SRI International

Using 3-Dimensional Meshes to Combine Image-Based and Geometry Constraints

Technical Note No. 536 (Revised)

August 24, 1994

By: Pascal V. Fua, Computer Scientist
Yvan G. Leclerc, Computer Scientist
Artificial Intelligence Center
Computing and Engineering Sciences Division

Approved for Public Release; Distribution Unlimited.

This work was supported in part by the Advanced Research Projects Agency under Contract No. DACA76-92-C-0008 and DACA76-92-C-0034.

333 Ravenswood Avenue • Menlo Park, CA 94025-3493 • (415) 326-6200 • FAX: (415) 326-5512 • Telex: 334486

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
24 AUG 1994		**00-08-1994 to 00-08-1994**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Using 3-Dimensional Meshes to Combine Image-Based and Geometry Constraints	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **SRI International,333 Ravenswood Avenue,Menlo Park,CA,94025**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**30**	

Using 3–Dimensional Meshes To Combine Image-Based and Geometry-Based Constraints

P. Fua and Y.G. Leclerc
SRI International
333 Ravenswood Avenue, Menlo Park, CA 94025, USA
(fua@ai.sri.com leclerc@ai.sri.com)

Abstract

A unified framework for 3–D shape reconstruction allows us to combine image-based and geometry-based information sources. The image information is akin to stereo and shape-from-shading, while the geometric information may be provided in the form of 3–D points, 3–D features or 2–D silhouettes. A formal integration framework is critical in recovering complicated surfaces because the information from a single source is often insufficient to provide a unique answer.

Our approach to shape recovery is to deform a generic object-centered 3–D representation of the surface so as to minimize an objective function. This objective function is a weighted sum of the contributions of the various information sources. We describe these various terms individually, our weighting scheme, and our optimization method. Finally, we present results on a number of difficult images of real scenes for which a single source of information would have proved insufficient.

Keywords : Surface reconstruction, Stereo, Shape-from-shading,Silhouettes, Geometric constraints.

1 Introduction

The recovering of surface shape from image cues, the so-called "shape from X" problem, has received tremendous attention in the computer vision community. But no single source of information "X," be it stereo, shading, texture, geometric constraints or any other, has proved to be sufficient across a reasonable sampling of images. To get good reconstructions of a surface, it is necessary to use as many different kinds of cues with as many views of the surface as possible. In this paper, we present and demonstrate a working framework for surface reconstruction that combines image cues, such as stereo and shape-from-shading, with geometric constraints, such as those provided by laser range finders, area- and edge-based stereo algorithms, linear features, and silhouettes.

Our framework can incorporate cues from many images of a surface, even when the images are taken from widely differing viewpoints, accommodating such viewpoint-dependent effects as self-occlusion and self-shadowing. It accomplishes this by using a full 3–D object-centered representation of the estimated surface. This representation is then used to generate synthetic views of the estimated surface from the viewpoint of each input image. By using standard computer graphics algorithms, those parts of the surface that are hidden from a given viewpoint can be identified and consequently eliminated from the reconstruction process. The remaining parts are then in correspondence with the input images, and the images and corresponding cues are applied to the reconstruction of the surface in an iterative manner using an optimization algorithm.

Recent publications describe the reconstruction of a surface using 3–D object-centered representations, such as 2.1/2–D grids [Robert *et al.*, 1992], 3–D surface meshes [Cohen *et al.*, 1991, Delingette *et al.*, 1991, Terzopoulos and Vasilescu, 1991, Vemuri and Malladi, 1991, McInerney and Terzopoulos, 1993, Koh *et al.*, 1994], parameterized surfaces [Stokely and Wu, 1992, Lowe, 1991], local surfaces [Ferrie *et al.*, 1992, Fua and Sander, 1992], particle systems [Szeliski and Tonnesen, 1992], and volumetric models [Pentland, 1990, Terzopoulos and Metaxas, 1991, Pentland and Sclaroff, 1991]. Most of these rely on previously computed 3–D data, such as the coordinates of points derived from laser range finders or correlation-based stereo algorithms, and reconstruct the surface by fitting it to these data in a least-squares sense. In other words, the derivation of the 3–D data from the images is completely divorced from the reconstruction of the surface.

In contrast, our framework allows us to directly use such image cues as stereo, shading, and silhouette edges in the reconstruction process while simultaneously incorporating previously computed 3–D data such as those mentioned above. In a previous publication [Fua and Leclerc, 1994] we describe how stereo and shading are used within the framework described below, and the relationship of this approach to previous work. Here, we focus on how an additional image cue (silhouette edges) and previously computed 3–D data are incorporated into our reconstruction process.

Combining these different sources of information is not a new idea in itself. For example, Blake *et al.* [1985] is the earliest reference we are aware of that discusses the complementary nature of stereo and shape-from-shading. Both Cryer *et al.* [1992] and Heipke *et al.* [1992] have proposed algorithms to combine shape-from-shading and stereo, while Liedtke *et al.* [1991] first uses silhouettes to derive an initial estimate of the surface, and then applies a multi-image stereo algorithm to improve the result. However, none of the algorithms we know of uses an object-centered representation and

an optimization procedure that are general enough to incorporate all of the cues that we present here. This generality should also make possible the use of a very wide range of other sources of information, such as shadows, in addition to those actually discussed here.

We view the contribution of this paper as providing both the framework that allows us to combine diverse sources of information in a unified and computationally effective manner, and the specific details of how these diverse sources of information are derived from the images.

In the next section, we describe our framework and the new information sources introduced here. Following this, we demonstrate that the framework successfully performs its function on real images and allows us to achieve results that are better than those we could derive from any one, or even two, sources of information.

2 Framework

Our approach to recovering surface shape and reflectance properties from multiple images is to deform a 3–D representation of the surface so as to minimize an objective function. The free variables of this objective function are the coordinates of the vertices of the mesh representing the surface, and the process is started with an initial estimate of the surface. Here we assume that images are monochrome, and that their camera models are known *a priori*.

We represent a surface $\mathcal{S}$ by a hexagonally connected set of vertices $\mathbf{V} = (v_1, v_2, \ldots, v_{n_v})$ called a *mesh*. The position of vertex v_j is specified by its Cartesian coordinates (x_j, y_j, z_j). Each vertex in the interior of the surface has exactly six neighbors.

Neighboring vertices are further organized into triangular planar surface elements called *facets*, denoted $\mathbf{F} = (f_1, f_2, \ldots, f_{n_f})$. The vertices of a facet are also ordered in a clockwise fashion. In this work, we require that the initial estimate of the surface have facets whose sides are of equal length. The objective function described below tends to maintain this equality, but does not strictly enforce it. In the course of the optimization, we refine the mesh by iteratively subdiving the facets into four smaller ones whose sides are still of roughly equal length.

In Figure 1, we show a shaded view and a wireframe representation of such a mesh. We also show what we call a "Facet-ID" image. For each input image, it is generated by encoding the index i of each facet f_i as a unique color, and projecting the surface into the image plane, using a standard hidden-surface algorithm. As discussed in Sections 2.3 and 2.4, we use it to determine which surface points are occluded in a given view and on which facets geometric constraints should be brought to bear.

2.1 Objective Function and Optimization Procedure

The objective function $\mathcal{E}(\mathcal{S})$ that we use to recover the surface is a sum of terms that take into account the image-based constraints—stereo and shape-from-shading—and the geometry-based constraints—features and silhouettes—that are brought to bear on the surface. To minimize $\mathcal{E}(\mathcal{S})$, we use an optimization method that is inspired by the heuristic technique known as a continuation method [Terzopoulos, 1986, Leclerc, 1989a, Leclerc, 1989b] in which we add a regularization term to the objective function and progressively reduce its influence. We define the total energy of the

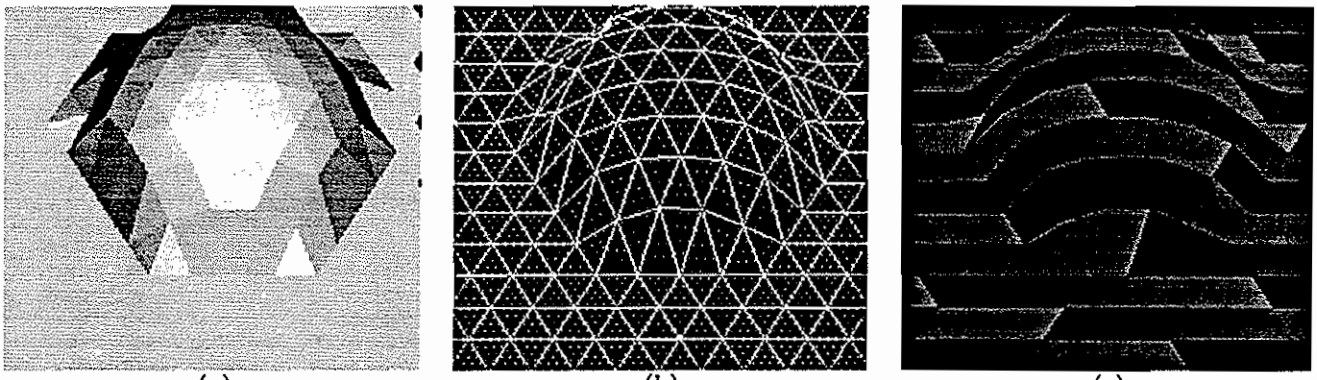

Figure 1: Projection of a mesh, and the Facet-ID image used to accommodate occlusions during surface reconstruction: (a) A shaded image of a mesh. (b) A wire-frame representation of the mesh (bold white lines) and the sample points in each facet (interior white points). (c) The Facet-ID image, wherein the color at a pixel is chosen to uniquely identify the visible facet at that point (shown here as a gray-level image).

mesh, $\mathcal{E}_T(\mathcal{S})$, as

$$\begin{aligned} \mathcal{E}_T(\mathcal{S}) &= \lambda_D \mathcal{E}_D(\mathcal{S}) + \mathcal{E}(\mathcal{S}) \\ \mathcal{E}_(\mathcal{S}) &= \sum_i \lambda_i \mathcal{E}_i(\mathcal{S}) \ . \end{aligned} \tag{1}$$

The $\mathcal{E}_i(\mathcal{S})$ represent the image and geometry-based constraints, and the λ_i their relative weights, as defined below. $\mathcal{E}_D(\mathcal{S})$, the regularization term, serves a dual purpose. First, we define it as a quadratic function of the vertex coordinates, so that it "convexifies" the energy landscape when λ_D is large and improves the convergence properties of the optimization procedure. Second, as shown in the appendix, in the presence of noise, some amount of smoothing is required to prevent the mesh from overfitting the data, and excessively wrinkling the surface.

In our implementation, we take $\mathcal{E}_D$ to be a measure of the curvature or local deviation from a plane at every vertex. We approximate this as follows.

Consider a perfectly planar hexagonal mesh for which the distances between neighboring vertices are exactly equal. Let the neighbors of a vertex v_i be ordered in clockwise fashion and let us denote them $v_{N_i(j)}$ for $1 \leq j \leq 6$. This notation is depicted in Figure 2(a). If the hexagonal mesh was perfectly planar, then the third neighbor over from the j^{th} neighbor, $v_{N_i(j+3)}$, would lie on a straight line with v_i and $v_{N_i(j)}$. Given that the intervertex distances are equal, this implies that coordinates of v_i equal the average of the coordinates of $v_{N_i(j)}$ and $v_{N_i(j+3)}$, for any j.

Given the above, we can write a measure of the deviation of the mesh from a plane as follows:

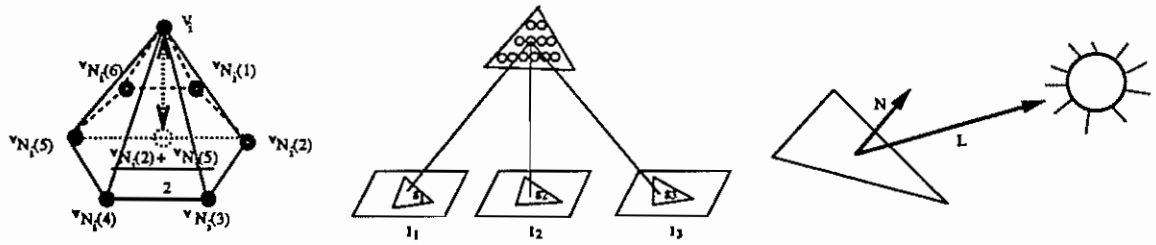

Figure 2: Vertices and facets of a mesh: (a) The six neighbors $N_i(j)$ of a vertex v_i are ordered clockwise. The deformation component of the objective function tends to minimize the distance between v_i and the midpoint of diametrically opposed neighbors, represented by the dotted circle. (b) Facets are sampled at regular intervals as illustrated here. The stereo component of the objective function is computed by summing the variance of the gray level of the projections of these sample points, the g_is. (c) The albedo of each facet is estimated using the facet normal $\vec{N}$, the light source direction $\vec{L}$, and the average gray level of the projection of the facet into the images. The shading component of the objective function is the sum of the squared differences in estimated albedo across neighboring facets.

$$\mathcal{E}_D(\mathcal{S}) = \sum_{i=1}^{n_v} \sum_{\substack{j=1 \\ k=N_i(j) \\ k'=N_i(j+3)}}^{3} (2x_i - x_k - x_{k'})^2 + (2y_i - y_k - y_{k'})^2 + (2z_i - z_k - z_{k'})^2 \qquad (2)$$

Note that this term is also equivalent to the squared directional curvature of the surface when the sides have equal lengths [Kass *et al.*, 1988]. This term can be made to accommodate multiple resolutions of facets by normalizing each term by the nominal intervertex spacing of the facets.

In previous implementations [Fua and Leclerc, 1994], we have performed optimization using a standard conjugate-gradient descent procedure [Press *et al.*, 1986]. However, the $\mathcal{E}_D$ term described here is amenable to a "snake-like" optimization technique [Kass *et al.*, 1988]. We embed the curve in a viscous medium and solve the equation of dynamics

$$\begin{aligned} \frac{\partial \mathcal{E}_T}{\partial S} + \alpha \frac{dS}{dt} &= 0, \\ \text{with } \frac{\partial \mathcal{E}_T}{\partial S} &= \frac{\partial \mathcal{E}_D}{\partial S} + \frac{\partial \mathcal{E}}{\partial S}, \end{aligned} \qquad (3)$$

where $\mathcal{E}_T$ is the total energy of Equation 1, α the viscosity of the medium, and S the state vector that defines the current position of the mesh that is the vector of the x,y, and z coordinates of the vertices. Since the deformation energy $\mathcal{E}_D$ in Equation 2 is quadratic, its derivative with respect to

S is linear, and therefore Equation 3 can be rewritten as

$$\begin{aligned} K_S S_t + \alpha(S_t - S_{t-1}) &= -\left.\frac{\partial \mathcal{E}}{\partial S}\right|_{S_{t-1}} \\ \Rightarrow (K_S + \alpha I) S_t &= \alpha S_{t-1} - \left.\frac{\partial \mathcal{E}}{\partial S}\right|_{S_{t-1}} \end{aligned} \tag{4}$$

where

$$\frac{\partial \mathcal{E}_D}{\partial S} = K_S S,$$

and K_S is a sparse matrix. Note that the derivatives of $\mathcal{E}_D$ with respect to x,y, and z are decoupled so that we can rewrite Equation 4 as a set of three differential equations in the three spatial coordinates:

$$\begin{aligned} (K + \alpha I) X_t &= \alpha X_{t-1} - \left.\frac{\partial \mathcal{E}}{\partial X}\right|_{X_{t-1}} \\ (K + \alpha I) Y_t &= \alpha Y_{t-1} - \left.\frac{\partial \mathcal{E}}{\partial Y}\right|_{Y_{t-1}} \\ (K + \alpha I) Z_t &= \alpha Z_{t-1} - \left.\frac{\partial \mathcal{E}}{\partial Z}\right|_{Z_{t-1}} \end{aligned}$$

where X,Y, and Z are the vectors of the x,y, and z coordinates of the vertices, and K is a sparse matrix. In fact, for our hexagonal meshes, K turns out to be a banded matrix and this set of equations can be computed efficiently using LU decomposition and backsubstitution. Note that the LU decomposition need be recomputed only when α changes. When α is constant, only the backsubstitution step is required. In practice α is computed automatically at the start of the optimization procedure so that a prespecified average vertex motion amplitude is achieved [Fua and Leclerc, 1990]. The optimization proceeds as long as the total energy decreases; when it increases the algorithm backtracks and increases α, thereby decreasing the step size.

We can optimize all three spatial components simultaneously. However, when dealing with surfaces for which motion in one direction leads to more dramatic changes than motions in others, as is typically the case with the z direction in Digital Elevation Models (DEMs), we have found the following heuristic to be useful. We first fix the x and y coordinates of vertices and adjust z alone. Once the surface has been optimized, we then allow all three coordinates to vary.

To speed the computation and prevent the mesh from becoming stuck in undesirable local minima, we typically use several levels of mesh size—three in the examples of Section 3—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.

2.2 Combining the Components

The total energy of Equation 1 is a sum of terms whose magnitudes are image- or geometry-dependent and are therefore not necessarily commensurate. One therefore needs to scale them

appropriately, that is to define the λ weights so as to make the magnitude of their contributions commensurate and independent of the specific radiometry or geometry of the scene under consideration.

From Equation 4, it can be seen that the dynamics of the optimization are controlled by the gradient of the objective function. As a result, we have found that an effective way to normalize the contributions of the various components of the objective function is to define a set of user-specified weights λ_i' such that

$$\sum_{1 \leq i \leq n} \lambda_i' < 1 \, .$$

These weights are then used to define the λs as follows

$$\begin{aligned} \lambda_i &= \frac{\lambda_i'}{\| \vec{\nabla} \mathcal{E}_i(\mathcal{S}^0) \|} \\ \lambda_D &= \frac{\lambda_D'}{\| \vec{\nabla} \mathcal{E}_D(\mathcal{S}^0) \|} \\ \lambda_D' &= f_w(\sum_i \lambda_i') \end{aligned} \tag{5}$$

where f_w is a monotonically decreasing function that approaches zero as $\sum_i \lambda_i'$ approaches one and $\mathcal{S}^0$ is the surface estimate at the start of each optimization step. In our implementation, we take $f_w(x) = ((1-x)/x)^2$ so that the regularization term has the same influence as the sum of all the others when $\sum_i \lambda_i' = 0.5$ We first proposed this normalization scheme in [Fua and Leclerc, 1990], and it is analogous to standard constrained optimization techniques in which the various constraints are scaled so that their eigenvalues have comparable magnitudes [Luenberger, 1984]. In practice we have found that, because the normalization makes the influence of the various terms comparable irrespective of actual radiometry or dimensions, the user-specified λ_i' weights are context-specific but not image-specific. In other words, we use one set of parameters for images of faces when combining stereo, shape-from-shading, and silhouettes, and another when dealing with aerial images of terrain using stereo and 3-D point constraints, but we do not have to change them for different faces or different landscapes. In our appendix, we use synthetic data to illustrate the behavior of our weighting scheme and its robustness, and in Section 3 we demonstrate its effectiveness in practice.

The continuation method of Section 2.1 is implemented by taking the initial value of $\sum_i \lambda_i'$ to be 0.5 and then progressively decreasing it while keeping the relative values of the λ_i's constant. We demonstrate our method's behavior using the aerial images of Figure 3 and evaluate our results against the "ground truth" supplied to us by a photogrammetrist from Ohio State University. In this example, we initialize a coarse resolution mesh by interpolating a correlation map derived using the images reduced by a factor of four. We first apply our continuation method to this coarse mesh using the stereo component of the objective function that is introduced in Section 2.4. Next, as discussed in Section 2.1, we increase the resolution of both the images and the mesh, reoptimize and repeat the process once more. At each level of resolution, as λ_D' decreases, the discrepancy between our surface model and the control points diminishes. In Figure 4(a,b,c), we show the corresponding

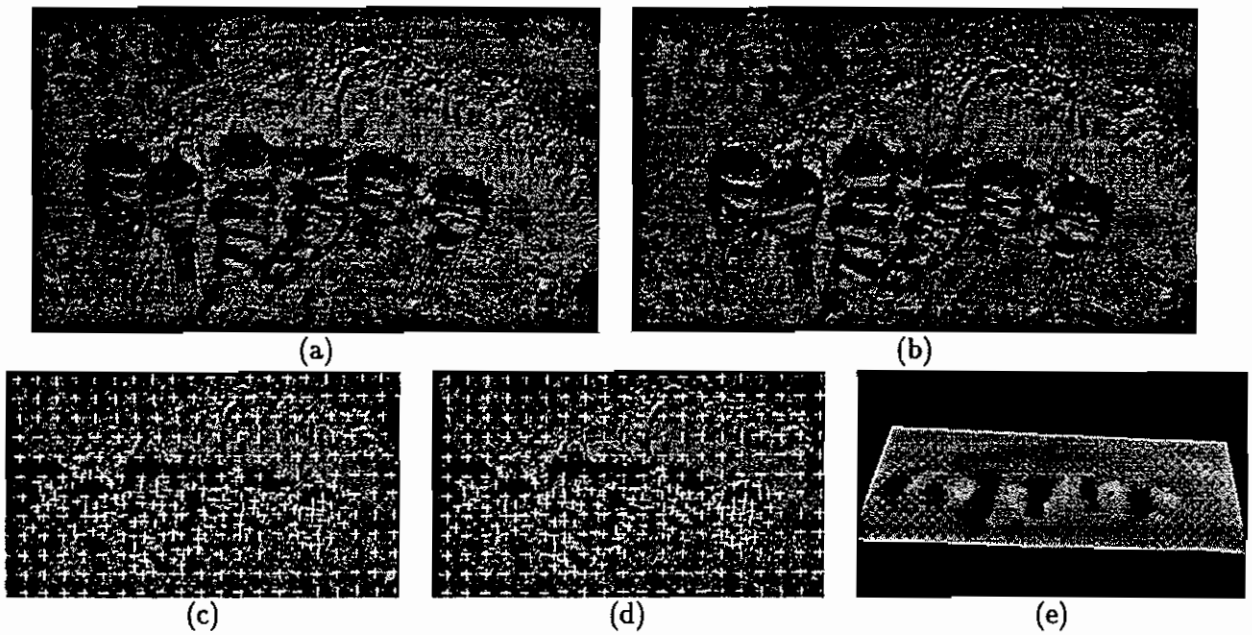

Figure 3: A test data set (courtesy of Ohio State University): (a,b) An aerial stereo pair. (c,d) Matched pair of points hand-entered by a photogrammetrist. (e) Shaded view of the triangulated surface formed by the corresponding 3–D points.

optimized meshes. In Figure 4(d), we plot the RMS[1] distance of the control points to the surface at the end of each optimization step. The final error at each level of resolution, denoted by the thick vertical lines, corresponds to an error in measured disparity that is smaller than half a pixel. Given the fact that the control points are not necessarily perfect themselves, this is the kind of performance one would expect of a precise stereo system [Güelch, 1988].

However, the real strength of our approach lies in the fact that it allows us to combine image-based constraints such as stereo with geometric constraints such as the ones introduced below, thereby making the reconstruction more robust in difficult situations.

Note that the photogrammetrist generated more control points in comparatively high-relief areas of the images of Figure 3(a,b) so that their triangulation, shown in Figure 3(c), forms an irregular mesh or TIN[2]. As shown in [McInerney and Terzopoulos, 1993, Koh *et al.*, 1994], the optimization of such irregular meshes can be achieved using a finite-element method. Our whole approach could therefore be extended to such irregular meshes and this will be the subject of future work.

[1]Root Mean Square
[2]Triangular Irregular Network

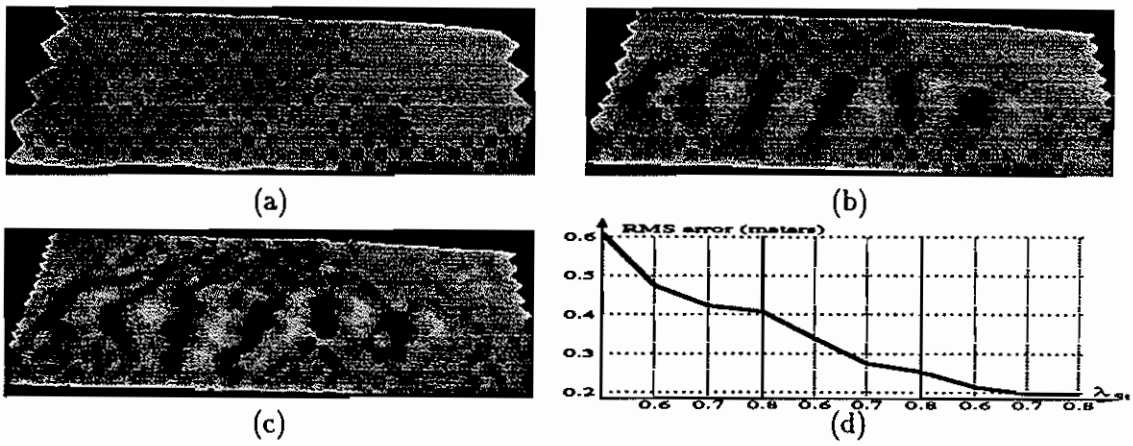

Figure 4: Behavior of the continuation method of Section 2.1: (a,b,c) Shaded views of the reconstructed surface at each level of resolution. At the coarsest level the images are 110x64 in size and the mesh vertices form a 24x23 array. To go from one level to the next, the image dimensions are doubled and each mesh facet is subdivided into four. (d) A plot of the RMS distance, in meters, of the control points of Figure 3(c,d) to the surface as the optimization proceeds. The thick vertical lines indicate a change in resolution and the dotted ones an increase by 0.1 of the stereo weight λ_{St} and corresponding decrease in the λ'_D regularization weight. At the highest resolution, an elevation error of 0.2 meter corresponds to an error of approximately 0.4 pixel in disparity.

2.3 Geometric Constraints

We have explored the constraints generated by 3–D points, 3–D linear features, and 2–D silhouettes.

2.3.1 3–D Points

3–D Points are treated as attractors and 3–D linear features are taken to be collections of such points. The easiest way to handle attractors is to model each one as a spring by adding the following term to the objective function

$$e_a = 1/2((x_a - x_i)^2 + (y_a - y_i)^2 + (z_a - z_i)^2) \tag{6}$$

where x_i,y_i, and z_i are the coordinates of the mesh vertex closest to the attractor (x_a, y_a, z_a). This, however, is inadequate if one wishes to use facets that are large enough so that attracting the vertices, as opposed to the surface point closest to the attractor, would cause unwarranted deformations of the mesh. This is especially important when using a sparse set of attractors. In this case, the energy term of Equation 6 must be replaced by one that attracts the surface without warping it. In our implementation, this is achieved by redefining e_a as

$$e_a = 1/2d_a^2 \tag{7}$$

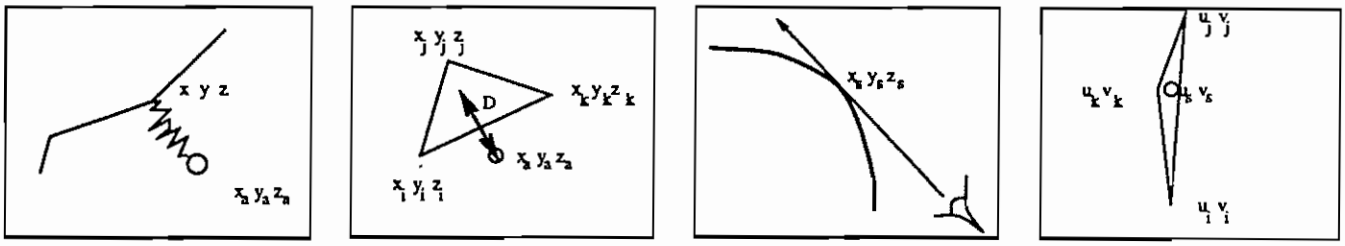

Figure 5: 3–D and 2–D point constraints: (a) Point attractor modeled as a spring attached to a vertex. (b) Point attractor modeled as a spring attached to the closest surface point. (c) Occlusion contours are the locus of the projections of the (x_s, y_s, z_s) surface points for which a camera ray is tangential to the surface. (d) In practice, the (u_s, v_s) projection of such a point must be colinear with the projections of the vertices of the facet that produces the observed silhouette edge.

where d_a is the orthogonal distance of the attractor to the closest facet. The normal vector to a facet can be computed as the normalized cross product of the vectors defined by two sides of that facet, and d_a as the dot product of this normal vector with the vector defined by one of the vertices and the attractor. Letting $(x_i, y_i, z_i)_{1 \leq i \leq 3}$ be the three vertices of a facet, consider the polynomial D defined as

$$\begin{aligned} D &= \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_a & y_a & z_a & 1 \end{vmatrix} \\ &= C_x x + C_y y + C_z z \end{aligned}$$

where C_x,C_y, and C_z are polynomial functions of x_i,y_i, and z_i. It is easy to show that the facet normal is parallel to the vector (C_x, C_y, C_z) and that the square of the orthogonal distance d_a^2 of the attractor to the facet can be computed as

$$d^2 = D^2/(C_x^2 + C_y^2 + C_z^2)$$

Finding the "closest facet" to an attractor is computationally expensive in general. However, in our specific case the search can be made efficient and fast if we assume that the 3–D points can be identified by their projection in an image. We project the mesh in that image, generate the corresponding Facet-ID image—which must be done in any case for other computations—and look up the facet number of the point's projection. This applies, for example, to range maps, edge- or correlation-based stereo data, and hand-entered features that can be overlaid on various images.

We typically recompute the facet attachments at every iteration of the optimization procedure so as to allow facets to slide as necessary. Since the points can potentially come from any number of such images, this method can be used to fuse 3–D data from different sources.

2.3.2 Silhouettes

Contrary to 3–D edges, silhouette edges are typically 2–D features since they depend on the viewpoint and cannot be matched across images. However, as shown in Figure 5(c), they constrain the surface tangent. Each point of the silhouette edge defines a line that goes through the optical center of the camera and is tangent to the surface at its point of contact with the surface. The points of a silhouette edge therefore define a ruled surface that is tangent to the surface. In terms of our facetized representation, this can be expressed as follows. Given a silhouette point (u_s, v_s) in an image, there must be a facet with vertices $(x_i, y_i, z_i)_{1 \leq i \leq 3}$ whose image projections $(u_i, v_i)_{1 \leq i \leq 3}$, as well as (u_s, v_s), all lie on a single line as depicted by Figure 5(d). This implies that the three determinants of the form

$$\begin{vmatrix} u_i & u_j & u_s \\ v_i & v_j & v_s \\ 1 & 1 & 1 \end{vmatrix}, \; 1 \leq i \leq 3, i < j \leq 3$$

must be equal to zero. We enforce this for each silhouette point by adding to the objective function a term of the form

$$e_s = 1/2 \sum_{1 \leq i \leq 3, i < j \leq 3} \begin{vmatrix} u_i & u_j & u_s \\ v_i & v_j & v_s \\ 1 & 1 & 1 \end{vmatrix}^2 \tag{8}$$

where the (u_i, v_i)s are derived from the (x_i, y_i, z_i) using the camera model.

As with the 3–D attractors described in Section 2.3.1, the main problem is to find the "silhouette facet" to which the constraint applies. Since the silhouette point (u_s, v_s) can lie outside the projection of the current estimate of the surface, we search the Facet-ID image in a direction normal to the silhouette edge for a facet that minimizes e_s and that is therefore the most likely to produce the silhouette edge. This, in conjunction with our coarse-to-fine optimization scheme, has proved a robust way of determining which facets correspond to silhouette points.

2.4 Image Constraints

In this work, we use two complementary image-based constraints: stereo and shape-from-shading.

The stereo component of the objective function is derived by comparing the gray levels of the points in all of the images for which the projection of a given point on the surface is visible, as determined using the Facet-ID image. This comparison is done for a uniform sampling of the surface, as shown in Figure 2(b). This method allows us to deal with arbitrarily slanted regions and to discount occluded areas of the surface.

The shading component, depicted in Figure 2(c), of the objective function is computed using a method that does not invoke the traditional assumption of constant albedo. Instead, it attempts

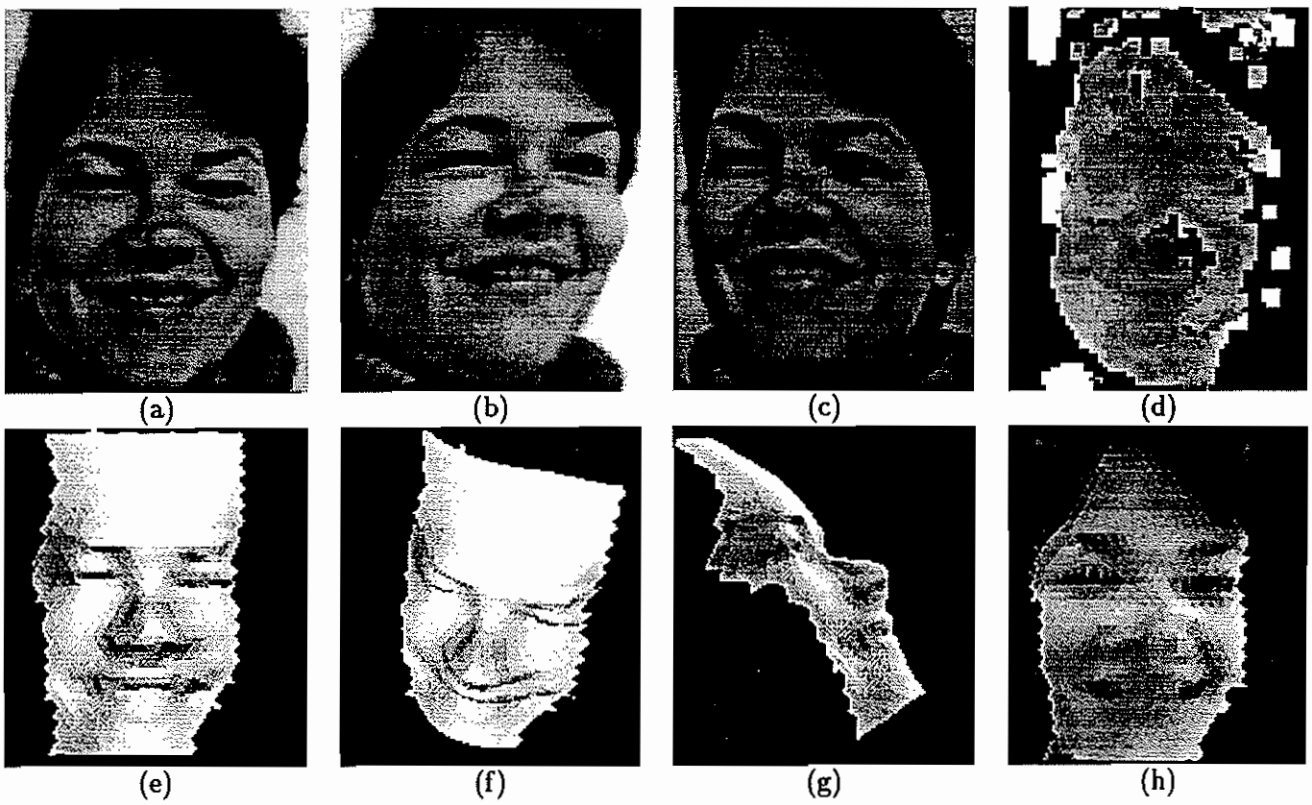

Figure 6: Recovering the shape of a face by combining stereo and shape-from-shading: (a) (b) (c) Triplet of face images (courtesy of INRIA). (d) Disparity map. (e) (f) (g) Shaded views of the reconstructed surface after optimization. (h) The recovered albedo map.

to minimize the variation in albedo across the surface, and can therefore deal with both constant albedo surfaces as well as surfaces whose albedo varies slowly.

Stereo information is very robust in textured regions but potentially unreliable elsewhere. We therefore use it mainly in textured areas by weighting the stereo component most strongly for facets of the triangulation that project into textured image areas. Conversely, the shading information is more reliable where there is little texture and is weighted accordingly.

These two terms are central to our approach: they are the ones that allow the combination of geometric information with image information. However, since their behavior and implementation have already been extensively discussed elsewhere, we do not describe them any further here and refer the interested reader to our previous publication [Fua and Leclerc, 1994]. In Figure 6, we show the reconstruction of a face using only stereo and shape-from-shading.

3 Applications

Our framework allows us to combine geometric constraints with image-based constraints to derive surface reconstructions and to refine previously computed surfaces. Here, we demonstrate its capabilities using difficult imagery.

3.1 From 3–D Constraints to Detailed Surfaces

Our system deals with the various sources of 3–D information, whether dense, such as range maps or correlation-based stereo disparity maps, or linear, such as hand-entered features or edge-based stereo disparity maps, in the same fashion. Both are sampled at regular intervals to generate collections of 3–D attractors that are used to define energy terms using Equation 6 or 7.

Especially in the case of sparse features, the "snake-type" optimization technique of Section 2.1 has proved more effective than more classical techniques such as conjugate gradient at propagating constraints across the mesh.

3.1.1 Dense 3–D Data

In Figure 7, we show an image of a face and a corresponding range map computed using structured light. Although it is fairly accurate, this particular method introduces artifacts that are highlighted in Figure 7(c). We first fit a surface to these points by starting from a flat surface and taking the total energy $\mathcal{E}_T$ of Equation 1 to be

$$\begin{aligned} \mathcal{E}_T &= \lambda_D \mathcal{E}_D + \lambda_A \mathcal{E}_A \\ &= \lambda_D \mathcal{E}_D + \lambda_A \sum_a e_a \end{aligned} \tag{9}$$

where the e_a are defined for each range-data point as the attraction terms of Equation 7. Because of the artifacts of the original range data, the resulting surface is approximately correct but excessively wrinkly, as shown in Figure 7(d) and (e). Of course, we could simply smooth the surface but we would then be at risk of losing important details such as the mouth or the fine structures on the side of the nose. Our approach provides us with a better way of dealing with this problem: we can fuse the range information with the shading information of the intensity image of Figure 7(a). To do so, we add to $\mathcal{E}_T$ the shading term defined in Section 2.4, that we denote $\mathcal{E}_{Sh}$:

$$\mathcal{E}_T = \lambda_D \mathcal{E}_D + \lambda_A \mathcal{E}_A + \lambda_{Sh} \mathcal{E}_{Sh}.$$

We restart the optimization from the flat initial surface. The new surface, shown in Figure 8, is much smoother, but the mouth is well preserved and the side of the nose better defined. Note, however, that in the side views the bottom of the nose is not flat enough. This is not surprising since the shading information is of no use there. We address this problem in Section 3.2.

3.1.2 Sparse 3–D Data

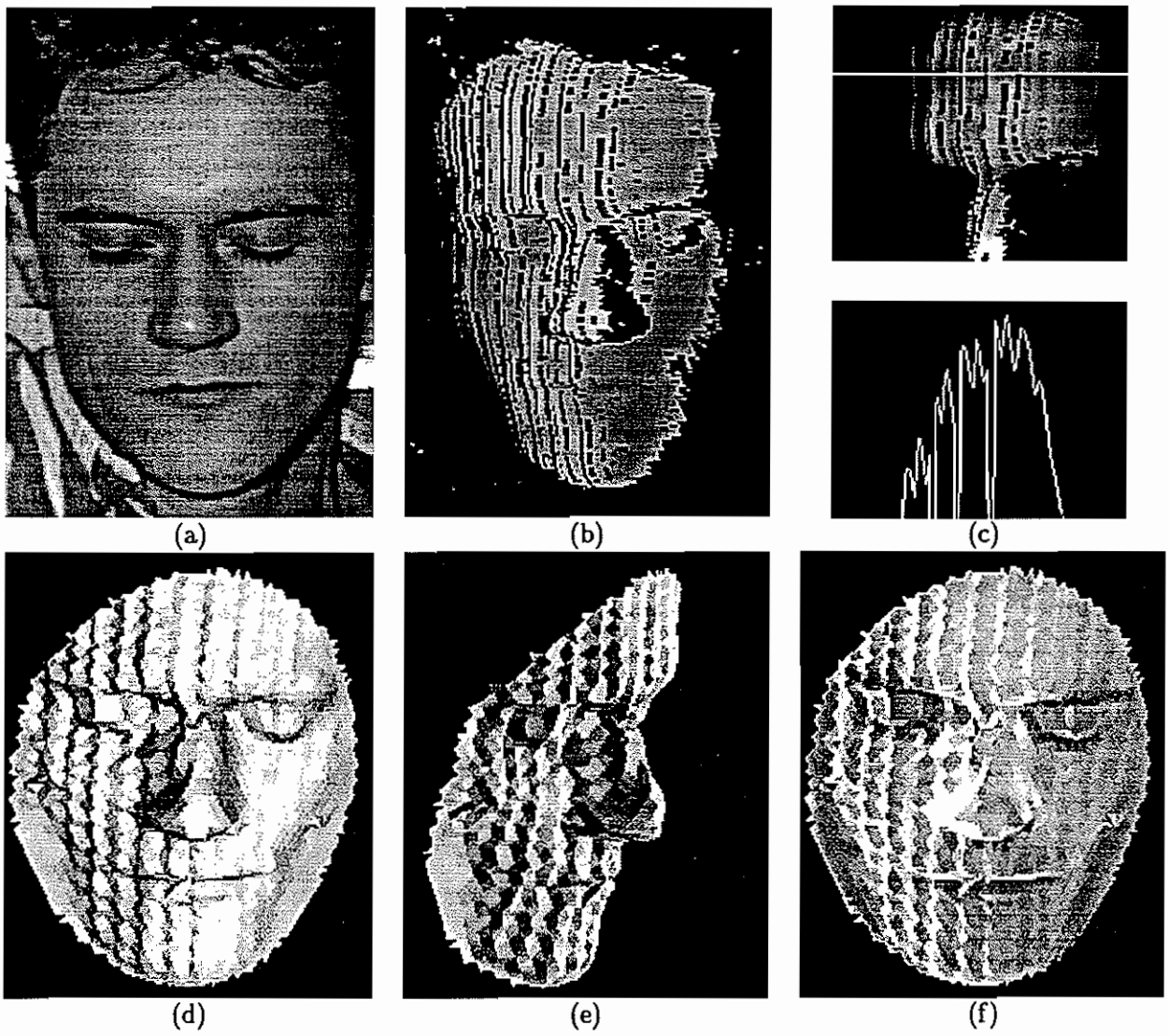

Figure 7: Fitting a surface to range data: (a) Image of a face (courtesy of ETH Zurich). (b) Corresponding range image computed using structured light. (c) A window of the range image in which gray levels have been stretched to emphasize the vertical wrinkles and the histogram of a horizontal slice. (d) (e) Shaded views of the surface reconstructed by using the range-data points as attractors. (f) The corresponding albedo map.

We now turn to sparse 3–D data. In Figure 9, we show a stereo pair of a rock outcrop forming an almost vertical cliff. Note that, even though the geometry is almost epipolar, these two images are very hard to fuse both for humans and for automated procedures. In Figure 9(c), we show the output of a correlation result [Fua, 1993] that gives no information about the shape of the outcrop.

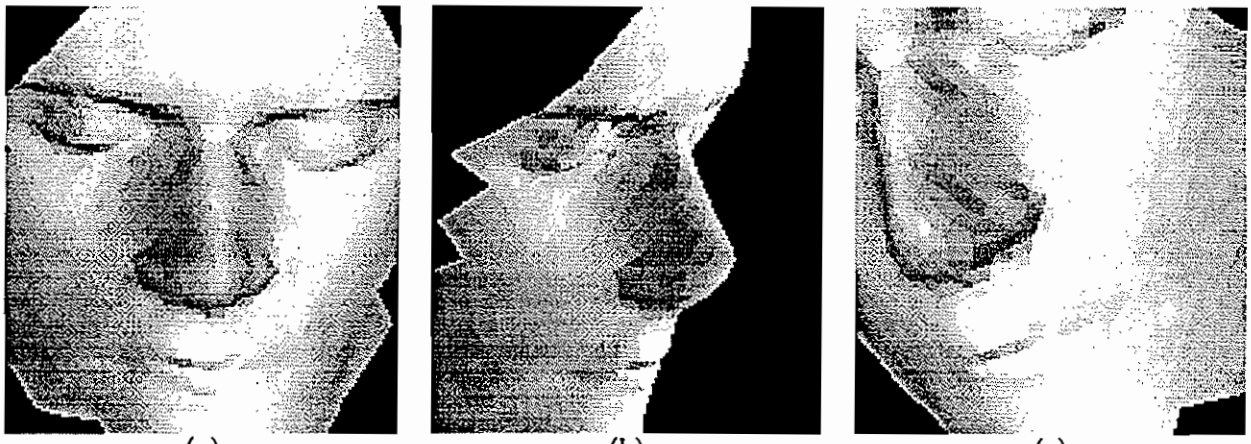

Figure 8: Combining range-data with shape-from-shading information: (a)(b)(c) Shaded views of the refined reconstruction of the face of Figure 7 using shading.

This can be attributed to the fact that, in the cliff area, the fundamental assumption underlying correlation-based stereo using a fixed-shape window is violated: the depth is not constant within a correlation window. To demonstrate the data-fusion capabilities of our approach, we supply the 3–D edges whose projections are shown in Figure 9(d) and (e). To do so, we have used the 3–D snakes [Fua and Leclerc, 1990] that are embedded in the SRI Cartographic Modeling Environment (CME) [Quam and Strat, 1991]: rough contours are hand-entered and treated as the projections of polygonal 3–D curves whose x,y, and z coordinates are then optimized to maximize the average edge strength along the projections. Alternatively, we could take advantage of the output of 3–D edge detectors such as those described in [Ayache and Lustman, 1987, Robert and Faugeras, 1991, Ma and Thonnat, 1992, Meygret *et al.*, 1990].

By using the energy term of Equation 9, we attract an initially flat surface to both the stereo data and the 3–D outlines and produce a shape estimate that is roughly correct but much too smooth, as can be seen in Figure 10(b) and (c).

By adding either the stereo term alone to $\mathcal{E}_T$, Figure 10(d), or both the stereo and shading terms, Figures 10(e) and (f), we can generate a much more realistic model of the surface. Note that in Figure 10(e) the cracks in the right side of the outcrop are well modeled. Our object-centered representation has no trouble accommodating the sharply slanted surfaces.

In Figure 11, we show another application of our technique in a semiurban environment using images of a model board. We have used the 3–D snakes to outline some of the linear features visible in the images. We then generate the rough estimate of the surface shape of Figure 12(b), and improve it using stereo as shown in Figure 12(c). In addition, we have used CME to model the buildings as extruded objects. We exploit them to mask out occluded areas when computing the stereo energy. This is achieved naturally in our system by using the projections of the building models in each view to zero out the corresponding Facet-ID image. In this way, the facet samples

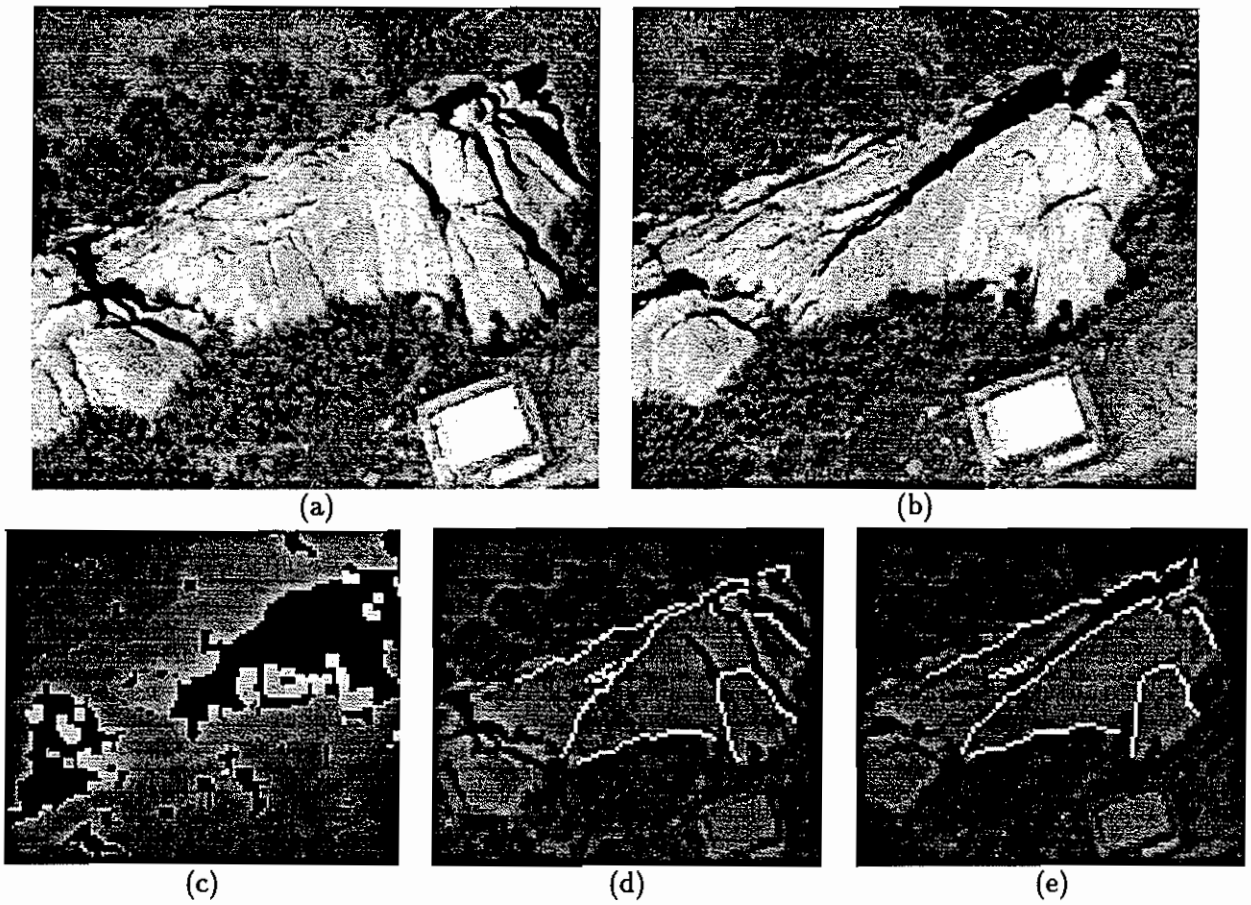

Figure 9: Semiautomated cartography of a rugged site: (a) (b) A hard-to-fuse stereo pair of a rock outcrop with an almost vertical cliff. (c) Disparity map. Within the outcrop the correlation-based algorithm provides almost no information; outside of it the terrain is almost flat. (d) (e) The projections of a few 3-D features outlined using 3-D snakes.

that project at these locations are discounted during the computation of the stereo energy defined in Section 2.4. Since buildings cannot be very well described by our smooth mesh, ignoring those pixels amounts to assuming that the terrain is smooth below the buildings and prevents the surface from wrinkling unduly.

3.2 Refining Previously Derived Models

So far, we have shown how our technique can be used to generate surface models "from scratch." However, very few vision algorithms—ours being no exception—consistently provide a perfect answer across scenes using a predetermined set of information sources and analysis parameters. For applications such as cartography or 3–D graphics, it is often important to be able to easily refine a

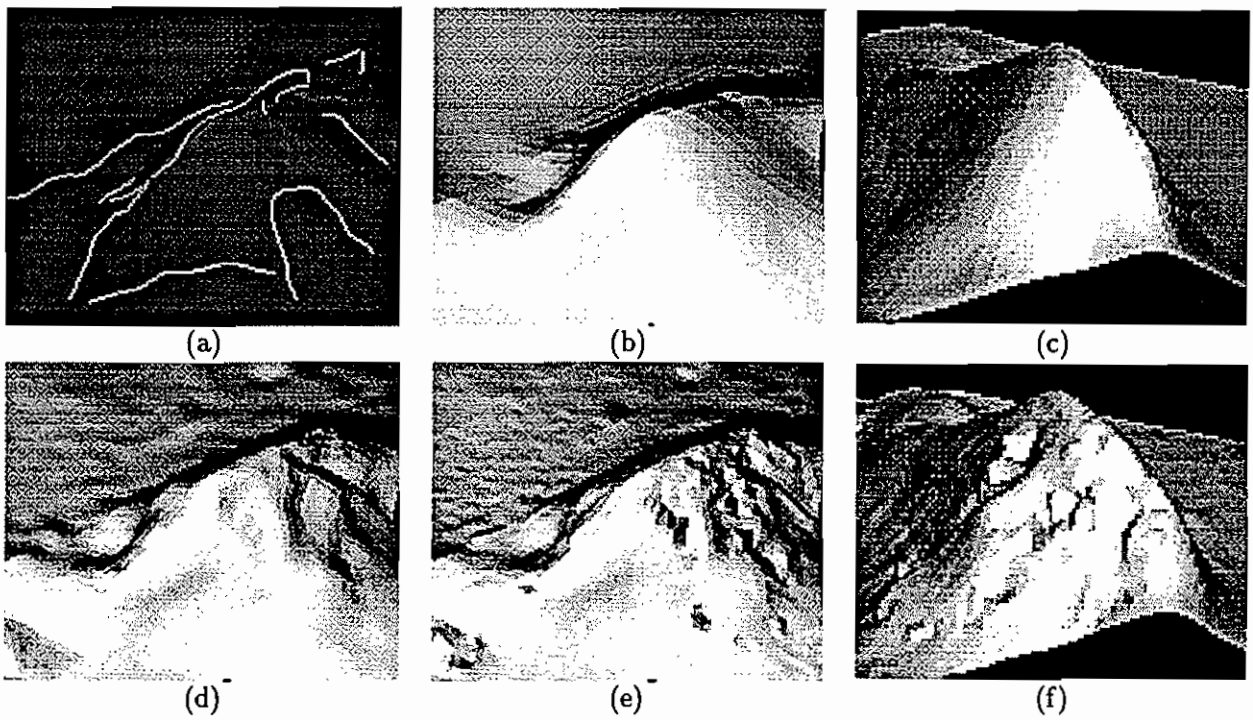

Figure 10: Combining 3–D constraints with stereo and shape-from-shading: (a) The recovery of the terrain for the aerial scene of Figure 9 starts with a flat surface that is attracted by the 3–D outlines and the 3–D cloud of points corresponding to the disparity map. (b) (c) Shaded views of the reconstructed surface using only those constraints. (d) Refinement using stereo. (e) (f) Refinement using both stereo and shape-from-shading.

previously derived result, such as an old DEM or the output of a fully automated procedure, using additional clues. This can be done using both 3–D contours and silhouettes.

We start with an example involving the two aerial images of Figure 13, at the top of which is a very sharp cliff that casts shadows on the ground. Starting from a coarse and inaccurate DEM, we generate the surface shown in Figure 13(e), using stereo alone. By computing the disparities associated with that improved model, we have visually checked that it is correct except in the immediate vicinity of the cliff, where it is too smooth. This should be expected: our objective function $\mathcal{E}_T$ includes a smoothness term, and the face of the cliff is not visible in those images and therefore provides no stereo clues. By sketching the edge of the cliff and the shadows with our 3–D snakes and using them to add an attraction term to the objective function, we can deform the surface slightly to produce the result shown in Figure 13(f) where the ridge is better defined. To further check the validity of our result, we have used the known sun direction to predict which parts of the ground are in shadow. To do this we generate a sun view, that is an orthographic view as seen from the sun's viewpoint, and the corresponding Facet-ID image. For every facet,

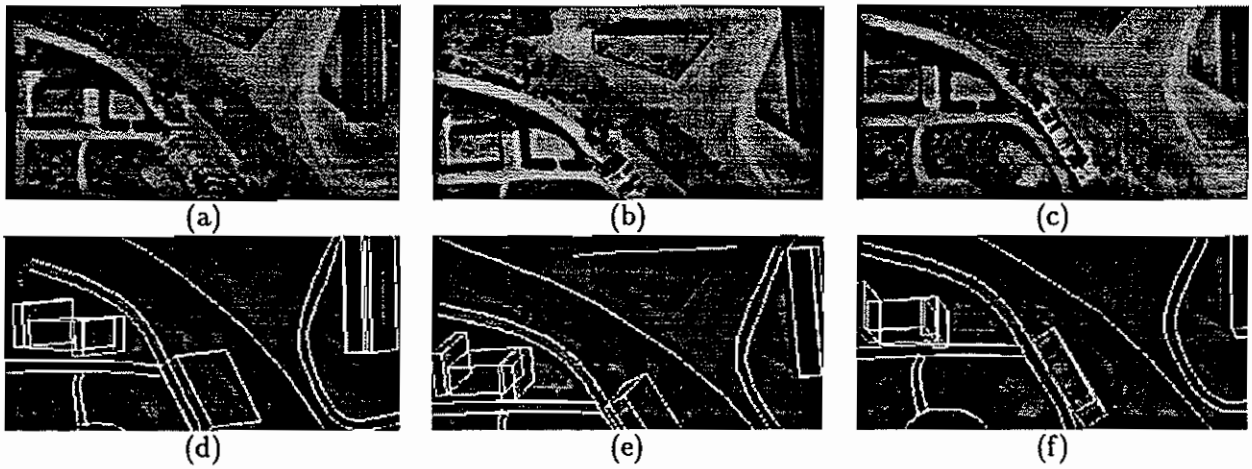

Figure 11: Semiautomated cartography of a semiurban site: (a) (b) (c) Three images taken with different light source directions. (d) (e) (f) Projections of hand-entered 3–D linear features and building blocks. Note that the bases of the buildings extend below the ground.

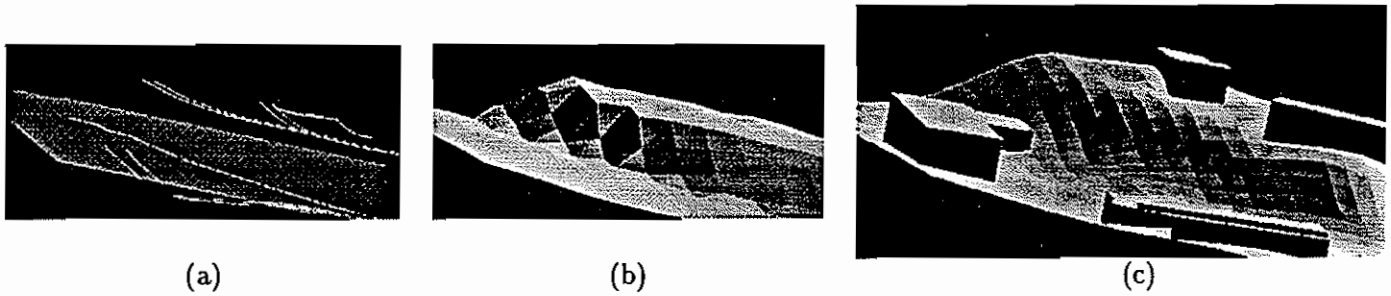

Figure 12: Combining 3–D constraints and visibility constraints with stereo: (a) The 3–D linear features of Figure 11 above the flat plane used as the initial surface estimate. (b) A rough estimate of the ground-level surface (c) Surface after optimization using both stereo and hand-entered buildings to mask occluded areas.

we compute the proportion of samples that are visible in this sun view as shown in Figure 13(g). The facets for which a large proportion of samples is occluded are those in shadow. As can be seen, these shadowed facets match the actual shadows fairly well, which leads us to believe that our reconstruction is accurate.

Silhouettes are also very good indicators of the quality of a reconstruction. For example, the reconstruction of the bottom of the nose in Figure 8 is not quite right as evidenced by its silhouette in the side view of the same man shown in Figure 14. However, we can use the silhouette constraints of Section 2.3.2 with the two silhouettes shown in the figure. The silhouettes are 2–D curves that

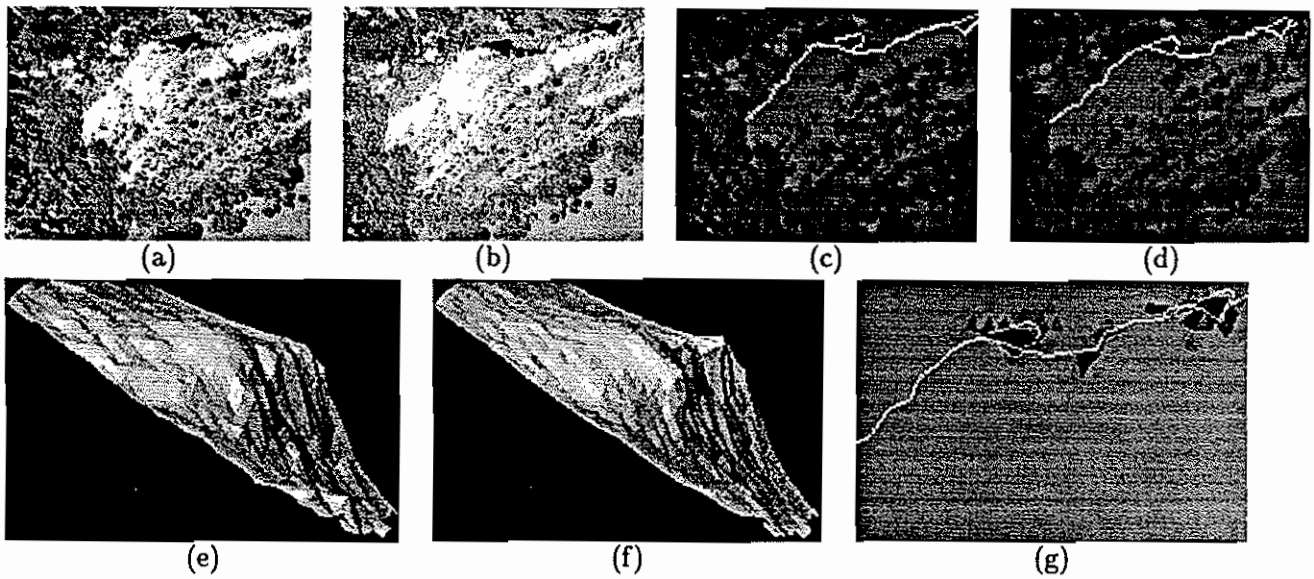

Figure 13: Improving and checking a DEM: (a) (b) An aerial stereo pair of a cliff with clearly visible shadows. (c) (d) The cliff's ridge and cast shadows outlined using 3-D snakes. (e) Reconstructed surface using stereo alone. (f) Reconstructed surface using both stereo and the 3–D outlines as attractors. (g) Predicted shadow areas in black. The prediction was carried out using the reconstruction shown in (f) and the known sun direction. Note that these hypothesized shadows closely match the actual ones. Note also that, were we to use the original reconstruction shown in (e) to perform this computation, no shadows would be predicted because the surface is too smooth.

have been outlined using 2–D snakes. In the manner of Section 3.1.1, we take the total energy $\mathcal{E}_T$ to be

$$\begin{aligned} \mathcal{E}_T &= \lambda_D \mathcal{E}_D + \lambda_S \mathcal{E}_S + \lambda_{Sh} \mathcal{E}_{Sh} \\ \mathcal{E}_S &= \sum_s e_s \end{aligned}$$

where the e_s are the silhouette attraction terms of Equation 8 and $\mathcal{E}_{Sh}$ the shading term described in Section 2.4. We use these terms to deform the nose region and generate the improved result shown in Figure 14(c).

The face reconstruction of Figure 6 presents us with a slightly different problem. We have used a correlation-based stereo algorithm to provide us with an initial estimate. This algorithm gave us no information on the sharply slanted parts of the face, which are therefore missing from the reconstruction. The silhouettes of the face, however, are clearly visible in Figure 15 and easy to outline. To take advantage of these, we again use a coarse-to-fine strategy. We start with a larger and coarser mesh that evolves under the influence of the silhouettes and the vertices of the original

(a) (b) (c)

Figure 14: Using silhouettes to improve a reconstruction: (a) The face of Figure 7 with a silhouette at the bottom of the nose outlined. (b) A side view of the same face with a second nose silhouette. (c) Shaded views of the refined reconstruction using both shading and the two silhouettes.

reconstruction that are treated as attractors. When the mesh has been refined and optimized, we complete the optimization procedure by turning on the full objective function:

$$\mathcal{E}_T = \lambda_D \mathcal{E}_D + \lambda_S \mathcal{E}_S + \lambda_{Sh} \mathcal{E}_{Sh} + \lambda_{St} \mathcal{E}_{St},$$

where $\mathcal{E}_{Sh}$ and $\mathcal{E}_{St}$ denote the shading and stereo terms presented in Section 2.4. The results are shown in Figure 15(c),(d) and (e).

The silhouettes used in the two examples above have been entered semiautomatically. But here again, we could take advantage of automatically extracted ones [Cipolla and Blake, 1990, Liedtke *et al.*, 1991, Vaillant and Faugeras, 1992].

4 Conclusion

We have presented a surface reconstruction method that uses an object-centered representation to recover 3–D surfaces. Our method uses both monocular shading cues and stereoscopic cues from any number of images while correctly handling self-occlusions. It can also take advantage of the geometric constraints derived from measured 3–D points and 2–D silhouettes. These complementary sources of information are combined in a unified manner so that new ones can be added easily as they become available.

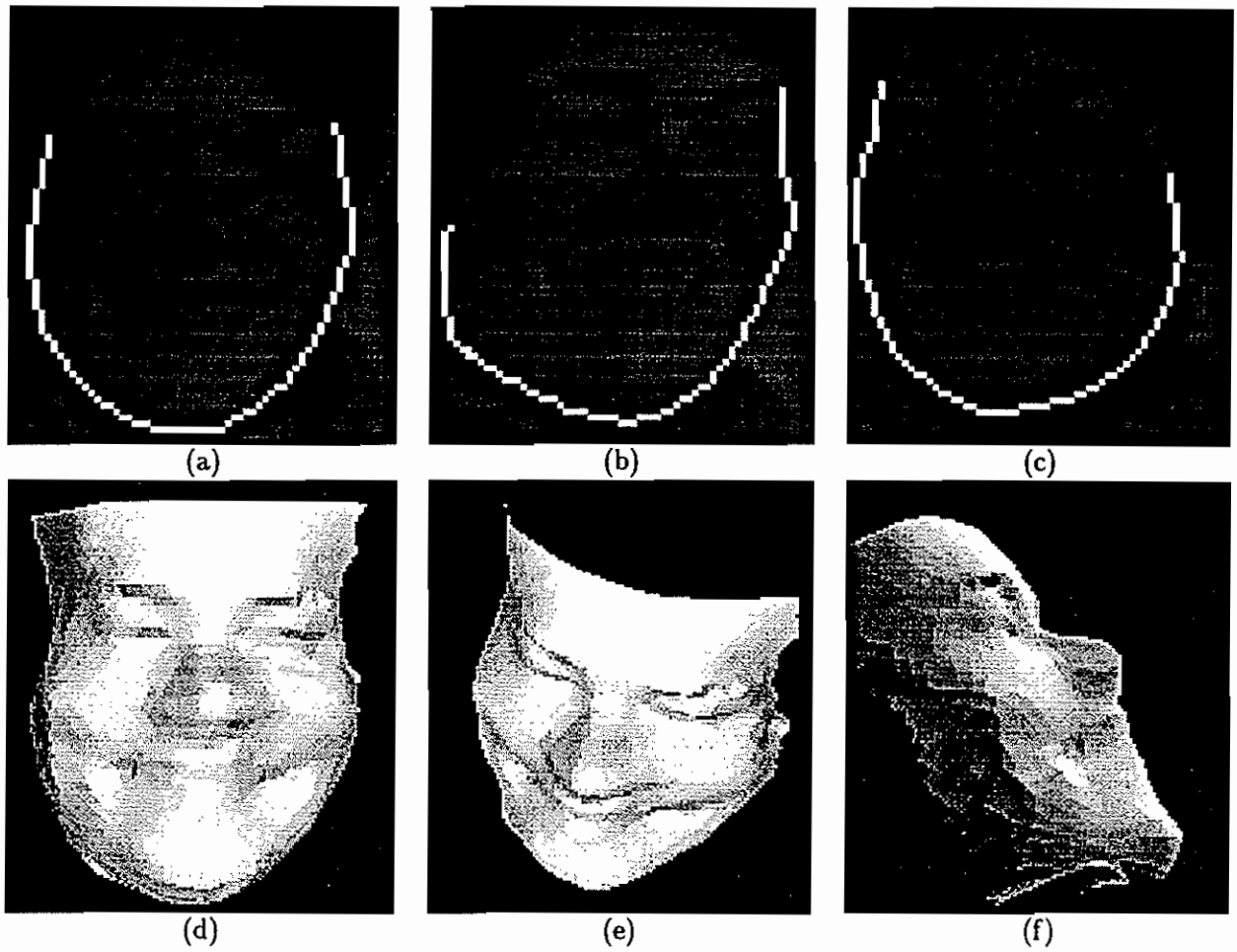

Figure 15: Using silhouettes to expand the scope of our method: (a) (b) (c) Silhouettes of the face in the three views of Figure 6 outlined using 2–D snakes. (d) (e) (f) Shaded views of reconstructed surface after optimization using stereo, shading, and the constraints provided by the silhouettes.

Using a variety of real imagery, we have demonstrated that the resulting method is quite powerful and flexible, allowing for both completely automatic reconstruction in straightforward circumstances, and for user-assisted reconstruction in more complex circumstances. User assistance is provided primarily through the introduction and identification of a small number of hand-entered linear and point features using semi-automated "snake" technology. The method is also controlled by a small number of parameters that specify the relative importance of the various information sources. These parameters typically do not need to be adjusted for images within a given class (such as face images or high-altitude aerial images), but only across classes.

The method has valuable capabilities for applications such as 3–D graphics model generation and high-resolution cartography in which a human can select the sources of information to be used and their relative importance. For example, in the case of mapping, one must ensure that the terrain model conforms to the feature data and does not violate any physical constraints: roads should be on the ground and not overly tilted, streams should stay within stream beds, buildings should not be floating in space, and so on. Our method allows one to both satisfy these constraints and account as well as possible for the observed image data.

In future work, we will study in a more quantitative manner the influence of the various terms of our objective function and their relative weights. This will require the use of ground-truth and carefully controlled conditions. We plan to set up a facility that will allow us to acquire the necessary data. We will also strive to replace some of the hand-entered geometric cues by automatically extracted ones and to investigate more complex topologies than the ones shown here. A principled way to do so would be to rephrase our modeling task as one of finding the "best" description of a scene in terms of the Minimum Description Length (MDL) principle [Rissanen, 1987, Leclerc, 1989a, Fua and Hanson, 1991]. It can be shown that the objective function that we propose here can be reformulated in terms of the MDL principle. After optimization using stereo and shape from shading, the surface ought to provide the best possible compromise between simplicity of description of the surface and fit to the image data in terms of the simple vocabulary of triangulated meshes. The extensions that we have described above allow us to enrich the vocabulary by adding new primitives—ridges, building, roads, and so on—that allow an even more effective description. This approach would give us a principled way to accept or reject new objects in our overall representation.

Acknowledgments

Support for this research was provided by various contracts from the Advanced Research Projects Agency. We wish to thank Hervé Matthieu, Olivier Monga, Olaf Kubler and Marjan Trobina from INRIA and ETH Zurich who have provided us with the face images and corresponding calibration data that appear in this paper and have proved extremely valuable to our research effort. We also wish to thank Amnon Krupnik from Ohio State University for supplying us with the photogrammetric data.

Appendix: Robustness of the constraint weighting scheme

In Section 2.2, we proposed a weighting scheme for the—in general noncommensurate—components of the objective function of Equation 1. In this appendix, we use a specific example to illustrate the ability of our method to combine stereo constraints and externally supplied 3–D and 2–D geometric constraints in the presence of noise and the relative insensitivity of our procedure to parameter settings.

Figure A.1: Three synthetic images generated by texture mapping the image of a face onto the hemispheric surface shown in Figure A.2 as seen from three different viewpoints.

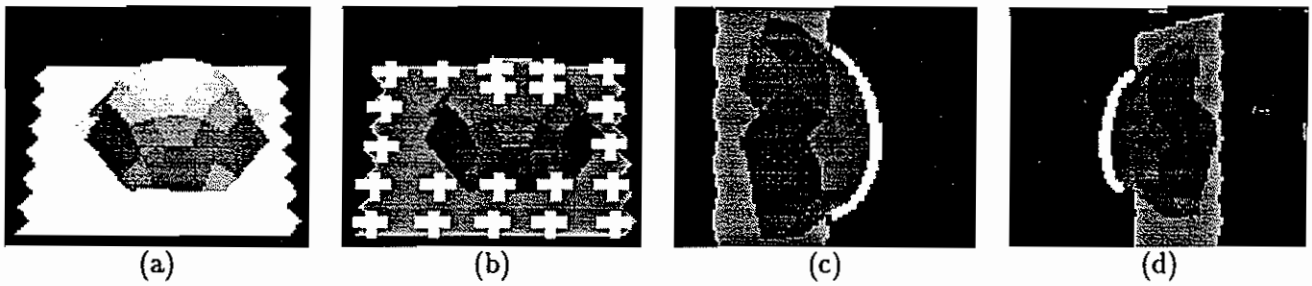

Figure A.2: (a) Hemispheric surface used to generate the images of Figure A.1 and taken to be the "ground truth" for the experiments described in this appendix (b) The 3–D geometric constraints are in the form of 25 regularly spaced 3–D points lying on the hemisphere, shown as white crosses, some of which are occluded. (c,d) The 2–D geometric constraints are in the form of two silhouette edges shown as thick white lines.

The images used here are shown in Figure A.1. They have been generated by texture mapping the image of a face onto the hemispheric surface of Figure A.2(a) as seen from three different viewpoints. We take the 3–D geometric constraints to be given by a set of 25 regularly spaced 3–D attractors lying on the hemisphere and shown in Figure A.2(b). The 2–D constraints are given by the two occluding contours of Figures A.2(c) and (d). We can therefore write the total objective

function of Equation 1 as

$$\begin{aligned} \mathcal{E}_T(\mathcal{S}) &= \lambda_D \mathcal{E}_D(\mathcal{S}) + \mathcal{E}(\mathcal{S}) \\ \mathcal{E}_(\mathcal{S}) &= \lambda_{St} \mathcal{E}_{St}(\mathcal{S}) + \lambda_A \mathcal{E}_A(\mathcal{S}) + \lambda_S \mathcal{E}_S(\mathcal{S}) \ . \end{aligned}$$

where $\mathcal{E}_{St}$ is the stereo term, $\mathcal{E}_A$ the sum of the 3–D attraction terms of Equation 7, and $\mathcal{E}_S$ the sum of the silhouette attraction terms of Equation 8. At the start of each optimization step, the λ_i coefficients are recomputed according to Equation 5 using a set of user-supplier λ'_i that specify the relative importance of the various terms.

Here we study the influence of the user-supplied weights, λ'_{St}, λ'_A and λ'_S, on the distance between the surface reconstructed by minimizing $\mathcal{E}_T$ and the "ground truth" surface of Figure A.2(a).

For each setting of the parameters, in Figures A.3, A.4, and A.5, we plot four curves corresponding to four different amounts of Gaussian white noise—of respective variance 2.5%, 5.0%, 7.5% and 10% of the images'dynamic range—added to the images to degrade the stereo term. The curves were obtained by averaging the results of several trials, all starting from a randomized flat surface and utilizing our continuation method with five increasing values of

$$S_\lambda = \lambda'_{St} + \lambda'_A + \lambda'_S,$$

the sum of the user-supplied weights, ranging from 0.5 to 0.9. As in Figure 4, the graphs represent the RMS reconstruction error at the end of each optimization step as a function of S_λ. In this set of experiments, we allowed only the z coordinates of the vertices to vary. We also fixed the boundary vertices so as to eliminate the effect of the gray-level discontinuities at the border between the texture-mapped part of the images and their black background.

The error is measured by the difference in elevation between the reconstructed vertices and the elevation they would have if they were on the actual "ground truth" surface of Figure A.2(a). Note that the difference in elevation between the top and the bottom of the hemisphere is 34 units of elevation and that an error of 1 unit of elevation corresponds to a difference in computed disparities of approximately 0.25 pixel for projections from the image of Figure A.1(a) into those of Figure A.1(b) and (c).

In Figure A.3, we show the behavior of our continuation method using stereo alone, that is taking λ'_A and λ'_S to be zero. In Figure A.3(a), we draw as solid lines the four curves derived using all three noisy images at the same time and in Figure A.3(b,c) those obtained using only two images at a time. For comparison's sake, we also plot as dashed lines the curves computed using noise-free images. As the abscissa is traversed rightwards, $S_\lambda = \lambda'_{St}$ increases and λ'_D decreases, resulting in curves having the same shape as that of Figure 4. Note, however, that for the higher noise values, the best result is not achieved for the largest value of S_λ but for one slightly smaller. As discussed in Section 2, in the presence of noise, smoothing is required to prevent the surface from overfitting the data.

In Figure A.4, we plot the equivalent curves for different values of λ'_A and λ'_S. Figures A.4(a) and (b) illustrate the use of the 3–D point constraints along with three-image stereo. Graph (a) was generated using $\lambda'_A = 0.4 S_\lambda$, $\lambda'_{St} = 0.6 S_\lambda$ and graph (b) using $\lambda'_A = 0.2 S_\lambda$, $\lambda'_{St} = 0.8 S_\lambda$. In

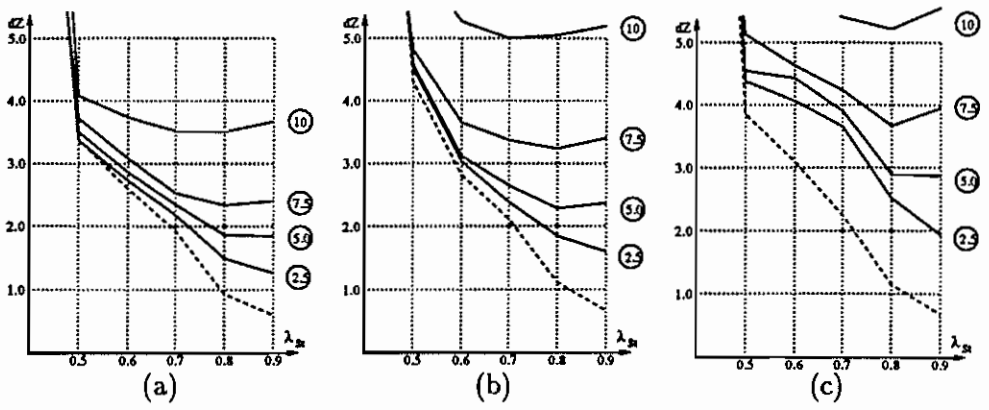

Figure A.3: Continuation method using stereo alone: plot of elevation errors as a function of the regularization parameters λ'_D, using all three images simultaneously (a), using only images 1 and 2 (b), and using only images 1 and 3 (c). The dashed curve corresponds to noise-free images and the four solid curves to increasing amounts of white noise being added to the images. Using all three images yields substantially better results than any of the pairs.

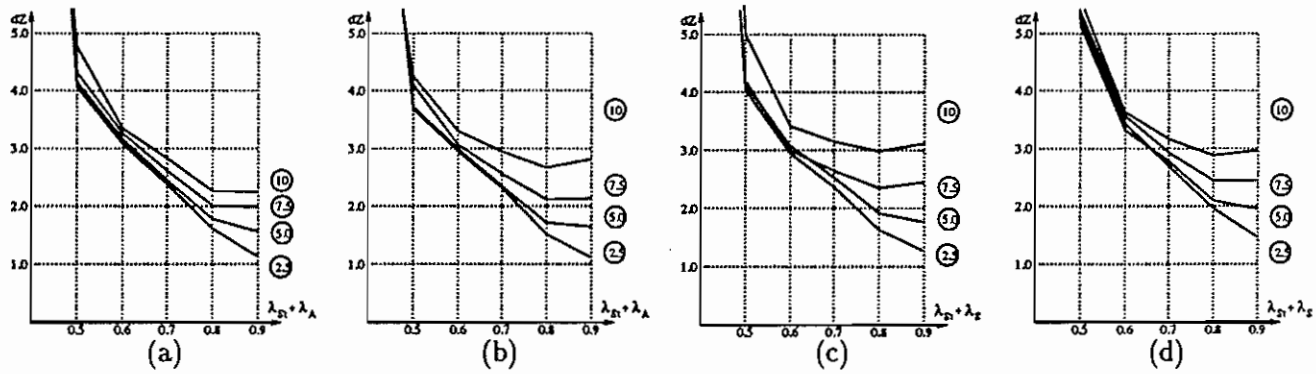

Figure A.4: Combining noise-free constraints with three-image stereo: (a) Using heavily weighted 3–D point constraints. (b) Using less heavily weighted 3–D point constraints (c) Using heavily weighted 2–D silhouette constraints. (d) Using less heavily weighted 2–D silhouette constraints.

other words, the geometric constraint is weighted more heavily in the first case than in the second. As expected, in the absence of noise the results are indistinguishable from those of Figure A.3(a). However, in the presence of noise, the constraints consistently improve the outcome. Since the 3–D points lie exactly on the constraint surface, the improvement is larger when the 3–D constraint is weighted more heavily. The same effect can be observed by using the 2–D silhouette constraints with $\lambda'_S = 0.4S_\lambda$, $\lambda'_{St} = 0.6S_\lambda$, graph (c), and $\lambda'_S = 0.2S_\lambda$, $\lambda'_{St} = 0.8S_\lambda$, graph (d). The average

improvement is not as large because the silhouette constraints are more localized but the observed trends are similar.

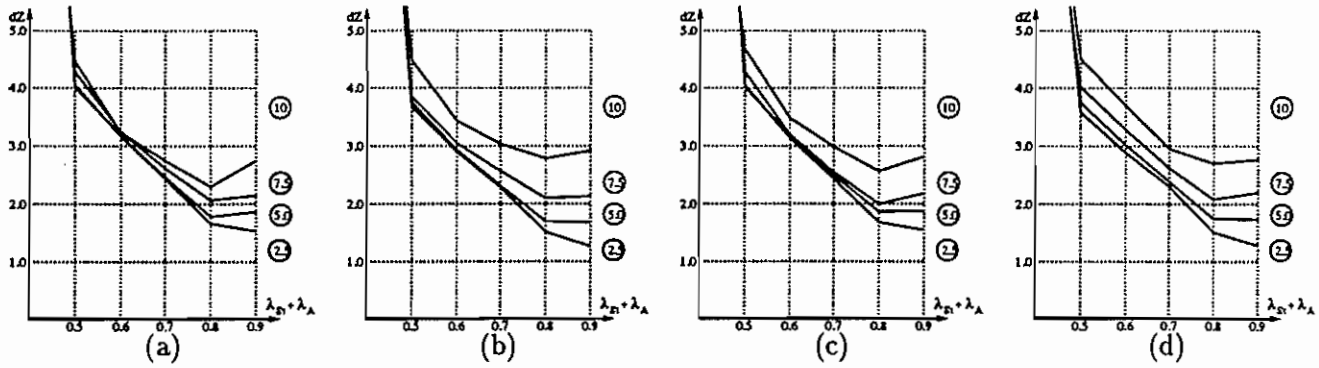

Figure A.5: Noisy 3–D constraints using the same parameters as in Figure A.4: (a) (b) The elevation of the attactors has been randomized by adding noise of variance 1. (c) (d) The elevation of the attactors has been randomized by adding noise of variance 2.

Note, however, that the constraints used above were "perfect" in the sense that the 3–D points lie exactly on the "ground truth" surface. This is not realistic in general as there always will be some imprecision. In Figure A.5, we show the result of rerunning the same experiments as before, after having randomized the elevation of the 3–D attractors. Since the precision of the constraints has now degraded, their use yields an improvement over stereo alone only when enough noise has been added to the images so that the reliability of the stereo term is less than that expected of the constraints, and this independently of the exact weights chosen.

We have shown that, on a specific example, our method for combining the constraints is robust in the presence of noise. The exact numbers we obtain may change slightly but the overall behavior of the optimization procedure is fairly constant for different settings of the user-supplied weights and yields intuitively satisfactory results.

Because of the extreme complexity of the image potentials, a full mathematical treatment of the behavior of the objective function is beyond the scope of this paper. However, in practice, we have observed the same relative invariance of the results with respect to changes of parameter settings.

References

[Ayache and Lustman, 1987] N. Ayache and F. Lustman. Fast and reliable passive trinocular stereovision. In *International Conference on Computer Vision*, June 1987.

[Blake *et al.*, 1985] A. Blake, A. Zisserman, and G. Knowles. Surface descriptions from stereo and shading. *Image Vision Computation*, 3(4):183–191, 1985.

[Cipolla and Blake, 1990] R. Cipolla and A. Blake. The dynamic analysis of apparent contours. In *International Conference on Computer Vision*, 1990.

[Cohen *et al.*, 1991] I. Cohen, L. D. Cohen, and N. Ayache. Introducing new deformable surfaces to segment 3D images. In *Conference on Computer Vision and Pattern Recognition*, pages 738–739, 1991.

[Cryer *et al.*, 1992] J. E. Cryer, Ping-Sing Tsai, and Mubarak Shah. Combining shape from shading and stereo using human vision model. Technical Report CS-TR-92-25, U. Central Florida, 1992.

[Delingette *et al.*, 1991] H. Delingette, M. Hebert, and K. Ikeuchi. Shape representation and image segmentation using deformable surfaces. In *Conference on Computer Vision and Pattern Recognition*, pages 467–472, 1991.

[Ferrie *et al.*, 1992] Frank P. Ferrie, Jean Lagarde, and Peter Whaite. Recovery of volumetric object descriptions from laser rangefinder images. In *European Conference on Computer Vision*, Genoa, Italy, April 1992.

[Fua and Hanson, 1991] P. Fua and A.J. Hanson. An optimization framework for feature extraction. *Machine Vision and Applications*, 4(2):59–87, Spring 1991.

[Fua and Leclerc, 1990] P. Fua and Y. G. Leclerc. Model driven edge detection. *Machine Vision and Applications*, 3:45–56, 1990.

[Fua and Leclerc, 1994] P. Fua and Y. G. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal of Computer Vision*, 1994. Accepted for publication, available as Tech Note 535, Artificial Intelligence Center, SRI International.

[Fua and Sander, 1992] P. Fua and P. Sander. Segmenting unstructured 3d points into surfaces. In *European Conference on Computer Vision*, Genoa, Italy, April 1992.

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

[Güelch, 1988] E. Güelch. Results of test on image matching of ISPRS WG III / 4. *International Archives of Photogrammetry and Remote Sensing*, 27(III):254–271, 1988.

[Heipke, 1992] C. Heipke. Integration of digital image matching and multi image shape from shading. In *International Society for Photogrammetry and Remote Sensing*, pages 832–841, Washington D.C., 1992.

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

[Koh *et al.*, 1994] E. Koh, D. Metaxas, and N. Badler. Hierarchical shape representation using locally adaptative finite elements. In *European Conference on Computer Vision*, Stockholm, Sweden, May 1994.

[Leclerc, 1989a] Y. G. Leclerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3(1):73–102, 1989.

[Leclerc, 1989b] Y. G. Leclerc. *The Local Structure of Image Intensity Discontinuities.* PhD thesis, McGill University, Montréal, Québec, Canada, May 1989.

[Liedtke *et al.*, 1991] C. E. Liedtke, H. Busch, and R. Koch. Shape adaptation for modelling of 3D objects in natural scenes. In *Conference on Computer Vision and Pattern Recognition*, pages 704–705, 1991.

[Lowe, 1991] D. G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(441-450), 1991.

[Luenberger, 1984] D. G. Luenberger. *Linear and Nonlinear Programming.* Addison-Wesley, Menlo Park, California, second edition, 1984.

[Ma and Thonnat, 1992] R. Ma and M. Thonnat. A robust and efficient contour-based stereo matching algorithm. Research report (in preparation), INRIA, 1992.

[McInerney and Terzopoulos, 1993] T. McInerney and D. Terzopoulos. A finite element model for 3d shape reconstruction and nonrigid motion tracking. In *International Conference on Computer Vision*, pages 518–523, Berlin, Germany, 1993.

[Meygret *et al.*, 1990] A. Meygret, M. Thonnat, and M. Berthod. A pyramidal stereovision algorithm based on contour chain points. In *European Conference on Computer Vision*, pages 83–88, Antibes, France, April 1990.

[Pentland and Sclaroff, 1991] A. Pentland and S. Sclaroff. Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:715–729, 1991.

[Pentland, 1990] A. Pentland. Automatic extraction of deformable part models. *International Journal of Computer Vision*, 4(2):107–126, March 1990.

[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.

[Quam and Strat, 1991] L. Quam and T.M. Strat. SRI image understanding research in cartographic feature extraction. In *International Society for Photogrammetry and Remote Sensing*, Munich, Germany, September 1991. Also available as Tech Note 505, Artificial Intelligence Center, SRI International.

[Rissanen, 1987] J. Rissanen. Minimum-description-length principle. *Encyclopedia of Statistical Sciences*, 5:523–527, 1987.

[Robert and Faugeras, 1991] L. Robert and O.D. Faugeras. Curve-Based Stereo: Figural Continuity and Curvature. In *Conference on Computer Vision and Pattern Recognition*, Maui, Hawaii, June 1991.

[Robert *et al.*, 1992] L. Robert, R. Deriche, and O.D. Faugeras. Dense depth recovery from stereo images. In *European Conference on Artificial Intelligence*, pages 821–823, Vienna, Austria, August 1992.

[Stokely and Wu, 1992] E. M. Stokely and S. Y. Wu. Surface parameterization and curvature measurement of arbitrary 3-d objects: five practical methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):833–839, August 1992.

[Szeliski and Tonnesen, 1992] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. In *Computer Graphics (SIGGRAPH'92)*, pages 185–194, July 1992.

[Terzopoulos and Metaxas, 1991] D. Terzopoulos and D. Metaxas. Dynamic 3D models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(703-714), 1991.

[Terzopoulos and Vasilescu, 1991] D. Terzopoulos and M. Vasilescu. Sampling and reconstruction with adaptive meshes. In *Conference on Computer Vision and Pattern Recognition*, pages 70–75, 1991.

[Terzopoulos, 1986] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:413–424, 1986.

[Vaillant and Faugeras, 1992] R. Vaillant and O.D. Faugeras. Using Occluding Contours for 3D Object Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1992.

[Vemuri and Malladi, 1991] B. C. Vemuri and R. Malladi. Deformable models: Canonical parameters for surface representation and multiple view integration. In *Conference on Computer Vision and Pattern Recognition*, pages 724–725, 1991.