appears in: *International Workshop on Very Low Bitrate Video Coding (VLBV98)*
October 8-9, 1998, Urbana, IL, USA

3D Face Modeling and Encoding for Virtual Teleconferencing

Stéphane Valente & Jean–Luc Dugelay

Institut Eurécom, Multimedia Communications Departement
B.P. 193, 06904 Sophia–Antipolis Cedex, France
{valente, dugelay}@eurecom.fr
http://www.eurecom.fr/~image

Abstract

In the context of a Virtual Teleconferencing *system, we present a head tracking algorithm based on an enhanced analysis/synthesis feedback loop which is able to handle very large rotations out of the image plane, although the camera is uncalibrated, the environment lighting is unknown, and no makeup highlights the speaker's face.*

1 Related Work

Face–cloning aims at animating a synthetic face model by analysing a video sequence of a real speaker. In the literature, many references concerning video–cloning report promising results, such as [8, 9, 7, 2, 4, 3]. The material presented in this paper has been derived in the context of a virtual teleconferencing system [10], where the users can meet other people in a virtual meeting room, and choose their positions within it. Its telecommunication aspects impose specific and challenging constraints on facial cloning, like the face analysis and synthesis frame–rates, the synthesis of the participants under different points of view depending on the viewer, the image processing delays, and the very low bandwidth networks available to transmit the animation parameters.

We present in this paper face modeling and global motion tracking techniques for such a system, that operates without colored marks taped on the speaker's face, deals with unknown lighting conditions and background, allows the users to move freely in front of the camera, and yields visual results that are highly realistic.

2 Face Modeling

We are currently using range data obtained from cylindrical geometry Cyberware range finders [1] to build person–dependent realistic face models. Such scanners produce a dense range image with its corresponding cylindrical color texture. However, the dataset cannot be used directly because it is too dense (in average 1.4 million vertices) and sometimes includes some outliers (as in figure 1(a)).

To achieve both visual realism and real–time computation, we need a geometric model with a limited number of vertices but with enough details in order to distinguish facial features such as the lips or eyebrows. We have developed a reconstruction system based on deformable simplex meshes [11] to build such models. Unlike classic approaches, those deformable models are handled as discrete meshes, and can be easily converted into triangle meshes.

In figure 1, we show the different stages of reconstruction from a Cyberware dataset where the hair information is missing and with some outliers. The deformable model is initialized as a sphere (figure 1(b)) and then deformed to roughly approximate the face geometry (figure 1(c)). The last stage consists in refining the mesh model based on the distance between the data and surface curvature (figure 1(d)).

The face model is then texture–mapped by associating to each vertex of the simplex mesh the (u, v) texture coordinates of its closest point in the range data. Where no range data is available (at the hair level for instance), we project the vertex on the image plane through the cylindrical transformation of the Cyberware acquisition. This algorithm therefore produces an accurate geometric and texture face model, suitable for real–time manipulation.

3 Face Global Motion Encoding

In this contribution, only the parameters for the speaker's global motion (3 translations and 3 rotations) are considered.

(a) (b) (c) (d)

Figure 1: Reconstruction of a geometric model from a Cyberware dataset: (a) range data (b) initialization; (c) main deformation; (d) mesh refinement — We have interactively selected the areas of interest (chin, ears, nose, lips) where the refinement is performed. The resulting mesh has 2084 vertices and was built in less than 5 minutes on a DEC Alphastation 233 MHz.

3.1 Analysis/Synthesis Feedback Loop

To provide a high level of realism, we propose to use $3D$ texture–mapped models to represent each speaker within the virtual area. Taking advantage of this realism, we propose a global motion tracking software implementing a differential block–matching algorithm tracking 2D feature points from synthesized patterns. The head tracking loop proceedes as follows (see figure 2):

- a Kalman filter predicts the head 3D position and orientation estimates at time t given the previous 2D feature points observations in all images until time $t-1$;
- using the estimated $3D$ parameters and the speaker's head model, search patterns for the facial features are synthesized, hence taking into account the scale and geometric deformations that can be expected given the user's position, and the background interference with the patterns. In addition, due to the $3D$ photometric compensation module described in section 3.2, the search patterns also reflect the expected face lighting;
- a reformulated block–matching algorithm finds the synthesized patterns in the real image taken at time t;
- the Kalman filter is then fed with the $2D$ observations of the facial features in the image plane to produce new estimates for the head $3D$ position and orientation at time $t+1$.

Our enhanced analysis/synthesis cooperation makes the face tracking more robust without requiring artificial marks, and supports very large rotations out of the image plane, as it can be seen on figure 2.

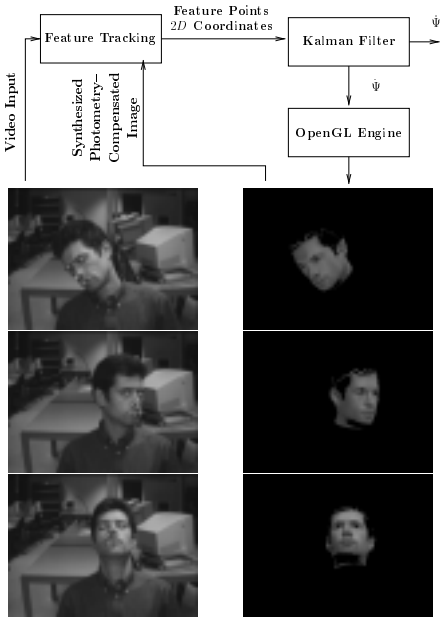

Figure 2: Feedback loop strategy based on a Kalman Filter and Synthetic Images — $\check{\Psi}$ and $\hat{\Psi}$ are the speaker's $3D$ position and orientation predicted and filtered estimates. The shown examples were extracted from a 30 seconds video sequence captured in a 320×242 resolution at 10 frames per second.

3.2 Photometric Compensation

Figures 3(a) and 3(b) actually show that the default illumination of the synthetic facial patterns would not allow any match with the user's face in a real environment, no matter how precise the geometric modeling is, firstly because "out–of–the–lab" environments generally have uncalibrated lightings, and secondly because the speakers generally do not have makeup to avoid specular highlights on their face. Using OpenGL, the $3D$ graphics industry–standard library, we render the face model with a set of ambient, diffuse and specular lights (see figure 3 (d)) to minimize the discrepancies between the synthetic and real faces throughout the tracking session (an illumination offset can be allowed to improve the reconstruction error, see figure 3 (c)) . The light intensities are determined by an estimation stage at the begining of the tracking session on a static view of the speaker. This way, the illumination compensation takes place directly at the $3D$ level, during the synthesis of facial patterns, and is performed by graphics hardware instead of explicit software computations. Of course, we do not claim that the exact scene illumination is recovered, nor that

the simulated lighting will be right for all the possible user's positions, but it helps gaining consistency between the synthetic and real facial features (see figure 3). We invite interested readers to refer to [11] for the theorical reasonings and technical details of the photometric modules.

(a) (b) (c)

Figure 3: Illumination compensation on a real face — from left to right: the speaker's head model with no directional light source, the speaker in a real environment, and the speaker's head model with $3D$ illumination compensation.

3.3 Block–Matching Synthetic Facial Features

Figure 4: Synthesized patterns are not mislead by the background presence.

The key aspect of the global motion tracking algorithm is the cooperation between the analysis and synthesis procedures: due to the illumination compensation performed at the beginning of the tracking session, it is possible to track the facial features of the user given by synthesized images using a differential block–matching algorithm, involving only linear computations. The analysis/synthesis feedback loop, governed by a Kalman filter predicting the global motion, results in two main advantages:

- the synthesized facial features automatically adapt to (or at least are not far from) local distortions in the image plane, such as variations of scale, geometry and changes of lighting due to the speaker's $3D$ motions;
- and because the facial patterns are synthesized over a black solid background, the block–matching algorithm can be restricted to the facial feature pixels to be more discriminative when the speaker's facial features are likely to be mistaken for a textured background (as in the head rotation of figure 4).

3.4 Discussion on the Tracking Robustness

The result of our face tracking algorithm can be seen in an Mpeg sequence available on the WWW [6]. Its speed mainly depends on the workstation graphics hardware acceleration and its video acquisition speed. On a O^2 SGI workstation, the analysis frame rate using 12 facial feature areas is:

- 1 image per second, when synthesizing patterns, and updating the Kalman filter for every frame;
- 10 frames per second, when disabling synthetic pattern calculation for every frame, but still enabling the Kalman filter — in this case, large face rotations might cause the system loose the user's head;
- full frame rate, when disabling both pattern synthesis and the Kalman filter — the system just tracks the facial features in $2D$, without recovering the head $3D$ position and rotation, and becomes very sensitive to rotations.

In fact, the individual facial features trackers work quite well, even during large face rotations when it becomes difficult to distinguish the facial features from the scene background (look for example at the speaker's right eye on figure 4). From our experiments, the main difficulty to obtain a robust face tracking system is the tuning of the Kalman filter: it requires to set noises for the observations and the system dynamics. On the one hand, if the noises are too small, the filter may become unstable, probably because of round–off errors, and the system looses the user's face, even if it is fed with the right $2D$ features positions. On the other hand, if the noises are too large, the system no longer takes into account the incoming facial features observations, and as a result, if the user starts moving in one direction, the filter will follow the face at first, but will never go back when the user returns toward his initial position.

Another question that might be raised is what happens when the user closes his eyes, smiles, or does anything that differs from the static facial expression of his model: in general, we can say that the system copes with it, probably because it melts enough facial features observations to allow a few of them to be wrong. We are currently working on the modeling

of facial expressions in the feedback loop (that have eventually to be analysed in the face cloning application), and we expect them to improve the system performance in such situations.

4 Face Restitution

4.1 Bandwidth Estimation

Prior to the beginning of the meeting session, the face models have to be downloaded by every participating site to represent each speaker, corresponding to a hundred of kilobytes per model, but then, only a limited bandwidth is necessary throughout the session: the 6 global motion parameters are each quantified and coded on two bytes, with no particular compression, resulting in a data stream of 12 bytes per frame.

4.2 Model Visualization

In the case of a virtual teleconferencing system, the data stream originating from a given site is sent for visualization to all the other sites, which interpret the data stream and insert the face model in the virtual environment according to the point of view local to the site (see figure 5). Our face model and the extracted parameters are general enough to be visualized on different platforms, and using different tools:

- our visualization software, based on the OpenGL library;
- a VRML browser [12];
- the MPEG–4 SNHC Face Player, when the International Standard will be released (due in November 1998) [5].

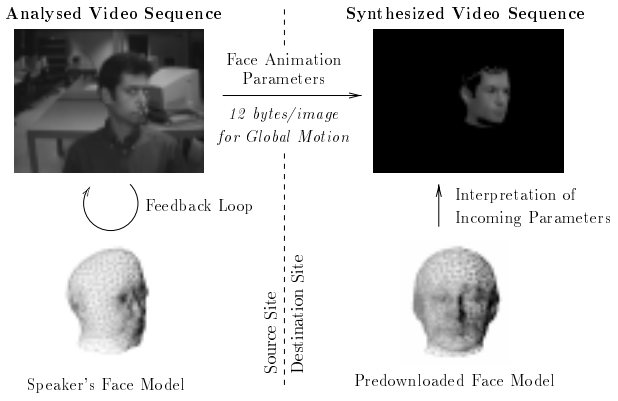

Figure 5: Animated Face Restitution.

For classical applications, such as "head & shoulder" type video compression where only two sites are involved and no *virtuality* is necessary, the face model is just synthesized under the same point of view of the analysis camera. Note that in this case, the illumination parameters computed for the feedback loop (see section 3.2) can be used by the synthesis software to produce a view more faithful to the original one.

Concluding Remarks

We have presented face modeling and global motion tracking techniques, based on an efficient simplex mesh formulation, and on an original analysis/synthesis feedback loop. A demonstration MPEG sequence is available on the WWW [6].

Preliminary results concerning the animation of the model facial features have already been obtained — by the time of the workshop, we expect to include them in a video demonstration.

Acknowledgments

This research is supported by the Eurécom Institute and its academic and industrial members.

References

[1] CYBERWARE Home Page. URL http://www.cyberware.com .

[2] I. A. Essa and A. Pentland. A vision system for observing and extracting facial action parameters. In *International Conference on Computer Vision and Pattern Recognition*, pages 76–83, Seattle, WA, June 1994.

[3] T. S. Huang and L. Tang. Model–based video coding — Some challenging issues. In Y. Wang, S. Panwar, S.-P. Kim, and H. L. Bertoni, editors, *Multimedia Communications and Video Coding*, pages 215–221. Plenum Press, New–York, 1996.

[4] H. Li, P. Roivainen, and R. Forchheimer. 3–D motion estimation in model–based facial image coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):545–555, June 1993.

[5] MPEG–4 synthetic/natural hybrid coding. URL http://www.es.com/mpeg4-snhc/ .

[6] MPEG demo of the face tracking system. URL http://www.eurecom.fr/~image/TRAIVI/valente-8points.mpg. (1,782,100 bytes).

[7] I. S. Pandzic, P. Kalra, and N. Magnenat Thalmann. Real time facial interaction. *Displays*, 15(3), 1995. *Butterworth — Heinemann.*

[8] A. Saulnier, M.-L. Viaud, and D. Geldreich. Real–time facial analysis and synthesis chain. In *International Workshop on Automatic Face— and Gesture— Recognition*, pages 86–91, Zurich, Switzerland, 1995.

[9] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6), June 1993.

[10] S. Valente and J.-L. Dugelay. A multi–site teleconferencing system using VR paradigms. In *Ecmast*, Milano, Italy, 1997.

[11] S. Valente, J.-L. Dugelay, and H. Delingette. Geometric and photometric head modeling for facial analysis technologies. Technical report, Institut Eurécom, 1998.

[12] VRML. URL http://vrml.sgi.com .