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. To recover complicated surfaces, single information sources often prove insufficient. In this paper, we present a unified framework for 3–D shape reconstruction that allows us to combine image-based constraints, such as those deriving from stereo and shape-from-shading, with geometry-based ones, provided here in the form of 3–D points, 3–D features or 2–D silhouettes.
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.

1 Introduction

The problem of recovering 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, 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, including images taken from widely differing viewpoints. It accomodates 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 used to generate synthetic views of the estimated surface from the viewpoint of each input image. Using standard computer graphics algorithms, those parts of the surface that are hidden from a given viewpoint can be identified and eliminated from the reconstruction process. The remaining parts are then in correspondence with the input images. The corresponding cues are applied in an iterative manner using an optimization algorithm.

Lecture Notes in Computer Science, Vol. 801
Jan-Olof Eklundh (Ed.)
Computer Vision – ECCV '94

In many recent publications about surface reconstruction, such as [Delingette *et al.*, 1991, Terzopoulos and Vasilescu, 1991, Szeliski and Tonnesen, 1992], the authors fit a surface to previously computed 3–D data, such as the output laser range finders or correlation-based stereo algorithms. In other words, the derivation of the 3–D data from the images is completely divorced from the surface reconstruction. 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. In a previous publication [Fua and Leclerc, 1993] 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 the incoporation of additional image cues, silhouette edges and previously computed 3–D data.

Combining these different sources of information is not a new idea in itself. For example, Blake *et al.* [1985] discuss 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] use silhouettes to derive an initial estimate of the surface and improve the result using multi-image stereo. 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. We then demonstrate that the framework successfully performs its function on real images and allows us to achieve results 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 triangulation representing the surface, and the process is started with an initial surface estimate. Here we assume that images are monochrome, and that their camera models are known *a priori*.

We represent a surface S by a hexagonally connected set of vertices called a *mesh*. Such a mesh is shown in Figure 1(a). The position of a vertex v_j is specified by its Cartesian coordinates (x_j, y_j, z_j).

For each input image, we generate a "Facet-ID" image 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

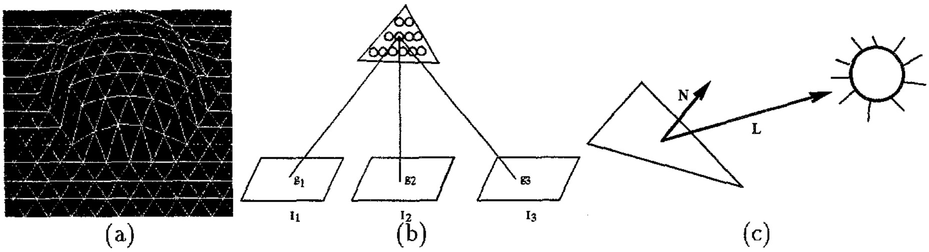

Fig. 1. Mesh representation and computation of the image terms of the objective function: (a) Wireframe representation of the a mesh. (b) Facets are sampled at regular intervals, the circles represent the sample points. The stereo component of the objective function is computed by summing the variance of the grey level of the projections of these sample points, the g_is. (c) Each facet's albedo is estimated using its normal N, the light source direction 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.

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, 1989] in which we add a regularization term to the objective function and progressively reduce its influence. We define the total energy of the mesh, $\mathcal{E}_T(\mathcal{S})$, as

$$\mathcal{E}_T(\mathcal{S}) = \lambda_D \mathcal{E}_D(\mathcal{S}) + \mathcal{E}(\mathcal{S}) \text{ with } \mathcal{E}_(\mathcal{S}) = \sum_i \lambda_i \mathcal{E}_i(\mathcal{S}) \ . \tag{1}$$

The $\mathcal{E}_i(\mathcal{S})$ represent the image and geometry-based constraints discussed below, and the λ_i their relative weights. $\mathcal{E}_D(\mathcal{S})$, 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, in the presence of noise, some amount of smoothing is required to prevent the mesh from overfitting the data, and wrinkling the surface excessively [Fua and Leclerc, 1993].

In our implementation, we take $\mathcal{E}_D$ to be a measure of the curvature or local deviation from a plane at every vertex. Using finite differences, $\mathcal{E}_D$ can be expressed as a quadratic form [Fua and Leclerc, 1993]

$$\mathcal{E}_D(\mathcal{S}) = 1/2(X^T K X + Y^T K Y + Z^T K Z) \ , \tag{2}$$

where X,Y, and Z are the vectors of the x,y and z coordinates of the vertices, and K is a sparse and banded matrix.

Because $\mathcal{E}_D$ is quadratic and decouples the three spatial coordinates, our energy term is amenable to a "snake-like" optimization technique [Kass *et al.*, 1988]. We treat S as a physical surface embedded in a viscous medium and evolving under the influence of the potential $\mathcal{E}_T$. We solve the minimization problem by solving the dynamics equation of this system. We can either optimize the three spatial components, X, Y and Z simultaneously or separately.

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, as a result, 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. Since the dynamics of the optimization are controlled by the gradient of the objective function, an effective way to normalize the contributions is to introduce a set of weights λ'_i such that $\lambda'_D = 1 - \sum_{1 \leq i \leq n} \lambda'_i > 0$. The λs are taken to be

$$\lambda_i = \frac{\lambda'_i}{\| \vec{\nabla} \mathcal{E}_i(\mathcal{S}^0) \|} \, , \; \lambda_D = \frac{\lambda'_D}{\| \vec{\nabla} \mathcal{E}_D(\mathcal{S}^0) \|} \, , \qquad (3)$$

where $\mathcal{S}^0$ is the surface estimate at the start of each optimization step. 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. The continuation method of Section 2.1 is implemented by first taking λ'_D to be 0.5 and then reducing it while keeping the relative values of the λ'_is constant.

2.3 Geometric Constraints

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

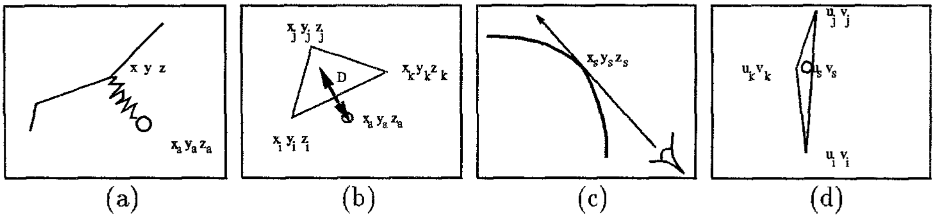

Fig. 2. 3–D and 2–D point constraints: (a) Point attractor modeled as a spring attached to a vertex. (b) Point attractor modeled as a spring attached to the closest surface point. (c) Occlusion contours are the locus of the projections of the (x_s, y_s, z_s) surface points for which a camera ray is tangential 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.

3–D Points They 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

$$e_a = 1/2((x_a - x_i)^2 + (y_a - y_i)^2 + (z_a - z_i)^2) \tag{4}$$

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 4 must be replaced by one that attracts the surface without warping it. In our implementation, this is achieved by redefining e_a as

$$e_a = 1/2d_a^2 \tag{5}$$

where d_a is the orthogonal distance of the attractor to the closest facet. It is easy to show that d_a^2 can be expressed as the ratio of two second order polynomial in terms of the vertex coordinates. These two sorts of attractors are depicted in Figure 2 (a,b). The search for the "closest facet" is made efficient and fast by assuming that the attractors can be identified by their projection in an image. We project the mesh into 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 images, this method can be used to fuse 3–D data from different sources.

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 2(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 2(d). This is enforced by adding, for each silhouette point, a term of the form

$$e_s = 1/2 \sum_{1 \leq i \leq 3, i < j \leq 3} \begin{vmatrix} u_i & u_j & u_s \\ v_i & v_j & v_s \\ 1 & 1 & 1 \end{vmatrix}^2 , \tag{6}$$

where the (u_i, v_i)s are the projections of the (x_i, y_i, z_i) using the camera model. This term constrains the determinants to be small and, therefore, the projections of the vertices and the silhouette point to be collinear.

As with the 3–D attractors, 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. As shown in Figure 1(b), this comparison is done for a uniform sampling of the surface. This method allows us to deal with arbitrarily slanted regions and to discount occluded areas of the surface.

The shading component of the objective function is computed using a method that does not invoke the traditional constant albedo assumption. Instead, it attempts to minimize the variation in albedo across the surface, and can therefore deal with surfaces whose albedo varies slowly. This term is depicted by Figure 1(c).

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

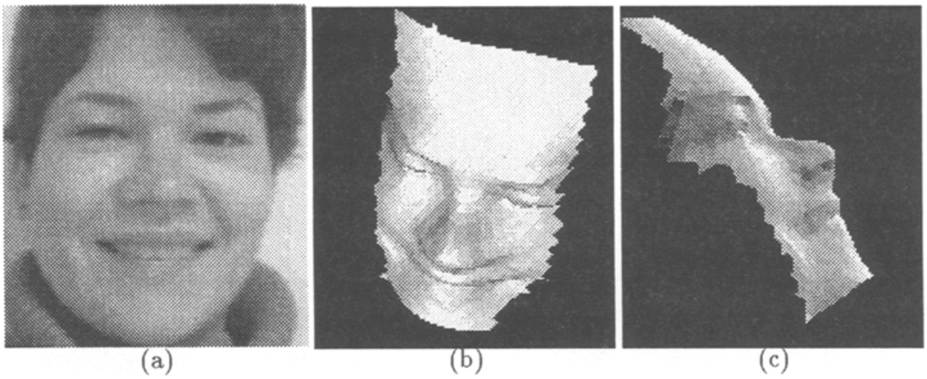

Fig. 3. Combining stereo and shape-from-shading: (a) First image of a triplet (courtesy of INRIA). (b,c) Shaded views of the reconstructed surface.

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, 1993]. In Figure 3, we show the reconstruction of a face using only stereo and shape-from-shading.

3 Applications

Our framework lets us combine geometric constraints with image-based constraints either to derive surface reconstructions or to refine previously computed surfaces. We now demonstrate its capabilities using difficult imagery.

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 edge-features, in the same fashion. They are sampled at regular intervals to generate collections of 3–D attractors or 2–D silhouette points.

Dense 3–D Data In Figure 4, we show an image of a face and a corresponding range map computed using structured light. Although fairly accurate, this particular method introduces artifacts in the range image. As a result, fitting a surface to this data by treating the range points as attractors yields the excessively wrinkly result shown in Figure 4(c). Simply smoothing would lose 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 4(a) by taking the objective function to be a weighted sum of the term that attracts the surface towards the range data and of the the shading term. The result, shown in Figure 4(d,e,f), is much smoother, but the mouth is well

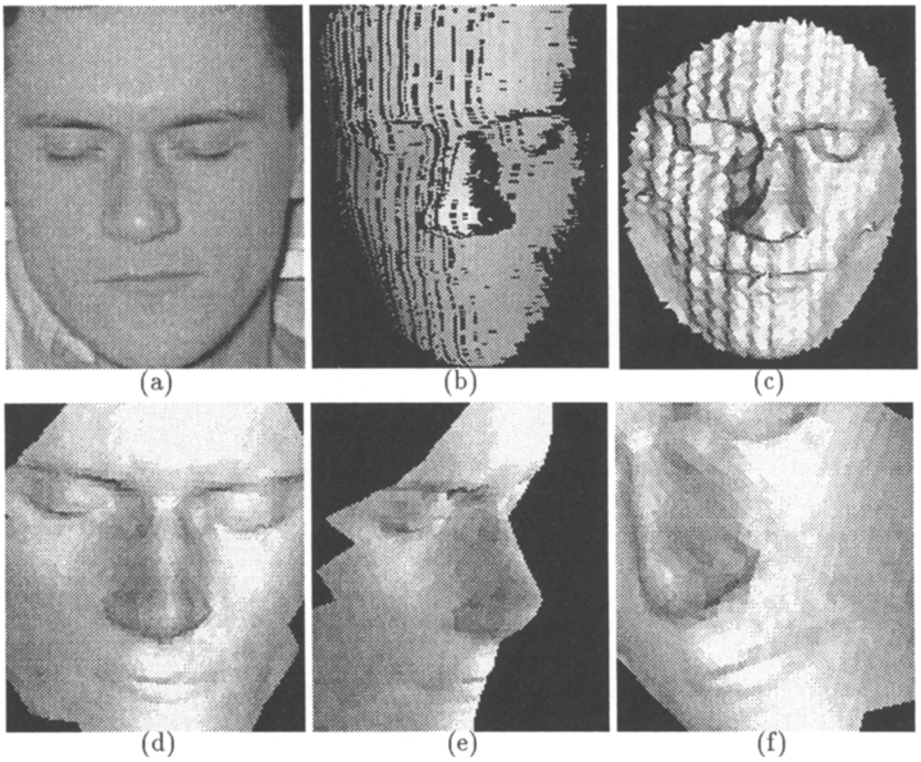

Fig. 4. Combining range data with shape-from-shading information: (a) Image of a face (Courtesy of ETH Zürich). (b) Corresponding range image computed using structured light. (c) Shaded views of the surface reconstructed by using the range-data points as attractors. (d)(e)(f) Shaded views of the reconstruction refined using shading.

preserved and the side of the nose better defined. Note, however, that in the side view 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 below.

Sparse 3–D Data We now turn to sparse 3–D data. In Figure 5, we show a stereo pair of a rock outcrop forming an almost vertical cliff. Correlation-based algorithms typically fail in the cliff area. To demonstrate the data-fusion capabilities of our approach, we have used the 3D–snakes embedded in the SRI Cartographic Modeling Environment to supply 3–D edges whose projections are shown in Figure 5(c,d).

We first attract an initially flat surface to both the output of a simple correlation-based algorithm—it yields information only in the flat parts of the scene—and the 3–D outlines and produce the roughly correct but excessively smooth estimate of Figure 5 (e). By adding either the stereo term alone to $\mathcal{E}_T$, Figure 5 (f), or both the stereo and shading terms, Figure 5 (g,h), we can generate a much more realistic model of the surface.

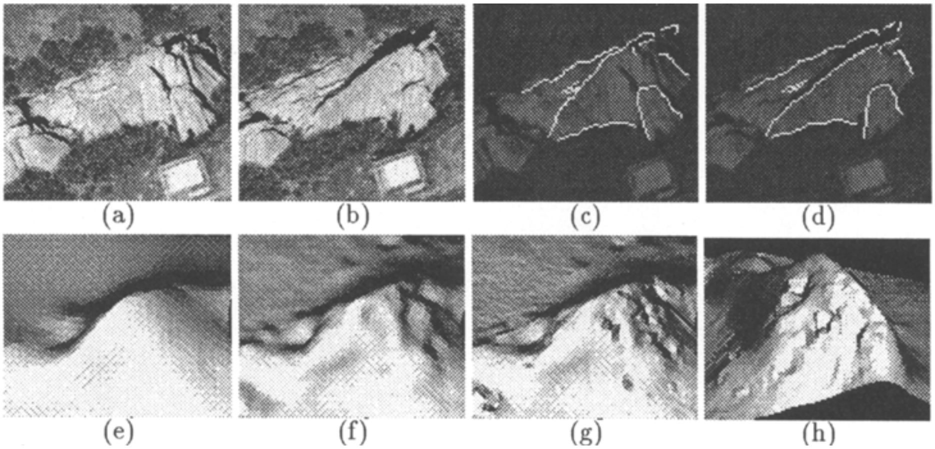

Fig. 5. Semiautomated cartography of a rugged site: (a,b) A hard-to-fuse stereo pair of a rock outcrop with an almost vertical cliff. (c,d) The projections of a few 3–D features outlined using 3–D snakes. (e) Reconstructed surface using the 3–D features as attractors. (f) Refinement using stereo. (g,h) Refinement using both stereo and shape from shading.

Silhouettes Very few vision algorithms consistently provide a perfect answer across scenes using a predetermined set of information sources and analysis parameters. It is often important to be able to easily refine a previously derived result, and silhouettes are very effective for this purpose. For example, the reconstruction of the bottom of the nose in Figure 4(e) is not quite right, as can be seen in Figure 6(b). To correct this, we use the silhouettes of Figure 6(a,b) that have been outlined using 2–D snakes. We take the total energy $\mathcal{E}_T$ to be a weighted sum of the silhouette attraction terms of Equation 5 and of the shading term of Section 2.4. We use these terms to deform the nose region and generate the improved result of Figure 6(c).

The face reconstruction of Figure 3 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 and easy to outline, as shown in Figure 6(d). To take advantage of these, we start with a larger and coarser mesh that evolves under the influence of the silhouettes and the vertices of the original 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 including stereo and shape-from-shading. The results are shown in Figure 6(e,f).

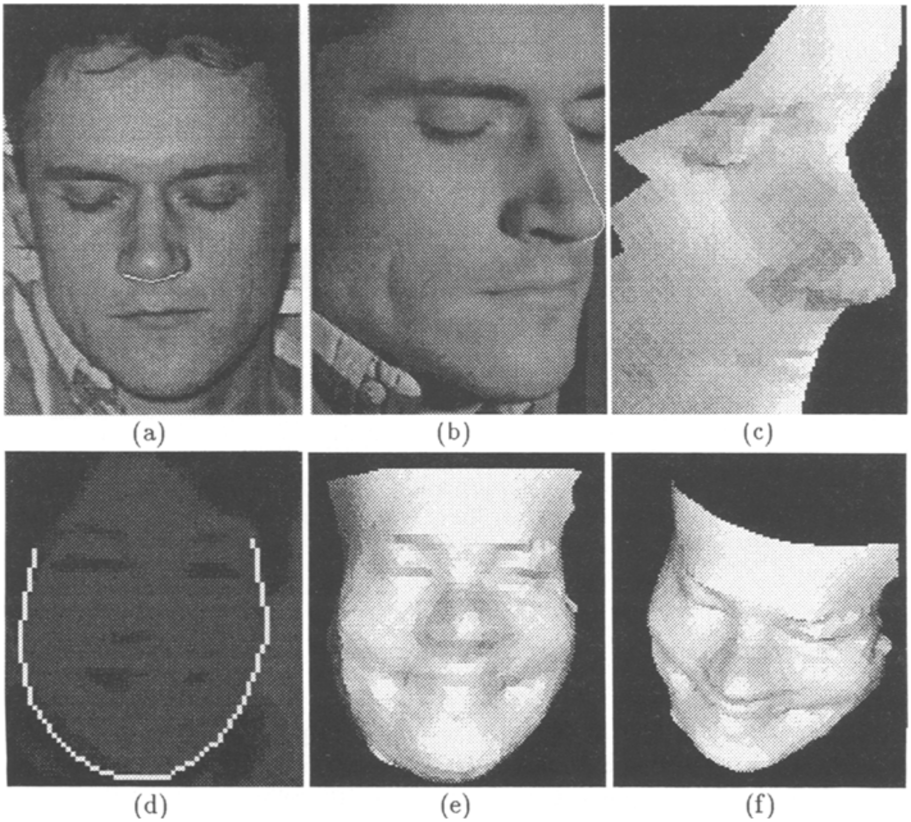

Fig. 6. Using silhouettes to improve a reconstruction: (a) The face of Figure with a silhouette at the bottom of the nose outlined. (b) A side view of the same face with a second nose silhouette. (c) Shaded views of the refined reconstruction using both shading and the two silhouettes. (d) Face silhouette outlined in the first image of the triplet of Figure . (e,f) Shaded views of the reconstructed surface after optimization using stereo, shading, and the constraints provided by the silhouettes.

4 Summary and 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.

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 ones. User assistance is provided primarily through the introduction of a small number of hand-entered linear and point features using semi-automated "snake" technology. The method is controlled by a small number of image-independent parameters that specify the relative importance of the various information sources.

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.

Acknowledgments

Support for this research was provided by various contracts from the Advanced Research Projects Agency. We wish to thank O. Kübler, M. Trobina, H. Matthieu, O. Monga, from ETH Zurich and INRIA who have provided us not only with the face images and corresponding calibration data but also with valuable advice.

References

[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.

[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.

[Fua and Leclerc, 1993] P. Fua and Y.G. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. In *ARPA Image Understanding Workshop*, Washington, D.C., April 1993.

[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.

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

[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 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.