EURASIP Journal on Image and Video Processing
a SpringerOpen Journal

RESEARCH Open Access

Infrared-based facial points tracking and action units detection in context of car driving simulator

Parisa Darvish Zadeh Varcheie[1,2*], Claude Chapdelaine[1] and Langis Gagnon[1]

Abstract

Facial expressions (FE) are one of the important cognitive load markers in the context of car driving. Any muscular activity can be coded as an action unit (AU) which are the building blocks of FE. Precise facial point tracking is crucial since it is a necessary step for AU detection. Here, we present our progress in FE analysis based on AU detection on face infrared videos in the context of a car driving simulator. First, we propose a real-time facial points tracking method (HCPF-AAM) using a modified particle filter (PF) based on Harris corner samples which is optimized and combined with an Active Appearance Model (AAM) approach. Robustness of PF, precision of Harris corner-based samples, and optimization of AAM result in a powerful facial points tracking on very low-contrast images acquired under near-infrared (NIR) illumination. Second, detection of the most common AUs in the context of car driving, identified by a certified Facial Action Coding System coder is presented. For detection of each specified AU, the spatio-temporal analysis of related tracked facial points is performed. Then, a combination of rule-based scheme with Probabilistic Actively Learned Support Vector Machines is developed to classify the features calculated from the related tracked facial points. Results show that with such a scheme, we can obtain more than 91% of precision in the detection of the five most common AUs for low-contrast NIR images and 90% of precision in the MMI dataset.

Introduction

The goal of the SPEED-Q [1] and COBVIS-D projects [2] is to develop a simulation environment for driver retraining, It is composed of a multi-sensor data acquisition and analysis system for driving performance assessment and cognitive load measurements. The persons are asked to drive in a simulator and then react to the monitored scenarios (Figure 1a). Their cognitive load varies according to the complexity level of the driving task (Figure 1b).

Facial expressions (FE) are one of the important cognitive load markers in the context of car driving. FE can be characterized globally or locally in terms of the whole facial attitude using the Facial Action Coding System (FACS) [3]. FACS is based on muscular activity underlying momentary changes in facial aspects where each change can be coded as a facial Action Unit (AU). AUs are the building blocks of any FE.

A certified FACS coder has manually analyzed 90 video sequences of 30 persons, acquired in the driving simulator, in order to identify the set of most frequent AUs depicted by car drivers. A total of 140 instances were identified composed of eye blinks, brow lowerer, jaw drops, lips apart, lip corner puller, and lip suck. We previously implemented a real-time eye blink detector that has been integrated in the car simulator [4]. Here, we present our progresses regarding real-time facial points tracking and AU detection on the facial images of the simulator, that are, very low-contrast frontal face images acquired under near-infrared (NIR) illumination (Figure 2).

Tracking of facial points has many applications in pattern recognition, such as FE analysis, face recognition, speech recognition, behavior analysis, etc. AUs' detection enables us to analyze FE for emotion, mood, deception, and attitude recognition.

The rest of the article is organized as follows: in the following section related studies are reviewed. The two

*Correspondence: parisa.darvish@crim.ca
[1]Vision and Imaging Team, Computer Research Institute of Montréal (CRIM), Montréal, QC, Canada
[2]Group of Research in Image Processing, Genetec Inc., Saint-Laurent, QC, Canada

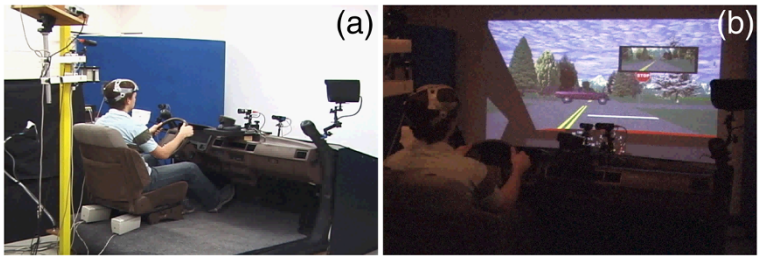

Figure 1 Simulation environment for driver retraining. (a) Driving simulator, **(b)** driver is asked to drive inside the simulator and respond to a one hour driving scenario displayed on the screen.

sections after the following section present our methodologies about facial point tracking and AU detection analysis, respectively. Finally, performance results and conclusions are presented at the end of the article.

Related studies

During the last decade many vision-based driving-assistance systems have been proposed for road safety improvement. Some studies have focused on pedestrian detection and tracking [5] and some others on drivers FE analysis. For example, Murphy-Chutorian and Trivedi [6] proposed a head pose detection systems for monitoring driver awareness application using Haar-wavelet Adaboost cascades, SVM classifiers, and appearance-based 3D particle filter (PF) tracking. Smith et al. [7] proposed a driver visual attention system using one color camera to detect eyeblinking, eye closures, and large mouth movement. Their system was color-based which is not appropriate during night time. A real-time system for monitoring driver vigilance using infrared images has been proposed in [8].

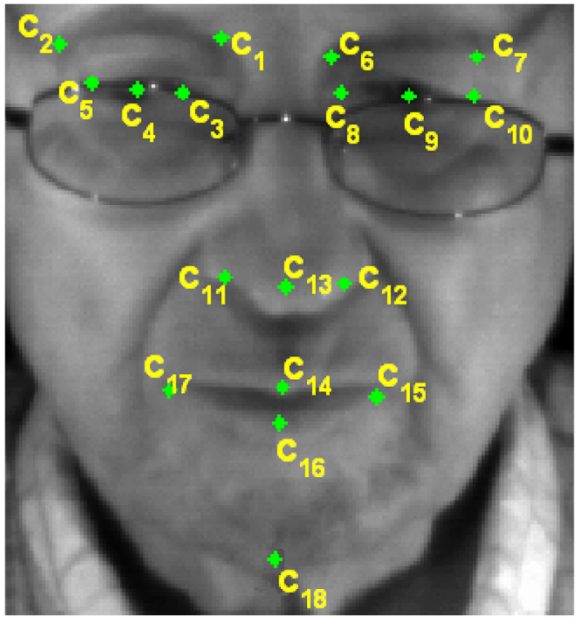

Figure 2 Labeled facial points used for tracking and AU detection analysis.

Classical PF has been introduced initially in [9]. After that, many types of modified PF tracking methods have been introduced for different kinds of object tracking, specifically facial point tracking.

Facial points tracking is a crucial preprocessing step to do driver FE analysis based on AU detection in face videos. In [10], facial point tracking has been performed using PF with factorized likelihoods (PFFL). PFFL is an extension of classical PF that uses color-based observation model that combines a rigid and morphological model. This method is dependent on color features without taking into account shape and texture to track facial points. In [11], active appearance model (AAM) tracking is proposed to model face using its texture and shape. They use a principal component analysis (PCA) scheme to build various facial models. Then, they compare different face models constructed from face shape and face texture with the initial face model using an optimization function to find the best match with its initial face model. AAM tracking has a good precision for facial points tracking but fails in the presence of occlusion or fast movements. Therefore, PF tracking in conjunction with AAM tracking (PFAAM) has been introduced in [12] that combines the robustness of PF with the precision of AAM. In this method, state vector composed of shape and texture of AAM face model (eight parameters) along with a likelihood measure and AAM search are used to compare sampled face models with target face model. Finally, AAM optimization is used to find the best match. Fleck et al. [13] have modified the PFAAM model by adding two different dynamic models to deal with occlusions and a local optimization step.

In our approach, we combine PF and AAM tracking methods differently where PF has a larger role than AAM. The main differences between our tracking method, called PF with Harris corner samples and AAM optimization (HCPF-AAM), and the others are listed below.

1. The proposed HCPF uses Harris corners for the PF sample set which provides selective samples with strong features. Therefore, the PCA analysis to build all facial changes is not used here.
2. Each facial corner is tracked individually and independently of other facial corners using HCPF for each facial point. Thus, the state vector in our HCPF is composed of only facial point coordinates which has less complexity and processing time than the eight parameters used in [12,13]. Independent facial point tracking provides better robustness in the case of partial occlusion and head movements.
3. AAM is used only in the optimization step to verify the combination of all the best samples for all facial points together. In our AAM model, we have decreased the number of facial points to 18.

There are approaches to detect AUs in static face images [14]; however, approaches on video analysis prove to have some advantages. FE results obtained from video analysis have higher confidence level than results from static face images. Indeed, neutral faces might contain some AUs that can be discriminated only by video analysis. For example, some upper neutral faces might appear to frown because of available wrinkles between two eyes.

Some approaches for AU detection from video analysis have been proposed. Tian et al. [15] proposed a neural network (NN) to recognize different AUs based on edge features, face wrinkles, and shape. Besides the complexity of tuning different parameters in NN-based approaches (e.g., number of layers, coefficients, etc.), their method is not applicable on our dataset because of low-contrast images where wrinkles or edges are not significantly visible. Using NN-based approaches Bartlett et al. [16] obtained a 91% average recognition rate and Tian et al. [15] 87.9%. Cohn et al. [17] used discriminant function analysis and obtained a 85% average recognition rate. Valstar et al. [18] used a probabilistic actively learned support vector machines (PAL-SVM) method to detect AUs in video sequences. In this method, for each AU an SVM classifier based on some features (distances based on facial points) is trained and a 85% average recognition rate was obtained. The main differences between our AU detection method compared to the method proposed in [18]:

1. In our method, PAL-SVM classifier is applied only on frames where facial changes related to an AU are detected by an apex/antapex detection scheme. This scheme depends on the variations of particular facial points distances corresponding to an AU.
2. The features in the PAL-SVM classifier have been modified based on analysis and rules of related muscles movement for each AU identified by a FACS coder.
3. We have added a rule-based method to PAL-SVM classifier results based on the AUs FACS definition.

The detail of our HCPF-AAM is explained in the following section.

Facial points tracking

We have proposed an adaptive PF method based on a Harris corner sampling tracker suited to our facial points tracking problem. Harris corner is used to extract feature point candidates. It is combined with a PF that is robust to non-Gaussian facial points distribution, person's head movements, and random motions. Target modeling and tracking are done based on samplings made around the predicted positions obtained by the PF and feature points are extracted by the Harris corner detector. The scoring of the sample features is done through normalization functions that are used to combine different measure values and to standardize their magnitudes within similar ranges. These normalization functions are applied to geometric and appearance features.

Figure 3 shows the whole facial points tracking and AU detection process. In this section, we concentrate on the building blocks of the system architecture which correspond to our real-time facial points tracking method. The other building blocks in Figure 3 for AU detection method will be described in the next section.

To do FE analysis, face elements are modeled with 18 anchor points called facial points according to the Face and Gesture Recognition Working group annotation standard [19] (Figure 2). Accurate facial point tracking is an essential preprocessing step for AU detection. Facial points tracking is performed by applying first a modified PF tracking method over regions containing strong features like corners (using a Harris corner detector [20]) and then optimizing the results with an AAM tracker. Indeed, PF is a robust tracking method but precise target locations might vary due to the image noise if sampling is done correctly. AAM is a precise tracking method but fails for fast movements or occlusions. Videos in our SPEED-Q dataset were acquired under NIR illumination with very low contrast and because of that, using PF or AAM alone fails in precise facial points tracking. Thus, AAM tracking optimization is applied to increase tracking precision. During the PF tracking, each facial point is tracked independently and optimized by the AAM tracking part, building the facial points all together into a face model.

PF tracking

We developed a PF tracker adapted to our NIR facial points tracking problem based on Harris corner detection. Some normalization functions are applied on the

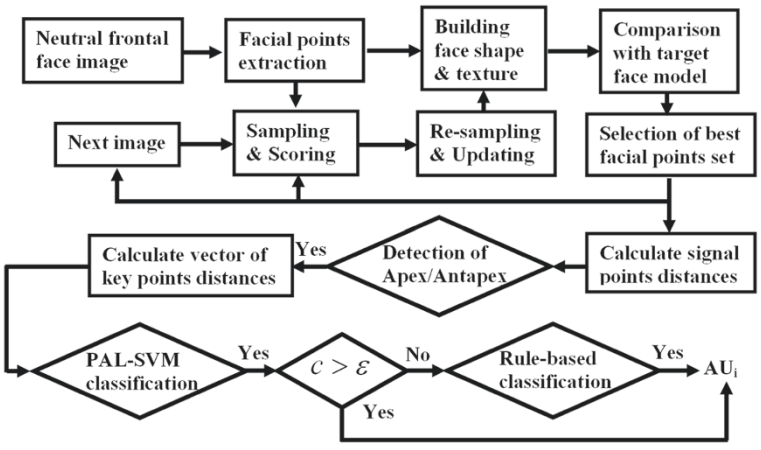

Figure 3 Facial points tracking and AU detection process, *c*: confidence level of PAL-SVM classification result, ϵ: threshold used for verification of PAL-SVM classification.

samples scoring and on geometric and appearance features. They are used to combine different measure values to normalize their magnitudes. Normalization of the sample scores might be done by any function, but analytical normalization functions allow tuning of the score values.

In PF, target (facial point) modeling and tracking are done based on sampling selected around the corners with strong features.

To have continuous FE analysis, the tracking algorithm must not lose the facial points throughout the video and must not be distracted by head movements. The PF is a Bayesian method that recursively estimates the state of the tracking target as a posterior distribution with a finite set of weighted samples. It operates using prediction and update phases. A sample is a prediction based on the state of tracking target. We find PF as an appropriate solution because of its robustness due to the facial points random and/or regular motions. We would need to find all possible candidates resulting either from all types of facial points motions or from head movements to find the best match candidate. On the other hand, the tracking should be cost effective to be used in real-time applications.

Some questions can be raised such as how many samples are required to cover all possible target positions? Where the samples should be located? How the samples should be distributed, locally, globally, or randomly? How much the processing time of the PF will be? The answer to these questions explains the main difference of our modified PF method against classical PF method. Classical PF alone would have not been appropriate, because we should have many samples to cover all possibilities everywhere in the image. Having of many samples increases the processing time and since the object tracking should be cost effective, the samples should be generated appropriately and selectively. In our modified PF based on Harris corner samples, PF samples are generated where there is only a strong corner. The head motion vector is removed by optical flow using a radial histogram scheme that is explained in the following. Then, our HCPF uses AAM tracking method to optimize the tracking results.

Observation model

A target is represented by a rectangle $\alpha \times \beta$ (e.g., 30×30) around each facial point. The state of the PF at each time t is defined as a vector $\vec{P}_t$ of the rectangle center coordinates as:

$$\vec{P}_t = (x(t), y(t)). \tag{1}$$

The initial state vector $\vec{P}_t$ containing the facial points coordinates is determined when the initial target face model is detected by the AAM tracker.

Sample scoring using normalization functions

To localize each facial point f, features of the ith sample s_{fi} are compared with the initial facial point model M_f, and a weight or a score ω_{fi} is given to each s_{fi} using a set of normalization functions. The following appearance and geometric-based measures are used for each facial point in our HCPF tracking method:

1. $\phi_p(s_f)$, the Euclidean distance between the sample coordinates and the corresponding previous position of the facial point, is the first measure done on a geometric feature:

$$\phi_p(s_f) = \sqrt{(x_c(s_f) - x_P)^2 + (y_c(s_f) - y_P)^2} \tag{2}$$

where $(x_c(s_f), y_c(s_f))$ and (x_P, y_P) are the sample center coordinates and corresponding previous facial

point center coordinates, respectively. This distance is normalized by a Gaussian function $\bar{\phi}_p(s_f)$ given by:

$$\bar{\phi}_p(s_f) = e^{\left(-\frac{\phi_p(s_f)}{2\sigma^2}\right)} \tag{3}$$

where σ has been determined experimentally to 10 (for image size 640×480).

2. $\phi_g(s_f)$, the 2D correlation coefficient between gray level values of the sample template image I_{s_f}, and the gray level values of corresponding previous position of the facial point template image I_L, is used as the measure for this appearance feature.

$$\phi_g(s_f) = \frac{\sum_\alpha \sum_\beta (I_{s_f}(x,y)) - \bar{I}_{s_f})(I_L(x,y) - \bar{I}_L)}{\sqrt{\sum_\alpha \sum_\beta (I_{s_f}(x,y) - \bar{I}_{s_f})^2 \sum_\alpha \sum_\beta (I_L(x,y) - \bar{I}_L)^2}} \tag{4}$$

where $\bar{I}_{s_f}$ and $\bar{I}_L$ denote the average value of I_{s_f} and I_L. $\bar{\phi}_g(s_f)$ is the considered normalization function and applied to $\phi_g(s_f)$ as :

$$\bar{\phi}_g(s_f) = \frac{\phi_g(s_f) + 1}{2} \tag{5}$$

3. $\phi_h(s_f)$, the Euclidean distance between the normalized gray level histogram of sample H_{s_f} and the normalized gray-level histogram of the previous position of the facial point sample H_L, is the measure for this appearance feature as:

$$\phi_h(s_f) = \sqrt{\sum_n (H_{s_f}[n] - H_L[n])^2} \tag{6}$$

where n is the histogram bin number. The normalization function $\bar{\phi}_h(s_f)$ is considered as:

$$\bar{\phi}_h(s_f) = 1 - \frac{\phi_h(s_f)}{\sqrt{2}} \tag{7}$$

4. $\phi_e(s_f)$, the 2D correlation coefficient between sample edge image, and the corresponding previous position of the facial point edge image, is used as the measure for this appearance feature and is similar to $\phi_g(s_f)$. $\bar{\phi}_e(s)$ is the considered normalization function similar to $\bar{\phi}_g(s)$ and applied to $\phi_e(s)$.
5. $\phi_{px}(s_f)$ and $\phi_{py}(s_f)$, the Euclidean distances between the normalized x and y projection histograms of the sample edge image pattern and the normalized x and y projection histograms of the corresponding previous position of the facial point edge image are measures used for x and y histogram projections, respectively; the same as $\phi_h(s_f)$. Their corresponding normalization functions $\bar{\phi}_{px}(s_f)$ and $\bar{\phi}_{py}(s_f)$ are similar to $\bar{\phi}_h(s_f)$.

The weight (score) of the sample s_{fi} at time t, ω_{fi}^t, is obtained by the sum of all normalization function values as:

$$\omega_{fi}{}^t = \bar{\phi}_p(s_{fi}{}^t) + \bar{\phi}_g(s_{fi}{}^t) + \bar{\phi}_h(s_{fi}{}^t) + \bar{\phi}_e(s_{fi}{}^t) + \bar{\phi}_{px}(s_{fi}{}^t) + \bar{\phi}_{py}(s_{fi}{}^t) \tag{8}$$

For each facial point target, s_f is the best sample at time t which has the maximum weights and is selected by:

$$s_f = \text{argmax}_{s_{fi} \in S_f} \{\bar{\phi}_p(s_{fi}) + \bar{\phi}_g(s_{fi}) + \bar{\phi}_h(s_{fi}) + \bar{\phi}_e(s_{fi}) + \bar{\phi}_{px}(s_{fi}) + \bar{\phi}_{py}(s_{fi})\} \tag{9}$$

Resampling and updating

Among all N_{fT} samples existing in each frame for each facial point, the $N_{s_{fi}}$ samples with the highest probabilities (weights) are selected. Thus, the current sample set S_{ft} is composed of $N_{s_{fi}}$ samples centered on (x_{fi}^t, y_{fi}^t) with probability ω_{fi}^t at time t, where (x_{fi}^t, y_{fi}^t) is the i^{th} sample coordinates of facial point f at time t and ω_{fi}^t is the related score of the ith sample. The sample set S_{ft} is an approximation of posteriori distribution of the target state (facial point f state) at time t. In our application, we observe that the PF state between two consecutive frames does not change significantly so only translation of sample coordinates around previous position of each facial point and around detected corners is taken into account. No rotation or scaling is applied because we assume frontal face view tracking. At each time t, the motion of the facial point is assumed to correspond to a dynamical first-order auto-regressive model given by:

$$P_t = P_{t-1} + \omega_t. \tag{10}$$

where P_t and P_{t-1} are the PF states at time t and $t-1$, respectively. ω_t is a multivariate Gaussian random variable and it correlates to random translation of the sample center coordinates. Thus, in the resampling step, N samples are generated by a Gaussian random function in a circular region of radius r_g around the centroid of the facial point region of interest (ROI). Indeed, r_g will be modified in each frame accordingly with the inverse of error E in Equation (12).

In our HCPF, current sample set S_{ft} is composed of two sample sets, S_{ft}^{target} and S_{ft}^{Harris}:

$$S_{ft} = S_{ft}^{\text{target}} + S_{ft}^{\text{Harris}} \tag{11}$$

S_{ft}^{target} is the current sample set that is composed of previous sample set at time $t-1$ and S_{ft}^{Harris} is the current sample set composed of strong corners. Using only previous sample set is not appropriate in our application since

it requires the accurate location of facial points as strong corners inside the face skin region. Thus, re-sampling is done based on two types of samples: samples around previous target position and Harris corners samples. If the person turns his head, the person motion vector is extracted by a pyramidal Lucas-Kanade optical flow [21] and the tracking process with HCPF is stopped. Optical flow extracts motion-based pixels with their related motion vectors. To determine where the person turns his head, a radial histogram of motion vectors is calculated. Each histogram bin is composed of the quantized length (r) and angle (θ) of the motion vector. The r and θ coordinates of the bin that has the maximum number of vectors are assigned to the head motion vector length and angle. Tracking process is re-initialized after the person turns back his head to frontal view.

Optimization with the AAM tracker

The AAMs [11] is a deformable template model that provides high precision on the facial points localization. It contains an iterative optimization such as Gauss-Newton that searches along the gradient direction for an improved parameter vector. AAM uses appearance (texture **g**) and geometry (mesh or shape **s**) features that are learned from face examples to fit the model to still images. AAM tracker looks for all possible changes of face models using a PCA analysis.

In our approach, there are two trackers: the first is the global tracker that tracks face using all facial points together in AAM tracker; the second is the partial tracker that tracks each facial point individually and provide facial point information for the global tracker. The way these two trackers communicate and are combined is done by replacing the PCA analysis step of the AAM tracker with the partial tracker.

We replace PCA analysis with best PF samples given by our HCPF to determine the optimal face model. In fact, possible face models are composed of samples with high score of each facial point. Face shapes and textures are extracted from given samples to determine the possible face models. Then the AAM tracker compares the M face models to find the best set of facial points, F_s, from given samples that best match with target face model. AAM tracker uses an optimization function based on Lorentzian norm to compare composed textures from possible facial points candidates as:

$$F_s = \text{argmin}_{i \in M} \left\{ \log \left(1 + \frac{E_i}{2\sigma^2} \right) \right\} \tag{12}$$

where E_i is the quadratic norm and is defined as:

$$E_i = (\mathbf{g}_{\text{model}} - \mathbf{g}_i)^2 \tag{13}$$

Therefore, F_s is selected as the best match for target facial points and would be used in the next section for AU detection. Lorentzian norm is used because it has robustness to outliers [22].

AU detection

To detect and recognize AUs, we have used a combination of rule-based scheme and PAL-SVM method. One or more facial muscles contraction causes changes in facial feature points positions and generates an AU of the FACS system. Each AU is encoded by analyzing the spatio-temporal relations between the tracked points.

From the FACS rules [3], we have identified for each AU a particular subset of facial points called key points and analyzed the spatio-temporal distances between them. We are using normalized Euclidean distance P_{12} between two key points P_1 and P_2 as the feature representing the changes in position of the fiducial facial points. For the selection of the key facial points, three important facts should be considered. First, the points should be related to the muscles changes where the contraction is happening. Second, the points should be discriminative enough to specify a particular AU from others. Third, sufficient number of distances should be measured and analyzed to accurately detect the related AU. In our analysis, we normalize P_{12} by dividing the current Euclidean distance by its reference value when the face is in neutral state. This distance normalization is performed to avoid head motions effect, scaling, and rotation changes from our calculation. In addition, the nose central point is used as the reference point and subtracted from all facial points to register all video frames within a sequence.

In FE analysis, AUs detection can be used either for emotion detection [23] or for cognitive loads assessment [24]. As explained before, in SPEED-Q and COBVIS-D projects, cognitive overloaded should be detected for driver retraining. We have selected several AUs for cognitive load assessment based on the work of King [24]. He described the cognitive FE without referring to their AU numbers which we had to infer. Instead, he classified expressions into upper, lower, and whole face groups. From [24] and an analysis of our SPEED-Q dataset by the FACS coder, we identified the following relevant AUs for our work: brow lowerer (AU4), jaw drops (AU26), lips apart (AU25), lip corner puller (AU12), and lips suck (AU28). Also, from [25] eye blinks (AU45) is stated as an AU highly relevant for cognitive load assessment. We had previously implemented a real-time eye blink detector integrated in the car simulator [4]. Table 1 shows the FACS rules used for recognition of the specified AUs. In this article, we concentrate on detection of mouth-related AUs and brow lowerer AU.

A feature vector $v = < d_{11}, d_{12}, \ldots, d_{1t}, d_{21}, \ldots, d_{tn} >$ with $t \times n$ elements is assigned to each AU. n varies according to the number of facial point distances used for each AU and d_{ij} is the specified facial point distance i at time j.

Table 1 FACS rules [3] for recognition of most common AUs occurring in the car driving simulator

AU	Definition
4	Lowers the eyebrow downward the eye
12	Pulls left and right lip corners obliquely upward (⌣ shape)
25	Part the lips
26	Drops the jaw close to the jaw relax
28	Pull lips into mouth or suck lips into mouth

t is the duration when an AU appears until it completely disappears. This vector is an input for a PAL-SVM classifier [18]. For each AU, a particular PAL-SVM classifier is used. The classification result is the presence (or not) of that particular AU with a confidence level parameter c in Figure 3. This confidence level determines the certainty of the classification result.

An AU occurs alone or in combination with other; in addition, it occurs for various durations. Thus, it is necessary that the vector be normalized in time. We have proposed an apex/antapex scheme to identify when a possible AU starts and ends. This scheme is a crucial step to determine the time duration (the length) of vector v with its start and end-points. To do so, we analyze the Euclidean distances between various signal points. Signal points are the facial points for which their Euclidean distance values change during an AU onset, apex and offset to almost a ⌣ or ⌢ shape such as in Figure 4. We analyze the curves of the signal points distance to find where and when local maximums or minimums are occurring inside a sliding window of length w_s as shown in Figure 4. Indeed, in each frame, the signal point distance curves are simultaneously swept with a window size w_s to find apex or antapex. Then, from the apex point, we start sliding backward and forward (left and right sides of the curve shown in Figure 4, respectively) to search in the previous and following frames for a neutral points distance value (i.e., equal to 1 since it is a normalized distance) with a variance of $\pm\delta$. w_l and w_r are the duration where an AU starts from its neutral value and ends to the apex/antapex and then starts from apex/antapex and ends to its neutral position, respectively. $t = w_l + w_r$ is the AU duration. Using this scheme, we can find the moment where an AU starts and ends as well as its duration. Then, the vector v is built by cropping a part of all key points distance curves of an AU from the estimated AU starting point to its ending point. This solution is appropriate when an AU occurs alone but not when an AU occurs in combination of others. In this later case, the distance curve might not touch the neutral value line and the vector v might have an unlimited time length. Thus, if w_l or w_r is greater than $w_s/2$, we limit the vector v time length to w_s (e.g., w_l and w_r is equal to $w_s/2$ each).

As discussed above, the vector v is built by cropping a part of all key points distance curves and is classified by PAL-SVM classifier with a confidence level c. We applied a rule-based method following the PAL-SVM classification if the confidence level c is below a threshold τ as detailed in Table 2. These thresholds are obtained experimentally.

Finally, our rule-based method is a combination of a set of AND/OR rules and different thresholds that are applied on the measured Euclidean distances between the key points as shown in the third column of Table 2.

Results and discussion

We have tested the tracking and AU detection algorithms on two types of dataset: the public MMI dataset [26] and our SPEED-Q dataset. The SPEED-Q dataset is composed of uncompressed video sequences of 30 subjects sitting in the driving simulator (90 video sequences, 3 video sequences per subject, with an average of 25 min/sequence at 30 fps). Figure 5 shows the proportion of relevant AUs found in the representative sample of our SPEED-Q dataset by a certified FACS coder. A total of 140 AUs were identified in this sample where the most common upper face AUs, out of 38, were eye blink

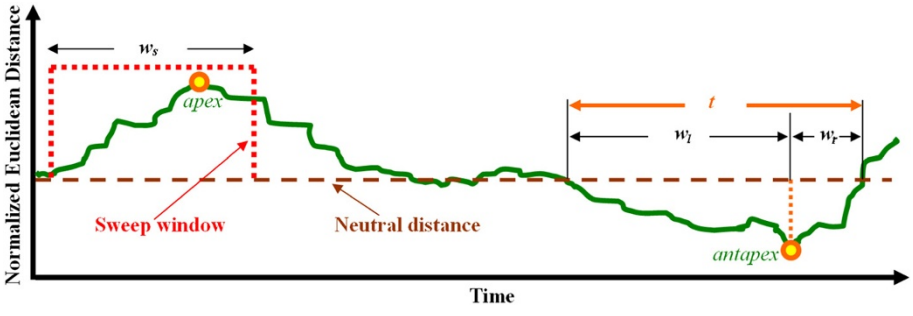

Figure 4 Sliding of the normalized Euclidean distance curve to detect apex/antapex locations (see text for details). The antapex term here is used to express the difference between the apexes in two opposite directions. The neutral point corresponds to an absence of AU. The apex is reached from the neutral expression when it's increasing and the antapex correspond to a Euclidien distance less than the neutral distance.

Table 2 Key points distance, signal points distances with thresholds used on distances of rule-based scheme for recognition of most common AUs occurring in the car driving simulator

AU	KPD	SPD	RBD
4	$DC_{2,7}, DC_{1,6}, DC_{1,3}, DC_{6,8}$	$DC_{1,6}, DC_{6,8}, DC_{1,3}$	$DC_{1,3} \leq \tau_7$ OR $DC_{6,8} \leq \tau_7$
12	$DC_{13,15}, DC_{13,17}, DC_{15,17},$	$DC_{15,17}$	$\tau_5 \leq DC_{15,17}$ AND $\tau_6 \leq DC_{14,16}$
25	$DC_{13,18}, DC_{14,16}, DC_{15,17}$	$DC_{14,16}, DC_{13,18}$	$\tau_2 \leq DC_{14,16} \leq \tau_1$
26	$DC_{13,18}, DC_{14,16}$	$DC_{14,16}, DC_{13,18}$	$\tau_3 \leq DC_{13,18}$
28	$DC_{14,16}, DC_{15,17}$	$DC_{13,18}, DC_{15,17}, DC_{14,16}, DC_{13,16}, DC_{14,18}$	$DC_{14,16} \leq \tau_4$

KDP, key points distances used in PAL-SVM classifier; SPD, signal points distances used in our apex/antapex detection scheme; RBD, thresholds and distances used in rule-based method; $DC_{i,j}$, normalized Euclidean distance between two facial points of C_i and C_j shown in Figure 2; τ_i, different thresholds used for particular distances in rule-based method.

(AU45) and brow lowerer (AU4). Also, the most common lower face AUs, out of 102, were jaw drop (AU26), lips apart (AU25), lip corner puller (AU12), and lip suck (AU28), respectively.

The five AUs we concentrate on are the AUs know to occur in a cognitive overload situation and the ones relevant to the task of car driving. We also based our selection on the most frequent AUs observable in our SPEED-Q video dataset. They were identified after a careful manual video analysis of the car driver faces by a certified FACS coder (C. Chapdelaine). The other existing AUs related to cognitive overload (e.g., head motion) were not statistically present enough in our dataset to be taken into account in this study. We present the result of our tracking method and AU detection method separately. The whole algorithm has been implemented on GPU to run in real time.

Facial points tracking performance

Our tracking method is a combination of HCPF and AAM techniques. In this section, the result of the proposed tracking method is compared with simple PF alone and AAM tracking. The training step is an offline process for the AAM tracking, done for the face model. We experimentally find that AAM model would be more robust and accurate if it is trained for each dataset individually. To train AAM tracking for the MMI dataset, we have selected 100 MMI video sequences and used only 6 images per video (the duration of MMI video sequences is shorter than for the SPEED-Q dataset). We have tested our HCPF-AAM tracker on 500 MMI video sequences. Similarly for the SPEED-Q dataset, we have selected 15 videos and used 70 images per each. We have used the rest of the SPEED-Q video sequences for test. We evaluate the tracking algorithm on those video parts where all facial points are visible and the face is frontal.

To specify the localization accuracy, circles with radius of 10, 20, and 50% of the nose length around each facial point are used (Figure 6) for two facial points (left and right lip corners). Nose length is assumed to be equal to the distance between the central nose point and middle point of the two interior eye corners on a neutral face. We evaluate the facial points tracking algorithm performance

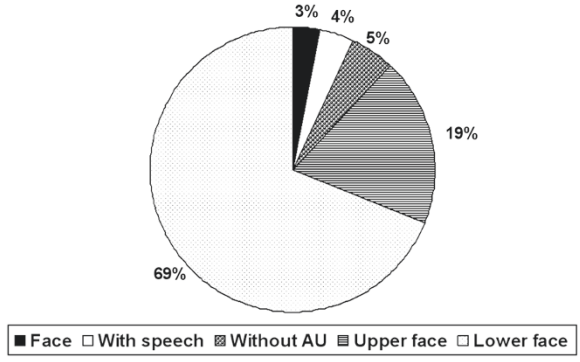

Figure 5 Founded AUs in the sample of SPEED-Q dataset.

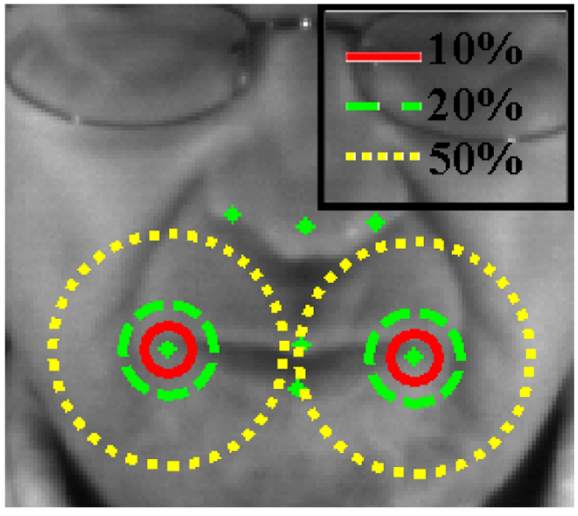

Figure 6 Circular zones around lip corners for measuring facial points tracking accuracy.

on the SPEED-Q video sequences where either all types of AUs are occurring or no AUs have been detected. We have used two metrics:

1. Precision (P) to calculate the facial point localization accuracy inside the particular circular ROI. It is defined as:
$$P = \frac{TP}{TP + FP} \tag{14}$$
where TP is the number of frames where the facial point is detected correctly inside a particular circular ROI. FP is the number of frames where the facial point is detected wrongly inside a particular circular ROI.
2. Track fragmentation (TF) is a measure of the lack of continuity of the tracking algorithm [27] and is defined as:
$$TF = \frac{F_{\text{out}} + FP}{N} \tag{15}$$
where F_{out} is the number of frames where the target is detected out of a particular circular ROI and N is the total number of frames. Therefore, TF shows the lost of facial point tracking either by false detection or by facial point being out of the ROI.

Tables 3 and 4 show the comparison of the proposed tracking algorithm performance with simple PF [9] and AAM tracking [11] methods in different circular ROI for our SPEED-Q dataset and MMI dataset [26]. P indicates the location distribution of the tracked facial points. Tables 3 and 4 show that P values for the proposed method is higher than for the PF and AAM tracking method. It means that the correct facial point detection rate of our method is higher than the two other methods. Also, by increasing the ROI size the facial points tracking precision decreases. This is shown by the precision Equation (12) where the TP and FP are counted only in frames where the facial point is detected inside a particular ROI. Therefore, TP has a fixed value for all ROIs and FP increases with larger ROIs which decreases the precision accordingly. The decreasing rate for P is less than for the AAM and PF methods. It means that most of the facial point candidates in our method are located near the true facial point and thus with the increase of ROI size, the FP does not increased very much. This is not the case for the AAM and PF methods. TF values for the proposed method is less than the two other methods. Indeed, our method has less continuity during facial points tracking. TF values are fixed by increasing of ROI size according to its definition in Equation (13). $F_{\text{out}} + FP$ is constant for all ROI circular regions because the number F_{out} of facial point candidates in smaller ROI contribute as FP in larger ROI size (i.e., as FP increases, F_{out} decreases but the sum remains constant). N has also no changes with increase of ROI size. N is equal to the sum of FP, F_{out} and TP, and TP is fixed in all ROI sizes thus N is also fixed.

Results of Table 3 and 4 show that our method outperforms the simple PF and AAM trackers. This can be explained by appropriate combination of the PF tracker and AAM face model. In our method, facial points are better localized using PF since strong corner candidates are added to each facial point. Moreover, each facial point is tracked individually but also groups of facial points are organized and optimized to find the best face model similar to target face model using AAM tracker. Furthermore, results of Tables 3 and 4 illustrate that some facial points (e.g., C_{14}, C_{15}, C_{16}, and C_{17}) are generally better tracked than some others (e.g., C_1, C_2, C_6, and C_7). This can be explained by the facial point features and texture around them which, for some of them, are more robust and discriminant. Results in Table 4 are better than that of Table 3 because of the nature of video data (low contrast images with NIR illumination in the SPEED-Q dataset rather than color optical images in the MMI dataset).

AU detection performance

A PAL-SVM classifier is required to be trained for each AU based on the particular key points distances used for each of them. We have only used a part of the MMI dataset for training the PAL-SVM classifiers and tested it on both MMI and SPEED-Q datasets. The number of videos used for training is AU4:80, AU12:70, AU25:100, AU26:100, AU28:20. The program has been tested on cropped videos from the SPEED-Q dataset where all types of facial AUs are present. To evaluate the algorithm performance, two metrics have been used which are listed below:

1. True Positive Rate (TPR) or sensitivity which is defined as:
$$TPR = \frac{TP}{TP + FN} \tag{16}$$
where TP is the number of times an AU is correctly detected in the video sequences. FN is the number of times a video sequence contains an AU but the AU is not detected.
2. False Positive Rate (FPR) which is defined as
$$FPR = \frac{FP}{FP + TN} \tag{17}$$
where FP is the number of times an AU is wrongly detected in a video sequence that do not contain that AU. TN is the number of times the detection of an AU is rejected in video sequences that do not contain an AU. The TN and FN values are the complement of TP and FP values, respectively. In the ideal case, TP and TN have the maximum values and FP and

Table 3 Precision and tracking fragmentation of the different facial points tracking methods for the SPEED-Q dataset

Rad	MC	TM	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}	C_{18}	$\overline{MC}$
		PF	49	51	71	70	72	48	52	72	70	69	62	65	63	67	69	63	66	67	*63*
10%	P%	AAM	53	52	74	72	70	55	54	67	69	71	65	66	63	71	68	64	67	70	*65*
		HCPF-AAM	75	76	87	88	84	74	72	89	86	83	82	84	83	93	95	92	96	89	**84**
		PF	42	41	65	66	63	39	41	62	64	61	53	56	52	58	53	59	57	64	*55*
20%	P%	AAM	45	46	68	69	67	43	44	66	68	65	57	53	52	64	66	62	65	65	*59*
		HCPF-AAM	72	70	86	84	84	71	70	85	82	80	81	82	80	91	92	92	93	87	**82**
		PF	36	37	60	62	59	35	36	61	59	60	48	50	46	52	50	49	51	57	*50*
50%	P%	AAM	40	39	62	66	62	37	38	61	58	55	52	51	50	59	58	52	55	60	*53*
		HCPF-AAM	67	69	85	84	77	66	64	84	80	78	78	79	77	87	88	86	89	85	**79**
		PF	27	26	22	24	23	28	27	23	25	22	25	24	24	22	23	21	20	21	*24*
–	TF%	AAM	25	27	21	20	19	24	22	19	20	23	22	23	21	20	22	17	19	20	*21*
		HCPF-AAM	14	12	8	9	9	13	11	8	7	9	10	12	11	8	6	7	6	6	**9**

Rad, radius circle size; TM, tracking method; MC, metric; $\overline{MC}$, average of metric values for all facial points; C_1 to C_{18}, facial points shown in Figure 2.

Table 4 Precision and tracking fragmentation of the different facial points tracking methods for the MMI dataset

Rad	MC	TM	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}	C_{18}	$\overline{MC}$
		PF	57	59	65	63	64	61	58	62	53	54	57	54	58	66	69	71	70	68	*61*
10%	P%	AAM	59	60	80	79	77	62	63	60	71	75	72	74	70	69	68	75	77	74	*70*
		HCPF-AAM	81	82	92	93	91	83	80	90	89	91	87	88	91	93	95	96	95	93	**89**
		PF	52	50	54	53	56	49	51	55	56	54	52	53	50	59	57	57	58	60	*54*
20%	P%	AAM	55	57	73	75	74	59	56	58	67	68	66	69	63	62	65	70	69	71	*65*
		HCPF-AAM	77	76	84	88	89	81	78	88	85	84	83	85	88	90	89	92	91	89	**85**
		PF	47	48	51	50	52	49	50	51	54	52	47	49	48	52	54	51	50	57	*50*
50%	P%	AAM	52	50	68	69	71	53	51	55	64	62	63	66	59	58	60	65	64	66	*60*
		HCPF-AAM	75	74	79	80	78	79	76	84	83	81	80	82	84	89	88	90	87	86	**81**
		PF	24	23	21	19	19	20	22	25	25	20	21	22	23	19	20	16	18	19	*20*
–	TF%	AAM	23	24	19	18	18	22	21	19	19	20	21	19	18	19	18	16	19	17	*19*
		HCPF-AAM	12	10	7	7	8	11	10	7	7	8	9	10	10	7	5	5	4	5	**8**

Rad, radius circle size; TM, tracking method; MC, metric; $\overline{MC}$, average of metric values for all facial points; C_1 to C_{18}, facial points shown in Figure 2.

Table 5 Various metric values of particular AUs for the SPEED-Q and MMI datasets

Dataset	MC	AU_{25}	AU_{26}	AU_{28}	AU_{12}	AU_4	$\overline{MC}$
SPEED-Q	*TP*	77	79	52	95	13	63
	FP	12	14	7	9	4	9
	TN	81	76	92	95	43	77
	FN	6	5	12	4	6	6
	TPR%	92	94	81	95	68	91
	FPR%	12	15	7	8	7	10
MMI	*TP*	89	92	23	82	28	63
	FP	9	7	4	6	3	6
	TN	76	64	53	82	36	62
	FN	6	5	5	8	10	7
	TPR%	93	94	82	91	73	90
	FPR%	10	9	7	6	7	8

MC, metric; $\overline{MC}$, average of metric values for all facial points.

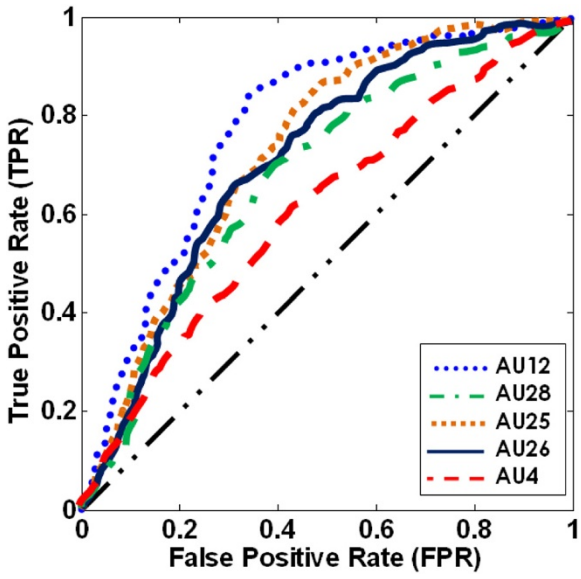

Figure 8 ROC curves of different AUs detection for the SPEED-Q dataset.

FN are zero. Since there is no ideal method with zero *FPR* and zero false negative rates, our algorithm has both false positive and false negative.

Table 5 shows the AU detection results of the proposed method for five specific AUs in the SPEED-Q dataset and MMI datasets. Also, Figure 7 illustrates the face neutral mode for each AU used in this study. Ideally, each AU should be detected with $TPR = 100\%$ and $FPR = 0$. However $TPR \neq 100\%$ because of some false detections. Besides, the AU recognition error causes by the PAL-SVM and rule-based classifications, most of the false alarms result from facial points tracking error and/or incomplete occurrence of an AU when it is combined with others. Facial points tracking error causes wrong key points distances and has direct effect on the false AUs detection process. AU occurs differently for each person with various intensities and time durations. In comparison, AUs are better detected in the MMI dataset since less *FPR* is obtained with almost equal *TPR*. This is because the MMI dataset has pure AUs with high quality color images. In the SPEED-Q dataset, some facial points such as eyebrows interior and exterior corners are less visible since images have low contrast. In addition, the corners for thick eyebrows are difficult to be correctly localized since eyebrows have uniform texture. It causes false tracking of facial points and therefore false alarms in AUs detection. This fact can be confirmed with ROC curves of the different AUs detection obtained for each SPEED-Q video sequence (Figure 8) and by results in Table 5 showing that the detection rate for AU4 is less than other AUs.

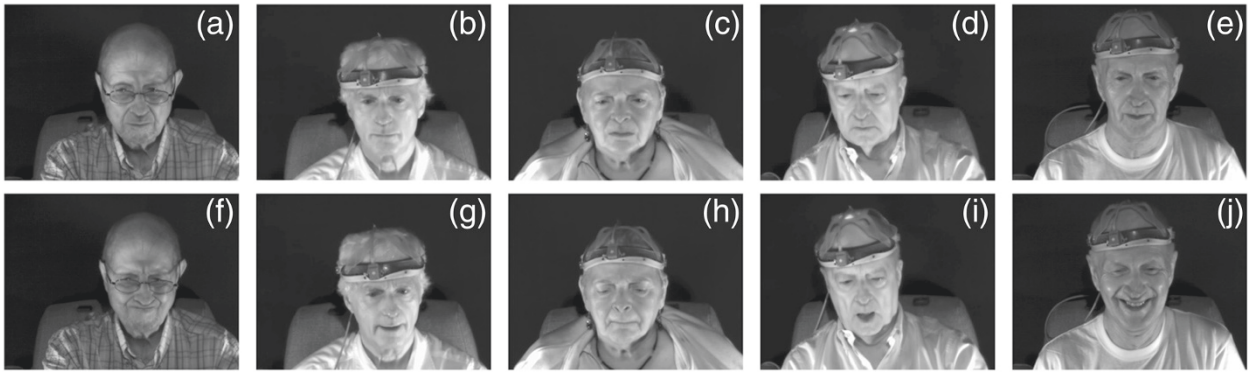

Figure 7 Various face modes versus neutral mode. (a–e) Neutral face images in SPEED-Q dataset; most common AUs detected in SPEED-Q driving simulator by our algorithm: **(f)** AU4 and AU28, **(g)** AU25, **(h)** AU28, **(i)** AU26, **(j)** AU12.

Conclusion

We presented a study on FE analysis based on AU detection of NIR videos in the context of a car driving simulator. We proposed a real-time facial points tracking method (HCPF-AAM) and a PAL-SVM rule-based AU detection technique. HCPF-AAM uses a modified PF tracking method based on Harris corner samples which is optimized and combined with an AAM technique. AAM is an accurate tracking method but fails in the case of fast movement or occlusions while PF can handle them. Results showed that PF when applied on Harris corner based samples and optimized with AAM, provide a powerful facial points tracking on very low contrast images with high precision and low tracking fragmentation. Detection of the most relevant AUs in the driving simulator context was done by a spatio-temporal analysis of related tracked facial points. A combination of rule-based scheme with PAL-SVM was developed to classify the features calculated from the related tracked facial points. Results assessed by a certified FACS coder have shows that such a scheme leads to more than 91% of precision for the detection of the five most common AUs relevant to the driving task for the SPEED-Q simulator and 90% of precision in the MMI dataset. Future work will consist of extending detection of additional AUs to be combined with those detected AUs in order to build a higher-level FE semantic analysis module.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This study was supported in part by the Canadian AUTO21 center of excellence research network (http://www.auto21.ca). We also thank our colleagues from Laval University, Prof. D Laurendeau and Prof. Teasdale, leaders of the SPEED-Q project. Also T Moszkowicz and M Lavallière who were responsible of acquiring multi-sensor data and software integration. We also thank our CRIM colleagues S Foucher and S Marchand for scientific help and GPU implementation.

Received: 15 December 2011 Accepted: 11 July 2012
Published: 18 September 2012

References

1. Auto21: Safe platform for evaluating/enhancing driver qualifications. http://www.auto21.ca/en/subcontent.php?page=ae2105. Accessed 20 November 2010
2. S Beauchemin, PDZ Varcheie, L Gagnon, D Laurendeau, M Lavalliére, T Moszkowicz, F Prel, N Teasdale, COBVIS-D: a computer vision system for describing the Cephalo-Ocular behavior of drivers in a driving simulator, Image Anal. and Recognit., Lecture Notes Comput. Sci. **5627**, 604–615 (2009)
3. P Ekman, W Friesen, *Facial Action Coding System: A Technique for The Measurement of Facial Movement* (Consulting Psychologists Press, Palo Alto, 1978)
4. M Lalonde, D Byrns, L Gagnon, N Teasdale, D Laurendeau, in *Fourth Canadian Conference on Computer and Robot Vision (CRV)* Real-time eye blink detection with GPU-based SIFT tracking (Montreal, QC, Canada, 2007), pp. 481–487
5. J Ge, Y Luo, G Tei, Real-time pedestrian detection and tracking at nighttime for driver-assistance systems, IEEE Trans. Intell. Transp. Syst. **10**, 283–298 (2009)
6. E Murphy-Chutorian, M Trivedi, Head pose estimation and augmented reality tracking: an integrated system and evaluation for monitoring driver awareness, IEEE Trans. Intell. Transp. Syst. **11**(2), 300–311 (2010)
7. P Smith, M Shah, N da Vitoria Lobo, Determining driver visual attention with one camera, IEEE Trans. Intell. Transp. Syst. **4**(4), 205–218 (2003)
8. L Bergasa, J Nuevo, M Sotelo, R Barea, M Lopez, Real-time system for monitoring driver vigilance, IEEE Trans. Intell. Transp. Syst. **7**, 63–77 (2006)
9. M Isard, A Blake, CONDENSATION—conditional density propagation for visual tracking, Int. J. Comput. Vis. **29**, 5–28 (1998)
10. I Patras, M Pantic, in *Proceedings of Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004* Particle filtering with factorized likelihoods for tracking facial features (2004), pp. 97–102
11. T Cootes, G Edwards, C Taylor, Active appearance models, IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 681–685 (2001)
12. S Hamlaoui, F Davoine, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005 (ICASSP'05)* Facial action tracking using an AAM-based condensation approach (Philadelphia, PA, USA, 2005)
13. S Fleck, M Hoffmann, K Hunter, A Schilling, in *Fourth Canadian Computer and Robot Vision, 2007 (CRV'07)* PFAAM: an active appearance model based particle filter for both robust and precise tracking (Montreal, QC, Canada, 2007), pp. 339–346
14. M Pantic, L Rothkrantz, Facial action recognition for facial expression analysis from static face images, IEEE Trans. Syst. Man Cybern. B: Cybernetics. **34**(3), 1449–1461 (2004)
15. YI Tian, T Kanade, J Cohn, Recognizing action units for facial expression analysis, IEEE Trans. Pattern Anal. Mach. Intell. **23**(2), 97–115 (2001)
16. MS Bartlett, JC Hager, P Ekman, TJ Sejnowski, Measuring facial expressions by computer image analysis, Psychophysiolgy. **36**, 253–263 (1999)
17. JF Cohn, AJ Zlochower, J Lien, T Kanade, AF Analysis, Automated face analysis by feature point tracking has high concurrent validity with manual FACS coding, Psychophysiolgy. **36**, 35–43 (1999)
18. M Valstar, I Patras, M Pantic, in *IEEE Workshops on Computer Vision and Pattern Recognition (CVPR)* Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data (San Diego, CA, USA, 2005), pp. 76–84
19. FGNET: Face and Gesture Recognition Working group. http://personalpages.manchester.ac.uk/staff/timothy.f.cootes/data/bioid_points.html. Acessed 19 November 2010
20. C Harris, M Stephens, A combined corner and edge detector. 147–152 (1988)
21. BD Lucas, T Kanade, in *Proceedings of Imaging Understanding Workshop* An iterative image registration technique with an application to stereo vision (1981), pp. 121–130
22. MB Stegmann, Active appearance models: theory, extensions and cases. Master's thesis, Informatics and Mathematical Modeling, Technical University of Denmark, Lyngby, 2000
23. A Bajpai, K Chadha, Real-time facial emotion detection using support vector machines, Int. J. Adv. Comput. Sci. Appl. **1**(2), 137–140 (2010)
24. W King, in *IEEE International Workshop on Robot and Human Communication* Reflections of thought: cognitive facial expressions in the human interface (Tsukuba, Ibaraki, Japan, 1996), pp. 195–200
25. M Recarte, E Pérez, A Conchillo, L Nunes, Mental workload and visual impairment: differences between pupil, blink, and subjective rating, Spanish J. Psychol. **11**(2), 374–385 (2008)
26. MF Valstar, M Pantic, MMI facial expression database. http://www.mmifacedb.com/. Accessed 19 November 2010
27. F Yin, D Makris, S Velastin, in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)* Performance evaluation of object tracking algorithms (Rio de Janeiro, Brazil, 2007)

doi:10.1186/1687-5281-2012-15
Cite this article as: Darvish Zadeh Varcheie *et al.*: **Infrared-based facial points tracking and action units detection in context of car driving simulator.** *EURASIP Journal on Image and Video Processing* 2012 **2012**:15.