Feature Extraction and Selection for Inferring User Engagement in an HCI Environment

Stylianos Asteriadis, Kostas Karpouzis, and Stefanos Kollias

National Technical University of Athens,
School of Electrical and Computer Engineering,
Image, Video and Multimedia Systems Laboratory,
GR-157 80 Zographou, Greece
stiast@image.ece.ntua.gr, {kkarpou,stefanos}@cs.ntua.gr

Abstract. In this paper, we present our work towards estimating the engagement of a person to the displayed information of a computer monitor. Deciding whether a user is attentive or not, and frustrated or not, helps adapting the displayed information of a computer in special environments, such as e-learning. The aim of the current work is the development of a method that can work user-independently, without necessitating special lighting conditions and with only requirements in terms of hardware, a computer and a web-camera.

Keywords: User engagement, Head Pose, Eye Gaze, Facial Feature tracking.

1 Introduction

While a lot of work has been done regarding the issue of user attention estimation in multi-user environments, such as meetings [7], very few articles have been published for estimating attention (engagement) in an HCI environment [6]. In this work, we propose a method for inferring user attention based on the extraction of features deriving from facial analysis. Such systems have mainly been proposed for estimating drivers' attention. For example, in [8], a monocular system is used and color precicates are employed to detect and track the driver's facial features. Facial geometry and eye closure are calculated and driver's attention is extracted with the use of three finite state automata. In [3], the authors use solely eyes closure to determine whether a driver is attentive or not. To this aim, they use a gaussian model to describe driver's attention and, eye closure for certain amounts of time would denote inattention or fatigue. In [10], the position in space of each facial feature is detected using a stereo vision system. Based on these positions, a least square algorithm estimates the head orientation and further analysis follows for the detection of the eye gaze. The combination of the above gives a good estimate of the direction at which a user is looking, however, it requires initialization. In our work, we combine head pose with eye gaze, as well as other biometrics, in a monocular environment, not necessitating any particular lighting conditions or calibration. There is not much work in bibliography combining the two features in an unconstrained environment. Typical work is the one

J.A. Jacko (Ed.): Human-Computer Interaction, Part I, HCII 2009, LNCS 5610, pp. 22–29, 2009.

reported in [12], where facial symmetry along with Gabor filters are used for estimating head pose and eye gaze respectively. A look-up table is then built for corresponding the resulting eye gaze and head pose with the final focus of attention estimate.

Here, we propose a work that can be summarized as follows: Face detection [11], followed by facial feature detection is the first step of our method, while tracking follows. Based on facial features' motion, a series of biometric measurements are extracted and their appropriateness is evaluated for inferring the level of frustration or attentiveness of a user in a Human-Computer-Interaction scenario. Our algorithm is able to recover and re-initialize in cases of occlusion or tracking failure.

2 Facial Points Detection and Tracking

The method reported in [1] is used for face and eye centre and mouth corner (here, enhanced with upper and lower lip points) localization. For the detection of the eye corners (left, right, upper and lower eyelids) a technique similar to that described in [13] is used. In the current work, the point between the nostrils and two points on each eyebrow have also been used, as will be discussed later. For nostrils detection, an area around a segment of the perpendicular to the inter-ocular line was extended starting from the middle of the eyes. The darkest row of this area is considered as the vertical position of the nostrils and the middle point of this row is further used for our experiments. In a similar manner, two points on each eyebrow are extracted, as the darkest points in a neighborhood above the eye corners. The above are illustrated in Fig. 1, where the luminance values of two search areas have been projected on the vertical axis. The minimum of the projections corresponds to the features in search. Tracking is done using a three-Pyramid Lukas-Kanade algorithm. Geometrical face models, and prototypes of natural human motion are employed for recovering from erroneous tracking (see subsection 3.1).

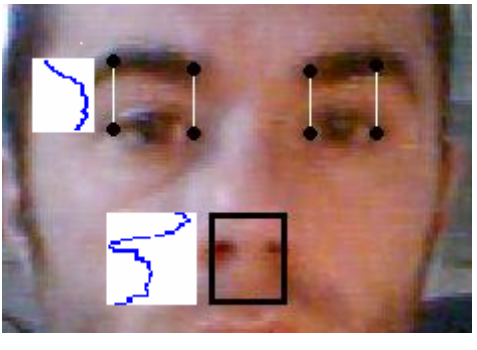

Fig. 1. Eyebrow and nose detection search regions

3 Feature Extraction

The features extracted in our method are the following: Head pose, eye gaze, eyebrow movements, head horizontal and vertical speed components, mouth horizontal and vertical opening, relative movements of the user back and forwards.

3.1 Head Pose Estimation

Head rotation is calculated by examining the translation of the point in the middle of the inter-ocular line with regards to its position when the user was rotated frontally (see Fig. 2), thus providing the Head Pose Vector $\overline{p} = [p_x \ \ p_y]$, where p_x and p_y are the horizontal and vertical components of the eye middle point's translation, respectively, normalized with the inter-ocular distance, as calculated at start-up, to cater for scale variations. The fraction of the inter-ocular distance with the vertical distance between the eyes and the mouth is monitored, and if it is restricted within certain limits with regards to its value at a frontal position, no rotation is decided. As tracking, many times, fails, thus giving false estimates of head pose (as well as the other features), a series of rules were integrated: After large rotations, some features are occluded and cannot be further recovered. In this case, when the user comes back to a frontal position, after n_{t1} frames, the pose length reduces in length but is above a certain threshold, as one of the eyes is not well tracked and the eye center is not at the same neighborhood as at start-up. In this case, the algorithm can re-initialize. The above can be modeled as in equations (1),(2):

$$\left\|\overline{p}(n)\right\| < thr_1 \cdot \left\|\overline{p}(n-n_{t1})\right\| \tag{1}$$

$$\mathrm{var}\left(\overline{p}_{n-n_{t2}:n}\right) < thr_2 \tag{2}$$

where $\|*\|$ denotes a vector length metric. Equations (1)-(2) are interpreted as follows: If the Head Pose Vector length at the current frame n is smaller than a fraction thr_1 of its value at frame n-n_{t1}, and its variance for the last n_{t2} frames (with $n_{t2}<n_{t1}$) is

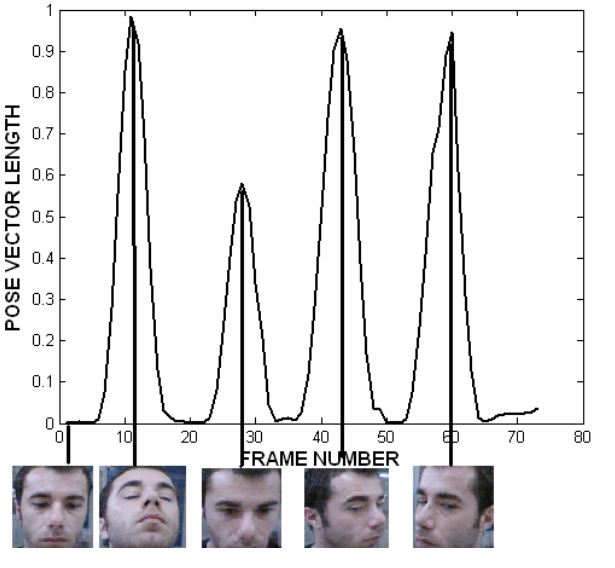

Fig. 2. Pose changes during a video of a person in front of a monitor

smaller than thr_2, the algorithm re-initializes. In our experiments, we used n_{t1}=10, n_{t2}=7, thr_1=0.7, thr_2=0.05. If the above conditions are met, but the user has not turned frontally, face detection fails and, thus, frontal rotation is not decided. In general conditions, however, the algorithm re-initializes by re-detecting the face, facial features and re-starting to track. Further constraints that are taken into account are related with the displacement of features in subsequent frames. By assuming an orthographic projection in the interval between two subsequent frames, it is expected that features are shifted in a uniform way. Finding such outliers and re-calculating the mean shift with the rest of the features helps positioning erroneous points to the position that would agree with the rest of the features' shift. As experiments showed, the above refinement is achieved after 7-10 iterations per frame.

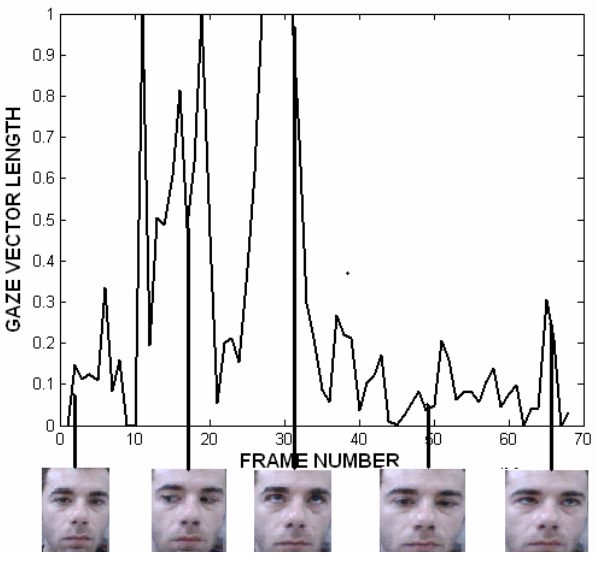

Fig. 3. Gaze changes during a video of a person not moving his head in front of a monitor

3.2 Eye Gaze Estimation

Eye gaze is extracted by monitoring the eye centre movements with regards to a coordinate system defined by the positions of the eye corners and eyelids at each frame (see Fig. 3). The resulting displacement provides with the Eye Gaze Vector $\overline{g} = [g_x\ g_y]$, where g_x and g_y is the horizontal and vertical component respectively.

3.3 Extraction of Further Features

The vertical movements of the eyebrows with regards to the upper eyelids are also extracted, and the horizontal and vertical components of the speed (in pixels per frame) of the head movements are calculated. Furthermore, mouth opening is calculated, with reference to the initial distance between the mouth corners and the lips distance at start-up. Finally, the inter-ocular distance changes are monitored, and

calculated as fractions of the eye centres' distance with regards to the first frame of each initialization of the system. In this way, when changes in inter-ocular distance are not due to head rotations, qualitative measurements of user movement back and forth are achieved.

4 Feature Selection

The experiments were conducted on a database consisting of children with learning difficulties, between the age of 8 and 10. The recorded videos were 720×576 pixels and the frame-rate 25fps. A total of about 10000 and 12250 frames were used for the case of attention/non-attention and frustration/non-frustration problems respectively. The videos were annotated by experts. One of the difficulties of the dataset was that the positive instances (frustration, non-attentiveness) were very few in comparison to the negative ones, and this limited the training session prototypes. In order to evaluate the appropriateness of each of the above features, the Fisher's exact test [4] was used. To this aim, the 3-bin histogram of each of the features was calculated and the resulting distribution for positive instances was compared against the distribution of the same feature throughout all videos, regardless of the state. In our case, we chose Fisher's exact test and not any other method (e.g. chi-square method) because it is ideal for small scale data. Indeed, in many occasions (for example, horizontal speed of the head when the user is frustrated), there are only a few instances where there are low or high values at the correspondent histogram bins. Fisher's exact test is ideal in cases of such small samples. For the event of non-attentiveness, among all features, tests showed that for the features of Head Pose, Eye Gaze, Inter-Ocular Distance Changes and Head Speed, the null hypothesis (that observed and expected distributions do not differ) should be rejected with higher confidence than for the rest of the features. For the event of frustration, Head Pose, Horizontal and Vertical Head speed and Eye Gaze do not follow the null hypothesis as much as the rest of the features do, as it was expected. Figure 4 justifies the rejection of some of the features due to high *p*-values, while Figures 5 and 6 illustrate examples of data for each class.

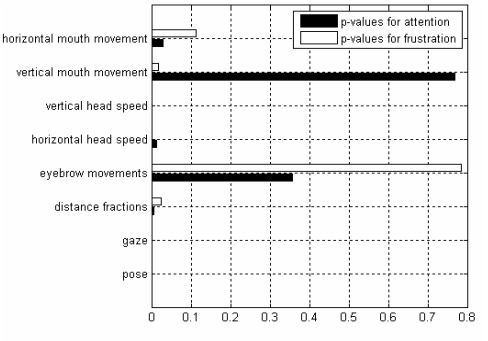

Fig. 4. *p*-values for feature selection in attention/non-attention and frustration/non-frustration scenarios

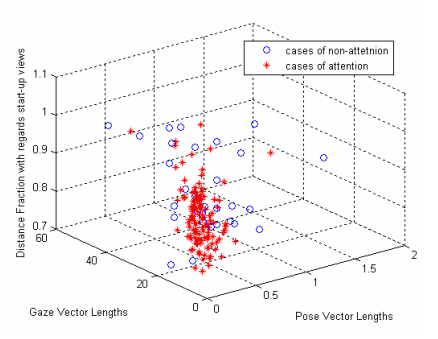

Fig. 5. Features used for attention/non-attention classification

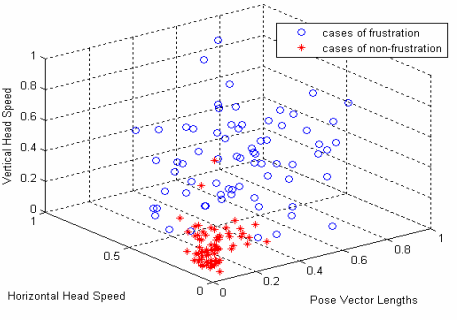

Fig. 6. Features used for frustration/non-frustration classification

5 Experimental Results

For testing the accuracy of our system, a Sugeno-type fuzzy [9] inference system was built for each case. The idea behind using fuzzy systems has to do with the fact that behavioral states do not necessarily belong to certain classes but, rather, they are fuzzy concepts. For example, frustration or distraction can be given confidence values, and outputs of fuzzy systems are ideal in this case. Prior to training, our data were clustered using the sub-cluster algorithm described in [2]. This algorithm, instead of using a grid partition of the data, clusters them and, thus, leads to fuzzy systems deprived of the curse of dimensionality. The number of clusters created by the algorithm determines the optimum number of the fuzzy rules. After defining the fuzzy inference system architecture, its parameters (membership function centers and widths), were acquired by applying a least squares and back-propagation gradient descent method [5]. Tables 1 and 2 summarize the results of the overall accuracy of our system in estimating the behavior of a user in the cases attention/non-attention and frustration/non-frustration experiments using different sets of features with low *p*-value as inputs and 1 or 0 the target states (attention/non-attention or frustration/non-frustration). Testing was done by adopting a leave-one-out protocol.

From tables 1 and 2, it can be seen that, in the case of frustration, although eye gaze has low *p*-value (see Fig. 3), excluding it from experiments does not deteriorate the results but they are marginally higher. This is due to the fact that, in cases of frustration, in our dataset, eye gaze vector length was strongly correlated with head pose vector length. Similarly, although Head Speed (horizontal and vertical) has low *p*-values in the attention tests, results have shown that, excluding these parameters from our decision systems, would improve the results. More careful observation of our data and the corresponding annotation gave the following explanation: Head speed is only large at the beginning of those time segments when a person is turning his/her head away from the camera. At those time segments, head pose vector has small values but head speed is high. However, such movements can also be met during attention time-stamps, as it is very frequent for a reader to make small rapid movements without changing his head pose a lot. For the above reason, it was decided to exclude head speed from our experiments in the case of attention estimation.

The database we used was acquired under normal lighting conditions, with very challenging subjects: Children with learning difficulties. Testing our system on such a dataset is challenging, not only because of its nature, but also due to the fact that annotation is subjective. However, the results obtained are extremely promising.

Table 1. Neuro-Fuzzy System decision accuracy for two different sets of low *p*-value features for detecting User Attention

Features	Overall success rates
Head Pose, Eye gaze, Distance changes, Head speed	84.00%
Head Pose, Eye gaze, Distance changes	88.00%

Table 2. Neuro-Fuzzy System decision accuracy for two different sets of low *p*-value features for detecting User Frustration

Features	Overall success rates
Head Pose, Horizontal and Vertical Head speed	82.00%
Head Pose, Horizontal and Vertical Head speed, Eye Gaze	80.63%

6 Conclusions and Future Work

We presented a method for the automatic estimation of the behavior of a person in front of an HCI environment. Our system is un-intrusive, thus, leaving space for spontaneous behavior, it does not depend on controlled conditions in terms of lighting, and this constitutes it ideal for different settings. Furthermore, since the system does not require any a-priory knowledge of the user or the camera, it does not need any kind of training or calibration beforehand. Future extensions of our work shall include the creation of a common framework for discriminating among a set of states simultaneously. To this aim, we will build a database suitable for our research and work on developing a facial feature tracker, highly specialized for such applications.

Acknowledgments

This work has been funded by the FP6 IP CALLAS (Conveying Affectiveness in Leading-edge Living Adaptive Systems), Contract number IST-34800 and by the IST Project 'FEELIX', (under contract FP6 IST-045169).

References

1. Asteriadis, S., Nikolaidis, N., Pitas, I., Pardàs, M.: Detection of facial characteristics based on edge information. In: 2nd International Conference on Computer Vision Theory and Applications (VISAPP), Barcelona, Spain, pp. 247–252 (2007)
2. Chiu, S.L.: Fuzzy Model Identification Based on Cluster Estimation. Journal of Intelligent and Fuzzy Systems 2(3), 267–278 (1994)
3. D'Orazio, T., Leo, M., Guaragnella, C., Distante, A.: A visual approach for driver inattention detection. Pattern Recognition 40(8), 2341–2355 (2007)
4. Fisher, R.A.: Statistical Methods for Research Workers. Hafner Publishing (1970)
5. Jang, J.S.R.: ANFIS: Adaptive-Network-Based Fuzzy Inference System. IEEE Transactions on Systems, Man, and Cybern. 23, 665–684 (1993)
6. Matsumoto, Y., Ogasawara, T., Zelinsky, A.: Behavior recognition based on head pose and gaze direction measurement. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Takamatsu, Japan, pp. 2127–2132 (2000)
7. Otsuka, K., Takemae, Y., Yamato, J.: A probabilistic inference of multiparty-conversation structure based on markov-switching models of gaze patterns, head directions, and utterances. In: ICMI, pp. 191–198 (2005)
8. Smith, P., Member, S., Shah, M., Lobo, N.D.V.: Determining driver visual attention with one camera. IEEE Trans. on Intelligent Transportation Systems 4, 205–218 (2003)
9. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man Cybern. 15(1), 116–132 (1985)
10. Victor, T., Blomberg, O., Zelinsky, A.: Automating driver visual behavior measurement. In: 9th Vision in Vehicles Conference, Australia (2001)
11. Viola, P.A., Jones, M.J.: Rapid object detection using a boosted cascade of simple features. In: Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 511–518 (2001)
12. Weidenbacher, U., Layher, G., Bayerl, P., Neumann, H.: Detection of head pose and gaze direction for human-computer interaction. In: André, E., Dybkjær, L., Minker, W., Neumann, H., Weber, M. (eds.) PIT 2006. LNCS, vol. 4021, pp. 9–19. Springer, Heidelberg (2006)
13. Zhou, Z.H., Geng, X.: Projection functions for eye detection. Pattern Recognition 37(5), 1049–1056 (2004)