EYELIDS AND FACE TRACKING IN REAL-TIME

Javier Orozco[†], P. Baiget[†] , J. Gonzàlez[‡], X. Roca[†]

[†] Computer Vision Center & Dept. d'Informàtica,
Edifici O, Campus UAB, 08193 Bellaterra, Spain
[‡] Institut de Robòtica i Informàtica Industrial (UPC – CSIC),
Llorens i Artigas 4-6, 08028, Barcelona, Spain

ABSTRACT
Tracing and tracking facial features in a precise manner are crucial tasks for Human Computer Interaction, facial expression recognition, and image retrieval. Eyelids tracking is an important and hard subject to evaluate human emotions due to the high expressivity of the eyes and its faster movement. For Active Appearance Models (AAM) is a challenge, since related frameworks have showed a good performance using edge detectors, color information, and thresholding techniques all of them depending on the quality of the image. We built two appearance-based models (ABM) to track simultaneously eyelids tracking with 3D head pose, eyebrows and lips in monocular video sequences. This paper has two main contributions. Firstly, we show that by adopting a non-occluded facial texture model in the eyes region, 3D head pose parameters can be obtained in an accurate and stable manner. Secondly, unlike previous approaches regarding eyelids tracking, we prove that the Online Appearance Models (OAMs) can be used for eyelids tracking without color information, eyes feature extraction, or edge detectors. Experiments in real videos show the feasibility and usefulness of this approach, down-weighting time and memory consumption, and improving the accuracy.

KEY WORDS
Eyelids Tracking, Face Tracking, AAM, OAM.

1 Introduction

Eyelids movements are motived by the relevance in expression of facial emotions, which can be evaluated in a gaze. To analyze eyes movements from deformable models is a complicated approach with few contributions as well as to analize the 3D head pose, brows, lips and eyelids tracking together.

Some authors have proposed deformable template matching [4], Hugh transform and color information of image for eyes state. Others have regarded the eyelids and face tracking problems in two different ways; on the one hand, eyelids tracking based on feature extraction, edge detector, or thresholding. On the other hand, 3D head and face tracking involving feature based-tracker and active appearance models. In [3] facial feature and head pose are followed without including eyelids and irises movements; however, color information and texturing are used.

To overcome the problem of appearance changes, recent works on faces have adopted statistical facial textures. Active Appearance Models (AAMs) have been proposed as a powerful tool to analyze facial images. Deterministic and statistical ABM methods have been proposed and used by some researchers handling successfully image variability and drift problems[6]. Indeed other authors have worked jointly head and eyelids movements, based on color information of the image, edge detector, and texture models, except for the facial action tracking [1, 9, 7]. On the contrary, the eyes tracking problem has received many contributions from works that used edge detectors, color space and Hough transformations, and template matching [5, 11]. In [10], eyelids state is inferred; finding a relative distance between the eyelid apex and the iris centre using edge detectors, describing only open and closed states.

This paper extends our previous work in two directions. Firstly, we show that it is possible to track the eyelids motion by adopting a non-occluded shape-free facial texture. We prove the dependency between the eyes region and the eyes-less template, in order to obtain a more accurate and stable 3D head pose. Secondly, unlike feature-based eyelids trackers, we show how the eyelids are tracked by using OAMs. Thus, our proposed approach is skillful for inferring eyes state without detecting eyes features such as the irises and eyes corners.

We do not use any edges and there is no assumption on the head pose. Eyelids motion is inferred at the same time with the 3D head pose and other facial actions. The eyelids state does not rely on eyes corners and iris detection. Faster eyelids tracking is not a straightforward task. The challenges are as follows: firstly, the upper eyelid is a highly deformable facial feature since it has a great freedom of motion. Secondly, the eyelid can completely occlude the iris and the sclera; a facial texture model may have two different appearances at the same location. Thirdly, eyelids motion is fast compared to eyebrows, lips and head motion.

The rest of the paper is organized as follows: in Section 2 the deformable 3D facial model is described. In Section 3, we explain the basis of the online adaptive appearance model. In Section 4, the eyelids and face tracking is

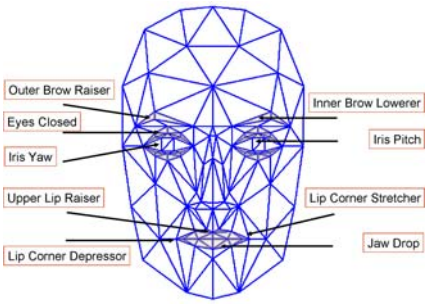

Figure 1. Candide Model

presented, we show the relationship between the face and the eyes region. In Section 5, we present the results from the hierarchical adaptive appearance-based tracker which tracks in real-time the 3D head pose, as well as some facial actions that include eyelids movements and some special and robust results. Finally, in Section 6 we present the conclusion.

2 Synthesizing Faces

2.1 3D Face Model

We use the *3D face Candide Model*[2] as a deformable face model to build an active appearance model. The Candide model is a wire-frame composed by 113 vertices and 183 triangles, used for modeling faces in 3D (Fig. 1).

This 3D face model takes into account the differences between specific human faces as well as the different facial expressions; the 3D deformable wire-frame records the 3D shape as local coordinates. The 3D vertices $\mathbf{P}_i$ give the 3D face model, the shape up to a global scale which can be described by the 3n-vector $\mathbf{V}$:

$$\mathbf{V}_n = V_0 + \mathbf{D}\vec{\vartheta} + \mathbf{A}\vec{\gamma} \tag{1}$$

where V_0 is the standard shape, matrix $\mathbf{D}$ accounts shape parameters, $\vec{\vartheta}$ is a control shape vector, matrix $\mathbf{A}$ accounts animation parameters, and $\vec{\gamma}$ is a animation control vector. One should relate the 3D coordinates to the image coordinate system in order to adopt the weak perspective projection model.

Thus a 3D vertex $\mathbf{P}_i = (X_i, Y_i, Z_i)^T \subset \mathbf{V}_n$ will be projected onto the image point $\mathbf{p}_i = (u_i, v_i)^t$ given by:

$$(u_i, v_i)^t = \mathbf{M}(X_i, Y_i, Z_i, 1)^t \tag{2}$$

The constant ϑ is dependent for each person. Estimating the vector ϑ can be carried out using either feature-based or featureless approaches. The state of the 3D model is given by the 3D head pose (three rotations and three translations) and the control vector $\vec{\gamma}$ comes from tracking or initialization within the vector $\mathbf{q}$:

$$\mathbf{q} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \vec{\gamma}^t]^t \tag{3}$$

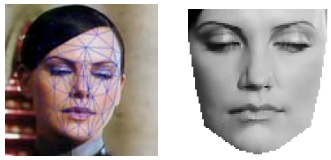

Figure 2. Face and 3D mesh. Shape Free.

2.2 Ground Texture

A shape-free texture represents a face texture which is obtained by the projection of the standard shape V_0 using a standard 3D pose (frontal view) onto an image with a given resolution. The texture of this geometrically normalized image is obtained by texture mapping from the triangular 2D mesh in an input image using a piece-wise affine transform Ψ. Mathematically the warping process applied to an input image $\mathbf{I}$ is denoted as: (Fig. 1).

$$\vec{\chi}(\mathbf{q}) = \Psi(\mathbf{I}, \mathbf{q}) \tag{4}$$

Where $\vec{\chi}$ is the shape-free texture and $\mathbf{q}$ the geometrical parameters. The images are one-dimensional vectors.

Indeed, having a video sequence depicting a moving face, tracking consists of estimating the 3D head pose and facial animation encoded by the geometrical vector $\mathbf{q}$. The model parameters associated with the current frame will be handled by the next frame. (Fig. 2) show the 3D face onto the input image with its warped version.

The Candide model has an initial position including eyes region information about eyes, which is considered outlier when the eyelids are being tracked. To use a bad outlier threshold parameter does not improve robustness because erroneous pixels in the statistical Gaussian model would be introduced, causing drifting problems.

The same standard model has a configuration of facial action for the eyelids, which involves self-occlude eyelids, creating confusion for the Jacobian matrix estimation and directions of gradient descent. These two reasons state to adapt the generic Candide model in order to avoid sclera and the iris information. Consequently, considering a dynamic model without occlusions, the eyes region vertices will move synchronized without self-overlapping.

3 Tracking with AAM

Tracking consists in estimating the 3D head pose and facial animations, following pixel by pixel, search variations of features or facial animations encoded in the geometrical vector $\mathbf{q}$. In other words, based in the Candide model, we should estimate the vector $\mathbf{q}$ (Eq. (6)) for each frame t that later will be handed over to the next frame.

We are interested in estimating for each input frame I_t the corresponding geometrical vector $\mathbf{q}_t$, observation is the warped texture associated with the animation parameters $\vec{\gamma}$.

We represent as $\hat{\chi}_t$ the tracked parameters and textures. For a given frame t, $\hat{q}_t$ will be the computed geometric parameters and $\hat{\chi}_t$ the corresponding shape-free patch, that is,

$$\hat{\chi}_t = \chi(\hat{\mathbf{q}}_t) = \Psi(\mathbf{I}_t, \hat{\mathbf{q}}_t) \tag{5}$$

From here, the vector $\mathbf{q}$ will be referenced according to the number of components for α^i and γ^j, Eq. (6, 7, 8).

$$\mathbf{q} = [\alpha^i, \gamma^j]^t \tag{6}$$

$$\alpha^i = [\theta_x, \theta_y, \theta_z, t_x, t_y, s]^t = [\alpha^0, ..., \alpha^5]^t \tag{7}$$

$$\gamma^j = [\gamma^0, ..., \gamma^6]^t \tag{8}$$

3.1 Appearance Models

The appearance model χ obeys a Gaussian with a center μ_i and a variance σ_i^2, which are vectors of k pixels (k are pixels different to zero) assumed to be independent.

The probability for each observation is given by:

$$p(\mathbf{I}_t|\mathbf{q}_t) = p(\chi_t|\mathbf{q}_t) = \prod N(\chi_i; \mu_i, \sigma_i) \tag{9}$$

It is assumed that the model holds all observations under an exponential with an updating factor ω. When the appearance is tracked for the current input image, the mean and variance are updated in the next frame with respect to the current parameters (Eq. 3), thus:

$$\mu_{t+1} = \omega\mu_t + (1-\omega)\hat{\chi}_t \tag{10}$$

$$\sigma_{t+1}^2 = \omega\sigma_t^2 + (1-\omega)(\hat{\chi}_t - \mu_t)^2 \tag{11}$$

This technique is simple, time-efficient and suitable in real-time. μ_i and σ_i are initialized with the first patch χ, but the above model is not used until the number of frames reach a certain value (e.g., the first 50 frames). For these frames, the ω factor is set to $1 - 1/t$.

3.2 Appearance Transition

An adaptive velocity model is adopted, when the adaptive motion velocity is predicted using a fixed function for estimating the transition state;

$$\mathbf{q}_t = \hat{\mathbf{q}}_{t-1} + \Delta\mathbf{q}_t \tag{12}$$

Where $\Delta\mathbf{q}_t$ is the shift in the geometric parameters. The current input image $\mathbf{I}_t$ is registered with the current appearance model minimizing the *Mahalanobis distance* between the warped texture and the current appearance mean,

$$\min_{q_t} M(\mathbf{q}_t) = \min_{q_t} D(\chi(\mathbf{q}_t), \mu_t) = \Sigma[\frac{(\chi_i - \mu_i)}{\sigma_i^2}] \tag{13}$$

Here, the appearance parameters μ and σ are known, and the Mahalanobis distance is minimized by an iterative first-order linear approximation.

Gradient-descent and Gauss-Newton iteration

Given a vector $\mathbf{q}_t = \hat{q}_{t-1} + \Delta\mathbf{q}_t$, so that the warped texture will be very close to the appearance mean,

$$\Psi(\mathbf{I}_t, \mathbf{q}_t) \quad \approx \quad \mu_t \tag{14}$$

Approximating the Eq. 14 via a first-order Taylor series expansion around $\hat{q}_{t-1}$ and using the Vanilla gradient descent method, we obtain the Jacobian matrix $\mathbf{J}_t$ [8]:

$$\Psi(\mathbf{I}_t, \mathbf{q}_t) \quad \approx \Psi(\mathbf{I}_t, \hat{q}_{t-1}) + \mathbf{J}_t(\mathbf{q}_t - \hat{q}_{t-1}) \tag{15}$$

$$\mu_t = \Psi(\mathbf{I}_t, \hat{q}_{t-1}) + \mathbf{J}_t(\mathbf{q}_t - \hat{q}_{t-1}) \tag{16}$$

$$\Delta\mathbf{q}_t = \mathbf{q}_t - \hat{q}_{t-1} = -\mathbf{J}_t^*[\Psi(\mathbf{I}_t, \hat{q}_{t-1}) - \mu_t]^1 \tag{17}$$

Thus, the change in the vector $\mathbf{q}_t$ is the change of the warped texture of the above image and the average current image. The solution for this vector is achieved by iterations until the error measurements for the Mahalanobis distance are minimum (Eq. 13). The Jacobian matrix is computed by approximate differences.

Advantages of the Vanilla gradient descent method [8] are the capability to accommodate appearance changes and its robustness, as it is computed for the current geometric configuration.

4 Eyelids and Face Tracking Problems

Related works with eyelids tracking and eyes states have shown the difficulty of this framework; relating this problem with fast motion. Several associated approaches only detect eyes blinking, that is, only eyes state detection is achieved, thus avoiding continuous movement tracking.

For surveillance systems, human computer interaction, and real-time applications, eyelids tracking should be obtained within a reduced time rate while maintaining the robustness of the whole system. However, the use of space color information and image transformations like contour detectors is time-consuming.

4.1 3D Head Pose, Eyebrows and Lips Tracking

3D head pose, eyebrows and lips tracking are stable using appearance-based trackers as described above. Combining the estimation of the geometrical vector gradient $\mathbf{q}$ with an especial function to constrain the descent curve, we can

[1] J_t^* is pseudo-inverse and transpose of Jacobian J

control the adaptive register of the estimated appearance on the probabilistic model, while setting a criterion to consider a pixel as an outlier.

The process will constrain the gradient descent, and the updating of the OAM. The $\hat{\xi}$ function is defined as:

$$\xi(y) = \begin{cases} 1 & \textbf{if} |y| \leq c \\ \frac{c}{|y|} & \textbf{if} |y| > c \end{cases} \quad (18)$$

Where y is the value of a pixel in the patch χ_t, normalized by the mean μ_i and the variance of the AAM at the same pixel σ_i^2. The constant c controls the outlier rate. We set C to 3 according to three times the variance.

The pixel y is outlier when $|y| > c$. The computation for Jacobian matrix to down-weight the influence of outlier pixels in the registration technique, according to:

$$\Delta \mathbf{q}_t = \mathbf{q}_t - \hat{q}_{t-1} = -\mathbf{J}_t^* \xi_t [\Psi(\mathbf{I}_t, \hat{q}_{t-1}) - \mu_t] \quad (19)$$

Illumination changes, image quality, different objects other than faces and quick movements are considered as outliers and are not included in appearance model. This outlier data is accepted as predefined before tracking. Moreover, head, brows, and lips move slower than the eyes.

We first attempt to compare the influence of the eyes region motion onto the head and face estimation. Using the same appearance-based tracker described in section 3, we assume that the geometrical vector **q** only accounts for six parameters of 3D head pose, and six face initial facial actions (eyebrows and lips).

Test details, Fig. 3, show 250-frames with nineteen blinks. Note that the facial actions are spontaneous. The camera is not in a frontal position, which is a challenge for the tracker, and still thus, the performance is good.

Our tracker was tested with two textures, the first one with the eyes region and the other one without the eyes region. In the figure, we can see the differences between the two results for the estimated scale and pitch angle. These differences are notorious when there is eyes blinking (the picks on the plot). The same differences could be observed on the OX angle. Due to this, the head pose estimation is affected because of the noise incorporated to the eyes region and their movement, and therefore, the tracking of the six facial actions will not be precise.

We can note in Fig. 3, how the pitch angle is affected for eyes closed in an average difference of 8.5 degrees at frames 160 to 170. At frame 170 the estimated scale deviation/error depth is of approximately 0.15, which corresponds to an in-depth error of about 18 centimeters[2].

4.2 Eyelids Tracking

In the above section, we have proved the relationship between the eyes region and the eyes-less face. We use a reference texture of 84 x 80 pixels, when there are 5447

[2]The exact value depends on the camera intrinsic parameters and the absolute depth

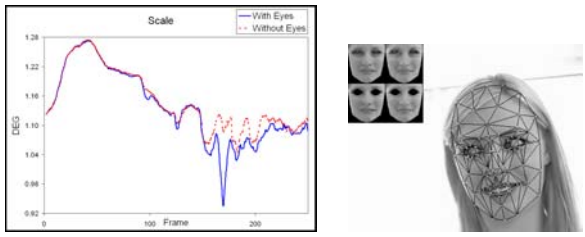

Figure 3. Scale difference with two textures. Textures with eyes and without eyes.

pixels filling the patch for the statistical model, μ and χ, the eyes region is represented by only 380 pixels in the image warped version (7% of the face).

In order to achieve a good eyelids tracking, robust and stable, we begin evaluating the dependence between those two regions, but in opposite direction. Can the eyes region be represented by an independent ABM?

To answer this question, we compare ABM (Fig. 4) for each region. The ABM for the eyes region will only estimate the eyes parameter γ^6 and another ABM will estimate the head, eyebrows and lips parameters $[\alpha^0, ..., \alpha^5, \gamma^0, ..., \gamma^5]^t$. The OAMs follow the same mathematic models and recursive filtering explained above.

The experimental tests have shown us that an ABM for the eyes region is very memory-inexpensive but is time-consuming in the search stage. Moreover, it is hard to obtain estimation from this appearance model. We conclude that the eyes tracking requires the face remains in order to build an efficient appearance model.

Following the two ABMs approaches we built two generic appearances, as:

- ABM 1: taking into account the 13 parameters of the Eq. (6, 7, 8), that means that the estimation result will be 13 dimensional, **q**=$[\alpha^i, \gamma^j]$, for $i = 0, ..., 5$ and $j = 0, ..., 6$.
- ABM 2: avoiding the eyes region and working only with 12 parameters, **q**=$[\alpha^i, \gamma^j]$, for $i = 0, ..., 5$ and $j = 0, ..., 5$.

Tracking Problem

ABM 1:

- Given an input image $\mathbf{I}_t$, the warped image version χ is obtained to deal with texture model of 84 x 80 resolution including the eyes region.
- Like in section 3, we construct here the ABM, in which the appearance A_t follows a Gaussian model.

$$p(\mathbf{I}_t|\mathbf{q}_t) = p(\chi_t|\mathbf{q}_t) = \prod N(\chi_i; \mu_i, \sigma_i) \quad (20)$$

$$\mu_{t+1} = \omega\mu_t + (1-\omega)\hat{\chi}_t \quad (21)$$

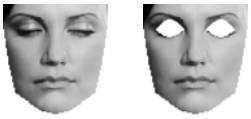

Figure 4. Ground Textures, χ for ABM1, and χ' for ABM2

$$\sigma^2_{t+1} = \omega\sigma^2_t + (1-\omega)(\hat{\chi}_t - \mu_t)^2 \qquad (22)$$

- A different learning parameter is used in the recursive filtering technique, ω; the objective is to include a lot of information from the current frame to the probabilistic model, because blinking is a quick facial action.
- The Jacobian estimation for gradient descent, is estimated out. The Jacobian components for $[\alpha^0, ..., \alpha^5, \gamma^0, ..., \gamma^5]$, are computed by partial differences regarding an increment step like δ and a small perturbation range. On other hand, the eyelids component, γ^6, is estimated by using a descent step like $5 * \delta$ and a perturbation range for overall FAP [3]; this difference arises from fast eyelids motion.
- Once the patch estimation is given, search process is started to look for the best adaptation. Here we use a Mahalanobis distance to compare the face warped in the face space following the algorithm explained in Section 3. Comparing the two ABMs, a backward-forward search factor is included, which adds exploitation to the algorithm and reduces exploration in relation to other parameters. Simultaneous convergence is achieved for all parameters. Local minimums are avoided and the FAP is ran completely.

  ABM 2:

- Like before, we build the image warped version χ' of the input image $\mathbf{I}_t$ using a ground texture without the eyes region. χ' is different to χ in regards to the eye region, they are two different textures.

  This appearance model is Fig. 4**(b)**.

$$p(\mathbf{I}_t|\mathbf{q}_t) = p(\chi'_t|\mathbf{q}_t) = \prod N(\chi'_i; \mu'_i, \sigma'_i) \qquad (23)$$

$$\mu'_{t+1} = \beta\mu'_t + (1-\beta)\hat{\chi'}_t \qquad (24)$$

$$\sigma'^2_{t+1} = \beta\sigma'^2_t + (1-\beta)(\hat{\chi'}_t - \mu'_t)^2 \qquad (25)$$

- Using the recursive filtering with a learning factor $\beta < \omega$, we keep more information from previous frames, even more information than from ABM1. This appearance learns quickly the texture model with the skill to handle outliers.

[3] facial action parameter account between -1 to 1.

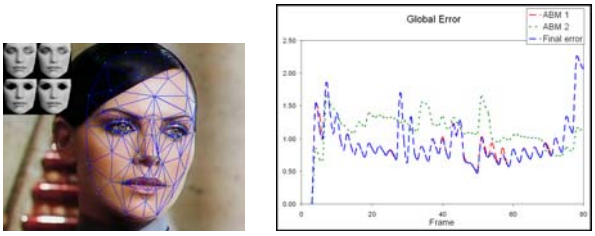

Figure 5. Input image and correct adaptation mesh. Two ABMs are at Left Top, the estimated and current textures for each model. Corresponding global errors for two ABMs.

- The Jacobian for the descent gradient is like the first one, only for $[\alpha^i, \gamma^0, ..., \gamma^5]$, when $i = 0, ..., 5$
- Given the patch estimation, we measure the distance and perform the search stage until the convergence is achieved.

In Fig. (5), we can see a notable difference between appearances. At the left top of the frame the estimated and current texture are shown for each of the models, the two upper patches of *ABM 1* and the two lower patches of *ABM 2*.

5 Experimental Results

We have used different image sequences for the experimentation and stabilization of the tracking, avoiding images from laboratory videos. Instead, we use compressed videos, or published in web format in order to track natural and unexpected movements and large angle variation; where scale variations are related to in-depth movements or camera zooming. The difficult task for the tracker was to work with low-resolution images, absent original frames, with only some illumination. To work with high resolution images, where the accuracy is tested. The output images are shown and explained.

1. Fig. 6 is 330-frames (200 x 260 pixels in the input face) captured as screener, thus the image resolution is low and even more in the eyes region.
2. Fig. 7 is an 82-image sequence that has been extracted from *Aeon Flux* movie[4](280 x 360 pixels in the input face). The difficulty with this sequence is its sort size, the static camera-plane is very short and the face movements are fast. The result proves the fast learning of the two trackers in real time.

The tracking experiments in a 3.2 GHz Pentium PC, without an optimized C code, has a performance of 500 MB as RAM memory-consume and 60 ms as average for each appearance model, using a big mask (84 x 80). All tests have reported false-positive estimation.

[4] ©Paramount Pictures and Likeshore Entertainment.

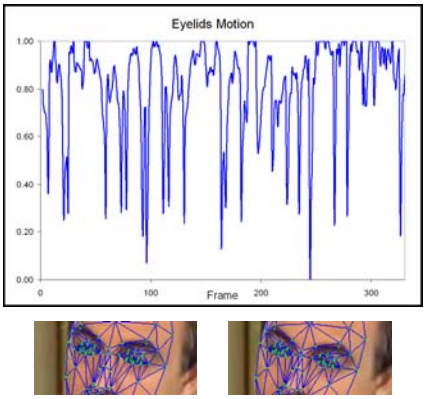

Figure 6. Plot with the tracking results. Zoom of 109 and 234 frames from a 330-image sequence.

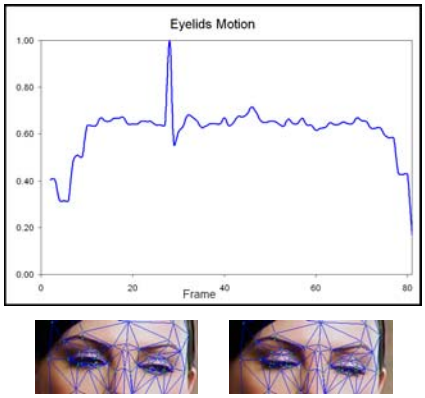

Figure 7. We show a plot with the tracking results. Zoom of the 2 and 81 frames, where the eyes status is near to close.

6 Conclusions

In this framework, we have extended the application of AAMs to perform a continuous eyelids tracking, we have proven the capability of estimating the 3D head pose, eyebrows, lips and eyelids with accuracy and in real-time, through the Gaussian model for the appearance and with help of a technique to analyze the descent in multidimensional spaces; we use a Gauss-Newton method.

The last 2-th and 3-th sequences (Fig. 6 and 7), like in human vision, the eyelids adaptation give an eyes state, which is closer in the second sequence than in the third one, due to the sort distance between the face and the camera.

Comparing this approach with other works in the same frame, our work has made important contributions. First, we have blended the 3D head pose, brows, lips and eyelids tracking, using deformable models and on-line appearance models. Second, we have avoided using any color information of the image regarding color spaces or edge detectors. We have recovered the mesh parameters using statistics and estimation finite series tools. Third, we have proposed a hierarchical tracking with its respective face space to refine the estimations related to the input image, to allow using small and big ground textures to reduce memory and time complexity.

References

[1] D.J. Fleet A.D. Jepson and T.F. El-Maraghi. Robust online appearance models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1296–1311, 2003.

[2] J. Ahlberg. "candide-3 - an updated parameterized face". *Tech. Rep. LiTH-ISY-R-2326, department of Electrical Engineering*, 2001.

[3] J. Ahlberg. An active model for facial feature tracking. *EURASIP Journal on Applied Signal Processing*, 2002(6):566–571, 2002.

[4] J. Deng and F. Lai. Region-based template deformation and masking for eye-feature extraction and description. *Pattern Recognition*, 30(3):403–419, 1997.

[5] H. Zha H. Liu, Y. Wu. Eye states detection from color facial image sequence. *In:SPIE International Conference on Image and Graphics*, 4875(2002):693–698, 2002.

[6] S. Sclaroff M.L. Cascia and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4):322–336, 2000.

[7] T. Moriyama and T. Kanade et al. Automatic recognition of eye blinking in spontaneosly ocurring behavior. *In International Conference on Pattern Recognition*, 2002.

[8] J. Nocedal and S. Wright. Numerical optimization. In *Springer, New York*, 1999.

[9] N. Tetsutani S. Kawato. Detection and tracking of eyes for gaze-camera control. *Image and Vision Computing*, 2004(22):10311038, 2004.

[10] A. Rosenfeld S. Sirhey and Z. Duric. A method of detecting and tracking irises and eyelids in video. *In International Conference on Pattern Recognition*, 35(6):1389–1401, 2002.

[11] J.F. Cohn Y. Tian, T. Kanade. Dual-state parametric eye tracking. *In: International Conference on Automatic Face and Gesture Recognition*, 2000.