LEVERAGING MACHINE LEARNING TO BUILD A DRIVER DROWSINESS DETECTION SYSTEM

[1]Roshan Roby, [2]Siva Shanker P S, [3]Vishal V S,

[1,2,3]Students, III CSE, St. Xavier's Catholic College of Engineering
Corespondence:ajithabose@sxcce.edu.in (Mrs. P.Ajitha, B.Tech.,M.E.,Department of Computer Science and Engineering,St. Xavier's Catholic College of Engineering).

ABSTRACT

Drowsiness and Fatigue of drivers are amongst the significant causes of road accidents. Every year, they increase the amounts of deaths and fatalities injuries globally.In this paper, a module for Advanced Driver Assis- tance System (ADAS) is presented to reduce the num- ber of accidents due to drivers fatigue and hence in- crease the transportation safety; this system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence.

We propose an algorithm to locate, track, and analyze both the drivers face and eyes to measure PERCLOS, a scientifically supported measure of drowsiness asso- ciated with slow eye closure.

Keywords—Drowsiness detection, ADAS, Face Detec- tion 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, al- tered physical condition or during long journeys. The sensation of sleep reduces the driver's level of vigil- ance producing dangerous situations and increases the probability of an occurrence of accidents. Driver drow- siness 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 technolo- gies to design and build systems that are able to moni- tor drivers and to measure their level of attention dur- ing the entire process of driving.

In this paper, a module for ADAS (Advanced driv- er 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 auto- matic detection of driver drowsiness based on visual information and artificial intelligence.

We propose an algorithm to locate, track and ana- lyze 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 drow- siness 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 pu- pils, an algorithm for detecting and tracking the driv- er'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 centre. Morphological operation and K- means is used 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 opti- mized Jones and Viola method [2]. The eye area is obtained by an horizontal projection. Finally, a new complexity function with a dynamic threshold to iden- tify the eye state.

Tian et Qin [9] build a system that checks the driv- er eye states. Their system uses the Cb and Cr compo- nents 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:

1) Face detection,
2) Eyes Location,
3) Face and eyes tracking,
4) Identification of the eyes states,
5) Calculation of PERCLOS and identification of driver state.

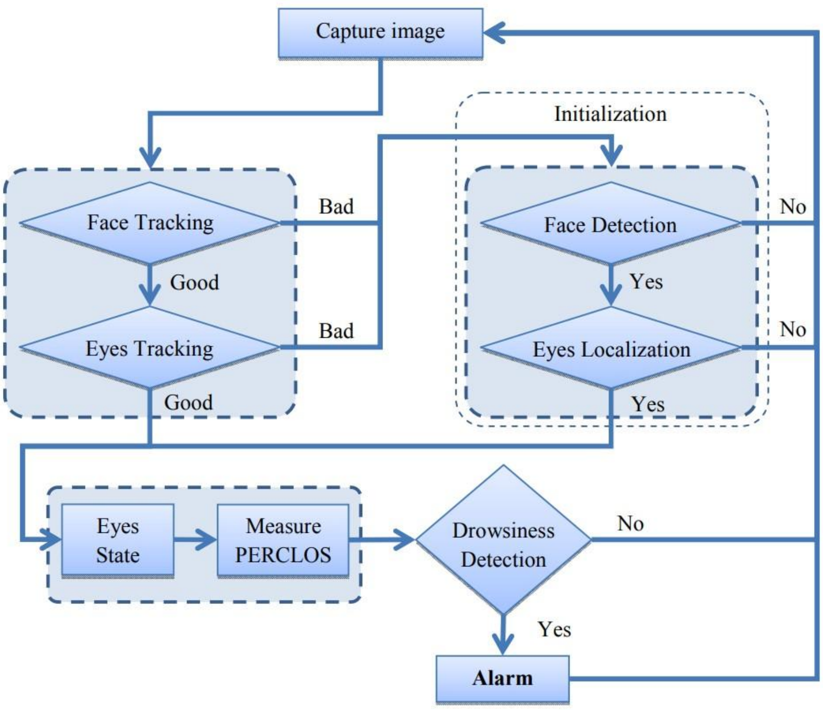


Figure 1 : Flowchart of the proposed system.

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.

We modelled 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 incre- ment 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 value 1).

We introduce improvements on the calculation al- gorithm 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) (Figure 2).

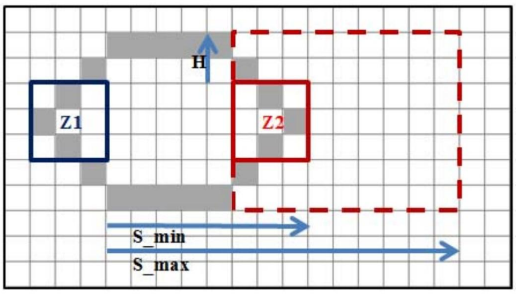


Figure 2 - The new method to improve the calculation of the symmetry in the image.

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 correspond- ing 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 va- riance 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 rectangu- lar 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.

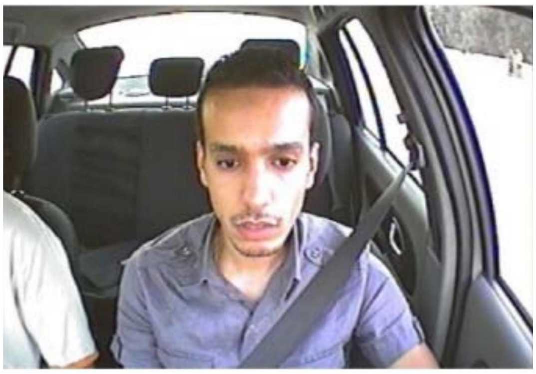
The eye area corresponds to the position which has a high measurement of symmetry. Then, in this ob- tained region, we calculate the symmetry again in both left and right sides. The highest value corresponds to the center of the eye. The result is shown in Figure 4.b

. (A) (B)

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

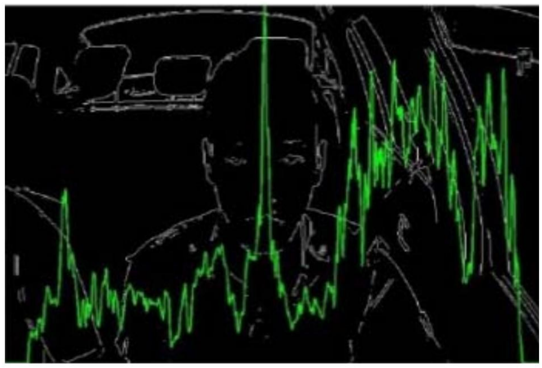
3.3. Tracking

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

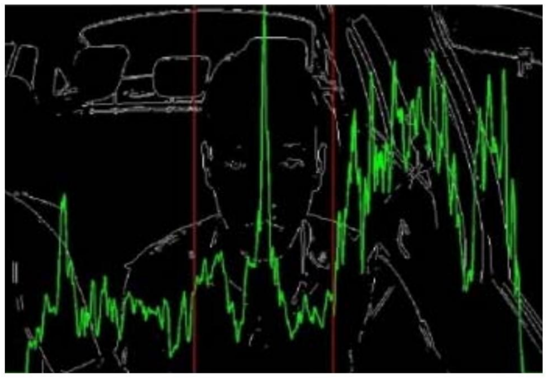


(a)

(b)

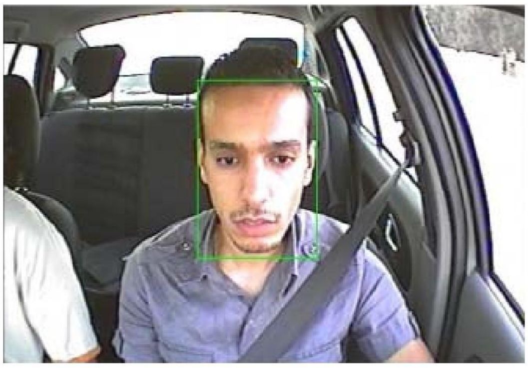


(c)



(d)

(e)



(f)

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.

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

We proposed to make a regular update of the refer- ence model M to adjust it every time when light condi- tions changes while driving, by making a tracking test:

$$Tracking \begin{cases} good & if\ SAD \le Th \\ bad & if\ SAD > Th \end{cases} \quad (2)$$

3.4. 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 “” is defined by test- ing the value “S” by a threshold:

$$\mathrm{State}_{\mathrm{eye}} = \begin{cases} \mathrm{Open} & \mathrm{if}\ \ S \ge \mathrm{Th} \\ \mathrm{Closed} & \mathrm{if}\ \ S < Th \end{cases} \quad (3)$$

The results are shown in Figure 5.

3.5. Driver State

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

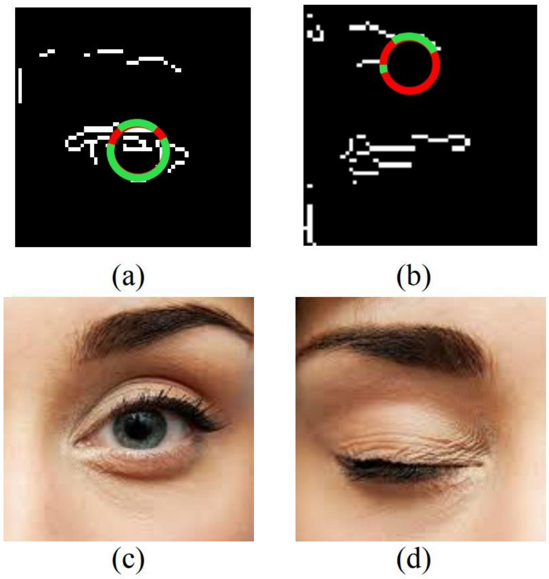


Figure 5 – Eyes states using HTC. (a) and (b) Edge detection , (c) and (d) Eyes states results.

3.6. How the Algorithm Works

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 sever- al drivers in the car with real driving conditions. We use an IR camera with infrared lighting system oper- ates automatically under the conditions of reduced luminosity and night even in total darkness.

The results of the eye states are illustrated in Ta- ble1, 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.

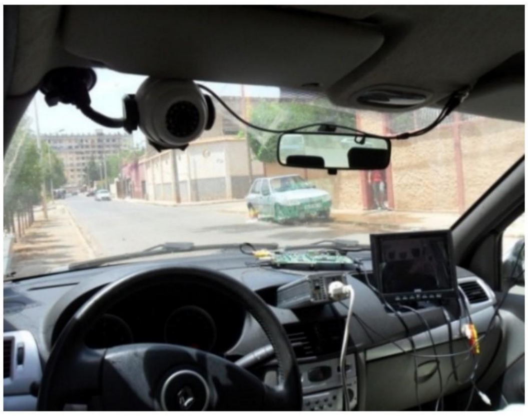


Figure 6 - Our system installed in the car based on IR camera

TABLE 1 – RESULTS OBTAINED FROM THE SYSTEM.

Driver	frames Number	False open	Eyes sates Closed	false rate
D1/day	420	17	0	4%
D2/day	430	15	0	3.5%
D3/day	245	7	1	3.2%
D1/night	200	3	1	2%
D1/night	200	1	0	0.5%
D1/night	200	6	3	4.5%

According to the obtained results, our system can determine the eye states with a high rate of correct decision.

5.CONCLUSION AND PERSPECTIVES

In this paper, we presented the conception and im- plementation of a system for detecting driver drowsi- ness based on vision that aims to warn the driver if he is in drowsy state.

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 autonom- ous system working in real time.

REFERENCES

[1] A. Malla, P. Davidson, P. Bones, R. Green and R. Jones, "Automated Video-based Measurement of Eye Closure for Detecting Behavioural 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]OpenCV. Open Source Computer Vision Library Reference Manual, 2001.

[5]S.Vitabile,A.PaolaandF.Sorbello,"BrightPupil Detection in an Embedded, Real-time Drowsiness Monitoring System", in 24th IEEE International Conference on Advanced Information Networking and Applications, 2010.

[6] B. 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 Cybernatics, 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.