Object-Centered Surface Reconstruction: Combining Multi-Image Stereo and Shading

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

Abstract

Our goal is to reconstruct both the shape and reflectance properties of surfaces from multiple images. We argue that an object-centered representation is most appropriate for this purpose because it naturally accommodates multiple sources of data, multiple images (including motion sequences of a rigid object), and self-occlusions. We then present a specific object-centered reconstruction method and its implementation. The method begins with an initial estimate of surface shape provided, for example, by triangulating the result of conventional stereo. The surface shape and reflectance properties are then iteratively adjusted to minimize an objective function that combines information from multiple input images. The objective function is a weighted sum of stereo, shading, and smoothness components, where the weight varies over the surface. For example, the stereo component is weighted more strongly where the surface projects onto highly textured areas in the images, and less strongly otherwise. Thus, each component has its greatest influence where its accuracy is likely to be greatest. Experimental results on both synthetic and real images are presented.

1 Introduction

The problem of recovering the shape and reflectance properties of a surface from multiple images has received considerable attention (Barrow and Tenenbaum 1978, Grimson and Huttenlocher 1992, Marr 1982, Okutomi and Kanade 1991, Terzopoulos 1988). This is a key problem not only in developing general-purpose vision systems, but also in specialized areas such as the generation of Digital Elevation Models from aerial images (Barnard 1989, Diehl and Heipke 1992, Hannah 1989, Kaiser *et al.* 1992, Wrobel 1991).

In this paper, we view the ultimate goal of a surface reconstruction method as finding an object-centered description of a surface from a set of input images that is sufficiently complete, in terms of its geometric and radiometric properties, that it is possible to generate an image of the surface from any viewpoint. In particular, the description should be sufficiently complete to reproduce the input images to within a certain tolerance, given models of the cameras, their relative locations, and expected noise.

Our surface reconstruction method uses an object-centered representation, specifically a triangulated 3–D mesh of vertices. Such a representation accommodates both geometric and radiometric information, as well as multiple images (including motion sequences of a rigid object) and self-occlusions. We have chosen to model the surface material using the Lambertian reflectance model with variable albedo. Consequently, the natural choice for the monocular information source is shading, while intensity is the natural choice for the image feature used in multi-image correspondence. Not only are these the natural choices given a Lambertian reflectance model, they are also complementary (Blake *et al.* 1985, Leclerc and Bobick 1991): intensity correlation is most accurate wherever the input images are highly textured, whereas shading is most accurate where the input images are untextured.

The reconstruction method is to minimize an objective function whose components depend on the input images and some measure of the complexity of the 3–D mesh. The method starts with an initial estimate for the mesh derived, for example, from the triangulation of conventional stereo results, and uses conjugate gradient descent to minimize the objective function. The image-dependent components of the objective function are related to the two sources of information mentioned above. We take advantage of the complementary nature of the information sources by weighting the components at each facet of the triangulated mesh according to the degree of texturing within the areas of the images that the facet projects to. The projection uses a hidden-surface algorithm to take occlusions into account.

In the following section, we describe related work and our contributions in this area. Following this we discuss some of the key issues in multi-image surface reconstruction and how to combine different sources of information for such purposes. We then describe in detail our specific procedure, discuss its behavior on synthetic data, and show some results on real images.

2 Related Work and Contributions

Three-dimensional reconstruction of visible surfaces continues to be an important goal of the computer vision research community. Initially, much of the work concentrated on $2\frac{1}{2}$–D image-centered reconstructions, such as Barrow and Tenenbaum's *Intrinsic Images* (Barrow and Tenenbaum 1978) and Marr's $2\frac{1}{2}$*–D Sketch* (Marr 1982). These view-centered surface representations have been the basis for quite successful systems for recovering shape and surface properties. Some have used single sources of information, such as sequences of range

data or intensity images (Asada *et al.* 1992, Hung *et al.* 1991), stereo (Diehl and Heipke 1992, Kaiser *et al.* 1992, Witkin *et al.* 1987, Wrobel 1991), and shading (Hartt and Carlotto 1989, Horn 1990, Terzopoulos 1988). Others have combined sources of information, such as shading and texture (Choe and Kashyap 1991), focus, vergence, stereo, and camera calibration (Abbot and Ahuja 1990). See (Aloimonos 1989) for further discussions on information fusion.

More recently, full 3–D representations have been used, such as 3–D surface meshes (Terzopoulos and Vasilescu 1991, Vemuri and Malladi 1991), 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).

As with the methods employing $2\frac{1}{2}$–D representations, those employing 3–D representations have used a variety of single image cues for reconstruction, such as silhouettes and image features (Cohen *et al.* 1991, Delingette *et al.* 1991, Terzopoulos *et al.* 1987, Tomasi and Kanade 1992, Wang and Wang 1992), range data (Whaite and Ferrie 1991), stereo (Fua and Sander 1992), and motion (Szeliski 1991). Liedtke *et al.* (1991) first uses silhouettes to derive an initial estimate of the surface, and then uses a multi-image stereo algorithm to improve on the result. Both their approach to deriving an initial estimate for the mesh and Szeliski and Tonnesen's approach (1992) are different from ours and this is an important topic for future research.

Of special relevance to this paper is research in combining stereo and shape from shading. Using $2\frac{1}{2}$–D representations, Blake *et al.* (1985) is the earliest reference we are aware of that discusses the complementary nature of stereo and shape from shading, but meaningful experimental results are not provided. Leclerc and Bobick (1991) discuss the integration of stereo and shape from shading, but their implementation uses stereo only as an initial condition to their height–from–shading algorithm. Cryer *et al.* (1992) combine the high–frequency output of a shape from shading algorithm with the low–frequency output of a stereo algorithm using filters designed to match those in the human visual system.

Using full 3–D representations, Heipke (1992) integrates stereo and shading, but assumes that the images can be separated beforehand into zones of variable albedo (where one does stereo) and areas of constant albedo (where one does shape from shading). This is in contrast to our approach described below, in which the optimization procedure dynamically adapts to the image data.

In this paper, we unify the idea of using 3–D meshes to integrate information from multiple images with that of using multiple cues. Our specific approach to this unification has led to a number of important contributions:

- We correctly deal with occlusions by using a hidden surface algorithm during the reconstruction process.

- Our stereo technique avoids the constant depth assumption of traditional correlation-

based stereo algorithms, effectively using self-adjusting variable-sized windows in the images.

- Our approach to shape from shading is applicable to surfaces with slowly varying albedo. This is a significant advance over traditional approaches that require constant albedo.

Finally, we view the specific manner in which the multiple cues are integrated together to be an important contribution in itself. The integration is achieved by using a weighting scheme for combining shape from shading and stereo that depends on the local degree of texturing in the input images. We establish, using both synthetic and real images, that it leads to significantly better results than using either cue alone.

To demonstrate the validity of the overall approach, we have implemented a computationally effective optimization procedure, and have demonstrated that it finds good minima of the objective function on both synthetic and real images.

3 Issues in Multi-Image Surface Reconstruction

We briefly discuss here some of the key issues in multi-image surface reconstructions, and outline how we address the issues here. These outlines will be expanded upon in Section 4.

3.1 Surface Shape and Its Representation

Since the task is to reconstruct 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.

Many object-centered surface representations are possible. However, practical issues are important in choosing an appropriate one. First, the representation should be general-purpose in the sense that it should be possible to represent any continuous surface, closed or open, and of arbitrary genus. Second, it should be relatively straightforward to generate an instance of a surface from standard data sets such as depth maps or clouds of points. Finally, there should be a computationally simple correspondence between the parameters specifying the surface and the actual 3-D shape of the surface, so that images of the surface can be easily generated, thereby allowing the integration of information from multiple images.

A regular 3–D triangulation is an example of a surface representation that meets the criteria stated above, and is the one we have chosen for this paper. In our implementation, all vertices except those on the edges have six neighbors and are initially regularly spaced. Such a mesh defines a surface composed of three-sided planar polygons that we call triangular

facets, or simply facets. Triangular facets are particularly easy to manipulate for image and shadow generation; consequently they are the basis for many 3-D graphics systems. These facets tend to form hexagons and can be used to construct virtually arbitrary surfaces. Finally, standard triangulation algorithms can be used to generate such a surface from noisy real data (Fua and Sander 1992, Szeliski and Tonnesen 1992).

3.2 Material Properties and Their Representation

Objects in the world are composed of many types of material, and the material type can vary across the object's surface in many ways. The key issues, therefore, are the type of material we wish to consider, and how its variation across the surface is to be represented. In general, one can represent a material type by its reflectance function, which maps the wavelength distribution and orientation of a light source, the normal to the surface, and the viewing direction into the color of the image at a point. This function is generally quite complex. However, there are reflectance functions that are not only much simpler, but are also quite common. Such functions are modeled using only one, or, at most, a few, parameters. Consequently, one can accurately model the material properties of a surface by representing these parameters at every point on the surface.

Probably the simplest, and most common, such function is the Lambertian reflectance function. For gray-level images, this function not only has a single parameter, albedo, which is the ratio of outgoing to incoming light intensity, but the image intensity is independent of viewpoint. Image intensity can therefore be used directly when computing surface properties, as explained in Section 4. For this reason, and for the time being, we have chosen to restrict ourselves to Lambertian surfaces. Possible extensions are discussed as future work in Section 6.

Having chosen a specific reflectance function, the remaining issue is how to represent the spatially varying parameter(s). In general, one needs to be able to represent independent parameter values at every point of the surface. In terms of the mesh representation of the surface, this implies some type of spatial sampling of each facet. We have chosen to use two types of spatial sampling. The first is most appropriate when the parameters vary quickly across the surface, and the second when they vary more slowly. For the former case, we use a uniform sampling of each facet, where the intersample spacing corresponds roughly to no more than one or two pixels in any of the images. For the latter case, we use a single value associated with each facet.

As we shall see later, both representations are necessary to handle the various sources of information; the relative importance of their contributions is weighted on a facet-by-facet basis as a function of the images.

3.3 Information Sources for Reconstruction

A number of information sources are available for the reconstruction of a surface and its material properties. Here, we consider two classes of information.

The first class comprises those information sources that do not require more than one image, such as texture gradients, shading, and occlusion edges. When using multiple images and a full 3-D surface representation, however, we can do certain things that cannot be done with a single image. First, the information source can be checked for consistency across all images, taking occlusions into account. Second, when the source is consistent and occlusions are taken into account, the information can be fused over all the images, thereby increasing the accuracy of the reconstruction.

The second class comprises those information sources that require at least two images, such as the triangulation of corresponding points between input images (given camera models and their relative positions). Generally speaking, this source is most useful when corresponding points can be easily identified and their image positions accurately measured. The ease and accuracy of this correspondence can vary significantly from place to place in the image set, and depends critically on the type of feature used. Consequently, whatever the type of feature used, one must be able to identify where in the images that feature provides reliable correspondences, and what accuracy one can expect.

The image feature that we have chosen for correspondence (although it is by no means the only one possible) is simply intensity, because the Lambertian reflectance model described earlier implies that the image intensity of a surface point is independent of the viewing direction. Therefore, corresponding points should have the same intensity in all images. Clearly, intensity can be a reliable feature only when the albedo varies quickly enough on the surface (and, consequently, the images are highly textured), the search space is sufficiently narrow, and the radiometry is the same in all images. Otherwise, there would be significant ambiguity in the correspondence of pixels across the images. Differences in radiometry, however, can be accommodated by first band-passing the images (Poggio *et al.* 1985, Barnard 1989).

In contrast to our approach, traditional correlation-based stereo methods use fixed-size windows in images to measure disparities, which will in general yield correct results only when the surface is parallel to the image plane. Instead, we compare the intensities as projected onto the facets of the surface. Consequently, the reconstruction can be significantly more accurate for slanted surfaces. Some correlation-based algorithms achieve similar results by using variable-shaped windows in the images. Control Data's work (Panton 1978), the Hierarchical Warp Stereo System (Quam 1984), Nishihara's real-time stereo matcher (1984), and the adaptative windows technique described in (Kanade and Okutomi 1990) are examples of such methods. However, they typically use only image-centered representations of the surface.

As for the monocular information source, we have chosen to use shading. There are a

number of reasons for this. First, we are using a Lambertian reflectance model, making shading a relatively simple source of information. Second, shading is most reliable when the albedo varies slowly across the surface, which is the natural complement to intensity correspondence, which requires quickly varying albedo. The complementary nature of these two sources should allow us to accurately recover the surface geometry and material properties for a wide variety of images.

In contrast to our approach, traditional uses of shading information assume that the albedo is constant across the entire surface, which is a major limitation when applied to real images. We overcome this limitation by improving upon a method to deal with discontinuities in albedo alluded to in the summary of (Leclerc and Bobick 1991). We compute the albedo at each facet using the normal to the facet, a light-source direction, and the average of the intensities projected onto the facet from all images. We use the local variation of this computed albedo across the surface as a measure of the correctness of the surface reconstruction. To see why albedo variation is a reasonable measure of correctness, consider the case when the albedo of the real surface is constant. When the geometry of the mesh is correct, then the computed albedo should be approximately the same as the real albedo, and hence should be approximately constant across the mesh. Thus, when the geometry is incorrect, this will generally give rise to variations in the computed albedo that we can take advantage of. Furthermore, by using a *local* variation in the computed albedo, we can deal with surfaces whose albedo is not constant, but instead varies slowly over the surface.

3.4 Combining and Using Information Sources

Simply put, our approach to surface reconstruction is to adjust the parameters of the surface (in the case of the mesh, this means the coordinates of the vertices), until the synthesized images of the surface are most consistent with the information sources described above. This approach requires a number of things. First, one must have an initial estimate of the surface. Second, one must know the light source direction, camera models, and their relative positions—we assume these are provided a priori—so that synthetic images of the surface can be generated. Third, one must have a way of quantifying what is meant by "most consistent with the information sources." Here, we use an objective function that is a linear combination of components, one for each information source, whose weights are determined on a facet-by-facet basis as a function of the images. Finally, one must have a computationally effective means of finding a surface, given the initial estimate, that is reasonably close to the best of all possible surfaces according to the objective function.

Our combined objective function has three components, two of which were mentioned above: an intensity correlation component, and an albedo variation component. A third component is a measure of the smoothness of the surface. The first two components are weighted differently at each facet as a function of the image intensities projected onto the facet, while the surface smoothness component has the same weight everywhere, but is

typically decreased as the iterations proceed.

Since the intensity correlation component depends on the differences in image intensities at a given point on a facet, it is most accurate when the images are highly textured in the areas that the facet projects to. To see this, consider the case when the images have constant intensity in the neighborhood of the projected facet: the difference in intensity will be a constant, independent of small variations in the facet's position or orientation. On the other hand, when the images are highly textured, small changes in the facet can significantly change the value of this component. Thus, we weight the intensity correlation component most strongly for those facets in which the projected image intensities are highly textured.

Conversely, the albedo variation component is most accurate when the intensities within a facet vary slowly. This is because we are assuming that the albedo varies slowly enough across the surface that a constant-albedo facet is a good model for the surface. Since the facets are planar, this should produce images whose intensities are constant within the projected facet. Thus, we weight the albedo variation component most strongly when the projected intensities within a facet vary slowly.

Since rapidly changing albedos produce highly textured image regions, our weighting scheme, in effect, turns off the shading component and turns on the stereo component in such regions. Thus, it provides the shape from shading component with boundary conditions at the edge of regions of slowly varying albedo.

The surface smoothness component is required as a stabilizing term because neither of the above components is likely to be exactly correct, the surfaces are not exactly Lambertian, and the camera positions are not exactly correct: there is noise in the images, and so on. Currently, we use the heuristic technique of starting with a relatively large weight for the smoothness component, and decrease it as the iterations proceed. The theoretically optimal point at which the smoothness weight should no longer be decreased is still an open question. Nonetheless, a single empirically determined value has been used with great success across all of the images presented in this paper when simultaneously using stereo and shape from shading.

4 Details of Surface Model and Optimization Procedure

As discussed in the previous section, 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. For the experiments described in this paper, we have derived this initial estimate using one of the various methods mentioned in Section 5.

The simplest one is to triangulate the smooth depth-map generated by the correlation-based stereo algorithm described in (Fua 1993).

4.1 Images and Camera Models

In this paper, we assume that images are monochrome, and that their camera models are known *a priori*. The set of gray-level images is denoted $\mathbf{G} = (g_1, g_2, \ldots, g_{n_g})$. A point in an image is denoted $\mathbf{u} = (u, v)$, and the intensity of point $\mathbf{u}$ in image g_i is denoted $g_i(\mathbf{u})$. For noninteger values of $\mathbf{u}$ we use bilinear interpolation over the four points represented by the floor and ceiling of the coordinates of $\mathbf{u}$.

The projection of an arbitrary point $\mathbf{x} = (x, y, z)$ in space into image g_i is denoted $\mathbf{m}_i(\mathbf{x})$. There are well-known methods for correcting both geometric and radiometric errors in images, as surveyed in (Baltsavias 1991). Thus, we assume that all effects of lens distortion and the like have been taken care of in producing the input images, so that the projection of a surface into an image is well modeled by a perspective projection. Thus, $\mathbf{u} = \mathbf{m}_i(\mathbf{x})$ can be written as:

$$\begin{aligned} \begin{bmatrix} U \\ V \\ W \end{bmatrix} &= M_i \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \\ u &= U/W \\ v &= V/W, \end{aligned}$$

where M_i is a three-by-four projection matrix.

4.2 Surface Representation

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. Vertices on the edge of a surface may have anywhere from two to five neighbors.

Neighboring vertices are further organized into triangular planar surface elements called *facets*, denoted $\mathbf{F} = (f_1, f_2, \ldots, f_{n_f})$. 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. The representation can be extended in a straight-forward fashion to support different surface resolutions by subdividing facets (which we have done but do not describe in detail here). However, facets of a given resolution will still be required to have approximately equal sides.

4.3 Objective Function

The objective function $\mathcal{E}(\mathcal{S})$ that we use to recover the surface is best described in two equations. In the first equation,

$$\mathcal{E}(\mathcal{S}) = \lambda_D \mathcal{E}_D(\mathcal{S}) + \mathcal{E}_G(\mathcal{S}), \tag{1}$$

$\mathcal{E}(\mathcal{S})$ is decomposed into a linear combination of two components. The first component, $\mathcal{E}_D(\mathcal{S})$, is a measure of the deformation of the surface from a nominal shape, and is independent of the images. This nominal shape represents the shape that the surface would take in the absence of any information from the images. For this paper, it is a plane. Higher-order measures, such as deformation from a sphere, are also possible.

The second component,

$$\mathcal{E}_G(\mathcal{S}) = \lambda_C \mathcal{E}_C(\mathcal{S}) + \lambda_S \mathcal{E}_S(\mathcal{S}) \tag{2}$$

depends on the images, and is the one that drives the reconstruction process. It is further decomposed into a linear combination of the two information sources described in the previous section: a multi-image correlation component, $\mathcal{E}_C(\mathcal{S})$, and a component that depends on the shading of the surface, $\mathcal{E}_S(\mathcal{S})$.

These components, and their relative weights, are described in more detail below.

4.3.1 Surface Deformation Component

As stated earlier, the surface deformation (or smoothness) component is a measure of the deviation of the mesh surface from some nominal smooth shape. When the nominal shape is a plane, we can 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 1(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:

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

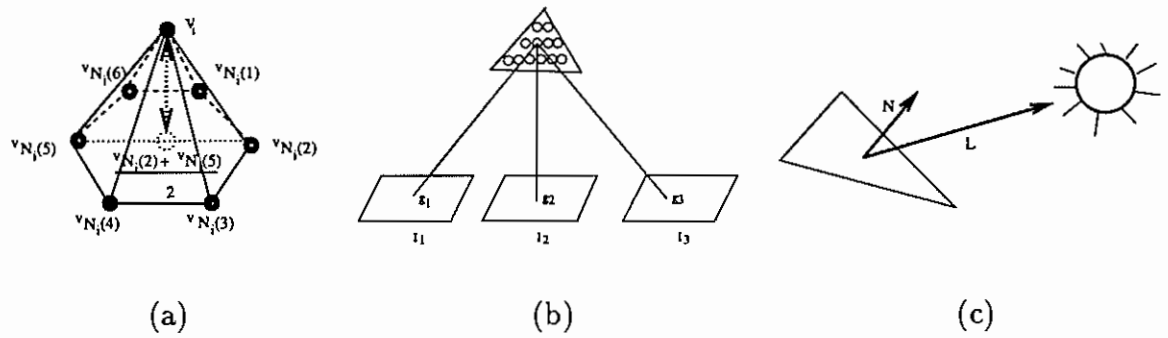


Figure 1: (a) The six neighbors $N_i(j)$ of a vertex v_i are ordered clockwise. The deformation component of the objective function tends to minimize the distance between v_i and the midpoint of diametrally opposed neighbors, represented by the dotted circle. (b) Facets are sampled at regular intervals as illustrated here. We use the gray levels of the projections of these sample points to compute the stereo score. (c) The albedo of each facet is estimated using the facet normal $\overrightarrow{N}$, the light source direction $\overrightarrow{L}$ and the average gray level of the projection of the facet into the images.

Note that this term is also equivalent to the squared directional curvature of the surface when the sides have approximately 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.

4.3.2 Multi-Image Intensity Correlation

The multi-image intensity correlation component is the sum of squared differences in intensity from all the images at a given sample-point on a facet, summed over all sample-points, and summed over all facets. This component is presented in stages in the remainder of this subsection.

First, we define the sample-points of a facet by noting that all points on a triangular facet are a convex combination of its vertices. Thus, we can define the sample-points $\mathbf{x}_{k,l}$ of facet f_k as:

$$\mathbf{x}_{k,l} = \lambda_{l,1}\,\mathbf{x}_{k,1} + \lambda_{l,2}\,\mathbf{x}_{k,2} + \lambda_{l,3}\,\mathbf{x}_{k,3},\ l = 4, \ldots n_s,$$

where $\mathbf{x}_{k,1}$, $\mathbf{x}_{k,2}$, and $\mathbf{x}_{k,3}$ are the coordinates of the vertices of facet f_k, and $\lambda_{l,1}+\lambda_{l,2}+\lambda_{l,3} = 1$. In practice, $\lambda_{l,1}$ and $\lambda_{l,2}$ are both picked at regular intervals in $[0, 1]$ and $\lambda_{l,3}$ is taken to be

$1 - \lambda_{l,1} - \lambda_{l,2}$. In the top half of Figure 1(b), we see an example of the sample-points of a facet.

Next, we develop the sum of squared differences in intensity from all images for a given point $\mathbf{x}$. Recall that a point $\mathbf{x}$ in space is projected into a point $\mathbf{u}$ in image g_i via the perspective transformation $\mathbf{u} = \mathbf{m}_i(\mathbf{x})$. Consequently, the sum of squared differences in intensity from all the images, $\sigma'^2(\mathbf{x})$, is defined by:

$$\begin{aligned} \mu'(\mathbf{x}) &= \frac{1}{n_i} \sum_{i=1}^{n_i} g_i(\mathbf{m}_i(\mathbf{x})) \\ \sigma'^2(\mathbf{x}) &= \frac{1}{n_i} \sum_{i=1}^{n_i} (g_i(\mathbf{m}_i(\mathbf{x})) - \mu'(\mathbf{x}))^2 \end{aligned}$$

Figure 1(b) illustrates the projection of a sample-point of a facet onto several images.

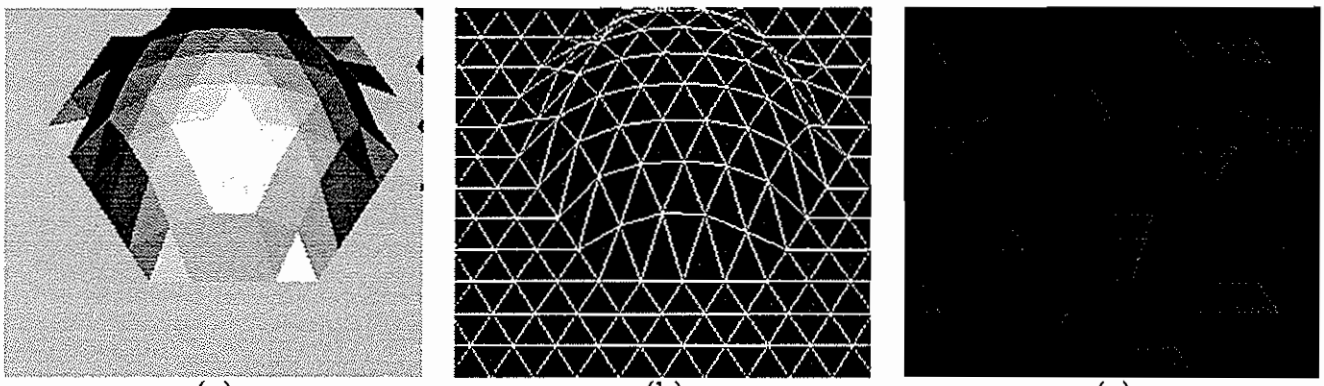
The above definition of $\sigma'^2(\mathbf{x})$ does not take into account occlusions of the surface. To do so, we use a "Facet-ID" image, shown in Figure 2. It is generated by encoding the index i of each facet f_i as a unique color, and projecting the surface into the image plane, using a standard hidden-surface algorithm. Thus, when a sample-point from facet f_k is projected into an image, the index k is compared to the index stored in the Facet-ID image at that point. If they are the same, then the sample-point is visible in that image; otherwise, it is not. Let $v_i(\mathbf{x}) = 1$ when point $\mathbf{x}$ is determined to be visible in image g_i by the method above, and $v_i(\mathbf{x}) = 0$ otherwise. Then, the correct form for the sum of squared differences in intensity at a point $\mathbf{x}$ is defined by:

$$\begin{aligned} \mu(\mathbf{x}) &= \frac{\sum_{i=1}^{n_i} v_i(\mathbf{x}) g_i(\mathbf{m}_i(\mathbf{x}))}{\sum_{i=1}^{n_i} v_i(\mathbf{x})} \\ \sigma^2(\mathbf{x}) &= \frac{\sum_{i=1}^{n_i} v_i(\mathbf{x}) \left(g_i(\mathbf{m}_i(\mathbf{x})) - \mu(\mathbf{x})\right)^2}{\sum_{i=1}^{n_i} v_i(\mathbf{x})} \end{aligned}$$

When the sample-point is visible in fewer than two images (that is, when $\sum_{i=1}^{n_i} v_i(\mathbf{x}) < 2$), the above variance has no meaning and is taken to be 0. Let s_k denote the number of facet samples for facet k for which the variance is meaningful. Summing $\sigma^2(\mathbf{x})$ over all sample-points and over all facets and normalizing by the number of meaningful sample-points yields the multi-image intensity correlation component:

$$\mathcal{E}_C(\mathcal{S}) = \frac{\sum_{k=1}^{n_f} c_k \sum_{l=4}^{n_s} \sigma^2(\mathbf{x}_{k,l})}{\sum_{k=1}^{n_f} s_k},$$

where c_k is a number between 0 and 1 that weights the contribution from each facet differently, depending on the average degree of texturing within a facet (see Section 4.3.4).



(a) (b) (c)

Figure 2: Illustration of the 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).

When the original surface giving rise to the images is sufficiently textured, this component should be smallest when the surface $\mathcal{S}$ closely approximates the original surface. However, when the surface has constant, or nearly constant, albedo this component would be small for many different surfaces. As an extreme example of this ambiguity, consider a planar surface with constant albedo. This produces images with constant intensity. Thus, this component will not be able to constrain the shape of the surface, since the difference in intensity will be zero for all surfaces.

4.3.3 Shading

The shading component of the objective function is the sum, over all facets, of the difference between the computed albedo of the facet and the computed albedos of all of its neighbors. The motivation for this component, and its precise form, follow.

Recall that the Lambertian reflectance model defines the intensity g at a point on a surface with a unit surface normal $\overrightarrow{N}$ as:

$$g = \alpha(a + b\overrightarrow{N} \cdot \overrightarrow{L}), \tag{3}$$

where α is the albedo of the surface, a is the magnitude of the ambient light, b is the magnitude of a point light source, and $\overrightarrow{L}$ is the direction of the point light source as depicted

in Figure 1(c).

Note that g is independent of the viewing direction. Consequently, if we were to image a planar Lambertian facet from several points of view, its intensity would be the same for all pixels in the projection of the facet. Conversely, if we were to measure the average intensity $\bar{g}_k$ of all of the pixels within the projection of a facet f_k, we could compute its albedo, α_k, as follows:

$$\alpha_k = \frac{\bar{g}_k}{(a + b\vec{N} \cdot \vec{L})}. \tag{4}$$

This assumes, of course, that the facet is well-modeled by a single albedo, that the variation in intensity is due only to noise, and that the light source is located at infinity. In this paper, we assume that the ambient and direct illumination (*i.e.*, a, b, and $\vec{L}$) are either given or estimated from the initial surface and images, as was done in (Leclerc and Bobick 1991).

The average intensity $\bar{g}_k$ of a facet is computed by scanning over all the Facet-ID images for index k, and taking the average of the intensities at matching points in the corresponding images. This computed albedo minimizes the mean squared error between the synthesized images of the mesh surface and the input images.

Now, if the original surface had exactly constant albedo, and if our mesh surface were a good approximation to the original surface, then the computed albedos should be approximately the same across all facets. Thus, some measure of the variation in computed albedos would be a good measure of the correctness of the mesh surface. If the albedo varies slowly across the surface, we propose that an appropriate measure of this variation is the difference between the computed albedo at the facet and the computed albedos of all its neighboring facets:

$$\mathcal{E}_S(\mathcal{S}) = \sum_{k=1}^{n_f} (1 - c_k) \sum_{j \in N_f(k)} (1 - c_j)(\alpha_k - \alpha_j)^2,$$

where $N_f(k)$ is the set of indices of the facets that are neighbors of facet f_k, and c_k and c_j are numbers between 0 and 1 that depend on the degree of texturing within facets f_k and f_j.

This term can be exactly zero only where the albedo is constant. However, as will be shown in Section 5, it provides a reasonable constraint on the variation of surface normals when the albedo variation is slow. It constrains the normals of neighboring facets projecting to areas of similar gray-levels to have similar orientations. As a result, it prevents the surface normals from varying wildly in the absence of strong image gray-level variations and acts as an image-dependent regularization term that prevents the surface from wrinkling in bland areas.

4.3.4 Combining the Components

Recall that the objective function $\mathcal{E}(\mathcal{S})$ is a linear combination of three components:

$$\mathcal{E}(\mathcal{S}) = \lambda_D \mathcal{E}_D(\mathcal{S}) + \lambda_C \mathcal{E}_C(\mathcal{S}) + \lambda_S \mathcal{E}_S(\mathcal{S}),$$

where the last two components are themselves linear combinations of subcomponents computed on a per-facet basis:

$$\begin{aligned} \mathcal{E}_C(\mathcal{S}) &= (\sum_{k=1}^{n_f} c_k \sum_{l=4}^{n_s} \sigma^2(\mathbf{x}_{k,l}))/\sum_{k=1}^{n_f} s_k \\ \mathcal{E}_S(\mathcal{S}) &= \sum_{k=1}^{n_f}(1-c_k) \sum_{j \in N_f(k)} (1-c_j)(\alpha_k - \alpha_j)^2. \end{aligned} \tag{5}$$

Thus, one needs to specify both the λs, defining the relative weights of the components, and the c_ks, defining the relative weights of the two image-based components for each facet.

The λ weights are defined as follows:

$$\begin{aligned} \lambda_D &= \frac{\lambda'_D}{\| \vec{\nabla} \mathcal{E}_D(\mathcal{S}^0) \|} \\ \lambda_C &= \frac{\lambda'_C}{\| \vec{\nabla} \mathcal{E}_C(\mathcal{S}^0) \|} \\ \lambda_S &= \frac{\lambda'_S}{\| \vec{\nabla} \mathcal{E}_S(\mathcal{S}^0) \|}, \end{aligned} \tag{6}$$

where $\mathcal{S}^0$ is the initial estimate of the surface, and the λ's are user-defined weights. Normalizing each component by the magnitude of its initial gradient allows the components to have roughly the same influence when the λ's are equal. Thus, the user can more easily specify the relative contributions of each component in an image-independent fashion. This normalization scheme was used with great success in (Fua and Leclerc 1990), and is analogous to standard constrained optimization techniques in which the various constraints are scaled so that their eigenvalues have comparable magnitudes (Luenberger 1984).

As mentioned earlier, the c_k weights are a function of the degree of texturing in the intensities projected within a facet f_k. A simple measure of the degree of texturing within a facet is the variance in intensity of all the pixels projecting onto the facet, denoted $\sigma_k(\mathcal{S})$ (using the Facet-ID image to accommodate occlusions). We have empirically determined that using the logarithm of $\sigma_k(\mathcal{S})$ yields the most stable results for a large set of images:

$$c_k = a \log (1 + \sigma_k(\mathcal{S})) + b, \tag{7}$$

where a and b are normalizing factors chosen so that the smallest c_k is zero, and the largest is one.

4.4 The Optimization Procedure

The purpose of the optimization procedure is to iteratively modify the surface $\mathcal{S}$ so as to minimize $\mathcal{E}(\mathcal{S})$, given some initial estimate $\mathcal{S}^0$, and some value for the weights λ'_S, λ'_C, and λ'_D (where $\lambda'_S + \lambda'_C + \lambda'_D = 1$) defined in Equation 6. Ideally, one would like to use as small a value of the deformation weight λ'_D as possible so as to minimize the bias introduced by this term. However, in practice, λ'_D serves a dual purpose. First, since the surface deformation term is a quadratic function of the vertex coordinates, it "convexifies" the energy landscape and improves the convergence properties of the optimization procedure. Second, as discussed above and shown in Section 5, in the absence of a smoothing term, the objective function may overfit the data and wrinkle the surface excessively. Furthermore, the c_k weights of Equations 5 and 7 are computed for the initial position of the mesh and are meaningful only when it is relatively close to the actual surface.

Consequently, we use an optimization method that is inspired by the heuristic technique known as a continuation method (Terzopoulos 1986, Leclerc 1989a, Leclerc 1989b, Leclerc and Bobick 1991). We first "turn off" the shading term by setting λ'_S (Equation 6) to 0 and setting λ'_D to a value that is large enough to sufficiently convexify the energy landscape but small enough to allow curvature in the surface. In this paper, we take the initial value of both λ'_D and λ'_C to be 0.5. Given the initial estimate $\mathcal{S}^0$, a local minimum of this approximate objective function is found, using a standard optimization procedure. Then, λ'_D is decreased slightly, and the optimization procedure is applied again, starting at the local minimum found for the previous approximation. This cycle is repeated until λ'_D is decreased to the desired value. Finally we "turn on" the shading term, compute the c_k weights and reoptimize. In all examples shown in Section 5, we use $\lambda'_C = \lambda'_S = 0.4$ and $\lambda'_D = 0.2$ for this final stage.

The stereo component effectively uses only zeroth-order information about the surface (i.e., the position of the vertices), whereas shading uses first-order information about the surface (i.e., its normals). Thus, by optimizing the stereo component first, we effectively compute the zeroth-order properties of the surface and set up boundary conditions that the shading component can then use to compute the first-order properties of the surface in textureless regions. In Section 5, we will show that this leads to a significant improvement over using the stereo component alone.

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 of the coordinates to vary simultaneously.

The optimization procedure we use at every stage is a standard conjugate-gradient descent procedure called FRPRMN (from (Press *et al.* 1986)) in conjunction with a simple line-search algorithm. The conjugate-gradient procedure requires three inputs: (1) a function that returns the value of the objective function for any $\mathcal{S}$; (2) a function that returns the gradient of $\mathcal{E}(\mathcal{S})$, that is, a vector whose elements are the partial derivatives of $\mathcal{E}(\mathcal{S})$ with respect to the vertex coordinates, evaluated at $\mathcal{S}$; and (3) an initial estimate $\mathcal{S}^0$.

Since it would be significantly slower to compute the gradient of $\mathcal{E}(\mathcal{S})$ using finite differences than analytically, we do the latter. The analytical expression of this gradient is conceptually straightforward, but is fairly complicated to derive manually. We have used the Maple [1] mathematical package to derive some of the terms. Maple directly yields the C code used in our implementation. We summarize the calculation of the derivatives below in general terms.

The derivatives of the stereo term are linear combinations of image intensity derivatives and of derivatives of the 3-D projections of points onto the images. Since we use bilinear-interpolation of image values, the first derivatives of image intensity are linear combinations of the image intensities in the immediate neighborhood of the projection. Since sample-points are linear combinations in projective space of the mesh vertices, their projections are ratios of linear combinations of the projections of the vertices, which themselves depend linearly on the vertex coordinates. Consequently, the derivatives of these projections are ratios of linear combinations of the vertex coordinates and squares of linear combinations of the vertex coordinates.

Similarly, the derivatives of the shading term depend on the derivatives of the surface normal, which can be easily derived analytically, and from the derivative of the mean gray-level in the facets. In this work, the shading term is used mainly in the fairly uniform areas where the latter derivative is assumed to be small and therefore neglected.

4.5 Computational Complexity and Convergence Issues

Each iteration of the conjugate gradient algorithm typically involves one evaluation of the gradient of the objective function and four to eight evaluations of the objective function itself. The cost of evaluating the stereo term grows as the product of the number of facets, the number of samples per facet, and the number of images. Because the albedo computation involves scanning the Facet-ID image, the dominant cost of evaluating the albedo term grows as the number of pixels per image times the number of images. The cost of evaluating the deformation term grows as the number of vertices and is small by comparison with the other two. For example, in the case of the face images shown in Subsection 5.2.2, the meshes have

[1] Trademark, Waterloo Maple Software

approximately 800 vertices and 1500 facets. We use six samples per facet and three 128x200 images. It takes about 0.6 second to evaluate the stereo energy, 0.3 second to evaluate the albedo energy and .01 second to compute the deformation energy on an R4000 SGI Indigo. Each iteration therefore takes from 5 to 10 seconds, and the computation of the final results shown in this paper took a little less than 10 minutes.

Since the optimization uses image derivatives, our technique is valid only if a majority of the facet samples project to within a few pixels of where they should be; otherwise the gradient of the objective function is meaningless and the algorithm cannot converge. This problem can be alleviated by using a coarse mesh applied to a coarse level of a gaussian pyramid, and progressively increasing the resolutions of both mesh and images. Proving the convergence of the algorithm in the general case is beyond the scope of the paper. However, in Section 5, we use both synthetic and real world examples to show that the algorithm converges when the condition stated above holds, that is, when the initial estimate is good enough for the vertices of the mesh to project to within a few pixels of their true locations.

Standard correlation-based techniques can provide starting points that have the required properties. For example, the specific algorithm we use in this paper (Fua 1993) has been shown to find few false matches and to yield a precision in the order of one pixel in disparity in the areas where it finds relatively dense matches.

5 Behavior of the Objective Function and Results

We first illustrate the behavior of the complete objective function using synthetic data. We then show that the same behavior can be observed with real data, allowing us to generate accurate 3-D reconstructions of real surfaces from multiple images.

5.1 Synthetic Data

To demonstrate the properties of the objective function of Equation 1 and the influence of the coefficients defined in Equations 6 and 7, we use as input the five synthetic images of a shaded hemisphere with variable albedo shown at the bottom of Figure 3, both with and without the addition of white noise. Each column of the figure illustrates the steps used in the creation of the image at the bottom of the column. We begin with a mesh and an albedo map, shown in the top row. Then, for each view, two images are produced. The first image (second row of the figure) is the albedo map texture-mapped onto the mesh from the final image's point of view. The second image (third row of the figure) is a shaded view of the mesh, using a constant albedo equal to one. The final image is the point-by-point product of these two images because, by Equation 3, the imaged intensity of a Lambertian surface is the product of the albedo (first image) and the inner product of the light source and the surface normal (second image).

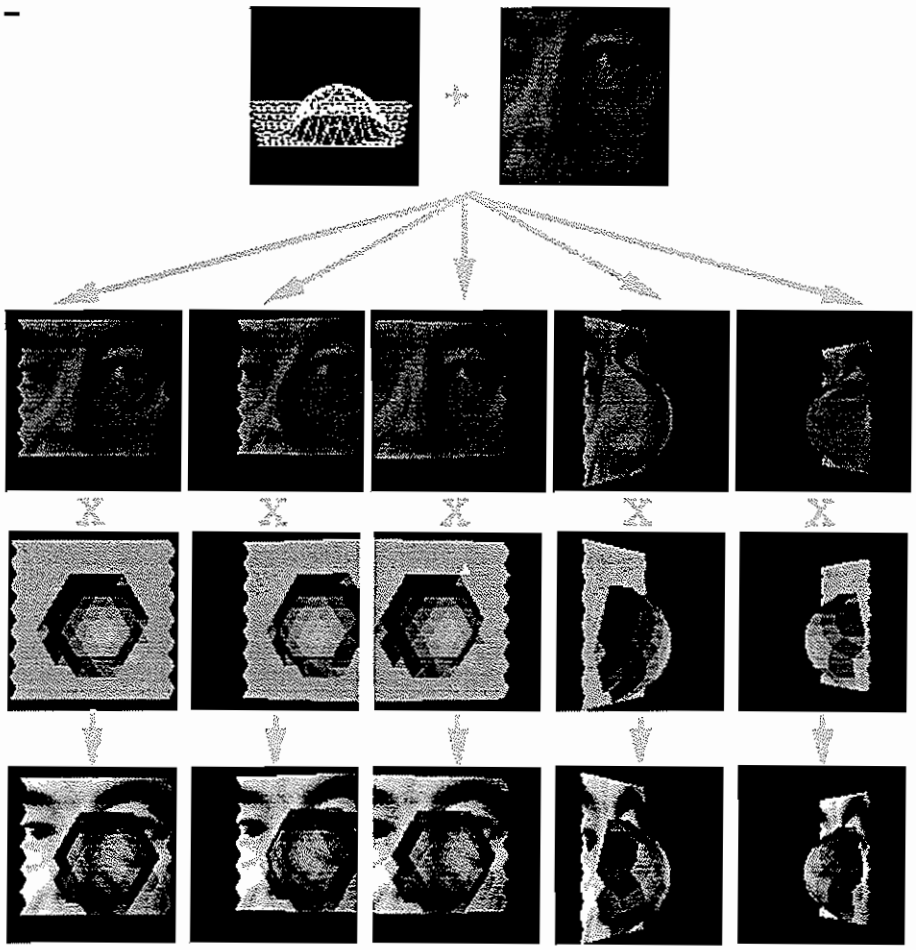


Figure 3: The making of synthetic images of a shaded hemisphere with variable albedo that conforms to our Lambertian model.

Figure 4 depicts graphically the result of our experiments. In each experiment we randomized the mesh by adding random numbers to the coordinates of the mesh vertices, and added different amounts of noise to the input images. We then used our optimization procedure to estimate the true hemispherical shape and true albedo map. More precisely, starting from our randomized initial estimate, we first use intensity correlation alone and progressively decrease the value of the λ'_D parameter of Equation 6 from 0.5 to 0. We then turn on the shading term by setting both λ'_D and λ'_S to 0.4, compute the c_ks of Equation 7, and optimize the full objective function. To show the stability of the process, we recompute the c_ks for the optimized mesh and perform a second optimization using the updated values.

The first column of Figure 4 is for experiments using only the first, second, and third im-

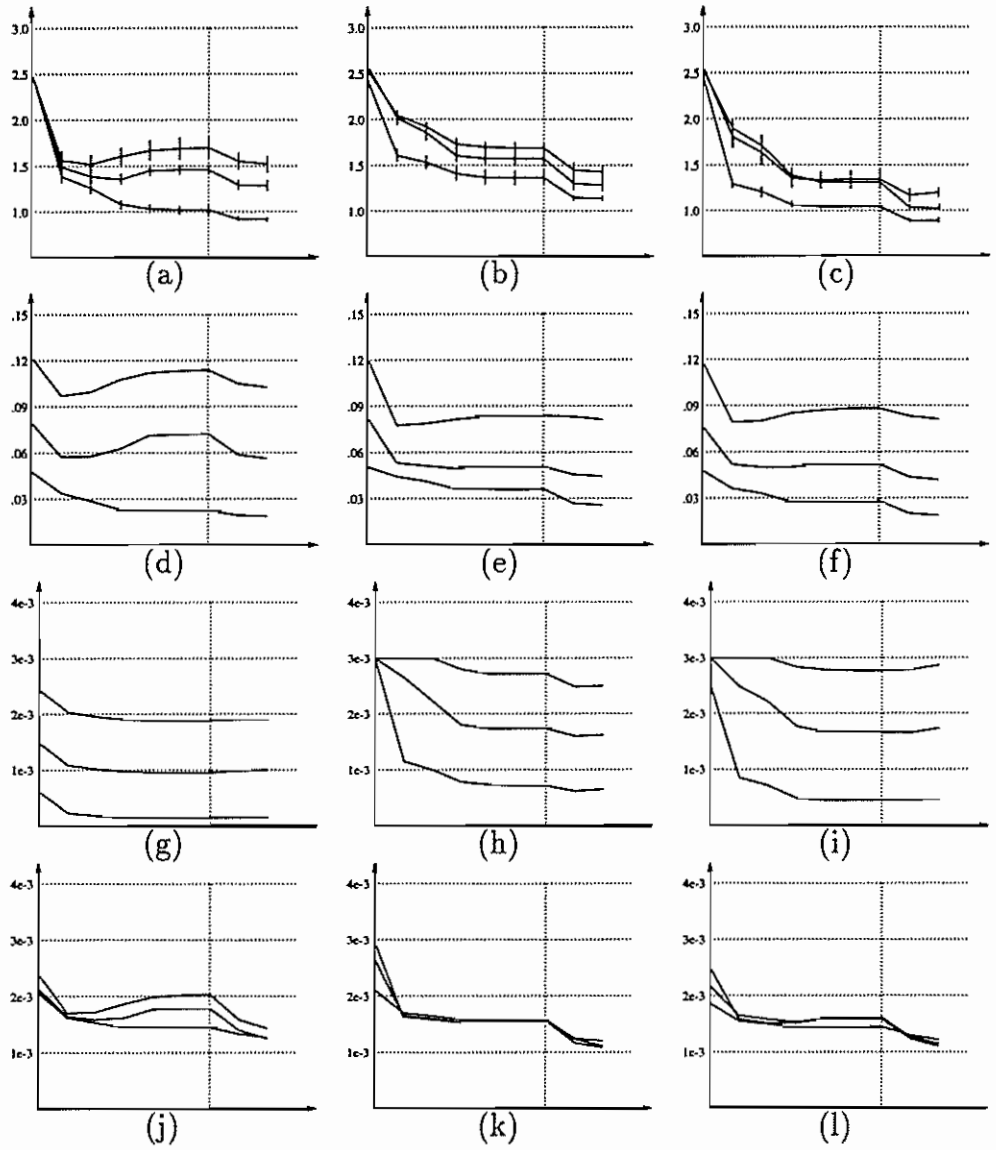


Figure 4: Graphs of the errors and objective function components while fitting a surface model to the synthetic shaded hemisphere images of Figure 3. These graphs are explained in detail in the text. (a,b,c) Average error in recovered elevation expressed in the same unit as the radius of the hemisphere, which is equal to 35. (d,e,f) Average error in recovered albedo. (g,h,i) $\mathcal{E}_C$, the stereo component of the energy. (j,k,l) $\mathcal{E}_S$, the shading component of the energy.

ages from Figure 3, where there is little self-occlusion. The second column is for experiments

using the first, fourth, and fifth images, where there is a significant amount of self-occlusion. Finally, the third column is for experiments using all five images. In this particular 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 first row from the top of Figure 4 is a graph of the average squared error in elevation (the ordinate) versus decreasing λ'_D (the abscissa). To the left of the dotted vertical line, only the intensity correlation component is used. To the right, both the intensity correlation and shading components are used. The different curves are for different amounts of noise in the input images. The bottom curve corresponds to no noise (other than quantization error), the middle curve is for a noise variance of 4% of the image dynamic range, and the top curve is for a noise variance of 8%. The short vertical lines along the curves indicate the standard deviation of the average error over the 20 experiments performed to derive each curve.

Note that an error of 1 unit in elevation corresponds to a difference in computed disparities of approximately 0.25 pixels for projections from image 1 into images 2 or 3 and of approximately 0.80 pixels for projections from image 1 into images 4 or 5. In these experiments, the noise added to the elevations was gaussian of variance randomly chosen between 1 and 5, resulting in errors in the projections in the order of one pixel for images 2 and 3, and of three pixels for images 4 and 5. In this particular case, however, much larger errors can be tolerated: the hemisphere has a radius of 35 elevation units and can be recovered starting from a flat sheet.

The second row of Figure 4 is a graph of the average error in computed albedo. The third row is the average value of the intensity correlation component, $\mathcal{E}_C(\mathcal{S})$, and the fourth row is the average value of the shading component, $\mathcal{E}_S(\mathcal{S})$.

Note that, as λ'_D decreases and stereo alone is used (*i.e.*, as the abscissa is traversed rightwards to the dotted vertical line), the average elevation error decreases when there is no noise in the input image (bottom curve), as does the average albedo error and the two components of the objective function. However, when the images are noisy, the elevation error (first row) stops decreasing and may even begin to increase as we start fitting to the gray-level noise, even though the value of the intensity correlation component (third row) continues to decrease, as it must. Furthermore, both the albedo error (second row) and the shading component (fourth row) also begin to increase when the elevation error does. This is natural since for smaller values of λ'_D the surface becomes rougher and its normals less well-behaved. As a result, the estimated albedos of Equation 4 become less reliable and noisier.

In other words, an increase in the shading component provides us with a warning that we are starting to overfit the data. This is a valuable behavior in itself. Furthermore, by turning on the shading component of our objective function (those parts of the graphs that are to the right of the vertical dotted line), we can bring down both the error in albedo

and the value of the albedo component with at worst a modest increase in the value of the stereo component, resulting in an overall reduction of the elevation error. Even when there is nothing but quantization noise in the image, the addition of the shading component can make a small, but still noticeable difference. The reason for this is twofold:

1. The shading component averages over whole facets and is therefore less sensitive to uncorrelated noise.

2. The shading component uses absolute intensity values, whereas the stereo component uses intensity differences. Thus, in the presence of noise in textureless areas, the signal-to-noise ratio for the absolute values (used by the shading component) is larger than for the differences (used by the stereo component), thereby making the shading term more robust.

However, in our experience, the shading term can be used reliably only when the surface is relatively close to the correct answer. This is not surprising since stereo deals directly with elevations, whereas shading deals with derivatives of elevation. Consequently, we have chosen the optimization schedule described above where we first optimize using stereo alone and turn on shading only later.

There is another important point to note about these results. The elevation errors in the second column, that is, those generated using images 1, 4, and 5 with a lot of self-occlusion are very close to those of the first column, that is those generated using images 1, 2, and 3 with little self-occlusion, while those in the final column (using all five images) are significantly better. In addition, the results for images 1,4, and 5 are even slightly better than those for images 1,2, and 3 in the presence of noise because the former correspond to larger baselines. In other words, having the same number of images, but with significant self-occlusions, does not hurt our procedure. Furthermore, adding new images that contain significant self-occlusions actually improves the results.

To further demonstrate the importance of being able to combine stereo and shape from shading, even in the presence of slowly varying albedo, we present in Figure 5 a second synthetic example. If we band-pass the images using a difference of gaussians, there is not enough texture for stereo to work effectively and the surface computed using stereo alone is not very good. However by combining shape-from-shading with stereo, the result improves markedly and the recovered surface becomes very close to the synthetic one used to generate the images. In this case our starting point was a flat plane, corresponding to errors of up to 4 pixels in the initial projections of the mesh vertices.

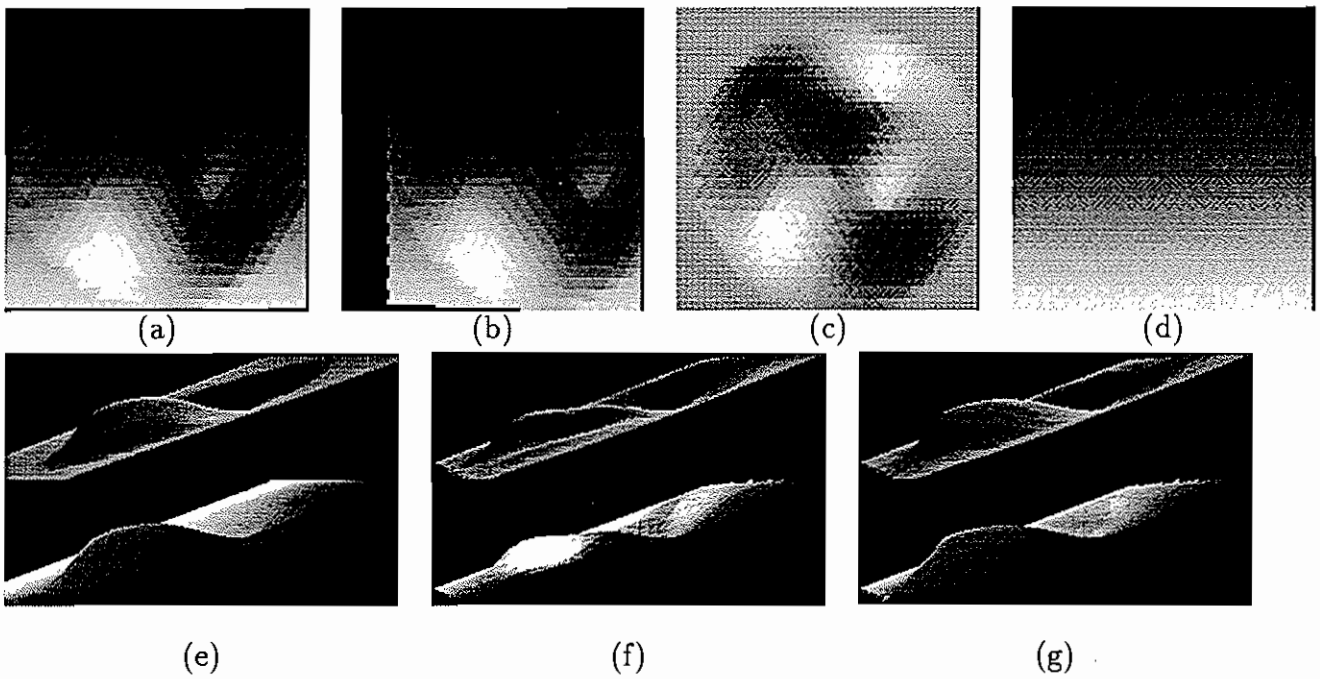


Figure 5: Combining shape from shading and stereo in the presence of a slowly varying albedo. (a,b) A synthetic stereo pair generated by rendering the shaded surface shown in (c) using the albedo map shown in (d). (e) The original shaded surface and albedo map from (c) and (d) seen from the side. (f) The surface and albedo map computed using stereo alone on difference of gaussians of the images and starting from a plane. (g) The surface and albedo map recovered by combining shape from shading and stereo. The difference-of-gaussian images do not retain enough information and stereo alone finds a poor quality solution. The shape-from-shading term, however, allows a better recovery of the surface even though the albedo is not constant.

5.2 Real Images

We now turn to real images and show that the same properties can be observed there.

5.2.1 Aerial Images

In Figure 6 we show the result of running the stereo component of our objective function on an aerial stereo pair of a sharp ridge. Note that the radiometry of the left and right images is actually slightly different. As suggested in Section 3.3, we correct for this in the computation

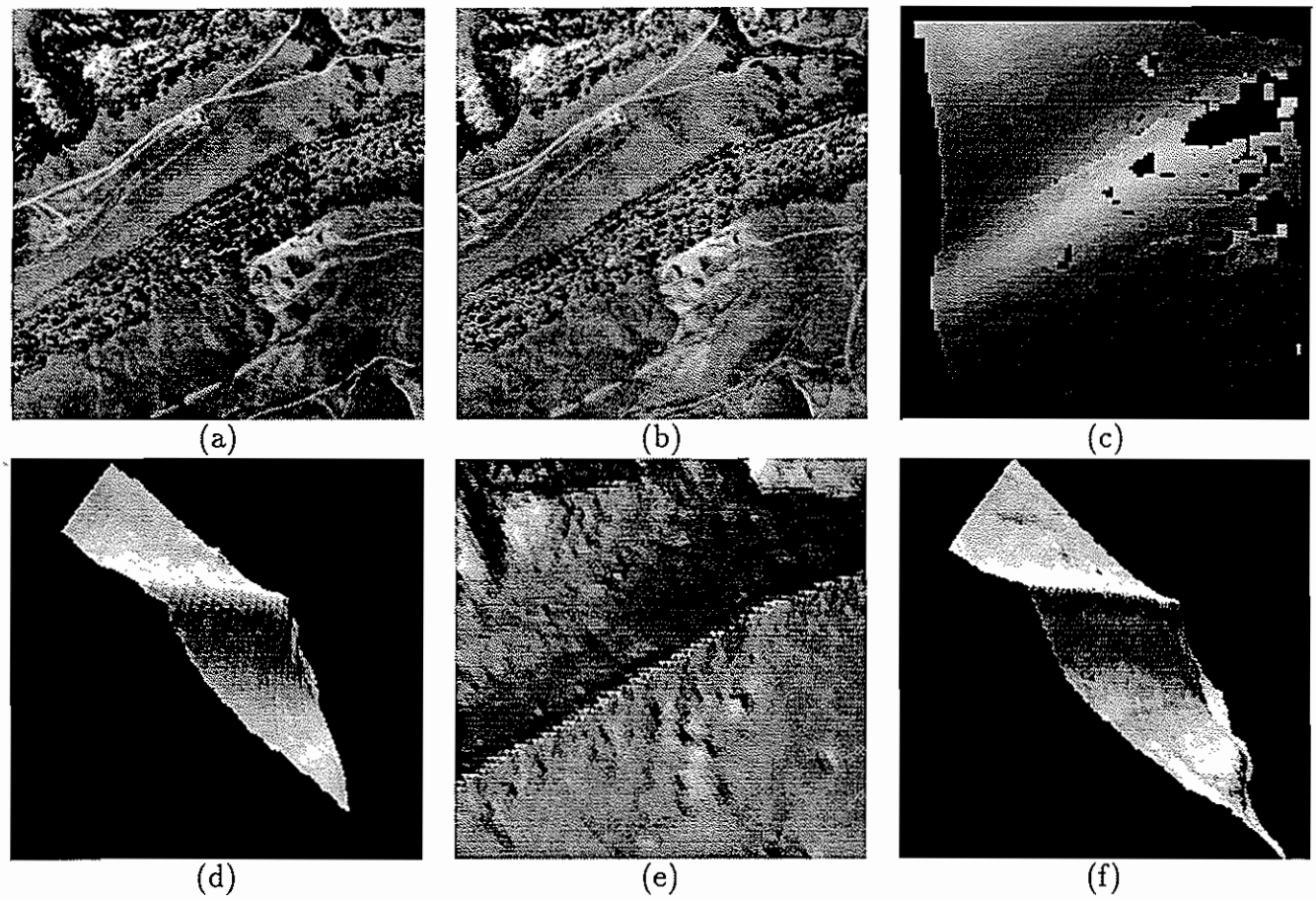


Figure 6: (a,b) A stereo pair of images of the Martin-Marietta ALV test site. (c) Disparity map computed using a correlation-based algorithm. The black areas indicate that the stereo algorithm could not find a match. Elsewhere, lighter grays indicate higher elevations. (d) The initial surface estimate derived by smoothing and interpolation of the disparity map. It is shown as a shaded surface viewed by an observer located above the upper left corner of the scene. (e,f) Shaded views of the mesh after optimization. Note that the ridge has become very sharp and that the shadow casting cliffs visible in the top portion of the image are recovered. They are clearly visible at the top of (e) and the bottom right corner of (f).

of the stereo term of our objective function by first high-pass filtering each image. Here, we use the difference between the image and its gaussian convolution.

We then optimize the mesh using the continuation method schedule described in section 4.4, that is, starting with $\lambda'_D = \lambda'_C = 0.5, \lambda'_S = 0.0$ and then progressively reducing λ'_D to 0.3 and increasing λ'_C to 0.7. Note that the recovered ridge is much sharper than in the original stereo result and that details in the upper part of the image are well recovered. The difference in the ridge elevation in the original and final estimate is approximately 40 feet, which translates to 2 pixels in disparity. Turning on the shape-from-shading term yields a result that is visually indistinguishable from the one shown here: the images are textured enough for stereo alone to be effective.

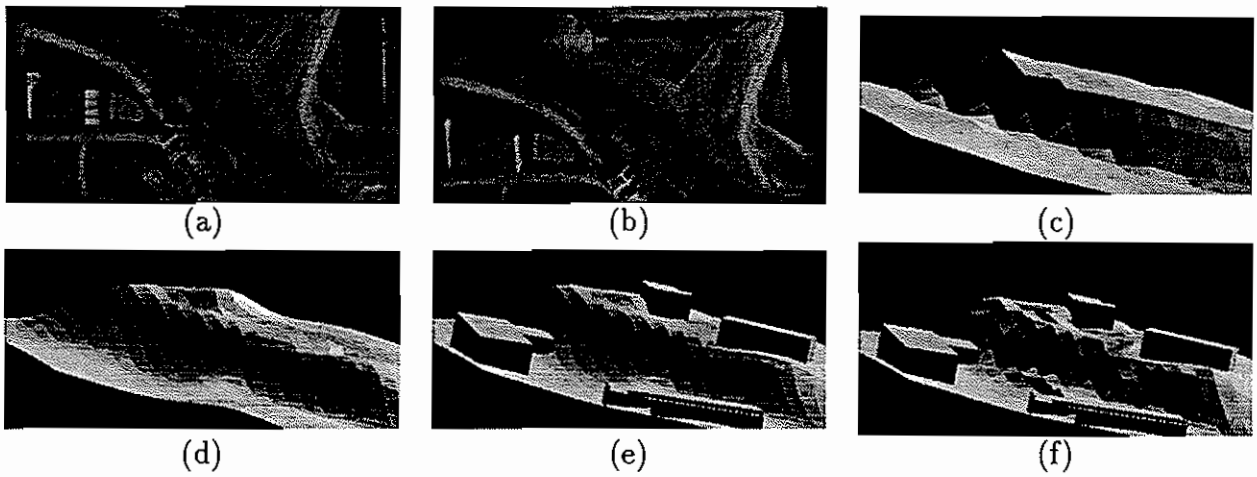


Figure 7: (a) (b) Two of a series of images of a semi-urban site. The images were taken with different light source directions. (c) A rough estimate of the ground-level surface (d) Surface after optimization using stereo alone. (e) Surface after optimization using both stereo and hand-entered buildings to mask occluded areas. (e) Surface after optimization using both stereo and shape from shading. In this case, the illuminations of the different images are different and there are very few bland, untextured areas. As a result, stereo alone performs better.

In Figure 7, we demonstrate a possible application of our technique to semiautomated cartography in a semiurban environment. We use images of a model board, each of them being taken with a different light-source direction. By fitting 3–D snakes to some of the roads in the scene and fitting a surface to them, we have generated a rough terrain model that we have then optimized using the same schedule as before. Because of the presence of buildings that cannot be well described by our mesh model and even though we use relatively large facets, the resulting surface model is too bumpy. We can improve upon this situation by manually specifying the locations of the buildings, modeling them as extruded objects, and using them to mask out occluded areas during the optimization. For comparison's sake,

we also show the result of simultaneously using stereo and shape from shading. In this case, because the illumination in each image is different and there are very few bland areas, the shape-from-shading term actually degrades the result.

5.2.2 Face Images

In Figure 8 we show two triplets of images of faces. They have been produced using the INRIA three-camera system (Faugeras and Toscani 1986) that provides us with the camera models we need to perform our computations. In this case it is essential to have more than two images to be able to reconstruct both sides of the face because of self-occlusions. For each triplet, we have computed disparity maps corresponding to images 1 and 2 and to images 1 and 3 and combined them to produce the depth maps shown in the rightmost column of the figure using the algorithms described in (Fua 1993).

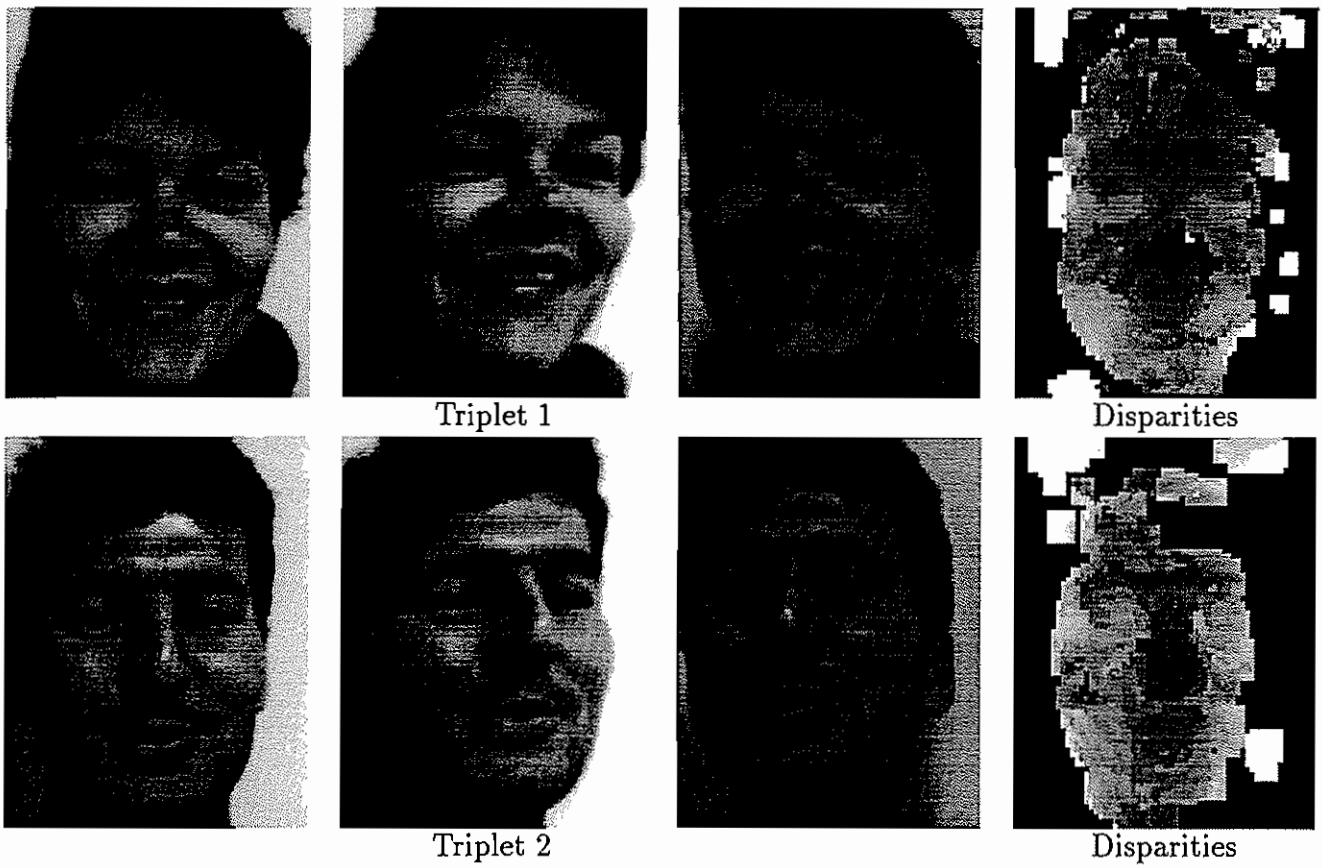


Figure 8: Triplets of face images and corresponding disparity maps (courtesy of INRIA).

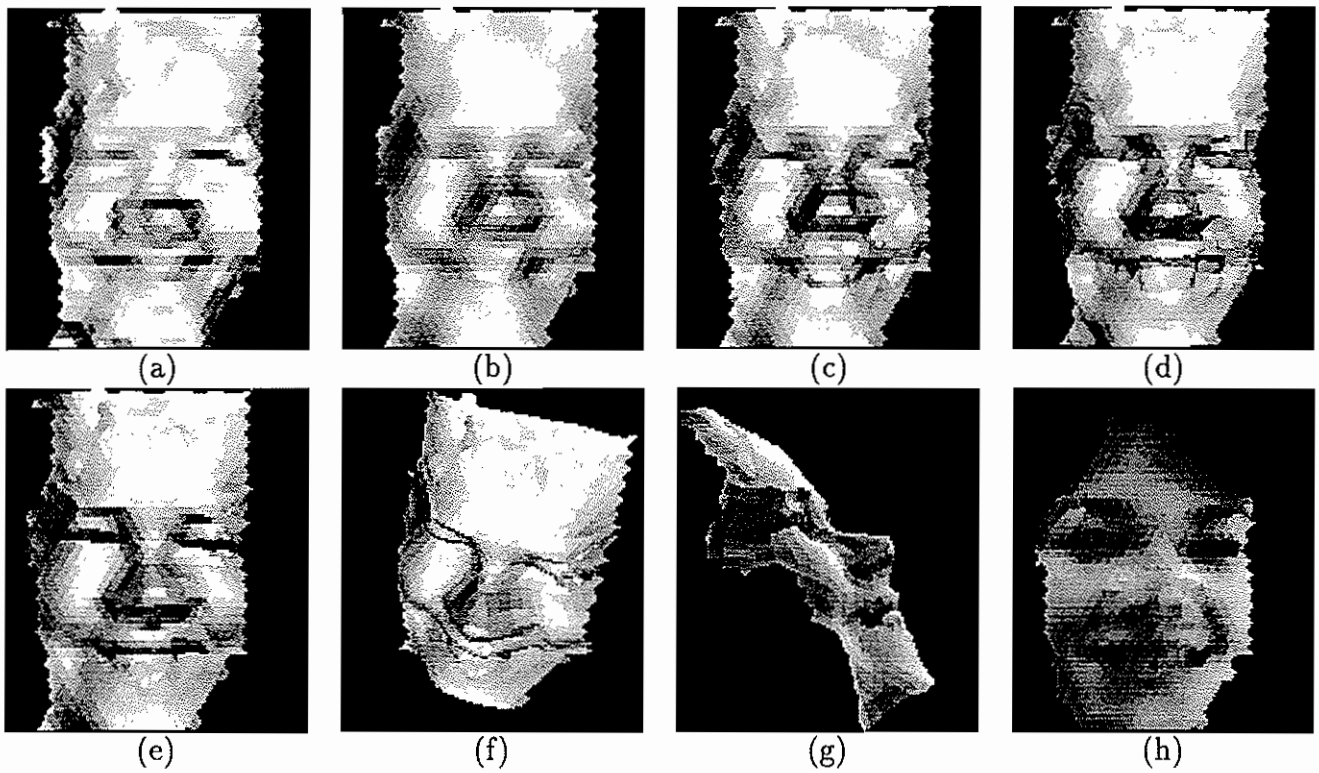


Figure 9: Results for the first triplet of Figure 8. (a) Shaded view of the mesh generated by smoothing and triangulating the computed disparity map. We use it as the starting condition for our optimization procedure. (b,c,d) The mesh after optimization using only the stereo term, with progressively less smoothing. (e,f,g) Several views of the mesh after optimization using both stereo and shading. (h) The recovered albedo map. The albedo of the nose appears fairly similar to that of the other skin areas, showing that its geometry has been well recovered. The main problem with this map is the dark streak caused by the self-shadowing crease on the right side of the face. Our algorithm does not currently handle shadows and incorrectly models them as areas of lower albedo.

The depth maps have then been smoothed and triangulated to produce the initial surfaces shown in the upper left corner of Figures 9 and 10. In the first row of these two figures, we show the result of the optimization using stereo alone as we progressively decrease the smoothness constraint and allow all three vertex coordinates to be adjusted. Note that in the first triplet (Figure 9), we recover more and more detail until the surface eventually

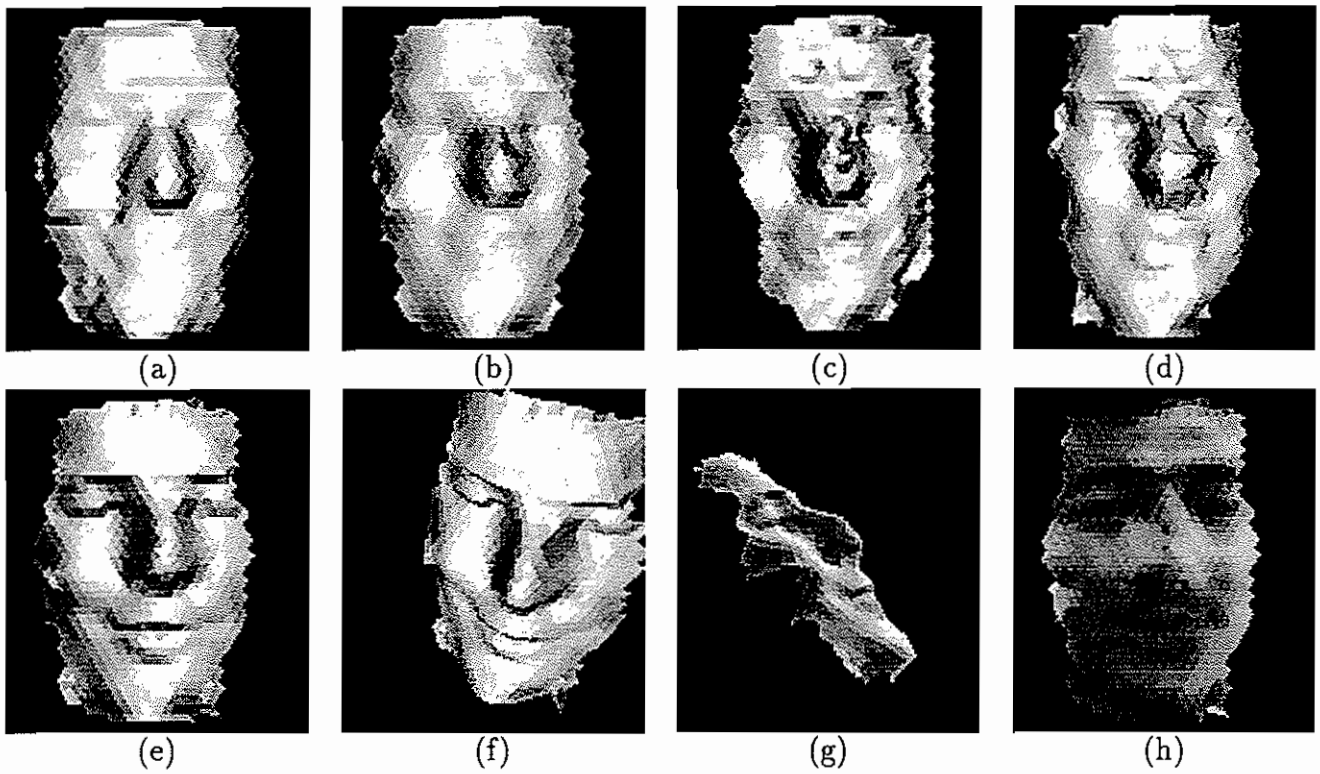


Figure 10: Results for the second triplet of Figure 8 presented in the same fashion as in Figure 9. There are strong specularities on the nose and the stereo term alone performs poorly. However, by using the shading term, the algorithm takes advantage of the monocular information present around the specularities and yields a much better result.

starts to wrinkle, without apparent improvement in accuracy. The second triplet poses an even more difficult problem: there are strong specularities on both the forehead and the nose that strongly violate our Lambertian model. Because there are very few other points that can be matched on the nose, the algorithm latches on to these specularities and yields a poor result. These two sets of images therefore present our algorithm with problems that are very similar to those discussed in Section 5.1.

In the bottom row of Figures 9 and 10, we show our final results obtained by turning on the shading term and reoptimizing the meshes. In Figure 12, we show the corresponding values of the c_k coefficients of Equation 7 and the contribution of each facet to the overall energy. For these images we did not know *a priori* the light-source direction; we therefore estimated it by choosing the direction that minimized the shading component of the objective

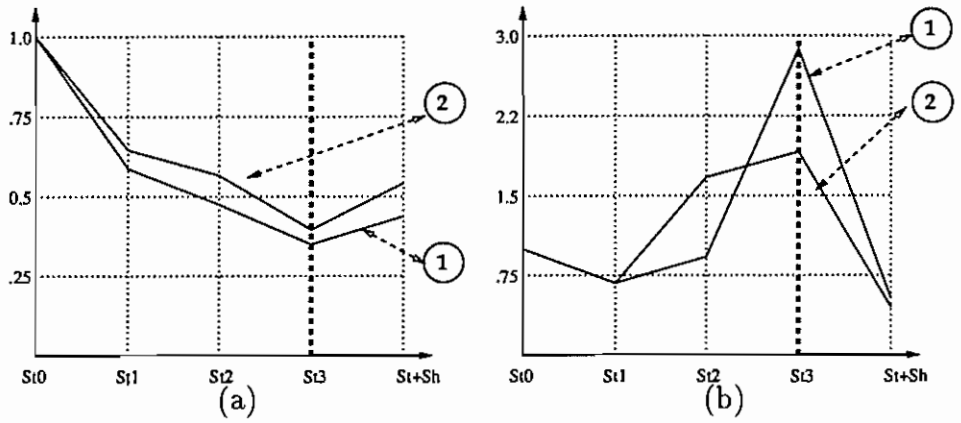


Figure 11: Values of the stereo (a) and shading (b) components of the objective function for the face images. The y axis represents the value of the components, and the x axis represents the various stages of the optimization. From left to right, we first use only stereo and decrease the smoothness and, to the right of the thick dotted line, we turn on the shading term. Each curve is labeled with the number of the corresponding image triplet, and all values have been scaled so that the initial ones are equal to 1.0.

function given the surface optimized using only the stereo component. The main features of both faces—nose, mouth, and eyes—have been correctly recovered. The improvement is particularly striking in the case of the face in Figure 10. The shading component was able to achieve this result because it uses the monocular information around the specularities. The stereo component cannot take advantage of the information around the specularities because very few points are visible in at least two images simultaneously, and because there is little texture. Of course, the effect of the specularities has not completely disappeared (there is indeed still a small artifact on the nose), but has been outweighed by the surrounding information. A more principled approach to solving this problem would be to explicitly include a specularity term in our shading model.

The graphs of Figure 11 depict the behavior of the stereo and shading components of the objective function for the two triplets. The four values of the scores to the left of the thick dotted line, St_0 to St_3, correspond to the results shown in the top row of Figures 9 and 10. The fifth value, $St + Sh$, corresponds to the final results when shading is turned on. These values have been scaled so that St_0 is equal to one for both triplets. As in the synthetic case, when using stereo alone, the stereo component always improves, but as the recovered surface becomes rougher the shading term degrades dramatically, indicating excessive wrinkling of the surface. However, when we turn on the shading component, the overall results improve

(a) (b) (c)

Figure 12: (a) Values of the c_k coefficients of Equation 7 for the final face result of Figure 9. The stereo term is dominant in areas where albedo changes rapidly such as the mouth and the eyes, and the shading term elsewhere. (b) The contribution of each facet to the stereo term. (c) The contribution of each facet to the shape from shading term.

significantly, even though the stereo component degrades slightly. For both faces, the major differences between the initial and final estimates occur in the nose area. In the original meshes, the nose tends to be oversmoothed resulting in differences of up to 15mm in terms of distance to the camera planes, or approximately six pixels in terms of disparity.

5.2.3 Ground-level Scene

In our final example, shown in Figure 13, we reconstruct a ground-level scene using three triplets of images acquired by the INRIA mobile robot. For each triplet, we have computed a correlation map. We have then used the technique described in (Fua and Sander 1992) to merge the resulting 3–D points and generate a Delaunay triangulation. Because the tops of some of the rocks are sharply slanted, the result is relatively rough and can be refined using our technique. As before, stereo alone with little smoothing yields a surface that is too wrinkly. However, by combining stereo and shape from shading, we compute a surface model in which the rock silhouettes are well defined.

6 Summary and Conclusion

We have presented a surface reconstruction method that uses an object-centered representation (a triangulated mesh) to recover geometry and reflectance properties from multiple images. It allows us to handle self-occlusions while merging information from several view-

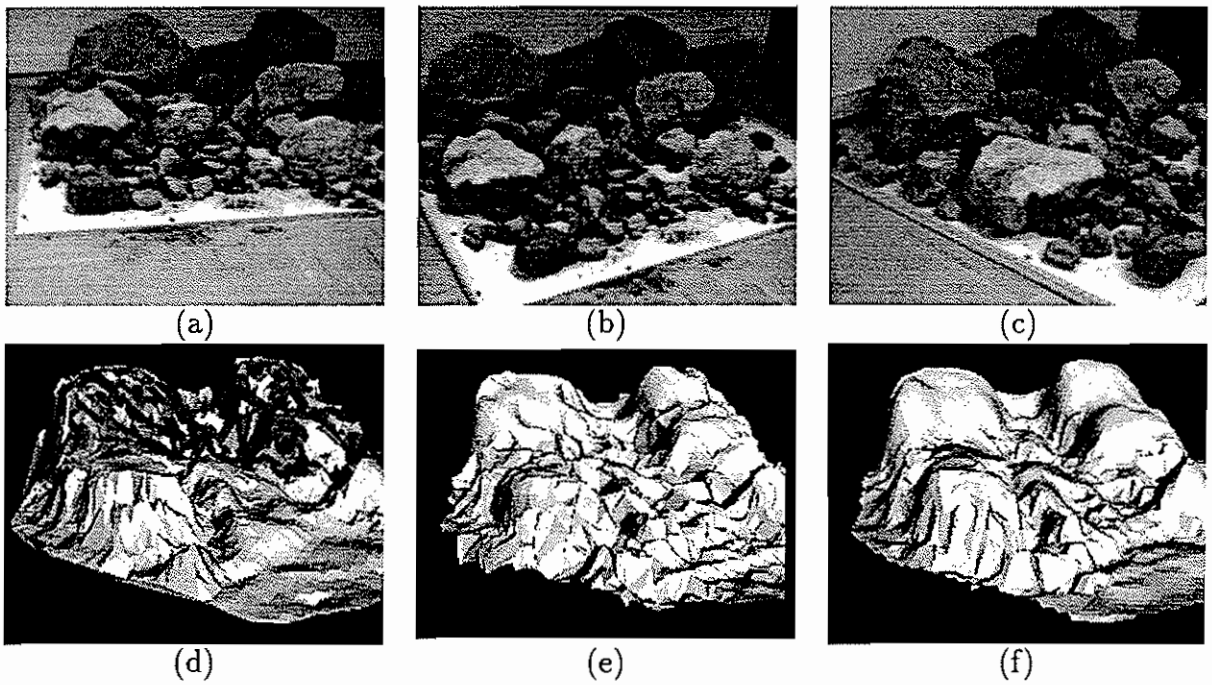


Figure 13: (a,b,c) The first images of three triplets taken by a mobile robot at different locations. (d) Rough ground level surface computed by combining correlation-based results from each triplet. (e) Optimized surface using stereo alone (f) Optimized surface using both stereo and shape from shading. (Courtesy of INRIA)

points, thereby allowing us to eliminate blindspots and make the reconstruction more robust where more than one view is available. The reconstruction process relies on both monocular shading cues and stereoscopic cues. We use these cues to drive an optimization procedure that takes advantage of their respective strengths while eliminating some of their weaknesses.

Specifically, 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. The stereo component compares 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 a hidden-surface algorithm. 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.

On the other hand, shading information is mostly helpful in textureless areas. Thus, we weight the shading component most strongly for facets that project into textureless areas.

For utilizing shading information, the component uses a new method that does not invoke the traditional assumption of constant albedo. Instead, it attempts 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. However, it does require the boundary conditions that are provided by the stereo information.

We have developed a weighting scheme that allows our system to use each source of information where it is most appropriate. As a result, for the large class of surfaces that roughly satisfy the Lambertian model, it performs significantly better than if it were using either source of information alone.

Here we have concentrated on Lambertian surfaces, where the image intensity of a surface point is independent of the direction from which it is viewed. Consequently, we have been able to directly use the intensity in both the stereo component of our algorithm, by using image-intensity correlation, and in the shape-from-shading component, by averaging the intensity at one surface point across all of the input images. Non-Lambertian surfaces cannot be dealt with in this manner.

In future work, we wish to extend our algorithm to the class of non-Lambertian surfaces for which it is possible to unambiguously compute the parameter(s) of the surface reflectance function given the image intensity, viewing direction, light-source direction, and surface normals (all of which are available during our optimization procedure, either directly or from the current estimate of the surface). An example of such a reflectance function is the model proposed by Oren and Nayar (1993) for known values of the surface roughness parameter. For this class of surfaces, we can use the estimated parameters to compute our objective function. For example, we can replace our intensity-correlation term by the variance of the estimated parameters across the input images at one surface point, and our shape-from-shading term by the average of these parameters across the input images at one surface point. This approach has the added advantage that it can be used in situations where the light-source direction is different for each image (as is often the case with aerial images).

Also in future work, we intend to investigate more complex topologies than the ones shown here, multiple resolutions and the shrinking or growing of the surface of interest. We have concentrated so far on a better understanding of the behavior of the objective function, but we believe that our approach can naturally support these extensions.

Acknowledgments

Support for this research was provided by various contracts from the Advanced Research Projects Agency. We wish to thank Hervé Matthieu and Olivier Monga, 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 thank the reviewers whose constructive criticisms have helped us improve the readability of the paper. Finally, we

would like to apologize to the members of the INRIA ROBOTVIS project, whose faces we have mercilessly deformed during the development of the algorithms discussed above.

References

Abbot, A. L. and Ahuja, N. (1990). Active surface reconstruction by integrating focus, vergence, stereo, and camera calibration. In *International Conference on Computer Vision*, pages 489–492.

Aloimonos, J. Y. (1989). Unification and integration of visual modules: an extension of the Marr paradigm. In *ARPA Image Understanding Workshop*, pages 507–551.

Asada, M., Kimura, M., Taniguchi, Y., and Shirai, Y. (1992). Dynamic integration of height maps into a 3D world representation from range image sequences. *International Journal of Computer Vision, 9(1)*, 31–54.

Baltsavias, E. P. (1991). *Multiphoto Geometrically Constrained Matching.* Ph.D. thesis, Institute for Geodesy and Photgrammetry, ETH Zurich.

Barnard, S. (1989). Stochastic stereo matching over scale. *International Journal of Computer Vision, 3(1)*, 17–32.

Barrow, H. G. and Tenenbaum, J. M. (1978). Recovering intrinsic scene characteristics from images. In *Computer Vision Systems*, pages 3–26, Academic Press, New York, New York.

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

Choe, Y. and Kashyap, R. L. (1991). 3-d shape from a shaded and textural surface image. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*, 907–919.

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

Cryer, J. E., Tsai, P.-S., and Shah, M. (1992). *Combining shape from shading and stereo using human vision model.* Technical Report CS-TR-92-25, U. Central Florida.

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

Diehl, H. and Heipke, C. (1992). Surface reconstruction from data of digital line cameras by means of object based image matching. In *International Society for Photogrammetry and Remote Sensing*, pages 287–294, Washington D.C.

Faugeras, O. and Toscani, G. (1986). The calibration problem for stereo. In *Conference on Computer Vision and Pattern Recognition*, pages 15–20, Miami Beach, Florida.

Ferrie, F. P., Lagarde, J., and Whaite, P. (1992). Recovery of volumetric object descriptions from laser rangefinder images. In *European Conference on Computer Vision*, Genoa, Italy.

Fua, P. (1993). A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications, 6(1)*. Available as INRIA research report 1369.

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

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

Grimson, W. E. L. and Huttenlocher, D. P. (1992). Introduction to the special issue on interpretation of 3-d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2)*, 97–98.

Hannah, M. J. (1989). A system for digital stereo image matching. *Photogrammetric Engineering and Remote Sensing, 55(12)*, 1765–1770.

Hartt, K. and Carlotto, M. (1989). A method for shape-from-shading using multiple images acquired under different viewing and lighting conditions. In *Conference on Computer Vision and Pattern Recognition*, pages 53–60.

Heipke, C. (1992). 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.

Horn, B. K. P. (1990). Height and gradient from shading. *International Journal of Computer Vision, 5(1)*, 37–75.

Hung, Y., Cooper, D. B., and Cernuschi-Frias, B. (1991). Asymptotic bayesian surface estimation using an image sequence. *International Journal of Computer Vision, 6(2)*, 105–132.

Kaiser, B., Schmolla, M., and Wrobel, B. P. (1992). Application of image pyramid for surface reconstruction with fast vision. In *International Society for Photogrammetry and Remote Sensing*, page 1, Washington, D.C.

Kanade, T. and Okutomi, M. (1990). A stereo matching algorithm with an adaptative window: Theory and experiment. In *ARPA Image Understanding Workshop.*

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

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

Leclerc, Y. G. (1989b). *The Local Structure of Image Intensity Discontinuities.* Ph.D. thesis, McGill University, Montréal, Québec, Canada.

Leclerc, Y. G. and Bobick, A. F. (1991). The direct computation of height from shading. In *Conference on Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii.

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

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

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

Marr, D. (1982). *Vision.* W. H. Freeman, San Francisco, California.

Nishihara, H. (1984). Practical real-time imaging stereo matcher. *Optical Engineering, 23(5).*

Okutomi, M. and Kanade, T. (1991). A multiple-baseline stereo. In *Computer Vision and Pattern Recognition 91*, pages 63–69, Maui, Hawaii.

Oren, M. and Nayar, S. (1993). Generalization of the lambertian model. In *ARPA Image Understanding Workshop*, pages 1037–1048.

Panton, D. J. (1978). A flexible approach to digital stereo mapping. *Photogramm. Eng. Remote Sensing, 44(12)*, 1499–1512.

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

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

Poggio, T., Torre, V., and Koch, C. (1985). Computational vision and regularization theory. *Nature, 317*.

Press, W., Flannery, B., Teukolsky, S., and Vetterling, W. (1986). *Numerical Recipes, the Art of Scientific Computing.* Cambridge U. Press, Cambridge, MA.

Quam, L. (1984). Hierarchical warp stereo. In *ARPA Image Understanding Workshop*, pages 149–155.

Stokely, E. M. and Wu, S. Y. (1992). 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.

Szeliski, R. (1991). Shape from rotation. In *Conference on Computer Vision and Pattern Recognition*, pages 625–630.

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

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

Terzopoulos, D. (1988). The computation of visible-surface representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, , 417–438.

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

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

Terzopoulos, D., Witkin, A., and Kass, M. (1987). Symmetry-seeking models and 3D object reconstruction. *International Journal of Computer Vision, 1*, 211–221.

Tomasi, C. and Kanade, T. (1992). The factorization method for the recovery of shape and motion from image streams. In *ARPA Image Understanding Workshop*, pages 459–472.

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

Wang, Y. F. and Wang, J. F. (1992). Surface reconstruction using deformable models with interior and boundary constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(5)*, 572–579.

Whaite, P. and Ferrie, F. P. (1991). From uncertainty to visual exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(1038-1049).*

Witkin, A. W., Terzopoulos, D., and Kass, M. (1987). Signal matching through scale space. *International Journal of Computer Vision, 1*, 133–144.

Wrobel, B. P. (1991). The evolution of digital photgrammetry from analytical phogrammetry. *Photogrammetric Record, 13(77)*, 765–776.