take a rectangle with an estimated size of face (Because the camera is fixed and the driver moves in a limited zone so we can estimate the size of the face using the camera focal length after the step of camera calibration) and we scan the ROI by searching the region that contains the maximum energy corresponding to the face (Figure 3).

We propose a checking on two axes: the position variance of the face detected according to time; i.e., in several successive images, it is necessary that the variance of the positions of the detected face is limited; because the speed of movement of the face is limited of some pixels from a frame to another frame which follows.

3.2 Eyes Localization: Since the eyes are always in a defined area in the face (facial anthropometric properties), we limit our research in the area between the forehead and the mouth (Eye Region of Interest 'eROI') (Figure 4.a). We benefit from the symmetrical characteristic of the eyes to detect them in the face. First, we sweep vertically the eROI by a rectangular mask with an estimated height of height of the eye and a width equal to the width of the face, and we calculate the symmetry. The eye area corresponds to the position which has a high measurement of symmetry. Then, in this obtained region, we calculate the symmetry again in both left and right sides. The highest value corresponds to the center of the eye. The result is shown in Figure 4.b.

3.3 Tracking: The tracking is done by Template Matching using the SAD Algorithm (Sum of Absolute Differences).

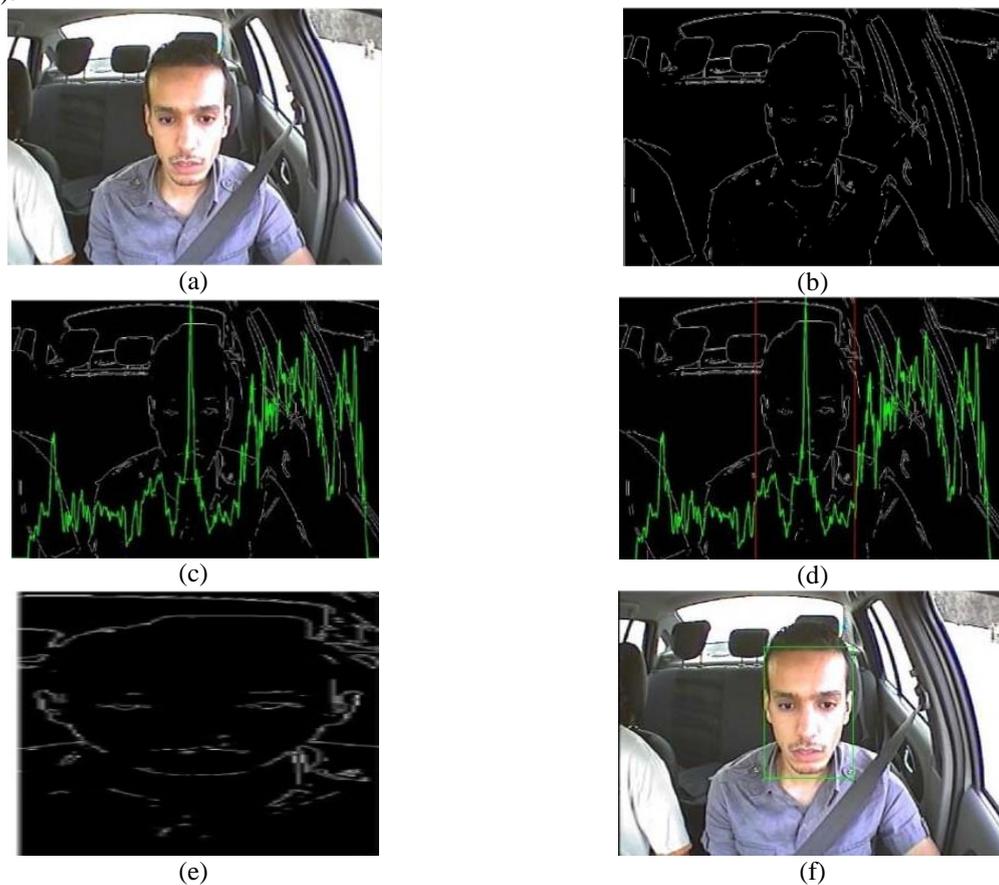


Figure 3: Face detection using symmetry. (a) Original image, (b) Edge detection, (c) Symmetry signal, (d) Localization of the maximum of symmetry, (e) Region of interest ROI (f)Result.



Figure 4: Eyes localization using symmetry. (a) eROI, Result.



Figure 5: Eyes states using HTC. (a) and (b) Edge detection, (c) and (d) Eyes states results N M detection, (c) and (d) Eyes states results.

$$SAD(x, y) = \sum_{j=1}^n \sum_{i=1}^n |I(x+i, y+j) - M(i, j)| \quad (1)$$

3.4 How the Algorithm Works: We proposed to make a regular update of the reference model M to adjust it every time when light conditions change while driving, by making a tracking test: good if $SAD \leq Th$

$$\text{Tracking} \begin{cases} \text{bad} & \text{if } SAD \geq Th \\ \text{good} & \text{if } SAD \leq Th \end{cases} \quad (2)$$

3.5 Eyes States: The determination of the eye state is to classify the eye into two states: open or closed. We use the Hough transform for circles [10] (HTC) on the image of the eye to detect the iris. For that, we apply the HTC to the edge image of the eye to detect the circles with defined rays, and we take at the end the circle which has the highest value in the accumulator of Hough for all the rays. Then, we apply the logical 'AND' logic between edges image and complete circle obtained by the HTC by measuring the intersection level between them "S". Finally, the eye state "Stateeye" is defined by testing the value "S" by a threshold:

$$\text{State} = \begin{cases} \text{Open} & \text{if } S \leq Th \\ \text{Closed} & \text{if } S > Th \end{cases}$$

The results are shown in Figure.5

Our system starts with the initialization phase, which is face and eyes detection to extract both face and eyes regions and take them as templates to track them in the following frames. For each tracking we test if that tracking is good or bad? If the tracking is bad we return to the initialization step, else we pass to the following steps which are: eyes states identification and driver state.

4. Experimental Results:

To validate our system (Figure 6), we test on several drivers in the car with real driving conditions. We use an IR camera with infrared lighting system operates automatically under the conditions of reduced luminosity and night even in total darkness. The results of the eye states are illustrated in Table 1, where the percentage error is the number of frames that have a false state of eye divided by the total number of frames multiplied by 100.



Figure 6: Our system installed in the car based on IR camera

We determine the driver state by measuring PER-CLOS. If the driver closed his eyes in at least 5 successive frames several times over a period of up to 5 seconds, it is considered drowsy.

5. Conclusion and Perspectives:

In this paper, we presented the conception and implementation of a system for detecting driver drowsiness based on vision that aims to warn the driver if he is in drowsy state. Then the vehicle should be stop by itself and the indication for the other vehicles are made. The message can be send to the corresponding authority of vehicle with the latitude and longitude value. This system is able to determine the driver state under real day and night conditions using IR camera. Face and eyes detection are implemented based on symmetry. Hough Transform for Circles is used for the decision of the eyes states. The results are satisfactory with an opportunity for improvement in face detection using other techniques concerning the calculation of symmetry. Moreover, we will implement our algorithm on a DSP (Digital Signal Processor) to create an autonomous system working in real time.

6. References:

1. Malla, P. Davidson, P. Bones, R. Green and R. Jones, "Automated Video-based Measurement of Eye Closure for Detecting Behavioral Microsleep", in 32nd Annual International Conference of the IEEE, Buenos Aires, Argentina, 2010.
2. P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001.
3. R. Lienhart and J. Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection", in Proceedings of the IEEE International Conference on Image Processing, 2002.
4. Open CV. Open Source Computer Vision Library Reference Manual, 2001.
5. S. Vitabile, A. Paola and F. Sorbello, "Bright Pupil Detection in an Embedded, Real-time Drowsiness Monitoring System", in 24th IEEE International Conference on Advanced Information Networking and Applications, 2010.
6. Bhowmick and C. Kumar, "Detection and Classification of Eye State in IR Camera for Driver Drowsiness Identification", in Proceeding of the IEEE International Conference on Signal and Image Processing Applications, 2009.
7. N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE Transactions on Systems, Man and Cybernetics, pp. 62-66, 1979.
8. T. Hong, H. Qin and Q. Sun, "An Improved Real Time Eye State Identification System in Driver Drowsiness Detection", in proceeding of the IEEE International Conference on Control and Automation, Guangzhou, CHINA, 2007.
9. Z. Tian et H. Qin, "Real-time Driver's Eye State Detection", in Proceedings of the IEEE International Conference on Vehicular Electronics and Safety, October 2005.
10. M. S. Nixon and A. S. Aguado, Feature Extraction and Image Processing, 2nd ed., Jordan Hill, Oxford OX2 8DP, UK, 2008.