A NON-INVASIVE APPROACH FOR DRIVING VIRTUAL TALKING HEADS FROM REAL FACIAL MOVEMENTS

Gabriele Fanelli[1], *Marco Fratarcangeli*[1,2]

[1]University of Rome "La Sapienza", Department of Computer and Systems Science, Italy
[2]Linköping Institute of Technology, Department of Electrical Engineering, Sweden

ABSTRACT

In this paper, we depict a system to accurately control the facial animation of synthetic virtual heads from the movements of a real person. Such movements are tracked using Active Appearance Models from videos acquired using a cheap webcam. Tracked motion is then encoded by employing the widely used MPEG-4 Facial and Body Animation standard. Each animation frame is thus expressed by a compact subset of Facial Animation Parameters (FAPs) defined by the standard. We precompute, for each FAP, the corresponding facial configuration of the virtual head to animate through an accurate anatomical simulation. By linearly interpolating, frame by frame, the facial configurations corresponding to the FAPs, we obtain the animation of the virtual head in an easy and straightforward way.

Index Terms— Face Tracking, Active Appearance Models, Inverse Compositional Algorithm, Facial Animation, 3D Motion Animation.

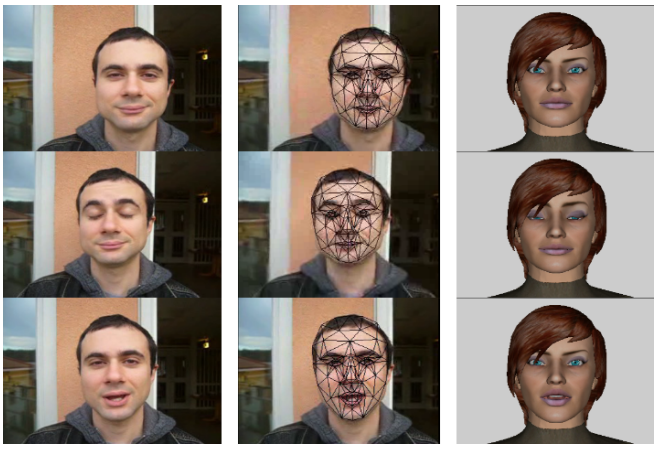

Fig. 1. From each input video frame (left), the facial movements are tracked (center), and then used to control a virtual talking head (right).

1. INTRODUCTION

Facial expressions, together with speech, convey the main part of information about moods, intentions, and feelings of a person. For this reason, computer vision and computer graphics researchers devoted great efforts to the development of techniques able to track, parameterize, and synthesize believable facial animation in an automatic, fast, and easy manner. This is an open research problem with a wide range of possible applications such as special effects, communications, visual speech synthesis and maxillofacial medicine. The complex structure of the face, however, makes this goal particularly hard to achieve. Furthermore, as humans, we are trained to observe and decode the subtlest movement of the face and, therefore, it is relatively easy to detect any small artifact in the motion of a virtual face.

This paper addresses the problem of realistically animate a virtual talking head at interactive rate by re-synthesizing facial movements tracked from a real person using cheap and non-invasive equipment, namely a standard webcam (Fig. 1). Using appropriately trained Active Appearance Models, our system is able to track the facial movements of a real person from a video stream and then parameterize such movements in the scripting language defined by the MPEG-4 FBA standard [1]. Each of these parameters corresponds to a key pose of a virtual face (Fig. 2), namely Morph Target (MT), a concept largely known in the computer graphics artists community. These initial key poses are automatically precomputed through an accurate anatomical model of the face composed by the underlying bony structure, the upper skull and the jaw, the muscle map, and the soft skin tissue. The morph targets are blended together through a linear interpolation weighted by the parameter's magnitude, achieving a wide range of facial configurations.

The remainder of this paper is organized as follows. Sec. 2 briefly presents related work on which our system is based. Sec. 3 describes the technique used to track the real face, and Sec. 4 focuses on the main principles of the already known techniques employed to build the morph targets using the anatomical model. Results are provided in Sec. 5, while conclusions, discussion, and ongoing work are depicted in Sec. 6.

Fig. 2. Morph Targets corresponding to the MPEG-4 FAPs for the six standard expressions (anger, disgust, fear, joy, sadness, surprise).

2. BACKGROUND

MPEG-4 Facial and Body Animation: We assume the reader to be familiar with the basic notions of MPEG-4 FBA. Interested readers may refer to excellent references such as [1] [2] covering this subject. The standard provides a feature-based parameterization for facial animation able to define and control the shape and the motion of a talking virtual head. Such a standard makes use of two main sets of parameters: Facial Definition Parameters (FDP), defining the shape of a face, and Facial Animation Parameters (FAP), controlling motion. The FDP set is comprised of 84 standardized feature points on the face (Fig. 4, left). The FAP set provides several layers of abstraction: 6 expression (Fig. 2), and 14 viseme parameters allow high-level specification of facial expressions. 68 lower-level parameters represent elementary facial motion, providing control over eyes, tongue, mouth, head orientation, and even nose and ears. For each animation parameter, the influenced feature points are specified by the standard.

Active Appearance Models (AAMs) are generative and parametric models commonly employed for face tracking. Dornaika and Ahlberg [3] built a 3D real-time face tracker based on the original AAM fitting algorithm, proposed by Cootes et al. [4]. Recently, Baker and Matthews [5] introduced a new algorithm, which solves the optimization problem using analytically calculated gradients, rather than numerically estimated ones.

3. FACE AND FACIAL FEATURES TRACKING

We used *independent* AAMs, consisting of two separated linear models: one for the shape and one for the appearance of a particular visual phenomenon. The shape is defined as a set of vertex coordinates forming a triangulated mesh, while the appearance is defined as the intensity of the pixels within this mesh. From the statistical analysis of sample face images, the models learn what are the permitted variations in shape and appearance for the class of objects of interest (in our case, the human face). We created the AAMs using the structure of an existing face model: Candide-3 [6], a MPEG-4 compliant wire frame model (Fig. 3, left).

Starting from a set of training images depicting one subject while performing different expressions under changing light conditions, we obtained a *person specific AAM*, able to model and track facial movements in that same subject. We suppose the face to be approximately in frontal pose but free to move on the image plane.

The mathematically correct and fast Project Out algorithm, recently proposed by Baker and Matthews [5], has been implemented, making our model able to automatically adapt to new face images at interactive rate.

3.1. AAM Construction

AAMs are usually built by manually locating a set of landmarks strategically disposed around the facial features and the face outline on the training pictures. We started from an existing wire frame parametric face model, Candide-3, composed by 113 vertices and 184 triangles [6]. Even though Candide-3 is a 3D model, we only used the 2D projection of its vertices on the image plane, since we require the input real face to be approximately in frontal pose. Using Candide-3 makes the extraction of the MPEG-4 FAPs easy, since most of the model's vertices correspond to MPEG-4 FDPs.

The shape of a face is described by a set of 2D vertex coordinates, stored as a vector. For each training image, the full set of coordinates is obtained by manually locating a subset of 31 key vertices and then solving a least square fitting problem with respect to the Candide-3 parameters (Fig. 3).

The collected shape vectors have been analysed using Principal Component Analysis [4], yielding a mean shape vector $\mathbf{S}_0$ and n eigenvectors $\mathbf{S}_i$ corresponding to the n largest eigenvalues, also called *shape deformation vectors*. AAMs allow linear shape variation, i.e., a new shape $\mathbf{S}$ can be expressed as:

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^{n} p_i \mathbf{s}_i, \tag{1}$$

with p_i being the shape parameters.

The appearance is defined as the pattern of pixel intensities within the mean shape $\mathbf{S}_0$ (we use $\mathbf{S}_0$ to denote also the pixels enclosed by the mean shape). To build the appearance model, PCA is used again on the "shape normalized" training images, i.e., after they have been mapped onto the mean shape. The obtained model permits to express a new appearance $A(\mathbf{x})$ as the linear combination of the mean appearance $A_0(\mathbf{x})$ and m *eigenfaces* $A_i(\mathbf{x})$:

$$A(\mathbf{x}) = A_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i A_i(\mathbf{x}) \qquad \forall \mathbf{x} \in \mathbf{s}_0, \tag{2}$$

where λ_i represent the appearance parameters.

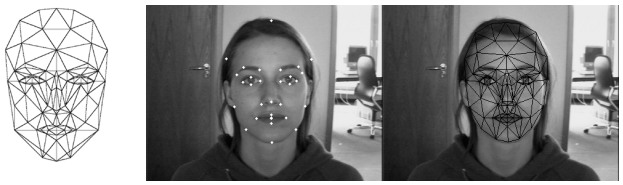

Fig. 3. (Left.) Candide-3 [6]. (Center.) Manually selected landmarks on a training image and (Right.) Candide-3 adapted to them. Input image from [7].

3.2. AAM Fitting

To match the AAMs to a new face image, the following expression has to be minimized with respect to both the shape parameters $\mathbf{p}$ and the appearance parameters λ_i:

$$\sum_{\mathbf{x}\in\mathbf{s}_0}\left[A_0(\mathbf{x})+\sum_{i=1}^{m}\lambda_i A_i(\mathbf{x})-I(\mathbf{W}(\mathbf{x};\mathbf{p}))\right]^2, \quad (3)$$

where $I(\mathbf{W}(\mathbf{x};\mathbf{p}))$ is the input image *warped* onto the mean shape $\mathbf{S}_0$ through the function $\mathbf{W}(\mathbf{x},\mathbf{p})$.

Recently, Baker and Matthews proposed the Inverse Compositional Algorithm [5] which allows for an accurate and fast fitting, actually reversing the roles of the input image $I(\mathbf{x})$ and the template $A_0(\mathbf{x})$ in the well-known, but slow, Lucas-Kanade Image Alignment algorithm [8].

Supposing that the appearance will not change much among different frames, it can be "projected out" from the search space [5]. The search can therefore focus only on the shape parameters, aiming to minimize:

$$\sum_{\mathbf{x}\in\mathbf{s}_0}\left[I(\mathbf{W}(\mathbf{x};\mathbf{p}))-A_0(\mathbf{W}(\mathbf{x};\Delta\mathbf{p}))\right]^2, \quad (4)$$

with respect to the incremental warp $\mathbf{W}(\mathbf{x};\Delta\mathbf{p})$ applied to the template $A_0(\mathbf{x})$. The solution to the above problem will need to be inverted and then composed to the current warp: $\mathbf{W}(\mathbf{x};\mathbf{p}) \leftarrow \mathbf{W}(\mathbf{x};\mathbf{p})\circ\mathbf{W}(\mathbf{x};\Delta\mathbf{p})^{-1}$.

Taylor-expanding expression 4 yields:

$$\sum_{\mathbf{x}\in\mathbf{s}_0}\left[I(\mathbf{W}(\mathbf{x};\mathbf{p}))-A_0(\mathbf{W}(\mathbf{x};\mathbf{0}))-\nabla A_0\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\Delta\mathbf{p}\right]^2, \quad (5)$$

which, assuming the identity warp ($A_0(\mathbf{W}(\mathbf{x};0))=A_0(\mathbf{x})$) for $\mathbf{p}=0$, has the following closed form solution:

$$\mathbf{\Delta p}=\mathbf{H^{-1}}\sum_{\mathbf{x}\in\mathbf{s_0}}\left[\nabla A_0\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T\left[I(\mathbf{W}(\mathbf{x};\mathbf{p}))-A_0(\mathbf{x})\right]. \quad (6)$$

$\mathbf{H}$ is the Gauss-Newton approximation of the Hessian matrix, defined as:

$$\mathbf{H}=\sum_{\mathbf{x}\in\mathbf{s}_0}\left[\nabla A_0\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T\left[\nabla A_0\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]. \quad (7)$$

The gradient of the template, $\nabla A_0(\mathbf{x})$, and the Jacobian of the warp, $\frac{\partial\mathbf{W}}{\partial\mathbf{p}}$, are both constant and can be precomputed, and so can the inverse of the Hessian matrix, leading to a fast algorithm. For more information please refer to [5].

The initial position and size of the model are evaluated thanks to an implementation of Viola and Jones' algorithm [9], able to locate a face in the image. The face detector is called again in case the fitting algorithm loses track for a sudden movement of the head.

3.3. Extracting the MPEG-4 FAPs

While tracking, the FAPs are computed by comparing the mean shape $\mathbf{S}_0$ and the current shape $\mathbf{S}$. Each FAP controls a set of FDPs, in turn corresponding to Candide-3 vertices. The FAP value is computed as the difference between the Candide-3 vertex coordinates in the current shape and in the mean shape, divided by a normalization factor, namely the MPEG-4 FAPU [1].

4. ANATOMICAL MODEL

Starting from a input 3D triangle mesh with MPG-4 FDPs associated (Fig. 4), we use some already developed techniques to build a face model conforming to the anatomical structure of the human head. We describe these methods and how we used them together with MPEG-4 FBA information to automatically build the anatomical face model in our earlier work [10]. We specify a mechanism to place the muscle models and the jaw in the anatomically correct position by using the MPEG-4 FDPs. The influence area as well as the minimum and the maximum magnitude of contraction of the muscle and the jaw, are automatically detected for each particular input mesh. Figure 4 shows the spring network and the facial muscle map built on a test face mesh.

For each morph target, we synthesize it acting on the proper group of muscles and, if needed, rotating the jaw. The deformation of the skin regions affected by the produced force fields is computed through the semi-implicit numerical integration scheme whose solution is the desired morph target. Using an anatomical model, we produce realistic looking morph targets in an automatic and efficient manner. It is worth noting that this method is not hard-coded around a particular input face mesh.

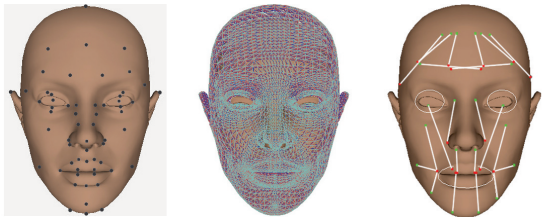

Fig. 4. Left. Front view of an input face mesh with MPEG-4 FDPs scattered data set associated. Center. Multi-layered structure representing the skin. Right. The muscle map.

5. RESULTS

The system has been tested on low-resolution (320×240) videos showing a real actor's face moving and talking. The tracker succeeded in following the head movements and the local deformations of the face, achieving real-time performance. Each frame is analyzed in approx. 40 ms on a 1.66 GHz processor, 2 MB cache and 2GB RAM. The synthesis of the corresponding motion on the virtual face is one order of magnitude lesser since it is achieved by linear interpolation of the corresponding tracked FAPs. Thus, the system works at interactive rate.

To test the Animatable Face Models (AFMs), we used a commercial player [11]. This player gets as input the AFM and a MPEG-4 FBA encoded data stream. Each frame of the facial animation is produced by a weighted linear interpolation of the morph targets corresponding to the FAPs specified by the data stream for that animation frame. This keeps the computational cost low while the achieved animation looks realistic. Thus, such AFMs are particularly suitable for virtual environments requiring interactive facial animation of embodied agents.

Figure 5 shows some results with different virtual heads.

6. CONCLUSIONS

We developed a system for controlling virtual talking heads at interactive rate tracking and encoding the facial movements of a real person depicted in monocular video streams recorded through a low-end camera. Future developments will focus on extending our system to support 3D rigid transformations of the real head (i.e., translations and out-of-plane rotations), and the iris movements. Fields of possible application include entertainment industry or human-computer interaction software, where cartoon-like characters could reproduce the expressions of real actors without the aid of expensive and invasive devices, or visual communication systems, where video conferences could be established even on very low bandwidth links.

Fig. 5. Controlling different virtual heads.

7. REFERENCES

[1] I.S. Pandzic and R. Forchheimer, Eds., *"MPEG-4 Facial Animation – The Standard, Implementation and Applications"*, John Wiley & Sons, LTD, Linköping, Sweden, 1st edition, 2002.

[2] J. Ostermann, "Animation of Synthetic Faces in MPEG-4," in *Computer Animation*, Philadelphia, Pennsylvania, June 1998, pp. 49–51.

[3] F. Dornaika and J. Ahlberg, "Fast and reliable active appearance model search for 3-d face tracking.," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, no. 4, pp. 1838–1853, 2004.

[4] T.F. Cootes, G.J. Edwards, and C.J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 681–685, Jan. 2001.

[5] I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 135 – 164, November 2004, In Press.

[6] J. Ahlberg, "Candide-3 - an updated parameterised face," Tech. Rep. LiTH-ISY-R-2326, Image Coding Group, Dept. of Electrical Engineering, Linkoping University, Sweden, 2001.

[7] BioID, "The BioID face database," 2001, Available at `http://www.humanscan.de/support/downloads/facedb.php`, accessed August 1, 2006.

[8] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision (ijcai)," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*, April 1981, pp. 674–679.

[9] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *In IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 609–615.

[10] M. Fratarcangeli, "Physically Based Synthesis of Animatable Face Models," in *Proceedings of the 2nd International Workshop on Virtual Reality and Physical Simulation (VRIPHYS05)*, Pisa, Italy, November 2005, ISTI-CNR, pp. 32–39, The Eurographics Association.

[11] Visage Technologies AB, 2006, `http://www.visagetechnologies.com`, accessed August 1, 2006.