Analysing Driver's Attention Level using Computer Vision

Luis M. Bergasa*, Jose M. Buenaposada†, Jesus Nuevo*, Pedro Jimenez*, Luis Baumela‡
*Department of Electronics, University of Alcala Alcalá de Henares (Madrid), Spain
bergasa,jnuevo,pjimenez@depeca.uah.es
† Departamento de Ciencias de la Computación, Universidad Rey Juan Carlos, Móstoles (Madrid), Spain.
josemiguel.buenaposada@urjc.es
‡Facultad Informatica, Universidad Politecnica de Madrid, Boadilla del Monte (Madrid), Spain
lbaumela@fi.upm.es

***Abstract*— This paper presents a system for evaluating the attention level of a driver using computer vision. The system detects head movements, facial expressions and the presence of visual cues that are known to reflect the user's level of alertness. The fusion of these data allows our system to detect both aspects of inattention (drowsiness and distraction), improving the reliability of the monitoring over previous approaches mainly based on detecting only one (drowsiness). Head movements are estimated by robustly tracking a 3D face model with RANSAC and POSIT methods. The 3D model is automatically initialized. Facial expressions are recognized with a model-based method, where different expressions are represented by a set of samples in a low-dimensional manifold in the space of deformations. The system is able to work with different drivers without specific training. The approach has been tested on video sequences recorded in a driving simulator and in real driving situations. The methods are computationally efficient and the system is able to run in real-time.**

I. Introduction

Driver inattention has been identified by various studies as a primary cause of traffic accidents, accounting for at least 25% of them [1]. According to the U.S. National Highway Traffic Safety Administration (NHTSA), falling asleep while driving is responsible for at least 100,000 automobile crashes annually [2], [3]. An annual average of roughly 70,000 nonfatal injuries and 1,550 fatalities result from these accidents. Moreover, these statistics do not deal with accidents caused by driver distraction either, which is believed to be a larger problem.

Driver inattention generally defines a reduced user vigilance level. Inattention status can be classified [4] into drowsiness and distraction. Distraction is usually divided into cognitive distraction, i.e. when driver thinks about something different than the driving task [5], and visual distraction, namely when the driver is not looking to the road, but paying attention to a different target, such as the radio system or a navigation device.

The research community has proposed different systems for monitoring a driver's level of vigilance, focusing mainly on fatigue detection. Various techniques have been used: physiological measures [6], steering wheel movements and lateral position on the lane [7] or computer vision [8], [9], [10], [11], [12]. This last approach has been favoured, as it is non-intrusive to the driver. Different studies [4], [13] have identified cues that indicate the presence of drowsiness, such as PERCLOS (percentage eye closure), blink frequency, gaze direction, and facial expressions (yawning, nodding, eyebrows rising). Most works detect more than one of those cues, and fuse the parameters for improved robustness [11], [12].

Detection of the mentioned cues on a video sequence is a challenging task, as the driver appearance may change from day to day, fast and severe illumination changes take place, and the face of the driver may become occluded due to head turns, clothing or the driver's hands. Different methods have been used to locate and track the face of the driver on the images. Active near-infrared illumination was used in [8], [11] to ease the location of the driver's eyes. In [14] the authors proposed the use of Active Appearance Models [15] to model the driver's face, and in [16] a Bayesian network is used for tracking facial landmarks.

Face detection and pose estimation has been a very active research field in computer vision, and a comprehensive number of methods have been developed [17]. Face pose estimation algorithms have been recently presented in [18], [19], [20], [21]. The FaceLab software [22] also estimates the face pose. These methods have been shown to work in a laboratory environment, and [21], [22] have been demonstrated in a real vehicle. However, the FaceLab software needs some degree of training for each user.

Most of the systems mentioned above have a strong focus on fatigue detection. Distraction, either visual or cognitive has received much less work, in part for its complexity [23] as new sources of distraction are emerging (such as in-vehicle information systems, IVIS). Also, although many psychological studies have been made, especially related to cell phone use [24], driver distraction trends are not completely understood yet. As a consequence, development of systems able to detect distraction has been modest.

In this paper we propose a system that detects both visual distractions and sleepiness. Visual distractions are detected by estimating the face pose of the driver, and sleepiness by detecting the presence of expressions such as yawing and eyebrows rising, and by calculating the PERCLOS parameter. The system is able to work with different users without prior training. The only requirement is for the driver to be in frontal position to the camera system for a few frames during initialization. This approach is similar to [19] in the automatic model initialisation step, but our system does not require the use of near-IR to locate the eyes, and relies on various facial features to robustly track the driver's face, even when the eyes are occluded.

II. System architecture

Our system uses two separated subsystems that work independently to estimate the face pose and to detect expressions. While this increases processing time, it improves the overall robustness of the system as the face is tracked redundantly. Both trackers share the initialisation phase, and they can access the results of the other tracker in case they get lost, which eases the recovery of the face position. The initialisation phase is a face detector based on Viola&Jones algorithm [25]. A diagram of the system architecture is shown in figure 1.

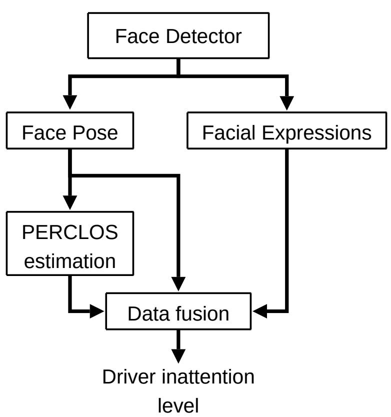


Fig. 1. System architecture

Images of the driver's face are captured using a calibrated stereo camera system, mounted on the car's dashboard in front of the driver. The facial expressions analysis subsystem and the estimation of the PERCLOS parameter work over only one of the images, while the face pose algorithm uses both.

After each frame has been processed, the face expression subsystem outputs the probabilities of several expressions taking place in that frame, and the face pose subsystem yields the estimated face pose. The latter result is used to locate the eyes within the face in one of the frames, and to estimate their degree of openness to calculate the PERCLOS parameter. This parameter is calculated in a similar fashion to [11]. These data are evaluated over a temporal window to smooth the system response, and then added to obtain a unified driver inattention level. If the resulting value is over a threshold, the driver is considered to be inattentive, and an alarm is risen.

III. Facial Expressions Analysis

The subsystem presented in this section is able to robustly track a human face and recognise the facial expressions in an unconstrained environment with sharp illumination changes, such as those occurring on a driver's face. To achieve this goal we use a subspace-based tracker, which is able to track the face while lighting is changing in direction and intensity.

A. Face tracking

Let $\mathtt{I}(\mathbf{x}, t)$ be the image acquired at time t, where $\mathbf{x}$ is a vector representing the co-ordinates of a point in the image, and let $\mathbf{I}(\mathbf{x}, t)$ be a vector storing the brightness values of $\mathtt{I}(\mathbf{x}, t)$. Let $f(\mathbf{x}, \boldsymbol{\mu})$ be a warping function representing the rigid motion of the face, $\boldsymbol{\mu}$ being the vector of rigid motion parameters. And let $\mathtt{B}$ represent the illumination subspace of the face. The final brightness constancy equation is given by

$$\mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}_t), t) = \bar{\mathbf{I}}(\mathbf{x}) + [\mathtt{B}\mathbf{c}_t](\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{F},$$

where vector $\mathbf{c}_t$ are the illumination appearance parameters, $k = \dim(\mathbf{c}_t)$, and $\mathcal{F}$ represents the set of pixels of the face used for tracking. In our experiments we have built an average model of illumination variations using examples comming from the PIE database [26]. The most remarkable characteristic of this appearance model is that it is subject independent and can be used for tracking any subject.

Tracking a face consists of estimating, for each image in the sequence, the values of the motion $\boldsymbol{\mu}$, and appearance $\mathbf{c}$, parameters which minimise the error function

$$E(\boldsymbol{\mu}, \mathbf{c}) = ||\mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}_t), t) - \bar{\mathbf{I}} - [\mathtt{B}\mathbf{c}_t](\mathbf{x})||^2. \qquad (1)$$

We use a Gauss-Newton procedure to solve this minimisation problem, using matrix factorization to reduce the computational cost of the minimization [27].

In our experiments we use a RTS (rotation, translation and scale) motion model, so $\boldsymbol{\mu} = (t_u, t_v, \theta, s)$, and $f(\mathbf{x}, \boldsymbol{\mu}) = s\mathtt{R}(\theta)\mathbf{x} + \mathbf{t}$, where $\mathbf{x} = (u, v)^\top$, $\mathbf{t} = (t_u, t_v)^\top$ and $\mathtt{R}(\theta)$ is a 2D rotation matrix.

B. Facial expressions recognition

The classification procedure used for facial expression recognition is based on a user-and-illumination-independent facial expression model. This model is built by tracking a set of sequences from the Cohn-Kanade data base [28]. We used the tracker introduced in the previous subsection to process the sequences from the data base. Once motion and illumination parameters have been estimated we can compute an illumination normalised version of the rectified image, $\mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}_t), t)$. Using illumination coefficients we can compute $\hat{\mathbf{I}}(f(\mathbf{x}, \boldsymbol{\mu}_t), t) = \mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}_t), t) - \mathtt{B}\mathbf{c}_t$. The dimension of the illumination normalised images is high (61×72 pixels images) compared with the number of samples for training. We perform Principal Component Analysis (PCA) and then Linear Discriminant Analysis (LDA) to reduce the dimensionality of the input data. The PCA+LDA subspace will be $n-1$-dimensional when we have n facial expressions. Trajectories associated with the same prototypical facial expression represent roughly similar facial deformations and, consequently, will be located in nearby positions in the PCA+LDA subspace.

We recognise facial expressions using a probabilistic procedure which combines the prior information stored in the expression manifold with the incoming data obtained from a temporally ordered sequence of images of a face. Let $\mathtt{I}_1, \ldots, \mathtt{I}_t$ be a temporally ordered image sequence of a face wearing one or more facial expressions and $\mathbf{x}_1, \ldots, \mathbf{x}_t$ be the temporally ordered set of co-ordinates of the image sequence in the facial expression subspace, which we will denote $\mathcal{X}_{1:t}$. Let $G_t = \{g_1, g_2, \ldots, g_c\}$ be a discrete random variable representing the facial expression at time t and X_t be a continuous random variable associated to the co-ordinates in the facial expression subspace of the image acquired at time t. We will denote by $P(g_i) \equiv P(G_t = g_i)$ the probability that the discrete random variable G_t takes value g_i and by $p(\mathbf{x}) \equiv p(X_t = \mathbf{x})$ the probability density function (p.d.f.) of the continuous variable $\mathbf{x}$ at time t.

The facial expression $g(t)$ at time t is obtained as the maximum of the posterior distribution of G_t given the sequence of images up to time t

$$g(t) = \arg\max_i \{P(G_t = g_i | \mathcal{X}_{1:t})\},$$

which we compute using a recursive Bayesian filter. A more detailed explanation of this method can be found in [27].

IV. Face Pose Analysis

Our approach to face pose estimation works by constructing and tracking a 3D model of the face. We will present its main characteristics in this section, and refer to [29] for additional details. The model is formed by a set of 3D points on the face. The 2D projections of these points on each camera are tracked on each frame, using the Simultaneous Modeling and Tracking (SMAT) [30] algorithm. From the projections, the 3D face pose is obtained, using the POSIT algorithm for pose extraction and RANSAC for outlier point elimination. After a set of correctly tracked points is obtained, the position of the outlier points is set accordingly to the estimated pose.

A. 3D Face model creation

The first step to create the model is to localize the user's face. As mentioned above, Viola & Jones [25] algorithm is used on both cameras to localize the position of a frontal face within the images. The face model is defined by thirty points that are tracked over successive frames. To choose appropriate points, a predefined standard face pattern is scaled and placed over the detected inner box containing the face on the left camera image. These points may not correspond to any good feature on the user's face to be used for tracking, so a characteristic feature close to each pattern point is chosen for tracking, as shown in figure 2. The Harris algorithm [31] is used to locate points with good contrast and tracking characteristics. Stereo correspondences of these points over the other camera are used to calculate its 3D coordinates.

(a) Left image (b) Right image

Fig. 2. Model Construction

We use a coordinate system affixed to the right camera to define the face model. The model points are referenced to another coordinate system, with origin on the central point of the model. The face pose is characterized by a translation within the camera coordinate system, and a pointing vector. This vector is defined as the normal vector to the face, and at the moment of the model creation is set to $\mathbf{v}_{ini} = (x, y, z) = (0, 0, -1)$.

B. Face tracking using SMAT

The Simultaneous Modeling and Tracking (SMAT) [30] is a recently developed technique for tracking objects in sequences without any previous training. SMAT works by building a library of exemplars obtained from previous frames in the sequence. The exemplars in the library, image patches in our case, are clustered based on their relative distance, and the medians of the clusters are used for fitting the model to the next frame. A new exemplar is included in one of the clusters depending on the distance to its medians, or a new cluster is created if the new exemplar is too far away from the existing ones. As a group of similar exemplars, each of these clusters will approximately represent different appearances of the same feature of the object. The resulting mixture model is fitted to the next frame.

The formulation of the SMAT algorithm is independent of the definition of distance and the minimization method used. We have used Zero-mean Normalized Cross-Correlation (ZNCC) and Sum of Squared Differences (SSD) as distances, and Gauss-Newton and the Nelder-Mead simplex method [32] for the minimization process.

C. Pose estimation

After the position of the tracking points has been updated for both the left and right frames independently, the 3D face pose is to be estimated from the 2D projection of each point. However, the matching process may not succeed for all points, and can result in errors or drifting for some of them. These errors negatively influence the accuracy of the estimated pose. Thus, a robust optimization method is required to estimate the best matching 3D face pose, that would detect as outliers the points that have been incorrectly tracked, so they can be safely discarded. We also consider that points that have been correctly tracked may have some random noise. The RANSAC algorithm is used to eliminate the outliers. 3D pose is obtained using DeMenthon's four point iterative pose estimation algorithm (POSIT) [33].

In each RANSAC iteration, seven points are randomly selected from the model, and used to calculate the pose (**R** and **T** matrix) using the POSIT algorithm. With this **R** and **T**, all 3D original points of the model are projected over the image plane, and the Euclidean distance from the tracking point to the corresponding projected point is calculated. If this distance is less than a threshold, this point is considered to be correct, and marked as an inlier. The RANSAC algorithm runs for enough iterations to guarantee a 99% of success with 50% of outliers.

This process is performed over the left and right frames independently, and the final pose estimation is calculated from the pose estimations as a weighted sum depending on the number of inliers. In case the number of inliers of any of the cameras is less than a threshold, set to half the total number of points, that estimation is discarded and the estimation of the other camera is used.

D. Tracking Failure detection and Recovery

Points identified as outliers by the RANSAC algorithm are moved to a corrected position, so they can be tracked on the following frames. The new position of the points is calculated by re-projecting the 3D model on both camera planes with the final estimated pose, $\mathbf{R}_{model}$ and $\mathbf{T}_{model}$.

V. Experimental Results

The method has been tested on videos recorded in a driving simulator and in a moving car. Two IEEE1394 cameras captured 25 frames per second. The algorithms run on a Core2 Duo at 2.4GHz running GNU/Linux. Length of the sequences ranges from 1 to 10 minutes, and total length is over an hour.

In this section we present the results for the different subsystems, and summarise the accuracy of the unified driver inattention level estimation.

A. Facial expression detection results

The training images for the face expressions classifier were obtained from the Cohn-Kanade data base [28]. We have trained a classifier for the upper face and another one for the lower face although the tracker uses the whole face model.

Figure 3 shows a few frames from a test sequence, and results of the expression recognition process for that same sequence are shown in figure 4.

From the test sequences analysis we can conclude that tracking is successful when the subject is driving normally (e.g., from frame 900 to 939 in the sequence shown in figure 3), and when some head motion appears (e.g., frames 1158 to 1180). During driving, illumination conditions change but system performance is not severely affected by these changes.

The expression model includes the neutral facial expression, yawn and eyebrows rising. These expressions are recognised properly. In the test sequences, the driver performed gestures around the eyes region including rising his eyebrows several times, trying to maintain himself awake, as shown in frames 972 and 1114 in figure 3. The drivers also yawned frequently and these gestures were correctly recognised (see from frame 1004 to 1050 in figure 4). In some situations the system does not give a correct classification. This is mainly because of tracking inaccuracies, as is the case for frame 945 where a sudden nodding takes place, and frame 1128.

B. Face Pose estimation results

The 3D model used by the face pose estimation algorithm is constructed over the first frame. The system chooses up to 30 characteristic tracking points to built the model. Pose is correctly estimated over face rotations, with Mean Absolute Errors (MAS) below 5° for pitch and roll angles. Points that are appear rotated over ±60° are considered occluded, and not taken into account for pose estimation. The more the face rotates, the more points become hidden, and thus the accuracy of the pose estimation falls. With this limitation, the system is able to track the face correctly up to ±40° degrees. Results for a video sequence are shown in figure 5.

C. Driver inattention level results

An unified driver inattention level is estimated from the results of the two subsystems. As can be observed in figures 4 and 5, the evolution in time of the estimated values is substantially different, and they have to be equalised accordingly before being combined. This is done by using a moving time window, that computes the total time expressions have been detected for in the last 30 seconds. This window width is the same used to calculate PERCLOS [11]. These values are then added to the other parameters, and if it is found to be over a threshold, the driver is considered to be inattentive to the road. This threshold has been determined experimentally.

Performance of the system in detecting inattentive events for the test sequences is about 92%, improving the results of the two subsystems alone.

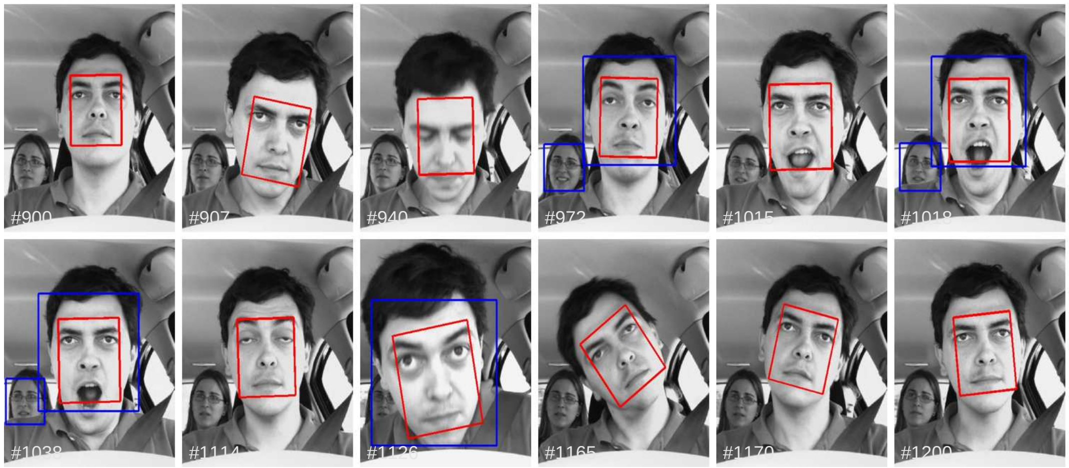


Fig. 3. Tracking results for a realistic sequence. Blue squares are the face detector results (used when the appearance tracker get lost). Red lines are the results of the appearance tracker.

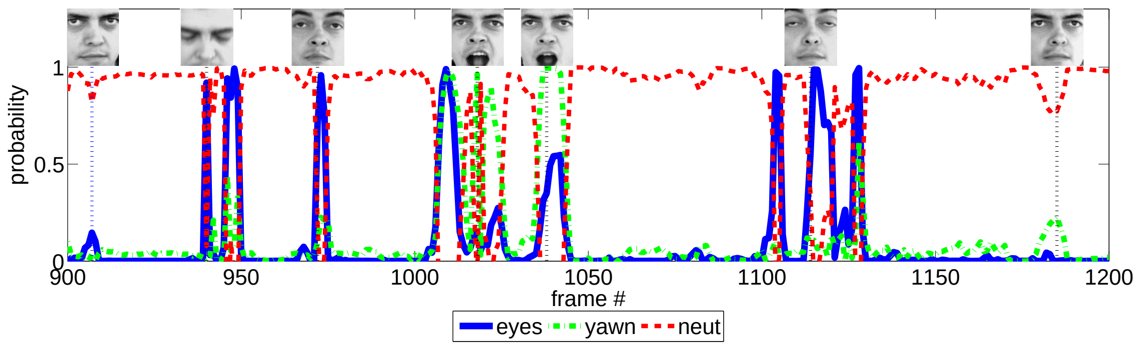
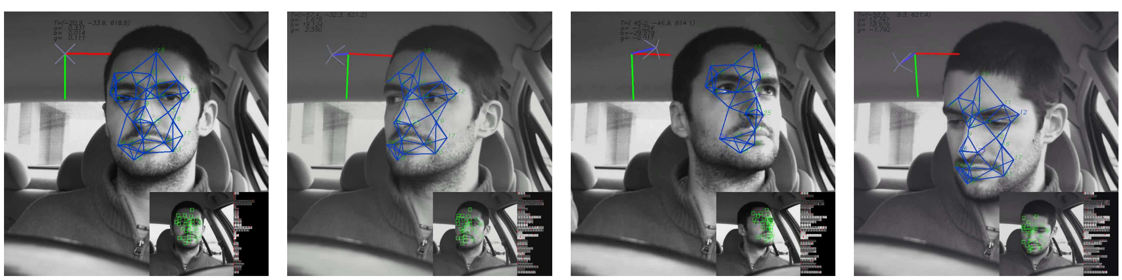


Fig. 4. Facial expression recognition in a realistic image sequence.



(a) Frame 25 (b) Frame 110 (c) Frame 225 (d) Frame 310

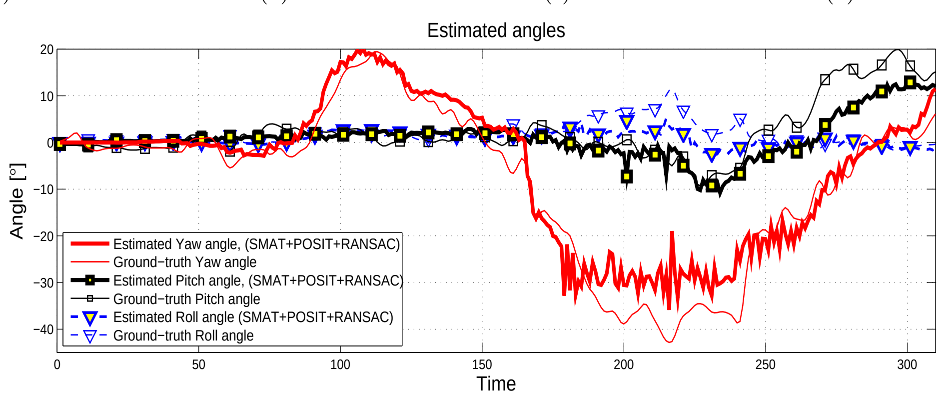


Fig. 5. Tracking and Pose estimation of the face of a driver A under day light driving conditions.

VI. Conclusions

This paper has presented a system for analysing the inattention level of a driver. Visual distraction detection is based on head pose estimation, and sleepiness detection relies on the PERCLOS parameter and facial expressions. The system is able to robustly track the face of the driver using appearance based methods, recognise facial expressions and estimate the pose of the head. The system also monitors the driver's eyes calculate the PERCLOS parameter. Estimated values are added to obtain a unified parameter. The system is able to run in real time.

The algorithms have been tested in video sequences recorded in a simulator and in a moving car. We plan to conduct several multi-hour tests to obtain a better evaluation of the system performance. We will continue the development of the system by incorporating more expressions to the face expression detector, and by including additional parameters along to PERCLOS, such as fixed gaze, and blink detection. More advanced data fusion techniques will also be studied.

Acknowledgments

The authors gratefully acknowledge funding from the Spanish *Ministerio de Educación y Ciencia* under contract MOVICON `TRA2005-08529-C02-02` and CABINTEC project `PSE-370100-2007-2`. They also thank Jeffrey Cohn and Takeo Kanade for providing the Cohn-Kanade image data base. J. Nuevo is also working under a researcher training grant from the Education Department of the Comunidad de Madrid and the European Social Fund.

References

[1] European Project FP6 (IST-1-507674-IP), "AIDE - Adaptive Integrated Driver-vehicle Interface," 2004–2008. [Online]. Available: http://www.aide-eu.org/index.html

[2] P. Rau, "Drowsy driver detection and warning system for commercial vehicle drivers: Field operational test design, analysis and progress," 2005.

[3] D. Royal, "Volume I - Findings; National Survey on Distracted and Driving Attitudes and Behaviours, 2002," The Gallup Organization, Tech. Rep., Mar. 2003.

[4] W. Wierwille, L. Tijerina, S. Kiger, T. Rockwell, E. Lauber, and A. Bittne, "Final report supplement – task 4: Review of workload and related research," USDOT, Tech. Rep. DOT HS 808 467(4), oct 1996.

[5] T. Victor, *Keeping Eye and Mind on the Road.* Uppsala: Acta Universitatis Upsaliensis, 2005.

[6] A. Kircher, M. Uddman, and J. Sandin, "Vehicle control and drowsiness," Swedish National Road and Transport Research Institute, Tech. Rep. VTI-922A, 2002.

[7] DaimerChryslerAG. (2001, June) The electronic drawbar. [Online]. Available: http://www.daimlerchrysler.com

[8] Q. Ji and X. Yang, "Real-time eye, gaze and face pose tracking for monitoring driver vigilance," *Real-Time Imaging*, vol. 8, pp. 357–377, Oct 2002.

[9] P. Smith, M. Shah, and N. da Vitoria Lobo, "Determining driver visual attention with one camera," *IEEE Trans. Intell. Transport. Syst.*, vol. 4, no. 4, pp. 205–218, 2003.

[10] A. C. (IST 2000-28062), "Awake-system for effective assessment of driver vigilance and warning according to traffic risk estimation," Sept. 2001-2004.

[11] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and E. López, "Real-time system for monitoring driver vigilance," *IEEE Trans. Intell. Transport. Syst.*, vol. 7, no. 1, pp. 1524–1538, Mar. 2006.

[12] R. Senaratne, D. Hardy, B. Vanderaa, and S. Halgamuge, "Driver Fatigue Detection by Fusing Multiple Cues," *Lecture Notes In Computer Science*, vol. 4492, p. 801, 2007.

[13] D. Dinges, "F. perclos: A valid psychophysiological measure of alertness as assesed by psychomotor vigilance," Federal Highway Administration. Office of motor carriers, Tech. Rep. MCRT-98-006, 1998.

[14] J. Nuevo, L. Bergasa, M. Sotelo, and M. Ocana, "Real-time robust face tracking for driver monitoring," *IEEE ITSC'06*, pp. 1346–1351, 2006.

[15] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 681–685, Jan. 2001.

[16] J. McCall and M. Trivedi, "Driver monitoring for a human-centered driver assistance system," *Proceedings of the 1st ACM international workshop on Human-centered multimedia*, pp. 115–122, 2006.

[17] M. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 1, pp. 34–58, 2002.

[18] P. Yao, G. Evans, and A. Calway, "Using affine correspondence to estimate 3-D facial pose," in *Image Processing, 2001. International Conference on*, vol. 3, 2001.

[19] Z. Zhu and Q. Ji, "Real Time 3D Face Pose Tracking From an Uncalibrated Camera," *CVPR Workshop 2004, Conference on*, pp. 73–80, 2004.

[20] Y. Hu, L. Chen, Y. Zhou, and H. Zhang, "Estimating face pose by facial asymmetry and geometry," *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pp. 651–656, 2004.

[21] E. Murphy-Chutorian, A. Doshi, and M. M. Trivedi, "Head pose estimation for driver assistance systems: A robust algorithm and experimental evaluation," in *Proceedings of ITSC*, 2007, pp. 709–714.

[22] Seeing Machines. (2004, Aug.) Facelab transport. [Online]. Available: http://www.seeingmachines.com/transport.htm

[23] T. A. Ranney, "Driver Distraction: A Review of the Current State-of-Knowledge," National Highway Traffic Safety Administration, Tech. Rep. DOT HS 810 787, Apr. 2008.

[24] A. Mccartt, L. Hellinga, and K. Bratiman, "Cell Phones and Driving: Review of Research," *Traffic Injury Prevention*, vol. 7, no. 2, pp. 89–106, 2006.

[25] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, May 2004.

[26] T. Sim, S. Baker, and M. Bsat, "The cmu pose, illumination, and expression database," *Trans. on PAMI*, vol. 25, no. 12, pp. 1615–1618, 2003.

[27] J. Buenaposada, E. Muñoz, and L. Baumela, "Efficient illumination independent appearance-based face tracking," *Image and Vision Computing*, 2008.

[28] T. Kanade, J. Cohn, and Y.-l. Tian, "Comprehensive database for facial expression analysis," in *Proc. of International Conference on Automatic Face and Gesture Recognition*, 2000, pp. 46–53.

[29] P. Jimenez, J. Nuevo, and L. Bergasa, "Face Pose Estimation and Tracking Using Automatic 3D Model Construction," in *IEEE CVPR 2008, 3D Face Processing Workshop*, Anchorage, Alaska, June 2008.

[30] R. Dowson, N.D.H.; Bowden, "Simultaneous modeling and tracking (SMAT) of feature sets," in *IEEE Conference on Computer Vision and Pattern Recognition 2005*, vol. 2, 2005, pp. 99–105.

[31] C. Harris and M. Stephens, "A combined corner and edge detector," *Alvey Vision Conference*, vol. 15, p. 50, 1988.

[32] J. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[33] D. F. Dementhon and L. S. Davis, "Model-based object pose in 25 lines of code," *Int. J. Comput. Vision*, vol. 15, no. 1-2, pp. 123–141, 1995.