Peláez, G.A., García, F., Escalera, A., & Armingol, J. M . (2014). Driver Monitoring Based on Low-Cost 3-D Sensors. *Transactions on Intelligent Transportation Systems*, 15 (4), pp. 11687-11708.
DOI: 10.1109/TITS.2014.2332613

Driver Monitoring Based on Low-Cost 3-D Sensors

Gustavo A. Peláez C., Fernando García, Arturo de la Escalera, and José María Armingol

***Abstract*—A solution for driver monitoring and event detection based on 3-D information from a range camera is presented. The system combines 2-D and 3-D techniques to provide head pose estimation and regions-of-interest identification. Based on the captured cloud of 3-D points from the sensor and analyzing the 2-D projection, the points corresponding to the head are determined and extracted for further analysis. Later, head pose estimation with three degrees of freedom (Euler angles) is estimated based on the iterative closest points algorithm. Finally, relevant regions of the face are identified and used for further analysis, e.g., event detection and behavior analysis. The resulting application is a 3-D driver monitoring system based on low-cost sensors. It represents an interesting tool for human factor research studies, allowing automatic study of specific factors and the detection of special event related to the driver, e.g., driver drowsiness, inattention, or head pose.**

***Index Terms*—Driver monitoring, face detection, head pose estimation, iterative closest points (ICP), point clouds.**

I. Introduction

Driver monitoring is crucial in human factors. It is important to monitor the driver to understand his needs, behavior, or errors. There are already available on the market technologies able to provide automatic driver monitoring, but the costs of the sensors used are high, and many of them require external and invasive devices, which may interfere with the driving process. Modern applications, based on computer vision, take advantage of the development of information technologies and computing capacities to create applications that are capable of monitoring the driver by noninvasive methods, although limitations inherent to the sensing technology are present, e.g., lack of depth information or light-dependent conditions. New technologies such as stereo and time-of-flight cameras are able to overcome some of these difficulties. Specifically, recent gaming sensors such as the Kinect from Microsoft are providing a new set of previously expensive sensors integrated in a low-cost device that can provide 3-D information together with some additional features.

The proposed solution monitors the driver based on the fusion of data from the Kinect sensors, i.e., depth information, color image, microphone, and infrared (IR) information. The fusion of this information from the different sources allows the combination of 2-D and 3-D algorithms to provide reliable face detection, accurate pose estimation, and trustable identification of facial features, such as eyes and nose. In order to determine the reliability of the orientation algorithm, tests were performed for the head pose orientation algorithm with an inertial measurement unit (IMU) attached to the head of the test subjects. The inertial measurements provided by the IMU were used as a ground truth for three degrees of freedom (3DoF) tests (yaw, pitch, and roll). Finally, tests results were compared with those available in the literature to check the performance that the algorithm presented.

Taking note that the Kinect sensor is a sensor specifically designed for indoor use, limitations thus arise when performing monitoring under real driving conditions. They will be discussed in this proposal. However, the algorithm presented is applicable to any point-cloud-based sensor (e.g., stereo cameras, time-of-flight cameras, and laser scanners) more expensive, but less sensitive to these limitations.

Manuscript received October 28, 2013; revised February 7, 2014, April 10, 2014, and June 18, 2014; accepted June 19, 2014. Date of publication July 15, 2014; date of current version August 1, 2014. This work was supported by the Spanish Government through the CICYT projects under Grants TRA2010-20225-C03-01 and TRA 2011-29454-C03-02. The Associate Editor for this paper was C. Olaverri-Monreal.

The authors are with the Intelligent Systems Laboratory, Department of Systems Engineering and Automation, Universidad Carlos III de Madrid, 28911 Leganés, Spain (e-mail: gpelaez@ing.uc3m.es; fegarcia@ing.uc3m.es; escalera@ing.uc3m.es; armingol@ing.uc3m.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2014.2332613

II. State of the Art

Techniques used to monitor the driver can be divided according to the sensing device used. There are three well-defined sensor sets.

The first set of sensors is biomedical sensing technologies, based on the measurements of biomedical signals. Although more robust due to the measurement of the direct input signal from the driver, they require intrusive methods [1], [2]. These intrusive methods can lead to important drawbacks such as a change of the behavior of the driver or lacks of comfort that make them not suitable for real applications.

The second set of sensors used for monitoring the driver are onboard sensors, which are mounted inside the vehicle and analyze the behavior of the driver by recording the retrieved information from the controller area network bus [3], [4]. The availability of this information makes them useful for commercial applications, although the dependence on the manufacturer and the need of arduous training processes make them strongly dependent on the available signals.

Finally, computer-vision-based sensors are an interesting tool for driver monitoring, which are acquiring high importance in recent times. The nonintrusive acquisition method is an important advantage, allowing driver monitoring with no disturbances or inconveniences in the driving process. In addition, the high information available on the images can be used to infer the state and the behavior of the driver. On the other hand, the difficulties inherent to computer vision algorithms should be addressed. The diverse cameras available require different applications that take advantage of the technologies of each device.

Standard color cameras can be used during daylight conditions and can use general computer vision algorithms, taking advantage of color information. In [5], the Viola–Jones face detection algorithm is used; later, condensation algorithms are used for tracking the eye state with detection based on a Gabor filter. To detect drowsiness in the driver, support vector machines are used for open/close detection, and percentage of closed eye (PERCLOSE) is used to identify the state of the driver. In [6], by using two cameras, rapid 3-D face modeling is performed using frontal and profile face information for accurate 2-D pose synthesis. In [7], feature extraction from the camera is used together with several parameters [i.e., percentage of eye closure over pupil over time, quantity of eyed closed over time (Microsleep), and the current car position] to monitor the driver. More focused on head pose estimation, the work in [8] provides 3-D tracking based on monocular localized gradient orientation histograms and support vector regressors. The work in [9] provides eye detection and head pose estimation; the latest is based on matching of specific features (eye distance) with a model and tracking the movement using optical

Fig. 1. Test vehicle IVVI 2.0.

flow. Other approaches take advantage of the specific features of the IR cameras, on which, due to the specific illumination condition, the pupil can be easily detected; thus, the detection of the eyes is easier [10], [11]. Finally, stereo systems are very useful because they provide 3-D information, but with the disadvantage of the high processing requirements [12]; some of the available commercial systems include these stereo systems [13], [14].

Face detection and head pose estimation is not only limited to the field of intelligent transport systems. The work in [15] provides a complete survey of the topic, providing full information of the results and categorizing the different available schemes according to the technique used for head pose estimation: appearance template methods [16], detector array methods [17], nonlinear regression methods [18], [19], manifold embedding methods [20], [21], flexible model methods [22], geometric methods [23], tracking methods [24], [25], and hybrid methods [20], [25], [26]. Some of results summarized in [15] are used to compare with this work, as it is depicted in the test section.

Few of these works take depth information into account due to hardware restrictions, price of the devices, or strong processing requirements. With the release of the Kinect, it is possible to obtain color images together with 3-D data and much more information based on an extremely low-cost sensor. Some authors have already tried to take advantage of the new possibilities that the Kinect offers: in [27], Fanelli *et al.* were able to discriminate in real time between the body and the head and to provide head pose estimation based on discriminative random regression forest (DRRF). Due to the fast DRRF approach, detection is done in 25 ms mean time; thus, real time is achieved. Finally, in [28], Kondori *et al.* used pixel-to-pixel least square approach; this solution is announced as real time (up to 40 ms), but no code or numerical results for the head pose estimation are provided.

In this paper, a novel point-cloud-based algorithm for head pose estimation is presented. It enhances the classical vision detection with depth information, allowing the use of accurate point clouds in order to provide precise results in real time.

III. System Description

The application presented in this paper is included in the project IVVI 2.0. It is the second research platform of the Intelligent Systems Laboratory. The vehicle is a commercial vehicle equipped with several sensors to test and develop different technologies to assist the driver (see Fig. 1).

Kinect is one of the sensors available in the platform, and it was attached to the dashboard, as shown in Figs. 1 and 2. The location of the Kinect sensor was carefully chosen according to some important constraints.

- Minimal distance required for Kinect range detection (0.5 m).
- Not to interfere with the driver's field of view; thus, the lowest possible location was selected to avoid this issue.
- Finally, interferences and blocks of the range signal due to the steering wheel maneuvering should be avoided (e.g., the hand of the driver blocks the line of sight). To avoid this, a compromise with the aforementioned field of view had to be found. In addition, the rotation of the sensor was also carefully selected to avoid this problem.

Other locations of the sensor were discarded since they did not fulfill these constraints.

Fig. 2 shows the diagram that gives an overall idea of the application. The proposed solution provides an advanced system able to detect the driver, estimate the movement of the face, detect the movement angles, and identify relevant regions of the face for further analysis. The algorithm starts by capturing the cloud of points from the Kinect sensor and applying a distance filter to remove the background. Later, a color image is built from the cloud, where a face will be searched for by applying a fast 2-D-based algorithm. Once the face is detected, a point cloud obtained from the original is created, only with the points corresponding to the face. The algorithm is constantly computing the head pose by comparing the actual point cloud with the first detected. The rotation matrix is obtained and, therefore, the Euler angles (3DoF) corresponding to the rotation between the two clouds (original and currently captured). Finally, the regions of interest for human factors are searched on the actual face, i.e., eyes and nose.

A. Face Detection and Segmentation

The cloud of points retrieved from the Kinect sensor is filtered in order to reduce the points to analyze, thus reducing the search space and the processing cost of further algorithms. With the restrictions of space in the vehicles' cockpit, it is possible to model the environment and1 take restrictions into account, such as the driver's distance to the sensor.

Once the cloud is filtered, a color image is obtained from it by projecting the points to the 2-D world and extracting the information of the three channels [red, green, and blue (RGB)]. As a result, a 2-D representation of the cloud (or color image) is obtained, and the driver's face is searched using the Viola–Jones algorithm [29].

With the face detected, the next step is to build a cloud with the corresponding points of the face. This is done by extracting only the points that correspond to the rectangle that contains the detected face (see Fig. 3). Because both the cloud and the color image have the same dimensions and are calibrated to match their center and borders (the corners of the image match the corners of the cloud), the position of the pixel of the image has an associated point in the cloud, and the cloud of the face is built. The obtained cloud is then cropped by 15% from each side (top and bottom of the rectangle) in order to focus on the space of the head that allows the estimation of head pose. By cropping the rectangle, irrelevant information that may lead to misinterpretations is eliminated, i.e., hair at the top and upper torso or neck at the bottom. The former can affect the cloud due to the inaccuracy of the depth sensor that may capture the points at one moment but ignore them in the next. The latter represent parts of the body that do not rotate with the head; thus, they should be removed for the head pose estimation.

The capturing process has an important role at the beginning of the algorithm, since once a face is detected, the cloud obtained is used as a reference for the comparison with the rest of the next-to-be-found clouds. Thus, calculated rotation will use this original pose as a reference. Cropping is also applied to the reference cloud.

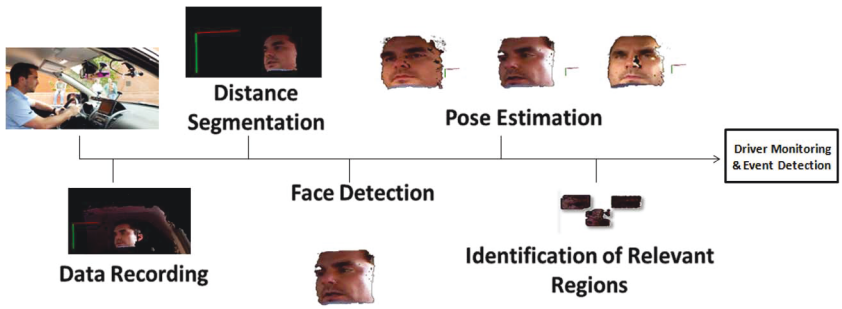

Fig. 2. Schema of the overall system.

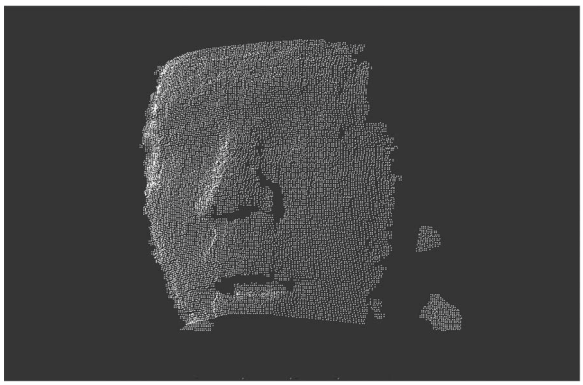

Fig. 3. Three-dimensional segmentation of a detected face without cropping.

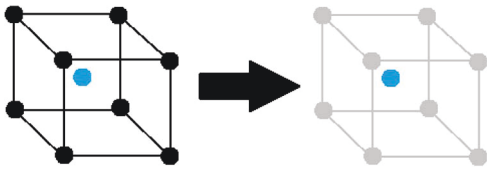

Fig. 4. Example of the voxel filter.

B. Identify the Rotation

Because computational time is paramount, the amount of points is reduced with a voxel filter (see Fig. 4). This will remove the points from the cloud in a determined space and replace them with one point only (its centroid). Filtering is applied to both the reference and the current cloud. It is important to determine the optimal value of the parameter of the voxel filter. If the parameter is too low, then the amount of points will be large, and the computational time reduction will not be appreciated. Too high and too few points will be available for the next algorithms, thus compromising the accuracy of the whole solution.

The point p' of the obtained cloud from the voxel filter is defined by (1), where N is the parameter that determines the size of the region to be considered when replacing it for a single point, i.e.,

$$p'(x', y', z') = \left(\frac{\sum_{i=0}^{N} x_i}{N}, \frac{\sum_{i=0}^{N} y_i}{N}, \frac{\sum_{i=0}^{N} z_i}{N} \right). \tag{1}$$

Once the reference and the current cloud are properly processed, the iterative closest points (ICP) algorithm is applied. This algorithm compares the two clouds and delivers the transformation matrix, which is composed of the rotation matrix and the translation vector. It does this by minimizing the sum of the square error of the distance between points. The steps the algorithm performs are given here.

1) Associate the points according to the nearest neighbor criterion (2). The closest points between the two point clouds are associated in order to allow the later comparison, i.e.,

$$d(p_1, p_2) = \|p_1 - p_2\| \tag{2}$$

where the point of the model cloud is p_1, and p_2 is the point of the cloud to be compared.

2) The transformation parameters rotation and translation are calculated between the two clouds using the minimum square, i.e.,

$$MC = \frac{1}{n} \sum_{i=1}^{n} (p_1 - p_2)^2. \tag{3}$$

Thus, transformation parameters applied to the latest point cloud are obtained for the later comparison.

3) The previous parameters are applied to the cloud to be evaluated, i.e.,

$$Mt = R \left(\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + T \right) \tag{4}$$

where T is defined as the translation vector $\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix}$ for the three axes for the cloud, and R is defined as the rotation matrix obtained from the multiplication of the three rotation matrices for each axis $R1$ (5), $R2$ (6), and $R3$ (7) in Z, Y, and X, respectively, i.e.,

$$R1 = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) & 0 \\ \sin(\Delta\theta) & \cos(\Delta\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$R2 = \begin{bmatrix} \cos(\Delta\delta) & 0 & \sin(\Delta\delta) \\ 0 & 1 & 0 \\ -\sin(\Delta\delta) & 0 & \cos(\Delta\delta) \end{bmatrix} \tag{6}$$

$$R3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\Delta\varphi) & -\sin(\Delta\varphi) \\ 0 & \sin(\Delta\varphi) & \cos(\Delta\varphi) \end{bmatrix}. \tag{7}$$

4) The fitness error is calculated between the target cloud with the parameters applied and the model cloud.

5) Steps 1–4 are repeated until the fitness error in (8) is lower than the minimum error specified or the number of iterations reaches its maximum previously specified, i.e.,

$$f(R, T) = \frac{1}{N} \sum_{i=1}^{N} \|m_i - Mt(t_i)\|^2. \tag{8}$$

In (8), N stands for the number of points of the clouds, and m_i is the point of the model cloud. f represents the fitness error derived from (2)–(4), i.e., the minimum square comparison between the first face model retrieved by the sensor, i.e., m_i, and the last, i.e., t_i. The rotation and translation parameters are represented by the transformation matrix Mt.

The ICP algorithm is applied in order to obtain the transformation matrix Mt determining the new coordinates of the target cloud, as shown in (4), corresponding to the rotation and translation of the original matrix.

From ICP, the matrix Mt (3×3) is obtained. It is from this matrix from where the Euler's rotation angles are obtained according to (9)–(11). The translation vector effect is eliminated by using the centroid of the point of clouds corresponding to the face obtained, i.e.,

$$\Delta\theta = \sin^{-1}(R_{2,1}) \tag{9}$$

$$\Delta\delta = \tan^{-1}\left(\frac{-R_{3,1}}{R_{1,1}}\right) \tag{10}$$

$$\Delta\varphi = \tan^{-1}\left(\frac{-R_{2,3}}{R_{2,2}}\right) \tag{11}$$

where $\Delta\theta$, $\Delta\delta$, and $\Delta\varphi$ are the rotation angles for pitch, roll, and yaw, respectively.

ICP needs to be properly configured in order to deliver reliable results. After a variety of tests, it was observed that the slight variations of the Viola–Jones algorithm when delivering the rectangle containing the face (therefore the cloud of points of the face) significantly altered the results obtained between measurements. In order to avoid these errors, a constant size is determined for the two clouds to be analyzed by ICP.

Finally, as the last part of the algorithm, the eyes and the nose are searched inside the area defined as face, based on physical constraints and Haar-like features. These facial features combined with the use of different techniques such as PERCLOSE analysis or gaze orientation according to the pupils are suitable for further human factors and driver behavior analysis.

IV. Results

In order to determine the reliability of the algorithm, a series of different tests was conducted with a total of 32 people, performing movement in the three axis: lateral movements (left to right), vertical movements (up and down), and roll movements (see Fig. 5). An IMU was attached to the back of the head of the test subjects for ground truth measurements.

The database created for the tests is available in [30]. It is composed of both 2-D and 3-D information from the Kinect sensor, as well as the IMU data for ground truth. The data set is composed of 32 subjects performing movements in all the angles measured (see Fig. 5). Three sets of movements were recorded for each subject: The first set was recorded in a controlled scenario (laboratory indoor) under artificial light, the second set was recorded inside the vehicle, with natural light, and both the first and second sets involved prearranged movements. A final set involved 30 s of driving movements (free movements). The prearranged movements performed by each subject in the first and second sets consisted of three consecutive movements: lateral movement (left–right), vertical (up–down), and roll movement of the head. Every subject repeated these movements three times (i.e., three recordings per set).

Overall results are shown in Table I, where it depicts the mean absolute error (MAE). Figs. 6–8 represent results corresponding to one of the tests performed.

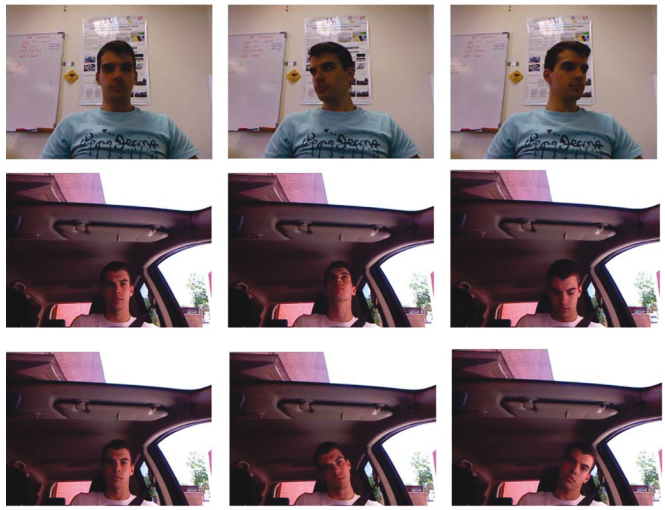

Fig. 5. Database subject example, with laboratory sequence (up) and vehicle sequence (center and down). The three movements are displayed: yaw movement (up), pitch movement (center), and roll (down).

TABLE I
Mean Angle Error in the Overall Test Between the IMU and ICP

Angle	Error
Pitch	2.1°
Yaw	3.7°
Roll	2.9°

Results of the errors per angle for the overall test. Measuring Mean Absolute Error (MAE) as: $\frac{1}{n}\sum_{i=1}^{n}|\text{Angle}_{\text{IMU}} - \text{Angle}_{\text{ICP}}|$

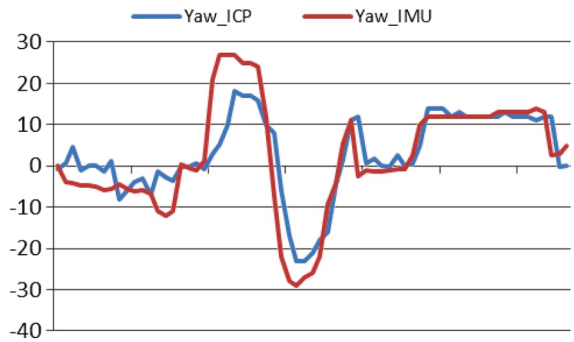

Fig. 6. Comparison of the angles obtained for yaw from ICP and the IMU. The Y-axis indicates the angle or rotation in degrees, and the X-axis indicates the time stamp in seconds.

The system was tested under different lighting conditions (i.e., every subject performed the test movements under natural light conditions during in-vehicle tests and artificial light in the laboratory). Errors due to light conditions were removed from the database.

The performance of the algorithm running on a PC Intel core i7 is up to 10 frames/s. Due to the continuity of the sequences, each movement contained more than 30 frames in order to allow statistical significance of the measurements. The experiments mainly focused (but not only) on yaw and pitch as these are the two main rotation angles that the driver performs. For example, yaw could be used to determine if a driver was distracted or not at a given time. Pitch could be used as a

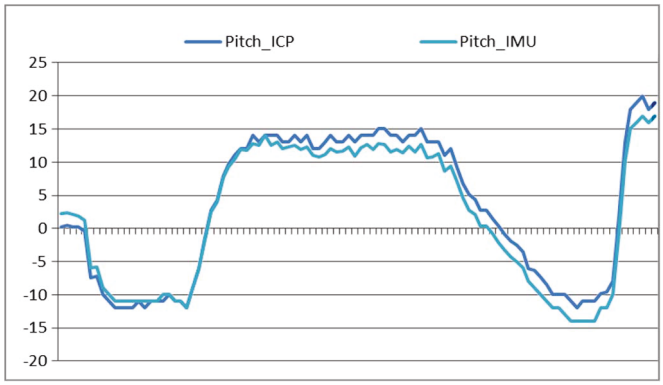

Fig. 7. Comparison of the angles obtained for pitch from ICP and the IMU. The Y-axis indicates the angle or rotation in degrees, and the X-axis indicates the time stamp in seconds.

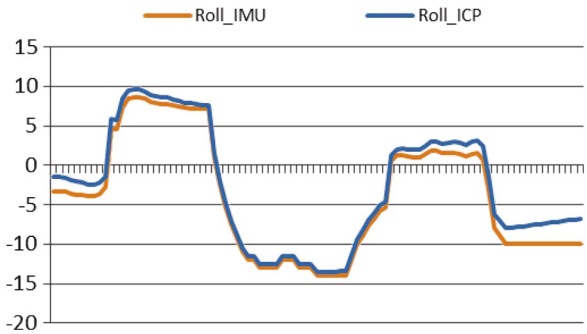

Fig. 8. Comparison of the angles obtained for roll from ICP and the IMU. The Y-axis indicates the angle or rotation in degrees, and the X-axis indicates the time stamp in seconds.

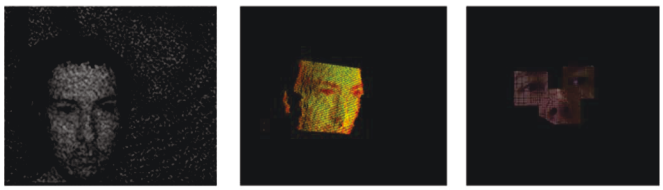

Fig. 9. Application results with the (a) original IR information from the sensor, (b) 3-D face detection, and (c) relevant regions identification (eyes and nose).

part of drowsiness detection, combined with other techniques such as the PERCLOS analysis of the driver (see Fig. 9).

As shown in Figs. 6–8, the system was able to track the movement of the head for its three angles that describe the rotation angles during the movements performed by the test subject with a very limited error.

As shown in Table I and Figs. 6–8, the face pose estimation algorithm was able to determine with success the position of the head, with error rates always better than 4°. Comparison of results provided by [15] shows a compendium of 38 different works with results regarding fine pose estimation. The work presented here provides considerably better results (in terms of MAE) than 33 of them (approximately 87%) and very similar to the other 5. Furthermore, the results shown in this paper are similar to other more recent approaches [8], [9]. Although a comparison of overall results is statistically insignificant, due to the diversity of the databases used, it is proved that the performance of the system, based on a low-cost sensor, is close to other state-of-the-art systems, even with the lack of a tracking stage that would allow smoother error rates. To allow real and statistically significant

Fig. 10. (a) and (b) Examples of normal face detection. (c) and (d) Error due to extreme light conditions.

comparison of results, and to check future improvements on the algorithm, the Intelligent Systems Laboratory [30] provides on its web page the database.

It can also be remarked that the present work represents a step forward in head pose estimation based on low-cost sensors, in relation to previous works, such as those proposed in [27] and [28]. The work in [27] provides fast and trustable face detection with head pose estimation, but the mean errors presented in the angle estimations are far from the results obtained with the present approach. In [28], on the other hand, the least square approach based on pixel-to-pixel comparison is more time-consuming. Furthermore, no face detection is performed, and results are not provided. The present work provides accurate results with low processing time.

During the development of the aforementioned tests, it could be proved that although the Kinect device was designed for analyzing objects of the size of a human body, it delivered good results for head pose estimation. Another advantage obtained from this system is the possibility of the reduction of false positives when detecting faces inside the RGB image. Due to a simple mask applied to the 2-D image, based on the distance discrimination from the depth information, the resulting image would deliver a much reduced search space for the face, therefore eliminating false candidates that have geometrical and chromatic similarities with a face.

V. Conclusion and Future Work

With the described solution, the orientation of the head was determined with the ICP algorithm. This was possible due to the collaboration between the analysis in 2-D from the image and the corresponding 3-D point in the cloud. In addition, the algorithm is able to locate relevant regions of the face, such as eyes and nose. The evidence provided proved the performance of the algorithm with accurate results. All these advances were implemented in a extreme low-cost sensor (Kinect) with a cost of around $100, considerably lower than other 3-D sensors available, which involves expenses of thousands of dollars, such as time-of-flight cameras or stereo vision systems.

It is also important to notice that the technology used showed high performances with a very limited cost, although it presents some limitations that should be mentioned. Since Kinect was designed for indoor applications, its usability for outdoor applications is limited; specifically, it is very sensitive to strong illumination [see Fig. 10(c)

and (d)]. This strong illumination appeared when intense sunlight beams directly hit the face of the driver. However, sunlight sensitivity is a common issue in most of video-based systems. In these specific situations, the depth information is lost, but color camera is still available; thus, it can be used for driver monitoring, as shown in [5]. On the other hand, although this is a limitation of the sensing device to take into account, the system performance was highly reliable under other conditions, i.e., not extremely strong illumination during daylight conditions, nightlight conditions, or indoor. This makes this algorithm suitable to a wide variety of applications, some of which are automatic human factor measurements for vehicles where the system would be able to be included in a test vehicle working under the appropriate weather conditions or in all circumstances inside a driving simulator. Another application where the system performance would be secured is night-fatigue monitoring. Finally, it should be remarked that, although the algorithm was designed and tested on the Kinect sensor, its usability is not limited to this sensor. It can easily be adapted to be used with any point-cloud-based sensor (e.g., time-of-flight cameras and stereo cameras).

Future work lines will focus on the analysis of the eyes, more specifically, the amount of time they are closed in order to determine if the driver is falling asleep. This information together with other important information for driver monition, such as eye gaze, will be combined with the presented head pose estimation to provide a full driver monitoring system.

REFERENCES

[1] C. Papadelis *et al.*, "Monitoring sleepiness with on-board electrophysiological recordings for preventing sleep-deprived traffic accidents," *Clin. Neurophysiol.*, vol. 118, no. 9, pp. 1906–1922, Sep. 2007.

[2] C. Papadelis *et al.*, "Monitoring driver's sleepiness on-board for preventing road accidents," *Stud. Health Technol. Inf.*, vol. 150, pp. 485–489, 2009.

[3] T. Wakita *et al.*, "Driver identification using driving behavior signals," in *Proc. IEEE Intell. Transp. Syst.*, 2005, pp. 396–401.

[4] Y. Takei and Y. Furukawa, "Estimate of driver's fatigue through steering motion," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2005, vol. 2, pp. 1765–1770.

[5] M. J. Flores, J. M. Armingol, and A. Escalera, "Real-time warning system for driver drowsiness detection using visual information," *J. Intell. Robot. Syst.*, vol. 59, no. 2, pp. 103–125, Aug. 2010.

[6] J. Heo and M. Savvides, "Rapid 3D face modeling using a frontal face and a profile face for accurate 2D pose synthesis," in *Proc. IEEE Int. Autom. Face Gesture Recog. Workshop*, 2011, pp. 632–638.

[7] X. Li, E. Seignez, and P. Loonis, "Vision-based estimation of driver drowsiness with ORD model using evidence theory," in *Proc. IEEE IV Symp.*, 2013, pp. 666–671.

[8] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 300–311, Jun. 2010.

[9] R. Oyini Mbouna, S. G. Kong, and M.-G. Chun, "Visual analysis of eye state and head pose for driver alertness monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1462–1469, Sep. 2013.

[10] I. Garcia, S. Bronte, L. M. Bergasa, J. Almazan, and J. Yebes, "Vision-based drowsiness detector for real driving conditions," in *Proc. IEEE IV Symp.*, 2012, pp. 618–623.

[11] M. J. Flores, J. M. Armingol, and A. de la Escalera, "Driver drowsiness warning system using visual information for both diurnal and nocturnal illumination conditions," *EURASIP J. Adv. Signal Process.*, vol. 2010, p. 3, 2010.

[12] H. Eren, U. Celik, and M. Poyraz, "Stereo vision and statistical based behaviour prediction of driver," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 657–662.

[13] Seeingmachines, [Accessed: 01-Feb-2014]. [Online]. Available: http://www.seeingmachines.com/

[14] Smarteye, [Accessed: 01-Feb-2014]. [Online]. Available: http://www.smarteye.se/

[15] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation in computer vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 607–626, Apr. 2009.

[16] J. Sherrah, S. Gong, and E. J. Ong, "Face distributions in similarity space under varying head pose," *Image Vis. Comput.*, vol. 19, no. 12, pp. 807–819, Oct. 2001.

[17] H. A. Rowley, S. Baluja, and T. Kanade, "Rotation invariant neural network-based face detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. (Cat. No.98CB36231)*, 1998, pp. 38–44.

[18] R. Stiefelhagen, J. Yang, and A. Waibel, "Modeling focus of attention for meeting indexing based on multiple cues," *IEEE Trans. Neural Netw.*, vol. 13, no. 4, pp. 928–938, Jul. 2002.

[19] M. Voit, K. Nickel, and R. Stiefelhagen, "Head pose estimation in single- and multi-view environments—Results on the CLEAR'07 benchmarks," in *Multimodal Technologies for Perception of Humans*, vol. 4625, R. Stiefelhagen, R. Bowers, and J. Fiscus, Eds. Berlin, Germany: Springer-Verlag, 2008, pp. 307–316, SE - 29.

[20] J. Wu and M. Trivedi, "A two-stage head pose estimation framework and evaluation," *Pattern Recog.*, vol. 41, no. 3, pp. 1138–1158, Mar. 2008.

[21] S. Yan *et al.*, "Learning a person-independent representation for precise 3D pose estimation," in *Proc. CLEAR*, 2007, pp. 297–306.

[22] J. X. J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2D + 3D active appearance models," in *Proc. IEEE Comput. Soc. Conf. CVPR*, 2004, vol. 2, pp. II-535–II-542.

[23] J.-G. Wang and E. Sung, "EM enhancement of 3D head pose estimated by point at infinity," *Image Vis. Comput.*, vol. 25, no. 12, pp. 1864–1874, Dec. 2007.

[24] G. Zhao, L. Chen, J. Song, and G. Chen, "Large head movement tracking using SIFT-based registration," in *Proc. ACM Int. Conf. Multimedia*, 2007, pp. 807–810.

[25] E. Murphy-Chutorian and M. M. Trivedi, "HyHOPE: Hybrid head orientation and position estimation for vision-based driver head tracking," in *Proc. IEEE Intell. Veh. Symp.*, 2008, pp. 512–8517.

[26] S. O. Ba and J. M. Odobez, "A probabilistic framework for joint head tracking and pose estimation," in *Proc. 17th ICPR*, 2004, vol. 4, pp. 264–267.

[27] G. Fanelli, T. Weise, J. Gall, and L. Van Gool, "Real time head pose estimation from consumer depth cameras," in *Proc. Int. Conf. Pattern Recog.*, 2011, pp. 101–110.

[28] F. A. Kondori, S. Yousefi, H. Li, S. Sonning, and S. Sonning, "3D head pose estimation using the Kinect," in *Proc. Int. Conf. Wireless Commun. Signal Process.*, 2011, pp. 1–4.

[29] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.

[30] Intelligent Systems Laboratory, [Accessed: 01-Jun-2014]. [Online]. Available: http://www.uc3m.es/islab