

# Incorporating Differential Constraints in a 3D Reconstruction Process

## Application to Stereo

Richard Lengagne and Pascal Fua \*  
Computer Graphics Laboratory  
Swiss Federal Institute of Technology (EPFL)  
1015 Lausanne, Switzerland  
{Richard.Lengagne, Pascal.Fua}@epfl.ch

### Abstract

*We propose to incorporate a priori geometric constraints in a 3-D stereo reconstruction scheme to cope with the many cases where image information alone is not sufficient to accurately recover 3-D shape.*

*Our approach is based on the iterative deformation of a 3-D surface mesh to minimize an objective function. We show that combining anisotropic meshing with a non-quadratic approach to regularization enables us to obtain satisfactory reconstruction results using triangulations with few vertices.*

*Structural or numerical constraints can then be added locally to the reconstruction process through a constrained optimization scheme. They improve the reconstruction results and enforce their consistency with a priori knowledge about object shape. The strong description and modeling properties of differential features make them useful tools that can be efficiently used as constraints for 3-D reconstruction.*

## 1. Introduction

Many vision problems, such as image restoration, segmentation, optical flow recovery, shape-from-X and many other can be formulated as optimization problems. This typically involves defining an objective function that is the sum of a data consistency term and a regularization term. This second term is made necessary by the inherent noisiness and possible incompleteness of image data. It is often taken to be quadratic because it then improves convergence properties by convexifying the optimization problem. This improvement, however, comes at a price: a quadratic term pe-

nalizes large variations of the function to recover and tends to isotropically smooth the solution, no matter what the true variations of the function ought to be.

This problem has been especially addressed in the case of image restoration [13], stereo reconstruction [12], optical flow computation [3], and motion estimation [1]. These approaches, however, are image-based whereas, given the task of reconstructing a surface from multiple images whose vantage points may be very different, we need a surface representation that can be used to generate images of the surface from arbitrary viewpoints, taking into account self-occlusion, self-shadowing, and other viewpoint-dependent effects. Clearly, a single image-centered representation is inadequate for this purpose. Instead, an object-centered surface representation is required. Furthermore, geometric constraints are typically easier to express in the 3-D world which makes the use of an object-centered representation even more desirable.

We therefore build upon our earlier surface reconstruction work [8] in which we used regular hexagonal 3-D meshes to reconstruct surfaces by minimizing an energy term that included a stereo and shape-from-shading term. It is effective but requires the use of very dense meshes to yield accurate reconstruction results. As a result, it becomes impractical for large areas.

In the first half of this paper, we show that replacing regular meshes by anisotropic ones and the quadratic regularization term by a non-quadratic one allows us to better preserve true surface variations while using much more compact meshes.

In the second half of the paper, we address the case where the constraint implied by the use of regularization term is not sufficient to provide a good solution to the optimization problem. In practice, this is far from a rare occurrence, whatever the chosen shape-from-X technique is. For example, in the case of stereo, this can be due to occlusions, specularities, lack of texture, or shape discontinuities.

---

\*This work was supported in part by the Swiss Federal Office for Education and Science under contract 00.0530 and in part by an INRIA (France) research grant.

It then becomes necessary to include in the reconstruction process a priori shape constraints. In this case, differential geometry and its powerful description and modeling capabilities let us derive useful features. We show that they can be efficiently used to provide extra information when the data information is clearly corrupted or not reliable enough.

Our contribution is therefore an approach that allows us to impose generic constraints that can overcome inherent image data noisiness in cases where specific model information is not available.

In the remainder of the paper, we first quickly review our earlier technique based on the use of regular triangulations. We introduce our anisotropic meshes and non-quadratic approach to regularization. We then show how this framework can be used to enforce differential constraints.

## 2. Related Work

In earlier work [8], we developed an approach to 3-D surface reconstruction. It uses regular meshes to combine stereo and shape-from-shading information. The surface is modeled as an hexagonally connected set of vertices. The position of a vertex  $V_j$  is specified by its Cartesian coordinates  $(x_j, y_j, z_j)$ . The mesh can be deformed by varying these coordinates to minimize an objective function that includes terms derived from stereo and shading information. Its state vector  $S$  is the vector of all  $x, y$ , and  $z$  coordinates.

In our present work, we only consider stereo information. The stereo component  $E_{stereo}$  of the objective function is derived by comparing the gray levels of the points in all the images for which the projection of a given point on the surface is visible. This comparison is done for a uniform sampling of the surface. This method allows us to deal with arbitrarily slanted regions and to discount occluded areas of the surface. The stereo term for one sample point can be mathematically expressed using:

$$\mu(M) = \frac{\sum_{i=1}^N v_i(M) I_i(M)}{\sum_{i=1}^N v_i(M)}$$

$$\sigma^2(M) = \frac{\sum_{i=1}^N v_i(M) (I_i(M) - \mu(M))^2}{\sum_{i=1}^N v_i(M)}$$

where  $I_i(M)$  denotes the intensity of the projection of point  $M$  in image  $i$ , and  $v_i(M)$  is equal to 1 if  $M$  is visible in image  $i$  and to 0 otherwise. We take the global stereo component to be the sum of the variances of the gray levels of the projections of all sample points.

In general,  $E_{stereo}$  is a highly non-convex function of the vertex positions. To minimize  $E_{stereo}$ , we use a “snake-type” [9] optimization technique. We define the total energy of the mesh,  $E_T$ , as

$$E_T = E_{reg} + \lambda E_{stereo} \quad (1)$$

$$E_{reg} = 1/2(X^T K X + Y^T K Y + Z^T K Z)$$

dvips -where  $E_{reg}$  is a quadratic regularization term that approximates the sum of the square of the surface curvatures,  $K$  is a sparse and banded matrix,  $X, Y$ , and  $Z$  are the vectors of the  $x, y$ , and  $z$  coordinates of the vertices. This regularization term serves a dual purpose. First, as before, it “convexifies” the energy landscape when  $\lambda$  is relatively small and improves the convergence properties of the optimization procedure. Second, in the presence of noise, some amount of smoothing is required to prevent the mesh from overfitting the data, and wrinkling the surface excessively.

To speed the computation and prevent the mesh from becoming stuck in undesirable local minima, we typically use several levels of mesh sizes to perform the computation. We start with a relatively coarse mesh that we optimize. We then refine it by splitting every facet into four smaller ones and reoptimizing. Finally, we repeat the split and optimization processes one more time.

This overall framework will be also used in the remainder of this article, using stereo information, only replacing the quadratic regularization term by a non-quadratic one and replacing uniform facet splitting by adaptive mesh generation.

## 3. Anisotropy

Anisotropy is an essential component of our approach. We exploit it in two different ways.

- **Mesh topology:** Obtaining accurate reconstruction results using regular meshes often requires a huge amount of vertices, which can soon become both computationally untractable and unsuitable for subsequent tasks, such as animation of a face mesh. We therefore advocate designing meshes whose topology reflects the surface’s geometric properties.
- **Regularization:** Quadratic terms tend to uniformly, or isotropically, penalize sharp surface variations. This is not always a good thing because sharp variations do occur in real surfaces. We will see that non-quadratic and anisotropic regularization avoids this problem.

In this section, we introduce our mesh adaptation strategy and our implementation of a non-quadratic regularization algorithm. They let us use much more compact meshes than the ones required by our earlier approach based on regular triangulations, while retaining excellent reconstruction capabilities.

### 3.1. Anisotropic surface meshing

The purpose of anisotropic meshing is to design a representation of the surface that reflects its geometrical properties. For instance, we want to generate a mesh that contains

more facets in curved areas than in flat areas. The mesh obtained is therefore non-uniform. Furthermore, we want the edges of the new mesh to be shorter in the directions of large curvature than in the directions of small curvature. For instance, if we consider the nose area on a face surface, we want the facets to be more elongated along the nose ridge than across the ridge. Our mesh generation is therefore governed by the principal curvatures and directions computed on an initial mesh.

To estimate differential properties, we use filtering techniques such as the ones already developed for typical feature extraction in 2D or 3D images [11]. We first compute a local parametrization  $(x(u, v), y(u, v), z(u, v))$  of the surface in the neighborhood of each vertex using normal geodesic coordinates. We then compute the principal curvatures and the principal curvature directions at each mesh vertex [4]. The meshing algorithm we use [2] takes as input a first arbitrary bidimensional mesh of the surface. We attach to each vertex of this mesh a set of 3 features  $(a, b, \theta)$  that govern the remeshing of the surface in the neighborhood of this vertex. The angle  $\theta$  is set as the angle made by the  $x$ -axis and the minimum curvature direction, while  $a$  and  $b$  are decreasing functions of the principal curvatures (details can be found in [10]).

### 3.2. Non-quadratic regularization

A non-quadratic regularizer over a bidimensional domain  $\Omega$  can be formally expressed as  $E_{reg} = \int \Phi(|\nabla z|) dx dy$  where  $(x, y)$  is a parametrization of  $\Omega$  and  $z$  is the function we want to recover. It is shown in [12] that such a regularization can be decomposed in 2 terms respectively expressing the regularization amount in the direction of  $\nabla z$  and the direction orthogonal to  $\nabla z$ . This leads to specific conditions on the  $\Phi$  function and to different kinds of such functions. One of the most popular ones is the function  $\Phi(x) = x$  called Rudin function in this context. It regularizes isotropically in areas where no variations occur and anisotropically where variations occur, i.e. it performs a directional regularization, producing more smoothing along the directions of large variations than across these directions. On the contrary, the Tikhonov function  $\Phi(x) = x^2$  used in quadratic regularization tends to smooth the solution isotropically even if large variations occur. In our 3D reconstruction context, we are looking for a depth function  $z(x, y)$ , with  $(x, y) \in \Omega$ , minimizing the sum of the above-defined stereo energy and of the non-quadratic regularization term that we write as:

$$E_{reg} = \int_{\Omega} \sqrt{z_x^2 + z_y^2} dx dy \quad (2)$$

A good way to handle this problem is to solve the Euler-Lagrange equation associated to Equation 2. The Euler-Lagrange equation, which is a necessary condition to reach

a minimum of  $E = \int_{\Omega} F(z, z_x, z_y) dx dy$  can be written as:

$$F_z - \frac{\partial}{\partial x} F_{z_x} - \frac{\partial}{\partial y} F_{z_y} = 0 \quad (3)$$

Since this equation is non-linear, the function  $z$  is the solution  $z(x, y, t)$  of an evolution problem with boundary conditions expressed as  $\nabla z = 0$  on  $\partial\Omega$ , the border of the domain  $\Omega$ . Following the scheme presented in [3], this problem can be formally stated as:

$$\left(\frac{\partial z}{\partial t}, g\right) + a_n(z, g) = L(g), \forall g \in H^1(\Omega) \quad (4)$$

with  $z(x, y, 0) = z^0(x, y)$ .

$a_n$  denotes a symmetrical bilinear form defined as

$$a_n(z^n, g) = \int_{\Omega} \frac{1}{\sqrt{(z_x^{n-1})^2 + (z_y^{n-1})^2}} \nabla z^n \nabla g dx dy \quad (5)$$

$L$  is a linear form defined as

$$L(g) = \int_{\Omega} F_z g(x, y) dx dy \quad (6)$$

$H^1(\Omega)$  denotes the Sobolev space  $\{g \in \mathcal{L}^2(\Omega); \frac{\partial g}{\partial x} \in \mathcal{L}^2(\Omega), \frac{\partial g}{\partial y} \in \mathcal{L}^2(\Omega)\}$  and  $(f, g)$  denotes the scalar product associated to the  $\mathcal{L}^1$ -norm.

$z(x, y)$  is then iteratively computed through a time discretization scheme. We use a finite element scheme that approximates  $z(x, y)$  as a linear combination of polynomial shape functions [14].

As we will see in Section 4, we want further on to compute differential properties up to the second order on some mesh vertices. Therefore, the degrees of freedom of a polynomial function defined on each facet of the mesh must be the depth of each vertex of this facet and their 1st and 2nd order partial derivatives with respect to  $x$  and  $y$ . This leads to 18 degrees of freedom per polynomial. The first polynomial satisfying this condition is a polynomial of degree 5 that involves 21 coefficients. Boundary conditions on the normal to each edge of the facet give 3 more equations that completely define the polynomial and ensure a  $\mathcal{C}^1$  continuity between adjacent facets.

Using this framework, the variables of the optimization problem are the depths  $z^i$ , the first-order partial derivatives of the depths  $z_x^i$  and  $z_y^i$ , and the second-order derivatives  $z_{xx}^i, z_{xy}^i, z_{yy}^i$ , which makes 6 variables per mesh vertex.

On a mesh containing  $N$  vertices, the depth of a given point  $z(x, y)$  can thus be written as a linear combination of shape functions:

$$z(x, y) = \sum_{i=1}^{6N} S_i N_i(x, y) \quad (7)$$

The set of  $S_i$  variables will be later referred to as the state vector. In practice, the shape functions are non-zero on a very limited domain. At a given point  $(x, y)$  located inside a facet, only the 18 shape functions associated to the 3 vertices of the facet are non-zero.

The minimization problem can then be written as a linear system, successively choosing as the function  $g$  of the variational formulation each of the shape functions  $N_j$ :

$$\sum_{i=0}^N S_i^{(n)} \left( \int_{\Omega} N_i(x, y) N_j(x, y) dx dy + \tau a_n(N_i, N_j) \right) = \sum_{i=0}^N S_i^{(n-1)} \int_{\Omega} N_i(x, y) N_j(x, y) dx dy + \tau L(N_j), 1 \leq j \leq N$$

This system can be written  $AS^n = B$ , which enables us to iteratively estimate the state vector  $S$ . The sparse matrix  $A$  has to be evaluated at each iteration and the system is solved using a LU decomposition of the matrix  $A$ .

If the global energy has increased between two iterations, we go back to the previous value of the state vector and rerun the optimization step using a new value of  $\tau$  equal to half of the previous one.

### 3.3. Experimental results

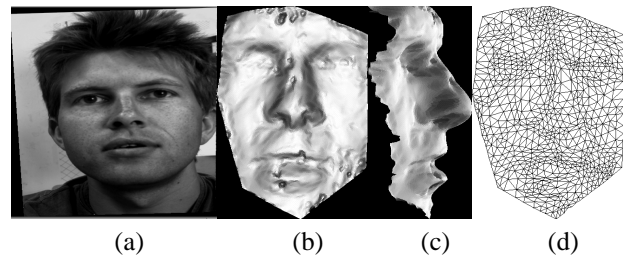
We demonstrate the effectiveness of our approach using face and terrain images.

#### 3.3.1 Faces

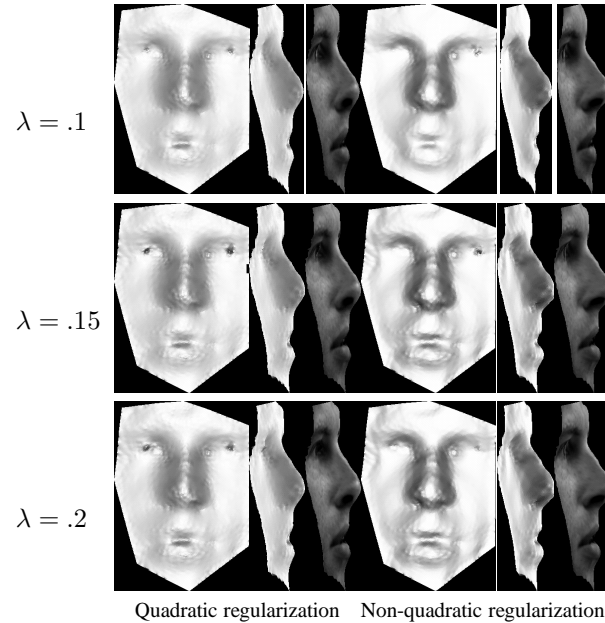
For the face images of Figure 1, we start from a rough mesh obtained by regularly triangulating a raw depth map given by a standard correlation algorithm from calibrated face images [5]. No post-processing whatsoever has been applied to the depth map. The principal curvatures and principal curvature directions are computed on this initial mesh. We then remesh the surface according to this differential information to produce an anisotropic triangulation (785 vertices) that is then deformed by our optimization algorithms, using either quadratic or non-quadratic regularization. In Figure 2, we show reconstruction results obtained using different values of the  $\lambda$  regularization parameter. As  $\lambda$  increases (i.e. the smoothing decreases), the quadratic regularization scheme still oversmooths the lips and the bottom of the nose, while the non-quadratic approach does not.

#### 3.3.2 Terrain

In the example of Figure 3, we focus on the reconstruction of a river bed. We first ran our earlier surface reconstruction technique of Section 2 using 4 image resolution levels, from a 120x117 image pair to a 936x928 one on increasingly dense meshes (from 192 to 11616 vertices). Since we



**Figure 1.** Stereo Reconstruction: (a) Left image of a stereo pair. (b,c) Initial reconstruction. (d) Anisotropic mesh.

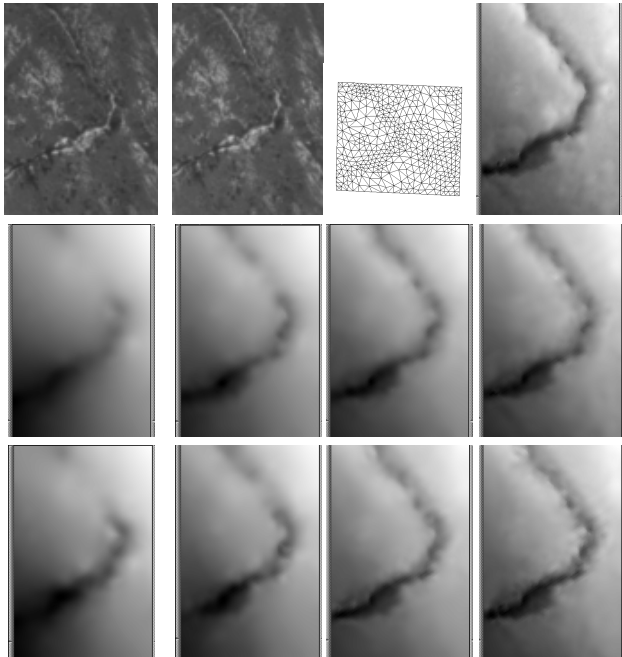


**Figure 2.** Anisotropic reconstruction results using quadratic (3 left columns) and non-quadratic (3 right columns) regularization terms and increasing amount of smoothing. Note that, as the  $\lambda$  stereo parameter increases from .1 in the first row to .2 in the third row, the quadratic approach still oversmooths the lips and the bottom of the nose, while the non-quadratic approach does not. The rightmost results have been texture-mapped.

have no ground truth, we use as a reference reconstruction the result obtained by the finest regular mesh.

The initial solution to the surface reconstruction problem is given by a coarse mesh (192 vertices) obtained by triangulating a depth map from the low resolution images. The result of the first optimization is then retriangulated using differential properties and given as an input to the second level of optimization. This multi-resolution process is then carried on until we reach the highest image and mesh resolution. Our approach allows us to keep only a limited number of vertices (464 at the finest level), even

when dealing with highest resolution images. The reconstruction errors obtained using adaptive meshes with 464 vertices at the highest image resolution lie between those obtained by regular meshes with 744 vertices (2nd image resolution) and with 2928 vertices (3rd image resolution). As shown by Figure 4, using non-quadratic regularization yields much better reconstruction results, especially around the river bed, since this area contains most of the depth variations.

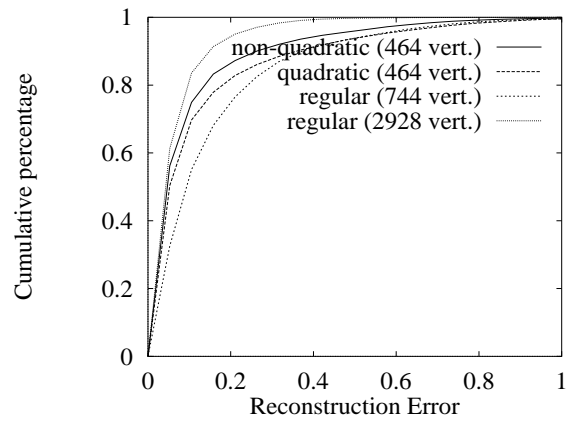


**Figure 3.** Quadratic and non-quadratic reconstruction of a river valley. 1st row: The stereo pair, the anisotropic mesh at the final resolution (464 vertices) and the reference reconstruction (20808 vertices) shown as a depth map. 2nd row: The reconstructions at the four resolution levels using quadratic regularization. 3rd row: The reconstructions at the four resolution levels using non-quadratic regularization

In the example of Figure 5, we use a relatively low level of image resolution (256x256 pixels) to run our anisotropic algorithm and a much higher level of image resolution (1024x1024 pixels) and mesh resolution (20808 vertices) to derive a reference reconstruction. Here again, the non-quadratic regularization yields a much better reconstruction of the central ridge than the quadratic one, using the same amount of smoothing in both cases.

#### 4. Differential constraints

Several of the results shown in the previous section emphasize the classical problems that stereovision must face:



**Figure 4.** The cumulative distribution of reconstruction errors: the reference reconstruction is obtained using a regular mesh with 11616 vertices. Results using the anisotropic mesh (464 vertices) using non-quadratic regularization, the anisotropic mesh (464 vertices) using quadratic regularization, and regular meshes with fewer vertices than the reference one (744 and 2928) are displayed.

in the case of the river bed, occlusions due to vegetation prevent us from obtaining a satisfactory reconstruction of the bottom of the valley. In the face case, specularity in the eye region can produce reconstruction failures in this area. We will see further on that poor quality images can also result in inaccurate reconstruction results. This is why we propose to add to our optimization algorithm constraints based on a priori knowledge on object shapes.

#### 4.1. Constraint Types

The good description and modelling properties brought by differential geometry make it a powerful tool that can be efficiently used to describe in mathematical terms some a priori knowledge. Moreover, since the variables of the optimization problem described earlier are the depth and the partial derivatives of the depth at each mesh vertex, we can easily express differential features such as normals, principal curvatures and principal curvature directions using these variables. Therefore, this leads to a constrained optimization algorithm.

The constraints we apply can be numerical, that is, we constrain the direction of the surface normal, of the principal curvature directions or of the value of one of the principal curvatures. This requires either accurate a priori knowledge of the surface shape in the neighborhood of a given location, or the existence of an object model. In other cases, these constraints can be structural: we only constrain the relations between a set of vertices and their neighbors. In this article, this kind of constraint is exemplified by the crest line constraint. If we know that a crest line must be present

somewhere on the surface, we “translate” this knowledge by specifying curvatures inequalities between vertices on the line and vertices in the direction orthogonal to the line.

## 4.2. Constrained optimization

Here we use a constrained optimization algorithm described in [7] that can be briefly summarized as the iterative achievement of two tasks:

- the projection of the current state vector on the constraint surface
- the optimization of the objective function along the projection of its gradient to the tangent plane to the constraint surface

Since most constraints we use are non-linear, due to the complexity of the formulas linking partial derivatives to principal curvature directions, we use a constraint propagation strategy that enables us to deactivate constraints that cannot be satisfied or that result in too large values of one of the components of the state vector or too large constraint values. This is especially useful when dealing with structural constraints that must be applied along a line.

## 4.3. Results

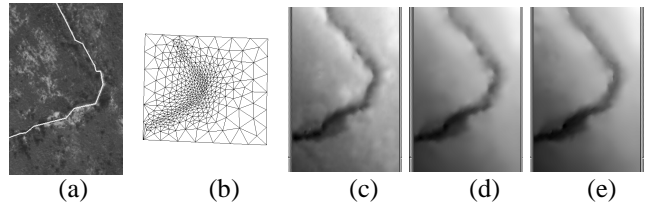
### 4.3.1 Terrain

In the example of Figure 6, we outlined manually the location of the river bed on one of the images. From the camera model and the result of the optimization algorithm of the previous section, we used this specification to remesh the surface and derive a set of mesh vertices where to apply the constraints. In this case, the “river valley” constraints we want to enforce are expressed as:

- Depths of mesh vertices along the line vary monotonically.
- The minimum curvature direction at each vertex along the line is the line direction itself.
- The minimum curvature at each vertex along the line is zero and the maximum curvature is positive.
- The maximum curvature at each vertex is larger than the maximum curvature at neighboring vertices in the direction orthogonal to the crest line (crest line constraint).
- The normal to the surface along the line is normal to the  $(x, y)$  plane.

Since the mesh is denser in the region of the valley, it is also necessary to express differential properties in terms of the depths of the vertices in order to constrain not only the shape of the surface inside the mesh facets (which is achieved using constraints on the partial derivatives of the surface) but also the depths of the vertices themselves. This is done using discrete linear approximations of the curvatures in the direction of the crest line and the direction orthogonal to the crest line. We thus come up with a set of constraints combining linear and non-linear constraints, and equality and inequality constraints. The latter are implemented using an active set strategy: among the inequality constraints, only the unsatisfied ones are activated as equality constraints. Once these constraints are satisfied, they are deactivated.

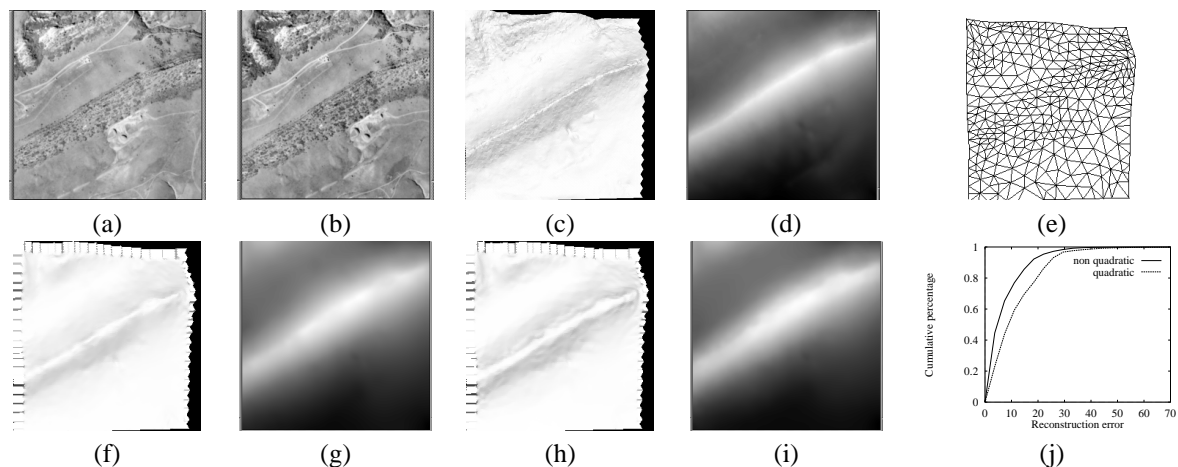
We show successively the reference result of the previous section, our result with the remeshed surface and no constraints and our result incorporating the constraints. It is clear that the latter result shows a better reconstruction of the river bed, especially in the upper part of the image, since it has removed the artifacts brought about by occlusions. This result is thus more consistent with an intuitive knowledge of what a river bed should be.



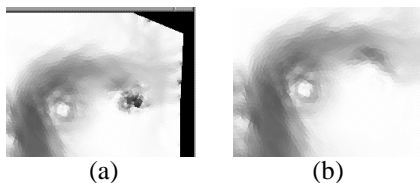
**Figure 6.** Applying a valley constraint. (a) Hand-supplied outline of the valley. (b) Constrained remeshing. (c) Reference reconstruction result using image information only. (d) Anisotropic reconstruction using image information only. (e) Constrained anisotropic reconstruction.

### 4.3.2 Face

In the left eye area in the images of Figures 1 and 2, we seek to improve the reconstruction by imposing that points in this area be umbilical: the two principal curvatures must have the same absolute value. Figure 7 depicts the result. In our adaptive meshing scheme, constraining the principal curvatures at one vertex, called the candidate vertex, is enough to enforce this constraint. Furthermore, we use the right eye area as a reference to constrain the depths of the vertices. The mean of the discrete curvatures in the directions defined by a reference vertex and its neighbors can be computed and used to constrain the depths of the candidate vertex and its neighbors.



**Figure 5.** Anisotropic results using quadratic and non-quadratic regularization. 1st row: The stereo pair (a and b), the reference reconstruction shown as a shaded mesh (c) and a depth map (d), the anisotropic mesh (e). 2nd row: The reconstructions with quadratic (f and g) and non-quadratic regularization (h and i) and the error graph (j). Here again, non-quadratic regularization yields results that are closer to the reference than the quadratic one.



**Figure 7.** Imposing a curvature constraint in the eye region. (a) Detail of the reconstruction of Figure 2 exhibiting a sharp peak. (b) Improved reconstruction after imposing an umbilical point constraint.

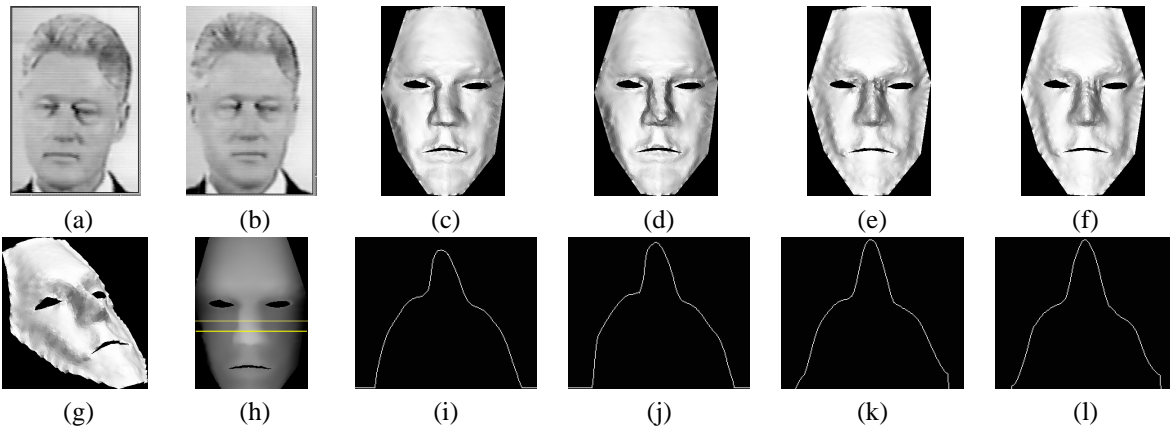
### 4.3.3 Video sequence

The  $168 \times 248$  face images of Figure 8 are of extremely poor quality because they have been digitized from an NTSC video tape. The calibration and initial reconstruction were computed by deforming a generic face model using a bundle-adjustment technique [6]. We are focusing here on the nose area, that is badly recovered in the initial reconstruction. After rotation, translation and scaling of the face model and registration in the same frame as the initial reconstruction, we specify the location of the nose ridge and remesh the surface according to that information. We then know which vertices must lie on the nose ridge. We define a set of vertices and facets on which we apply constraints. These facets and vertices will typically be adjacent facets and neighboring vertices to the vertices lying on the ridge. We can compute the normals to the facets of the surface in the generic model (used as reference) and use this information as constraints to the corresponding facets on the can-

didate mesh. This will constrain the depths of concerned vertices. In addition, we also constrain the principal curvatures, the principal curvature directions and the normals at the vertices of the candidate mesh using the face model and a dense resampling of the model (this resampling is computed from the estimation of the partial derivatives of the surface and the approximation polynomials of the previous section). This constrains the partial derivatives of the surface at each concerned vertex. These constraints are only enforced at the beginning of the process, since there is no way to know whether the candidate nose has exactly the shape of the generic nose, but these weak constraints are used to better condition the state vector before we run the optimization algorithm with hard constraints. These hard constraints will enforce that, throughout the optimization process, the nose ridge be a crest line: this constraint is imposed on the partial derivatives and on the depths through a discrete approximation. The poor quality of the images can generate artifacts like the one created in the left eye region. This can be improved if we further add an umbilical point constraint on the partial derivatives. The final reconstruction of the nose shows a much better symmetry than the initial one, even if we have not explicitly integrated symmetry constraints into the process.

## 5. Conclusion

In this article, we have presented a generic framework for 3-D reconstruction based on the minimization of an objective function. It incorporates differential constraints based on a priori information about the local 3-D object shape.



**Figure 8.** Using low quality images. (a,b)  $168 \times 248$  images digitized from an NTSC video tape. (c) Initial reconstruction with crooked nose. (d) Reconstruction after imposing the initial constraints. (e) Reconstruction after imposing the crest line constraint. (f,g) Reconstruction after imposing the umbilical point constraint. (h) Corresponding depth map. (i,j) Depth along the two scan lines shown in (h) for the initial reconstruction. (k,l) Depth along the same two scan lines for the final reconstruction. Note that they are much more symmetric in the nose region, even though no symmetry constraint has been explicitly applied.

First, we presented our minimization scheme that uses anisotropic triangulated meshes and non-quadratic regularization to generate 3-D reconstructions that preserve shape variations and requires only triangulations having relatively few vertices.

We then addressed several classical stereovision problems that can hardly be solved using the image data and the regularization constraint alone. The differential constraints used can be numerical or structural and our optimization scheme can efficiently deal with non-linear equality and inequality constraints. We demonstrated on aerial and face images that our approach is generic enough to be applied to various cases where classical stereo algorithms fail.

## References

- [1] M. Black. Recursive non-linear estimation of discontinuous flow fields. In *Third European Conference on Computer Vision*, pages 138–145, May 1994.
- [2] 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.
- [3] I. Cohen and I. Herlin. Non uniform multiresolution method for optical flow and phase portrait models: Environmental applications. *International Journal of Computer Vision*, 33, 1:29–49, September 1999.
- [4] M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, 1976.
- [5] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision Applications*, 6(1), pages 35–49, 1993.
- [6] P. Fua. Regularized Bundle-Adjustment to Model Heads from Image Sequences without Calibration Data. *International Journal of Computer Vision*, 38(2), July 2000.
- [7] P. Fua and C. Brechbuhler. Imposing hard constraints on soft snakes. In *European Conference on Computer Vision, Cambridge, UK*, volume II, pages 495–506, 1996.
- [8] P. Fua and Y. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal on Computer Vision*, 16:35–56, September 1995.
- [9] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal on Computer Vision*, 1(4): 321–331, 1988.
- [10] 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.
- [11] O. Monga, R. Lengagne, and R. Deriche. Extraction of the zero-crossing of the curvature derivative in volumic 3d medical images: a multi-scale approach. In *Conference on Computer Vision and Pattern Recognition, 1994, Seattle, USA*, 1994.
- [12] L. Robert and R. Deriche. Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities. In *European Conference on Computer Vision, Cambridge, U.K.*, volume 1, pages 439–451, 1996.
- [13] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [14] O. Zienkiewicz and R. Taylor. *The finite element method*, volume 1,2. Mc Graw-Hill, 1988.