Online Appearance-based Face and Facial Feature Tracking

F. Dornaika and F. Davoine

CNRS HEUDIASYC
Compiegne University of Technology
60205 Compiegne Cedex, FRANCE
{dornaika, fdavoine}@hds.utc.fr

Abstract

We propose a simple framework that utilizes online appearance models for 3D face and facial feature tracking with a deformable model. Adapting the geometrical parameters for each frame adopts a steepest ascent method in the observation likelihood using a local exhaustive and directed search in the parameter space. The observation likelihood is based on the current appearance and the registered images. The developed framework is straightforward and has the following advantages. First, it does not require any a priori statistical facial texture. Second, it does not require any a priori transition model for the 3D motion. Video sequences featuring large head motions, large facial animations, and external illumination variations are successfully tracked, which demonstrate the efficiency of the developed framework.

1. Introduction

Object tracking is required by many vision applications, especially in video technology and visual interface systems. The ability to track facial motion is useful in applications such as face-based biometric person authentication, expression analysis, and human computer interaction. Detecting and tracking faces in video sequences is a challenging task because faces are non-rigid and their images have a high degree of variability. A huge research effort has already been devoted to detecting and tracking of facial features in 2D and 3D (e.g., [2, 3, 5, 6]).

To solve the tracking problem, one may use feature-based approaches which are utilized in 3D vision. These approaches proceed as follows. First, a set of facial features is extracted from the initial frame automatically or manually. Then, a motion-based tracker attempts to locate (match) them in the subsequent frame. An optimization process is then carried out to recover the 3D pose and the possible facial animations. However, such trackers very often suffer from the drifting problem (error accumulation) since facial features do not have enough stable local appearance due to many factors. To overcome the problem of appearance changes, recent works on faces adopted statistical facial textures. For example, the Active Appearance Models have been proposed as a powerful tool for analyzing facial images [4, 8]. In [1], the author proposed a framework that tracks face and facial features using Active Appearance Models. While statistical appearance-based tracking methods are promising with respect to some aspects, they are depending on the imaging conditions under which the learning is performed. Thus, by changing these conditions, one should repeat the whole learning process. Also, they are related to a specific class of objects.

Recently, 2D tracking approaches have adopted on line appearance models [7, 9]. In [9], a deterministic and stochastic approach was developed to track the 2D motion of faces using an affine transform. In this paper, we propose a simple and efficient tracking framework having the following advantages. First, the framework does not require any *a priori* statistical facial texture. Second, it does not require any *a priori* transition model for the 3D motion. Third, the approach simultaneously tracks the head 3D pose and the facial animations. The rest of the paper is organized as follows. Section 2 describes the deformable 3D face model. Section 3 states the tracking problem. Section 4 describes the utilized appearance and observation likelihood. Section 5 describes the search algorithm utilizing a combined exhaustive and directed search. Section 6 presents some experimental results.

2. Modelling faces

2.1. A deformable 3D model

Building a generic 3D face model is a challenging task. Indeed, such a model should account for the differences be-

0-7695-2128-2/04 $20.00 (C) 2004 IEEE

tween different specific human faces as well as between different facial expressions. This modelling was explored in the computer graphics, computer vision, and model-based image coding communities. In our study, we use the 3D face model *Candide*. This 3D deformable wireframe model was first developed for the purpose of model-based image coding and computer animation. The 3D shape of this deformable 3D wireframe model is directly recorded in coordinate form. The 3D face model is given by the 3D coordinates of the vertices $\mathbf{P}_i, i = 1, \ldots, n$ where n is the number of vertices. Thus, the shape up to a global scale can be fully described by the $3n$-vector $\mathbf{g}$ – the concatenation of the 3D coordinates of all vertices $\mathbf{P}_i$. The vector $\mathbf{g}$ can be written as:

$$\mathbf{g} = \overline{\mathbf{g}} + \mathbf{S}\,\sigma + \mathbf{A}\,\alpha \tag{1}$$

where $\overline{\mathbf{g}}$ is the standard shape of the model, and the columns of $\mathbf{S}$ and $\mathbf{A}$ are the Shape and Animation Units, respectively. A Shape Unit provides a way to deform the 3D wireframe such as to adapt the eye width, the head width, the eye separation distance etc. Thus, the term $\mathbf{S}\,\sigma$ accounts for shape variability (inter-person variability) while the term $\mathbf{A}\,\alpha$ accounts for the facial animation (intra-person variability). The shape and animation variabilities can be approximated well enough for practical purposes by this linear relation. Also, we assume that the two kinds of variability are independent.

In this study, we use 12 modes for the Shape Units matrix and six modes for the Animation Units matrix. Without loss of generality, we have chosen the following Action Units: 1) Jaw drop, 2) Lip stretcher, 3) Lip corner depressor, 4) Upper lip raiser, 5) Eyebrow lowerer, 6) Outer eyebrow raiser. These Action Units are enough to cover most common facial actions (mouth and eyebrow movements).

In Equation (1), the 3D shape is expressed in a local coordinate system. However, one should relate the 3D coordinates to the image coordinate system. To this end, we adopt the weak perspective projection model. We neglect the perspective effects since the depth variation of the face can be considered as as small compared to its absolute depth. Therefore, the mapping between the 3D face model and the image is given by a 2×4 matrix, $\mathbf{M}$, encapsulating both the head 3D pose and the camera parameters.

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

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

For a given person, σ is constant. Estimating the vector σ can be carried out using either feature-based or featureless approaches. Therefore, the state of the 3D model is given by the 3D head pose (three rotations and three translations) and the control vector α. This is given by the vector $\mathbf{b}$:

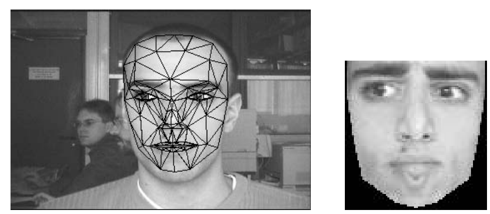
$$\mathbf{b} = [\theta_x, \ \theta_y, \ \theta_z, \ \lambda\,t_x, \ \lambda\,t_y, \ \lambda\,t_z, \ \alpha^T \ ]^T \tag{3}$$

2.2. Shape-free facial images

A face texture is represented as a shape-free texture (geometrically normalized image). The geometry of this image is obtained by projecting the standard shape $\overline{\mathbf{g}}$ (wireframe) 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 the input image using a piece-wise affine transform, $\mathcal{W}$. Mathematically, the warping process applied to an input image $\mathbf{y}$ is denoted by:

$$\mathbf{x}(\mathbf{b}) = \mathcal{W}(\mathbf{y}, \mathbf{b}) \tag{4}$$

where $\mathbf{x}$ denotes the shape-free texture and $\mathbf{b}$ denotes the geometrical parameters. Here images are represented by one-dimensional vectors. Figure 1 illustrates the warping process applied to an input image. Two resolution levels have been used for the shape-free textures, encoded by 1300 and 5392 pixels.



(a) **(b)**

Figure 1. **(a)** An input image with correct adaptation. **(b)** The corresponding shape-free image (5392 pixels).

3. The tracking problem

Given a video sequence depicting a moving face, the tracking consists of estimating (for each frame) the 3D pose of the face as well as the facial animations encoded by the control vector α. In other words, one would like to estimate the vector $\mathbf{b}_t$ (Eq.(3)) for each frame t. In a tracking context, the model parameters associated with the current frame will be handed over to the next frame.

4. Appearance-based observation likelihood

In order to recover the model parameters, i.e. the vector $\mathbf{b}$, an observation likelihood should be available. This likelihood quantifies the consistence of the hypothetical model parameters with the observation made at the same instant of time. This observation likelihood is denoted by $p(\mathbf{y}_t|\mathbf{b}_t)$. Any function p should rely on some texture model with which the observation $\mathbf{y}_t$ is compared.

Obviously, statistical texture models can be used in the observation likelihood. For example, an active appearance model search utilizes *eigenfaces* as a texture model [1]. In our case, however, we seek simplicity and flexibility, that is, there is no prior statistical texture model. Therefore our idea is to exploit the video sequence itself, that is, the tracking and the learning processes are running in tandem. For this purpose, an appearance $\mathbf{A}_t$ is built on-line from previous shape-free textures. This appearance should be able to explain the stable observations and allow the recovery of the geometric model.

For an input image $\mathbf{y}_t$ and a hypothetical model $\mathbf{b}$, the observation likelihood is given by:

$$p(\mathbf{y}_t|\mathbf{b}) \cong < \mathbf{x}_t(\mathbf{b}), \mathbf{A}_t > \quad (5)$$

where $<,>$ is the zero-mean normalized cross correlation, $\mathbf{A}_t$ is the current appearance, and $\mathbf{x}_t(\mathbf{b})$ is the warped version of the input image $\mathbf{y}_t$ using the geometric parameters $\mathbf{b}_t$ (Eq. (4)).

For the current image $\mathbf{y}_t$, the optimal vector $\mathbf{b}_t^*$ under the ML (Maximum Likelihood) framework is given by:

$$\mathbf{b}_t^* = \arg\max_{\mathbf{b}}(p(\mathbf{y}_t|\mathbf{b})) = \arg\max_{\mathbf{b}}(< \mathbf{x}_t(\mathbf{b}), \mathbf{A}_t >) \quad (6)$$

5. Combined exhaustive and directed search

Obviously, there is no closed-form solution to Eq.(6). We approach this optimization problem using a combined exhaustive and directed search in the parameter space. Without loss of generality, we assume that the dimension of the vector $\mathbf{b}$ is 12. The optimization consists of two successive stages which are invoked for each frame in the video sequence:

1. *Exploration stage.* At each iteration $l = 1, 2, \cdots, L$, the locally best parameter, $b_j^{[l]}$, is chosen by changing each parameter $i \in \{1, \cdots, 12\}$ under fixed values of the remaining parameters, $[b_k^{[l-1]} : k \neq i, ; k \in \{1, \cdots, 12\}]$. The choice yields the largest increase in the likelihood measure (Eq. (5)) providing that the parameters differ by only the value of the locally best parameter $b_j^{[l]}$. The exploration is repeated while the likelihood increases further. For each iteration and for each parameter, the exploration locally exhausts a given number of the equispaced parameter values within the range $[b_i^{[l-1]} - \Delta_i/2, b_i^{[l-1]} + \Delta_i/2]$. The components of $\mathbf{b}^{[0]}$ are set to the values computed at the previous frame.

2. *Search stage (refinement stage).* The exploration converges to a final maximum value $p(\mathbf{y}_t|\mathbf{b}^{[L]})$, and the vector $\mathbf{b}^{[L]}$ allows for inferring the possible steepest ascent direction in the parameter space. The search along this direction refines further the obtained parameters $\mathbf{b}_\mu = \mathbf{b}^{[L]} + \mu\,(\mathbf{b}^{[L]} - \mathbf{b}^{[0]})$. Once the model geometry $\mathbf{b}_t^* = \mathbf{b}_\mu$ is found, it is handed over to the next frame.

In our implementation, Δ_i is set to 12 degrees for the rotation angles and to 1/40*th* of the image size for the 2D translations. The number of equispaced steps is set to 20. The refinement step is set to one hundredth of the distance between the initial solution $\mathbf{b}^{[0]}$ and the rough solution $\mathbf{b}^{[T]}$.

To gain computational efficiency, we can set an upper bound L_{max} for the number of iterations. Furthermore, the exhaustive exploration stage adopts a coarse-to-fine scheme for the equispaced values.

To update the current appearance model $\mathbf{A}_t$ to $\mathbf{A}_{t+1}$ after $\mathbf{x}(\mathbf{b}_t^*)$ becomes available, we assume the updating weights on the current observation, $\mathbf{x}(\mathbf{b}_t^*)$, and the appearance model $\mathbf{A}_t$ are λ and $(1-\lambda)$ respectively, the appearance $\mathbf{A}_t$ can be updated as

$$\mathbf{A}_{t+1} = \lambda\,\mathbf{x}(\mathbf{b}_t^*) + (1-\lambda)\,\mathbf{A}_t \quad (7)$$

The update rate λ can be chosen experimentally like most incremental algorithms. With this updating scheme, the old information stored in the model decays exponentially over time. It is worthwhile noting that using the byproduct of the above framework, i.e. the sequence of textures $\mathbf{x}(\mathbf{b}_t^*)$, one can build an eigenface system associated with the tracked face by means of batch or incremental Principal Component Analysis.

6. Experimental results

Figure 2 displays the adaptation results associated with eight frames of a 750-frame-long sequence featuring quite large pose variations as well as large facial animations. The sequence is of resolution 720x480 pixels. As can be seen with the very little prior information, the 3D motion of the face as well as the facial actions associated with the mouth and the eyebrows are accurately recovered. The upper left corner shows the current appearance as well as the shape-free texture. Also, one can notice that the mouth animations correspond well to the actual ones despite that the tracker has no prior knowledge neither on the texture appearance nor on the transition motions of such animations. Figure 3 displays the estimated value of the yaw angle, the vertical translation, the lip stretcher, and the brow raiser as a function of the frames of the sequence.

Figure 4 displays the adaptation results associated with two video sequences. The girl sequence features external illumination variations obtained with a moving light source just above the camera.

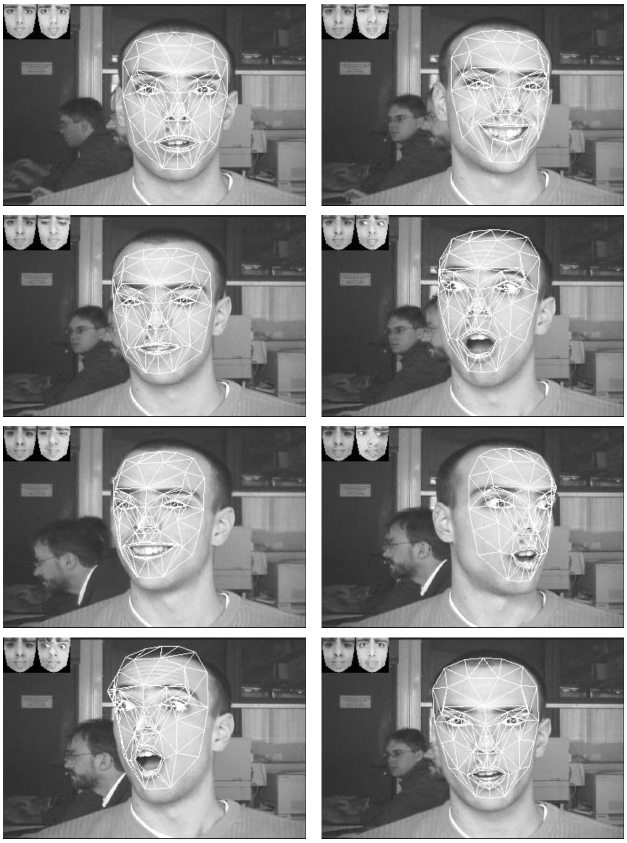


Figure 2. Face and facial tracking results. Frames 63, 181, 222, 282, 418, 492, 522, and 683, 720x480 pixels in size, in a 750-frame-long sequence. The upper left corner shows the current appearance $\mathbf{A}_t$ and the shape-free texture.

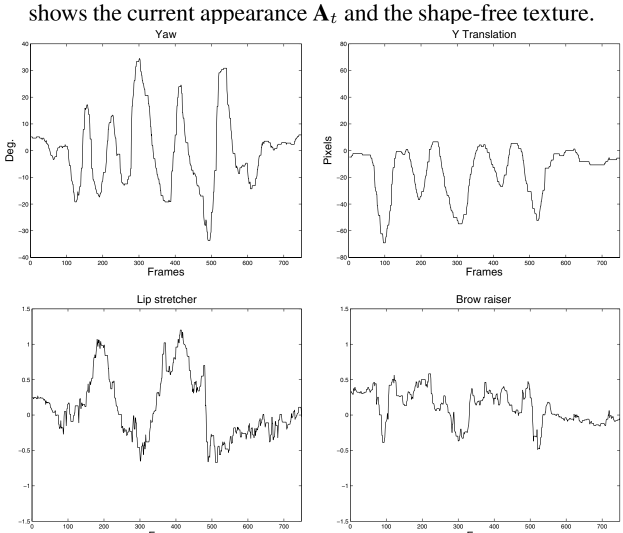


Figure 3. The yaw angle (large value for frame 282), the vertical translation, the lip stretcher (large values for frames 181 and 418), and the brow raiser (large value for frame 522).

Figure 4. Face and facial feature tracking results associated with two sequences. The second sequence features illumination variations.

References

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

[2] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[3] M.L. Cascia, S. Sclaroff, 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.

[4] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–684, 2001.

[5] S.B. Gokturk, J.Y. Bouguet, and R. Grzeszczuk. A data-driven model for monocular face tracking. In *Proc. IEEE International Conference on Computer Vision*, 2001.

[6] T.S. Jebara and A. Pentland. Parameterized structure from motion for 3D adaptative feedback tracking of faces. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

[7] A.D. Jepson, D.J. Fleet, 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.

[8] I. Matthews and S. Baker. Active appearance models revisited. Technical Report CMU-RI-TR-03-02, The Robotics Institute, Carnegie Mellon University, 2002.

[9] S. Zhou, R. Chellappa, and B. Mogghaddam. Adaptive visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Transactions on Image Processing*, To appear 2004.

0-7695-2128-2/04 $20.00 (C) 2004 IEEE