J Intell Robot Syst
DOI 10.1007/s10846-009-9391-1

Real-Time Warning System for Driver Drowsiness Detection Using Visual Information

Marco Javier Flores · José María Armingol · Arturo de la Escalera

Received: 3 November 2008 / Accepted: 18 November 2009

Abstract Traffic accidents due to human errors cause many deaths and injuries around the world. To help in reducing this fatality, in this research, a new module for Advanced Driver Assistance System (ADAS) for automatic driver drowsiness detection based on visual information and Artificial Intelligence is presented. The aim of this system is to locate, to track and to analyze the face and the eyes to compute a drowsiness index, working under varying light conditions and in real time. Examples of different images of drivers taken in a real vehicle are shown to validate the algorithm.

Keywords Driver's drowsiness · Neural networks · Support vector machine · Gabor filter · Artificial intelligence · ADAS · Computer vision

1 Introduction

ADAS is part of the active safety systems that interact much more with drivers to help them avoid traffic accidents, indeed, its goal is to contribute in the reduction of traffic accidents, by using new technologies; that is, incorporating new systems for increasing vehicle security, and at the same time, decreasing the danger situations

M. J. Flores · J. M. Armingol (✉) · A. de la Escalera
Intelligent Systems Laboratory, Universidad Carlos III de Madrid,
C/. Butarque 15, 28991, Leganés, Madrid, Spain
e-mail: armingol@ing.uc3m.es
URL: www.uc3m.es/islab

M. J. Flores
e-mail: mjflores@ing.uc3m.es
URL: www.uc3m.es/islab

A. de la Escalera
e-mail: escalera@ing.uc3m.es
URL: www.uc3m.es/islab

Published online: 05 December 2009

that may arise during driving, due to human errors. In this scenario, vehicular security research is focused on driver analysis, in this particular case; drowsiness and distraction are studied more intensely [4].

Drowsiness appears in situations of stress and fatigue in an unexpected and inopportune way, and it may be produced by sleep disorders, certain type of medications, and even, boredom situations, for example, driving for a long time. In this sense, sleepiness sensation diminishes the level of vigilance, and it produces danger situations and increases the probability that an accident occurs.

It has been estimated that drowsiness causes between 10% and 20% of traffic accidents with dead [31] and injured drivers [11], whereas the trucking industry shows 57% of fatal truck accidents for this fatality [2, 22]. Fletcher et al. in [12] goes further and has mentioned that 30% of all traffic accidents have been caused by drowsiness and Brandt et al. [4] presents statistics in which 20% of all accidents are caused by fatigue and inattention. In USA drowsiness is responsible for 100,000 traffic accidents whose costs are about $12,000 million [28]. In Germany, one of four traffic accidents have their origin in drowsiness, in England 20% off all traffic accidents are produced by drowsiness [16] and in Australia 1,500 million dollars has been spent on this fatality [24].

In this context, it is important to use new technologies to design and to build systems that will monitor drivers, and measure their level of attention throughout the whole driving process. Fortunately, people in a state of drowsiness produce several visual cues that can be detected on the human face, they are:

- Yawn frequency,
- Eye-blinking frequency,
- Eye-gaze movement,
- Head movement and,
- Facial expressions.

Taking advantage of these visual characteristics, computer vision is the feasible and appropriate technology to treat this problem. This article presents the drowsiness detection system of the IVVI (Intelligent Vehicle based Visual on Information) vehicle [1]. The goal of this system is to estimate driver drowsiness automatically and to prevent drivers falling asleep while driving.

The organization of the paper is as follows. Section 2 presents an extended state of the art divided by light conditions. Section 3 introduces the proposed method that consists of face and eye detection, face and eye tracking and the drowsiness index based on support vector machine. Finally, in section 4 results and conclusions are shown.

2 Related Work

To increase the traffic security and to reduce the number of traffic accidents, numerous universities, research centers, automotive companies (Toyota, Daimler Chrysler, Mitsubishi, etc.) and governments (Europe Union, etc.) are contributing in the development of ADAS for driver analysis [2], using different technologies. In this sense, the use of visual information to know the driver's drowsiness state and understand his/her behavior is an active research field.

This problem requires the recognition of human behavior when in a state of sleepiness through the analysis of the eyes and the face (head). This is a difficult task, even for humans because there are many factors involved, for instance, changing illumination conditions and a variety of possible face poses. Taking into account the illumination, the state of the art has been divided in two parts; one is the systems that work with natural daylight; another is the systems which work with the help of illumination systems based on near infrared (NIR) illumination.

2.1 Systems with Daylight Illumination

To analyze driver drowsiness several systems have been built in recent years. They usually require simplifying the problem to work partially or under special environments, for example, D'Orazio et al. [10] has proposed an eye detection algorithm that searches for the eyes in the whole image assuming that the iris is always darker than the sclera and based on the Hough transform for circles and geometrical constraints the eyes candidates are located, next, they are passed to a neural network that classify between eyes and non-eyes. This system is able to classify the eyes as being in an open or closed state. The main limitations of this algorithm are: it is applicable only when the eyes are visible in the image, and it is not robust at changing illumination. Horng et al. [17] has shown a system that uses a skin color model over HSI space for face detection, edge information for eye localization and dynamical template matching for eye tracking. Using eyeball color information, it identifies the eye state and computes the driver's state, i.e., asleep or alert; if the eyes are closed over a five consecutive frames, the driver is dozing. Brandt et al. [4] has shown a system that monitors driver fatigue and inattention. For this task, he uses the Viola & Jones (VJ) method [34] to detect the driver's face. Using the optical flow algorithm over eyes and head this system is able to compute the driver state. Tian and Qin in [31] have built a system for verifying the driver's eye state. Their system uses Cb and Cr components of the YCbCr color space; with vertical projection function this system localizes the face region and with horizontal projection function it localizes the eye region. Once the eyes are localized the system computes eye state using a complexity function. Dong and Wu [11] have presented a system for driver fatigue detection, which uses a skin color model based on bivariate Normal distribution and Cb and Cr components of the YCbCr color space. After localizing the eyes, it computes the fatigue index utilizing the eyelid distance to classify open eyes and closed eyes; if the eyes are closed over five consecutives frames, the driver is regarded as dozing, alike to the Horng's work. Branzan et al. [5] also presents a system for drowsiness monitoring using template matching to analysis the eye state.

2.2 Systems with Infrared Illumination

In this case, due to night-time light conditions, Ji et al. [21] and Ji and Yang [22] has presented a drowsiness detection system based on NIR illumination and stereo vision. This system localizes the eye position by using image differences based on the bright pupil effect. Afterwards, this system computes the blind eyelid frequency and eye gaze to build two drowsiness indices: PERCLOS (percentage of eye closure over time) [28] and AECS (average eye closure speed). Bergasa et al. [2] also has developed a non-intrusive system using infrared light illumination, this system computes driver vigilance level using a finite state automata (FSM) [3] with six eye states that computes several indices, among them, PERCLOS; also, this system is able to detect inattention through face pose analysis. Another work using this type of illumination is presented by Grace [14] for measuring slow eyelid closure. Systems using NIR illumination work well under stable lighting conditions [2, 9]; however, this is a shortcoming for applications in real vehicles, where the light is changing all the time. In this scenario, if the spectral pupils disappear, then it will be difficult to detect the eyes.

3 System Design to Drowsiness Detection

This paper presents a system to detect the driver's drowsiness that works on grayscale images. The scheme of the system is shown in Fig. 1 in which six modules are presented:

- Face detection
- Eye detection
- Face tracking
- Eye tracking
- Drowsiness detection and
- Distraction detection

Each one of these parts will be explained in the following subsections.

3.1 Face Detection

To localize the face, this system uses the VJ object detector which is a machine learning approach for visual object detection. It uses three important aspects to make an efficient object detector based on the integral image, AdaBoost technique and cascade classifier [34]. Each one of these elements is important for processing the

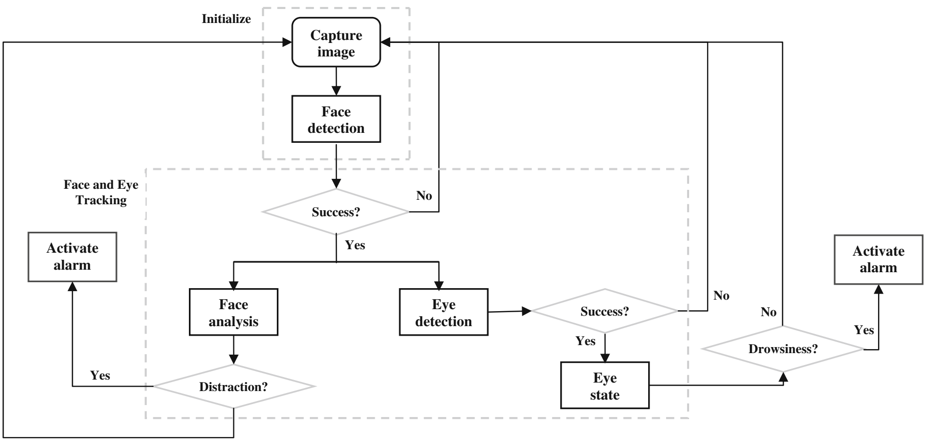


Fig. 1 Algorithm scheme

images efficiently and in near real-time with 90% of correct detection. A further important aspect of this method is its robustness under changing light conditions. However, in spite of the above-mentioned features, its principal disadvantage is that it cannot extrapolate and does not work appropriately when the face is not in front of the camera axis. Such would be the case when the driver moves his/her head; however, this shortcoming will be analyzed later on.

Continuing with the algorithm description, when driver's face is detected, it is enclosed within a rectangle RI (region of interest) which is addressed by left-top corner coordinates $P_0 = (x_0, y_0)$ and right-bottom corner coordinates $P_1 = (x_1, y_1)$, as can be observed in Fig. 2a–c. Indeed, the rectangle size comes from experimental analysis developed on the face database that has been created for this task.

3.2 Eye Detection

Localizing the eye position is a difficult task because different features define the same eye depending, for example, the area of the image where it appears or its iris color, but the main problem during driving is the changing ambient light conditions.

Once the face has been located through the rectangle RI in the previous section, using the face anthropometric properties [13] which come from face database analysis, two rectangles containing the eyes are obtained. Preliminary, this system uses $\mathrm{RI_L}$ for the left eye rectangle and $\mathrm{RI_R}$ for the right eye rectangle as can be seen in the following four equations and they are shown in the Fig. 3.

$$(u_{0L}, v_{0L}) = (x_0 + w/6, y_0 + h/4) \tag{1}$$

$$(u_{1L}, v_{1L}) = (x_0 + w/2, y_0 + h/2) \tag{2}$$

$$(u_{0R}, v_{0R}) = (x_0 + w/2, y_0 + h/4) \tag{3}$$

$$(u_{1R}, v_{1R}) = (x_1 - w/6, y_1 - h/2) \tag{4}$$

where $w = x_1 - x_0$ and $h = y_1 - y_0$.

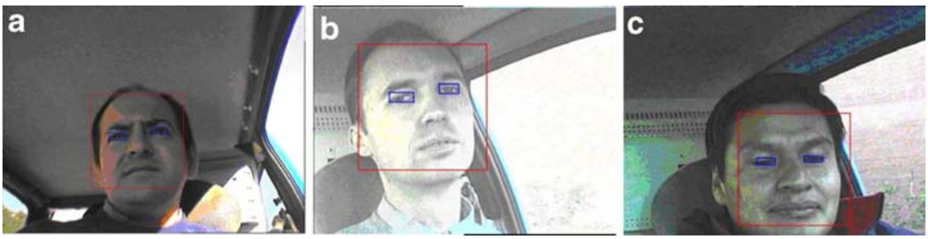


Fig. 2 Viola & Jones method

Fig. 3 Eye rectangles RI_R and RI_L

After the previous step; the exact position of each eye is searched for by incorporating information from grey-level pixels. The main idea is to obtain a random sample from the pixels that belong to the eye area, and then, to adjust a parametric model. Figure 4 shows this procedure in which a random sample is extracted in (a) and an elliptic model is adjusted in (b). In this case, the eye state is independent, i.e., it can be open or closed.

To extract the random sample the following algorithm is proposed. Let $I(x, y) \in [0, 255]$ be the pixel value in the position (x, y), then:

- Generate the image J by means of the following equation:

$$J(x, y) = \frac{I(x, y) - m}{\sigma} \tag{5}$$

where m and σ are the mean and the standard deviation, respectively. These parameters are computed over the eye rectangles located previously.

- Generate the image K using the Eq. 6:

$$K(x, y) = \begin{cases} J(x, y) - 256 * \delta_1 & if \quad J(x, y) \geq 0 \\ 256 * \delta_2 + J(x, y) & if \quad J(x, y) < 0 \end{cases} \tag{6}$$

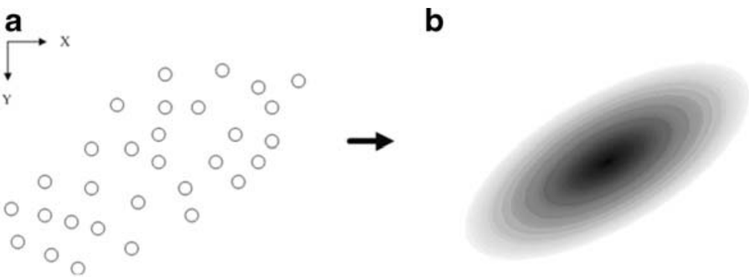


Fig. 4 **a** Random sample, **b** eye parametric model

where $\delta_1 = \max(0, ceil(J(x, y)/256) - 1)$, $\delta_2 = \max(1, ceil(|J(x, y)|/256))$ and $ceil(x)$ is the function that returns the smallest integer larger than x.

- Obtain the binary image, B, from image K through the Eq. 7, namely,

$$B(x, y) = \begin{cases} 255 & if \quad K(x, y) \geq \kappa \\ 0 & other \quad case \end{cases} \tag{7}$$

where κ is computed by Ostu's method [29] which is used to compute an automatic threshold, Fig. 5b.

- Compute the gradient image, G, using the Sobel horizontal (S_x) and vertical (S_y) edge operator followed by an image contrast enhancement [20], Fig. 5c.

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \qquad S_y = -S_x^T \tag{8}$$

- Compute the logarithm image [35], L, with the objective to enhance the iris pixels that are the central part of the eye, Fig. 5d.

$$L(x, y) = \log(1 + I(x, y)) \tag{9}$$

Starting from the pixels that have been extracted from the images B, G and L; it is possible to obtain the random sample previously mentioned. This sample presents an ellipse shape and an elliptic model has been adjusted over this by using the expectation maximization algorithm (EM) [26]. The ellipse center has been given special attention, because, it allows the exact position of the eye center to be obtained. The ellipse axes determine the width and height of the eyes. The result is shown in Fig. 6b.

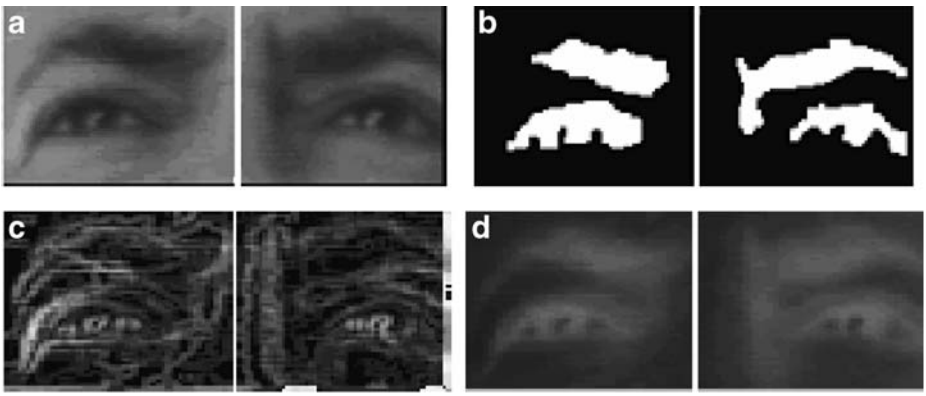


Fig. 5 Eye location through R_L and R_R, **a** grayscale image, **b** binary image (B), **c** gradient image (G), and **d** logarithm image (L)

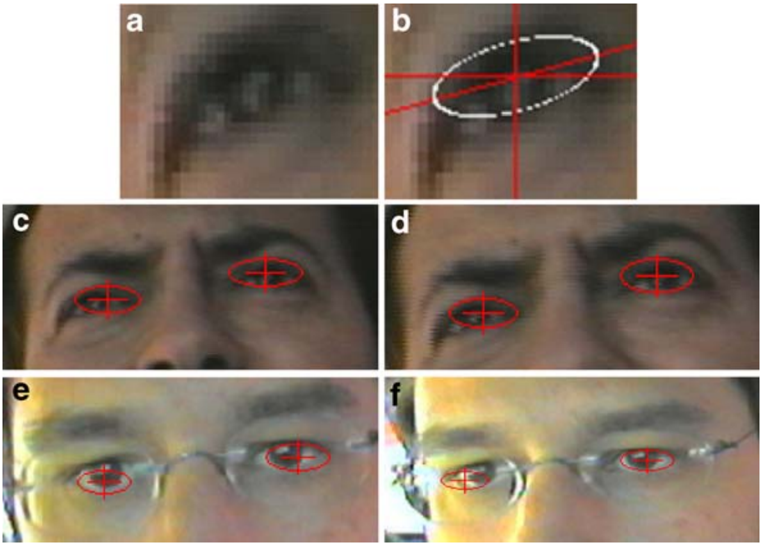


Fig. 6 Expectation maximization algorithm over the spatial distribution of the eye pixels, **a** eye image, **b** ellipse parameters: center, axes and inclination angle. **c**–**f** Other examples of this procedure

The main reason to use the pixel information through a random sample is because head movement, illumination changes, etc. do not allow complete eye pixel information to be obtained, i.e., only partial information of the eye in the images B, G and L is available; where an elliptic shape prevails. This random information makes it feasible to use an algorithm that computes the parameters of a function to approximate eye ellipse shape. EM computes the mean, variance and the correlation of X and Y coordinates that belong to the eye. The initial parameters to run EM are obtained from a regression model adjusted with the least square method. The number of iterations to run EM is fixed in 10, and the sample size is taken at least one third of the rectangle area RI_R. These parameters will be used in the eye state analysis below.

3.3 Tracking

There are a number of reasons for tracking. One is problems that were found with the VJ during this research. Another is the necessity to track the face and the eyes continuously from frame to frame. A third reason is to satisfy the real-time conditions that reduce the search space. The tracking process has been developed using the Condensation algorithm (CA) in conjunction with the neural networks (NN) for face tracking and with template matching for eye tracking.

3.3.1 The Condensation Algorithm

This contribution implements the Condensation algorithm that was proposed by Isard and Blake [18, 19] for tracking active contours using a stochastic approach.

CA combines factored sampling (Monte-Carlo sampling method) with a dynamical model that is governed through the state Eq. 10.

$$X_t = f\left(X_{t-1}, \xi_t\right) \tag{10}$$

where X_t is the state at instant t, $f(\cdot)$ is an nonlinear equation and depends on a previous state plus a white noise. The goal is to estimate the state vector X_t with the help of system observation which are the realization of the stochastic process Z_t governed by the measurement equation:

$$Z_t = h\left(X_t, \eta_t\right) \tag{11}$$

where Z_t is the measure system at time t, $h(\cdot)$ is another nonlinear equation that links the present state plus a white noise. The processes ξ_t and η_t are each one white noise and are independent of each other. Also, these processes in general are non-Gaussian and multi-modal. It must be pointed out that X_t is an unobservable underlying stochastic process.

3.3.2 Neural Networks

McCulloch and Pitts proposed the first model of an artificial neuron in 1943 which was based on its corresponding biological neuron [25]. Since then, neural networks have evolved and they have been used in a wide variety of problems of pattern recognition and classification, coming from engineering and social science [27, 33]. Figure 7 shows several face examples used for training a backpropagation neural network.

Before training the neural network, a preprocessing step that consists of two parts is necessary:

- Contrast modification using gamma correction given by Eq. 12 with $\gamma = 0.8$ which has been determined experimentally [30].

$$J\left(x, y\right) = I\left(x, y\right)^{\gamma} \tag{12}$$

- Remove the contour points through the operation AND with a mask of Fig. 8a.

After that, the characteristic vector that consists of the gray-level values of the pixels coming from the face image is extracted. The rate of classification subsequent to the training is more than 93%.

Fig. 7 Examples of a face database which contain faces in different orientations: **a** left profile, **b** front view, **c** right profile, **d** down profile, and **e** up profile

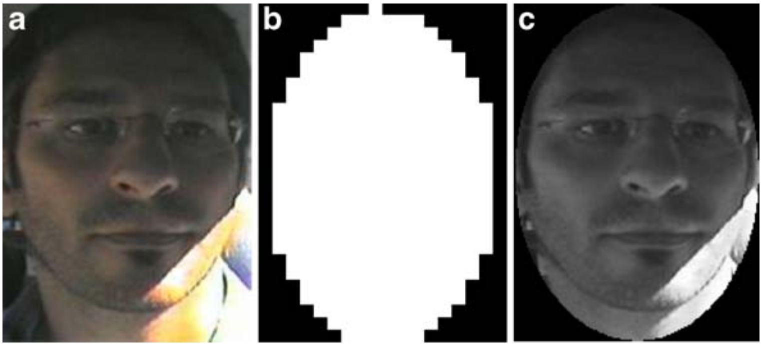


Fig. 8 Mask for face training and its result

3.3.3 Face Tracking

Previously, it has been mentioned that the VJ method has problems detecting faces when they deviate from nominal position and orientation; so, to correct this disadvantage the tracking face has been developed. To show this shortcoming, Fig. 9 shows several instants of time where the VJ method does not find the driver's face, in this sense, Fig. 10 presents an extended example, where the true position and the VJ position are represented over a frame sequence. The true position has been obtained manually retrieved.

The chief problem of the VJ method is that it is only able to localize the human face when it is in frontal position of the camera. This drawback leads to an unreliable system of driver analysis throughout the driving process that is highly dynamic, for example, when looking at the mirror. Much effort has gone into correcting this problem; so, an efficient tracker has been implemented using CA in conjunction with a backpropagation neural network.

Through recursive probabilistic filtering of the incoming image stream, the state vector

$$X_t = (x_c, y_c, u_c, v_c, w, h)^T \in R^6 \tag{13}$$

Fig. 9 The driver's face is not found by the Viola & Jones method at several time instants

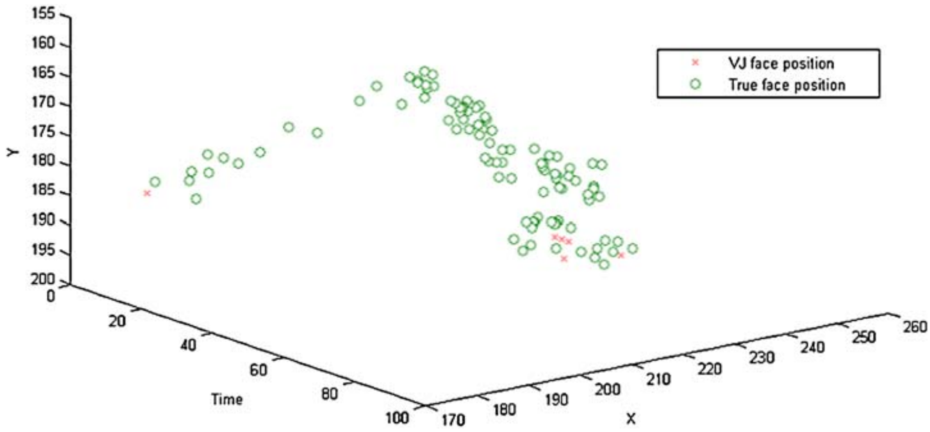


Fig. 10 Example where the VJ method does not find the driver's face in a 100-frame sequence

of a driver's face is estimated for each time step t. It is characterized by its position, velocity and size. Let (x_c, y_c) represent its position the center, (u_c, v_c) be its velocity in x and y direction and (w, h) be its size in pixels. In the same way, the measure vector is given by Eq. 14.

$$Z_t = (x_c, y_c, w, h)^T \in R^4 \tag{14}$$

The dynamics of the driver's face is modeled as a second order autoregressive process AR(2), according to Eq. 15.

$$X_t = A_2 X_{t-2} + A_1 X_{t-1} + \xi_t \tag{15}$$

where A is the transition matrix proposed by [19] and ξ_t represents the system perturbation at time t. The most difficult part in CA is to evaluate the observation density function, in this contribution to compute the weight $\pi_t^{(j)} = p\left(z_t \middle| x_t = s_t^{(j)}\right)$ for $j = 1, ..., N$, at time t, a neural network value in the range [0,1], which gives an approximation of the face and non-face in conjunction with the distance with

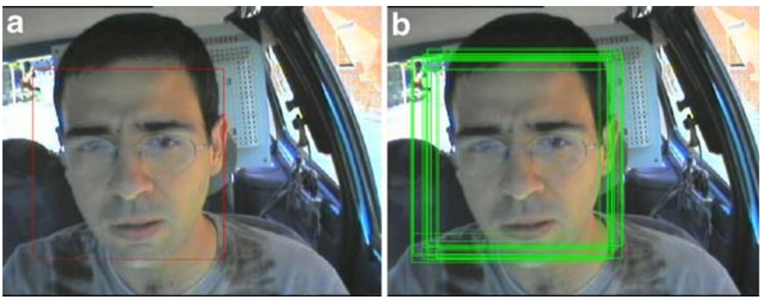


Fig. 11 One time step of the Condensation algorithm **a** predicted region, **b** particles regions

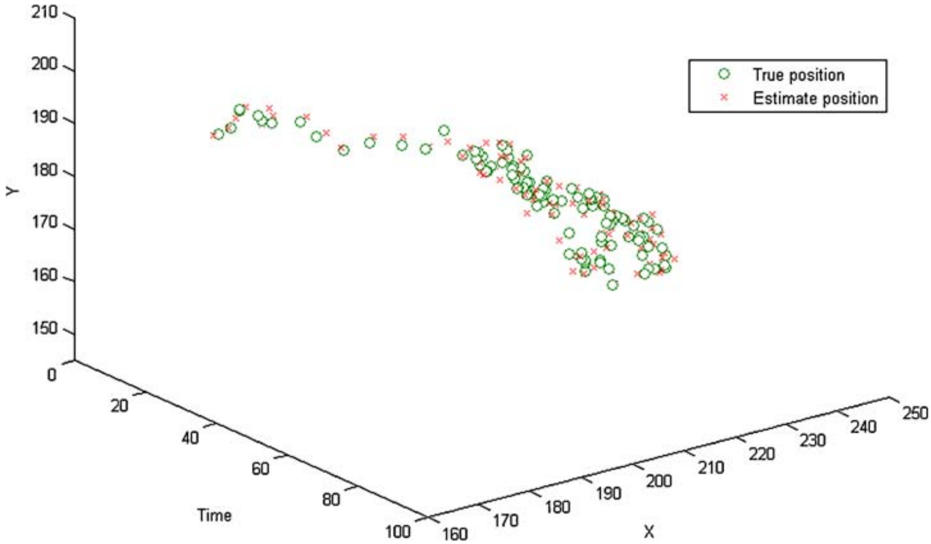


Fig. 12 Trajectory of the real and estimated face-center in a 100-frame sequence using the proposed tracker

respect to the face to track. This is similar to the work of Satake and Shakunaga [32] who have used the sparse template matching for computing the weight $\pi_t^{(j)}$ of the sample $s_t^{(j)}$ for $j = 1, ..., N$. In this contribution, the neural network value is used as an approximate value for the weights.

The density function of the initial state is $p(x_0) = N(z_0, \Sigma_0)$, where z_0 is computed by the VJ method and Σ_0 is given in [22]. Figure 11b depicts a particle representation and Fig. 12 shows the tracking process in which the green circle is the true position and a red cross characterizes a particle or a hypothesis, whereas the Fig. 13 shows the probability over time. This tracker is highly flexible because the neural network includes faces and non-faces with different head orientations and under various illumination conditions. Table 1 presents more results over several sequences of drivers faces. The sequences come from the drivers' database, which was taken

Fig. 13 Estimated value of the a posteriori density of the face-center in a 100-frame sequence using the proposed tracker, the face is detected in the fourth frame

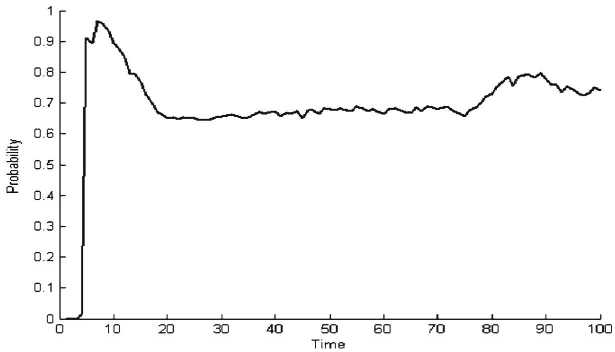


Table 1 Result of face tracking

Driver	Total frames	Tracking failure	Correct rate (%)
D1	960	60	93.75
D2	900	22	97.55
D3	500	45	91.00
D4	330	15	95.45
D5	1400	50	96.42

to develop these experiments. The true position of the faces has been obtained manually retrieved.

3.3.4 Eye Tracking

For this task, the state of the eye is characterized by its position and velocity over the image. Let (x, y) represent the eye pixel position at time t and (u, v) be its velocity at time t in x and y directions, respectively. The state vector at time t can, therefore, be represented by Eq. 16.

$$X_t = (x, y, u, v)^T \in R^4 \tag{16}$$

The transition model is given by Eq. 17 which is a first autoregressive model AR(1).

$$X_t = AX_{t-1} + \xi_t \tag{17}$$

The evaluation of the observation density function is developed by a template matching strategy [32] that was truncated to reduce the false detection. CA is initialized when the eyes are detected with the method from the previous section plus a white noise and it is similar to the case of face tracking. Figure 14 depicts the eye

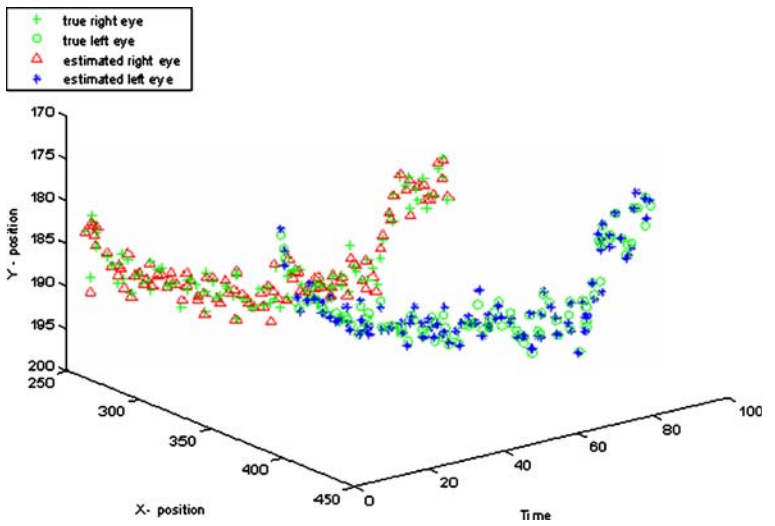


Fig. 14 Trajectory of the real and estimated eyes-center in a 100-frame sequence

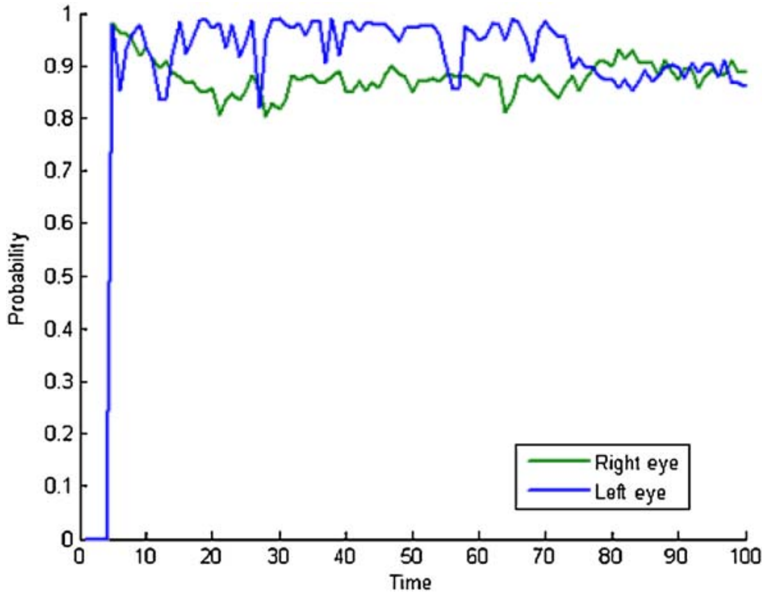


Fig. 15 Estimated value of the a posteriori density of the eye-center in a 100-frame for right and left eyes, the eyes are detected in the four frame

trajectory tracking and Fig. 15 shows the compute values of the a posteriori density function of each eye, both on a sequence of 100 images, whereas Table 2 shows the eye tracking results that has been developed in several sequences of images.

3.4 Eye State Detection

To identify drowsiness through eye analysis it is necessary to know its state: open or closed, through the time and develop an analysis over time, i.e., to measure the time that has been spent in each state. Classification of the open and closed state is complex due to the changing shape of the eye, among other factors, the changing position and the rotating of the face, and variations of twinkling and illumination. All this makes it difficult to analyze eye in a reliable manner. For the problems that have been exposed a supervised classification method has been used for this challenging task, in this case, a support vector machine (SVM). Figure 16 presents the schema proposed for eye state verification.

Table 2 Result of eye tracking

Driver	Total frames	Tracking failure	Correct rate (%)
D1	960	20	97.91
D2	900	30	96.60
D3	500	8	98.40
D4	330	14	95.75
D5	1400	90	93.57

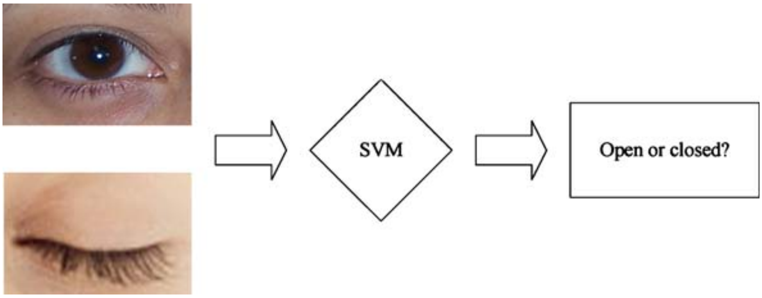


Fig. 16 SVM schema for eye state verification

3.4.1 Support Vector Machine

SVM classification [6, 8, 15] is rooted in statistical learning theory and pattern classifiers, it uses a training set, $S = \{(x_i, y_i) : \ i = 1, \cdots, m\}$, where x_i is the characteristic vector in R^n, $y_i \in \{1, 2\}$ represents the class, in this case 1 for open eyes and 2 for closed eyes, and m is the number of elements of S. From a training set a hyperplane is built that allows the classification between two classes and minimizes the empirical risk function [15].

Mathematically, SVM consists of finding the best solution to the following optimization problem:

$$\begin{aligned} & \min_{\alpha} f(\alpha) = \frac{1}{2}\alpha^T Q\alpha - e^T\alpha \\ & \text{s.t.} \\ & 0 \leq \alpha_i \leq C, \quad i = 1, \cdots, m \\ & y^T\alpha = 0 \end{aligned} \tag{18}$$

where e is a m by the 1 vector, C is an upper bound, Q is a m by m matrix with $Q_{ij} = y_i y_j K(x_i, x_j)$ and $K(x_i, x_j)$ is the kernel function. By solving the above quadratic programming problem, SVM tries to maximize the margin between data points in the two classes and minimize the training errors simultaneously; Fig. 17 depicts the mapping of the input space to a high dimensional feature space through a nonlinear transformation and its maximization process.

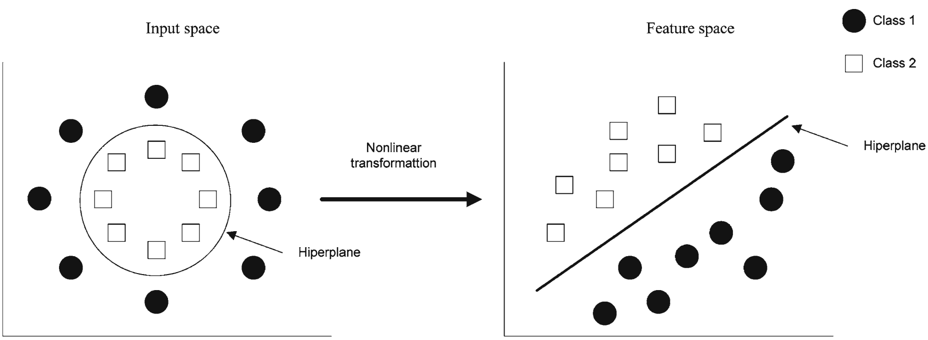


Fig. 17 SVM representation

3.4.2 Eye Characteristic Extraction Using Gabor Filter

The Gabor filter was used by Daugman for image analysis, changing the orientation and scale [9, 23]. Indeed, they are multi-scale and multi-orientation kernels. They can be defined by Eq. 19 that is a complex function.

$$g(x, y, \theta, \phi) = \exp\left(-\frac{x^2 + y^2}{\sigma^2}\right) \exp\left(i2\pi\theta\left(x \cos(\phi) + y \sin(\phi)\right)\right) \tag{19}$$

where θ and ϕ are the scale and orientation parameters, σ is the standard deviation of the Gaussian kernel that depends upon the spatial frequency to measured, i.e. θ. The response of the Gabor filter to an image is obtained by a 2D convolution operation. Let $I(x, y)$ denote the image and $G(x, y, \theta, \phi)$ denote the response of a Gabor filter with scale θ and orientation ϕ to an image at point (x, y) on the image plane. $G(\cdot)$ is obtained by (20).

$$G(x, y, \theta, \phi) = \iint I(p, q)\, g(x - p, y - q, \theta, \phi)\, dpdq \tag{20}$$

Some combinations of scales and orientations are more robust for the classification between open eye and closed eye. Indeed, three scales, four orientations have been used to generate Fig. 18, they are {1,2,3} and {0, π/4, π /2, 3π/4} that were obtained experimentally over an image of size 30 by 20.

Once the response of a Gabor filter is obtained, the eye characteristic vector is extracted by a sub-window procedure described by Chen and Kubo [7] and denoted

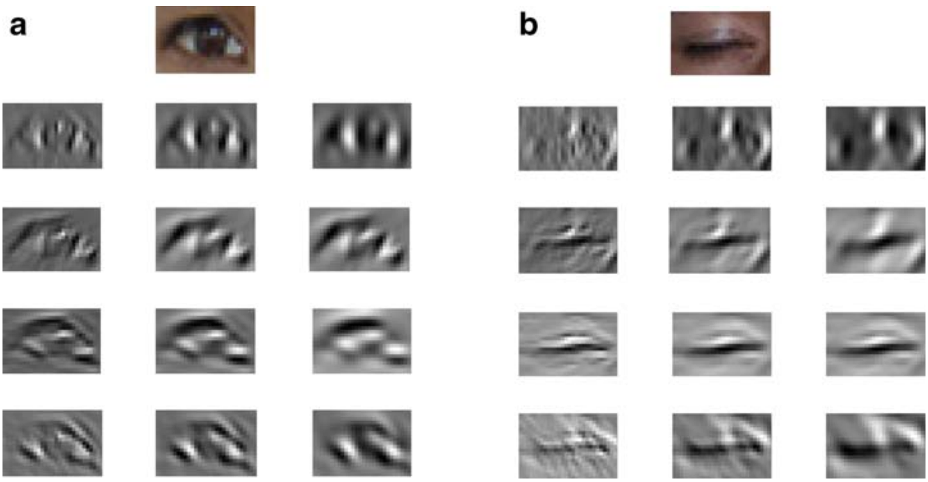


Fig. 18 Gabor filter for $\theta = \{0,1,2\}$ and $\phi = \{0, \pi/4, \pi/2, 3\pi\ /4\}$

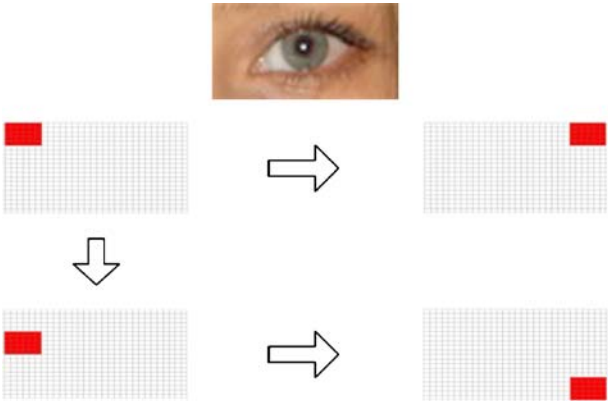


Fig. 19 Sub-window images from the Gabor filter

by $d \in R^{360}$. This vector is computed by Eq. 21 over each sub-window of size 5 by 6. Figure 19 shows the sub-window diagram.

$$d_i^{\theta,\phi} = \frac{1}{30} \sum_{y=1:5} \sum_{x=1:6} G(x, y, \theta, \phi) \qquad i = 1, \ldots, 20 \tag{21}$$

To do this work a training set has been built that consists of open eyes and closed eyes. The images come from diverse sources, under several illumination conditions and are of different races. A further important aspect of this eye database is that it contains images of different eye colors, i.e., blue, black, green see Fig. 20.

Previous to SVM training, it is indispensable to process each image that consists of histogram equalization, filter with the median filter, followed by the sharpen filter. The median filter is used to reduce the image noise, whereas the sharpen filter is used to enhance the borders.

The main objective of training SVM is to find the best parameters and the best kernel that minimizes Eq. 11, so, after several training experiments of the SVM, it was decided to use the RBF kernel, i.e., $K(x_i, x_j)$ is $\exp\left(-\gamma \left\|x_i - x_j\right\|^2\right)$, $C = 30$ and $\gamma = 0.0128$; these parameters reach a high training classification rate that is about 93%.

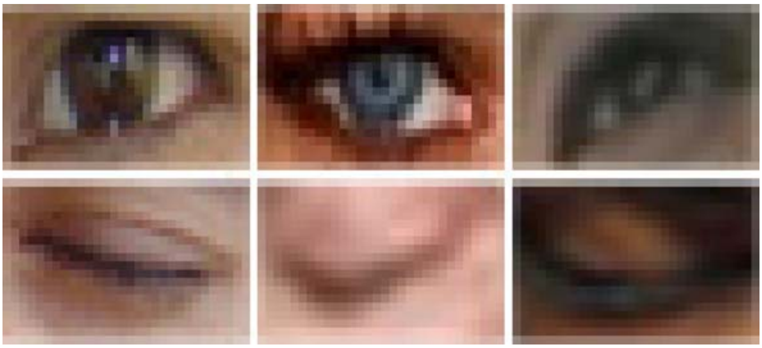


Fig. 20 Examples of eye database

Table 3 Result of eye state analysis

Driver	Total frames	Eyes open	Eyes closed	Correct rate (%)
D1	960	690/700	258/260	98.90
D2	900	520/560	339/340	96.27
D3	500	388/400	99/100	98.00
D4	330	150/170	152/160	91.61
D5	1,400	891/980	401/420	93.19

Table 3 presents several results of this method computed over a several sequences of drivers. It shows that a high correct rate of classifications.

3.5 Drowsiness Index

The eye-blinking frequency is an indicator that allows a driver's drowsiness (fatigue) level to be measured. As in the works of Horng et al. [17] and Dong and Wu [11], if five consecutive frames or during 0.25 s are identified as eye-closed the system is able to issue an alarm cue; PERCLOS [28] also is implemented in this system.

Figure 21 presents an instantaneous result of this system over a driver's image, whereas Fig. 22 pictures the evolution drowsiness index graph for a driver's drowsiness sequence.

3.6 Distraction

Distraction may also cause traffic accidents, it is estimated that it is the cause of about 20% of them [4]. To detect distraction the driver's face should be studied because the pose of the face contains information about one's attention, gaze and level of fatigue [22]. To verify the driver's distraction, this contribution has implemented the following procedure.

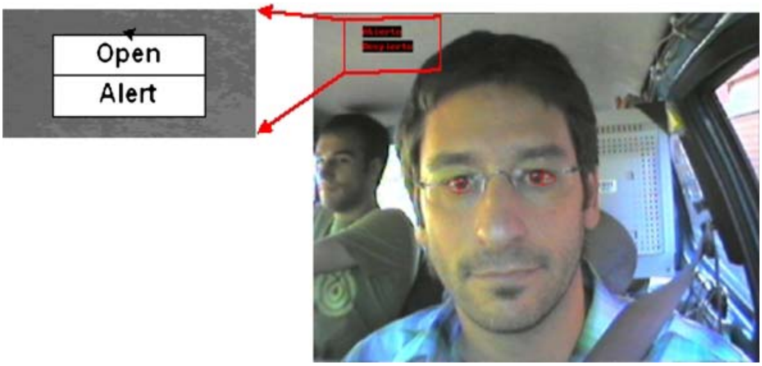


Fig. 21 System instantaneous result

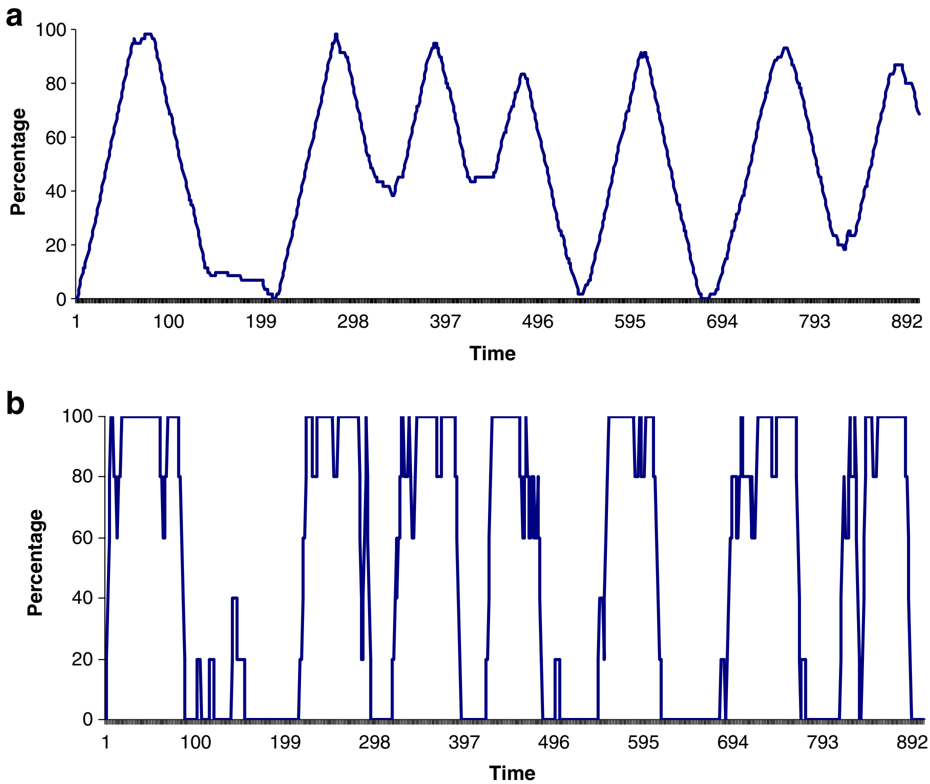
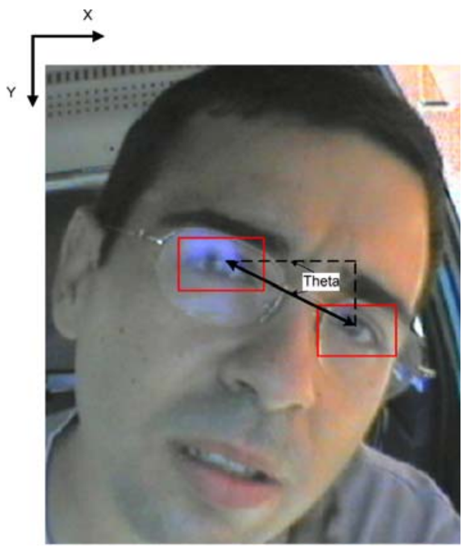


Fig. 22 Drowsiness index graph in a 900-frame sequence of a drowsy driver, **a** Perclos, **b** Horng-Dong and Wu index

Fig. 23 Face orientation



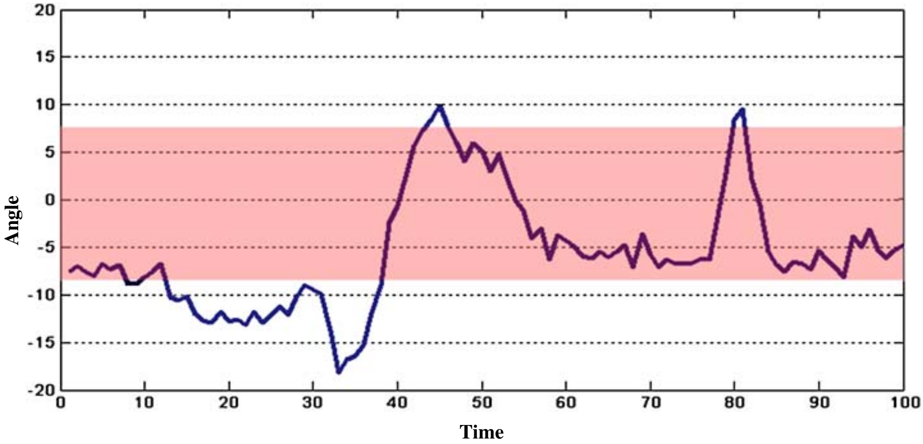


Fig. 24 Head-orientation monitoring over time in a 100-frame sequence

3.6.1 Face Orientation

Driver's face orientation is estimated using the eye position, through Eq. 22.

$$\theta = \tan^{-1}\left(\frac{\Delta x}{\Delta y}\right) \tag{22}$$

where $\Delta x = x_2 - x_1$, $\Delta y = y_2 - y_1$, (x_1, y_1) and (x_2, y_2) correspond to the left and right eye positions. Equation 23 presents the classification limits. Figure 23 depicts an example of face orientation, whereas, Fig. 24 also shows and extended example of driver's face orientation from eyes through a sequence of images.

$$\begin{cases} Left \quad if \quad \theta > 8^\circ \\ Front \quad if \quad |\theta| \leq 8^\circ \\ Right \quad if \quad \theta < -8^\circ \end{cases} \tag{23}$$

Fig. 25 **a** IVVI vehicle, **b** processing system, **c** driver's camera

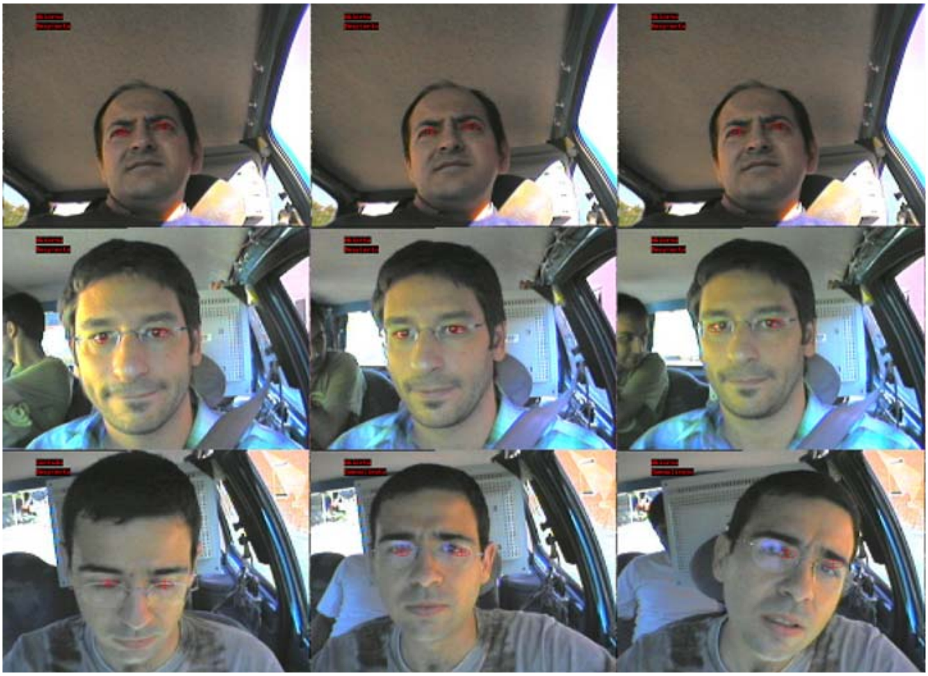


Fig. 26 Different stages of the proposed algorithm on several instants of time, driving conditions and different drivers

3.6.2 Head Tilt

The above method has a problem when using a monocular camera, so, to correct this drawback, this contribution implements a head-tilt based on neural networks. Let us remember that the driver's face database is made up of examples of faces in five orientations, so, the face is passed to neural networks to know its orientation, especially for the up and down cases. If the system detect that the face position is not frontal, an alarm cue is issued to alert the driver of a danger situation.

4 Conclusions

In this paper, a research project to develop a non-intrusive driver's drowsiness system based on Computer Vision and Artificial Intelligence has been presented. This system uses advanced technologies for analyzing and monitoring drivers eye state in real-time and in real driving conditions. Based on the results presented on Tables 1, 2 and 3, the proposed algorithm for face tracking, eye detection and eye tracking is robust and accurate under varying light, external illuminations interference, vibrations, changing background and facial orientations.

To acquire data to use while developing and testing the algorithms, several drivers were recruited; they were exposed to a variety of difficult situations commonly encountered on the roadway. This guarantees and confirms that these experiments have proven robustness and efficiency in real traffic scenes. The images were taken with the camera inside the IVVI vehicle, Fig. 25c. IVVI is an experimental platform used to develop the driver assistance systems in real driving conditions. It is a Renault Twingo vehicle, Fig. 25a, equipped with a processing system, Fig. 25b, which processes the information comes from the cameras. Finally, Fig. 26 shows an example that validates this system.

For future work, the objective will be to reduce the percentage error, i.e., to reduce the false alarms; for this, extra experiments will be developed, using additional drivers and incorporating new modules.

Acknowledgements This work was supported in part by the Spanish Government through the CICYT projects VISVIA (Grant TRA2007-67786-C02-02) and POCIMA (Grand TRA2007-67374-C02-01).

References

1. Armingol, J.M., de la Escalera, A., Hilario, C., Collado, J., Carrasco, J., Flores, M., Pastor, J., Rodriguez, F.: IVVI: intelligent vehicle based on visual information. Robot. Auton. Syst. **55**, 904–916 (2007). doi:10.1016/j.robot.2007.09.004
2. Bergasa, L., Nuevo, J., Sotelo, M., Vazquez, M.: Real time system for monitoring driver vigilance. In: IEEE Intelligent Vehicles Symposium, Parma, 14–17 June 2004
3. Brookshear, J.G.: Theory of Computation: Formal Languages, Automata and Complexity. Addison Wesley Iberoamericana, Reading (1993)
4. Brandt, T., Stemmer, R., Mertsching, B., Rakotomirainy, A.: Affordable visual driver monitoring system for fatigue and monotony. IEEE Int. Conf. Syst. Man Cybern. **7**, 6451–6456 (2004)
5. Branzan, A., Widsten, B., Wang, T., Lan, J., Mah, J.: A computer vision-based system for real-time detection of sleep onset in fatigued drivers. In: IEEE Intelligent Vehicles Symposium, pp. 25–30 (2008)
6. Chang, C., Lin, C.: LIBSVM: a library for support vector machine (2001). www.csie.ntu.edu.tw/~cjlin/libsvm
7. Chen, Y.W., Kubo, K.: A robust eye detection and tracking technique using Gabor filters. In: Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IEEE, vol. 1, pp. 109–112 (2007)
8. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines and other Kernel-Based Learning Methods. Cambridge University Press, Cambridge (2000)
9. Daugman, J.G.: Uncertainty relation for resolution in space, spatial frequency and orientation optimized by two-dimensional cortial filters. J. Opt. Soc. Am. 2(7), 1160–1169 (1985)
10. D'Orazio, T., Leo, M., Distante, A.: Eye detection in faces images for a driver vigilante system. IEEE Intelligent Vehicles Symposium University of Parma, Italy, 14–17 June (2004)
11. Dong, W., Wu, X.: Driver fatigue detection based on the distance of eyelid. In: IEEE Int. Workshop VLSI Design & Video Tech., Suzhou, China (2005)
12. Fletcher, L., Petersson, L., Zelinsky, A.: Driver assistance systems based on vision in and out of vehicles. In: IEEE Proceedings of Intelligent Vehicles Symposium, pp. 322–327 (2003)
13. Gejgus, P., Sparka, M.: Face Tracking in Color Video Sequences. The Association for Computing Machinery Inc., New York (2003)
14. Grace, R.: Drowsy driver monitor and warning system. International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design (2001)
15. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.A.: Feature Extraction: Foundations and Applications. Springer, Berlin (2006)
16. Hagenmeyer, L.: Development of a multimodal, universal human–machine-interface for hypovigilance-management-systems. Ph.D. thesis, University of Stuttgart (2007)
17. Horng, W., Chen, C., Chang, Y.: Driver fatigue detection based on eye tracking and dynamic template matching. In: Proceedings of the IEEE International Conference on Networking, Sensing & Control (2004)
18. Isard, M., Blake, A.: Condensation: conditional density propagation for visual tracking. Int. J. Comput. Vis. **29**(1), 5–28 (1998). doi:10.1023/A:1008078328650

19. Isard, M.A.: Visual motion analysis by probabilistic propagation of conditional density. Ph.D. thesis, Oxford University (1998)
20. Jafar, I., Ying, H.: A new method for image contrast enhancement based on automatic specification of local histograms. IJCSNS Int. J. Computer Sci. Netw. Secur. **7**(7), 1–10 (2007)
21. Ji, Q., Zhu, Z., Lan, P.: Real time nonintrusive monitoring and prediction of driver fatigue. IEEE Trans. Veh. Technol. **53**(4), 1052–1068 (2004). doi:10.1109/TVT.2004.830974
22. Ji, Q., Yang, X.: Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. Real-Time Imaging **8**, 357–377 (2002)
23. Liu, C.: Gabor-based kernel PCA with fractional power polynomial models for face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **26**(5), 572–581 (2004)
24. Longhurst G.: Understanding Driver Visual Behaviour. Seeing Machine Pty Limited, Acton (2002)
25. Looney, C.G.: Pattern Recognition Using Neural Networks, Theory and Algorithms for Engineers and Scientists. Oxford University Press, Oxford (1997)
26. McLachlan, G.J.: The EM Algorithm and Extensions. Wiley, New York (1997)
27. Mujtaba I.M.: Application of Neural Networks and Other Learning Technologies in Process Engineering. Imperial College Press, London (2001)
28. NHTSA: evaluation of techniques for ocular measurement as an index of fatigue and the basis for alertness management. Final report DOT HS 808762, National Highway Traffic Safety Administration, Virginia 22161, USA (1998)
29. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. **9**, 62–66 (1979). doi:10.1109/TSMC.1979.4310076
30. Parker, J.R.: Practical Computer Vision Using C. Wiley, New York (1994)
31. Tian, Z., Qin, H.: Real-time driver's eye state detection. In: IEEE International Conference on Vehicular Electronics and Safety, pp. 285–289 (2005)
32. Satake, J., Shakunaga, T.: Multiple target tracking by appearance-based condensation tracker using structure information. In: Proceedings of the 17th International Conference on Patter Recognition (ICPR'04), vol. 3, pp. 294–297 (2004)
33. Swingler, K.: Applying Neural Networks: A Practical Guide. Academic, New York (1996)
34. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Conference on Computer Vision and Pattern Recognition (2001)
35. Wu, Y., Liu, H., Zha, H.: A new method of detecting human eyelids based on deformable templates. In: IEEE International Conference on Systems, Man and Cybernetics, pp. 604–609 (2004)