International Journal of Computer Vision 31(1), 31–50 (1999)

Structure and Motion Estimation from Dynamic Silhouettes under Perspective Projection

TANUJA JOSHI*
Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, USA

NARENDRA AHUJA AND JEAN PONCE
Beckman Institute, University of Illinois, Urbana, IL 61801, USA

Received April 11, 1996; Revised September 25, 1998

Abstract. We address the problem of estimating the structure and motion of a smooth curved object from its silhouettes observed over time by a trinocular stereo rig under perspective projection. We first construct a model for the local structure along the silhouette for each frame in the temporal sequence. The local models are then integrated into a global surface description by estimating the motion between successive time instants. The algorithm tracks certain surface features (parabolic points) and image features (silhouette inflections and frontier points) which are used to bootstrap the motion estimation process. The entire silhouettes along with the reconstructed local structure are then used to refine the initial motion estimate. We have implemented the proposed approach and report results on real images.

Keywords: motion estimation, 3D structure estimation, silhouettes, occluding contours, trinocular stereo, frontier points, epipolar curves

1. Introduction

As an object moves with respect to a camera observing it, the deformation of its silhouette over time reflects the characteristics of its shape and motion. For objects with little or no surface detail such as surface markings or texture, silhouettes are indeed the most important cues for the estimation of object structure and motion, and several methods have been proposed for structure estimation from silhouettes under *known* camera motion (Blake and Cipolla, 1992; Boyer and Berger, 1997; Giblin and Weiss, 1987; Kutulakos and Dyer, 1994; Szeliski and Weiss, 1993; Vaillant and Faugeras, 1992; Zheng, 1994). These approaches have demonstrated that given a set of three or more nearby views of a smooth object, the structure of the object up to second order can be obtained along its occluding contour.

*This work was performed while Tanuja Joshi was with the Beckman Institute, University of Illinois, Urbana, IL 61801, USA.

Here, we address the problem of estimating *both the structure and motion* of a smooth object from its silhouettes observed over time by a trinocular stereo rig (see Cipolla et al., 1995; Giblin et al., 1994 for recent approaches in the monocular case). The proposed approach will be useful in a situation in which the viewer is a passive observer and has no knowledge or control of the motion of the object in the scene (Kutulakos and Dyer (1994) have developed a complementary approach, in which a viewer plans its motion for building a global model of an object). The main application of our technique is in automatic model construction, e.g., for recognition, rendering, or virtual reality.

1.1. Problem Statement and Approach

Consider a smooth curved object and a camera observing it. The viewing cone grazes the object along the *occluding contour* and intersects the image plane along the *silhouette* of the object. Since, by definition,

the surface normal is orthogonal to the corresponding viewing ray at each point of the occluding contour, both the 3D occluding contour and the 2D silhouette depend on the relative positions of the camera and the object. In particular, when the camera moves relative to the object, the silhouettes appearing in successive images will be the projections of different 3D contours on the surface. In contrast, although the projections of viewpoint-independent object features such as corners or surface creases appear to move in the image as the observing camera moves in space, they remain the projections of static 3D points.

In this paper, we address the problem of estimating the structure and motion of a smooth curved object from its silhouettes observed over time by a trinocular stereo rig under perspective projection. To relate the successive silhouettes, a model for the local structure is constructed that can be used to estimate the motion. We use trinocular imagery for our analysis, since the three images can be used to recover the model parameters of the local structure (up to second order). As noted by others (Blake and Cipolla, 1992; Vaillant and Faugeras, 1992), trinocular imagery is possibly beneficial in one more way: three frames are sufficient to differentiate between viewpoint-dependent and viewpoint-independent edges in the image. Thus, the input of our algorithm consists of a sequence of triples of images taken by a trinocular stereo rig over time (Fig. 1). The local structure is estimated using the three silhouettes observed at each time instant.

Note that we need to establish a relationship or correspondence between points on pairs of silhouettes. As noted earlier, the 3D occluding contour changes as the object or its observer moves in space, therefore there is no true 3D point-to-point correspondence between successive silhouettes. For the triple of images observed at a given time by the three cameras, we take advantage of the known epipolar geometry to establish correspondences between distinct 3D points lying on a common epipolar curve and then estimate local structure parameters as explained in Section 2.

Establishing correspondences between images taken by a single camera at successive time instants is more complicated since the epipolar geometry is unknown in this case: unless we know the correspondences we cannot estimate the motion, but we need to know the motion, or at least the epipolar geometry, to establish the correspondences for all the points on the silhouette. To bootstrap the matching and motion estimation processes, we first use detectable and trackable silhouette features—namely *inflections* and *frontier points*—to construct an initial estimate of the unknown motion. In turn, this enables us to estimate the epipolar geometry, thus allowing the identification of correspondences for the rest of the silhouette. We then iteratively refine the motion estimate, updating the correspondences for all the silhouette points at each iteration.

The rest of this paper is organized as follows: Section 2 presents an algorithm for structure estimation using trinocular imagery. The algorithm used for motion estimation from dynamic silhouettes under perspective projection is discussed in Section 3. To demonstrate the feasibility of our method, experimental results using sequences of real images are presented throughout. We conclude with comments in Section 4.

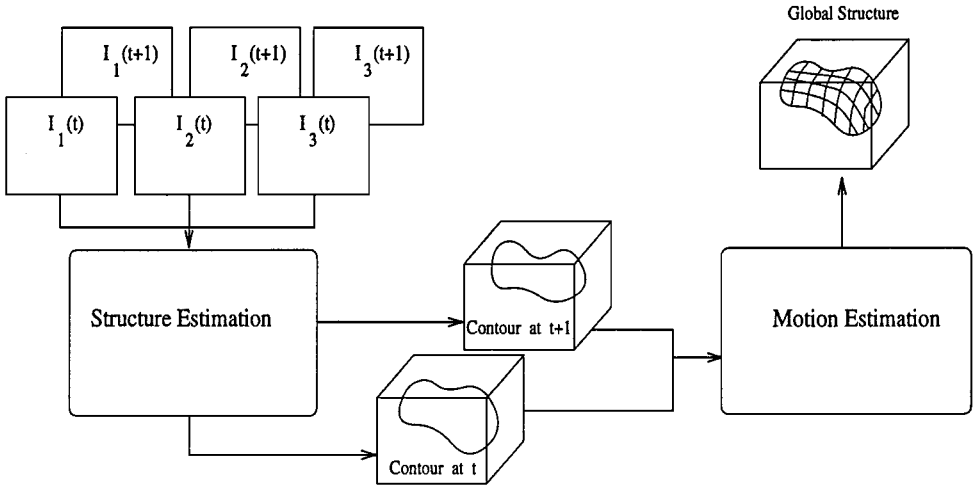

Figure 1. Block diagram of the approach.

2. Structure Estimation Using Trinocular Imagery

2.1. Previous Work

The study of the quantitative relationship between the shape of a surface and its silhouette(s) was pioneered by Koenderink (1984) and Giblin and Weiss (1987). Koenderink (1984) investigated the shape information contained in a *single* silhouette and proved that the sign of the observed surface's Gaussian curvature at an occluding contour point is the same as the sign of the silhouette's curvature. Giblin and Weiss (1987) were the first to exploit *multiple* silhouettes to reconstruct the shape of a smooth object. Their approach is based on the fact that a smooth surface (excluding the concavities that are never visible on the silhouettes) is the envelope of its tangent planes. Assuming that the viewing directions correspond to a great circle of directions, they reduced the problem of analyzing the envelope of tangent planes to the simpler problem of computing the envelope of tangent lines in a plane. They reconstructed the object surface by obtaining the depth, the Gaussian curvature, and the mean curvature along the occluding contour.

Since this early work, several methods have been proposed for structure estimation using silhouettes under general known camera motion (Blake and Cipolla, 1992; Boyer and Berger, 1997; Giblin and Weiss, 1987; Szeliski and Weiss, 1993; Vaillant and Faugeras, 1992; Zheng, 1994). They have demonstrated the feasibility of determining the structure of a smooth object up to second order along its occluding contour from a set of three or more nearby views. Blake and Cipolla (1992) have presented a differential formulation for the case of a continuous sequence of images taken under general known motion. Vaillant and Faugeras (1992) have shown how a triple of silhouettes can be used to reconstruct a local second-order model of the surface along the occluding contour. Their algorithm projects three matching viewing rays onto the *radial plane* (i.e., the plane containing the central ray and the surface normal), then estimates the curvature of the *radial curve* (i.e., the intersection of the radial plane with the surface) by constructing a circle tangent to the three projected rays. More recently, Szeliski and Weiss (1993) have used the epipolar plane instead of the radial plane for the estimation of one of the surface curvatures. For the case of multiple frames, they have applied tools from estimation theory (Kalman filtering and smoothing) to make optimal use of each measurement. Although their method is quite robust, it is exact only for the case of linear camera motion. Recently, Boyer and Berger (1997) have developed an algorithm that is applicable to the general motion case without having to make any approximations. This method also builds a second-order local model of the surface, but it relies on an exact depth computation formulation.

The approaches to shape reconstruction from silhouettes discussed so far address the problem of constructing *local* surface models along the occluding contour. In contrast, Zheng (1994) has studied the reconstruction of *global* 3D models from silhouettes. His algorithm takes as input a continuous sequence of images taken with pure rotation about a fixed axis, and analyzes the spatio-temporal volume obtained from the image sequence. Occluding contours are differentiated from fixed edges by exploiting the fact that the trace of an occluding contour in the spatio-temporal volume is different from the trace of a fixed 3D edge. Concave parts of the model can also be identified. One of the main applications considered is the construction of human face models. In related work, Seales and Faugeras (1994) have shown how to integrate the local models of Vaillant and Faugeras (1992) into global surface models, and Zhao and Mohr (1994) have developed a method for constructing B-spline surface models from a sequence of silhouettes.

We will next describe our structure estimation algorithm. It is related to the approaches of Vaillant and Faugeras (1992) and Szeliski and Weiss (1993), and it constructs a local paraboloid model of the surface along the occluding contour from a triple of silhouettes. The corresponding equations are derived in Section 2.2. We obtain the parameters of this model at each point by first finding correspondences among the three frames of the trinocular rig, as described in Section 2.3. Section 2.4 presents our method for estimating the model parameters. Experimental results on synthetic as well as real images are reported in Section 2.5. We discuss possible ways to distinguish silhouettes from viewpoint-independent edges in Section 2.6.

2.2. Modeling the Local Structure

The local structure (up to second order) at a surface point P is defined by the 3D location of P in some global coordinate frame (in our case, the coordinate frame of the central camera), the surface normal

at P, the two principal directions in the tangent plane and the principal curvatures. At each point P, we define a local coordinate frame (X_l, Y_l, Z_l) whose origin coincides with P, the X_l-axis being aligned with the outward normal, and the Y_l and Z_l-axes being aligned with the principal directions. The local surface up to second order is a paraboloid, given by the equation

$$X_l = \frac{1}{2}\kappa_1 Y_l^2 + \frac{1}{2}\kappa_2 Z_l^2, \tag{1}$$

where κ_1 and κ_2 are the principal curvatures at P. The signs of κ_1 and κ_2 define the point type: if κ_1 and κ_2 have the same sign (resp. opposite signs), P is an elliptic (resp. hyperbolic) point. The elliptic points project onto convex silhouette points while the hyperbolic points project onto concave ones. If either κ_1 or κ_2 is zero, P is a parabolic point and the silhouette has an inflection (Vaillant and Faugeras, 1992; Koenderink, 1984; Blake and Cipolla, 1990). Equation (1) can be rewritten in matrix form as:

$$\boldsymbol{Q}_l^T \boldsymbol{M}_l \boldsymbol{Q}_l = 0, \tag{2}$$

where $\boldsymbol{M}_l$ is a symmetric 4×4 matrix and $\boldsymbol{Q}_l = [X_l, Y_l, Z_l, 1]^T$ is the vector of homogeneous coordinates of a point Q on the paraboloid at P.[1]

The rigid transformation parameters defining the local coordinate frame at P with respect to the coordinate frame of the central camera, together with the principal curvatures κ_1 and κ_2, completely determine the local structure at P. Let (X, Y, Z) be the camera-centered coordinate frame, where the Z-axis coincides with the optical axis and the XY-plane is the image plane (Fig. 2). Let the point P (which is the origin of the local frame) be at (X_0, Y_0, Z_0) in the camera-centered frame.

We denote the angle made by X_l axis (the surface normal, $\boldsymbol{N}_P$) with the normal to the silhouette ($\boldsymbol{n}_p$) by ζ as shown in Fig. 2.[2] We denote the angle between the X axis and the normal to the silhouette by θ and the angle between the viewing direction and one of the principal directions (say the Z_l axis) by γ.

The six-tuple $(\theta, \gamma, \zeta, X_0, Y_0, Z_0)$ defines the local frame with respect to the camera-centered frame. To completely describe the object surface locally, we need to specify $\theta, \gamma, \zeta, X_0, Y_0, Z_0, \kappa_1$ and κ_2 for each point on the silhouette. A point Q in the local coordinate frame defined at point P can be represented in the coordinate frame of the central camera by

$$\boldsymbol{Q} = \boldsymbol{T}_0 \boldsymbol{R}_0 \boldsymbol{Q}_l, \tag{3}$$

where $\boldsymbol{T}_0$ is the 4×4 matrix for a translation by (X_0, Y_0, Z_0), $\boldsymbol{R}_0$ is the 4×4 matrix for the rotation between the two coordinate frames, and $\boldsymbol{Q}_l =$

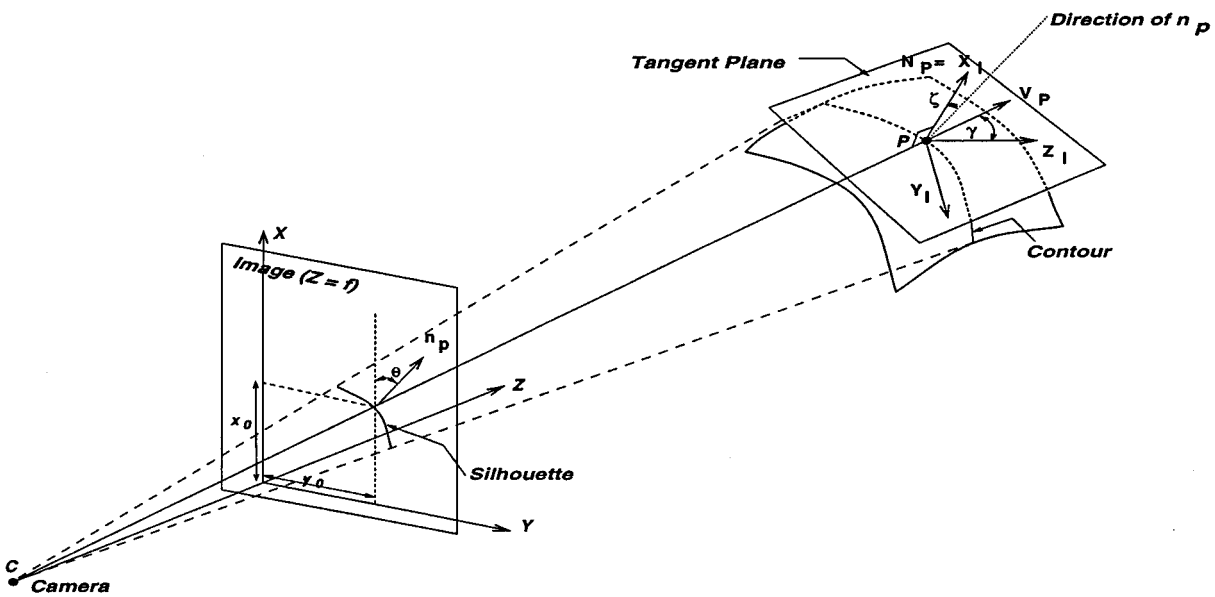

Figure 2. Projection geometry.

$[X_l, Y_l, Z_l, 1]^T$ and $\boldsymbol{Q} = [X, Y, Z, 1]^T$ are the homogeneous coordinate vectors of Q in the local and camera-centered frames respectively. After eliminating $\boldsymbol{Q}_l$ between (2) and (3) we obtain the equation of the paraboloid in the camera-centered frame as:

$$\Sigma \stackrel{\text{def}}{=} \boldsymbol{Q}^T \boldsymbol{M} \boldsymbol{Q} = 0, \tag{4}$$

where

$$\boldsymbol{M} = \boldsymbol{T}_0^{-1T} \boldsymbol{R}_0 \boldsymbol{M}_l \boldsymbol{R}_0^{-1} \boldsymbol{T}_0^{-1}. \tag{5}$$

By definition, the surface normal at the contour point P is orthogonal to the viewing direction. This condition can be expressed as:

$$\boldsymbol{N}_P \cdot \boldsymbol{V}_P = 0, \tag{6}$$

where $\boldsymbol{N}_P = [\frac{\partial \Sigma}{\partial X}, \frac{\partial \Sigma}{\partial Y}, \frac{\partial \Sigma}{\partial Z}]^T$, and $\boldsymbol{V}_P = [X_0, Y_0, Z_0]^T$ (Fig. 2). This is a linear condition in (X, Y, Z), implying that the contour of the paraboloid is a planar curve. Eliminating Z between (4) and (6) gives the equation of the silhouette in the image coordinates, which is a conic section.

The paraboloid model described in this section is used to model the structure at each point along the silhouette. From a single projection of the paraboloid, we can obtain five constraints on the eight structure parameters (θ, γ, ζ, X_0, Y_0, Z_0, κ_1 and κ_2):

- The surface normal $\boldsymbol{N}_P$ at P is orthogonal to the viewing direction $\boldsymbol{V}_P$. Also the tangent $\boldsymbol{t}_p$ to the silhouette at p lies in the tangent plane at P, implying that $\boldsymbol{N}_P$ is orthogonal to $\boldsymbol{t}_p$. Thus knowing $\boldsymbol{t}_P$ and $\boldsymbol{V}_P$ we can compute the surface normal $\boldsymbol{N}_P$ completely:

$$\boldsymbol{N}_P = \boldsymbol{t}_p \times \boldsymbol{V}_P. \tag{7}$$

  This gives the angles θ and ζ directly.
- The image coordinates (x_0, y_0) of p give two constraints on X_0, Y_0 and Z_0:

$$\begin{cases} x_0 = \dfrac{f X_0}{Z_0}, \\ y_0 = \dfrac{f Y_0}{Z_0}, \end{cases} \tag{8}$$

  where f is the focal length.
- The curvature of the silhouette at p gives a constraint on κ_1, κ_2 and γ.

To complete the local structure model, we need to estimate the depth Z_0 and obtain two more constraints on κ_1, κ_2, and γ. As explained in Section 2.4 we obtain the required constraints using the matched points from the other two images of the trinocular imagery.

The estimation of structure at a point relies on the fact that correspondence can be obtained in all the images of the trinocular imagery at a given time. At degenerate situations (such as viewing a hyperbolic patch along asymptotic directions), we cannot find reliable correspondences and hence do not get a reliable structure estimation.

The next section presents a method for establishing the correspondences.

2.3. *Finding Correspondences*

When the relative motion between the object and the camera is known for a pair of images (either in the trinocular stereo rig or in the sequence taken by a camera), we can define an epipolar plane for each point in each image. Let I_1 and I_2 be the two images taken by two cameras with optical centers C_1 and C_2 respectively. The epipolar plane for these two cameras and a point p_1 (which is the projection of a contour point P_1) in image I_1 is defined as the plane passing through p_1, C_1 and C_2. For two images $I_1(t)$ and $I_1(t+1)$ taken by a camera at times t and $t+1$, the epipolar plane for a point p_1 in $I_1(t)$ is defined as the plane passing through p_1 and the instantaneous translational velocity. In either case, the epipolar lines are defined as the intersections of the epipolar plane with the two image planes.

Similar to the conventional stereo case, we can match points lying on corresponding epipolar lines. The difference here is that the two matched image points are not projections of the same 3D point (Fig. 3). Matching based on epipolar geometry was employed in previous approaches as well (Blake and Cipolla, 1992; Szeliski and Weiss, 1993; Vaillant and Faugeras, 1992; Joshi et al., 1994).

Using the epipolar plane for camera C_2 we can find a match point p_2 in image I_2. Similarly if we have a third camera C_3, the corresponding epipolar plane can be used to find a match point p_3 in image I_3. Thus we have a triple of points matched in the three images using the epipolar geometry.[3]

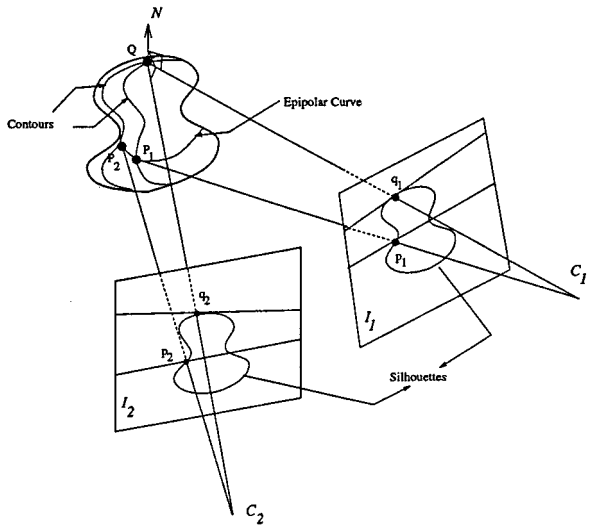

Figure 3. Epipolar geometry.

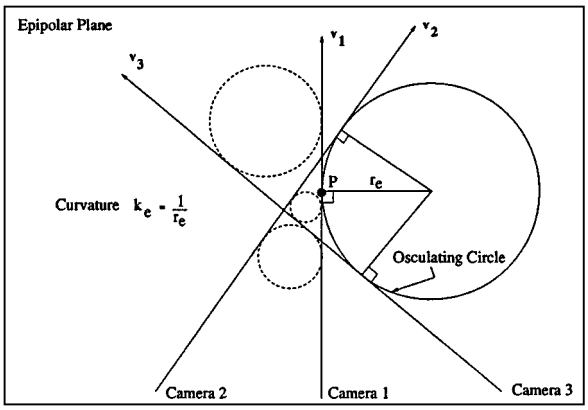

Figure 4. Estimating the osculating circle.

Consider for a moment a continuous relative motion between a camera and an object. At each point on the silhouette, the points matched using the instantaneous epipolar geometry trace a surface curve everywhere tangent to the corresponding family of viewing rays. This is the *epipolar curve* (Blake and Cipolla, 1992; Szeliski and Weiss, 1993; Vaillant and Faugeras, 1992). In the trinocular imagery case, the points matched using epipolar geometry will lie on the corresponding epipolar curve (Fig. 3). The epipolar curve is regular at points such as P_1 in Fig. 3, where the surface normal is not perpendicular to the epipolar plane. However, at points such as Q in Fig. 3, the epipolar plane is tangent to the surface, and the surface points Q_1 and Q_2 merge. Thus the matched points q_1 and q_2 are the projection of the same 3D point Q. The epipolar curve degenerates at such points and has a cusp (Giblin and Weiss, 1995). Points like Q are the *frontier points* for the corresponding camera motion (Giblin et al., 1994). We will use them to estimate the motion parameters as explained in Section 3.

2.4. Structure Estimation

Since the epipolar curve is tangent to the view line at every point, we can estimate the osculating circle to the epipolar curve by finding the circle that is tangent to the three viewing rays. When the three optical centers of the trinocular imagery are collinear, the three images share a common pencil of epipolar planes, and the three viewing rays are coplanar. For general camera configurations however, the three rays are not coplanar, and we project them onto a common plane.

As shown in Fig. 4, in general there are four circles tangent to a set of three lines in a plane (Vaillant and Faugeras, 1992). But if we know which side of each view line the object lies, the circle can be detected uniquely (the solid circle in Fig. 4). The point where this circle touches the central view line is an estimate of the 3D surface point. This gives the depth of the point along the central view line. The curvature κ_e of the circle is an estimate of the curvature of the epipolar curve. We can compute the normal curvature κ_n of the surface along the view line (which is the tangent direction of the epipolar curve) using κ_e and the angle β between the surface normal and the epipolar plane normal:

$$\kappa_n = \kappa_e \cos\beta. \tag{9}$$

Knowing the normal curvature gives us a constraint on the surface curvatures κ_1, κ_2 and the angle γ, since the normal curvature along a tangent direction making an angle β' with one of the principal directions is related to κ_1 and κ_2 by the Euler formula (do Carmo, 1976):

$$\kappa_n = \kappa_1 \cos^2\beta' + \kappa_2 \sin^2\beta'. \tag{10}$$

Here, $\beta' = \frac{\pi}{2} - \gamma$. We obtain one more constraint on the surface structure based on the fact that the tangent to the 3D contour is along a direction conjugate to the view line (Koenderink, 1990).[4] Once the depth of the points along the silhouette is computed, we can estimate the direction of the tangent to the 3D contour; this gives a

constraint on κ_1, κ_2 and γ. This constraint, along with the constraint given by the curvature of the silhouette in the central image, and the constraint given by Eqs. (9) and (10) give us three equations which are solved to obtain the values of the structural parameters κ_1, κ_2 and γ.

2.5. Implementation and Results

Before presenting our results, let us make a few remarks about our implementation. Because the epipolar geometry is known for the three cameras, we first identify the frontier points as points with surface normal parallel to the epipolar plane normal. We then find a correspondence between the sets of frontier points such that the matched frontier points lie on the corresponding epipolar lines. A match between the frontier points induces a match between the sections of the silhouettes lying between the frontier points. We then use the epipolar lines to find a match for each point on these sections. A simple method for finding the epipolar match of a silhouette point p_1 in image I_1 is to search for the silhouette point in image I_2 that lies on the epipolar line. However, because we have the silhouette in image I_2 in discrete form, we can at most find a point, say p_2, with the shortest distance from the epipolar line. To improve the accuracy of the reconstruction, we use interpolation to construct a more accurate intersection of the silhouette with the epipolar line. We find the neighboring point p_3 of the point p_2 such that p_2 and p_3 lie on opposite sides of the epipolar line and use linear interpolation to estimate the intersection of the silhouette and the epipolar line.

It should also be noted that our method for constructing a local model of the surface structure is only an approximation, since we project the three viewing rays into a common plane before estimating the osculating circle of the epipolar curve. We have also implemented the exact method proposed by Boyer and Berger (1997), where the paraboloid model is constructed without any approximation, but the results have not proven to be significantly better in our experiments.

2.5.1. Synthetic Data. We have applied our structure estimation algorithm to synthetic data corresponding to a sphere of radius 101.33 mm being observed by a trinocular rig from a distance of 1500 mm. The focal length is 25 mm, and the cameras are positioned at three points on a great circle centered at the center of the sphere, making the three viewing directions noncoplanar. Because a sphere is observed, the depth of each point on the occluding contour is constant and the normal curvature along each view ray is equal to the inverse of the radius of the sphere.

As we move along the silhouette, the angle β between the surface normal and the epipolar plane normal changes. It gets smaller as we approach a frontier point. Figure 5(a) displays the plots of the depth error as a function of $\cos\beta$ for different values of the stereo angle. We can see that for a given stereo angle, the depth estimation becomes unreliable as we approach a frontier point. The main reason for this behavior is that at the frontier point, the epipolar curve has a cusp, and hence our structure estimation—which is based on fitting an osculating circle to the epipolar curve—fails.

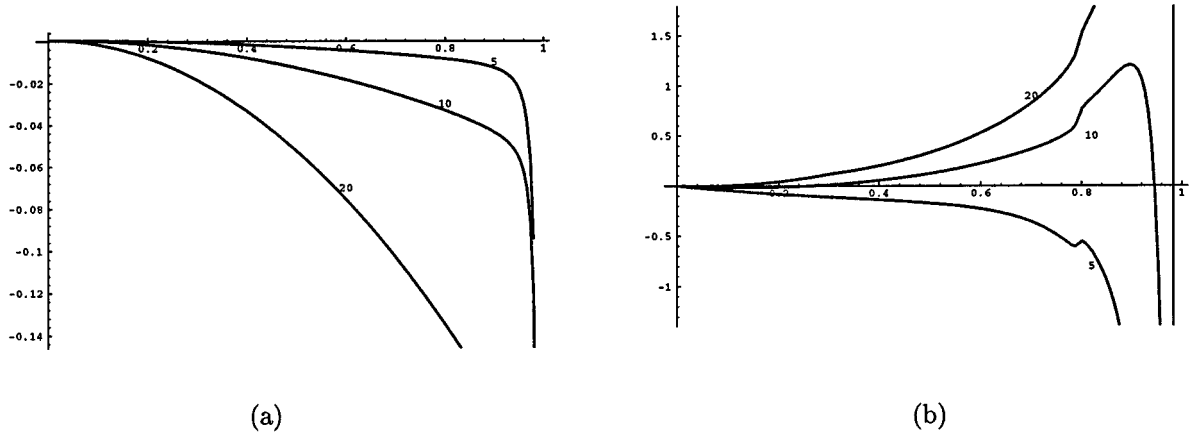

Figure 5. Error plots. (a) Depth error in mm vs. $\cos\beta$ for stereo angles of 5°, 10°, and 20°; (b) error in radius of curvature in mm vs. $\cos\beta$ for stereo angles of 5°, 10°, and 20°.

Figure 5 also demonstrates the effect of changing the stereo angle. The depth error gets larger as we increase the stereo angle (this is unlike the conventional stereo case). This behavior is due to the fact that the three non-coplanar view rays are projected onto a common plane. As the stereo angle becomes larger, the non-coplanarity of the view rays increases, increasing the error due to projection. A similar behavior is observed for the error in the radius of curvature, as shown in Fig. 5(b). The error becomes larger as we approach a frontier point and as the stereo angle is increased.

2.5.2. Real Data. We have applied the method described here to three trinocular sequences of real images. These were generated using a single calibrated camera observing an object placed on a turntable. We have simulated trinocular imagery by taking three images a fixed angle apart. A set of such triples taken successively constitutes a sequence.

In these experiments, images were acquired by a CCD camera with a 25 mm lens at a resolution of 640×480 pixels. To estimate the intrinsic parameters of the camera, and the axis of rotation of the turntable, first a non-planar calibration grid was placed on the turntable, and a sequence of fourteen images was acquired. For each image, Tsai's method was used to estimate the intrinsic and extrinsic parameters (Tsai, 1987). The extrinsic parameters were then used to determine the axis of rotation.

Image edges were detected by thresholding and linking. The linked edges were smoothed with Gaussian filter. For the smoothed curve, a cubic polynomial was fit over a sliding window of constant arclength, and the curvature and tangent at each point on the image curve were computed using the coefficients of the cubic fit.

Figures 6 and 7 present sample results on one of the sequences. Figure 6(a)–(c) shows a sample triple of images of this sequence. Figure 6(d)–(f) shows the detected contours in the corresponding images. The reconstructed 3D contour is displayed in Fig. 6(g), and the recovered Gaussian curvature along the contour is shown in Fig. 6(h). The gaps in the reconstructed contour occur because, as we approach the frontier points, i.e., as the angle between the surface normal and the epipolar plane normal becomes smaller, the epipolar-match-based reconstruction becomes less reliable. In fact, we do not estimate the structure at the frontier points. Figure 7 shows the improvement in the depth and curvature values reconstructed using the linear interpolation described in Section 2.5. To improve the smoothness of the plots further, especially the Gaussian curvature ones, which depend on second derivatives, an alternate approach would be to use B-splines, since that would directly yield derivatives.

Figures 8 and 9 show sample results for the bottle and duck sequences.

2.6. Discriminating between Viewpoint-Dependent and Independent Edges

It is desirable for an algorithm to detect the cases for which it is not applicable and to signal such cases to the user of the algorithm. In the algorithms presented here, we assume that the edges present in the scene are silhouettes. In this section, we address the problem of distinguishing silhouettes from viewpoint-independent edges in the scene, such as orientation boundaries or surface markings.

The viewpoint-independent edges can be considered to be viewpoint-dependent edges in the limit, where the radius of curvature becomes zero. Here the epipolar curve has radius of curvature equal to zero as well. Hence our structure estimation which is based on fitting an osculating circle to the epipolar curve fails. It is important to detect these edges, and treat them separately. Blake and Cipolla (1992) and Vaillant and Faugeras (1992) have shown that three images taken from known relative positions are sufficient to differentiate between viewpoint-dependent and viewpoint-independent edges in the scene.

It can be seen intuitively as follows: for a pair of images seen from two cameras, we can find a match for a point using the epipolar geometry. Assuming the edge to be viewpoint-independent, we can estimate the corresponding 3D point using triangulation. If we have another frame, we can estimate the 3D point using this frame as well. If the two estimated 3D points coincide, i.e., if the three lines of sight at the matched points are congruent, it is a viewpoint-independent edge; otherwise, it is a viewpoint-dependent one.

The above references have demonstrated that this distinction can be made even in the presence of noise. Once this distinction is made, they can be treated separately. In fact, motion estimation algorithms based on viewpoint-independent edges can be utilized to further constrain the motion parameters. See (Arbogast and Mohr, 1992; Faugeras and Papadopoulo, 1993) for motion estimation based on viewpoint-independent edges.

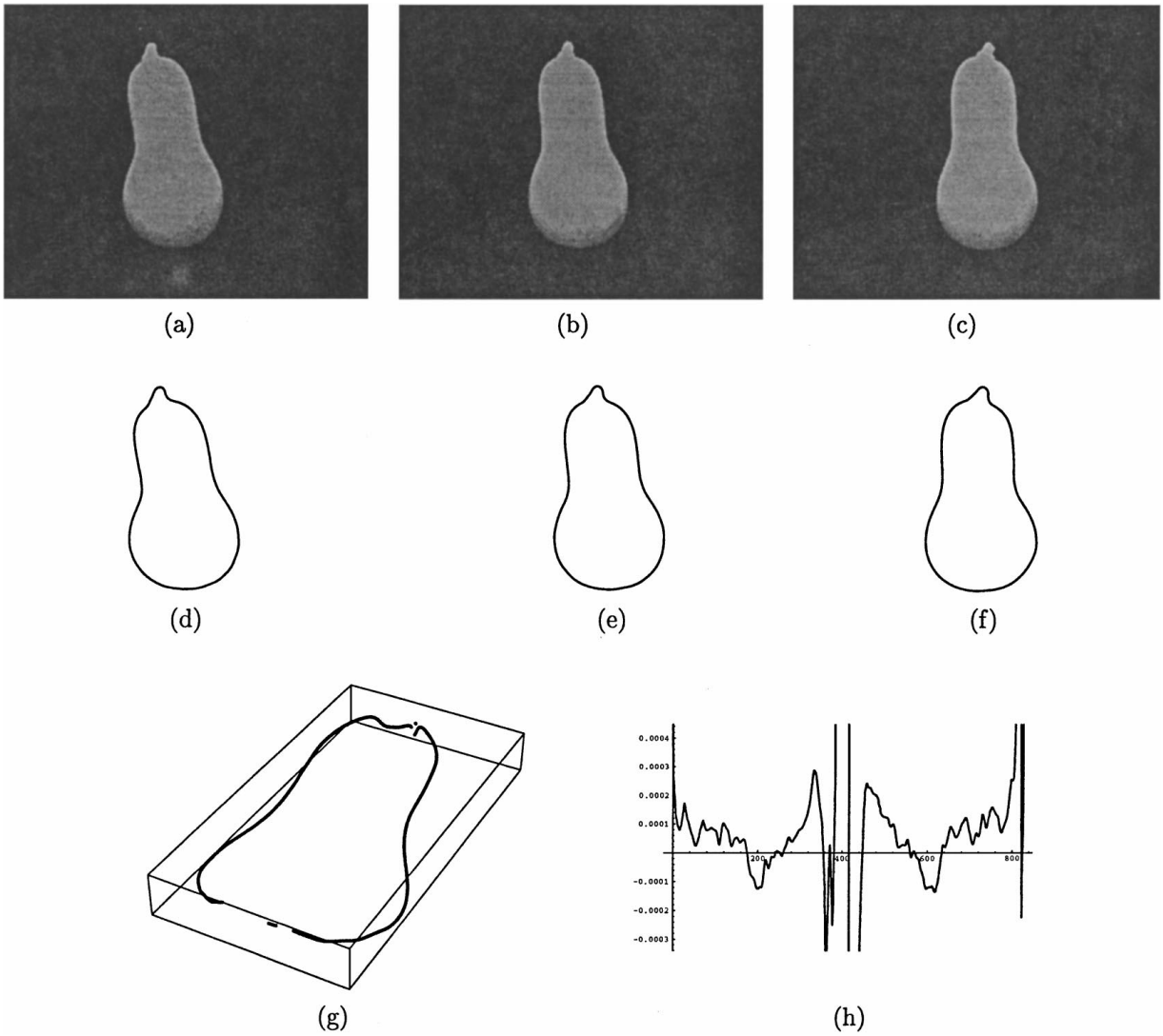

Figure 6. Squash data: (a)–(c) a sample triple of images of the trinocular imagery; (d)–(f) corresponding detected contours; (g) recovered 3D contour; (h) recovered Gaussian curvature along the contour.

3. Motion Estimation

3.1. Previous Work

Motion estimation from silhouettes is more difficult than from fixed 3D features because of the viewpoint-dependent nature of the occluding contours. One of the first efforts in this area is due to Rieger (1986). For the special case of orthographic projection of a smooth object rotating about a fixed axis, he noted an important property of the changing occluding contours: even though they slip over the surface with rotation, there are a few fixed points (the frontier points mentioned earlier) that lie at the intersection of the successive occluding contours. Taking advantage of some a priori information, Rieger showed that the angle of gaze and the depth of the point can be recovered.

This idea has been extended further by Giblin et al. (1994), who have investigated the recovery of structure and motion from a monocular sequence of silhouettes. They consider the case of a smooth curved object rotating about a fixed axis with constant angular velocity, and they have shown that, given a set of orthographic silhouettes covering a complete rotation of the object

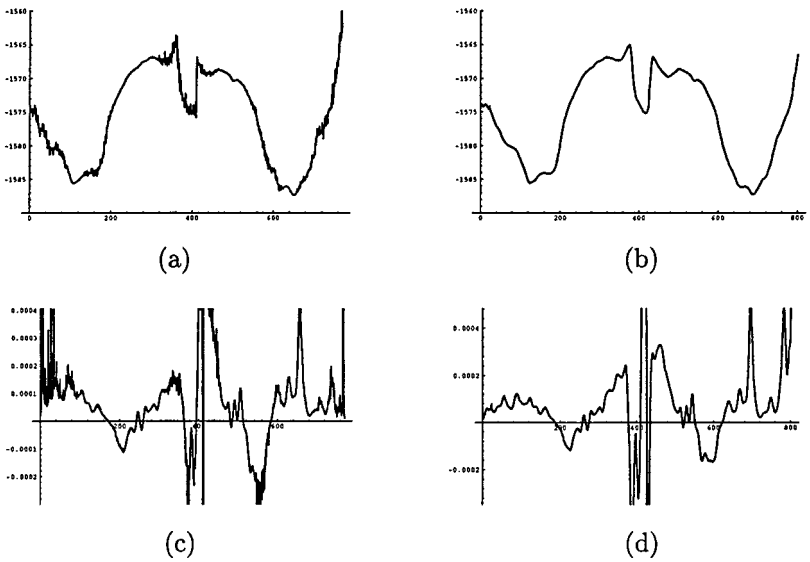

Figure 7. Effect of interpolation: (a)–(b) Depth around the silhouette before and after interpolation, respectively; (c)–(d) the Gaussian curvatures along the silhouette before and after interpolation, respectively.

about a fixed axis, the rotation axis and velocity can be recovered along with the visible region of the object surface. For the case of known angular velocity, they have also shown that the rotation axis can be recovered from the silhouettes of the object over a short time interval, and reported experimental results on synthetic data.

Recently, Cipolla et al. (1995) have attacked the general motion case. They, too, use the frontier points to constrain the motion of the viewer. After deriving properties of the normal velocity at the frontier points, they have developed an iterative procedure to estimate the essential matrix between a pair of images. They have reported the problem of sometimes converging to a local minimum due to a bad initial guess.

Our method also uses the frontier points. However, we use these only to get an initial estimate of the translation for a given estimate of the rotation. We solve the problem of obtaining a good initial point for our iterative method by using another set of features: the inflections of the silhouette.

We obtain 3D contours on the surface in the successive frames $I_1(t)$ and $I_1(t+1)$ of the central camera using trinocular imagery. These contours are related by an unknown motion. Let us assume that between consecutive time instants t and $t+1$, the object undergoes a rotation of angle α about the axis $\mathbf{\Omega} = [\omega_x, \omega_y, \omega_z]^T$ passing through the origin, followed by a translation $[t_x, t_y, t_z]^T$. We denote by $\boldsymbol{T}$ the matrix corresponding to the translation and by $\boldsymbol{R}$ the matrix corresponding to the rotation.

The motion estimation is done in two steps. In the first step, described in Section 3.2, we use the change in the surface normal at parabolic points to estimate the rotation parameters. We then construct an initial estimate of the translation parameters by using the scaled orthographic frontier points as an approximation of the perspective ones. In the second step, the estimate of motion parameters is refined using the rest of the silhouette. Both the steps perform non-linear least-squares minimization. The next two sections describe these two steps in detail. Experimental results are presented in Section 3.4.

3.2. Obtaining an Initial Estimate of the Motion Parameters

3.2.1. Rotational Parameters. The inflections of the silhouette are projections of parabolic points on the object surface (Vaillant and Faugeras, 1992; Koenderink, 1984; Blake and Cipolla, 1990). Parabolic points are defined as the points where the Gaussian curvature is zero. On generic surfaces these points lie on continuous curves called parabolic curves. Under relative motion between the object and the viewer, the inflections in successive images will be projections of neighboring points lying on the parabolic curve.

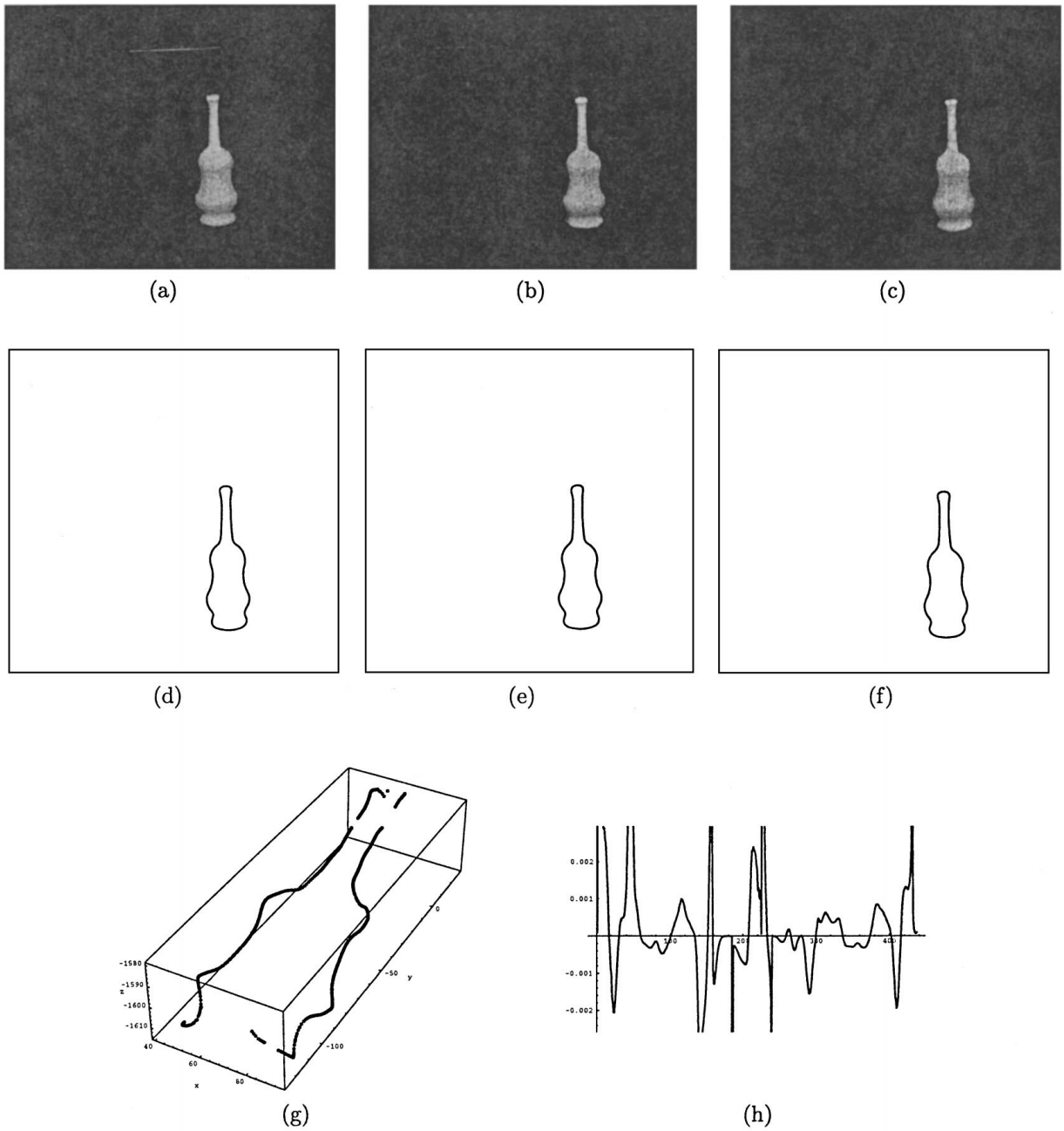

Figure 8. Bottle data: (a)–(c) a sample triple of images of the trinocular imagery; (d)–(f) corresponding detected contours; (g) recovered 3D contour; (h) recovered Gaussian curvature along the contour.

A parabolic point has a single asymptotic direction (which is also one of the principal directions) and the surface is locally cylindrical. Let us begin with the following lemma (Joshi et al., 1994, 1997) (Fig. 10).

Lemma 1. *At a parabolic point, the infinitesimal change in surface normal corresponding to any infinitesimal displacement on the surface is perpendicular to the asymptotic direction.*

Proof: Let us parameterize the surface at a parabolic point P by two parameters u and v, such that the u-axis is along the asymptotic direction $\boldsymbol{A}$ at P and the v-axis is along a direction perpendicular to $\boldsymbol{A}$ in the tangent

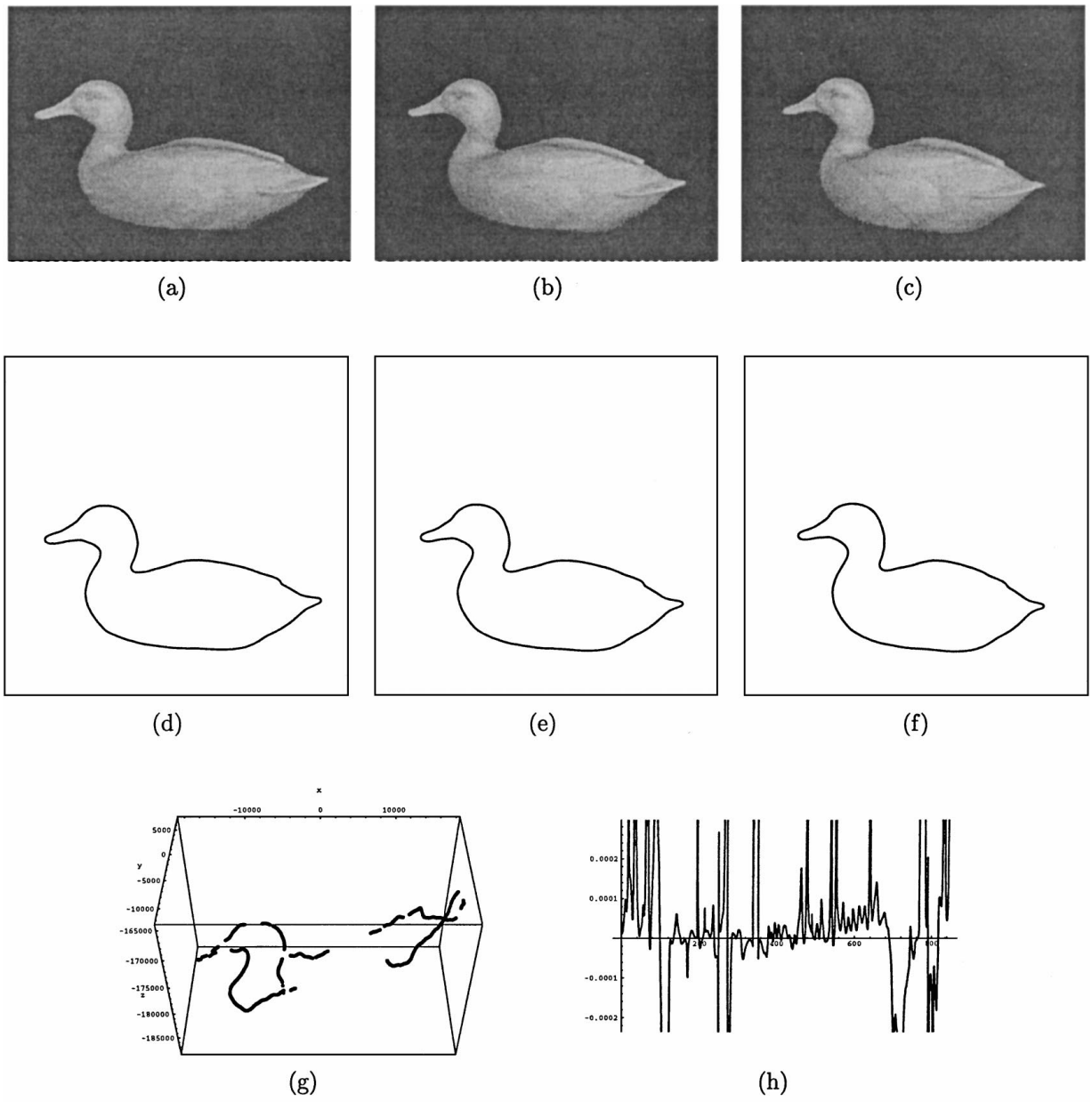

Figure 9. Duck data: (a)–(c) a sample triple of images of the trinocular imagery; (d)–(f) corresponding detected contours; (g) recovered 3D contour; (h) recovered Gaussian curvature along the contour.

plane. The u- and v-axes span the tangent plane. The second fundamental form in this basis is given by:

$$\boldsymbol{II} = \begin{bmatrix} 0 & 0 \\ 0 & k \end{bmatrix}. \tag{11}$$

The change in the normal corresponding to an infinitesimal displacement along the direction $\boldsymbol{D} = [u, v]^T$ in the tangent plane is given by:

$$\boldsymbol{dN} = \boldsymbol{II}\,\boldsymbol{D} = [0, kv]^T. \tag{12}$$

Hence the change in the normal corresponding to a small displacement in the $\boldsymbol{D}$ direction is along the v-axis, thus orthogonal to the asymptotic direction (the u-axis). □

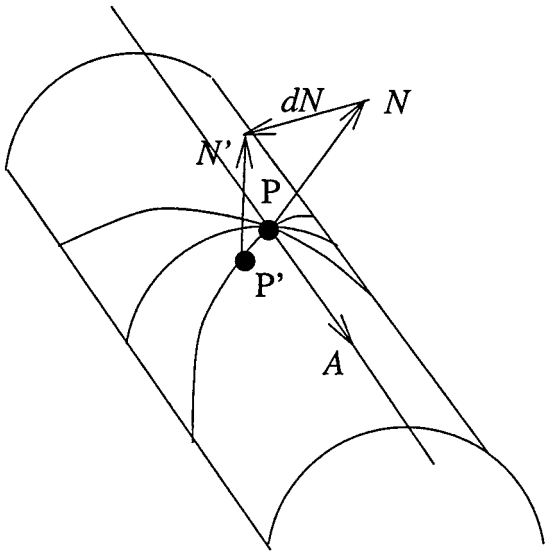

Figure 10. The change in the surface normal at a parabolic point.

Alternately, the above Lemma can also be proved as follows. As mentioned earlier, the change in surface normal along any direction $\boldsymbol{D}$ in the tangent plane is perpendicular to the direction conjugate to $\boldsymbol{D}$. Moreover, at a parabolic point, every direction in the tangent plane is conjugate to the asymptotic direction $\boldsymbol{A}$. Thus the change in surface normal along any direction in the tangent plane is perpendicular to the asymptotic direction.

Consider a silhouette inflection p which is the projection of a parabolic point P onto the central image $I_1(t)$. According to Lemma 1, as we track p in the next image $I_1(t+1)$ of the central camera, the corresponding infinitesimal change $d\boldsymbol{N}$ in the surface normal will be orthogonal to the asymptotic direction, say $\boldsymbol{A}$, at P. Let p' be the tracked inflection in $I_1(t+1)$, which is the projection of P', a neighboring parabolic point on the surface, and let the unit surface normal at P' measured from the viewpoint at time $t+1$ be $\boldsymbol{N}'$, it follows from Lemma 1 that

$$\boldsymbol{A} \cdot d\boldsymbol{N} = \boldsymbol{A} \cdot (\boldsymbol{R}^{-1}\boldsymbol{N}' - \boldsymbol{N}) = \boldsymbol{A} \cdot \boldsymbol{R}^{-1}\boldsymbol{N}' - \boldsymbol{A} \cdot \boldsymbol{N} = 0. \tag{13}$$

But the asymptotic direction $\boldsymbol{A}$ lies in the tangent plane, implying $\boldsymbol{A} \cdot \boldsymbol{N} = 0$. Equation (13) reduces to:

$$\boldsymbol{A} \cdot \boldsymbol{R}^{-1}\boldsymbol{N}' = 0. \tag{14}$$

Note that the above equation is valid for an infinitesimal motion. We parameterize the rotation using three angles—the angle ϕ made by $\boldsymbol{\Omega}$ with the Z-axis, the angle ψ between its projection in XY-plane and the X-axis, and the rotation angle α. To recover the rotation parameters, we need to track at least three inflections. With $n \geq 3$ inflections present in images $I_1(t)$ and $I_1(t+1)$, we use least-squares minimization with the objective function given by:

$$\sum_{i=1}^{n} [\boldsymbol{A}_i \cdot (\boldsymbol{R}^{-1}\boldsymbol{N}'_i)]^2. \tag{15}$$

The minimization for Eq. (15) is done over the three parameters ϕ, ψ and α. Note that we have to first find correspondences between the sets of inflections on the silhouettes in the two images. Since inflections are discrete points on the silhouette, they are in general few in number and all possible matches between the two sets of inflections can be considered. Moreover, if the motion is small, the ordering of the inflections along the silhouette is maintained in general. This condition further reduces the number of possible matches. We select correspondences such that (1) the ordering of the matched inflections along the silhouette is maintained, and (2) the angle between the normals at the matched inflections is small.

We need to estimate the asymptotic direction $\boldsymbol{A}$ at each inflection at time t. It is interesting that Eq. (14) can be used to compute $\boldsymbol{A}$ at a given inflection using trinocular imagery: we have matched inflections in the three images at time t with known relative viewpoints from the calibration. Thus we know $\boldsymbol{R}$ for each pair of images. We can compute $\boldsymbol{A}$ using Eq. (14) since we know $\boldsymbol{N}'$ at each inflection.

3.2.2. Translational Parameters. Under scaled orthographic projection, rotation parameters alone determine the direction of the epipolar plane normal. Thus, with an estimate of the rotation parameters the epipolar plane normal can be estimated and in turn frontier points can be detected. Once the frontier points have been detected and matched, the estimation of the translational parameters becomes a linear problem (Joshi et al., 1994). The perspective case is more complicated since the rotational parameters alone do not determine the epipolar plane's normal. Thus, strictly speaking, the frontier points cannot be detected unless the translation parameters are estimated first.

However, our goal here is only to obtain an initial estimate of the motion parameters, and we use the orthographic frontier points as approximations to the real ones. Our experiments have shown that the epipolar plane's normal estimated under the scaled orthographic projection approximation is close to the correct perspective one.[5] Given an initial estimate of the rotation parameters, we estimate the epipolar plane normal with the scaled orthographic projection approximation and use it to detect the orthographic frontier points. Using this approximation gives a good initial estimate of the translational parameters, which is then refined in the second step of our algorithm.

Recall that the matched frontier points are projections of the same 3D point. Therefore if a frontier point f in image $I_1(t)$ matches with the frontier point f' in image $I_1(t+1)$, then we have:

$$\boldsymbol{F}' = \boldsymbol{R}\boldsymbol{F} + \boldsymbol{T}. \tag{16}$$

This implies that for a given rotation, estimating the translation becomes a linear problem once the frontier points have been detected and matched. This is taken as the initial estimate of the translation parameters for a given estimate of the rotation parameters.

3.3. *Refining the Motion Estimate*

In the second step, we use the structure along the entire silhouette to refine the estimate of the motion parameters obtained in the first step.

With an estimate of $\boldsymbol{R}$ and $\boldsymbol{T}$, we can determine the location of the camera in the frame $I_1(t+1)$ relative to the local paraboloids in the frame $I_1(t)$. We can also determine the epipolar plane for each point in the image $I_1(t)$. Once the structure parameters of the local paraboloids and the epipolar plane are known, we can estimate the curvature of the epipolar curve at each point. Consider a point p_i on the silhouette in image $I_1(t)$ (Fig. 11). We can predict the match point ${}^{p}p_i'$ in frame $I_1(t+1)$ as the projection of a point ${}^{p}P_i{}'$ which satisfies the following two conditions:

1. ${}^{p}P_i{}'$ lies on the estimated osculating circle of the epipolar curve, and
2. the surface normal at ${}^{p}P_i{}'$ is orthogonal to the viewing direction.

Note that we can also detect the epipolar match point ${}^{d}p_i'$ in image $I_1(t+1)$ as the intersection point of the silhouette at $t+1$ and the estimated epipolar line corresponding to p_i.

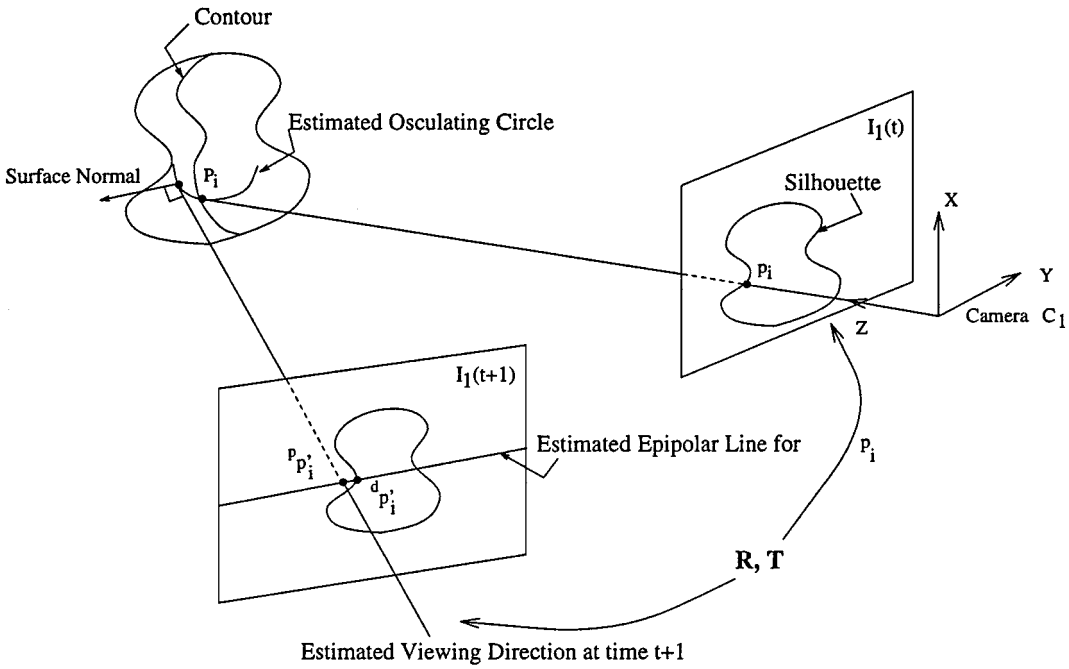

Figure 11. Predicted and detected epipolar match.

In the refinement step, we iteratively minimize the sum of the squared distance between ${}^{p}p'_i$ and ${}^{d}p'_i$ for all silhouette points p_i. The minimization is over the six-dimensional space of $\boldsymbol{R}$ and $\boldsymbol{T}$ parameters. In our algorithm, we iterate on the rotation parameters, and at each iteration step we perform a non-linear least-squares minimization to determine the best translation parameters that give the minimum sum of distances between predicted and observed silhouette points.

The algorithm for motion estimation can be written as follows.

1. Obtain an initial estimate of the rotation parameters $(\alpha_0, \phi_0, \psi_0)$ using tracked inflections as explained in Section 3.2.1. Set $\alpha = \alpha_0, \phi = \phi_0$ and $\psi = \psi_0$.
2. For the given rotation parameters α, ϕ, ψ
   (a) Detect and match the frontier points f and f' in the two central frames with the scaled orthographic projection approximation. Using the matched frontier points, compute the initial estimate of translation parameters as explained in Section 3.2.2.
   (b) Knowing the local structure at each point on the silhouette, refine the estimate of the translation parameters to minimize the sum S of the squared distance between the predicted and the detected epipolar match points for all the silhouette points. The sum S is given by:

$$S = \sum_{i=1}^{n} \text{dist}\left({}^{p}p'_i, {}^{d}p'_i\right)^2.$$

3. Minimize S by updating the values of the rotation parameters ϕ, ψ and α, and repeating Step 2.

3.4. Implementation and Results

We can potentially consider the entire silhouette for the computation of the sum S. But as observed in Section 2.5.1 the structure estimation using epipolar matches becomes less reliable as we approach the frontier points, we exclude points close to the frontier points from the computation.

Note that we have the reconstructed 3D occluding contour at each time instant. Moreover, when we predict a match point, we know the predicted 3D contour point as well. Hence, we could use the 3D distance between the detected and predicted match points as an error measure. But this may result in the propagation of errors in the structure estimation results to the motion estimation results, as confirmed by our experiments. Therefore, we have chosen the 2D distance as our error measure.

For the first step of estimating rotation parameters, we had to detect inflections of the silhouette. Since the silhouette is locally flat near an inflection, they are difficult to localize. Interestingly though, for the same reason, the estimation of the direction of the silhouette tangent is very robust. The first iterative step of our algorithm, is entirely based on the direction of the surface normal at the parabolic points. Under scaled orthographic projection, the surface normal depends on only the direction of the silhouette tangent at the tracked inflections. However, under perspective projection, it also depends on the position of the inflection along the silhouette. For reliable detection of the inflections, the image edges were smoothed using Gaussian filters of increasing variance. For every smoothed curve, all curvature zero-crossings are labeled as inflections, and the stable ones are extracted by tracking them from the coarsest to the finest scale. Since the inflections are sparse, a greedy algorithm was sufficient for determining correspondence through the scale space.

We use least-squares minimization based on the Levenberg-Marquart algorithm (IMSL Math/Library). This minimization has proven to be very stable with respect to the choice of initial value. The result of this step is used as the starting value for the search of the second iterative step of refining the estimate. We used a minimization technique based on the downhill-simplex algorithm (Press et al., 1994) for this iterative step.

If $\boldsymbol{R}$ and $\boldsymbol{T}$ represent the relative motion from time t to $t+1$, $\boldsymbol{R}^T$ and $-\boldsymbol{R}^T\boldsymbol{T}$ represent the relative motion from $t+1$ to t. We can use the structural parameters estimated at time $t+1$ to predict the silhouette at time t, making use of all the structural information available. In practice, this has improved the performance of the motion estimation algorithm.

We have applied the motion estimation algorithm to the image sequences mentioned in Section 2.5. In these sequences, the rotation axis of the turntable is not parallel to the image plane of the camera. Thus, the three effective optical centers are not collinear, and the effective motion of the object is a general one. It should be noted that the "true" camera motion has been computed for the three sequences through calibration of the camera setup and turntable, and this has allowed us to conduct quantitative tests of our motion estimation method.

Table 1. Result of the motion estimation for the squash.

		Rotation axis $\boldsymbol{\Omega}(\omega_x, \omega_y, \omega_z)$	Error in $\boldsymbol{\Omega}$	Rotation angle α	Error in α	Translation (t_x, t_y, t_z) in mm
	True	(0.008, 0.94, 0.33)	—	5.0	—	(125.6, 0.64, −5.03)
1	1st step	(0.0339, 0.942, 0.335)	1.5	5.05	0.0493	(127, −2.94, −4.98)
	Final	(−0.0197, 0.944, 0.33)	1.61	5.17	0.172	(128, 4.69, −5.35)
2	1st step	(0.0677, 0.953, 0.294)	4.0	5.59	0.586	(142, −8.4, −6.15)
	Final	(0.0312, 0.947, 0.32)	1.42	5.25	0.248	(132, −2.57, −5.41)
3	1st step	(−0.0424, 0.942, 0.334)	2.92	5.003	0.003	(125, 7.7, −5.08)
	Final	(−0.11, 0.939, 0.324)	6.82	5.002	0.002	(125, 17, −5.49)
4	1st step	(−0.0557, 0.948, 0.314)	3.79	5.04	0.0409	(127, 9.29, −3.23)
	Final	(0.00372, 0.944, 0.33)	0.268	4.87	0.132	(122, 1.19, −4.88)
5	1st step	(0.163, 0.933, 0.322)	8.9	4.72	0.277	(117, −19.7, −3.95)
	Final	(0.0475, 0.942, 0.332)	2.25	4.98	0.0185	(125, −4.69, −4.77)
6	1st step	(0.0469, 0.909, 0.415)	5.72	4.68	0.317	(113, −4.19, −3.4)
	Final	(−0.0301, 0.946, 0.324)	2.24	5.17	0.165	(130, 6.13, −5.15)
7	1st step	(−0.0855, 0.956, 0.282)	6.07	5.35	0.35	(136, 14.4, −6.2)
	Final	(0.0105, 0.949, 0.315)	0.915	5.42	0.424	(137, 0.476, −5.87)

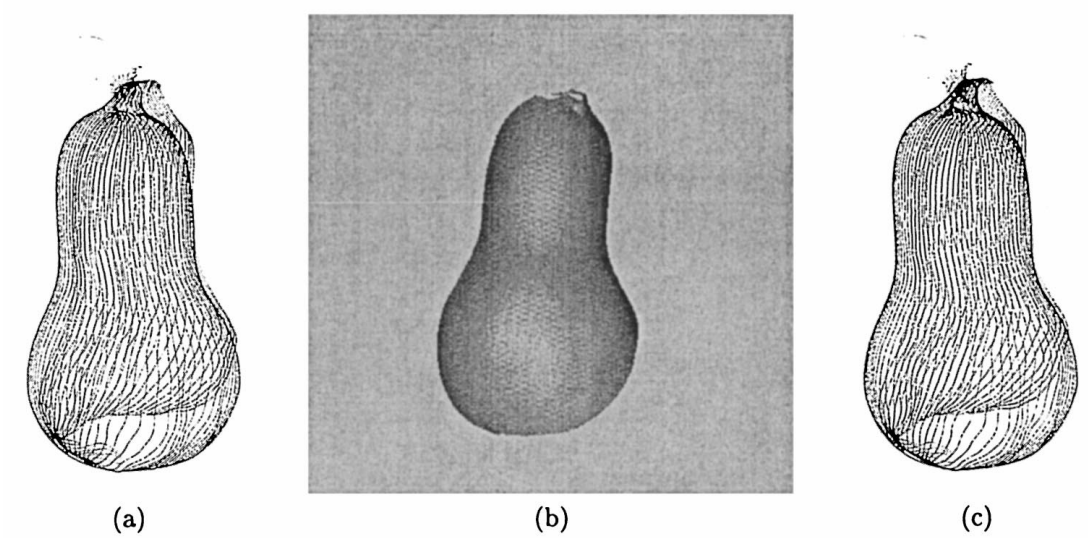

Figure 12. (a) and (b) Two views of the global structure of the squash after 30 frames; (c) structure recovered from calibrated motion.

3.4.1. Squash Sequence. The effective stereo angle for this sequence was 10°, with 5° rotation between successive time instants. Table 1 lists the recovered motion parameters after each step on a sample set of frames. For each step, we also list the angle between the true and estimated rotation axes and the error in the rotation angle. All angles are in degrees. Although the motion was constant throughout the sequence, this information was not used in the algorithm. We have found that the first step of minimization is stable with respect to the initial guess. The result of this step is used as the initial guess for the second step. Because the rotation is modeled to be about an axis passing through the origin, a small error in the rotation parameters results in a relatively large error in translational parameters.

Figure 12(a) shows the global structure recovered after 30 frames. We have displayed the reconstructed

Table 2. Result of the motion estimation for the bottle.

		Rotation axis $\mathbf{\Omega}(\omega_x, \omega_y, \omega_z)$	Error in $\mathbf{\Omega}$	Rotation angle α	Error in α	Translation (t_x, t_y, t_z) in mm
	True	(−0.042, 0.97, 0.25)	—	5.0	—	(130.1, 6.99, −5.17)
1	1st step	(−0.07, 0.97, 0.22)	2.07	8.86	3.86	(232.6, 19.53, −13.14)
	Final	(0.0312, 0.947, 0.32)	1.42	5.25	0.25	(132.9, −1.36, −4.6)
2	1st step	(−0.08, 0.93, 0.35)	6.59	6.24	1.24	(142, −8.4,−6.15)
	Final	(−0.12, 0.96, 0.24)	4.47	6.91	1.91	(179.5, 24.7, −8.9)
3	1st step	(0.11, 0.97, 0.22)	8.9	5.94	0.94	(156.2, −17.0, −5.68)
	Final	(−0.07, 0.97, 0.25)	1.75	2.03	2.97	(50.33, 4.78, −3.25)
4	1st step	(−0.49, 0.79, 0.37)	28.6	0.77	4.23	(12.19, 11.3, −2.31)
	Final	(−0.02, 0.97, 0.26)	1.57	5.76	0.76	(150.23, 3.92, −6.23)
5	1st step	(0.03, 0.96, 0.28)	4.22	4.38	0.62	(112.65, −2.46, −4.42)
	Final	(−0.11, 0.97, 0.24)	3.82	6.94	1.94	(181.6, 22.97, −9.94)
6	1st step	(−0.29, 0.93, 0.22)	14.34	7.08	2.08	(178.8, 59.10, −11.80)
	Final	(−0.04, 0.97, 0.24)	0.52	6.54	1.54	(172.9, 19.03, −9.23)

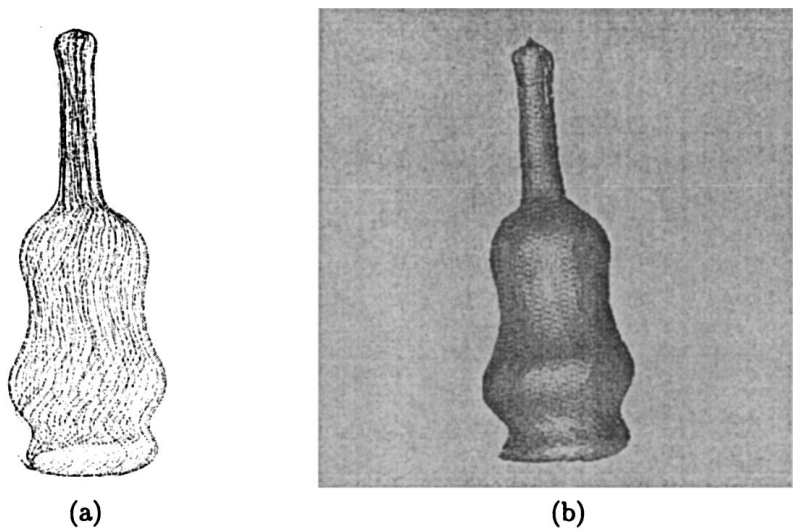

Figure 13. The global structure of the bottle after 25 frames.

3D occluding contours placed in a common coordinate frame after de-rotating them using the estimated motion. A shaded version is shown in Fig. 12(b).

The contours in Fig. 12 seem to spiral a bit instead of closing the structure. At first, this may appear to be a result of accumulation of errors in the pair-wise motion estimates. However, we compared the reconstruction using the estimated motion to that obtained using the true motion. We confirmed that the spiral nature of the structure was a result of errors in calibration: for comparison, Fig. 12(c) shows the reconstruction using the true motion.

3.4.2. Bottle Sequence. The effective stereo angle for this sequence was 10°, with 5° rotation between successive time instants. Table 2 lists the recovered motion parameters after each step on a sample set of frames for the bottle. In our experiments, we noticed that the rotation axis would show up close to the true rotation axis, whereas the estimated rotation angle would be larger than the true value of 5°. This happened because the object is symmetric and the rotation axis was almost parallel to the axis of symmetry. We modified the first step of the algorithm as follows. First, an initial estimate of the motion using the inflection and frontier

Table 3. Result of the motion estimation for the duck decoy.

		Rotation axis $\mathbf{\Omega}(\omega_x, \omega_y, \omega_z)$	Error in $\mathbf{\Omega}$	Rotation angle α	Error in α	Translation (t_x, t_y, t_z) in mm
	True	(0.012, −0.983, −0.18)	—	−3.0	—	(90.04, 1.65, −2.82)
1	1st step	(0.34, −0.93, −0.07)	19.7	−3.38	−0.38	(96.51, 35.26, −5.19)
	Final	(0.04, −0.98, −0.15)	2.21	−3.02	0.02	(91.15, 4.09, −3.27)
2	1st step	(−0.27, −0.91, −0.28)	18.04	−5.97	−2.97	(167.8, −47.76, −7.7)
	Final	(0.02, −0.98, −0.17)	0.41	−2.993	−0.006	(89.91, 2.04, −2.75)
3	1st step	(−0.05, − 0.99, −0.1)	5.93	−4.78	−1.78	(149.0, −8.6, 35.4)
	Final	(−0.002, −0.98, −0.15)	1.53	−3.33	0.33	(100.4, 0.24, −4.81)
4	1st step	(−0.023, −0.99, −0.04)	7.87	−3.43	0.43	(102.2, −2.3, −8.9)
	Final	(0.039, −0.98, −0.193)	1.76	−2.92	−0.077	(87.35, 4.15, −2.08)
5	1st step	(0.17, −0.78, −0.59)	27.94	−1.54	1.46	(32.79, 10.65, −45.0)
	Final	(0.01, −0.98, −0.18)	0.43	−3.07	0.07	(92.07, 1.19, −2.23)

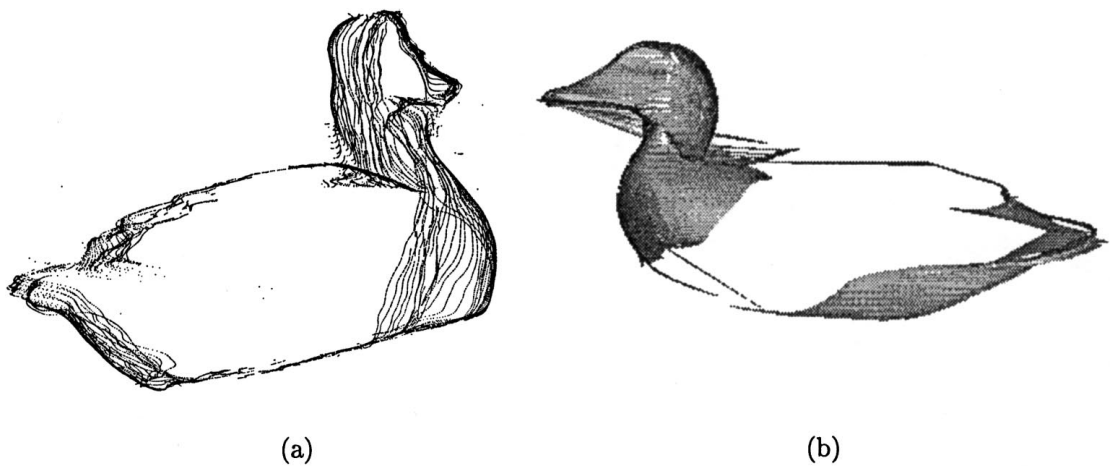

Figure 14. The global structure of the duck after 36 frames.

points at a given time t was computed. The objective function S was computed using this motion estimate. We then compared this S to the one obtained using the final estimated motion at time $t-1$. We selected the motion that yields a smaller value of S between these two, to initiate the search of the refinement step. Figure 13 shows the global structure recovered over 25 frames.

3.4.3. Duck Decoy Sequence. The effective stereo angle for this sequence was 6°, with 3° rotation between successive time instants. The modified algorithm was applied to this sequence. Table 3 lists the recovered motion parameters after each step on a sample set of frames. Figure 14 shows the global structure recovered from 36 frames.

The motion estimation results on the three sequences presented here show that the rotation angle is typically recovered within a degree for the squash and the duck sequences, while within two degrees for the bottle sequence.

4. Conclusions

Although estimating structure and motion from silhouettes is more difficult than using viewpoint-independent features, we have the advantage that more information about the surface is available even from a single silhouette: the surface normal, the sign of the Gaussian curvature, and a constraint on the principal curvatures at the surface point. We have used the relationship between certain silhouette features

(inflections) and a local model of the surface structure to estimate both the motion and the global surface structure from perspective views of the silhouettes of a moving object. To estimate the motion, we have also used another set of points on the silhouette: the frontier points.

The main motivation of this research was finding out what we can do in the *absence* of the viewpoint-independent or internal boundaries. The results obtained on real images are encouraging and demonstrate the validity of the method. Clearly, to get more robust results, and for a more practical system, it should utilize all the information available in the images.

We have demonstrated that structure and motion from silhouettes can be used as a component of a more complete system. In a more complete system, additional detail e.g., concavities—which never show up on the silhouette—could be recovered using other "shape-from-X" methods. Moreover, to get more robust results, we can incorporate other features present in the image to further constrain the structure as well as the motion parameters. For example, if surface markings or internal boundaries are present, applicable techniques (Arbogast and Mohr, 1992; Faugeras and Papadopoulo, 1992) can be used to recover the 3D edges and to estimate the motion parameters. The recovered 3D edges can be incorporated in the global structure using confidence measures; whereas the motion parameters can provide a starting point for our second iterative step.

Acknowledgments

This work was supported in part by the National Science Foundation under Grant IRI-9224815 and by the Defense Advanced Research Projects Agency and the National Science Foundation under Grant IRI-8902728. We thank D.J. Kriegman and B. Vijayakumar for providing the data used in our experiments.

Notes

1. Notation: boldface letters denote coordinate vectors or arrays. We use uppercase letters to denote 3D points in the scene and lowercase letters to denote their projections in the image, e.g., p is the projection of a 3D point P, whose coordinate vector in the camera's frame is $\boldsymbol{P}$.
2. Under orthographic projection, the silhouette normal ($\boldsymbol{n}_p$) will be parallel to the surface normal ($\boldsymbol{N}_P$), and the angle ζ will be equal to zero.
3. Since the matched points are not projections of the same 3D point, perhaps a more appropriate term would be 'related' points. However the epipolar geometry based matching that we use is a natural extension of the conventional stereo matching. Hence the use of the terminology.
4. Two directions $\boldsymbol{U}$ and $\boldsymbol{V}$ are said to be conjugate if the change in the surface normal along direction $\boldsymbol{U}$ is orthogonal to direction $\boldsymbol{V}$ and vice versa.
5. This approximation is not applicable under all circumstances. For instance, if the object is close to the camera.

References

Arbogast, E. and Mohr, R. 1992. An egomotion algorithm based on the tracking of arbitrary curves. In *Proc. European Conference on Computer Vision*, pp. 467–475.

Blake, A. and Cipolla, R. 1990. Robust estimation of surface curvature from deformation of apparent contours. In *Proc. European Conference on Computer Vision*, pp. 465–474.

Blake, A. and Cipolla, R. 1992. Surface shape from the deformation of apparent contours. *International Journal of Computer Vision*, 9(2):83–112.

Boyer, E. and Berger, M. 1997. 3D surface reconstruction using occluding contours. *International Journal of Computer Vision*, 22(3):219–233.

Cipolla, R., Astrom, K., and Giblin, P. 1995. Motion from the frontier of curved surfaces. In *Proc. International Conference on Computer Vision*, pp. 269–275.

do Carmo, M. 1976. *Differential Geometry of Curves and Surfaces*. Prentice-Hall: Englewood Cliffs, NJ.

Faugeras, O. and Papadopoulo, T. 1992. Disambiguating stereo matches with spatio-temporal surfaces. *Geometric Invariance in Computer Vision.* MIT Press: Cambridge, MA, pp. 310–331.

Giblin, P., Pollick, F., and Rycroft, J. 1994. Recovery of an unknown axis of rotation from the profiles of a rotating surface. *Journal of the Optical Society of America*, 11A:1976–1984.

Giblin, P. and Weiss, R. 1987. Reconstruction of surfaces from profiles. In *Proc. International Conference on Computer Vision*, pp. 136–144.

Giblin, P. and Weiss, R. 1995. Epipolar curves on surfaces. *Image and Vision Computing*, 13(1):33–44.

"IMSL Math/Library" Version 2 Users Manual 1991. *Visual Numerics.*

Joshi, T., Ahuja, N., and Ponce, J. 1994. Silhouette-based structure and motion estimation of a smooth object. In *Proc. ARPA Image Understanding Workshop*, pp. 1237–1243.

Joshi, T., Ponce, J., Vijayakumar, B., and Kriegman, D. 1997. HOT curves for modelling and recognition of smooth curved 3D shapes. *Image and Vision Computing*, 15:479–498.

Koenderink, J. 1984. What does the occluding contour tell us about solid shape. *Perception*, 13:321–330.

Koenderink, J. 1990. *Solid Shape*. MIT Press: Cambridge, MA.

Kutulakos, K. and Dyer, C. 1994. Global surface reconstruction by purposive control of observer motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 331–338.

Press, W., Flannery, B., Teukolsky, S., and Vetterling, W. 1994. *Numerical Recipes in C*. Cambridge University Press.

Rieger, J. 1986. Three-dimensional motion from fixed points of a deforming profile curve. *Optics Letters*, 11(3):123–125.

Seales, B. and Faugeras, O. 1994. Building three-dimensional CAD/CAM models from image sequences. In *Second CAD-Based Vision Workshop*, Pittsburgh (PA), pp. 116–123.

Szeliski, R. and Weiss, R. 1993. Robust shape recovery from occluding contours using a linear smoother. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 666–667.

Tsai, R. 1987. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras. *IEEE Journal of Robotics and Automation*, RA-3(4): 323–344.

Vaillant, R. and Faugeras, O. 1992. Using extremal boundaries for 3D object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):157–172.

Zheng, J. 1994. Acquiring 3D models from sequences of contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):163–178.

Zhao, C. and Mohr, R. 1994. Relative 3D regularized B-spline surface reconstruction through image sequences. In *Proc. European Conference on Computer Vision*, Lecture Notes in Computer Science, vol. 800, Springer-Verlag, pp. 417–426.