2008 IEEE Intelligent Vehicles Symposium
Eindhoven University of Technology
Eindhoven, The Netherlands, June 4-6, 2008

Real-Time Drowsiness Detection System for an Intelligent Vehicle

Marco Javier Flores, José María Armingol and Arturo de la Escalera, *Member, IEEE*

***Abstract*—In the last years, the traffic accidents study is become important because they produce several died and hurt around the world. To help in reducing this fatality, in this paper, a new Advanced Driver Assistance System (ADAS) for automatic driver's drowsiness detection based on visual information and Artificial Intelligent is presented. This system works on several stages to be fully automatic. In addition, the aim of this algorithm is to locate and to track the face and the eyes to compute a drowsiness index. Examples of different driver's images taken over real vehicle are shown to validate the algorithm that works in real time.**

I. Introduction

ADAS goal is to contribute in traffic accident reduction by using new technologies; this is, increasing the vehicles security, and at the same time, decreasing the danger situations that may be generated during driving process.

Current research is interested in the driver's drowsiness study, also denominated fatigue and related closely with distraction [13]. Drowsiness is presented in stress and fatigue situations in an unexpected and inopportune way. The dream sensation generates the decrease vigilance level state, and this factor produces danger situations and increases the probability of causing some accident. Drowsiness may also be produced by dream's illnesses, certain type of medications, and even, bored situations, such as driving for a long time. It has been estimated that drowsiness produces among 10% and 20% of traffic accidents with dead drivers [3] and hurt drivers [17]. Whereas trucking industry produces 57% of fatal truck accidents for this fatality [5], [7]. Fletcher *et al.* in [14] goes further on and has mentioned that 30% of total traffic accidents have been produced by drowsiness and Brandt *et al.* [13] presents a statistics in which 20% of all accidents are caused by fatigue and inattention.

In this context, it is important to design and to build systems that allow us monitoring the drivers, and at the same time, to measure their level of attention during whole driving process.

Manuscript received November 26th. 2007.

Marco Javier Flores is with the Department of System Engineering and Automation from Universidad Carlos III de Madrid, C. Butarque 15 CP 28911, Leganés (Madrid) Spain (phone: 34-916249970; fax: 34-916246070; e-mail: mjflores@ing.uc3m.es).

José María Armingol and Arturo de la Escalera are with the Intelligent Systems Laboratory at Department of System Engineering and Automation, Universidad Carlos III de Madrid, Leganés (Madrid) Spain (e-mail: armingol@ing.uc3m.es, escalera@ing.uc3m.es).

Fortunately, people in drowsiness produce several typical visual cues that can be detected on the human face:

- yawn frequency,
- eye-blinking frequency and 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. The goal of this system is to estimate automatically driver's drowsiness and to avoid driver's asleep during the driving process.

The organization of the paper is as follows. Section 2 presents an extended state of the art. Section 3 introduces the proposed method for face location and eye detection and their later tracking. Finally, in section 4 results and conclusions are shown.

II. Related Work

To analyze driver's drowsiness several systems have been built. They usually require simplifying the problem to work partially or under special environments, for example, Ji *et al.* in [4] and [5] has presented a detection drowsiness system based on infrared light illumination and stereo vision. This system localizes the eye position 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 and AECS. Bergasa *et al.* [7] also has developed a non-intrusive system using infrared light illumination, this system computes driver vigilance level using a finite state automata with six eye states that computes several indices, among them, PERCLOS; on the other hand, the system is able to detect inattention through face pose analysis. Another work using infrared illumination is presented by Grace *et al.* [20] for measuring slow eyelid closure. D'Orazio *et al.* in [6] has proposed an eye detection algorithm that searches the eyes on the whole image assuming that the iris is always darker than the sclera and based on circle Hough transform and geometrical constrains 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 between open or closed state. The mains limitations of this algorithm are: it is applicable when the eyes are only visible in the image, and it is not robust at changing illumination.

Horng *et al.* [2] 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 color information of the eyeballs, it identifies the eye state and computes the driver's state, i.e., asleep or alert. Brandt *et al.* in [13] has shown a system that monitors the driver fatigue and inattention. For this task, he uses the Viola & Jones (VJ) method 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 [3] 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 [17] 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 between open eyes and closed eyes.

III. 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 presented in Fig. 1.

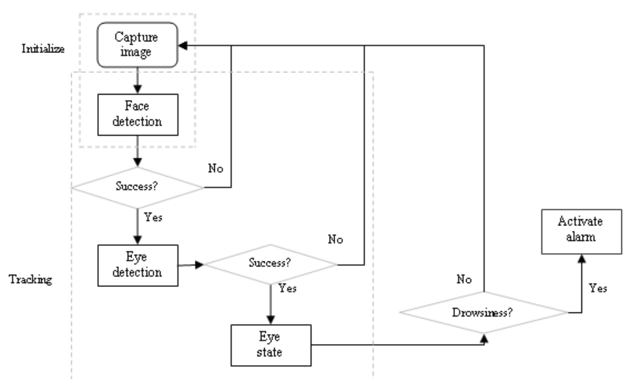


Fig. 1: Algorithm scheme.

A. *Face Detection*

To localize the face, this system uses 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 [1]. Each one of these elements is important to process the images efficiently and 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, its principal disadvantage is that it can not 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.

Fig. 2: Face and eye detection in different drivers.

Continuing with the algorithm description, when driver's face is detected, it is enclosed with a rectangle R which is addresses 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. 2 (a), (b), (c). Indeed, rectangle size comes from experimental analysis developed on the face database that has been created for this task.

B. *The Condensation Algorithm*

This contribution implements the Condensation algorithm (CA) that was proposed by Isard and Blake [8] 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 equation:

$$X_t = f(X_{t-1}, \xi_{t-1}) \quad \textbf{(1)}$$

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 systems observation which are realization of the stochastic process Z_t governed by the measurement equation:

$$Z_t = h(X_t, \eta_t) \quad \textbf{(2)}$$

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 independent among them. 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. CA will be used to track the face and the eyes.

C. *Face Tracking*

The chief problem of VJ method is that it is only able to localize the human face when it is in frontal position at camera. This drawback induces to have an unreliable system to driver's analysis during all the driving process that is highly dynamic. Much effort has been put to correct this problem; so, a tracker has been implemented using a neural network in conjunction with CA. Over the predicted area an exhaustive search is developed to find the exact position of the face, in this step, the algorithm accepts or rejects if the predicted area contains a face using the neural network, which is a backpropagation neural network [18], [19]. The initial state for CA comes from the rectangle R of previous section. Fig. 3 depicts the tracking process.

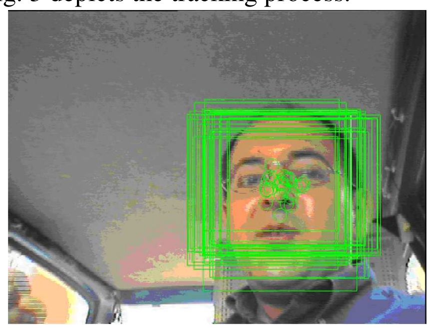


Fig. 3: One time step of the Condensation algorithm.

This tracker is highly flexible because the neural network includes faces and nonfaces in different head orientation and various illumination conditions. Fig. 4 shows several face examples used to train the neural network. The rate of classification is more than 93%.

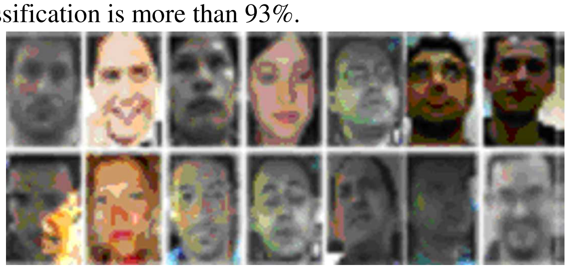


Fig. 4: Examples of face database

D. *Eye Detection*

To localize 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 R in previous section, using the face anthropometric properties [12] which come from face database analysis, two rectangles containing the eyes are obtained. Preliminary, this system uses R_L for left eye rectangle and R_R for right eye rectangle as can be seen in the following four equations.

$$(u_{0L}, v_{0L}) = (x_0 + w/6, y_0 + h/4) \quad \textbf{(3)}$$

$$(u_{1L}, v_{1L}) = (x_0 + w/2, y_0 + h/2) \quad \textbf{(4)}$$

$$(u_{0R}, v_{0R}) = (x_0 + w/2, y_0 + h/4) \quad \textbf{(5)}$$

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

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

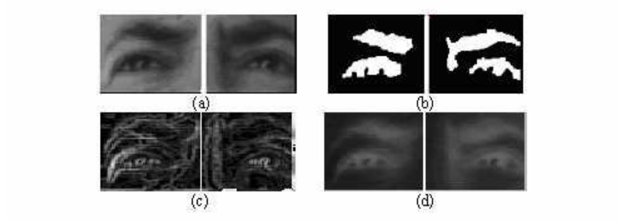
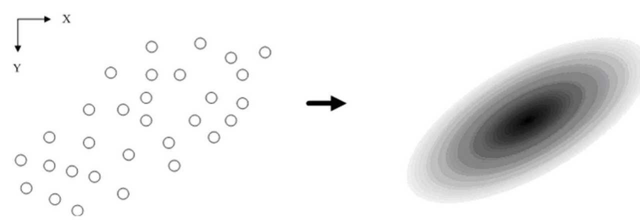


Fig. 5: **.** Eye location through R_L and R_R, (a) grayscale image, (b) binary image (B), (c) gradient image (G) and (d) logarithm image (L).

After the previous step; the exact position of each eye is looking for through incorporating information from grey-level pixels. The main idea is to obtain a random sample from the pixels that belongs the eye area, and then, adjust a parametric model. Fig 6 shows this procedure.



(a) (b)

Fig. 6: (a) Random sample, (b) eye parametric model

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:

$$J(x, y) = \frac{I(x, y) - m}{\sigma} \quad \textbf{(7)}$$

  where m and σ are the mean and the standard deviation, respectively. They are computed over the eye rectangles located previously.
- Generate the image K using the equation:

$$K(x, y) = \begin{cases} J(x, y) - 256 * \delta_1 & if \quad J(x, y) \geq 0 \\ 256 * \delta_2 + J(x, y) & if \quad J(x, y) < 0 \end{cases} \quad \textbf{(8)}$$

  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 equation (9), namely,

$$B(x,y) = \begin{cases} 255 & if \quad K(x,y) \geq \kappa \\ 0 & other \quad case \end{cases} \qquad \textbf{(9)}$$

where κ is computed by Ostu's method [11], Fig. 5 (b).

- Compute the gradient image, G, using the Sobel horizontal and vertical edge operator followed by an image contrast enhancement [16], Fig. 5 (c).
- Compute the logarithm image [15], L, with the objective to enhance the iris pixels that are the central part of the eye, Fig. 5 (d).

With the pixels come from B, G and L images; it was possible to obtain the random sample previously mentioned. This sample presents an ellipse shape. Over this sample an elliptic model has been adjusted using the expectation maximization algorithm (EM). Special attention has received the ellipse center, because, it allows to obtain the exact position of the eye center. The ellipse axes determine the width and height of the eyes. The result is shown in Fig. 7 (b), whereas Fig. 2 (d), (e), (f) the eye position generated for this procedure are depicted.

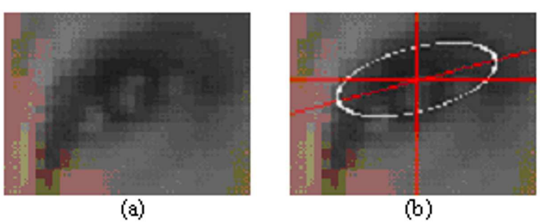


Fig. 7: Expectation maximization algorithm over the spatial distribution of the eye pixels, (a) eye image, (b) ellipse parameters: center, axes and inclination angle.

The reason to use the pixels information through a random sample is because head movement, illumination changes, etc. do not allow to obtain complete pixels eye information, i.e., only partial information of the eye in B, G and L images are available; where an elliptic shape prevails. This random information makes 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. These parameters will be used in the eye state analysis below.

E. Eye Tracking

There are a number of reasons for tracking. One is the VJ's problems mentioned above. Another is the necessity to track the eyes continuously from frame to frame. A third reason is to satisfy the real-time conditions reducing the eye search space. For this task, the state vector of CA is formed of eye position over the image, its velocity and the width and height of the box that contains the eye:

$$X_t = (x, y, \dot{x}, \dot{y}, w, h)^T \qquad \textbf{(10)}$$

CA is initialized when the eyes are localized with the method of previous section. Fig. 8 depicts the eye trajectory tracking over a sequence of 100 images, whereas Table I shows the eye tracking results that has been developed in four sequences of images. The four sequences come from the drivers' database, which was taken to develop these experiments.

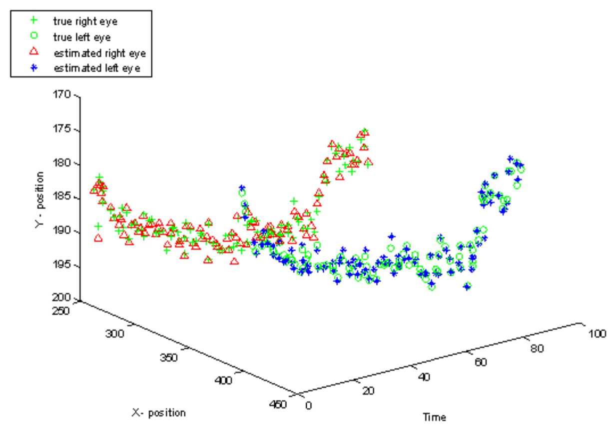


Fig. 8: Trajectory of the real and estimated eyes-center in a 100-frame sequence.

F. Eye State Detection

To identify drowsiness through eye analysis is necessary to know its state: open or closed, through the time and develop an analysis over the time, i.e., to measure the time has been spent in each state. The classification among the open and closed state is complex due to the changing shape of the eye, among other factors, changing position and face rotating, twinkling and illumination variations. All this makes difficult to use only color cues 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, support vector machine (SVM) classification [9], [10], [21] which is rooted in statistical learning theory and pattern classifiers.

SVM 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 training set a hyperplane is built that allows the classification between two classes and minimizes the empirical risk function [21].

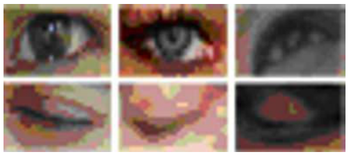


Fig. 9: Examples of eye database

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

$$\begin{aligned} &\min_{\alpha} f(\alpha) = \frac{1}{2}\alpha^T Q\alpha - e^T\alpha \\ &\text{s.t.} \\ &0 \le \alpha_i \le C, \quad i = 1,\cdots,m \\ &y^T\alpha = 0 \end{aligned} \qquad \textbf{(11)}$$

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.

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 different races. A further important aspect of this eye database is that contains images of different eye colors, i.e., blue, black, green. Previous to SVM training, it is indispensable to process each image that consists on histogram equalization, filter with the median filter, followed by the sharpen filter and to normalize in the [0,1] interval. 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 equation (11), so, after several training experiments of the SVM, it has decided to use the RBF kernel, i.e., $K(x_i, x_j)$ is $\exp(-\gamma\|x_i - x_j\|^2)$, $C = 35$ and $\gamma = 0.0128$; these parameters reach high training classification rate that is about 94%.

G. *Drowsiness Index*

The eye-blinking frequency is an indicator that allows to measure driver's drowsiness (fatigue) level. As in the works of Horng *et al.* [2] and Dong and Wu [16], if five consecutive frames or during 0.25 seconds are identified as eye-closed the system is able to issue an alarm cue; Perclos also is implemented in this system.

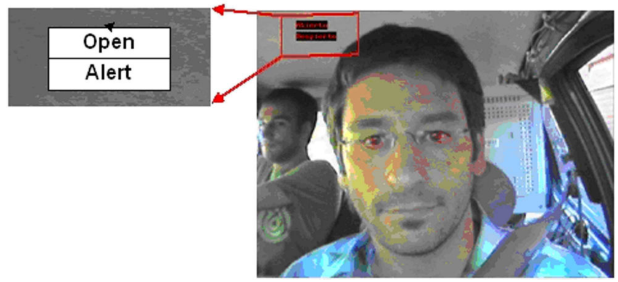


Fig. 10: System instantaneous result

Fig. 10 presents an instantaneous result of this system over a driver's image, whereas Fig. 13 pictures the evolution drowsiness index graph for a drowsiness driver's sequence.

IV. CONCLUSIONS AND RESULTS

In this paper, a non-intrusive driver's drowsiness system based on computer vision and Artificial Intelligent has been presented. This system uses advanced technologies for analyzing and monitoring driver's eye state at real-time and real-driving conditions. 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. Furthermore, all drivers used in these experiments were exposed to a variety of difficult situations commonly encountered in a 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. 11(c). IvvI is an experimental platform used to develop the driver assistance systems in real driver conditions. It is a Renault Twingo vehicle, Fig. 11(a), equipped with a processing system, Fig. 11(b), which processes the information comes from the cameras.

In future works, a driver's distraction identification system will be developed. Finally, Fig. 12 shows an example that validates this system.

TABLE I
RESULT OF EYE TRACKING

	Total Frames	Tracking Failure	Correct Rate
Seq. 1	960	20	97.91%
Seq. 2	900	30	96.60%
Seq. 3	500	8	98.40%
Seq. 4	330	14	95.75%

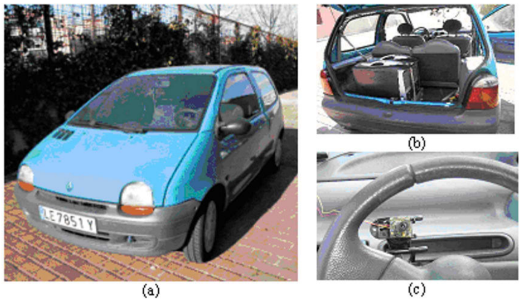


Fig. 11: (a) IvvI vehicle, (b) processing system, (c) driver's camera.

Fig. 12:. Drowsiness index graph in a 900-frame sequence of a drowsy driver, (a) Perclos, (b) Horng-Dong and Wu index.

TABLE II
RESULT OF EYE STATE ANALYSIS

	Total Frames	Eyes Open	Eyes Closed	Correct Rate
Seq. 1	960	690/700	258/260	98.90%
Seq. 2	900	520/560	339/340	96.27%
Seq. 3	500	388/400	99/100	98.00%
Seq. 4	330	150/170	152/160	91.61%

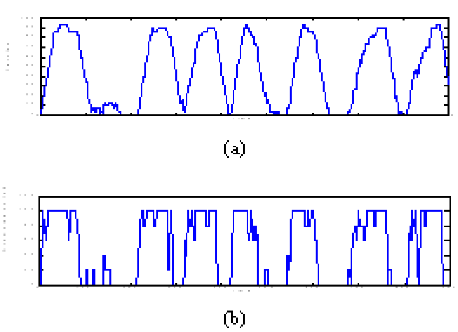


Fig. 13: Evolution drowsiness index graph

REFERENCES

1. Viola P. and Jones M.: Rapid Object Detection using a Boosted Cascade of Simple Features. Accepted Conference on Computer Vision and Pattern Recognition (2001).
2. TABLE Horng W., Chen C. and Chang Y.: Driver Fatigue Detection Based on Eye Tracking and Dynamic Template Matching. Proceedings of the IEEE International Conference on Networking, Sensing & Control (2004).
3. Tian Z. and Qin. H.: Real-time Driver's Eye State Detection. IEEE International Conference on Vehicular Electronics and Safety, Pg. 285-289 (2005).
4. Q. Ji, Z. Zhu y P. Lan. : Real Time Nonintrusive Monitoring and Prediction of Driver Fatigue, IEEE Transactions on Vehicular Technology, Vol. 53, Nº 4 (2004).
5. Ji Q. and Yang. X.: Real-Time Eye, Gaze, and Face Pose Tracking for Monitoring Driver Vigilance. Real Time Imaging, Nor. 8, Pg. 357-377, Elsevier Science Ltd, (2002).
6. D'Orazio T., Leo M. and Distante A.: Eye Detection in Faces Images Vigilante System. IEEE Intelligent Vehicles Symposium University of Parma, Italy June, 14-17 (2004).
7. Bergasa L., Nuevo J., Sotelo M. and Vazquez M.: Real Time System for Monitoring Driver Vigilance. IEEE Intelligent Vehicles Symposium (2004).
8. Isard M. and Blake A.: Condensation: conditional density propagation for visual tracking. International Journal on Computer Vision, 29(1), pp. 5-28 (1998).
9. Cristianini N. and Shawe-Taylor J.: An introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press (2000).
10. Chang C. and Lin C.: LIBSVM: a library for support vector machine, (2001), URL: www.csie.ntu.edu.tw/~cjlin/libsvm
11. Otsu N.: A threshold selection method from gray-level histograms. IEEE Trans. Systems, Man and Cybernetics, Vol. 9, pp. 62-66 (1979).
12. Gejgus P. and Sparka M. Face Tracking in Color Video Sequences. The Association for Computing Machinery Inc. (2003).
13. Brandt T., Stemmer R., Mertsching B., and Rakotomirainy A. Affordable Visual Driver Monitoring System for Fatigue and Monotony. IEEE International Conference on Systems, Man and Cybernetics. Vol. 7, pp. 6451-6456, (2004).
14. Fletcher L., Petersson L. and Zelinsky A.: Driver Assistance Systems based on Vision In and Out of Vehicles. IEEE Proceedings of Intelligent Vehicles Symposium, pp. 322-327, (2003).
15. Wu Y., Liu H. and Zha H.: A New Method of Detecting Human Eyelids Based on Deformable Templates. IEEE International Conference on Systems, Man and Cybernetics, pp. 604-609 (2004).
16. Jafar I., Ying H.: A new method for Image Contrast Enhancement Based on Automatic Specification of Local Histograms. IJCSNS International Journal of Computer Science and Network Security, Vol.7 No.7, July (2007).
17. Dong W. Wu X.: Driver Fatigue Detection Based on the Distance of Eyelid. IEEE Int. Workshop VLSI Design & Video Tech. Suzhou-China (2005).
18. Swingler K.: Applying neural networks: a practical guide. Academic Press (1996).
19. Mujtaba .I.M.: Application of neural networks and other learning technologies in process engineering. Imperial College Press (2001).
20. Grace R.: Drowsy Driver Monitor and Warning System. International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design (2001).
21. Guyon I., Gunn S., Nikravesh M. And Zadeh L. A.: Feature Extraction: Fundations and Applications. Springer (2006).