SUBSPACE EYETRACKING FOR DRIVER WARNING

Fernando De la Torre, Carlos Javier Garcia Rubio, Elisa Martínez

Departament of Communications and Signal Theory. La Salle School of Engineering.
Universitat Ramon LLull. Passeig Bonanova, 8. Barcelona 08022. Spain

ABSTRACT

Driver's fatigue/distraction is one of the most common causes of traffic accidents. The aim of this paper is to develop a real time system to detect anomalous situations while driving. In a learning stage, the user will sit in front of the camera and the system will learn a person-specific facial appearance model (PSFAM) in an automatic manner. The PSFAM will be used to perform gaze detection and eye-activity recognition in a real time based on subspace constraints. Preliminary experiments measuring the PERCLOS index (average time that the eyes are closed) under a variety of conditions are reported.

1. INTRODUCTION

About 50% of crashes are due to driver's distraction according to National Center for Statistics & Analysis. Monitoring driver's activity form a basis of a copilot which potentially can reduce the number of accidents by detecting situations such as drowsiness or lack of attention. Because distraction while driving is a leading safety issue, the aim of this paper is to develop a monitoring system which can warn the driver in anomalous situations.

There exist several methods to potentially characterize/measure the driver's behavior (such as drowsiness, distraction, etc.). We can divide them into[1] :

1. Fitness-for-duty technologies. These methods are based on performing tests to the driver (i.e. measuring ocular physiology) to evaluate his/her capacity.

2. Mathematical models of alertness dynamics joined with ambulatory technologies. This approach involves the application of mathematical models that predict operator alertness/performance at different times based on circadian circles and related temporal antecedents of fatigue.

3. Vehicle-based performance technologies. Additional hardware is added to the transportation system to control the operator (i.e. truck lane deviation, steering, speed variability, etc...).

4. In-vehicle, on-line technologies. Technologies in this category seek to record some biobehavioral dimension(s) of an operator, such as features of the eyes, face, head, heart, brain electrical activity, reaction time etc., on-line (e.g., continuously, during driving).

Our interest is focused on a non-invasive monitoring techniques which analyze driver eyes' activity. There exist a huge literature [2, 3, 4, 5] on generic eye-trackers for human computer interaction, gaze detection, etc. However, in many applications the identity of the driver remains the same over time. For such a reason, in this paper we will explore the use of person-specific facial appearance models (PSFAM) [6] to achieve a more reliable tracker. PSFAM will be learned automatically from a training session with spoken commands. This appearance based model will allow parameterizing the state of the eye. Analyzing the temporal time series of the coefficients it will be possible to infer situations of distraction. Preliminary results measuring the *PERCLOS* index (proportion of time that a subject's eyes are closed over a specified period) under a variety of conditions is reported.

The paper is organized as follows: section 2 describes the previous work, section 3 explains the training system and section 4 report tracking issues. Section 5 describes the decisional system and section 6 the conclusions.

2. PREVIOUS WORK

A lot of research has been done to develop eye tracking systems. Traditional methods use color, optical flow, contours, etc [2, 3]. In the context of fatigue driver detection, a promising approach makes use of barely-visible infrared light to compute the PERCLOS index [4], which works specially well during the night. However, previous approaches do not exploit the fact, that in most real applications the user's identity remains the same during the route. In this paper, we will take advantage of this constancy and we will develop an eye tracker based on PSFAM [6], a model adapted to the user.

Since the early work of Kirby and Sirovich [7] parameterizing the face using Principal Component Analysis, many computer vision people have used this technique to parameterize shape, appearance or motion [8, 9, 10, 11].

Let us consider a set of m images with N pixels, given by the columns of a matrix $D \in \Re^{N \times m}$. These columns form the training set with all possible configurations of the texture of an object. The matrix D^1 can be factorized using Singular Value Decomposition (SVD), $D \approx U \Sigma V^T$. The k first columns of the orthogonal matrix $U \in \Re^{N \times k}$, the eigenvectors of the covariance matrix DD^T, will expand the principal subspace of the columns of D, $\Sigma \in \Re^{k \times k}$ will contain the singular values and the orthogonal matrix $V \in \Re^{p \times k}$ will expand the row space of D. An image I will be represented by projection onto eigenvectors u_j (columns of U), i.e $I \simeq \sum_{j=1}^{k} c_j u_j = Uc$ where c are the projection coefficients. The first k eigenvectors are selected to take into account 90-95 % of the variance in the training set.

3. AUTOMATIC TRAINING

In this section we describe an automatic algorithm for learning a person-specific eye model given an image sequence of the user performing different eyes' activities. During the training session, the speaker of the computer will indicate with spoken commands what the user should do (e.g. close your eyes, open your eyes, etc). Once we have the image sequence recorded, we find the face region by detecting the area where the motion is bigger than a threshold (using a simple motion detection algorithm). The eyes are located by means of a simple blink detection algorithm. Later, an initial appearance model is computed (using the SVD). Some images of the initial training set can be seen in fig. 1.

As we can observed, the training images in figure 1 are not perfectly registered due to the person's head movement. The goal is to learn an improved model without any manual intervention. In order to construct a better model from this unregistered visual data, we will follow recent work [6] on automatic learning person-specific facial appearance models. Unlike [6], in this paper we will use a discrete search algorithm rather than a continuous one in order to speed up the search.

Fig. 1. Some frames from the training set.

Once we have an initial estimation of the position of the eyes, we iteratively compute the SVD of the data while finding the best geometrical transformation (rotation, translation and scale) which register the data with respect to the subspace [6]. In fig. 2 we can see how we reduce the global error ($MSE = 1/m \sum_{i=1}^{m} ||d_i - UU^T d_i||_2^2$) over iterations. In fig. 3 we can see the results once the algorithm has converged. Observe how we have a better estimation of the eyes' position.

---

^{1}We assume zero mean, otherwise the mean is subtracted off.
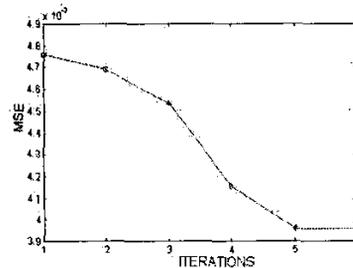

Fig. 2. Normalized MSE error for 6 iterations (8 bases).

Fig. 3. Final Learned Model.

Once the images are registered, we compute the SVD and keep the basis which preserve 90% of the energy. Typically, we can reconstruct 1000 images, with less than 10% error with 25 basis.

4. TRACKING ISSUES

Once we have constructed a model for the eye's variation, tracking will be achieved by registering a new image with respect to the learned model.

4.1. Multiresolutive parameter estimation

In order to track, we explore an exhaustive search algorithm over scales, translations and rotations [11]. However, the exhaustive searching process can be computationally very intensive, mostly depending on the discretization of the parameter space. Following previous work on Active Appearance Models [9, 10, 12] we will learn the relation between the deformation in the image plane and the perturbation in the parameter space. As previous work, we will learn this mapping as a linear one, that is: $\delta p = A \delta T$, where δp is an increment of the geometric parameters(position, scale and orientation) and δT is the variation in image texture. A is a matrix which can be learned using simple linear regression. Therefore, tracking will be achieved by simple matrix multiplications. For learning this matrix, we usually make use of a big training set (2000 images) which we synthetically perturb (with changes in position, scale and rotation). See [10] for more details. In our case, we will learn this matrix in a multiresolution (within the same scale) framework, i.e. we will have three matrices, one for capturing different ranges in the parameters. For the first resolution level the perturbations that we allow are 16 pixels for translation, 20 % in scale and rotation changes of $\pm 15\,^\circ$. In the next resolution levels, we decimate in a factor of two the trans-

lation. Table 1 shows the absolute error that we can achieve for each resolution level.

In order to avoid some local minima and to make the tracker more robust, we also learn a subspace for non-eye variation. For constructing this subspace, we collect data from the training set and we gather the patches which are close to the eyes, but which do not include the eyes. Once the iterative process for tracking is achieved, we test which of the subspaces have less error (the eye/non-eye subspace), avoiding in this way some local minima.

Matrix	X	Y	Scale	Rotation
1	±3	±3	±0.04878	±3 °
2	±1	±1	±0.02243	±1 °
3	±0.3	±0.4	±0.01289	±0.5 °

Table 1. Matrix of Absolute Errors: From left to right we have computed the absolute error from X, Y translations, scale (error relative to a 64x16 pixel image vector) and rotation corrections.

4.2. Dealing with illumination

In the context of driver warning, an important issue is how to recognize eye activity invariant to illumination changes. Rather than trying to model all existing illumination changes (e.g. due to cast shadows, saturation effects, global illumination, etc.) in an accurate manner, we will use a simple normalization (e.g. subtracting the mean and dividing by the norm) and in the learning step (computing the A matrix), we perturb the images with random patterns that reassemble local illumination variations and saturations effects.

4.3. Recognition models

In this section, we describe how to perform eyes' activity recognition. We distinguish between six possible eyes' configurations (close, looking forward, left, right, up and down). Given the training data and the spoken commands, we know which subset of images correspond to each state of the eye. We extend each of these subsets of images with new perturbed ones to take into account the illumination changes as well as little geometric transformations, as described bellow. Finally for each of the extended set of images we create six linear models based on the SVD which preserve the same amount of energy. Once a new image has been registered, we can recognize the six possible states of the eye by simply projecting the data on each of the subspaces and measuring the reconstruction error.

5. OVERALL SYSTEM

The overall system is composed of 5 blocks: preprocessing, initialization, tracking, analysis and anomalous situation detection. See fig. 4.
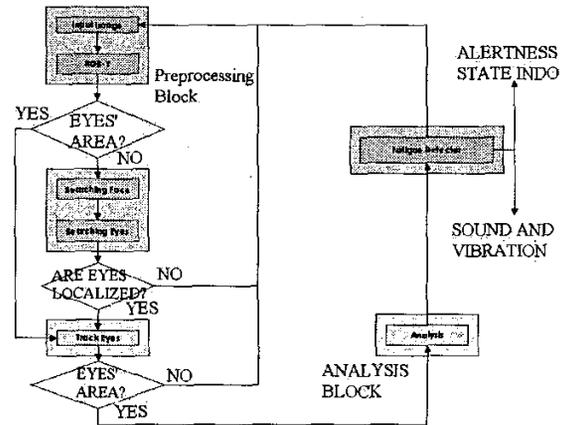

Fig. 4. Overall System

The preprocessing block consist on two steps, in the first one we capture images from the camera through the USB port and in the second one we convert the image from color to graylevel. In the initialization block (we assume that the model has been learned), we locate where the eyes are. First we reduce the search area looking for the face, in a second phase we look for the eyes in this reduced area. If we find the eyes, then we jump to the tracking block, otherwise we remain in the initialization block. In the tracking block, given the eyes' position, we will correct all geometric changes with the method explained in section 4. In the tracking block we can make use of any of the three resolution motion matrices whether we want more accuracy or not. Also, in the case that the tracker is lost, the tracking block has a way of fast recovering by local search in the area of interest. If the tracking is successful, that is, if the error of the eyes' subspace is bigger than the non-eyes' subspace and the estimated parameters are within a reasonable margin, we jump to the analysis block.

The analysis block, classify the eyes' activity in six classes (close, looking forward, left, right, up and down). However, in the preliminary experiments, we are mostly interested in classifying two basic configurations, when the eyes are open or close. In our experiments, we will consider an anomalous situation when the eyes are close, however, temporal information of the behavior of the eyes will be useful for predicting potentially strange situations.

We define *PERCLOS* index as the time ratio that a user has the eyes closed for slow blinking. To compute *PERCLOS*, we must observe the eyes behavior within a temporal

gate of at least 30 seconds, and monitoring all the blinks larger than 0.2-0.4 seconds. The computer will warn the driver if the PERCLOS value is bigger than 0.012 (1% of the time), considering it as an anomalous behavior. In table 2 we classify the warning levels.

PERCLOS Index	Warning Driver Level
≤ 0.012	High
0.012-0.024	Medium - High
0.024-0.048	Medium
0.048-0.096	Low - Medium
0.096-0.192	Low
> 0.192	Null

Table 2. Warning driver levels

In figure 5 we can see the screen of our application for eye tracking. The application has been tested with 5 people (with and without glasses), generating previously his/her PSFAM. The initialization takes about 330 mseg and the application can track the eyes at 15-20 frames per second on a PentiumIII-870Mhz. All the experiments have been developed in a room controlled environment, so the illumination conditions didn't change from training. We have tested the behaviour of the algorithm with different indoors illumination conditions and just in 2% of the sequences the algorithm has failed to recognize anomalous situations (more than 1% of the time with eyes' closed). The error where mostly produced by illumination changes which have not been registered during the training.

Fig. 5. The application shows the personal profile, the eyes pose, the PERCLOS index, the warning level assigned and the frame rate.

6. CONCLUSIONS

We have presented a real time eye tracking-analysis system with application to detecting anomalous situations while driving. We have used subspace methods to develop a invariant pose classifier, which allows the computation of the PERCLOS index. Although a promising approach, more research needs to be done in order to validate the system in a car environment.

7. REFERENCES

[1] L. Hartley, T. Horberry, N. Mabbott, and G. Krueger, "Review of fatigue detection and prediction technologies," Tech. Rep., Institute for Research in Safety and Transport, 2000.

[2] R.-L. Hsu, M. Abdel-Mottaleb, and A. Jain, "Face detection in color images," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, May 2002, vol. 24, pp. 696-706.

[3] J. Bishop and I. Evans, "Automatic head and face gesture recognition," Tech. Rep. FUTH TR001, Future of Technology and Health, 2001.

[4] R. Grace, "Drowsy driver monitor and warning system," Tech. Rep., Robotics Institute Carnegie Mellon, 2001.

[5] A. H. Gee and R. Cipolla, "Determining the gaze of face in images," Tech. Rep. CUED/F-INFENG/TR 174, Cambridge, 1994.

[6] F. de la Torre and M. J. Black, "Robust parameterized component analysis: Theory and applications to 2d facial modeling," in *European Conf. on Computer Vision*, 2002, pp. 653-669.

[7] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *J. Opt. Soc. Am. A*, vol. 4, no. 3, pp. 519-524, March 1987.

[8] M. J. Black and A. D. Jepson, "Eigentracking: Robust matching and tracking of objects using view-based representation," *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63-84, 1998.

[9] T. F. Cootes and C. J. Taylor, "Statistical models of appearance for computer vision," in *World Wide Web Publication, February 2001. (Available from http://www.isbe.man.ac.uk/bim/refs.html)*.

[10] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *European Conference Computer Vision*, 1998, pp. 484-498.

[11] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *PAMI*, vol. 19, no. 7, pp. 137-143, July 1997.

[12] X. Hou, S. Z. Li, and H. J. Zhang, "Direct appearance models," in *Computer Vision and Pattern Recognition, volume I*, 2001, pp. 828-833.