Low Cost 3D Face Acquisition and Modeling

Emmanuel Garcia, Jean-Luc Dugelay
Dept. of Multimedia Communications, Institut Eurécom
2229 route des Crêtes, B.P. 193, 06904 Sophia Antipolis - France
{garciae,dugelay}@eurecom.fr

Hervé Delingette
INRIA, Epidaure, 2004 route des Lucioles, B.P. 93
06904 Sophia Antipolis - France

Abstract

In this article we present a low cost 3D face modeling scheme that uses only an overhead projector, a camera and a plane object. We deal with the calibration of such a system, the 3D face reconstruction itself, and then the texturing of the resulting 3D shape.

1. Introduction

Being able to reproduce from speech or video signals the facial expressions of a human face on a synthetic head model is of tremendous importance to many multimedia applications (e.g. MPEG-4 [1]). Such models can be divided into two main categories: avatars and clones. Avatars are rough and unrealistic and generally only symbolic representations of the human face, but are easy to design and then to animate. On the other hand, a clone must look like the face of a real person. For some applications, such as virtual teleconferencing [2], the use of realistic faces is mandatory. The acquisition of realistic faces unfortunately requires special purpose equipment such as the Cyberware machine [3], that is of high cost and only available to few users. An approximation of a clone can be achieved by employing a generic 3D geometry and only customize the model by mapping on it a real texture extracted from one [4] or two pictures (face and side view). To overcome this limitation, mainly in terms of geometry, some laboratories investigate the possibility to design realistic faces only from images. The proposed approaches can be divided into two main categories: passive and active. The former is often based on a multi viewpoint analysis [5] and the latter one often uses the projection of a grid on the user [6]. In this paper, we propose and describe a complete framework (see figure 1) for realistic face acquisition and modeling, including both geometry and photometry data.

2. Proposed approach

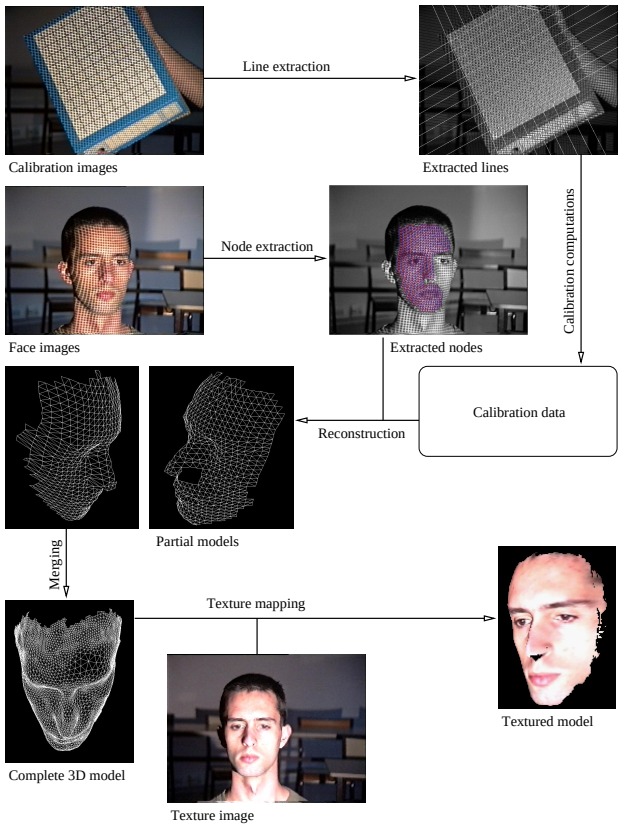


Figure 1: Overview of our system

Our face reconstruction scheme uses only a single camera, an overhead projector and a plane calibration object. A grid pattern is projected onto the face of the subject and its

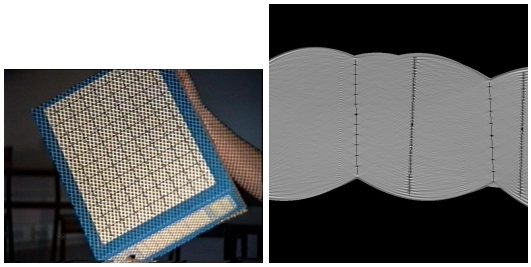
deformations are analyzed in the image plane of the camera to obtain the accurate 3D shape of the visible part of the face.

To make an Euclidean (preserving length ratios and angles) reconstruction possible, the system must be calibrated beforehand. This is done by analyzing the image of an a priori known calibration object instead of a face. This object is a plane on which a square grid is drawn and it must be observed in at least 3 different positions (a similar calibration method, not dealing with the calibration of a projector, but accounting for lens distortion has been studied earlier in [7] where details, as well as quantitative results can be found).

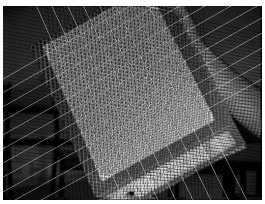
A few 3D models constructed from different views are necessary to build a complete 3D model of the face. Then an image of the face without the light pattern is mapped onto the complete 3D model to obtain a textured 3D model.

3. Line detection in calibration images

The calibration step uses images of a plane on which a (dark) grid pattern is drawn and on which a (light) grid pattern is projected (Fig. 2a). Assuming a distortionless pinhole projection model for the camera, these grids are seen as four sets of almost parallel lines (two sets for the image of each grid). So, the calibration images have a very simple structure. All we need is to detect the lines in each set of almost parallel lines to characterize the image of each grid and to carry on calibration computations.



a b



c

Figure 2: Line detection in calibration images

Although the set of lines is very dense, we can locate all of them accurately by using the Radon transform (projection of the image in every direction, similar to Hough transform for lines). In the Radon space, each bright line is represented by a bright point and each dark line by a dark point. A set of almost parallel lines is represented by a set of almost aligned points. Since we have four such sets of lines to detect, we look for four sets of almost aligned points in the Radon space (Fig. 2b) which is fairly easy if the resolution of the transform is high enough to get a clear separation between these points. Figure 2c shows the detected lines superimposed on the initial image.

4. Calibration computations

Calibration includes two goals. First, to determine the intrinsic parameters of the camera (distortion not accounted for), and then, to determine the position of the overhead projector with respect to the camera and its intrinsic parameters (given a set of coordinates attached to the square grid drawn on the projected slide).

Our calibration method is based on the fact that the whole calibration problem can be expressed in terms of seven homographies (shown in fig. 3). This said, the calibration is no more than an ordered estimation of these homographies.

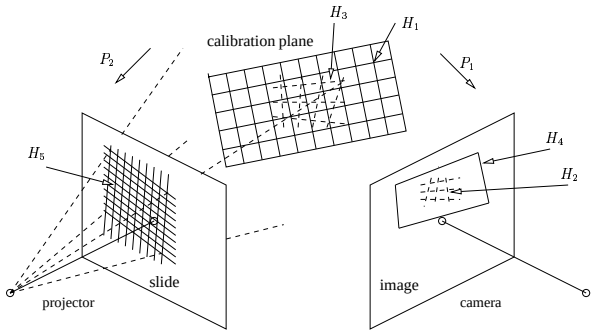


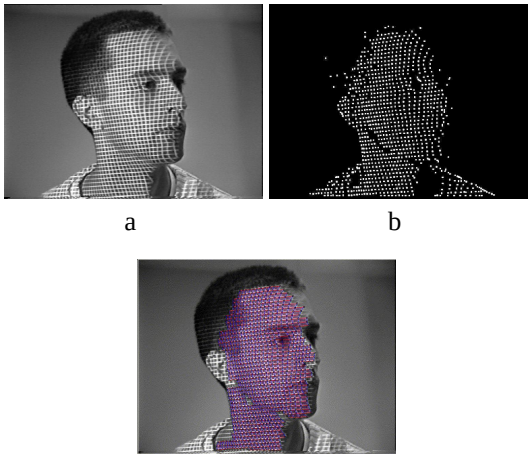
Figure 3: Calibration model

Basically, all grids, or deformed images of grids, either observed in the image plane, or drawn on the slide or calibration plane, can be defined as a homographic transform from the projective 2D space representing the numbering of nodes into either a projective 2D space representing the slide or image, or the projective 3D space. These homographies are denoted H_i on the figure. The other two homographies, denoted P_i, represent the projections of the 3D space into the image plane of the camera, or onto the slide of the overhead projector.

All the calibration information is contained in homographies P_i that we estimate after having computed homographies H_i. From the previous step (line extraction in images) we can estimate homographies H_2 and H_4 for a given image. The knowledge of these two homographies for at least

3 different images allows us to estimate successively all homographies, including P_1 and P_2.

5. Node detection in face images



a b c

Figure 4: Grid extraction in face images

The face images used to reconstruct the 3D shape feature a deformed bright grid pattern (Fig. 4a). The first step for reconstruction is to estimate the coordinates of the grid's nodes and their relative logical coordinates (integer row and column number, also referred to as the connectivity graph) on the grid. To achieve this, we first detect horizontal and vertical lines (not edges) using low level filtering, thresholding and mathematical morphology. Then we set the nodes of the grid to be the intersections of the detected horizontal and vertical lines (Fig. 4b). Their relative logical coordinates are obtained by ordering them on the horizontal and vertical line to which they belong. When a node is missing or is incorrectly detected, which is likely to happen especially around the nose or the eyes, a manual intervention is possible to set things right. Figure 4c shows the detected grid on the face.

6. Partial 3D reconstruction

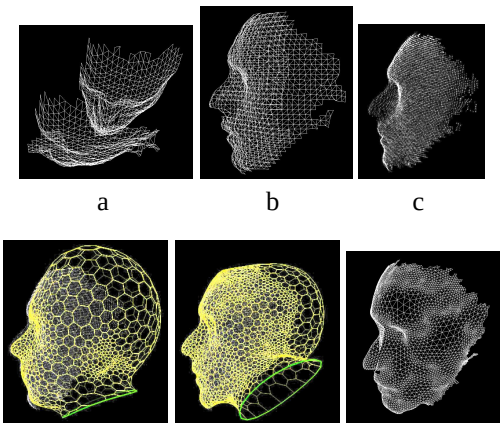
Since our method requires to detect the light grid pattern projected onto the face, this pattern must be visible under a high angle of incidence. Thus, from a single front view of the face it is generally not possible to discern the grid pattern on the sides of the nose or of the head. Therefore, a few side views are also used. Still, one view is enough to build a partial 3D model. This is simply done by triangulation using the coordinates of the nodes detected in the image of the face and the complete geometrical configuration of the system, known from the calibration step.

The point here is, given a node in the image, to identify its associated node on the slide. This cannot be reliably estimated independently for each node due to the fact that our data (discrete images) is not perfect. So we use the constraint that two adjacent nodes on the slide have two adjacent images in the camera. Applying this constraint on all nodes allows us to determine consistently and up to a small global translation error, which node of the slide has a given image in the camera image plane.

The 3D partial models obtained are all reconstructed in physical length units and have the same scale. Yet, they are obtained for different positions of the head of the subject and cannot be merged directly. They first have to be registered.

7. Complete 3D reconstruction

To register the partial 3D models obtained before, we order them and register each partial model with respect to the previous one. Registering two partial models (Fig. 5a) is performed in two steps. First they are registered roughly by manually selecting a few feature points (a limited subset of points used in the MPEG-4 standard for face animation, that is to say: corner of the eyes or mouth, tip of the nose, etc.) on the original images they were obtained from, and by computing the rigid transform to be applied to one of them in order to minimize the mean square distance between their corresponding feature points.



a b c d e f

Figure 5: Merging of partial models into a complete one

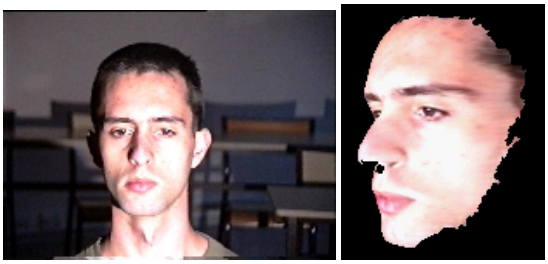
Then they are adjusted more finely by minimizing a potential function that measures the distance between surfaces of two partial models (Fig. 5b). The minimization is done iteratively by gradient descent. Although the process looks like an Iterated Closest Point (ICP) algorithm, it is slightly

different since the potential function used is basically a (limited) double integral on the two surfaces to be registered.

Once the partial models are registered with respect to each other, they are merged to obtain a unique complete 3D model of the face. This is done by first computing a cylindrical depth map for each partial model and by averaging them into a unique depth map of the complete model (Fig. 5c). A mesh of the model is then obtained by iteratively adapting the mesh of a generic face on this depth map (Fig. 5d, 5e). The mesh used is a so-called simplex mesh [8] that is converted, after adaptation to our data, into a triangular mesh of the face (Fig. 5f).

8. Texture mapping

Some preliminary tests have indicated that it would be very difficult, if not impossible, to remove the grid pattern from the face images where it is present, using image processing techniques, while leaving a clean high resolution texture.



a b

Figure 6: Mapping of a texture image

So, we chose to map a picture of the face, taken without projecting the grid pattern on it (Fig. 6a), onto the global 3D model obtained in the previous step. The point here is that the texture image taken is independent from the images taken for reconstruction: the subject may have moved. Thus, we have to register the complete 3D model with respect to that texture image. That is, we have to find the location of the 3D model in space for which its image in the camera is the texture image taken. To do this we use the same manually selected feature points that we already used for registering partial models (we just have to select them also on the texture image) and we compute the rigid transform to be applied to the 3D model for which the feature points on the model project onto the corresponding feature points in the texture image. Then we simply project the texture back onto the model. This registering of the model with respect to the texture image, and the mapping of the texture, can be done accurately since the 3D model obtained is an accurate model of the face. Figure 6b shows the model, textured with the image given in fig. 6a, under a different point of view.

Finally, a few texture images can be used (especially to cover both the front and side views of the face) and averaged into a single texture map.

9. Concluding remarks

We are currently working to obtain a complete automatic framework. Right now our algorithms still require some manual operations, in particular, the user has to point out some features points. They could be detected automatically or even made uncessary for the registration of the texture if we were able to recover the texture from the face images used for reconstruction and where the projected grid pattern is present.

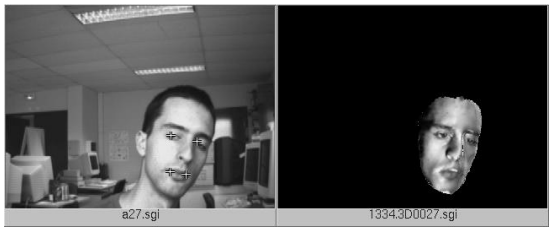


Figure 7: Face tracking with our model

Yet, our scheme for low-cost reconstruction of highly realistic 3D models is already effective. Faces obtained by the preliminary implementation of the proposed algorithms have already been successfully used (Fig. 9) in our face tracking system (described in [9]) although it had originally been designed for models obtained via a 3D scanner (i.e. Cyberware machine).

Apart from the automation of our algorithms, some work must be devoted to correcting the texture with respect to lighting conditions, so that they can be illuminated under any other lighting conditions in future use, and to making the models MPEG-4 compliant. All in all we think we have a promising approach, especially in the way the 3D reconstruction and the texturing are done separately.

Acknowledgements

This work is supported in part by FRANCE TELECOM Research and INRIA Sophia Antipolis (COLOR 2000).

References

[1] *Overview of the MPEG-4 standard*, Technical Report ISO/IEC/JTC1/SC29/WG11 N2725, Int. Organization for Standardization, Seoul Korea, March 1999.

[2] J.-L. Dugelay, S. Valente and K. Fintzel, *Synthetic/Natural Hybrid Video Processings for Virtual Teleconferencing Systems*, Proc. Picture Coding Symp., April 1999.

[3] CYBERWARE Home Page - `http://www.cyberware.com`

[4] A.C. Andrés del Valle and J. Ostermann, *Design of Quickly Adaptable Models for Talking Heads to be used in Interactive Internet Applications*, Internal Report, AT&T Labs research, 2000.

[5] European Project ACTS-092 PANORAMA - `http://www.tnt.uni-hannover.de /project/eu/panorama/`

[6] M. Proesmans and L. Van Gool, *One shot 3D shape and Texture Acquisition of Facial Data*, Audio- and Video-based Biometric Person Authentification, Crans-Montana, March 1997, pp. 411-418.

[7] Zhengyou Zhang, *A Flexible New Technique for Camera Calibration*, Technical Report MSR-TR-98-71, Microsoft Research, 1998.

[8] H. Delingette, *General Object Reconstruction Based on Simplex Meshes*, Int. Journal of Computer Vision 32 (2), 1999, pp. 111-146.

[9] S. Valente and J.-L. Dugelay, *Face Tracking and Realistic Animations for Telecommunicant Clones*, IEEE Multimedia Computing and Systems Magazine, January-March 2000, pp. 34-43.