Regularized Bundle-Adjustment to Model Heads from Image Sequences without Calibration Data

P. Fua
Computer Graphics Lab (LIG)
Swiss Federal Institute of Technology (EPFL)
CH-1015 Lausanne
Switzerland
Pascal.Fua@epfl.ch

International Journal of Computer Vision, 38(2), July 2000

Abstract

We address the structure-from-motion problem in the context of head modeling from video sequences for which calibration data is not available. This task is made challenging by the fact that correspondences are difficult to establish due to lack of texture and that a quasi-euclidean representation is required for realism.

We have developed an approach based on regularized bundle-adjustment. It takes advantage of our rough knowledge of the head's shape, in the form of a generic face model. It allows us to recover relative head-motion and epipolar geometry accurately and consistently enough to exploit a previously-developed stereo-based approach to head modeling. In this way, complete and realistic head models can be acquired with a cheap and entirely passive sensor, such as an ordinary video camera with minimal manual intervention.

We chose to demonstrate and evaluate our technique mainly in the context of head-modeling. We do so because it is the application for which all the tools required to perform the complete reconstruction are available to us. We will, however, argue that the approach is generic and could be applied to other tasks, such as body modeling, for which generic facetized models exist.

1 Introduction

In earlier work, we have proposed an approach to fitting complex head animation models, including ears and hair, to registered stereo pairs and triplets. Here, we extend this approach so that it can take advantage of image sequences taken with a single camera, without requiring calibration data.

Our challenge, here, is to solve the structure from motion problem in a case where

- Correspondences are hard to establish and can be expected to be neither precise nor reliable due to lack of texture.
- A Euclidean or Quasi-Euclidean [Beardsley *et al.*, 1997] reconstruction is required for realism.

- The motion is far from being optimal for most of the auto-calibration techniques that have been developed in recent years.

To overcome these difficulties, we have developed an approach based on bundle-adjustment that takes advantage of our rough knowledge of the face's shape, in the form of a generic face model, to introduce regularization constraints. This has allowed us to robustly estimate the relative head motion. The resulting image registration is accurate enough to use a simple correlation-based stereo algorithm to derive 3–D information from the data. We can then fit a 3–D facial animation mask [Kalra *et al.*, 1992] using our earlier work [Fua and Miccio, 1998, Fua and Miccio, 1999].

We chose to demonstrate and evaluate our technique mainly in the context of head-modeling because it is the application for which we have all the tools required to perform the complete reconstruction task. However, the difficulties discussed above are not specific to head modeling and are pervasive. In that sense the solution we propose is generic: It is applicable to any modeling problem for which a rough shape model is available.

Our contribution is a robust algorithm that takes advantage of our generic knowledge of the shape of the object to be reconstructed, in this work a head, to effectively recover both motion and shape even though the images typically exhibit little texture and are therefore hard to match. Furthermore, this technique has been fully integrated into a complete approach that goes from images to high-quality models with very little manual intervention. Thus, we can create realistic and sophisticated animation models using a cheap, entirely passive and readily available sensor.

As more and more people have video cameras attached to their computers, our approach will be usable to quickly produce clones for video-conferencing purposes. It will also allow the exploitation of ordinary movies to reconstruct the faces of actors or famous people that cannot easily be scanned using active techniques, for example because they are unavailable or long dead.

In the remainder of this paper, we first describe related approaches to relative-motion recovery and head modeling. We then introduce our own approach to registration and demonstrate its robustness using real video sequences. Next, we show reconstructions obtained by using these motion estimates to register the images; deriving 3–D information by treating consecutive images as stereo pairs; and, in the end, fitting the animation mask to the 3–D data. Finally, we use synthetic and Monte Carlo simulations to show that the assumptions we make in this paper can be expected to hold for typical camera configurations. Our earlier fitting procedure [Fua and Miccio, 1999] is described briefly in the appendix.

2 Related Work

2.1 Bundle-Adjustment and Autocalibration

Bundle-adjustment is a well established technique in the photogrammetric community [Gruen and Beyer, 1992]. However, it is typically used in a context, mapping or close-range photogrammetry, where reliable and precise correspondences can be established. Also, because it involves nonlinear optimization, it requires good initialization for proper convergence.

Lately, it has been increasingly used in the computer vision community to refine the output of auto-calibration techniques. There again, however, most results have been demonstrated in man-made environments where feature points can be reliably extracted and matched across images. One cannot assume that those results carry over directly in the case of ill-textured objects and low quality correspondences.

These auto-calibration techniques have been the object of a tremendous amount of work [Faugeras *et al.*, 1992, Hartley *et al.*, 1992, Luong and Viéville, 1996, Triggs, 1997, Pollefeys *et al.*, 1998] and effective methods to derive the epipolar geometry and the trifocal tensor from point correspondences have been devised [Zhang *et al.*, 1995, Fitzgibbon and Zisserman, 1998]. However, most of these methods assume that it is possible to run an interest operator such as a corner detector [Pollefeys *et al.*, 1998, Fitzgibbon and Zisserman, 1998] to extract from one of the images a sufficiently large number of points that can then be reliably matched in the other images. However, when using images such as the ones shown in Figure 1, we cannot depend on such interest points because faces exhibit too little texture. We must expect that whatever points we extract can only be matched with relatively little precision and a high probability of error.

(a)

(b)

Figure 1: Input video sequences: Five out of nine consecutive images of a short video sequences of two different people. The images are of size 376×258 and 488×208 respectively

Autocalibration algorithms tend to be sensitive to such errors, as illustrated by Figure 2. We treated three consecutive images in the video sequence of Figure 1(b) as two independent stereo pairs that share the central image and ran Zhang's image matcher [Zhang *et al.*, 1995] independently on both image pairs. The resulting epipolar geometry is depicted by Figure 2(b,d). These images were acquired in our lab with a relatively long focal length and the head motion was close to being horizontal. Consequently, the epipolar lines should also be almost horizontal and the epipoles should be very far away. The epipolar geometry of Figure 2(b,d) is, therefore, clearly wrong. Of course, we want to stress that this example is not meant to belittle in any way the quality of Zhang's algorithm that has been acknowledged as one of the best of its kind.[1] Visual inspection of the correspondences shows very few mismatches. But, as for most algorithms in this class, even relatively minor matching errors can create major problems.

To some extent, this problem can be alleviated by using more than two images at a time [Beardsley *et al.*, 1997]. However, in our case, this approach can only be of limited use because typical short sequences of moving faces, such as the ones shown in Figure 1, often fail to exhibit rotational motion about two truly independent axes. As a result, the corresponding camera geometries are close to being degenerate for these methods [Sturm, 1997, Zisserman *et al.*, 1998].

[1] It also has the great merit of being freely available on the web and we are prepared to make ours similarly available for testing and comparison purposes.

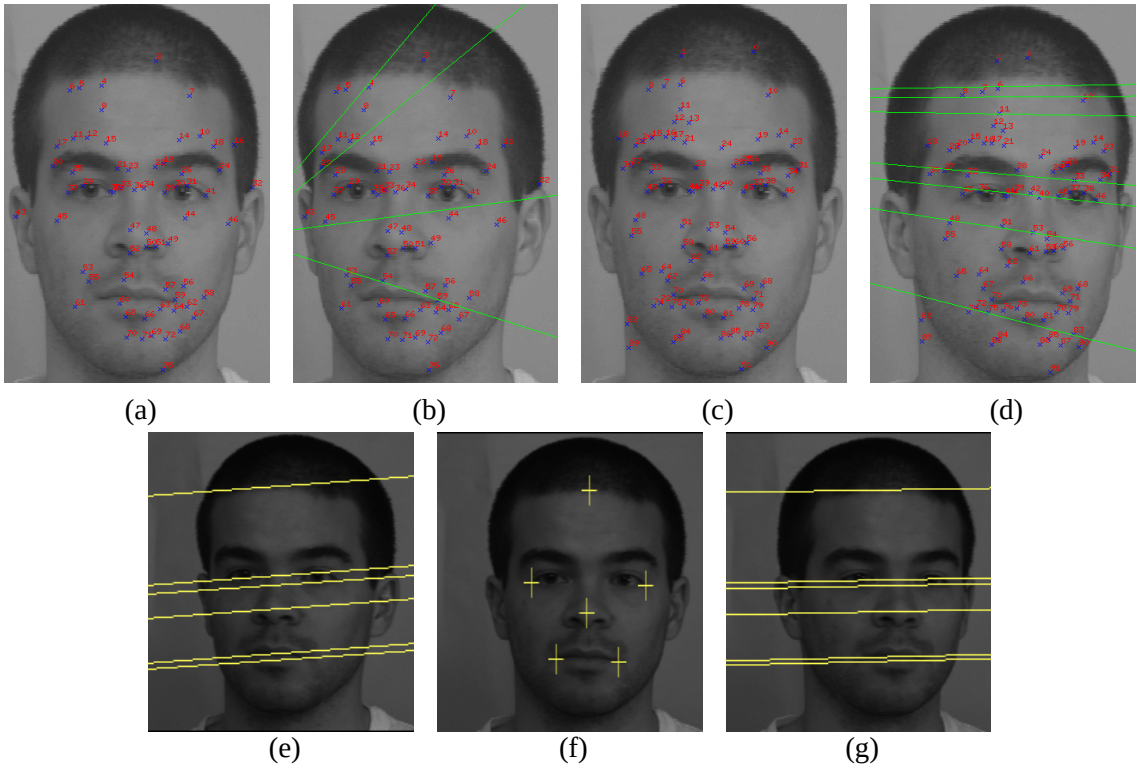


Figure 2: Computing the epipolar geometry without and with a model. (a,b) Running Zhang's algorithm [Zhang *et al.*, 1995] on two consecutive images of the video sequence of Figure 1(b). The matches, shown as numbered crosses, are mostly correct. However, the epipolar geometry, depicted by the solid lines is not. (c,d) The output of an independent run of Zhang's system on a different image pair. (e,f,g) The epipolar geometry recovered by the algorithm described in this paper. The lines in (e,g) are the epipolar lines that correspond to the crosses in (f).

In short, while the structure-from-motion problem is well understood from a theoretical point of view, model-free techniques are too sensitive to noise to be directly applicable both to our specific problem and to all modeling tasks that involve the difficulties described in the introduction.

In the case of head tracking, a generic 2–D face model can be used to estimate roughly estimate pose from appearance [Lanitis *et al.*, 1995]. However for 3–D reconstruction purposes and more precise estimation, using a 3–D model is, in general, more effective. Jebara and Pentland [1997] introduce shape constraints based on allowable deformation modes derived from a collection of Cyberware^{tm} scans of real heads. When such a database is available, this certainly is an effective approach. However, to make it fully general, one would require large number of instances of the target object, making it difficult to derive in practice.

By contrast, in this work, we will show that a simple, and easily obtainable, facetized model can be used to derive effective shape constraints. These are key to a practical solution of our reconstruction problem. As shown in Figure 2(e,f,g), by using these constraints, we can recover a consistent epipolar geometry. This is crucial for us because we treat consecutive images in a sequence as stereo pairs that provide the 3-D data required to compute the results of Section 4.

2.2 Head Modeling

In recent years much work has been devoted to modeling faces from image and range data. There are many effective approaches to recovering face geometry. They rely on stereo [Devernay and Faugeras, 1994, Fua and Leclerc, 1995], shading [Leclerc and Bobick, 1991, Samaras and Metaxas, 1998], structured light [Proesmans *et al.*, 1996], silhouettes [Tang and Huang, 1996] or low-intensity lasers. However, if the goal is to fit a full animation model to the data, recovering the head as a simple triangulated mesh does not suffice. To be suitable for animation, such a model must have a large number of degrees of freedom. Some approaches use very clean data—the kind produced by a laser scanner or structured light—to instantiate them [Lee *et al.*, 1995]. Among approaches that rely on image data alone, many require extensive manual intervention, such as supplying silhouettes in orthogonal images [Lee and Thalmann, 1998] or point correspondences in multiple images [Pighin *et al.*, 1998].

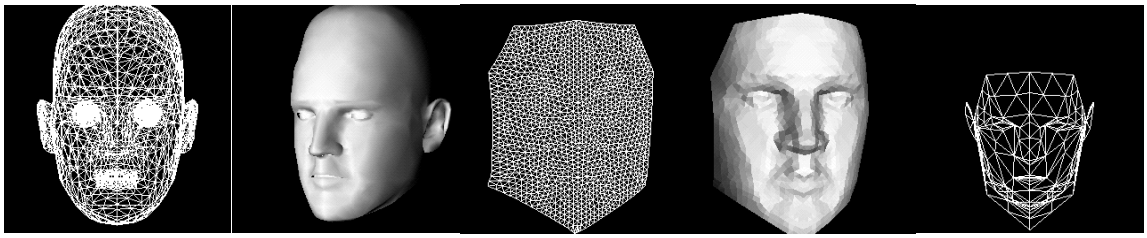
Successful approaches to automating the fitting process have involved the use of optical flow [DeCarlo and Metaxas, 1998] or appearance based techniques [Kang, 1997] to overcome the fact that faces have little texture and that, as a result, automatically and reliably establishing correspondences is difficult. This latter technique is closely related to ours because head shape and camera motion are recovered simultaneously. However, the optical flow approach avoids the "correspondence problem" at the cost of making assumptions about constant illumination of the face that may be violated as the head moves. This tends to limit the range of images that can be used, especially if the lighting is not diffuse.

More recently, another extremely impressive appearance-based approach that uses a sophisticated statistical head model has been proposed [Blanz and Vetter, 1999]. This model has been learned from a large database of human heads and its parameters can be adjusted so that it can synthesize images that closely resemble the input image or images. While the result are outstanding even when only one image is used, the recovered shape cannot be guaranteed to be correct unless more than one is used. Because the model is Euclidean, initial camera parameters must be supplied when dealing with uncalibrated imagery. Therefore, the technique proposed here could be used to initialize the Blanz & Vetter system in an automated fashion. In other words, if we had had their model, we could have used it to develop the technique described here. However, for practical reasons, it was not available. Instead, we used the model described below.

2.3 Face Model

In this work, we use the facial animation model that has been developed at University of Geneva and EPFL [Kalra *et al.*, 1992]. It can produce the different facial expressions arising from speech and emotions. Its multilevel configuration reduces complexity and provides independent control for each level. At the lowest level, a deformation controller simulates muscle actions using rational free form deformations. At a higher level, the controller produces animations corresponding to abstract entities such as speech and emotions.

The corresponding skin surface is shown in its rest position in Figure 3(a,b). We will refer to it as the *surface triangulation*. Our goal is to deform the surface without changing its topology. This is important because the facial animation software depends on the model's topology and its configuration files must be recomputed every time it is changed, which is hard to do on an automated basis.

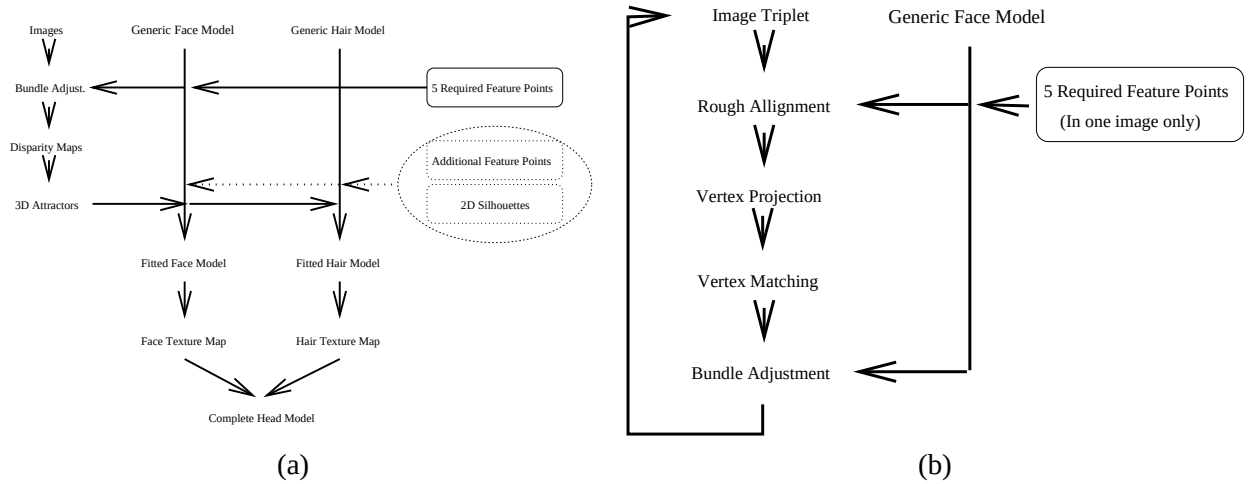


(a) (b) (c) (d) (e)

Figure 3: Animation model. (a) Wireframe model used to animate complete heads. (b) Shaded view of the model (c) Regular sampling of the face used to perform bundle adjustment. (d) Shaded view of the resampled triangulation. (e) Control triangulation used to deform the face.

3 Relative Motion Recovery

Our complete approach to head-modeling is summarized by Figure 4(a). Our earlier work [Fua and Miccio, 1998, Fua and Miccio, 1999] assumed calibrated images. In this section, we focus on the implementation and evaluation of a regularized bundle-adjustment technique that allows us to perform this task in the absence of calibration data. Figure 4(b) depicts graphically the steps of this procedure and we describe them in detail below.



(a) (b)

Figure 4: Head modeling procedure: (a) Flow chart of the whole procedure. The manually and semi-automatically entered data appears on the right hand-side. The location of 5 2-D points must be supplied, the rest is optional. The dotted rectangle in the diagram's upper left side encompasses the regularized bundle-adjustment procedure that this paper focuses on. It is depicted in more details in (b). (b) Flow chart of the bundle-adjustment procedure itself as described in Section 3.

Here, we choose sequences in which the subjects keep a fairly neutral expression and we treat the head as a rigid object. We assume that the intrinsic camera parameters remain constant throughout the sequence. In theory, given high precision matches, bundle-adjustment can recover both intrinsic parameters and camera motion [Gruen and Beyer, 1992]. The same holds true for recent auto-calibration techniques but, as discussed in Section 2.1, typical video sequences of head images are close to exhibiting degenerate motions as far as these techniques are concerned [Sturm, 1997, Zisserman *et al.*, 1998]. There again, extremely precise matches would be required.

In practice, however, face images exhibit little texture and we must be prepared to deal with the potentially poor quality of the point matches. Therefore, we have chosen to roughly estimate the intrinsic parameters and to concentrate on computing the extrinsic ones using bundle-adjustment: We use an approximate value for the focal length and assume that the principal point remains in the center of the image.

By so doing, we generate 3–D models that are deformed versions of the real heads. When the motion between the camera viewpoints is a pure translation or the cameras are assumed to be orthographic, this deformation can be shown to be an affine transform [Luong and Viéville, 1996]. In the general case, however, the use of approximate internal parameters causes violations of the epipolar geometry. It has been shown [Baratoff and Aloimonos, 1998] that reconstruction errors caused by such violations can, in some cases, be modeled as a Cremona transform, which is quadratic.

In this section, we use real video sequences to show that, in practice, the deformation is still adequately modeled by an affine transform. In Section 5, we will use synthetic data and Monte Carlo simulations to verify that, for typical camera configurations, modeling the deformation as an affine one is, in general, an excellent approximation. Furthermore, the closer the approximate value of the focal length to its true value, the closer that affine transform is to being a simple rotation, translation and scaling. In other words, the fact that our results are precise up to an affine transform that does not deform the face severely is no accident and does not depend on the specific geometry of the sequences used here. We will also argue that the relatively poor quality of the matches that can be obtained on a face cause imprecisions in motion recovery that are at least as severe as those produced by the use of approximate intrinsic camera parameters. Therefore, there is no compelling reason to use a more sophisticated approach to camera model recovery—for example, one that can also recover the internal camera parameters, as long as the precision of the point matches cannot be drastically improved.

3.1 Initialization

One well known limitation of bundle-adjustment algorithms is the fact that, in order to ensure convergence, one must provide initial values for both camera positions and the x, y, and z coordinates, which should not be too far from their true values.

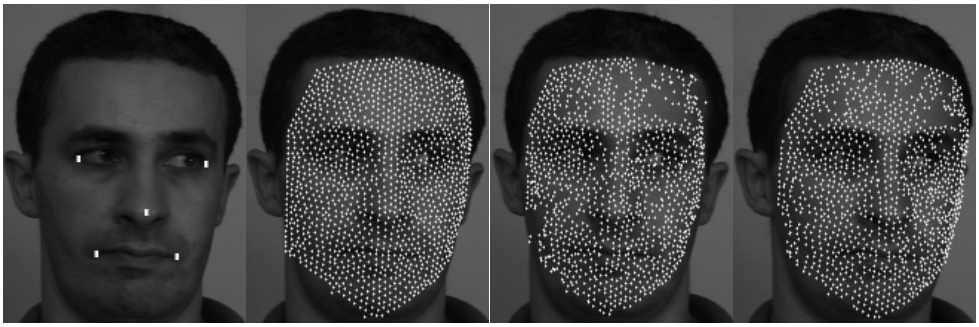
To initialize the process for a video sequence such as the ones depicted by Figure 1, we manually supply the approximate 2–D location of five feature points in one reference image: nose tip, outer corners of the eyes and outer mouth corners, as shown in Figure 5(a). Note that we supply 2–D locations in one image *only*, which is very easy, as opposed to trying to manually register the points, which would require much more work if one wanted to do so with any degree of accuracy.

We usually choose as our reference image one in which the subject faces the camera and we take this image to be image number one. The system then automatically finds the position and orientation of the central camera that brings the keypoints' projections as close as possible to those positions.

This guarantees that points on the 3–D face model roughly project on the imaged face. We then estimate the positions and orientations for the two images on either side of the central image, as described in the following section.

3.2 Bundle-Adjustment using Approximate Internal Parameters

To estimate the positions and orientations for the two images on either side of the central image, we begin by retriangulating the surface of the generic face model of Figure 3(a,b) to produce the regular mesh depicted



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

Figure 5: Tie points: (a) The five manually supplied keypoints used to compute the orientation of the first camera. (b) The projections of the vertices of the bundle-adjustment triangulation of Figure 3. (c) into the central image of Figure 1(a). (c,d) Matching points in the images immediately following and immediately preceding the central image of Figure 1(a) in the sequence. These correspondences have been computed using a simple correlation algorithm and take visibility constraints for the central image into account.

by Figure 3(c,d) that we call the *bundle-adjustment triangulation*. As shown in Figure 5(b), our initial orientation choice guarantees that the bundle-adjustment triangulation's vertices projections roughly fall on the face. We match these projections into the other images using a simple correlation-based algorithm [Fua, 1993]. Figure 5(c,d) depicts the results. For each of these *tie* points, we write two *observation equations*:

$$\begin{aligned} Pr_u^j(x_i, y_i, z_i) &= u_i^j + \epsilon_{u_i^j} \\ Pr_v^j(x_i, y_i, z_i) &= v_i^j + \epsilon_{v_i^j} \end{aligned} \tag{1}$$

where (u_i^j, v_i^j) is the expected projection of point i in image j, that is, one of the white dots in Figure 5(b,c,d); $Pr_u^j(x, y, z)$ and $Pr_v^j(x, y, z)$ denote the two image coordinates of the actual projection of point (x_i, y_i, z_i) in image j using the current estimate of the camera models; and, $\epsilon_{u_i^j}, \epsilon_{v_i^j}$ the projection errors to be minimized. For each j, Pr_u^j and Pr_v^j depend on the six external parameters that define the camera position and orientation.

Given n tie points that project in the three images, the position parameters can be recovered by minimizing the observation errors in the least square sense, that is by minimizing the objective function $\mathcal{E}$:

$$\begin{aligned} \mathcal{E} &= \sum_{1 \le i \le n} w_i e_i \\ e_i &= \sum_{1 \le j \le 3} \delta_i^j ((Pr_u^j(x_i, y_i, z_i) - u_i^j)^2 + (Pr_v^j(x_i, y_i, z_i) - v_i^j)^2)) \ , \end{aligned} \tag{2}$$

with respect to six external parameters of each camera and of the x, y and z coordinates. δ_i^j is either zero or one depending on whether the tie point i is visible in image j or not. w_i is a weight associated to point i. It is initially taken to be 1 for all points and is then adjusted as discussed below. The solution can only be found up to a global rotation, translation and scaling. To remove this ambiguity, we fix the position of the first camera and one additional parameter such as the distance of one vertex in the triangulation. The unknown parameters therefore are the 12 external parameters of the second and third camera and the n x, y and z coordinates.

In practice, we represent the camera models for all the cameras as 3x4 projection matrices. The algorithm goes through the following steps:

1. Generate an initial 3x4 projection matrix for the first camera with the principal point in the center of the image.

$$Prj = \left| \begin{array}{cccc} \text{f'} & 0 & \text{xdim/2} & 0 \\ 0 & \text{f'} & \text{ydim/2} & 0 \\ 0 & 0 & 1 & 0 \end{array} \right| , \quad (3)$$

   where $xdim, ydim$ are the image dimensions and f' is an approximation of the true camera constant, i.e. focal length times a scale factor, f, expressed here in pixels per millimeter.

2. Compute a 4x4 rotation-translation matrix M such that the five keypoints of Figure 5(a) once multiplied by this matrix project as close as possible to the hand-picked locations in the central image. The 3x4 matrix that represents the camera model for the first image is then be taken to be $Tr = Prj * M$, ensuring that the five keypoints project in the vicinity of the five hand-picked locations. As a consequence, all the other vertices of the bundle-adjustment triangulation also project on the face, as shown in Figure 5(b). These projections are taken to be the u_i^1, v_i^1 of Equation 2. We then match the points that are expected to be visible in the two images that immediately precede and succeed the central image in the video sequence. We use a simple correlation-based algorithm [Fua, 1993] to obtain the $(u_i^j, v_i^j)_{2 \leq j \leq 3}$ of Equation 2. Figure 5(c,d) depicts the results. Note that not all the points are matched and that there is a number of erroneous matches.

3. Take the initial positions of the other cameras to be equal to that of the first.

4. Given these initial values, use the Levenberg-Marquardt algorithm [Press *et al.*, 1986] to minimize the objective function $\mathcal{E}$ of Equation 2 with respect to the camera positions and the tie points' 3–D coordinates.

This yields the camera models for the two images on either side of the central image and an estimate of the bundle-adjustment triangulation's shape. To compute the following camera positions, the image immediately succeeding the central image becomes the new central image. We project the bundle-adjustment triangulation's vertices into it, compute the matching points in the image that follows in the sequence and rerun the bundle-adjustment algorithm to compute the position of the corresponding camera. We then iterate until the end of the sequence. We proceed similarly for the images that precede the central image in the sequence.

3.3 Robust Bundle-Adjustment

The procedure outlined above is generic bundle-adjustment [Gruen and Beyer, 1992], with fixed intrinsic parameters. If the correspondences were perfect, this would be sufficient to retrieve the motion parameters. However the point correspondences can be expected to be noisy and to include mismatches. To increase the robustness of our algorithm, we augment the standard procedure in two ways:

1. Iterative reweighted least squares. Because some of the point matches may be spurious, we use a variant of the Iterative Reweighted Least Squares [Beaton and Turkey, 1974] technique. We first run the bundle adjustment algorithm with all the weights w_i of Equation 2 set equal to 1. We then recompute these weights so that they are inversely proportional to the final residual errors. We minimize our criterion again using these new weights and iterate the whole process until the weights stabilize. This typically takes three iterations. More specifically, for each tie point, we compute the average residual error ϵ_i:

$$\frac{\sum_j \delta_i^j (Pr_u^j(x_i, y_i, z_i) - u_i^j)^2 + (Pr_v^j(x_i, y_i, z_i) - v_i^j)^2}{\sum_j \delta_i^j} \quad .$$

We then take w_i to be $exp(\frac{-\epsilon_i}{\overline{\epsilon_i}})$, where $\overline{\epsilon_i}$ is the median value of the ϵ_i for $1 \leq i \leq n$. In effect, we use $\overline{\epsilon_i}$ as an estimate of the noise variance and we discount the influence of points that are more than a few standard deviations away.

2. Regularization. The tie points are the vertices of our bundle-adjustment triangulation and represent a surface that is known to be smooth. We prevent excessive deformation by adding a regularization term $\mathcal{E}_D$ to the objective function $\mathcal{E}$ of Equation 2.

To compute this term, we first rewrite our observation equations as:

$$\begin{aligned} Pr_u^j(x_i + dx_i, y_i + dy_i, z_i + dz_i) &= u_i^j \\ Pr_v^j(x_i + dx_i, y_i + dy_i, z_i + dz_i) &= v_i^j \end{aligned} \tag{4}$$

where x_i, y_i, z_i are the vertex coordinates that are now fixed and dx_i, dy_i, dz_i are displacements that become the actual optimization variables.

If the surface were continuous, we could take $\mathcal{E}_D$ to be the sum of the square of derivatives of the dx_i, dy_i and dz_i across the surface. We approximate this by treating the bundle-adjustment triangulation's facets as C^0 finite elements and evaluating $\mathcal{E}_D$ as follows. We introduce a stiffness matrix K such that

$$\mathcal{E}_D = 1/2(dX^t K dX + dY^t K dY + dZ^t K dZ) \tag{5}$$

approximates [Zienkiewicz, 1989] the sum of the square of the derivatives of displacements across the triangulated surface when dX, dY, and dZ are the vectors of the dx_i, dy_i and dz_i. In other words, $\mathcal{E}_D$ is the discretization of

$$\iint \left[\left(\frac{\partial dX(u,v)}{\partial u} \right)^2 + \left(\frac{\partial dY(u,v)}{\partial v} \right)^2 + \left(\frac{\partial dZ(u,v)}{\partial v} \right)^2 \right] du\, dv \quad .$$

We can now enforce smoothness by minimizing the regularized objective function $\mathcal{E}_T$:

$$\begin{aligned} \mathcal{E}_T &= \lambda \mathcal{E}_\mathcal{D} + \sum_i w_i e_i \\ e_i &= \sum_j \delta_i^j ((Pr_u^j(x_i + dx_i, y_i + dy_i, z_i + dz_i) - u_i^j)^2 \\ &\qquad + (Pr_v^j(x_i + dx_i, y_i + dy_i, z_i + dz_i) - v_i^j)^2) \quad , \end{aligned} \tag{6}$$

where λ is a smoothing coefficient. As will be shown below, the result is not very sensitive to the exact choice of λ and we take it to be 1.0 by default.

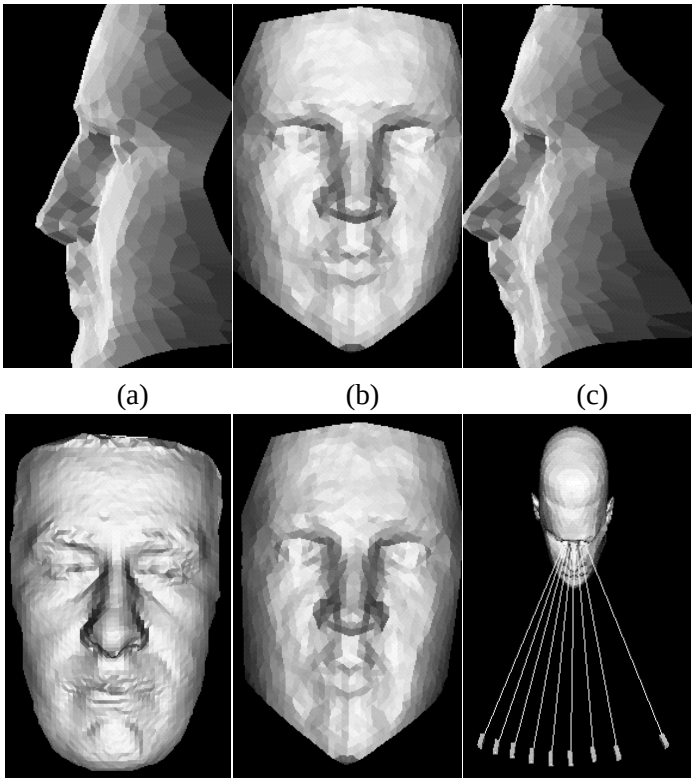


Figure 6: Comparison with laser scanner data: (a) Profile view of the bundle-adjustment triangulation of Figure 3(c,d) *before* adjustment. (b,c) Frontal and profile view of the bundle-adjustment triangulation after bundle adjustment but *before* stereo reconstruction. (d) Laser scanner output. (e) Bundle-adjustment triangulation of (b,c), affine transformed to be as close as possible, in the least-squares sense, to (d), the scanner's output. The deformation with respect to (b,c) is mild and close to being a simple scaling along the coordinate axes. (f) Recovered relative camera positions for the whole sequence.

In Section 5 we will use synthetic data to show that this "regularized bundle-adjustment" works well in the presence of both significant tie points coordinates imprecision and of rates of mismatches of up to 30 %, whereas the standard implementation of bundle-adjustment breaks down completely. In the case of the image triplet of Figure 5, our procedure yields the bundle-adjustment triangulation depicted by Figure 6(b,c).

To quantify our result's quality, we have used a Minolta^{tm} laser scanner to acquire the model of the same head shown in Figure 6(d). The theoretical precision of the laser is approximately 0.3 millimeters and it can therefore be considered as a reasonable approximation of ground truth. Of course, in practice, even the laser exhibits a few artifacts. However, since the two reconstruction methods are completely independent of one another, places of agreement are very likely to be correct for both.

To show that the deformation induced by our arbitrary choice of internal camera parameters is indeed close to being as an affine transformation, we have computed the affine transform A that brings the bundle-adjustment triangulation closest to the laser-scanner model. As shown in Figure 6(e), the deformation introduced by the affine transform is relatively mild. In fact, it can be closely approximated by a scaling along each of the coordinate axes. Note, however, that the resulting mask still does not seem very realistic: This is to be expected since we only gather information at the vertices of the bundle-adjustment triangulation and nowhere else. It is for this reason that the further fitting step of Section 4 is required to exploit all the

available stereo data.

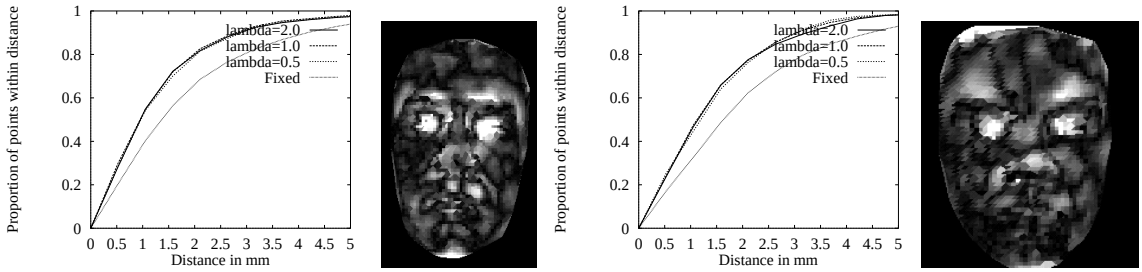


Figure 7: Quantitative evaluation of the bundle-adjustment algorithm. (a) For the subject of Figure 1(a), proportion of the 3–D points in the laser output that are within a given distance of the bundle-adjustment triangulation, after it has been deformed using an affine transform. We show three almost superposed curves corresponding to three different values of the regularization parameter λ. The fourth curve depicts the equivalent result obtained by fixing the 3–D location of the bundle-adjustment triangulation's vertices. (b) Graphic depiction of the distances. The surface of the laser output is intensity-coded so that the white areas are those that are furthest away from the bundle-adjustment triangulation. White corresponds to an error greater than 5 millimeters. (c,d) Same thing for the subject of Figure 1(b)

For a more quantitative estimate, we plot the proportion of 3–D points in the laser output that are within a given distance of the deformed bundle-adjustment triangulation. We have performed this computation for three different values—0.5, 1.0, and 2.0—of the regularization parameter λ. The resulting plot appears in Figure 7(a). We have also repeated this entire procedure for the subject of Figure 1(b), resulting in the graphs of Figure 7(c). Note that:

1. For both faces, the three curves are essentially superposed, indicating the relative insensitivity of the bundle-adjustment procedure to the value of λ. The corresponding median reprojection errors are in the order of 0.25 pixels, which guarantees a good and consistent epipolar geometry.

2. The median distance in all cases is approximately 1 millimeter which, given the camera geometry, corresponds to a shift in disparity of less than 1/5 a pixel. These distances for $\lambda = 1.0$ appear in the first column of Table 1. The precision of the correlation based algorithm we use is in the order of half a pixel, outliers excluded [Fua, 1993]. We therefore conclude that our bundle adjustment algorithm performs an effective and robust averaging of the input data.

3. However, if λ becomes too large, the bundle-adjustment triangulation becomes rigid. Its shape cannot adapt to match that of the face; the reprojection errors of Equation 2 become too big to guarantee a satisfactory epipolar geometry. To illustrate this point, in each graph, we have plotted a fourth curve that corresponds to the result obtained by fixing the bundle-adjustment triangulation's shape instead of allowing it to deform. The reprojection errors grow much larger (> 1 pixel) and the resulting epipolar geometry stops being good enough for high quality Euclidean reconstruction.

To estimate numerically the severity of the shape deformation provoked by our procedure we proceed as follows: Given the the affine transform A represented by a 4×4 matrix, we perform a singular value decomposition of its rotational component. This yields $U^t DV$ where U and V are 3×3 rotation matrices and D is a 3×3 diagonal matrix. If A were a pure rotation, translation and scaling, all the eigenvalues of D

$f' = 4200$	Median dist. after Bundle-Adjust.	Deformation after Bundle-Adjust.	Median dist. after Model Fitting	Deformation after Model Fitting
Head of Fig. 1(a)	0.90 mm	1.19	0.83 mm	1.13
Head of Fig. 1(b)	1.10 mm	1.09	0.92 mm	1.14

Table 1: Metric comparison against the scanner's output. (Column 2) Median distance of the bundle-adjustment triangulation's vertices to the scanned surface after affine transformation. (Column 3) Measure of the deformation produced by this affine transformation. (Column 4,5) Equivalent measures for the final surface-triangulation.

would be equal to the scaling factor. Thus, we take the ratio of D's maximum to minimum eigenvalue to be a measure of how much A deforms the shape. The results are shown in the second column of Table 1.

In the examples shown here, we have systematically taken the approximate focal length of Equation 3 to be $f' = 4200$. Figure 8 shows the effect of changing the value of f'. Using the images of Figure 5, we performed the same computation for values of f' ranging from 3500 to 5000. We plot here

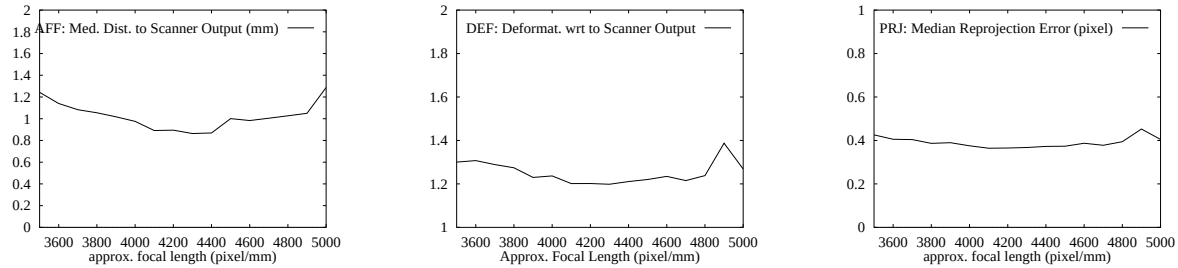


Figure 8: Changing the value of f' in the bundle-adjustment procedure. **AFF**, **DEF**, and **PRJ**, are described at the end of Section 3.3. They are shown here as functions of f' in (a),(b), and (c), respectively.

AFF : the median distance in mm of the scanner output vertices to the affine transformed triangulation;

DEF : the severity of the shape deformation, measured as described above;

PRJ : the reprojection error in pixel, that is, the median distance of the bundle-adjustment triangulation vertices's projections to the expected values, such as the white dots' location in Figure 5.

For verification purposes, we have calibrated—without using the calibration data in *any* of our computations—the camera used to acquire the sequences of Figure 5 using the INRIA CamCal package[Tarel and Vezien, 1996]. It yields a value of f approximately equal to 4000. It is only approximate because, CamCal, in effect, can also trade changes in the value of f against changes of the estimated distance of the camera to the calibration grid it uses.

In any event, the curves that appear in Figure 8 are relatively flat in the vicinity of $f = 4000$ and the corresponding values of **AFF**, **DEF**, and **PRJ**, are entirely consistent with those found by running Monte Carlo simulations, as will be seen in Section 5.

By repeating this computation over all overlapping triplets of images in the video sequences we can compute all the camera positions depicted by Figure 6(f). To ensure shape consistency across the sequence, given a new triplet, we fix the camera positions for the two images that belong to a previous triplet and allow only the third one to move. In effect, for each new image, we compute the camera position only once

and, then, fix it. In theory, once all the images have been registered, we could rerun our bundle adjustment procedure using all images simultaneously but we have not found this to be necessary in practice.

4 Refined Head Models

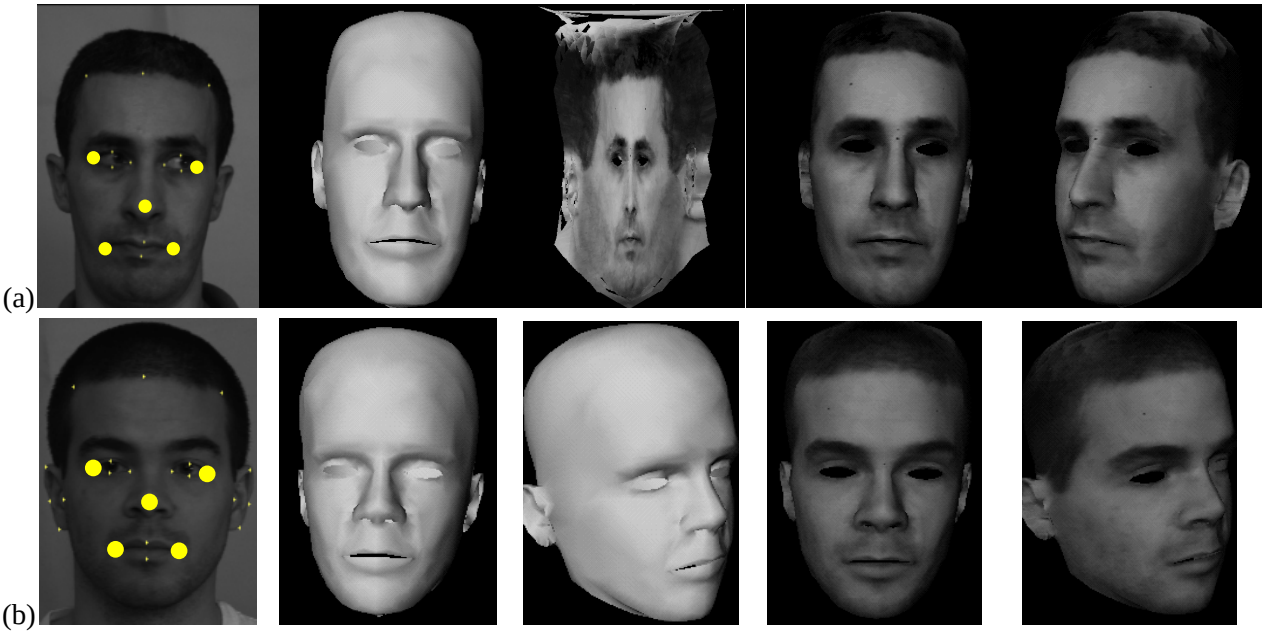


Figure 9: Fitting the complete animation mask. (a) For the subject of Figure 1(a): The manually supplied 2–D features points; shaded view of the complete head model; the cylindrical texture map; and, textured model. (b) For the subject of Figure 1(b), the manually supplied 2–D features points; two shaded and two textured views of the head model.

Given the camera models computed above, we can now recover additional information about the surface. We use a simple correlation-based algorithm [Fua, 1993] to compute a disparity map for each pair of consecutive images in the video sequences. We then turn each valid disparity value into a 3–D point and fit our animation mask to these 3–D points by minimizing an objective function. This approach is depicted by Figure 4(a) and summarized in the appendix. For additional details, we refer the interested reader to our earlier publications [Fua and Miccio, 1998, Fua and Miccio, 1999]. Alternatively, we could have used an appearance-based technique [Kang, 1997, Blanz and Vetter, 1999] that optimizes shape and camera position simultaneously. Because such approaches perform a gradient-style minimization, a good starting point such as the one our algorithm provides would almost certainly be helpful.

Figure 9 depicts the final models for the two video sequences we have used so far. To ensure that some of the key elements of the face—corners of the eyes, mouth and hairline—project at the right places, we have manually supplied the location of the projection in one image of a few feature points such as the ones shown in the first column of Figure 9: Our objective function incorporates a term that forces the projection of the generic mask's corresponding vertices to be close to them [Fua and Miccio, 1998, Fua and Miccio, 1999]. Note that the five manually supplied points used to initialize the bundle-adjustment procedure of Section 3.1, shown as disks, form a subset of these feature points. In practice, we supply all these points initially and then let the system run automatically. To produce these face models, the manual intervention

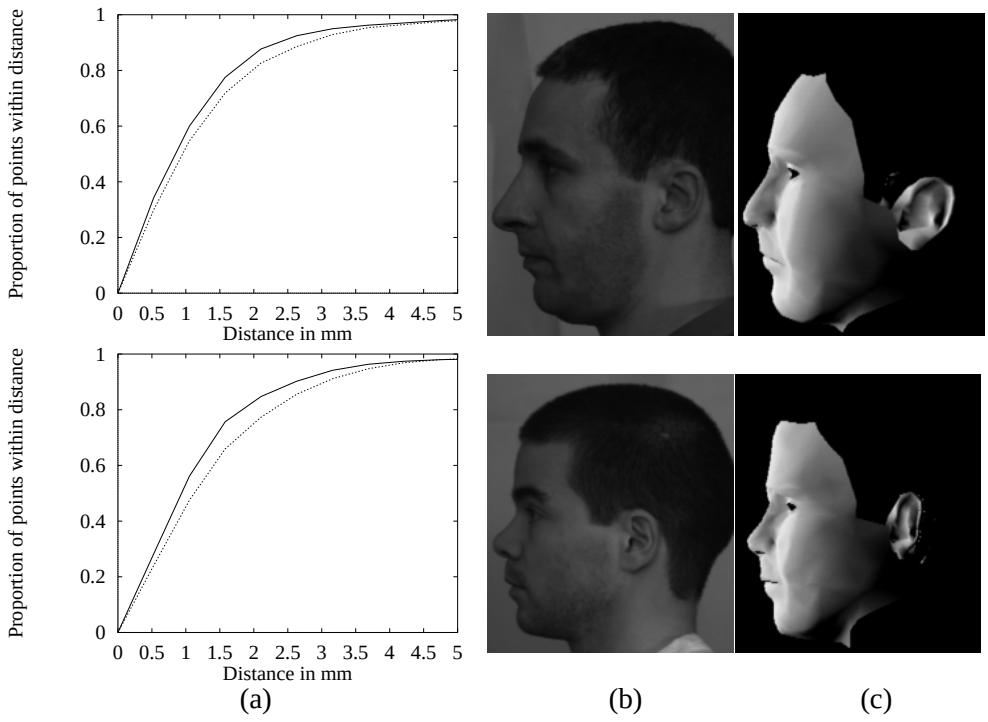


Figure 10: Quantitative and qualitative evaluation of the reconstructions' quality for both subjects of Figure 1. (a) Proportion of 3–D laser points that are within a given distance of the affine-transformed face surface. In both graphs, the solid line corresponds to the final head model and the dotted one to the bundle-adjustment triangulation for $\lambda = 1.0$. (b) Profile images of both subjects that has *not* been used to perform the computation. (c) The corresponding face model shown in a similar pose.

required therefore reduces to supplying these few points by clicking on their approximate locations in one, and only one, image, which can be done quickly.

Because stereo tends to fail in the hair regions, the shape of the top of the head has been recovered by semi-automatically delineating in each image of the video sequence the boundary between the hair and the background and treating it as a silhouette that constrains the shape [Fua and Miccio, 1998]. Given the final head model, the algorithm creates a cylindrical texture map, such as the one shown in the first row of Figure 9.

In Figure 10(a), we use again the Minolta^{tm} laser scanner output to evaluate the reconstruction's quality. As in Section 3.3, we compute the affine transform that best maps the reconstructed face onto the laser output and plot the proportion of 3–D laser points that are within a given distance of the deformed face surface. In both cases, the corresponding curves appears as a solid black line. For comparison's sake, we also plot as a dotted line the corresponding distribution for the bundle-adjustment triangulation in the case $\lambda = 1.0$. As expected, using additional stereo data has brought an improvement as evidenced by the fact that the solid curve is above the dotted one. In the third column of table 1 we indicate the corresponding median values of the distances.

To evaluate these results qualitatively, in Figure 10, we show two side views of the heads that have *not* been used to perform the computation and show the reconstructed models seen in a similar pose. Note that the face outlines corresponds quite accurately except where stereo can be expected to fail because the surface slopes away from the camera: Bottom of the nose and of the chin.

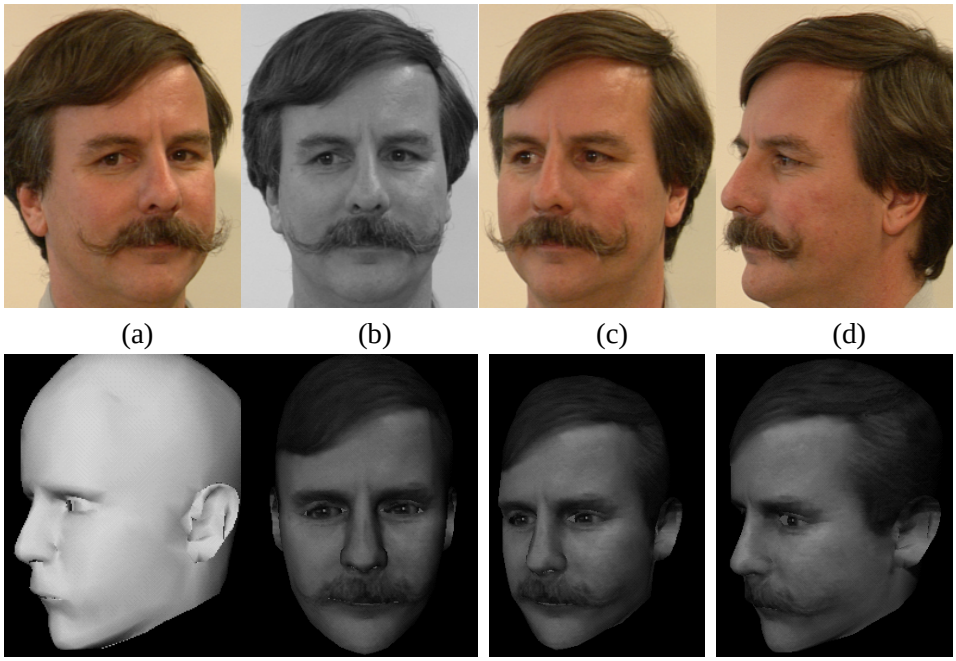


Figure 11: A man with a mustache. (a,b,c) Three color images of a sequence of seven. (d) A profile view that was not used during the computation. (e) Shaded view of the reconstructed model. The points on the mustache are treated as outliers and ignored. (f,g,h) Texture-mapped views of the head model. In this and following figures, the eye texture comes from a standard library and could be improved by using the actual images of the character's eyes.

In Figures 11 and 12 we show additional models reconstructed from images acquired with a different camera, a color one, but using the same parameters for our algorithm. All these models can be animated to produce synthetic expressions such as the ones shown in the first two rows of Figure 12. These video sequences and 3–D models are available on our website ligwww.epfl.ch/f̃ua/faces/.

To check the applicability of our technique to reconstructing famous actors from old movies, we scanned a number of scenes from "Key Largo" with H. Bogart and L. Bacall and "Queen Christine" with G. Garbo. Obviously, most of the shots cannot be used directly by the technique described in this paper because actors talk or change their facial expressions. However, all we need is a short clip, about one second, in which the actor turns his head without speaking. In both cases, we were able to find and process such clips to generate models. Unfortunately, we cannot show these results because the movie companies have not granted us a license to use the images for publication purposes. Nevertheless, an interesting extension of this work would be to use these 3–D animation models to track face's deformation as actors express their emotions and, thus, learn the animation parameters that could be used to synthesize personalized expressions.

In future work, we plan to extend this technique to whole body modeling, as illustrated by Figure 13 and 14. We found these stereo pairs on the web. We manually, and very roughly, aligned a generic female torso model with the first image of each one. We then used this model as both our bundle-adjustment triangulation and our surface triangulation: We projected its vertices onto these first images, computed matches in the second and used our regularized bundle-adjustment technique to compute the relative position of the camera. We then used this information to rectify the images, compute a disparity map, generate 3–D points and deform the generic model. In the absence of ground truth, it is difficult to judge the quality of

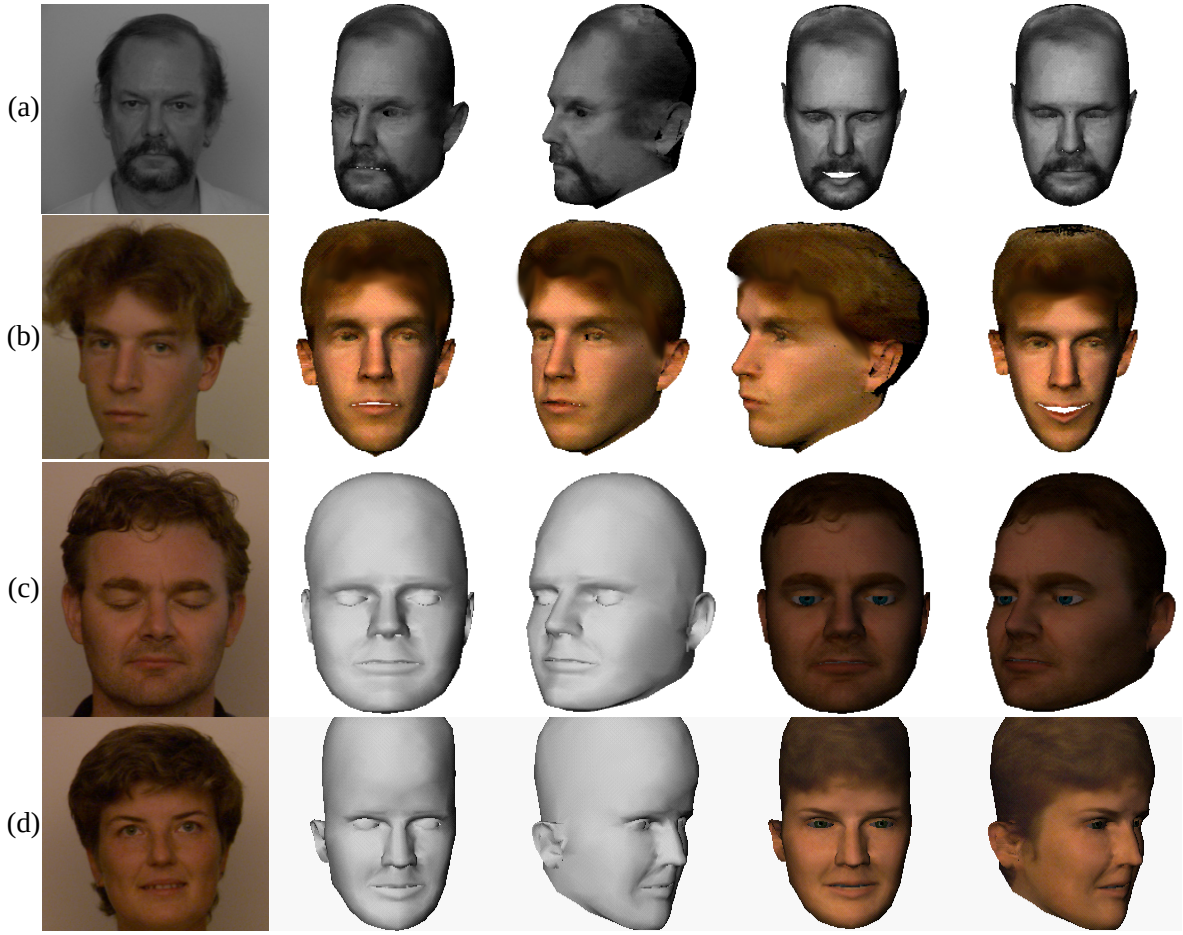


Figure 12: Reconstruction and animation. (a) Central image of a sequence, two texture-mapped views of the corresponding head model; and two synthetic expressions, opening of the mouth and closing of the eyes. (b) Central image of another sequence; three views of the corresponding head model; and a synthetic smile.(c,d) Reconstructed models for two additional people

the result but the shape appears to be reasonable and, once texture-mapped, would look realistic. This is very encouraging given the fact that, on these image pairs, Zhang's algorithm [Zhang *et al.*, 1995] extracts very few matches and puts the epipoles within the images, which is clearly not realistic. To extend this to full body modeling from video sequences, we intend to use an articulated body model [Boulic *et al.*, 1995, Thalmann *et al.*, 1996]. To account for the fact that the person can move, we will treat the joint angles as variables in the regularized bundle-adjustment minimization.

5 Validity of our Deformation Model

There are two main sources of errors in our reconstruction procedure:

1. We use approximate internal parameters for the cameras.
2. The point matches we use as input to our bundle-adjustment procedure can be imprecise or even wrong.

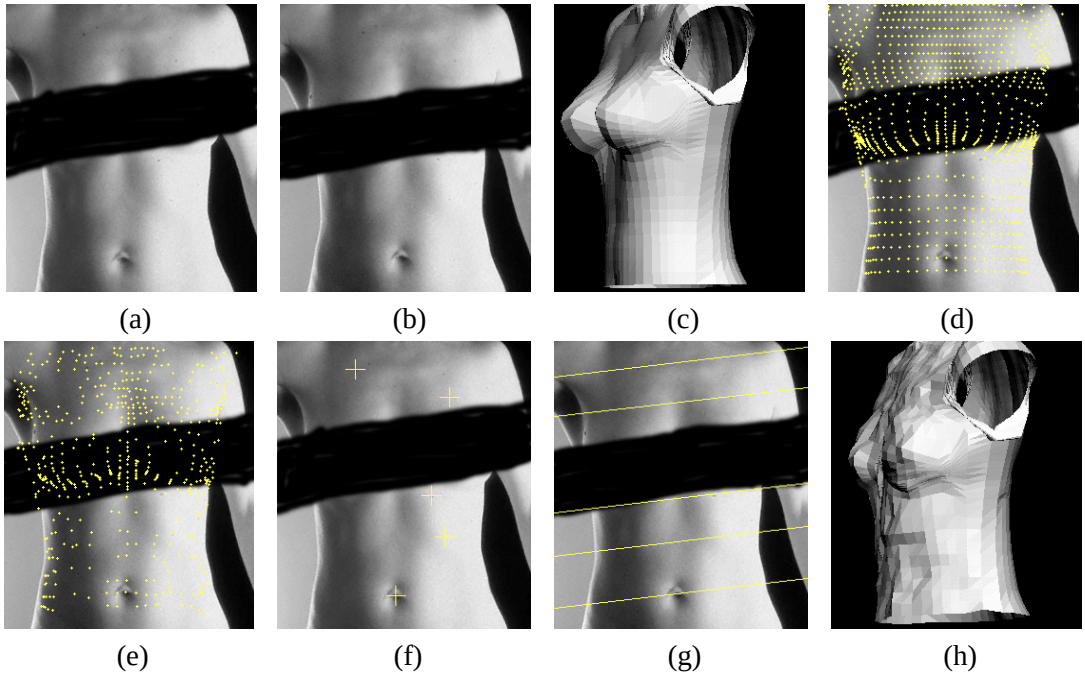


Figure 13: Body Modeling. (a,b) A stereo pair found on the world wide web. The diagonal black bands have been added after the fact so that these images can be published. (c) Generic female torso. (d) Projection of the generic torso's vertices after rough alignment. Note that the alignment is far from perfect. (e) Matching points in the second image. Because the skin is fairly smooth, there are relatively few of them. (f,g) Recovered epipolar geometry. The lines in (g) are the epipolar lines that correspond to the crosses in (f). (h) Generic torso deformed to match the stereo data.

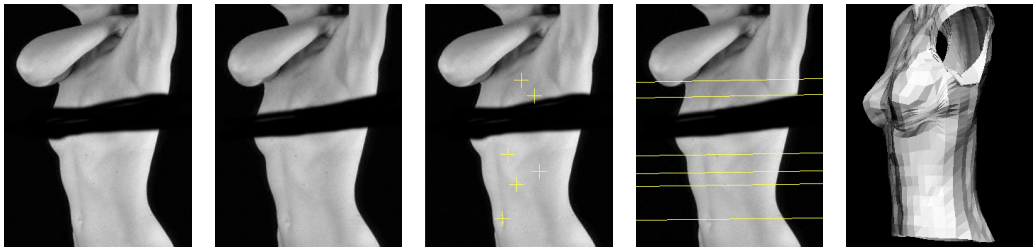


Figure 14: Body Modeling. (a,b) Another stereo pair found on the world wide web. (c,d) Recovered epipolar geometry. (e) Generic torso deformed to match the stereo data.

As discussed in Section 3 and 4, we model the overall deformation due to these error sources as an affine transform. Here, we use synthetic data and Monte Carlo simulations to show that:

1. For typical camera configurations, the deformation produced by using approximate camera parameters is indeed very close to being an affine transform even when there is a substantial amount of rotation between the cameras. This deformation is not severe as long as the approximate focal length is "reasonably" close to the actual one. Below, we will argue that for typical camera configurations, focal length estimates that are between 80 % and 120 % of the truth are amply sufficient.

2. The poor quality of the matches introduces another deformation that can also be modeled as an affine

transform. It does not substantially deform the shape if a sufficient regularization constraint is introduced.

This justifies our use of a fixed focal length and of an affine transform to model the overall deformation of our reconstructed models with respect to reality. Because the two error sources described above produce deformations of approximately the same magnitude, there is little point in using a more sophisticated approach to camera model recovery when much more accurate point matches cannot be obtained. The focal length cannot be reliably recovered from such data. Under our fixed focal length assumption, a small change in focal length during image acquisition would be modeled as a translation of the camera away or towards the subject without much loss of accuracy.

5.1 Influence of the Internal Parameters

To gauge the effect of using approximate internal parameters as opposed to exact ones, we used a random number generator to produce sets of 3–D points, such as the one shown in Figure 15(a). For each such set M, we

1. Define a camera model C_0 with focal length f, located at distance z from M;
2. Define two additional camera models, C_1 and C_2 that correspond to the same camera, and focal length, after randomly chosen rotation and translation;
3. Generate the projections of the points in M using C_0, C_1 and C_2 to produce the 2–D points m_0, m_1, and m_2;
4. Run the bundle-adjustment algorithm of Section 3 using m_0, m_1, and m_2 and an approximate focal length f' as input. This results in the reconstruction of M', a set of 3–D points, with $M' = M$ when $f' = f$;
5. Compute the affine transform A that best maps M' onto M in the least squares sense and let $A.M'$ be the points in M' transformed by A.

More specifically, in our experiments, M is formed by randomly distributed 3–D points in a $20 \times 20 \times 10cm$ volume, which approximates a face's dimensions. C_0, the first camera, is located at a distance z of either $50cm$ or $100cm$ from M. The motion of C_1 and C_2 with respect to C_0 is defined by random translation and rotation vectors that are scaled so that $||t||$, the translation vector's norm, is either $5cm$ or $10cm$ and $||r||$, the rotation vector's norm, is either 0.1, 0.2 or 0.3 radians.

All three cameras have the same focal length $f = 1mm$ and their principal points are taken to be the origin of the image planes' coordinate systems. Note that the chosen value of f is arbitrary. Changing it only produces an overall scaling of the projection values. What is significant is f'/f, the ratio of the focal length used to run the bundle adjustment algorithm to the real one.

We then ran our bundle-adjustment algorithm, using values of f' ranging from $0.5mm$ to $2.0mm$ and shifting the principal points by values of up to $0.03mm$. For each set of points M, corresponding reconstruction M' and affine transform A, as in Section 3.3, we evaluate

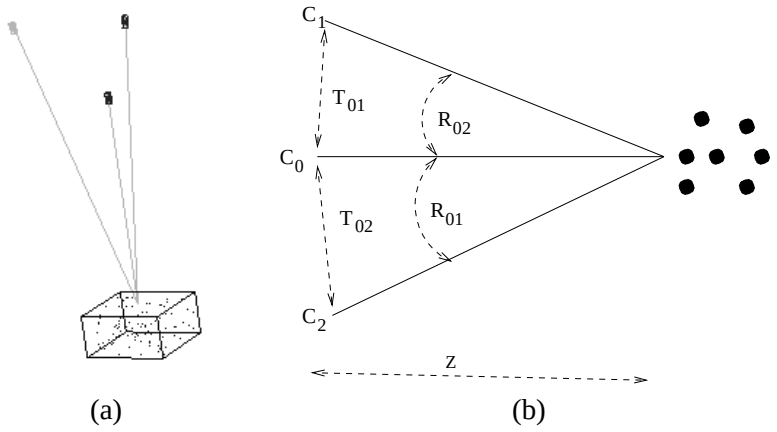


Figure 15: Synthetic data used to quantify the influence of using approximate internal parameters. (a) 3–D points uniformly distributed in a bounding box and three synthetic cameras. (b) The camera configuration is defined by Z, the distance of the first camera to the 3–D points, and by the rotation and translation vectors R and T that define the motion of the other two cameras with respect to the first.

AFF Validity of the affine approximation: The mean square distance of $A.M'$, the reconstructed points after affine transformation, to M. In the graphs below it is shown in a range from 0 to $10mm$. The closer to zero **AFF** is, the better the affine approximation.

DEF The severity of the deformation: We use the measure proposed in Section 3.3. We perform a singular value decomposition of A and compute the ratio of the largest to the smallest eigenvalue of the resulting diagonal matrix.

PRJ The reprojection error: The mean square distance of the projections of the reconstructed points to the actual values. In the graphs below, it is expressed in millimeters in a range from 0 to $0.002mm$. For cameras such as the ones we have used to acquire the images shown in Section 3, to convert the projection values expressed in mm to pixel values, one ought to multiply these values by a factor in the order of 4000 to 5000. In other words, this range corresponds to reprojection errors between 0 and approximately 8 to 10 pixels.

In Figure 16, we show the values of **AFF**, **DEF** and **PRJ** as a function of the ratio f'/f. Each row corresponds to a particular camera setup with $z \in \{50cm, 100cm\}$ and $||t|| \in \{5cm, 10cm\}$. If there were no rotation between the cameras, that is if $||r|| = 0$, the affine approximation would be perfect [Luong and Viéville, 1996], up to the numerical accuracy of our least-squares solver. Bearing this in mind, in each case, we plot three separate curves, for $||r|| = 0.1$, $||r|| = 0.2$ and $||r|| = 0.3$.

For each configuration, we performed one hundred trials. The curves depict the mean values of **AFF**, **DEF** and **PRJ** and the error bars represent variances. The further away the cameras are, that is the greater z, the lesser the perspective distortion and the better the affine approximation as measured by **AFF** is. This is not surprising since a large z corresponds to a situation where the camera can almost be considered as an orthographic one and, therefore, where the affine model is strictly correct. Similarly the smaller the rotation component $||r||$, the better the approximation. Note however that even for the "worst case scenario" plotted here—that is, $||r|| = 0.3rad$, $||z|| = 50cm$, and $||t|| = 5cm$—**AFF** remains below $1mm$ for $0.9 < f'/f < 1.1$ and below $2mm$ for $0.8 < f'/f < 1.2$. In other words, for typical face sequences with a camera that does not come too close to the head, the affine approximation is an excellent one.

The **PRJ** curves that depict the reprojection errors have the same overall shape as the **AFF** curves. For our worst case scenario, the reprojection errors remain below $0.00025mm$ for $0.8 < f'/f < 1.2$. For real cameras such as the ones used in this paper, this corresponds to errors smaller than 1 pixel. In other words, as long as the bundle adjustment algorithm produces residuals that are smaller than 1 pixel on average, the affine approximation can be considered as valid. This naturally leads to a very simple heuristic to guess a usable value of f' if none is available: Perform the bundle adjustment computation for several values of f' until one is found that yields sufficiently small residuals.

The previous results were computed without shifting the cameras' principal points. In Figure 17, we show three sets of **AFF**, **DEF** and **PRJ** curves computed in our worst case scenario and for

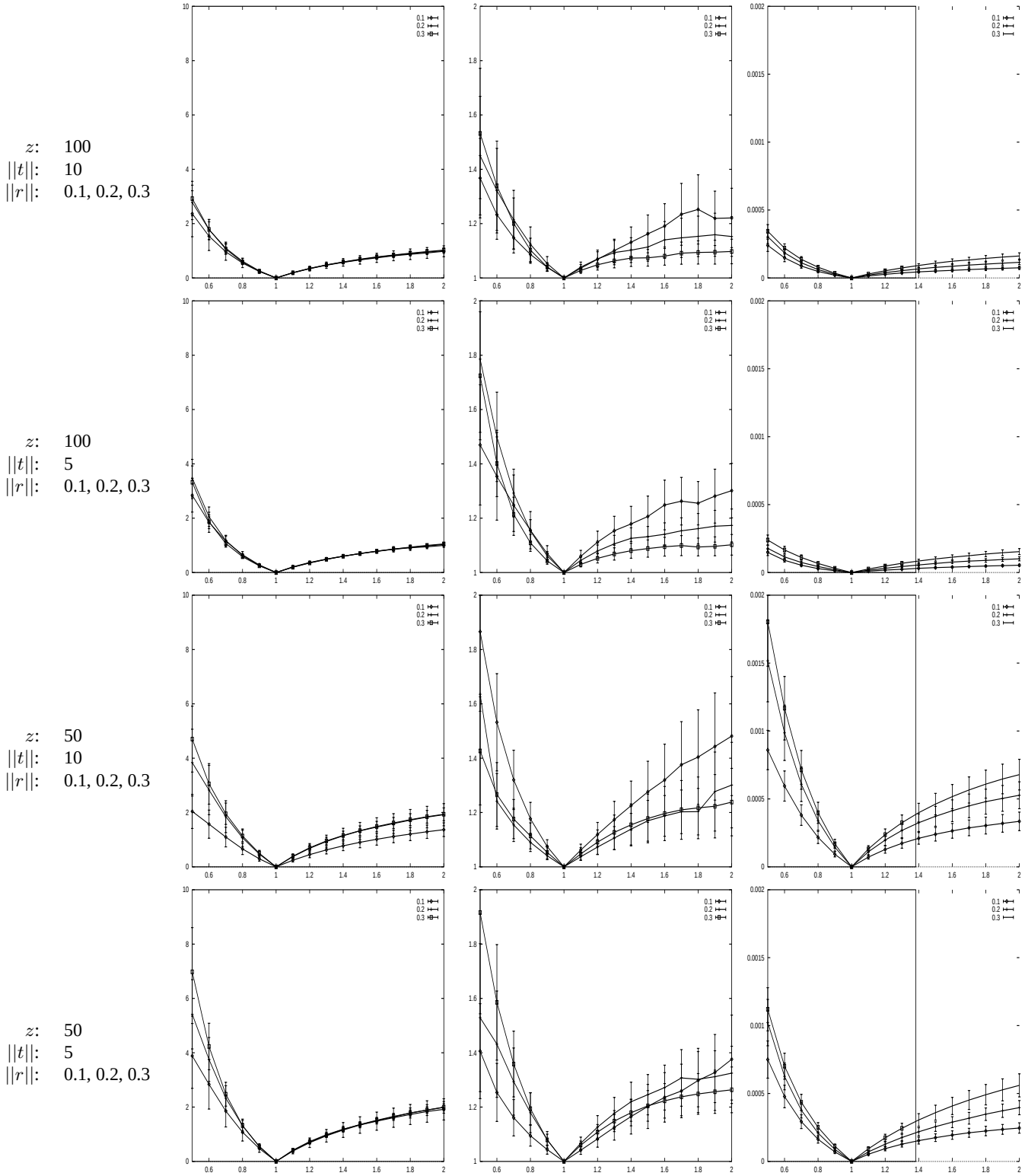


Figure 16: Using an approximate focal length for different camera configurations. **AFF** appears in the first column as a function of the ratio of the focal length used to perform the bundle adjustment to the real one. Similarly **DEF** and **PRJ** are shown in the second and third columns. See Section 5.1 for details.

shifts of the principal points of 0, $0.01mm$, $0.02mm$, and $0.03mm$. For real cameras such as the ones used in this paper, this corresponds to shifts of approximately 0, 50, 100 and 150 pixels. This shift has very little influence and all four curves are almost superposed in all three graphs. Therefore, in practice, we can safely assume that the principal point is in the center of the image.

z: 50
$||t||$: 5
$||r||$: 0.1, 0.2, 0.3

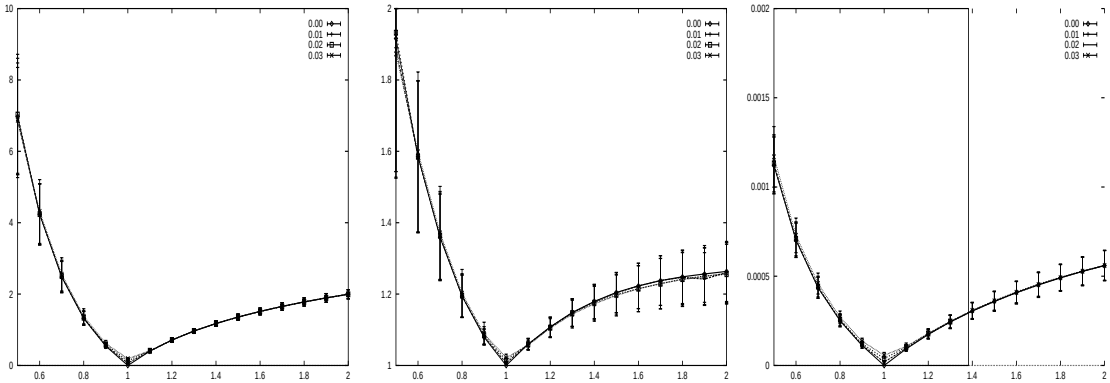


Figure 17: Using an approximate focal length and shifting the principal points. **AFF**, **DEF** and **PRJ** are depicted as in Figure 16 for z: $50cm$ $||t||$: $5cm$ $||r||$: $0.3rad$ and increasing shifts of the assumed location of the principal point.

5.2 Influence of Mismatches

The point matches we use as input to our bundle-adjustment cannot be expected to be either precise or error-free. To quantify the effectiveness of our regularized bundle-adjustment algorithm, we use the synthetic *surface triangulation* of Figure 18(a) and the corresponding pyramid shaped *bundle-adjustment triangulation* of Figure 18(b). The flat part of the surface triangulation is the plane of equation $z = 0$ and the spherical part has a radius of $35mm$. Again we use three synthetic cameras located at a distance of about $60cm$ with motions with respect to one another that include a rotational component of approximately $0.3rad$. In other words, this configuration approximates the "worst case scenario" of Section 5.1.

Given these camera models, we use as input to our algorithm the projections of points that have the same x, y coordinates as the bundle-adjustment triangulation's vertices but belong to the surface triangulation. To simulate the errors that can be expected from our stereo matcher, we corrupt these projections by adding two kinds of noise:

1. White noise with variance $\sigma_{noise} \in \{0.5pixel, 1.0pixel\}$. This simulates the imprecision of the matches. Given the specific camera geometry used here, a disparity error of 1.0 pixel translates to an error of 3 to 6 mm in terms of reconstruction accuracy. This represents 10 to 20 % of the 35 mm radius of the sphere, and is therefore a significant amount of noise.

2. Outliers whose coordinates are random. These outliers replace some of the actual projections to simulate mismatches. In the plots below, we introduce a proportion $rate_{out} \in \{0\%, 10\%, 20\%, 30\%\}$ of such gross errors.

Given these randomized projections, we ran our algorithm for values of the λ regularization parameter ranging from 0.0 to 1.0. The algorithm's aim is to recover camera models that are good enough to perform the stereo reconstruction of Section 4. To gauge this, we have adopted the following procedure: For each

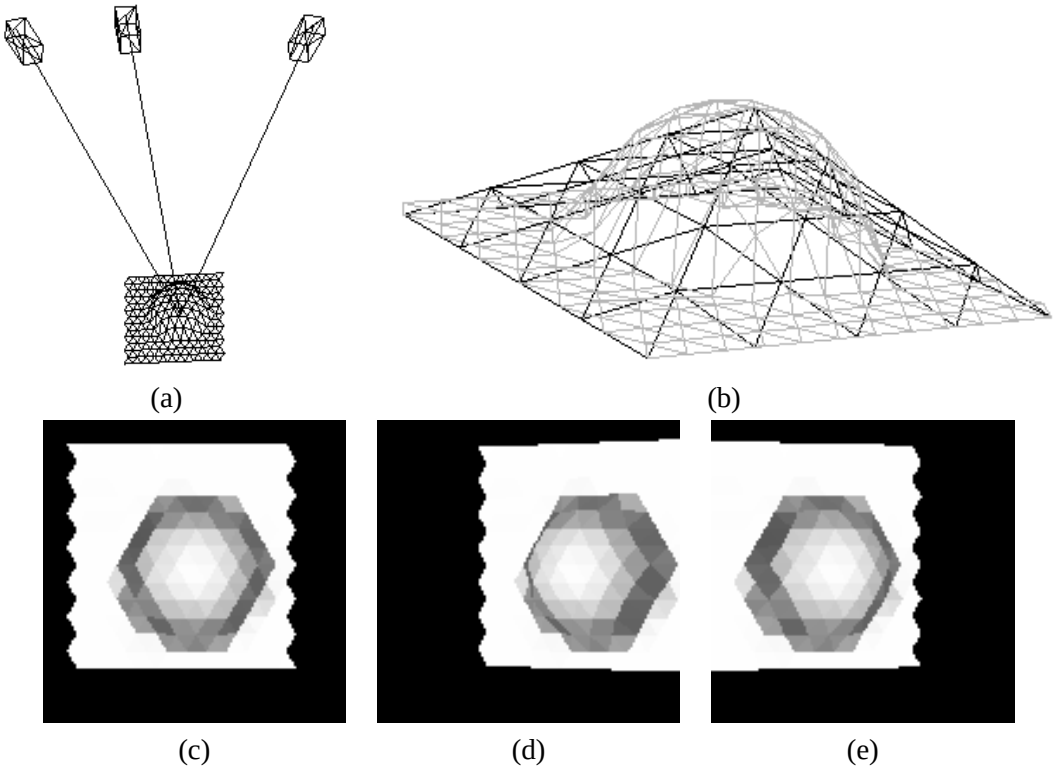


Figure 18: Synthetic data used to quantify the influence of mismatches and the robustness of our approach. (a) Half sphere used as the surface triangulation and synthetic cameras. (b) Corresponding pyramid shaped bundle-adjustment triangulation.(c,d,e) Shaded representations of the surface triangulation's projections in the three cameras.

trial run, we use the real projections of the vertices of the surface triangulation and the recovered camera models to compute two sets of 3–D points, a first one M_{12} using camera 1 and 2, and the second one M_{13} using camera 1 and 3. We evaluate:

DST The average distance between corresponding points in M_{12} and M_{13}: If the camera models were perfect, both these sets of reconstructed points should have the same 3–D location as the original 3–D vertices and **DST** would be zero. In the graphs below it is shown in a range from 0 to $4mm$, which correspond to reprojection errors in the order of one pixel or less.

AFF The affine nature of the deformation: Our reconstruction procedure fits a surface to the stereo data. This fitting operation averages the data. To simulate this behavior and the bias introduced by our regularization term, we take M_{123} to be the midpoint of corresponding points in M_{12} and M_{13} and compute the affine transform A that best maps M_{123} onto the original 3–D vertices. In the graphs below, we plot the mean square distance between $A.M_{123}$, the points in M_{123} transformed by A, and those original vertices. Small values of **AFF** indicate that the bias introduced by our regularization is well modeled by an affine transform.

DEF The severity of the deformation: We use the same measure as before.

Figure 19 depicts the mean values of **DST**, **AFF** and **DEF** as a function of λ after a hundred trials for each value of σ_{noise} and $rate_{out}$. The variances appear as error bars.

σ_{noise}: 0.5
$rate_{out}$: 0.0 to 0.3

σ_{noise}: 1.0
$rate_{out}$: 0.0 to 0.3

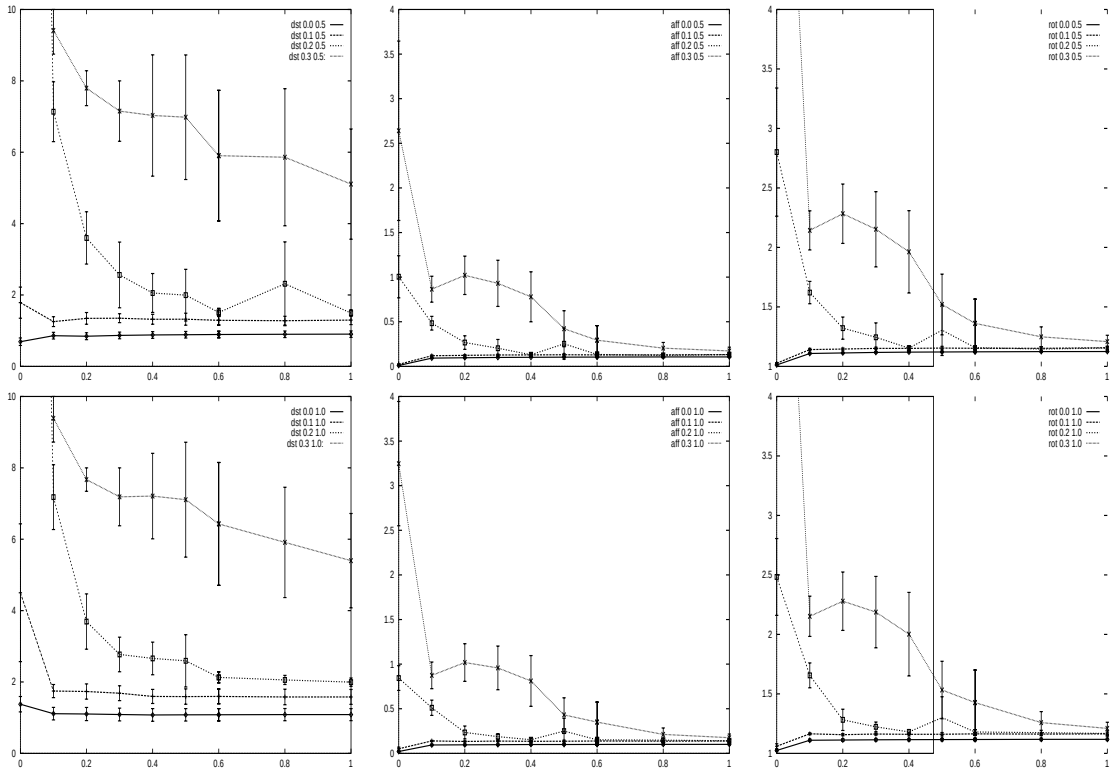


Figure 19: Influence of mismatches. **DST**, **AFF**, and **DEF** are represented in the first, second and third column respectively. Each row corresponds to one value of the noise. In all graphs, each curve corresponds to a different value of $rate_{out}$. See Section 5.2 for details.

In all graphs, we plot four curves, one for each value of $rate_{out}$. The lowest curve corresponds to $rate_{out} = 0$. In this specific case, the reconstruction errors are caused exclusively by the white noise added to the projections. The resulting deformation is well modeled by an affine transform that is almost a rotation as evidenced by the fact that **DEF** is close to 1.0. **DST** is approximately $1.0mm$, which translates to reprojection errors in the order of 0.2 pixels. Note that, in this specific case, **DEF** increases slightly when λ is non zero, indicating the bias introduced by the regularization. In all other cases, that is for $0 \leq rate_{out} \leq 20\%$, the exact opposite happens: **DST**, **AFF** and **DEF** decrease towards values that are not significantly larger than the ones computed for $rate_{out} = 0$. For $rate_{out} = 30\%$, **DST** starts to increase significantly but the averaging operation we perform results in values of **AFF** and **DEF** that remain low, at least when λ is large enough. In other words, a small amount of regularization, that is $\lambda = 0.2$ deals effectively with outliers rate of up to 20%. The use of a larger regularization term mitigates the impact of even higher rates, at the possible cost of a small amount of affine deformation in the resulting shape.

Obviously, the exact shape of these curves is heavily influenced by the closeness of the initial bundle-adjustment triangulation's shape to the actual one. Our experiments with numerous real sequences of heads shows that the bundle adjustment triangulation of Figure 3(c,d) is close enough, after stretching in all three directions to accommodate variations in proportions, to yield excellent results. The values of **DEF** that appear in table 1 are consistent with those reported in this section.

6 Conclusion

We have shown that by incorporating model-based constraints in the framework of bundle-adjustment, we are able to effectively tackle the structure-from-motion problem in a case where correspondences are difficult to establish.

As a result, we have been able to develop an integrated and largely automated approach to fitting a complete head model to images without requiring calibration data and with very limited manual intervention. Using this technique, such models can be produced cheaply and fast using an entirely passive sensor, even though the images we use may have relatively little texture. This has direct applications in the field of video communication and entertainment and will allow the fast generation of realistic avatars from widely available data.

Using, on the one hand, synthetic data and, on the other hand, images and laser scans of the same people, we have shown empirically that the models we create are very good approximations up to an affine transform. We chose to demonstrate the complete approach for one specific application for which we can perform the complete modeling task from images to usable models. However, because the constraints we impose only depend on the existence of a generic shape model for the objects to be reconstructed, this approach can potentially be generalized to many other deformable shapes. Specifically, we are interested in modeling the complete body. In future work, we will use articulated models to extend our approach to this new application field.

Appendix

In this appendix, we summarize briefly the technique we have used to fit the animation mask to the image data [Fua and Miccio, 1998, Fua and Miccio, 1999]. Given a set of *registered* images, it is designed to fit a facial animation model with minimal manual intervention. In the examples presented in this paper, the video sequences were initially uncalibrated and registration has been achieved using the technique of Section 3. However, calibrated stereo pairs or triplets can be used as well. The animation model [Kalra *et al.*, 1992] we use can produce the different facial expressions arising from speech and emotions. Our fitting procedure takes the following steps:

- **Obtain 3–D information:** We compute disparity maps for each image pair, fit local surface patches to the corresponding 3–D points, and use these patches to compute a central 3–D point and a normal vector.
- **Model the Face:** We attach a coarse control mesh such as the one shown in Figure 3(e) to the face of the animation model and perform a least squares adjustment of this control mesh, so that the model matches the previously computed data. We weight the data points according to how close—in the least squares sense—they are to the model and use an iterative reweighting technique to eliminate the outliers. We then subdivide the control mesh and repeat the procedure to refine the result.
- **Generate a texture map:** We use the original images to compute a cylindrical texture map that allows realistic rendering. This is achieved by first generating a cylindrical projection of the head model and then, for each projected point, finding the images in which it is visible and averaging the corresponding gray-levels.

The only manual intervention that is mandatory is supplying the location of the five key feature points of Figure 5(a) in one single image. In order to ensure proper animation, it is important to guarantee that

important features of the model—mouth and corners of the eyes especially—project at the correct locations in the face. The system is therefore set up so that we have the option to supply a few additional 2–D feature points such as the ones shown in the first column of Figure 9. Because only the 2-D location of these points need to be specified, this can be done very quickly.

References

[Baratoff and Aloimonos, 1998] G. Baratoff and Y. Aloimonos. Changes in Surface Convexity and Topology Caused by Distorsions of Stereoscopic Visual Space. In *European Conference on Computer Vision*, pages 188–202, Freiburg, Germany, June 1998.

[Beardsley *et al.*, 1997] P. A. Beardsley, A. Zissermam, and D. W. Murray. Sequential update of projective and affine structure from motion. *International Journal of Computer Vision*, 23(3):235–259, 1997.

[Beaton and Turkey, 1974] A. E. Beaton and J.W. Turkey. The Fitting of Power Series, Meaning Polynomials, Illustrated on Band-Spectroscopic Data. *Technometrics*, 16:147–185, 1974.

[Blanz and Vetter, 1999] V. Blanz and T. Vetter. A Morphable Model for The Synthesis of 3–D Faces. In *Computer Graphics, SIGGRAPH Proceedings*, Los Angeles, CA, August 1999.

[Boulic *et al.*, 1995] R. Boulic, T. Capin, Z. Huang, L. Moccozet, T. Molet, P. Kalra, B. Lintermann, N. Magnenat-Thalmann, I. Pandzic, K. Saar, A. Schmitt, J. Shen, and D. Thalmann. Environment for Interactive Animation of Multiple Deformable Human Characters. In *Eurographics*, pages 337–348, Maastricht, Netherlands, August 1995.

[DeCarlo and Metaxas, 1998] D. DeCarlo and D. Metaxas. Deformable Model-Based Shape and Motion Analysis from Images using Motion Residual Error. In *International Conference on Computer Vision*, pages 113–119, Bombay, India, 1998.

[Devernay and Faugeras, 1994] F. Devernay and O. D. Faugeras. Computing Differential Properties of 3–D Shapes from Stereoscopic Images without 3–D Models. In *Conference on Computer Vision and Pattern Recognition*, pages 208–213, Seattle, WA, June 1994.

[Faugeras *et al.*, 1992] O.D. Faugeras, Luong Q.-T., and S.J. Maybank. Camera self-calibration: theory and experiments. In *European Conference on Computer Vision*, pages 321–334, Santa-Margerita, Italy, 1992.

[Fitzgibbon and Zisserman, 1998] A.W. Fitzgibbon and A. Zisserman. Automatic Camera Recovery for Closed or Open Image Sequences. In *European Conference on Computer Vision*, pages 311–326, Freiburg, Germany, June 1998.

[Fua and Leclerc, 1995] P. Fua and Y. G. Leclerc. Object-Centered Surface Reconstruction: Combining Multi-Image Stereo and Shading. *International Journal of Computer Vision*, 16:35–56, September 1995.

[Fua and Miccio, 1998] P. Fua and C. Miccio. From Regular Images to Animated Heads: A Least Squares Approach. In *European Conference on Computer Vision*, pages 188–202, Freiburg, Germany, June 1998.

[Fua and Miccio, 1999] P. Fua and C. Miccio. Animated Heads from Ordinary Images: A Least Squares Approach. *Computer Vision and Image Understanding*, 75(3):247–259, September 1999.

[Fua, 1993] P. Fua. A Parallel Stereo Algorithm that Produces Dense Depth Maps and Preserves Image Features. *Machine Vision and Applications*, 6(1):35–49, Winter 1993.

[Gruen and Beyer, 1992] A. Gruen and H.A. Beyer. System Calibration through Self-Calibration. In *Calibration and Orientation of Cameras in Computer Vision*, Washington D.C., August 1992.

[Hartley *et al.*, 1992] R.I. Hartley, R. Gupta, and T. Chang. Stereo from Uncalibrated Cameras. In *Conference on Computer Vision and Pattern Recognition*, pages 761–764, 1992.

[Jebara and Pentland, 1997] T.S. Jebara and A. Pentland. Parametrized Structure from Motion for 3D Adaptive Feedback Tracking of Faces. In *Conference on Computer Vision and Pattern Recognition*, pages 144–150, Porto Rico, June 1997.

[Kalra *et al.*, 1992] P. Kalra, A. Mangili, N. Magnenat Thalmann, and D. Thalmann. Simulation of Facial Muscle Actions Based on Rational Free Form Deformations. In *Eurographics*, 1992.

[Kang, 1997] S. B. Kang. A Structure from Motion Approach using Constrained Deformable Models and Apperance Prediction. Technical Report CRL 97/6, Digital, Cambridge Research Laboratory, October 1997.

[Lanitis *et al.*, 1995] A. Lanitis, C.J. Taylor, and T.F. Cootes. A Unified Approach to Coding and Interpreting Face Images. In *International Conference on Computer Vision*, pages 975–980, Cambridge, MA, June 1995.

[Leclerc and Bobick, 1991] Y. G. Leclerc and A. F. Bobick. The Direct Computation of Height from Shading. In *Conference on Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii, June 1991.

[Lee and Thalmann, 1998] W.S. Lee and N. Magnenat Thalmann. From Real Faces To Virtual Faces: Problems and Solutions. In *3IA*, Limoges, France, 1998.

[Lee *et al.*, 1995] Y. Lee, D. Terzopoulos, and K. Waters. Realistic Modeling for Facial Animation. In *Computer Graphics, SIGGRAPH Proceedings*, pages 191–198, Los Angeles, CA, August 1995.

[Luong and Viéville, 1996] Q.-T. Luong and T. Viéville. Canonical Representations for the Geometries of Multiple Projective Views. *Computer Vision and Image Understanding*, 64(2):193–229, 1996.

[Pighin *et al.*, 1998] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D.H. Salesin. Synthesizing Realistic Facial Expressions from Photographs. In *Computer Graphics, SIGGRAPH Proceedings*, volume 26, pages 75–84, July 1998.

[Pollefeys *et al.*, 1998] M. Pollefeys, R. Koch, and L. VanGool. Self-Calibration and Metric Reconstruction In Spite of Varying and Unknown Internal Camera Parameters. In *International Conference on Computer Vision*, 1998.

[Press *et al.*, 1986] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes, the Art of Scientific Computing*. Cambridge U. Press, Cambridge, MA, 1986.

[Proesmans *et al.*, 1996] M. Proesmans, L. Van Gool, and A. Oosterlinck. Active acquisition of 3D shape for Moving Objects. In *International Conference on Image Processing*, Lausanne, Switzerland, September 1996.

[Samaras and Metaxas, 1998] D. Samaras and D. Metaxas. Incorporating Illumination Constraints in Deformable Models. In *Conference on Computer Vision and Pattern Recognition*, pages 322–329, Santa Barbara, June 1998.

[Sturm, 1997] P. Sturm. Critical Motion Sequences for Monocular Self-Calibration and Uncalibrated Euclidean Reconstruction. In *Conference on Computer Vision and Pattern Recognition*, pages 1100–1105, Puerto Rico, June 1997.

[Tang and Huang, 1996] L. Tang and T.S. Huang. Analysis-based facial expression synthesis. *ICIP-III*, 94:98–102, 1996.

[Tarel and Vezien, 1996] J.P. Tarel and J.M. Vezien. Camcal Manual: A Complete Software Solution for Camera Calibration. Technical Report 0196, INRIA, September 1996.

[Thalmann *et al.*, 1996] D. Thalmann, J. Shen, and E. Chauvineau. Fast Realistic Human Body Deformations for Animation and VR Applications. In *Computer Graphics International*, Pohang, Korea, June 1996.

[Triggs, 1997] B. Triggs. Autocalibration and the Absolute Quadric. In *Conference on Computer Vision and Pattern Recognition*, pages 609–614, 1997.

[Zhang *et al.*, 1995] Z. Zhang, R. Deriche, O. Faugeras, and Q. Luong. A Robust Technique for Matching two Uncalibrated Images through the Recovery of the Unknown Epipolar Geometry. *Artificial Intelligence*, 78:87–119, 1995.

[Zienkiewicz, 1989] O. C. Zienkiewicz. *The Finite Element Method*. McGraw-Hill, 1989.

[Zisserman *et al.*, 1998] A. Zisserman, D. Liebowitz, and M. Armstrong. Resolving ambiguities in auto-calibration. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences (A)*, 356(1740):1193 – 1211, May 1998.