Non-verbal feedback on user interest based on gaze direction and head pose

S. Asteriadis, P. Tzouveli, K. Karpouzis, S. Kollias
Image, Video and Multimedia Systems Laboratory
National Technical University of Athens
GR-157 80 Zographou, Greece
{stiast, tpar, kkarpou}@image.ntua.gr,
stefanos@cs.ntua.gr

Abstract

When users of computer systems are given the opportunity to provide feedback on their preferences or interests in most cases they are presented with a questionnaire to be filled. However, this means of interaction requires an additional cognitive step, since the user is required to rationalize feelings and attitudes by answering predetermined questions; as a result, users often skip this part or answer hastily, depriving systems of valuable criticism 'from the field'. In this paper, we present a system which relates data extracted from the posture and movement of the user's head and eye gaze with states related to interest and engagement. This system is deployed in the context of Human Computer Interaction, where users read documents from a computer screen and provides non-verbal input which, in turn, can be further processed.

1. Introduction

Non-verbal feedback is an important aspect of Human-Computer Interaction (HCI), since it presents users with the opportunity to inform the system about related preferences or dispositions without the need to answer specific questions, usually predetermined by the system developers, or entering free text comments. From the part of the system, an unintrusive method of receiving this kind of information is much more robust, since it produces replicable results, and can be deployed when users actually operate the system via an interface, alleviating the need for an additional step.

In this paper, we build on information provided by facial features of users in front of a computer screen to calculate measurable indicators of their interests towards particular electronic documents. More specifically, we use the position of prominent points around the eyes and the position of the irises to reconstruct a 3-D vector which illustrates the direction of gaze and head pose. This vector provides an indication of whether the particular user looks into the screen or not and whether their eyes are fixed at a particular spot for long periods of time. In addition to this, statistical measures of these features and their transformation over time (motion, fluidity, etc.) are associated with facial and head expressivity, which is also a major factor of non-verbal input.

This paper is organized as follows: Section 2 describes the lower-level processes used to locate and track the prominent facial feature point, while Section 3 correlates this information with particular affective states, providing actual results from system deployment. Section 4 concludes the paper.

2. Detection and tracking of facial features – gaze and pose estimation

2.1 Facial feature detection and tracking

Facial feature extraction is a crucial step to numerous applications such us face recognition, human-computer interaction, facial expression recognition, surveillance and gaze/pose detection. In their vast majority, the approaches in bibliography use face detection as a preprocessing step. This is usually necessary in order to tackle with scale problems, as, localizing a face in an image is more scale-independent than starting with the localization of special facial features. When only facial features are detected (starting from the whole image and not from the face region of interest), the size and the position of the face in the image have to be predetermined and, thus, such algorithms are devoted to special cases, such as driver's attention recognition [13] where the user's position with regards to a camera is almost stable. In such techniques, color [13] predicates, shape of facial features and their geometrical relations [18] are used as criteria for the extraction of facial characteristics.

On the other side, facial features detection is more scale-independent when the face is detected as a pre-processing step. In this case, the face region of interest can be normalized to certain dimensions, thus making the task of facial feature detection more robust. For example, in [4] a multi-stage approach is used to locate features on a face. First, the face is detected using the boosted cascaded classifier algorithm by Viola and Jones [14]. The same classifier is trained using facial feature patches to detect facial features. A novel shape constraint, the Pairwise Reinforcement of Feature Responses (PRFR) is used to improve the localization accuracy of the detected features. In [11], a three stage technique is used for eye center localization. The Hausdorff distance between edges of the image and an edge model of the face is used to detect the face area. At the second stage, the Hausdorff distance between the image edges and a more refined model of the area around the eyes is used for more accurate localization of the upper area of the head. Finally, a Multi-Layer Perceptron (MLP) is used for finding the exact pupil locations. In [14], an SVM-based approach is used for face detection. Following, eye-areas are located using a feed-forward neural network and the face is brought to a horizontal position based on the eye positions. Starting from these points, edge information and luminance values, are used for eyebrow and nostrils detection. Further masks are created to refine the eye positions, based on edge, luminance and morphological operations. Similar approaches are followed for the detection of mouth points.

In this work, prior to eye and mouth region detection, face detection is applied on the face images. The face is detected using the Boosted Cascade method, described in [14]. The output of this method is usually the face region with some background. Furthermore, the position of the face is often not centered in the detected sub-image. Since the detection of the eyes and mouth will be done on blocks of a predefined size, it is very important to have an accurate face detection algorithm. Consequently, a technique to postprocess the results of the face detector is used.

More specifically, a technique that compares the shape of a face with that of an ellipse is used. This technique is based on the work reported in [11]. According to this, the distance map of the face area found at the first step is extracted. Here, the distance map is calculated from the binary edge map of the area. An ellipsis scans the distance map and a score that is the average of all distance map values on the ellipse contour *el*, is evaluated.

$$score = \frac{1}{el} \sum_{(x,y)\in el} D(x,y)$$

where D is the distance map of the region found by the Boosted Cascade algorithm. This score is calculated for various scale and shape transformations of the ellipses. The transformation which gives the best score is considered as the one that corresponds to the ellipses that best describes the exact face contour. The lateral boundaries of the ellipses are the new boundaries of the face region.

A template matching technique follows for the facial feature area detection step: The face region found by the face detection step is brought to certain dimensions and the corresponding Canny edge map is extracted. Subsequently, for each pixel on the edge map, a vector pointing to the closest edge is calculated and its *x,y* coordinates are stored. The final result is a vector field encoding the geometry of the face. Prototype eye patches were used for the calculation of their corresponding vector fields and the mean vector field was used as prototype for searching similar vector fields on areas of specified dimensions on the face vector field. The similarity between an image region and the templates is based on the following distance measure:

$$E_{L_2} = \sum_{i\in R_k} \| v_i - m_i \|$$

where $\| \|$ denotes the L_2 norm. Essentially for a *NxM* region R_k the previous formula is the sum of the euclidean distances between vectors v_i of the candidate region and the corresponding m_i of the mean vector field (template) of the eye we are searching for (right or left). The candidate region on the face that minimizes E_{L2} is marked as the region of the left or right eye. To make the algorithm faster we utilize the knowledge of the approximate positions of eyes on a face.

For the eye center detection, the normalized area of the eye is brought back to its initial dimensions on the image and a light reflection removal step is employed. The grayscale image of the eye area is converted to a binary image and small white connected components are removed. The areas that correspond to such components on the original image are substituted by the average of their surrounding area. The final result is an eye area having reflections removed. Subsequently, horizontal and vertical derivative maps are extracted from the resulting image and they are projected on the vertical and horizontal axis respectively. The mean of a set of the largest projections is used for an estimate of the eye center. Following, a small window around the detected point is used for the darkest patch to be detected, and its center is considered as the refined position of the eye center.

For the detection of the eye corners (left, right, upper and lower) a technique similar to that described in [15] is used: Having found the eye center, a small area around it is used for the rest of the points to be de-

tected. This is done by using the Generalized Projection Functions (GPFs) which are a combination of the Integral Projection Functions (IPFs) and the Variance Projection Functions (VPFs). The integral projection function's value on row (column) *x (y)* is the mean of its luminance intensity, while the Variance Projection Function on row *x* is its mean variance. The GPF's value on a row (column) *x (y)* is a linear combination of the corresponding values of the derivatives of the IPF and VPF on row *x* (column *y*):

$$GPF_u(x) = (1-a) * IPF_u^{'}(x) + a * VPF_u^{'}$$

$$GPF_v(y) = (1-a) * IPF_v^{'}(y) + a * VPF_v^{'}$$

Local maxima of the above functions are used to declare the positions of the eye boundaries.

For the mouth area localization, a similar approach to that of the eye area localization is used: The vector field of the face is used and template images are used for the extraction of a prototype vector field of the mouth area. Subsequently, similar vector fields are searched for on the lower part of the normalized face image. However, as the mouth has, many times, similar luminance values with its surrounding skin, an extra factor is also taken into account. That is, at every search area, the mean value of the hue component is calculated and added to the inverse distance from the mean vector fields of the mouth. Minimum values declare mouth existence.

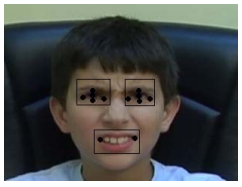

Figure 1: Detected facial features

For the extraction of the mouth points of interest (mouth corners), the hue component is also used. Based on the hue values of the mouth, the detected mouth area is binarized and small connected components whose value is close to 0^o are discarded similar to the light reflection removal technique employed for the eyes. The remainder is the largest connected component which is considered as the mouth area. The leftmost and rightmost points of this area are considered as the mouth corners. An example of detected feature points is shown in Figure 1.

Once the positions of the facial feature points of interest are known on a frontal face, tracking can follow. In this way, gaze detection and pose estimation can be determined, not only on a single frame, but on a series of frames. Also, calculating changes of the interocular distance in a series of frames, it is easy to determine the distance of a user from the camera. Furthermore, tracking saves computational time, since detecting the characteristics at every frame is more time demanding, and can achieve better results in cases of large displacement of the face from its frontal position. In our case, tracking was done using an iterative, 3-pyramid Lucas-Kanade tracker [7]. An example of a person's movement with relation to the camera is shown in Figure 2.

Figure 2: Example of determining a person's movement towards the camera

2.2 Gaze detection and pose estimation

In recent bibliography, most gaze detection and pose determination techniques need special hardware setup. Examples of such cases are the work described in [6], where a large resolution image of the iris is necessary and the work in [17], where a specific architecture has to be followed. In other cases, intrusive devices have to be worn by the user [3], making the system less appropriate for wide-range applications.

In the current work, features are detected and tracked, allowing for a relative freedom of the user, under good lighting conditions. Under these circumstances, the gaze directionality can be approximately determined, which is enough for attention recognition purposes, as well as for general decisions regarding one's gaze. For gaze detection, the area defined by the four points around the eye is used. Eye areas depicting right, left, upper and lower gaze directionalities are used to calculate mean grayscale images corresponding to each gaze direction. The areas defined by the four detected points around the eyes, are then correlated to these images.

$$H_r = \frac{(R_{r,l} - R_{r,r})}{\max(R_{r,l}, R_{r,r}, R_{r,u}, R_{r,d})}, V_r = \frac{(R_{r,u} - R_{r,d})}{\max(R_{r,l}, R_{r,r}, R_{r,u}, R_{r,d})}$$

$$H_l = \frac{(R_{l,l} - R_{l,r})}{\max(R_{l,l}, R_{l,r}, R_{l,u}, R_{l,d})}, V_l = \frac{R_{l,u} - R_{l,d}}{\max(R_{l,l}, R_{l,r}, R_{l,u}, R_{l,d})}$$

The normalized differences of the correlation values of the eye area with the left and right as well as upper and lower mean gaze images are calculated with the above equations, where $R_{i,j}$ is the correlation of the *i* (i=*left,right*) eye with the *j* (*j=left, right, upper, lower*) mean grayscale image. The normalized value of the horizontal and vertical gaze directionalities (conventionally, angles) are then the weighted mean:

$$H = ((2-l)\cdot H_r + l\cdot H_l)/2$$

$$V = ((2-l)\cdot V_r + l\cdot V_l)/2$$

Where l is the fraction of the mean intensity in the left and right areas. This fraction is used to weight the gaze directionality values so that eye areas of greater luminance are favored in cases of shadowed faces.

To estimate the pose of a face based on the features detected, orthographic projection can be assumed for a linear system to be constructed, since depth information is not necessary for pose estimation. The pose of the face is a problem of estimating the direction of the face-plane which depends on the changes of the distances between facial characteristics. Thus, if the eye and mouth centers are considered, it is possible to initialize a triangle *A,B,C,* with *A,B* being the left and right eye centers and *C* the mouth center at the frontal view. Let *A',B'* and *C'* be the displaced positions of these points, which are considered to be known since the points are tracked. The α,β,γ rotation angles around the *y,z,x*-axis are [1]:

$$\alpha = \arccos\frac{C_y'}{C_y\cos\gamma} \quad \beta = \arccos\frac{A_x'}{A_x\cos\gamma} \quad \gamma = \arcsin\frac{C_x'}{C_y}$$

2.3 Hand detection and tracking

Regarding gesture analysis, several approaches have been reviewed for the head-hand tracking module all of them mentioned both in [19] and in [16]. From these only video based methods were considered since motion capture or other intrusive techniques would interfere with the person's emotional state. The major factors taken under consideration are computational cost and robustness, resulting in an accurate near real-time skin detection and tracking module. The general process involves the creation of *moving skin* masks, namely skin color areas that are tracked between subsequent frames. By tracking the centroid of those masks, we produce an estimate of the user's movements. A priori knowledge concerning the human body and the circumstances when filming the gestures was incorporated into the module indicating the different body parts (head, right hand, left hand).

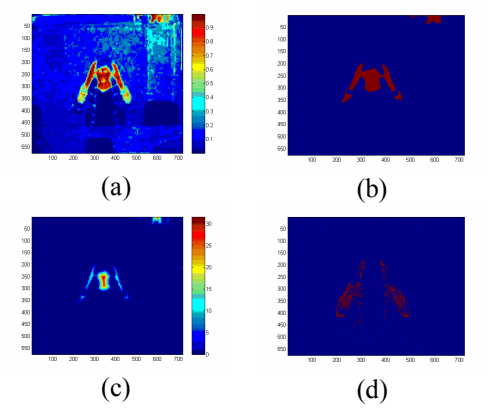

Figure 3: Key steps in hand detection and tracking (a) Skin probability (b) Thresholding & Morphology operators (c) Distance transformation (d) Frame Difference

For each frame a skin color probability matrix is computed by calculating the joint probability of the Cr/Cb image values. The *skin color* mask is then obtained from the skin probability matrix using thresholding. Possible moving areas are found by thresholding the difference between the current frame and the next, resulting in the *possible-motion* mask. This mask does not contain information about the direction or the magnitude of the movement, but is only indicative of the motion and is used to accelerate the algorithm by concentrating tracking only in moving image areas. Both color and motion masks contain a large number of small objects due to the presence of noise and objects with color similar to the skin. To overcome this, morphological filtering is employed on both masks to remove small objects. All described morphological operations are carried out with a disk-structuring element with a radius of 1% of the image width. The distance transform of the color mask is first calculated and only objects above the desired size are retained. These objects are used as markers for the morphological reconstruction of the initial color mask. The color mask is then closed to provide better centroid calculation. For the next frame, a new moving skin mask is created, and a one-to-one object correspondence is performed. Object correspondence between two frames is performed on the color mask and is based on object centroid distance for objects of similar (at least 50%) area. In the case of hand object merging and splitting, e.g., in the case of clapping, we establish a new matching of the left-most candidate object to the user's right hand and the right-most object to the left hand. The Sagittal plane information of the gesture was ignored since it would require depth information from the video stream and it would make the performance of the proposed algorithm very poor or would require a side camera and parallel processing of the two streams. The object

correspondence heuristic makes it possible to individually track the hand segments correctly, at least during usual meaningful gesture sequences (see Figure 3). In addition, the fusion of color and motion information eliminates any background noise or artifacts, thus reinforcing the robustness of the proposed approach.

3. Affective states

3.1 Prerequisites for the detection of affective states

The human face is the site for major sensory inputs and major communicative outputs. It houses the majority of our sensory apparatus as well as our speech production apparatus. It is used to identify other members of our species, to gather information about age, gender, attractiveness, and personality, and to regulate conversation by gazing or nodding. Moreover, the human face is our pre-eminent means of communicating and understanding somebody's affective, cognitive, and other mental states and intentions on the basis of the shown facial expression. Hence, automatic analysis of the human face is indispensable in the context of natural HCI [9].

The first step in facial information processing is face detection, i.e., identification of all regions in the scene that contain a human face. The problem of finding faces should be solved regardless of clutter, occlusions, and variations in head pose and lighting conditions. The presence of non-rigid movements due to facial expression and a high degree of variability in facial size, color and texture make this problem even more difficult. Numerous techniques have been developed for face detection in still images ([10]). However, most of them can detect only upright faces in frontal or near-frontal views [15]. A method that can handle out-of-plane head motions is the statistical method for 3D object detection proposed by [5]. Other methods, e.g. [8], emphasize statistical learning techniques and use appearance features, including the real-time face detection scheme proposed by Viola and Jones ([12]), which is arguably the most commonly employed face detector in automatic facial expression analysis.

Gaze direction, that is, the direction to which the eyes are pointing in space, is a strong indicator of the focus of attention, and it has been studied extensively [2]. Eye tracking systems can be grouped into head-mounted or remote, and infra-red-based or appearance-based. Infra-red-based gaze tracking systems employ the so-called red-eye effect, i.e., the difference in reflection between the cornea and the pupil. Appearance-based gaze trackers employ computer vision techniques to find the eyes in the input image and then determine the orientation of the irises. While head-mounted systems are the most accurate, they are also the most intrusive. Infrared systems are more accurate than appearance-based, but there are concerns over the safety of prolonged exposure to infra-red lights.

3.2 Classification of affective states

Table 1 summarizes the initial mapping between positions of the detected feature points, distances between them and gaze and pose direction to the different affective states. This mapping is currently used to develop and test feature extraction algorithms and will be adapted when the recordings have been annotated at the different recording sites. Following annotation of the recorded data, mapping of the above features and user states can be further mapped to affective states, i.e. *Distracted*, *Tired/Sleepy, Not paying attention, Attentive, Full of interest, Curious*, etc. Examples of early decisions regarding attention are shown in the following figures.

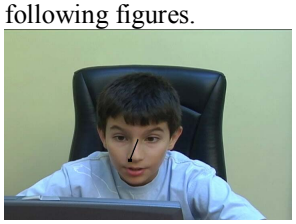

Figure 4: paying attention to the screen, eyes open/full of interest

Figure 5: paying attention, eyes not wide open/curious

Figure 6: paying attention, eyes closed/tired

Figure 7: distracted from the screen

User state	Related feature or feature point(s)	Condition
eyes looking at the screen	gaze direction	angle < 15 deg
eyes not looking at the screen	gaze direction	angle > 15 deg
eyes wide open	eye area	full size
head is moving (fast, slowly etc.)	skin area	centroid motion > ½ of skin area
head is not moving	skin area	centroid motion < 1/10 of skin area
head approaches screen	eye corners	interocular distance increasing
head moves away from the screen	eye corners	interocular distance decreasing
mouth wide open	mouth area	size increasing
hand covers	mouth area	occluded by an-

mouth		other skin patch
hand covers eyes	eye area	occluded by another skin patch

Table 1: User states and relation to feature points and gaze direction

4. Conclusions

An important possibility of non-verbal feedback concerning the interest of a person towards a web page, multimedia presentation, video clip or any other form of electronic document is the degree of engagement towards the computer screen it is shown on. Head pose, direction of gaze, as well as measurements of expressivity are a vital part of this kind of feedback. We present a system used in the context of HCI to extract the degree of interest and engagement of students reading documents from a computer screen. In the following, this system will be user to correlate student performance and individual reading habits with the presence of dyslexia and to provide measurable feedback on the progress of therapy.

5. References

[1] A. Yilmaz and M. A. Shah, Automatic feature detection and pose recovery for faces, *Asian Conference on Computer Vision*, 23-25, 2002.

[2] A.T. Duchowski, A Breadth-First Survey of Eye Tracking Applications, *Behavior Research Methods, Instruments, and Computing*, 34(4):455-70, 2002.

[3] D. Beymer, M. Flickner, M., Eye gaze tracking using an active stereo head, *CVPR*, 2, 451–458, 2003

[4] D. Cristinacce, T. Cootes and I. Scott, A multi-stage approach to facial feature detection, *15th British Machine Vision Conference*, pp. 231-240, 2004.

[5] H. Schneiderman, T. Kanade, A statistical model for 3D object detection applied to faces and cars, *CVPR*, 746-751, 2000.

[6] J.G. Wang, E. S. and Venkateswarku, R., Eye gaze estimation from a single image of one eye, 9th *IEEE Int. Conf. on Computer Vision*, 2003.

[7] J.Y. Bouguet, Pyramidal Implementation of the Lucas Kanade Tracker, OpenCV Documentation.

[8] K.S. Huang, M.M. Trivedi, Robust real-time detection, tracking, and pose estimation of faces in video, *ICPR*, vol. 3, 965-968, 2004.

[9] M. Pantic, I. Patras, Dynamics of facial expression: Recognition of facial actions and their temporal segments from face profile image sequences, *IEEE Trans. SMC-B*, 36(2): 433-449, 2006.

[10] M.H. Yang, D.J. Kriegman and N. Ahuja, Detecting faces in images: A survey, *IEEE Trans. PAMI*, 24(1): 34-58, 2002.

[11] O. Jesorsky, K. J. Kirchberg and R. W. Frischholz, Robust face detection using the Hausdorff distance, *3rd Conf. AVBPA*, 90-95, 2001.

[12] P. Viola, M. Jones, Robust real-time face detection, *J. Computer Vision*, 57(2): 137-154, 2004.

[13] P. Smith, M Shah, N. da Vitoria Lobo, Determining Driver Visual Attention with One Camera, *IEEE Trans. Intelligent Transportation Systems*, Vol 4, No. 4, pp. 205-218, 2003.

[14] P. Viola and M. Jones, Rapid object detection using a boosted cascade of simple features, *IEEE Computer Vision and Pattern Recognition*, Vol. 1, pp. 511-518, 2001.

[15] S. Ioannou, G. Caridakis, K. Karpouzis, S. Kollias, Robust Feature Detection for Facial Expression Recognition, *EURASIP Journal on Image and Video Processing*, Hindawi Publishing Corporation, 2007.

[16] S. Ong, S. Ranganath (2005) Automatic Sign Language Analysis: A Survey and the Future beyond Lexical Meaning, *IEEE. PAMI*, 27(6), 873 – 891.

[17] SensoMotoric Instruments. EyeLink Gaze Tracking. www.smi.de

[18] T. D' Orazio, M. Leo, G. Cicirelli and A. Distante, An algorithm for real time eye detection in face images, *ICPR*, Vol. 3, 278-281, 2004.

[19] Y. Wu, T. Huang (2001). Hand modeling, analysis, and recognition for vision-based human computer interaction. *IEEE Signal Proc.,* 18, 51-60.

[20] Z.H. Zhou and X. Geng, Projection functions for eye detection, *Pattern Recognition*, 37 (5), 1049-1056.