# 3D Face Modeling from Stereo and Differential Constraints

Richard Lengagne[1]

Pascal Fua[2], Olivier Monga[1]

1: INRIA Rocquencourt
Domaine de Voluceau BP105
78153 Le Chesnay, FRANCE
Richard.Lengagne@inria.fr

2: Computer Graphics Lab
EPFL
CH-1015 Lausanne, SWITZERLAND
fua@lig.di.epfl.ch

## Abstract

*This paper proposes a way to incorporate a priori information in a 3D stereo reconstruction process from a pair of calibrated face images. In our framework, a 3D mesh modeling the surface is iteratively deformed in order to minimize an energy function in a snake-like process. Differential information about the object shape is used to generate an anisotropic mesh that can both fulfill the compacity and the accuracy requirements. Moreover, in areas where the stereo information is not reliable enough to accurately recover the surface shape, because of inappropriate texture or bad lighting conditions, we propose to incorporate some geometric constraints related to the differential properties of the surface. These constraints can be intuitive or can refer to some predefined geometric properties of the object to be reconstructed. They can be applied to scalar fields, such as curvature values, or structural features, such as crest lines, governing their location, number, or spatial organization. We demonstrate our approach using faces.*

## Keywords

*Stereo Reconstruction, Deformable Models, Differential Geometry, Adaptive Meshes, Constrained Optimization*

## 1 Introduction

3D face modeling is currently receiving a lot of attention among the Computer Vision and Computer Graphics communities and is a thriving research field that can yield to various applications such as virtual reality, animation, face recognition, etc... In all these applications, the reconstructed face needs to be compact and accurate, especially around significant areas like the nose, the mouth, the orbits, etc... These areas can often be characterized in terms of differential properties of the surface, and a great effort has to be done in order to accurately reconstruct those features. Several attempts to deal with that problem have been made. In [2], the differential properties of the surface are inferred from a disparity map and used to modify the shape of a correlation window. In [11], crest line extraction is performed on a 3D model and used to improve the reconstruction around sharp ridges. These methods improve the accuracy of the reconstruction but do not suffice if the initial depth map is not reliable. For instance, it is well known that bad lighting conditions or lack of texture can make correlation-based stereo fail. Consequently, the image information alone is sometimes not sufficient to recover the shape. In [5], constraints on the depth of a given set of points on a surface mesh are applied in order to improve terrain reconstruction. In [9], curvature information and structural features such as crest lines are extracted from the 3D model or interactively specified in order to generate an anisotropic surface mesh that reflects the geometric properties of the object. In this paper, we propose a further step towards incorporation of a priori information in the reconstruction process from a pair of calibrated face images. Differential information is used to constrain the topology of a mesh modeling the surface and the parameters of an analytical surface model, through the specification of low(high)-curvature areas, or structural features. Mathematically, this incorporation is achieved via constrained mesh optimization. We show preliminary results of this ongoing work, which aims at building 3D face models from Computer Vision techniques, that are as compact and accurate as possible and are consistent with a priori constraints about the face geometry.

## 2 The reconstruction process

### 2.1 An energy minimization scheme

Our reconstruction process is based on the iterative deformation of a 3D triangular mesh (i.e. a collection of vertices, triangular faces and edges) modeling the face in order to minimize an energy function $E$. The reconstruction process is thus treated as a snake-like process ([7],[6],[11]).

The initial mesh is given by triangulating the depth map provided by a standard correlation algorithm [4]. The energy function is the weighted sum of two terms: one external term $E_{ext}$, whose minimization makes the model fit to the data, and one internal term $E_{int}$, whose minimization constrains the model to be smooth enough.

The external term $E_{ext}$ is derived from the stereo information. We assume that the projections of a given 3D point in the 2 image planes have the same intensities. The purpose is thus to minimize the intensity difference between the two projections (see [6] or [11] for more details). Notice that this process will behave the same way as the correlation algorithm, i.e. it will fail in the same cases: lack of texture, lighting problems,...

The internal term $E_{int}$ is a regularization term which tends to minimize the deviation of the mesh from a plane. Its purpose is to minimize the global curvature of the surface. It is a quadratic term (which helps the convergence of the optimization process) and a function of the second order derivatives of the surface. It restricts the set of all possible solutions to the most “regular” ones.

Consequently, we minimize $E = \lambda_{ext}E_{ext} + \lambda_{int}E_{int}$. We thus have to find a trade-off between data-fitting (through the stereo term) and the smoothness of the solution (through the regularization term).

To perform this optimization, we have implemented a finite-element scheme. Consequently, for each facet of the mesh, we have an analytical expression of the surface. The depth $Z$ of each surface point is expressed as a polynomial function of the two other coordinates $X$ and $Y$. This polynomial is of degree 5, which guarantees that the surface is piecewise $C^1$ (see [10], [12]). The parameters of the optimization process are the depths of each vertex, as well as the partial derivatives of the depth with respect to $X$ and $Y$. Consequently, if the mesh is composed of $n$ vertices, we come up with a $6n$-variable state vector:

$$(Z_1, ..., Z_n), (\frac{\partial Z_1}{\partial X}, ..., \frac{\partial Z_n}{\partial X}),$$
$$(\frac{\partial Z_1}{\partial Y}, ..., \frac{\partial Z_n}{\partial Y}), (\frac{\partial^2 Z_1}{\partial X^2}, ..., \frac{\partial^2 Z_n}{\partial X^2}),$$
$$(\frac{\partial^2 Z_1}{\partial X \partial Y}, ..., \frac{\partial^2 Z_n}{\partial X \partial Y}), (\frac{\partial^2 Z_1}{\partial Y^2}, ..., \frac{\partial^2 Z_n}{\partial Y^2}).$$

We take as initial values of the $Z_i$ the depth values given by the triangulation of the depth map. To compute the initial values of the partial derivatives, we have locally approximated the surface by a quadric and computed the partial derivatives of the surface as being the partial derivatives of the corresponding quadric. These values define the polynomial approximation of the surface. Reciprocally, we can compute the partial derivatives and differential properties of the surface from the analytical expression of the surface.

## 2.2 Adaptive meshes

The computation time can be very high if we keep a very large number of vertices. Moreover, if we further want to use our 3D model for animation purposes, for instance, a large number of points can very soon become untractable. Therefore, we have to reduce the number of vertices and to keep the points in the most significant areas of the face. Furthermore, this has to be achieved with as much automation as possible. For instance, we would like to keep many points in the nose area, the orbits, the mouth, i.e. areas which are likely to act as landmarks in an animation process. All these areas can be characterized by geometrical properties of the surface, especially differential properties. Indeed, areas like the nose ridge, the orbits, can be expressed in terms of high curvature areas, or crest lines, whereas the cheeks, the forehead (where we would like a small number of facets) can be described as low curvature areas.

We have thus chosen to refine the 3D model according to the differential properties of the surface that can be easily inferred from the analytical expression of the surface or estimated by a local quadric approximation. The surface described by the finite element model is $C^\infty$ inside each facet, and $C^1$ between two facets. Besides, the second order partial derivatives are uniquely defined at each vertex, since they belong to the parameters of the optimization problem. Consequently, it is meaningful to compute the surface curvatures at each vertex. This computation is straightforward: we can easily compute the first and the second fundamental forms associated to the surface, respectively denoted by their matrices $M_1$ and $M_2$, and the Weingarten endomorphism $W = -M_1^{-1}M_2$ (see [3]). The principal curvatures and the principal curvature directions are respectively the eigenvalues and the eigenvectors of $W$. We briefly review the computation of the principal curvatures:

If $k_{max}$, $k_{min}$, $K$, and $H$ respectively denote the maximum curvature, the minimum curvature, the gaussian curvature $(= k_{max}k_{min})$ and the mean curvature $(= \frac{1}{2}(k_{max}+k_{min}))$ of the surface at a given vertex, and if $Z_x$ denotes the partial derivative of $Z$ with respect to $X$ at this vertex, we can write:

$$K = \frac{Z_{xx}Z_{yy} - Z_{xy}^2}{(1 + Z_x^2 + Z_y^2)^2}$$
$$H = \frac{1}{2}\frac{(1 + Z_x^2)Z_{yy} - 2Z_xZ_yZ_{xy} + (1 + Z_y^2)Z_{xx}}{(1 + Z_x^2 + Z_y^2)^{\frac{3}{2}}}$$
$$k1 = H + \sqrt{H^2 - K}$$

$k2 = H - \sqrt{H^2 - K}$

If $\mid k_1 \mid \leq \mid k_2 \mid$, $k_{min} = k_1$ and $k_{max} = k_2$.

Otherwise, $k_{min} = k_2$ and $k_{max} = k_1$.

As described in [9], we generate an adaptive mesh governed by the principal curvatures and the principal curvature directions of the surface.

The algorithm can be summarized as follows:

- compute on the initial mesh the principal curvatures $k_{max}$ and $k_{min}$ and the principal curvature directions $\vec{t}_{max}$ and $\vec{t}_{min}$.
- specify for each vertex of the initial mesh the three parameters (two scalar values $h_1$ and $h_2$ and an angle $\theta$) of an ellipse centered on the vertex which governs the generation of a new mesh.
- optimize the new mesh by minimizing the energy function $E = \lambda_{ext} E_{ext} + \lambda_{int} E_{int}$.

The algorithm completely remeshes a 2D domain, which is taken here to be a frontal projection of the face. Therefore, the vertices will not be at the same locations anymore and the surface will be sampled according to the sets of $h_1$, $h_2$ and $\theta$. These values govern the local topology of the new mesh in the vicinity of the old vertex they are attached to. As shown in figure 1, the angle $\theta$ determines in which direction the new facet in the remeshed surface will be “elongated”. This direction will be given by $\vec{t}_{min}$. In other terms, the edges of the new facets will be longer in the minimum curvature direction than in the maximum curvature direction (those two directions are orthogonal). This is rather intuitive: for instance, in the case of the nose ridge, the minimum curvature direction lies along this ridge. We want to capture as many details as possible in the direction orthogonal to this ridge, since there is a high curvature change in that direction. Consequently, it is natural to generate longer edges in the minimum curvature direction (i.e.along the ridge) than in the maximum curvature direction (i.e. across the ridge). The scalar values $h_1$ and $h_2$ determine the average lengths of the edges in those two directions. They are decreasing functions of $k_{max}$ and $k_{min}$, since we want more facets in low curvature areas. Typically, they are chosen as inverses of a second order polynomial function. $h_1$ is determined by the minimum curvature and $h_2$ is determined by the maximum curvature. This procedure uses a mesh generation software developed for the Computational Field Simulations ([1]).

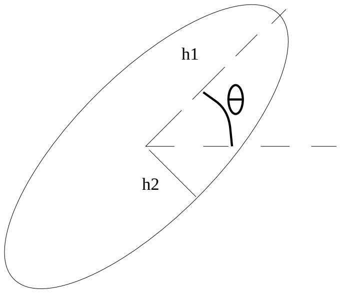

**Figure 1. the ellipse defining the local topology of the new mesh**

We show in figures 2 to 4 some experimental results derived using this method. Figure 2 shows a stereo pair of a face in a rectified position (here, the horizontal disparities are zero). The image sizes are 512x512.
Figure 3 shows the initial depth map obtained by a correlation algorithm, and the triangulation of the depth map in high resolution (4627 vertices). Such a number of vertices is much too large for any kind of subsequent applications. Our purpose is thus to selectively reduce the number of vertices while keeping a good reconstruction accuracy and, if possible, improve the reconstruction in significant areas.
Figure 4 shows an anisotropic mesh of the face and the result of its optimization. This new mesh has 248 vertices. Of course, it is hard to visually compare the initial mesh and this one in shaded views (which are planar approximations of the surface) since the number of points has been roughly divided by 18, but the main point is to keep a good reconstruction around typical features like the nose or the orbits.

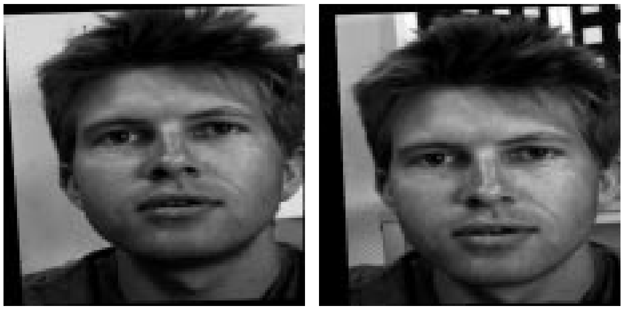

**Figure 2. a stereo pair of a face**

## 2.3 Limitations of the scheme

We have therefore generated a new mesh which is much more compact than the original one and which preserves

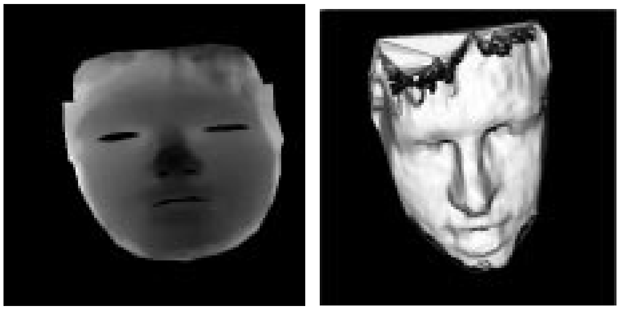

Figure 3. the depth map and the initial mesh (4627 vertices)

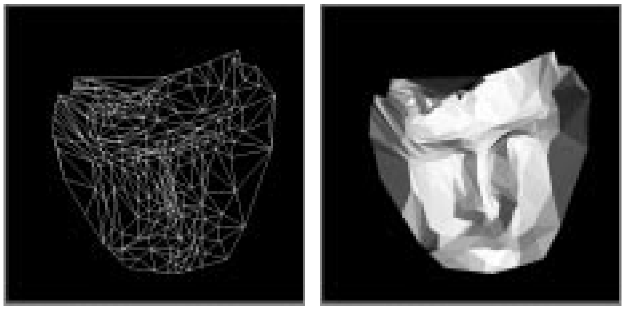

Figure 4. the anisotropic mesh and the result of its deformation (248 vertices)

the high curvature areas. However, this method can only be used if the initial 3D model is good enough to yield reliable curvature information. In many cases, the initial model is too far from the true surface to produce such information. For instance, in the above example, the 3D shape of the forehead cannot be recovered accurately from stereo information alone because of the presence of hair, which will make the correlation process fail in this area. This is analogous to the case of terrain reconstruction in presence of vegetation, which will make the recovery of the 3D shape impossible. In other cases, bad lighting conditions will produce the same undesirable effects. Therefore, it seems necessary to incorporate in the reconstruction process extra information that can help the recovery of the 3D shape. Mainly, this incorporation has two goals:

- compensate the reduction of the number of vertices in order to preserve a good reconstruction accuracy.
- compensate the insufficiency or the inadequateness of the information contained in the image to accurately reconstruct the 3D shape.

## 3 Incorporating a priori knowledge

### 3.1 A priori knowledge and differential properties

When we want to reconstruct an object, we have a rough idea about its shape, especially the location of typical features like crest lines, the spatial relation between these lines or the existence of patches that we can describe as "flat", "spherical", "cylindrical",etc... This kind of a priori knowledge can be of great interest where the classical stereo methods fail because of the reasons expressed above.

The a priori knowledge that a user can have about the shape he wants to reconstruct can be intuitive ("This region is flat, or spherical") or can rely on well-known geometric properties, which can come from anthropometry in case of face reconstruction, or geology, in case of terrain reconstruction, etc...

In any case, this a priori knowledge can very often be expressed in terms of differential properties. For instance, the knowledge "This area is flat" is obviously "translated" as:
at each vertex, $k_{max} = k_{min} = 0$.
"This area is spherical" means: at each vertex, $k_{max} = k_{min}$.
We can also express "structural" knowledge such as "There is a crest line here", and interactively outline the crest on the depth map (or, ideally, on the images) in differential terms: geometrically, a crest line is defined as a set of zero-crossings of the derivative of the maximum curvature in the maximum curvature direction, i.e. the set of points such that $dk = \nabla k_{max}.\vec{t}_{max} = 0$. The location of the crest on the depth map gives its location on the 3D mesh. The line goes through several facets and separates areas where $dk > 0$ from areas where $dk < 0$ (see figure 5). Notice that $\nabla k_{max}$ can be easily derived from the analytical expression of the surface. Imposing a constraint on the location of the crest line is thus equivalent to imposing on several vertices $dk < 0$ and on others $dk > 0$.

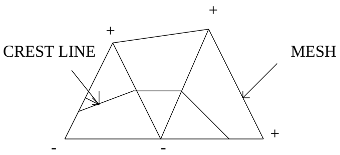

Figure 5. the crest line going through the facets nad the sign of $dk$

### 3.2 Constrained optimization

Incorporating a priori knowledge in the reconstruction process can be achieved using constrained optimization. All the constraints are expressed in terms of the partial derivatives which are the parameters of the optimization process. We use for that purpose a constrained optimization software especially designed for large systems [8] (which is our case, since we have 6 parameters per vertex).

### 3.3 Applications

So far, we have only tested our constrained on synthetic data (reconstruction of a sphere from a noisy initial state, using the constraint of equal curvatures, and reconstruction of a ridge with outlining the crest line). We have also reconstructed the forehead of the face shown in the previous section, using the a priori assumption that the part of the skull above the orbits is roughly spherical.
In the latest example, we first constrain the topology of the mesh to be rather uniform and isotropic ($h_1 = h_2$ and $\theta = 0$ with the notations of the previous section), since the curvatures are globally the same on this area. Notice that in the previous section, the program had generated many facets in some areas of the forehead, since the correlation algorithm providing the initial depth map had failed in reconstructing a smooth surface. We then minimize $E = \lambda_{ext} E_{ext} + \lambda_{int} E_{int}$ under the following constraints:

$$\forall i \in \{1,..,n\}, k_{max}(i) = k_{min}(i) \qquad (1)$$

$$\forall (i,j) \in \{1,..,n\}^2, k_{max}(i) = k_{max}(j) \qquad (2)$$

$$\forall (i,j) \in \{1,..,n\}^2, k_{min}(i) = k_{min}(j). \qquad (3)$$

where $i$ denotes the i-th vertex.

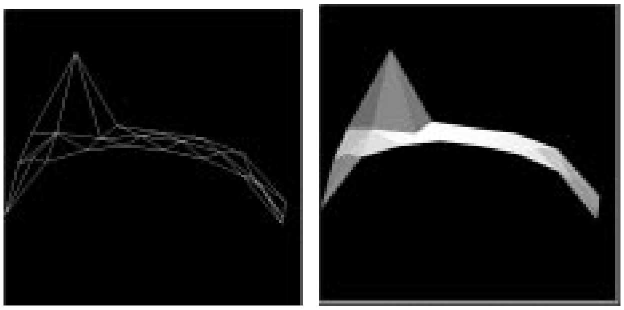

**Figure 6. the reconstructed forehead without differential constraints: the mesh and a shaded view**

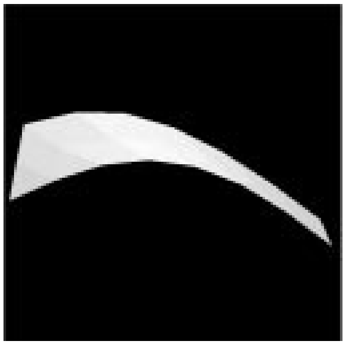

**Figure 7. the forehead: the final reconstruction after incorporating differential constraints**

In this example, we showed that we could achieve our three goals:

- produce a compact and accurate reconstruction.
- get rid of some problems induced by stereo methods.
- be consistent with the a priori knowledge about the object shape and about its differential properties.

Note that, if our goal had only been to flatten this area, we could have merely minimized $E = \lambda_{ext} E_{ext} + \lambda_{int} E_{int}$ with $\lambda_{int}$ set to a very large value, thus constraining the shape to converge to a plane. The sets of properties (1), (2) and (3) would have been fulfilled as well, but the weight of the image information would have been too small to make the model converge towards the real shape, i.e. a sphere. On the contrary, in our constrained optimization scheme, the image information is still present, that makes the global shape look like a sphere, and the differential constraints act locally to avoid some undesirable behaviors such as the one due to the presence of hair. These experiments are still preliminary, but our purpose is to build a general framework that could be applied to different cases when conventional stereo fails.

## 4 Conclusion

We have proposed a way of interactively reconstructing from stereo a complex 3D object like a face using a priori information about its shape and its differential properties. This kind of information can be of great interest when dealing with objects whose texture generally make conventional stereo algorithms fail, or captured in bad lighting conditions. Our long-term purpose is to develop an interactive image-based modeling software that takes into account some a priori knowledge that a user can have about the differential properties of the object to reconstruct.

## References

[1] H. Borouchaki, M. J. Castro-Diaz, P. L. George, F. Hecht, and B. Mohammadi. Anisotropic adaptive mesh generation in two dimensions for cfd. In *5th International Conference on Numerical Grid in Computational Field Simulations, Mississippi State University, USA*, April 1996.

[2] F. Devernay and O. Faugeras. Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. In *CVPR'94, Seattle,USA*, pages 208–213, 1994.

[3] M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, 1976.

[4] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision Applications, 6(1)*, 1993.

[5] P. Fua and C. Brechbuhler. Imposing hard constraints on soft snakes. In *European Conference on Computer Vision, Cambridge, U.K., II, p. 495-506*, 1996.

[6] P. Fua and Y. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal on Computer Vision*, 1995.

[7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes : Active contour models. *International Journal on Computer Vision, 1(4) : 321-331*, 1988.

[8] C. Lawrence, J. L. Zhou, and A. L. Tits. User's guide for cfsqp version 2.4: A c code for solving (large scale) constrained nonlinear (minimax) optimization problems, generating iterates satisfying all inequality constraints. *Electrical Engineering Department and Institute for Systems Research, University of Maryland, College Park, MD 20742*.

[9] R. Lengagne, O. Monga, and P. Fua. Using differential constraints to reconstruct complex surfaces from stereo. In *CVPR'97, San Juan, Puerto Rico, USA*, pages 1081–1086, 1997.

[10] W. Neuenschwander. *Elastic Deformable Contour and Surface Models for 2D and 3D Image segmentation*. PhD thesis, Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, 1995.

[11] R.Lengagne, J.-P. Tarel, and O. Monga. From 2d images to 3d face geometry. In *Proceedings of the 2nd International Conference on Automated Face and Gesture Recognition (FG'96), Killington, Vermont, U.S.A.*, 1996.

[12] O. Zienkiewicz and R. Taylor. *The finite element method*, volume Vol 1,2. Mc Graw-Hill, 1988.