# Performance driven facial animation by appearance based tracking

José Miguel Buenaposada[1], Enrique Muñoz[2], and Luis Baumela[2]

[1] Dpto. de Informática, Estadística y Telemática,
ESCET, Univ. Rey Juan Carlos,
C/ Tulipán, s/n, 28933 Móstoles, Madrid, Spain.
jmbuenaposada@escet.urjc.es
[2] Fac. de Informática, Univ. Politécnica de Madrid,
Campus de Montegancedo s/n, 28660 Boadilla del Monte, Madrid, Spain.
kike@dia.fi.upm.es, lbaumela@fi.upm.es

**Abstract.** We present a method that estimates high level animation parameters (muscle contractions, eye movements, eye lids opening, jaw motion and lips contractions) from a marker-less face image sequence. We use an efficient appearance-based tracker to stabilise images of upper (eyes and eyebrows) and lower (mouth) face. By using a set of stabilised images with known animation parameters, we can learn a re-animation matrix that allows us to estimate the parameters of a new image. The system is able to re-animate a 32 DOF 3D face model in real-time.

## 1 Introduction

Automated computer animation of faces and avatars is an area of intense research for its application in the television, computer games and film industry. Performance driven animation is usually done by motion capture using markers on the face. Computer vision provides an alternative non-intrusive marker-less approach to motion capture.

Generally, the face shapes of the actor and that of the animated model are different. So, a method to adapt the motion of the former to the latter is needed [1]. There are two ways to achieve this: parametrisation and motion modification. By facial motion modification we mean to adapt the vertex deformation due to facial motion to the new facial model. In [1] were introduced some algorithms and heuristics to translate the facial expression motion from a facial model into another with different surface structure. Procedures based on parametrisation aim to describe motion with a set of values that, when applied to any facial model, will produce a similar expression. Among the parametrised systems we can distinguish those that use standard facial expressions coding, like FACS[2, 3] or MPEG-4 FAPS [4, 5], and those that use and *ad-hoc* coding [6, 7]. When the abstraction level of the animation parameters is high, then the estimation of these parameters is more difficult. This is due mainly to the weak relationship between image measurements and control parameters.

In this paper we present a method that estimates high level animation parameters from a marker-less face image sequence. We will use a muscle-based 3D face model resulting in a parametrised motion capture algorithm. We have previously developed an efficient appearance based tracker [8] that locates and tracks the eyes and the mouth in spite of the non-rigid motion of the face. The main contribution of this paper is a procedure to estimate the animation parameters of a 3D face model from stabilised images of the eyes and mouth obtained from our tracking algorithm. This procedure is composed of two training steps, one for building an eigenspace for tracking, and another one for learning a linear relation between the animation parameters and the stabilised images. In the following sections we will present this algorithm and some results.

## 2 Appearance based tracking

The tracking algorithm presented in this section can be seen as an extension of the Hager and Belhumeur's *Jacobian factorisation* [9] where we impose no restrictions on the PCA-based subspace model used. It is also related to the Black and Jepson's *Eigentracking* [10], but instead of computing the motion parameters by using a gradient descent procedure in which the target image Jacobian must be computed for each frame in the sequence, as in [10], we use a set of precomputed motion templates which alleviate the computations that have to be performed on line.

Let $P$ be the image of a target. The subspace constancy equation holds for all pixels in the target [10]:

$$I(f(\mathbf{x},\boldsymbol{\mu}),t) = [\mathtt{B}\mathbf{c}(t)](\mathbf{x}) \quad \forall x \in P, \tag{1}$$

where $\mathbf{x}$ is the vector of co-ordinates of a point in image $I$, $\mathtt{B}$ is the subspace base matrix, $\mathbf{c}$ is the vector of subspace coefficients, and $I(f(\mathbf{x},\boldsymbol{\mu}),t)$ is the image acquired at time $t$ rectified with motion model $f(\mathbf{x},\boldsymbol{\mu})$ and motion parameters $\boldsymbol{\mu}$. By $[\mathtt{B}\mathbf{c}](x)$ we denote the value of $\mathtt{B}\mathbf{c}$ for the pixel with position $\mathbf{x}$ in the image. Matrix $\mathtt{B}$ is of dimension $N \times k$, where $N$ is the number of pixels per image and $k$ is the number of basis vectors in the subspace. Intuitively (1) states that the rigidly rectified image $I(f(\mathbf{x},\boldsymbol{\mu}),t)$ can be expressed as a linear combination of the appearance subspace basis vectors, $\mathtt{B}$[3].

Tracking consists on estimating for each image in the sequence the values of the motion, $\boldsymbol{\mu}$, and appearance, $\mathbf{c}$, parameters which minimise the error function

$$E(\boldsymbol{\mu},\mathbf{c}) = ||\mathbf{I}(f(\mathbf{x},\boldsymbol{\mu}_t),t) - \mathtt{B}\mathbf{c}(t)||^2,$$

where $\mathbf{I}(\mathbf{x})$ is $I(\mathbf{x})$ in vector form (scanning $I$ by rows or columns). In order to make Gauss-Newton iterations, a Taylor series expansion of $\mathbf{I}$ at $(\mathbf{x},t)$ is performed, producing a new error function

$$E(\delta\boldsymbol{\mu},\mathbf{c}) = ||\mathtt{M}\delta\boldsymbol{\mu} + \mathbf{I}(f(\mathbf{x},\boldsymbol{\mu})) - \mathtt{B}\mathbf{c}||^2,$$

[3] We assume that that the average image has been included as the first column of $\mathtt{B}$.

where $\mathtt{M} = \frac{\partial \mathbf{I}(f(\mathbf{x},\boldsymbol{\mu}))}{\partial \boldsymbol{\mu}}$ is the $N \times n$ ($n = \dim(\boldsymbol{\mu})$) Jacobian matrix of $I$ (note that dependence on $t$ has been dropped for convenience). In the following subsections we will introduce a procedure for precomputing a set of motion templates which efficiently minimise (2) for any linear subspace model.

### 2.1 Jacobian matrix factorisation

One of the obstacles for minimising (2) on line, while tracking, is the computational cost of estimating $\mathtt{M}$ for each frame. Following an approach similar to [9], $\mathtt{M}$ can be expressed in terms of the gradient of the subspace basis vectors, $\mathtt{B}_\nabla$, which are constant, and the motion and appearance parameters $(\boldsymbol{\mu}, \mathbf{c})$, which vary over time. If we choose a motion model $f$ such that $\mathtt{C} f_{\mathbf{x}}(\mathbf{x}_i, \boldsymbol{\mu})^{-1} f_{\boldsymbol{\mu}}(\mathbf{x}_i, \boldsymbol{\mu}) = \Gamma(\mathbf{x}_i) \boldsymbol{\Sigma}(\boldsymbol{\mu}, \mathbf{c})$, then $\mathtt{M}$ can be factored into

$$\mathtt{M}(\boldsymbol{\mu}, \mathbf{c}) = \begin{bmatrix} \mathtt{B}_\nabla(\mathbf{x}_1)\Gamma(\mathbf{x}_1) \\ \vdots \\ \mathtt{B}_\nabla(\mathbf{x}_N)\Gamma(\mathbf{x}_N) \end{bmatrix} \Sigma(\boldsymbol{\mu}, \mathbf{c}) = \mathtt{M}_0 \Sigma(\boldsymbol{\mu}, \mathbf{c}),$$

where $\mathtt{B}_\nabla(\mathbf{x}_i)$ is the Jacobian of $\mathtt{B}$ with respect to the image co-ordinates. Then $\mathtt{M}_0$ is a constant matrix and $\Sigma$ depends on $\mathbf{c}$ and $\boldsymbol{\mu}$.

### 2.2 Minimising $E(\boldsymbol{\mu}, \mathbf{c})$.

As $\mathtt{M}$ depends on both, $\boldsymbol{\mu}$ and $\mathbf{c}$, (2) defines a nonlinear cost function over $\delta\boldsymbol{\mu}$ and $\mathbf{c}$. The optimisation algorithm that we use first assumes $\mathbf{c}$ constant and computes the minimum of $E(\boldsymbol{\mu}, \mathbf{c})$ w.r.t. $\boldsymbol{\mu}$,

$$\delta\boldsymbol{\mu} = -(\Sigma^\top \mathcal{M} \Sigma)^{-1} \Sigma^\top \mathtt{M}_0^\top [\mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}), t + \tau) - \mathtt{B}\mathbf{c}(t)],$$

where $\mathcal{M} = \mathtt{M}_0^\top \mathtt{M}_0$. Then it minimises $E$ over $\mathbf{c}$ assuming $\boldsymbol{\mu}$ constant,

$$\mathbf{c} = \mathtt{B}^\top [\mathtt{M}\delta\boldsymbol{\mu} + \mathbf{I}(f(\mathbf{x}, \boldsymbol{\mu}), t + \tau)].$$

Once we have $\mathbf{c}$, we can refine the estimation of $\delta\boldsymbol{\mu}$ by using (2.2) again. Normally two or three iterations are enough to reach a stable solution. We have developed the factorisation for the rotation-translation-scale, the affine and the projective motion models [8]. In this paper we will use a projective motion model, $f(\mathbf{x}, \boldsymbol{\mu}) = \mathtt{H}\mathbf{x}$, where $\mathtt{H}$ is a $3 \times 3$ homography.

## 3 Reanimation

The philosophy to performance driven animation of a 3D face model we propose, is similar to the Valente and Dugelay's one [4]. We will use stabilised view images of the user's eyes and mouth with known animation parameters to estimate a linear relationship between grey levels and animation parameters. In order to estimate the control parameters of their face model, Valente and Dugelay use optical flow and not raw grey levels as we do. They use a very realistic 3D face model of each particular user. Therefore, by driving their model with a set of control parameters it was possible to get the corresponding optical flow for each face region. Valente and Dugelay use a feature based tracker (five features) and a Kalman filter to get the normalised images of different face regions. As their tracker is not designed to deal with non-rigid motion, it is not clear how is it going to work with extreme facial expressions.

In our case, the appearance based tracker of section 2 allows us to track the most informative face areas in spite of the non-rigid motion due to facial expressions. With the tracker we can extract stabilised images of any part of the face for each frame in the sequence. In this section we are going to show how to estimate the face animation parameters from stabilised images of the lower and the upper part of the face.

### 3.1 Animation parameters estimation

In order to estimate the animation parameters for a given face region we will use $e$ example images each with $N$ pixels. Let $\mathtt{I}$ be an $N \times e$ matrix, where each column $\mathbf{i}_j$ has one of the example images (e.g. scanning the image by rows), and let $\mathtt{A}$ be an $a \times e$ matrix, where each column $\mathbf{a}_j$ represents the animation parameters, $\mathbf{a}$, corresponding to the appearance in $\mathbf{i}_j$ [4]. Then $\mathtt{D}_e$ is an $(N+a) \times e$ matrix:

$$\mathtt{D}_e = \begin{bmatrix} \mathtt{I} \\ \mathtt{W}_A\mathtt{A} \end{bmatrix} = \begin{bmatrix} \mathbf{i}_1 & \cdots & \mathbf{i}_e \\ \mathtt{W}_A[\mathbf{a}_1 & \cdots & \mathbf{a}_e] \end{bmatrix}, \qquad (2)$$

where $\mathtt{W}_A$ is a diagonal matrix of weights that takes into account the different scale of the animation parameters and grey levels. The weight matrix we use, is $r\mathtt{I}$ where $r^2$ is the rate between the grey levels variability and total variability in the animation parameters. In the Direct Appearance Models framework [11] it is used a similar matrix but for grey levels and shape parameters.

By computing PCA of matrix $\mathtt{D}_e$, we get $\mathtt{B}_l$, the subspace basis expanded by the $l$ eigenvectors[5] corresponding to the bigger eigenvalues of the covariance matrix $(\mathtt{D}_e\mathtt{D}_e^\top)$, which can be written as

$$\mathtt{B}_l = \begin{bmatrix} \mathtt{B}_i \\ \mathtt{B}_a \end{bmatrix}.$$

Using the $(N+a) \times l$ matrix, $\mathtt{B}_l$, the vector $\mathbf{c}_l$, that represents the relation between the images in $\mathtt{I}$ and the animation parameters in $\mathtt{A}$ can be estimated. By using $\mathbf{c}_l$, we can approximate each pair $(\mathbf{i}, \mathbf{a})$ by $(\mathbf{i}^*,\mathbf{a}^*)$ in such a way that:

$$\begin{bmatrix} \mathbf{i}^* \\ \mathtt{W}_A\mathbf{a}^* \end{bmatrix} = \mathtt{B}_l\mathbf{c}_l, \mathbf{c}_l = \mathtt{B}_l^\top \begin{bmatrix} \mathbf{i} \\ \mathtt{W}_A\mathbf{a} \end{bmatrix}.$$

[4] We assume, that all examples, $\mathbf{i}_j$, and animation parameters, $\mathbf{a}_j$, are mean centred.

[5] Note that we use two eigenspaces, one for tracking and the other for reanimation.

Given an image $\mathbf{i}$, and $\mathtt{B}_i$ and $\mathtt{B}_a$ matrices from training, the re-animation problem is to estimate the corresponding animation parameters, $\mathbf{a}^*$. From the structure of $\mathtt{B}_l$ we can write $\mathtt{B}_i\mathbf{c}_l = \mathbf{i}$, where $\mathbf{c}_l$ is the only unknown. In general, the number of image pixels $N$ is much bigger than $l$ and the solution for $c_l$ will be given by the minimisation of

$$\mathbf{c}_l^* = arg \min_{\mathbf{c}_l} ||\mathtt{B}_i\mathbf{c}_l - \mathbf{i}||^2 = pinv(\mathtt{B}_i)\mathbf{i}, \tag{3}$$

where the $l \times N$ matrix $pinv(\mathtt{B}_i)$, is the pseudo-inverse of $\mathtt{B}_i$ computed by using SVD. And then, the animation parameters that corresponds to the image $\mathbf{i}$ are given by

$$\mathtt{W}_A\mathbf{a}^* = \mathtt{B}_a pinv(\mathtt{B}_i)\mathbf{i} = \mathtt{R}_i^a\ \mathbf{i}, \tag{4}$$

where the $a \times N$ matrix, $\mathtt{R}_i^a$, is constant and can be precomputed. As we get $\mathtt{W}_A\mathbf{a}^*$ from (4), it is needed to multiply it by $(\mathtt{W}_A)^{-1}$ in order to obtain the animation parameters estimation, $\mathbf{a}^*$, in the right scale.

## 4 Experiments

In all the experiments conducted[6] in this section the face is splited in the upper face (the eyes region) and the lower face (mouth region) areas. As the motion of the two regions is almost independent we can build two appearance models needing less examples on each (a modular eigenspace). Nevertheless, our tracker uses the grey levels from both regions to compute motion parameters but maintaining separate appearance parameters.

### 4.1 Quantitative experiments

In the first experiment we would like to assert the quality of the re-animation. To do so, we use a modified version of the Parke and Waters' 3D face model [12] with 32 degrees of freedom. The 3D face model is used to render three image sequences: a training sequence for the eyes (630 images), a training sequence for the mouth (540 images) and a test sequence (1225 images, see figure 1). The facial expressions in the test sequence are different from the ones used in the training sequences.

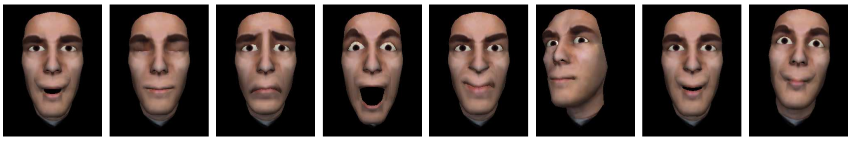

**Fig. 1.** Some of the 75 key-frames used to render the test sequence (1125 images).

[6] See videos in `http://www.dia.fi.upm.es/~lbaumela/FaceExpressionRecognition/`

In the eyes training sequence there is only non-rigid motion in the upper area of the face. Therefore, the stabilised images of the eyes can be extracted automatically by tracking the mouth area with a simple template tracker (using a mouth template). Similarly, as in the mouth training sequence there is only non-rigid motion in the lower face, the mouth stabilised images are computed by rigidly tracking the eyes. We have extracted a region of the eyes with $N_{eyes} = 60 \times 35$ pixels and a region of the mouth with $N_{mouth} = 53 \times 43$ pixels (that will be used both in tracking and re-animation). The normalised images of the 3D model (from the two training sequences) and the ground truth animation parameters allows us to compute $\mathtt{R}_a^i$, for each of the face regions (upper and lower face).

In the experiment conducted we use the projective motion model for appearance based tracking. In order to compute the eigenspace matrix for tracking, $\mathtt{B}$, we use all the training normalised images. For computing the re-animation matrix, $\mathtt{R}_a^i$, we use the 540 and 629 example pairs (images and animation parameters) for eyes and mouth, respectively.

The jaw opening parameter (see figure 2 left) is estimated very accurately except around the frame 830 in which the face is out of the frontal position to the camera. The overall estimation of the pupil horizontal orientation (see figure 2 middle and right) is quite good except in frames 222 to 435, in which the face is not frontal to the camera, and around frame 1050, in which the model is cross-eyed (and we don't have such configuration in the examples).

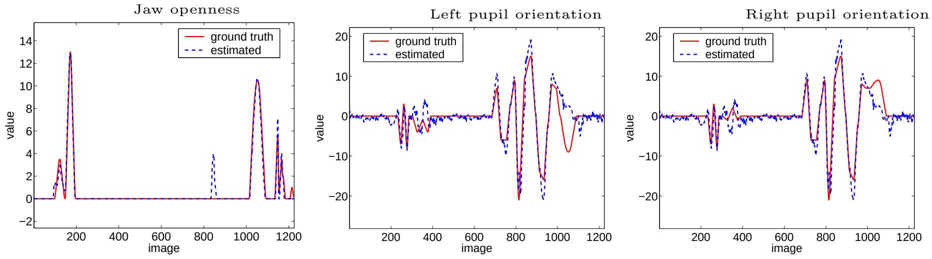

**Fig. 2.** Synthetic experiment results. 8 On the left, it is shown the estimated jaw openness, in the middle the estimated horizontal rotation for the left pupil and on the right the horizontal orientation of the right pupil.

### 4.2 Qualitative validation

We have tested our re-animation system with five different users. The main problem here is the selection of the examples for re-animation (the pairs normalised images, animation parameters). The solution we have adopted is to use a set of known face expressions in the 3D model (key frames) and select manually the corresponding normalised images of the user's eyes (21 examples) and mouth (18 examples). By doing so, we get the set of examples needed for the re-animation training. We use all the training normalised images for computing the eigenspace tracking matrices, B.

All the qualitative experiments were made by taking a very long sequence of images and using half of the sequence for training and the other half for tracking. In figure 3 are shown some of the results for two of the experiments. In the first experiment we used a 4925 image sequence: 2190 images for training and 2735 for testing. And in the second one we used a 4421 images sequence: 2360 images for training and 2061 images for testing. Due to lack of space we can not show all the five re-animation experiments.

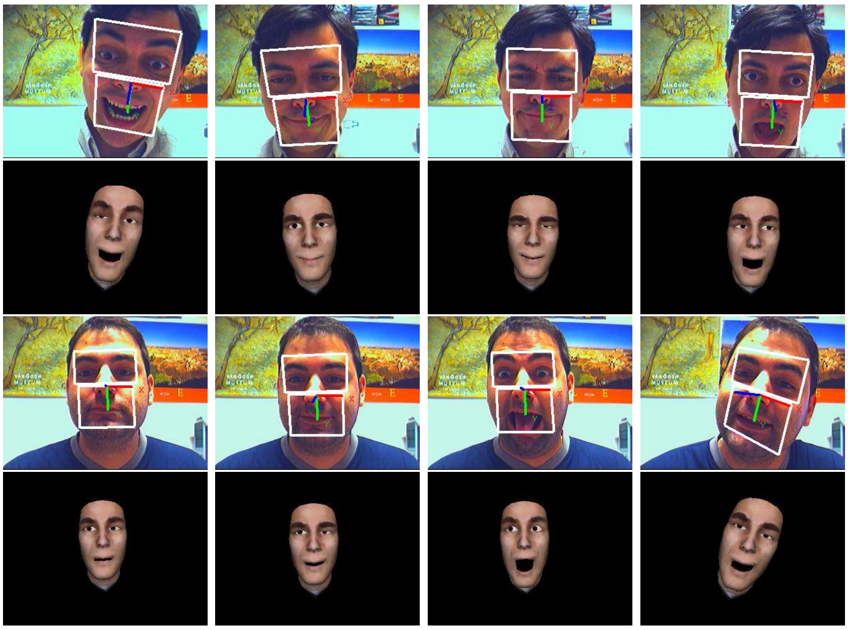

**Fig. 3.** Results of two of the qualitative experiments. In first row, appearance based tracking results for first user (the two face areas locations are overlayed in white) and in second row animation results. In third row, appearance based tracking results for the second experiment and fourth row animation results.

## 5 Conclusions

In this paper we have shown one of the applications of facial analysis: performance driven animation. The animation system presented can be adapted, by training, to any user and illumination conditions and the current implementation of our appearance based tracker (not optimised) can track the upper part of the face at 25 fps and the whole face at 15 fps. Given that the re-animation only needs the multiplication of matrix $\mathtt{R}_i^a$ by the grey levels of the corresponding normalised image, it allows the animation of the 3D model in real time.

Some issues still remain open. The adaptation to a new user is in part manual, mainly because we have not studied how to choose automatically the user images that correspond to facial expressions in the 3D model. We are currently building a robust tracker, which efficiently deal with occlusions and gross illumination changes.

## Acknowledgement

The authors gratefully acknowledge funding from the Spanish Ministry of Science and Technology under grant number TIC2002-000591. Enrique Muñoz was funded by a FPU grant from the Spanish Ministry of Education. We also thanks the anonymous reviewers of this paper for their feedback.

## References

1. Jun-yong, Neumann, U.: Expression cloning. In: Proc. of SIGGRAPH, ACM (2001) 277–288
2. Cohn, J., Kanade, T., Moriyama, T., Ambadar, Z., Xiao, J., Ga, J., Imamura, H.: A comparative study of alternative facs coding algorithms. Technical report, Robotics Institute, Carnegie Mellon University (2001)
3. Tian, Y., Kanade, T., Cohn, J.: Recognizing action units for facial expression analysis. PAMI **23** (2001) 97–115
4. Stéphane Valente, Ana C. Andrés Del Valle, J.L.D.: Analysis and reproduction of facial expressions for realistic communicating clones. Journal of VLSI Signal Processing **29** (2001) 41–49
5. Ahlberg, J.: Using the active appearence algorithm for face and facial feature tracking. In: Proc. of 2nd Int. Workshop on Recognition, analisys and tracking of faces and gestures in real time systems, RATFG-RTS'01. (2001) 68–72
6. Terzopoulos, D., Waters, K.: Analysis and synthesis of facial image sequences using physical and anatomical models. PAMI **15** (1997)
7. Buck, I., Finkelstein, A., Jacobs, C., Klein, A., Salesin, D.H., Seims, J., Szeliski, R., Toyama, K.: Performance-driven hand-drawn animation. In: Proc. of Int. Symposium on Non Photorealistic Animation and Rendering, NPAR'2000. (2000) 101–108
8. Buenaposada, J., Muñoz, E., Baumela, L.: Efficient appearance-based tracking. In: Proc. of Workshop on Nonrigid and Articulated Motion, IEEE (2004)
9. Hager, G., Belhumeur, P.: Efficient region tracking with parametric models of geometry and illumination. PAMI **20** (1998) 1025–1039
10. Black, M.J., Jepson, A.D.: Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. IJCV **26** (1998) 63–84
11. Hou, X., Li, S.Z., Zhang, H., Cheng, Q.: Direct appearance models. In: Proc. of CVPR, IEEE (2001)
12. Parke, F.I., Waters, K.: Computer Facial Animation. AK Peters Ltd (1996)