In Proceedings of CVPR '96

Structure and Motion of Curved 3D Objects from Monocular Silhouettes*

B. Vijayakumar David J. Kriegman
Dept. of Electrical Engineering
Yale University
New Haven, CT 06520-8267

Jean Ponce
Computer Science
University of Illinois
Urbana, IL 61801

Abstract

The silhouette of a smooth 3D object observed by a moving camera changes over time. Past work has shown how surface geometry can be recovered using the deformation of the silhouette when the camera motion is known. This paper addresses the problem of estimating both the full Euclidean surface structure and the camera motion from a dense set of silhouettes captured under orthographic or scaled orthographic projection. The approach relies on a viewpoint-invariant representation of curves swept by viewpoint-dependent features such as bitangents, inflections and contour points with parallel tangents. Feature points, which form stereo frontier points between non-consecutive images, are matched using this representation. The camera's angular velocity is computed from constraints derived from this correspondence along with the image velocity of these features. From the angular velocity, the epipolar geometry is ascertained, and infinitesimal motion frontier points can be detected. In turn, the motion of these frontier points constrains the translation component of camera motion. Finally, the surface is reconstructed using established techniques once the camera motion has been estimated.

1 Introduction

Most approaches for estimating the 3-D structure of an object from pictures taken by a moving camera are based on establishing a correspondence between viewpoint-independent image features. This correspondence is explicit in feature-based approaches where tokens such as points [5, 17] or lines [16] are tracked through an image sequence, and implicit in approaches using infinitesimal motion where the challenge is estimating the motion field [9]. For objects with few surface markings and little texture, the most reliable image feature is the object's silhouette, i.e., the projection into the image of the curve, called the occluding contour, where the cone formed by the optical rays grazes the surface. As the camera moves, the occluding contour changes, and when the camera's motion is known, it is possible to estimate the second-order structure of the observed surface along the occluding contour from the corresponding deformation of the silhouette: this was first established by Giblin and Weiss for orthographic projection with coplanar viewing directions [7], and then extended to 3D objects under perspective projection by others [1, 2, 4, 15, 18]. It has also been shown how to actively move the camera to reconstruct the entire surface using these methods [12].

A method has been developed for estimating *both* the surface structure and the camera motion from perspective images acquired by a calibrated trinocular rig [10]. For a single moving camera, techniques have been proposed for recovering the camera motion when it is constrained to be a rotation about a fixed axis[6, 14]. More recently, a method was presented for determining the epipolar geometry for infinitesimal and finite motions [3], but this technique is iterative and requires an initial guess for the direction of translation or for the essential matrix.

We address the problem of estimating both the full Euclidean surface structure and the camera motion from a dense set of silhouettes captured under orthographic or scaled orthographic projection. The critical observation is that while the local information conveyed by the deformation of the silhouette is not sufficient to completely determine the observer's motion, it can be combined with the more global information conveyed by correspondences established between non-consecutive images to recover the whole motion up to a unique scale factor that is constant over the full trajectory.

In this work, we model the camera by scaled orthographic projection. For a point $P \in \mathbb{R}^3$ and camera whose origin is at O and whose image plane is spanned by the orthogonal vectors $\boldsymbol{i}$ and $\boldsymbol{j}$, the coordinates (X, Y) of the projected point are given by:

$$\begin{cases} X &= \xi \boldsymbol{i} \cdot (P - O) \\ Y &= \xi \boldsymbol{j} \cdot (P - O) \end{cases} \tag{1}$$

where $(O, \boldsymbol{i}, \boldsymbol{j})$ is the camera's coordinate frame, (x_0, y_0, z_0) are the coordinates in this frame of some reference point, and $\xi = 1/z_0$. The vector $\boldsymbol{k} = \boldsymbol{i} \times \boldsymbol{j}$ is the viewing direction and is also denoted by $\boldsymbol{v}$. For pure orthographic projection, ξ is taken to be constant and without loss of generality we shall choose $\xi = 1$.

For a moving camera, $O, \boldsymbol{i}, \boldsymbol{j}$ are functions of time, and so the camera's motion can be represented by its linear velocity $\dot{O}$ and angular velocity $\boldsymbol{\Omega}$. Furthermore the distance to the reference point might also be changing, and so ξ may be a function of time. We

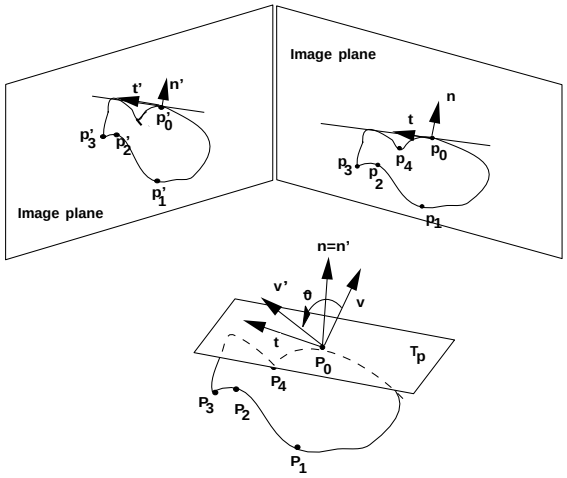

Figure 1: **Stereo Frontier Points:** Between two images, the stereo frontier points on a surface are the points of intersection of the two occluding contours.

assume that ξ is constant in this paper; however the method is extended in [20] to situations where ξ varies.

1.1 Smooth Surfaces, Epipolar Geometry and Frontier Points

For a smooth surface observed under orthographic projection, the occluding contour is the set of surface points where the surface normal $\boldsymbol{n}$ is orthogonal to the viewing direction $\boldsymbol{v}$, and the silhouette is the projection of the occluding contour. Note that the occluding contour and the silhouette necessarily depend on the viewing direction.

Between any two images taken by a camera modelled with orthographic projection, there is a one-parameter family of epipolar planes whose normals are orthogonal to the two viewing directions $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ associated with the cameras [5]. In stereo vision, the projection of an epipolar plane onto two epipolar lines is used to help establish correspondences. Now for two images of the same smooth surface, the occluding contours will be distinct curves. However, there is a set of isolated surface points where the two occluding contours intersect. At these points, which are called the *stereo frontier points*, $\boldsymbol{n}$ is orthogonal to both $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. At such points, the surface normal is orthogonal to the epipolar plane. Figure 1 shows an example.

A continuously moving camera establishes a local epipolar geometry at each instant in time [4, 8]. In particular, the normal to each plane in the family of epipolar planes is given by:

$$\boldsymbol{n}_f = \boldsymbol{v} \times \dot{\boldsymbol{v}} = \boldsymbol{\Omega} - (\boldsymbol{\Omega} \cdot \boldsymbol{v})\boldsymbol{v}$$

since $\dot{\boldsymbol{v}} = \boldsymbol{\Omega} \times \boldsymbol{v}$. Because the camera is moving, the set of points on the surface that are orthogonal to $\boldsymbol{n}_f$ define a surface curve known as the *frontier curve* [3, 8, 10]. A point on this curve will be referred to as an *infinitesimal frontier point*. See Figure 2.

At these points, the reconstruction method of [4] breaks down, but the infinitesimal frontier points can be detected when the camera motion is known.

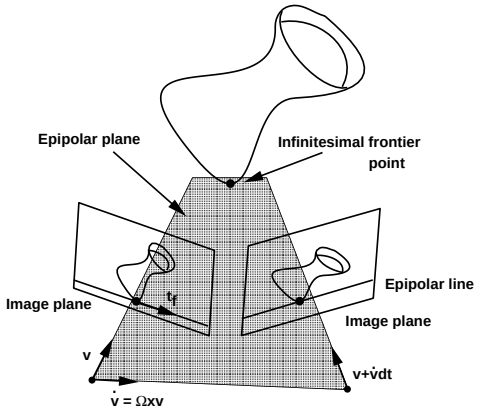

Figure 2: **Infinitesimal Frontier Points:** Infinitesimal frontier points defined by camera angular velocity.

1.2 Outline of the Algorithm

The reconstruction algorithm is composed of the following steps which will be detailed subsequently.

1. For each image in the sequence, detect the silhouette and locate feature points (bitangents, inflections, parallel tangents) on the contour.
2. Track the feature points through the image sequence and construct a set of invariant curves, one for each tracked bitangent and inflection.
3. For each image in the sequence, do the following:
   (a) For each bitangent or inflection and associated parallel tangents, use the invariant curves constructed in step 2 to find corresponding features in another image such that the corresponding features are stereo frontier points of the pair of images.
   (b) From three or more sets of corresponding stereo frontier points, compute the angular velocity $\boldsymbol{\Omega}$ of the camera frame.
   (c) From $\boldsymbol{\Omega}$, determine the infinitesimal epipolar geometry and locate the infinitesimal frontier points.
   (d) From constraints imposed by the motion of the frontier points, compute the linear velocity of a point on a reference curve.
   (e) Using the epipolar parameterization, reconstruct the occluding contour in a frame whose origin is at the reference point.
4. Using the computed camera motion and reconstructed occluding contours, construct a surface representation in a common coordinate system.

In the next four sections, we only consider pure orthographic projection ($\xi = 1$). An extension to weak perspective (scaled orthographic projection) where ξ is function of time can be found in [20]. Under pure orthographic projection, the component of camera motion along the viewing direction cannot be determined; two images will be identical if they are taken by cameras whose locations differ only by translation along the viewing direction. Like Tomasi and Kanade, we perform reconstruction with respect to some feature point on the object [17], but in our case, this feature lies on a surface curve and is viewpoint-dependent.

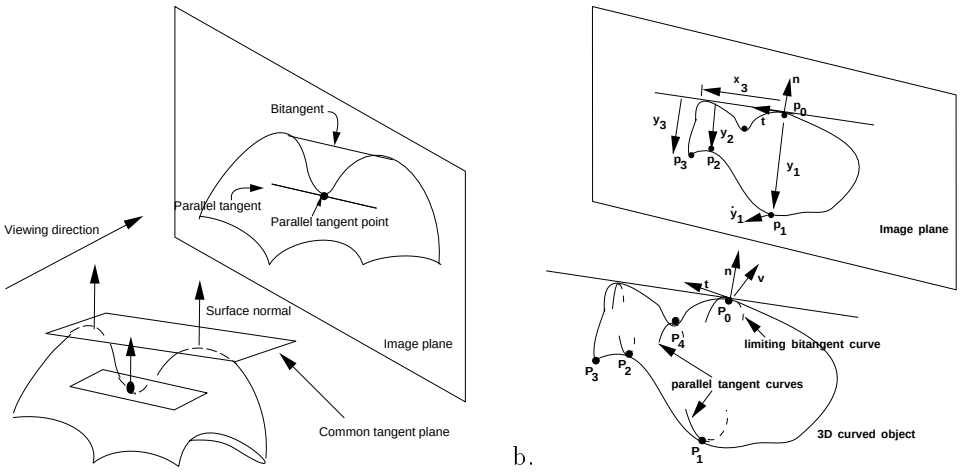

Figure 3: **Parallel Tangents: a**. Surface points whose tangent planes are parallel to the tangent plane at a limiting bitangent project onto points with parallel image tangents. **b**. The limiting bitangent developable and parallel tangents define surface curves.

2 Finding Stereo Frontier Points using Invariant Curves

In order to compute the angular velocity $\boldsymbol{\Omega}$ of the camera frame, corresponding stereo frontier points are established between feature points of the silhouette and other images in the sequence. To find these correspondences, we build on concepts developed for a recognition approach based on invariant curves [19].

It is well known that the projection of parabolic points are inflections of the silhouette, and that the projection of points on a limiting bitangent developable are bitangents of the silhouette [11, 19, 21]. Using one of these points as a feature P_0, consider the other points on the surface whose tangent planes are parallel to that of the feature. As shown in Fig. 3, all of these surface points project to points on the silhouette whose contour tangents are parallel. Up to occlusion, for any viewing direction orthogonal to the surface normal $\boldsymbol{n}$ at P_0, these parallel tangent points will project to the silhouette and can be detected. Turning this observation around, for any pair of camera locations whose viewing directions are orthogonal to $\boldsymbol{n}$, the feature and the points with parallel tangents will all be stereo frontier points of the two images, and the tangent planes are epipolar planes.

Now, let us assume that an inflection or bitangent endpoint P_0 and n other silhouette points P_i $(i = 1, .., n)$ with parallel tangents can be tracked through a sequence of images formed by a moving camera. At one instant in time, all tracked parallel tangent points have the same surface normal $\boldsymbol{n}$. If $\boldsymbol{v}$ is the viewing direction, we construct a right-handed orthonormal coordinate frame $(\boldsymbol{t}, \boldsymbol{n}, \boldsymbol{v})$, such that $(\boldsymbol{t}, \boldsymbol{n})$ forms a basis of the image plane, and $\boldsymbol{t}$ is the direction of the tangent to the projection of the contour at P_0 and P_i. See Fig. 3.b. We write, for $i = 1, .., n$,

$$P_i - P_0 = x_i \boldsymbol{t} + y_i \boldsymbol{n} + z_i \boldsymbol{v}, \qquad (2)$$

where $P_i - P_0$ denotes the vector joining P_0 to P_i. Note that x_i, y_i, and their temporal derivatives $\dot{x}_i$ and $\dot{y}_i$ are directly observable given a sequence of images. On the other hand, z_i is unknown, and the coordinates of the vectors $\boldsymbol{t}$, $\boldsymbol{n}$, $\boldsymbol{v}$ in a fixed coordinate system are also unknown.

We define the invariant curve associated with the feature points P_0 and P_i $(i = 1, .., n)$ as the trace of the parameterized curve defined by $\boldsymbol{I}(t) = (y_1(t), \ldots, y_n(t))$ in $\mathbb{R}^n$, where t is the time parameter. In practice, the orthographic projection model is valid when the moving camera remains at a fairly constant distance from the object. This definition of invariant curves can be extended to the scaled orthographic projection model by tracing a curve on the unit sphere of $\mathbb{R}^n$. In this case, the image coordinates are of the form $X_i = \xi x_i$ and $Y_i = \xi y_i$, where $\xi = 1/z_0$ is the depth of some reference point. We define the vector $\boldsymbol{J}(t) = (Y_1(t), \ldots, Y_n(t))$, and the invariant curve is the trace of the parameterized curve $\boldsymbol{K}(t) = \frac{1}{|\boldsymbol{J}(t)|}\boldsymbol{J}(t) = \frac{1}{|\boldsymbol{I}(t)|}\boldsymbol{I}(t)$.

An important property of invariant curves is that they do not depend on the motion of the camera used to construct them, yet provide an efficient way of matching observables measured in one image to a given curve. This property has been used in the past in recognition experiments [19]. Furthermore, a necessary condition for two sets of features detected in two images to be stereo frontier points is that the value of this invariant must be equal.

Much more can be said about the structure and properties of these invariant curves (see [19]), but in this paper we will simply use them to identify the three sets of stereo frontier points that are needed to compute $\boldsymbol{\Omega}$. To find these, the invariant curves for all features and their parallel tangents over the entire sequence of images are constructed. To find stereo frontier points in an image, a feature and its parallel tangents are selected and an invariant is computed. The point on some invariant curve that is closest to the computed invariant is found. The corresponding sets of features in both images are taken to be stereo frontier points for that pair since they satisfy the necessary condition outlined above.

3 Motion Constraints from Parallel Tangents

To compute $\boldsymbol{\Omega}$ for an image, we shall need three sets of stereo frontier points between the given image and three other images in the sequence. As we shall see, each stereo frontier point provides one quadratic constraint on the components of $\boldsymbol{\Omega}$. To obtain this constraint, (2) is differentiated with respect to time and the dot product of the result with $\boldsymbol{n}$ is formed. After some simple manipulation, this yields

$$\dot{y}_i = -x_i \dot{\boldsymbol{t}} \cdot \boldsymbol{n} - z_i \dot{\boldsymbol{v}} \cdot \boldsymbol{n}. \tag{3}$$

Let $\boldsymbol{\omega}$ denote the angular velocity vector associated with the moving frame $(\boldsymbol{t}, \boldsymbol{n}, \boldsymbol{v})$. Recall that $\boldsymbol{n}$ is aligned with the surface normal of the bitangent developable or inflection, that $\boldsymbol{t}$ lies in the image plane, and that $\boldsymbol{v}$ is the viewing direction. We have $\dot{\boldsymbol{t}} = \boldsymbol{\omega} \times \boldsymbol{t}$ and $\dot{\boldsymbol{v}} = \boldsymbol{\omega} \times \boldsymbol{v}$, thus (3) can be rewritten as

$$\dot{y}_i + x_i \boldsymbol{\omega} \cdot \boldsymbol{v} - z_i \boldsymbol{\omega} \cdot \boldsymbol{t} = 0. \tag{4}$$

Solving for z_i in (4) yields

$$z_i = \frac{\dot{y}_i + x_i \nu}{\tau}, \tag{5}$$

where $\nu = \boldsymbol{\omega} \cdot \boldsymbol{v}$ and $\tau = \boldsymbol{\omega} \cdot \boldsymbol{t}$ are the components of $\boldsymbol{\omega}$ along the $\boldsymbol{v}$ and $\boldsymbol{t}$ directions.

Let us assume that along the camera trajectory, the features with parallel tangents are observed in another image from another viewing direction $\boldsymbol{v}'$. This correspondence is identified using the invariant curves discussed in the previous section. For this second image, construct as before the corresponding coordinate system $(\boldsymbol{t}', \boldsymbol{n}', \boldsymbol{v}')$. See Fig. 1.a. Note that $\boldsymbol{n}' = \boldsymbol{n}$ is the normal to the epipolar plane between the two images. Let θ be the angle of rotation about $\boldsymbol{n}$ which maps $\boldsymbol{v}$ onto $\boldsymbol{v}'$. The change of coordinates between the two images can be written as

$$\begin{cases} x' = x\cos\theta - z\sin\theta, \\ y' = y, \\ z' = x\sin\theta + z\cos\theta. \end{cases} \tag{6}$$

Combining (5) and the first row of (6) yields

$$(-\tau\cos\theta + \nu\sin\theta)x_i + \tau x_i' + \sin\theta \dot{y}_i = 0. \tag{7}$$

For n parallel tangents, define the vectors $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, $\boldsymbol{x}' = (x_1', \ldots, x_n')^T$, and $\dot{\boldsymbol{y}} = (\dot{y}_1, \ldots, \dot{y}_n)$. Equation 7 can then be rewritten in vector form as:

$$(-\tau\cos\theta + \nu\sin\theta)\boldsymbol{x} + \tau\boldsymbol{x}' + \sin\theta\dot{\boldsymbol{y}} = 0, \tag{8}$$

which implies that the vectors $\boldsymbol{x}$, $\boldsymbol{x}'$, and $\dot{\boldsymbol{y}}$ are linearly dependent. Recall that corresponding stereo frontier points are found through indexing the invariant curves. An additional condition that the correspondence must satisfy is that the $3 \times n$ dimensional matrix formed from $\boldsymbol{x}, \boldsymbol{x}'$ and $\dot{\boldsymbol{y}}$ must be rank 2. The rank can be determined through singular value decomposition, used to verify correspondences.

When $\boldsymbol{x}$ and $\boldsymbol{x}'$ are linearly independent and $n \geq 2$, $\dot{\boldsymbol{y}}$ can be expressed as a linear combination of $\boldsymbol{x}$ and $\boldsymbol{y}$ or $\dot{\boldsymbol{y}} = d\boldsymbol{x} + e\boldsymbol{x}'$. Since $\boldsymbol{x}, \boldsymbol{x}'$ and $\dot{\boldsymbol{y}}$ are measured, d and e can be computed using linear least squares.

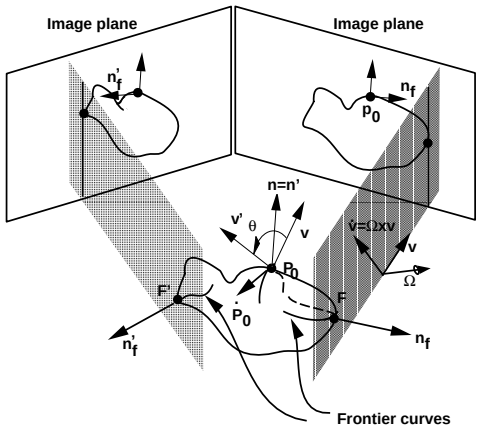

Figure 4: $\dot{P}_0$ is determined by constraints established from the relative motion of the image of P_0 and the frontier points in two image sequences.

Substituting d and e into (8) yields after some algebraic manipulation:

$$\begin{cases} \tau^2 + (\nu + d)^2 = e^2, \\ \tan\theta = \dfrac{\tau}{\nu + d}. \end{cases} \tag{9}$$

The first equation provides a quadratic constraint on two components of the angular velocity of the moving frame associated with the parallel tangents while the second one can be used to compute θ from τ and ν. To compute the angular velocity $\boldsymbol{\Omega}$ of the camera coordinate system, we first note that the feature's coordinate system $(\boldsymbol{t}, \boldsymbol{n}, \boldsymbol{v})$ and the camera coordinate system $(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$ simply differ by a rotation about viewing direction, $\boldsymbol{v} = \boldsymbol{k}$. If α denotes the angle of the rotation about $\boldsymbol{v}$ that maps $\boldsymbol{i}$ onto $\boldsymbol{t}$, then $\boldsymbol{\Omega} = \boldsymbol{\omega} - \dot{\alpha}\boldsymbol{v}$. Note that both α and its temporal derivative $\dot{\alpha}$ are directly observable from the image sequence.

Let $(\Omega_i, \Omega_j, \Omega_k)$ denote the coordinates of $\boldsymbol{\Omega}$ in the camera's coordinate system $(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$. Since $\boldsymbol{\omega} = \boldsymbol{\Omega} + \dot{\alpha}\boldsymbol{v}$, we have:

$$\begin{cases} \tau = \Omega_i\cos\alpha + \Omega_j\sin\alpha, \\ \nu = \Omega_k + \dot{\alpha}. \end{cases} \tag{10}$$

Substituting for τ and ν in (9) yields

$$(\Omega_i\cos\alpha + \Omega_j\sin\alpha)^2 + (\Omega_k + d + \dot{\alpha})^2 = e^2. \tag{11}$$

Thus three bitangents yield three quadratic equations in $\Omega_i, \Omega_j, \Omega_k$. This system of equations can be solved using a global numerical method such as homotopy continuation [13]. Substituting the corresponding solutions into (10) and (9) yields the values of τ, ν and θ for the parallel tangent features in each image.

4 Motion Constraints from Infinitesimal Frontier Points

Now the remaining difficulty is to estimate the linear velocity of some feature point (e.g. an endpoint of a bitangent or a parabolic point), and we turn to constraints derived from the motion of the infinitesimal frontier points. See Fig. 4. We choose this feature point P_0 as the origin of a coordinate system, and

estimate its velocity $\dot{P}_0$ by establishing three linear constraints on $\dot{P}_0$.

First, we note that $\dot{P}_0$ lies in the tangent plane of the surface since the trace of $P_0(t)$ is a surface curve. Thus, we immediately have the constraint $\dot{P}_0 \cdot \boldsymbol{n} = 0$.

As discussed in Section 1.1, the frontier points lie on a surface curve determined by the camera motion. For a fixed surface and moving camera under orthographic projection where the viewing direction is $\boldsymbol{v}(t)$ and the velocity of $\boldsymbol{v}$ is $\dot{\boldsymbol{v}} = \boldsymbol{\Omega} \times \boldsymbol{v}$, the frontier points define a curve on the surface satisfying

$$\begin{cases} \boldsymbol{n}_f \cdot \boldsymbol{v} = 0, \\ \boldsymbol{n}_f \cdot \dot{\boldsymbol{v}} = 0, \end{cases} \tag{12}$$

where $\boldsymbol{n}_f$ denotes the surface normal at the frontier point F. That is, they lie on the occluding contour, and the surface normal is orthogonal to $\dot{\boldsymbol{v}}$. This allows us to detect the frontier points in an image. If we define the vector $\boldsymbol{t}_f = \boldsymbol{n}_f \times \boldsymbol{v}$, the vectors $\boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v}$ form an orthonormal basis of $\mathbb{R}^3$, and $\boldsymbol{t}_f$ is tangent to the image contour at the projection of F onto the image plane.

Let $\boldsymbol{\omega}_f$ denote the rotational velocity associated with the basis $(\boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$. As before, the $(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{v})$ and $(\boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$ bases differ by a rotation about $\boldsymbol{v}$ with angle β, that maps $\boldsymbol{i}$ onto $\boldsymbol{t}_f$. The angular velocity $\boldsymbol{\omega}_f$ of the $(\boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$ basis is related to the angular velocity $\boldsymbol{\Omega}$ of the camera by:

$$\boldsymbol{\omega}_f = \boldsymbol{\Omega} + \dot{\beta}\boldsymbol{v}.$$

Thus we can compute $\boldsymbol{\omega}_f$ from measurements. Recalling from (12) that $\boldsymbol{n}_f \cdot \dot{\boldsymbol{v}} = 0$, we have

$$(\boldsymbol{\omega}_f \times \boldsymbol{v}) \cdot \boldsymbol{n}_f = 0 \Longleftrightarrow \boldsymbol{\omega}_f \cdot \boldsymbol{t}_f = 0.$$

That is, $\boldsymbol{\omega}_f$ lies in a plane spanned by $\boldsymbol{n}_f$ and $\boldsymbol{t}_f$ and can be expressed as: $\boldsymbol{\omega}_f = \chi_f \boldsymbol{n}_f + \nu_f \boldsymbol{v}$ where χ_f and ν_f are known. Applying $\boldsymbol{\omega}_f$ to $\boldsymbol{n}_f$, we have $\dot{\boldsymbol{n}}_f = -\nu_f \boldsymbol{t}_f$. Now, the image coordinates (x_f, y_f) of a frontier point F can be expressed in the $(P_0, \boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$ frame as:

$$\begin{aligned} x_f &= (F - P_0) \cdot \boldsymbol{t}_f \\ y_f &= (F - P_0) \cdot \boldsymbol{n}_f. \end{aligned}$$

Differentiating y_f with respect to time, noting that $\dot{F} \cdot \boldsymbol{n}_f = 0$, and simplifying yields

$$\dot{P}_0 \cdot \boldsymbol{n}_f = -\nu_f x_f - \dot{y}_f.$$

Since ν_f, x_f and $\dot{y}_f$ are known, we can compute $\dot{P}_0 \cdot \boldsymbol{n}_f$.

A second image sequence containing the projection of P_0 is used to establish a third constraint on $\dot{P}_0$ (i.e., P_0 is a stereo frontier point of both images). This second image could have been used to establish one of the constraints on $\boldsymbol{\Omega}$. If $\boldsymbol{n}'_f$ is the normal to infinitesimal epipolar plane at time t' in the second image sequence, then we can determine $\dot{P}_0(t') \cdot \boldsymbol{n}'_f$ as above.

Now, the only problem is to relate $\dot{P}_0(t') \cdot \boldsymbol{n}'_f$ to $\dot{P}_0(t)$. To do this, we re-parameterize the second image sequence by the same time parameter as the first one using the arc length s of the matching invariant curve.

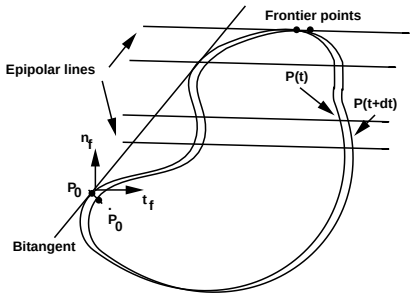

Figure 5: The entire occluding contour can reconstructed using the epipolar parameterization once $\boldsymbol{\Omega}$ and $\dot{P}_0$ are known.

Algebraically, if t' denotes the time associated with the second sequence, we have

$$\frac{dP_0}{dt'} = \frac{dt}{dt'}\frac{dP_0}{dt} = \frac{dt}{ds}\frac{ds}{dt'}\frac{dP_0}{dt} = \frac{|d\boldsymbol{K}/dt'|}{|d\boldsymbol{K}/dt|}\frac{dP_0}{dt}.$$

Thus we can relate components $\dot{P}_0$ measured in both coordinate systems. In particular, we can use the second image sequence to measure $\dot{P}_0 \cdot \boldsymbol{n}'_f$, and since we have the additional constraint $\dot{P}_0 \cdot \boldsymbol{n} = 0$, we obtain a system of three linear constraints which uniquely determines the velocity $\dot{P}_0$ of P_0.

5 Occluding Contour Reconstruction

Up to this point, the angular velocity of the camera $\boldsymbol{\Omega}$ and the linear velocity $\dot{P}_0$ of a feature P_0 have been determined. We now reconstruct the whole occluding contour using the epipolar parameterization of the surface in a coordinate system whose origin is at P_0 and whose axes are $(\boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$.

Let P be a surface point with coordinates (x, y, z) in the coordinate system $(P_0, \boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$. By definition, the epipolar curves are everywhere tangent to the viewing direction [4, 8]. In other words $\dot{P} \times \boldsymbol{v} = 0$ or $\dot{P} \cdot \boldsymbol{t}_f = 0$ and $\dot{P} \cdot \boldsymbol{n}_f = 0$. See Figure 5. Recalling that $\boldsymbol{\omega}_f = \chi_f \boldsymbol{n}_f + \nu_f \boldsymbol{v}$, the velocity of the image coordinates of P is:

$$\begin{cases} \dot{x} = \nu_f y - \chi_f z - \dot{P}_0 \cdot \boldsymbol{t}_f, \\ \dot{y} = -\nu_f x - \dot{P}_0 \cdot \boldsymbol{n}_f. \end{cases} \tag{13}$$

The second equation can be used to establish correspondences between points in the given image and the next one in the sequence. For a point (x, y) on the silhouette at time t, the corresponding point in the image at time $t + \delta t$ must lie along the line parallel to $\boldsymbol{t}_f$ whose coordinate along the $\boldsymbol{n}_f$ axis is $y + \dot{y}\delta t$. Once a match is determined, $\dot{x}$ in the $\boldsymbol{t}_f, \boldsymbol{n}_f$ coordinate system can be measured. Solving the first equation in (13) yields the depth z, of the reconstructed point.

$$z = \frac{1}{\chi_f}(-\dot{x} + \nu_f y - \dot{P}_0 \cdot \boldsymbol{t}_f). \tag{14}$$

An alternative method for computing the depth once the motion has been determined is presented in [15] and may be more robust.

Equation (14) is used to reconstruct the coordinates of points on the occluding contour for a single image in the $(P_0, \boldsymbol{t}_f, \boldsymbol{n}_f, \boldsymbol{v})$ frame. The orientation $R(t)$ of the camera frame can be computed from $\boldsymbol{\Omega}$ through integration over the entire sequence of images. Furthermore, $\dot{P}_0$ can be integrated to obtain $P_0(t)$. Once $P_0(t)$ and $R(t)$ are expressed in a common coordinate system, the reconstructed occluding contour each image can also be written in the common system.

6 Implementation and Results

The presented method has been completely implemented in Common Lisp, and applied to a synthetic sequence to validate the implementation and determine the effects of noise. In previous work, we have constructed the invariant curves from real images and used them for recognition [19]. The scene is composed of four spheres shown in Fig. 6.a. Noiseless images were generated using graphics techniques, and the camera trajectory was sampled at one degree increments. All of the images are composed of four circles. Between any two non-intersecting circles, there are four bitangents (two outer and two inner) and a total of 24 bitangents for the four spheres; for each bitangent, there are six parallel tangents. Of these, three bitangents and three associated parallel tangents are shown in Fig. 6.b.

The camera's trajectory can be divided into four subsequences. The primary trajectory, in which reconstruction was performed, corresponds to rotating the viewing direction about the horizontal axis. In three additional subsequences, seven bitangent developables are fully revealed and matched using the invariant curves. Figure 6.c shows the invariant curves computed from the four subsequences for the three bitangents. For the synthetic image shown in Fig. 6.b, the correspondences were determined for each of the bitangents. The bitangent and frontier points shown in Fig. 6.d and the features with horizontal tangents in Fig. 6.b are stereo frontier points of the two images. As discussed in Section 3, these correspondences are used to compute the camera's angular velocity.

From the computed angular velocity $\boldsymbol{\Omega}$, the epipolar geometry can be established, and the eight frontier points shown in Fig 6.e were detected. $\dot{P}_0$ is then computed from the velocity of these eight frontier points. Next, the occluding contour for a single image is reconstructed and shown from two orthogonal viewpoints in Figs. 6.f,g. As expected for spheres, the occluding contour is circular and lies in a plane. Finally, the reconstructed contours for the entire sequence are placed in a common coordinate system. Figures 6.h,i show two orthogonal views of the reconstructed surface along with the trace of the viewing direction on a sphere. Because the primary trajectory covers 35°, only a portion of the sphere can be reconstructed. Since $\boldsymbol{\Omega}$ is constant over the camera trajectory, the viewing direction $\boldsymbol{v}$ lies in a plane, and the frontier curve degenerates to a point. Consequently, the reconstructed surface is like an orange slice; each reconstructed occluding contour defines a meridian of a reconstructed sphere, and the poles are frontier points.

Simulation experiments were conducted to determine the effects of noise on reconstruction using synthetic images with a resolution of 512 by 512 pixels. Random noise with a uniform distribution was added to the pixel coordinates of detected features. Since the accuracy of feature detection is anisotropic, the ratio of the magnitude of the noise in the normal and tangential directions of the contours was varied as well as the size of the interval. Statistics were gathered from approximately 200 trials, and the performance was compared to ground truth. Figure 7.a shows mean error in the direction of $\boldsymbol{\Omega}$. For noise levels ranging between $\pm.25$ pixels and ± 2 pixels, the error in $\boldsymbol{\Omega}$ ranged from 2° to 16°. Figures 7.b,c show the mean and variance of the computation of θ; note that the error is nearly zero mean, but that the variance increases with noise. Finally, the mean error of the location of detected frontier points is illustrated in Fig. 7.d.

7 Discussion

In this paper, we have presented a method for recovering the motion of a camera and structure of a curved 3D object from a sequence of silhouettes detected in images. The key observation is that under orthographic projection, stereo frontier points and the infinitesimal motion frontier points establish constraints on the observer motion. The invariant curve representation, that was previously introduced for recognition [19], can be used to establish correspondences between nonconsecutive images in the sequence. Once the observer motion has been computed, the 3-D Euclidean structure can be recovered using established techniques.

It is well known that structure and motion can only be recovered up to a common scale factor. In this case, the common factor is ξ which we have taken to be unity (see Equation 1). Thus, if ξ is known, the surface structure and the entire camera trajectory are completely determined; otherwise, they are only computed up to an unknown factor.

The method has also be extended to weak perspective where the scale parameter ξ is also varying with time. To date, we have just tested our implementation on synthetic images. While we are anxious to apply it to real images, the noise analysis indicates that reasonably accurate reconstruction will only be possible if features can be detected with sub-pixel resolution.

Acknowledgments: This work was supported in part by the National Science Foundation under Grant IRI-9224815. D. Kriegman was supported in part by a National Science Foundation NYI Grant IRI-9257990. Jean Ponce was supported in part by the Center for Advanced Study of the University of Illinois at Urbana-Champaign.

References

[1] E. Arbogast and R. Mohr. 3D structure inference from image sequences. *Journal of Pattern Recognition and Artificial Intelligence*, 5(5), 1991.

[2] E. Boyer and M. Berger. 3D surface reconstruction from occluding contours. 1995. Preprints.

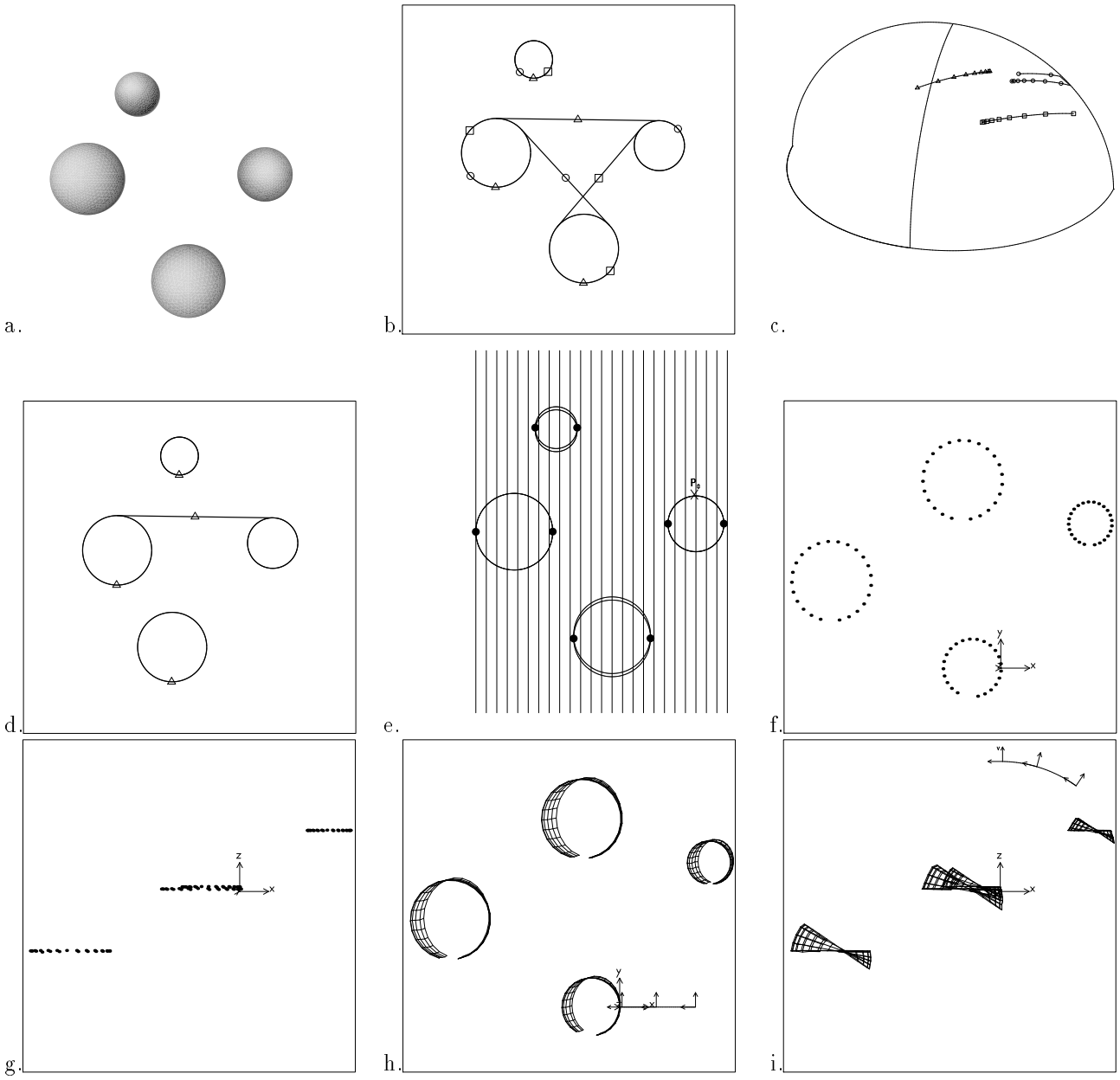

Figure 6: **Reconstruction from a simulated image sequence: a**. The scene; **b**. A synthetic image from the sequence, three bitangents, and the parallel tangents used for reconstruction; **c**. The invariant curves computed from the tracks of the three bitangents and corresponding parallel tangents shown in Fig. 6.a; **d**. The bitangent and parallel tangents in this image and the corresponding one in the image in Fig. 6.b are stereo frontier points, and have been found using the invariant curves; **e**. Two consecutive images in the sequence after aligning their epipolar lines, the infinitesimal epipolar geometry, and the frontier points. In parts **b**,**c**, and **d**, the square, triangle and circles indicate corresponding sets of feature points; **f,g**. Two views of the reconstructed occluding contour from just one image; **h,i**. The reconstructed surface seen from two views along with the camera trajectory and frames. The origin of the coordinate system in parts f,g,h, and i is $\mathbf{P_0}$.

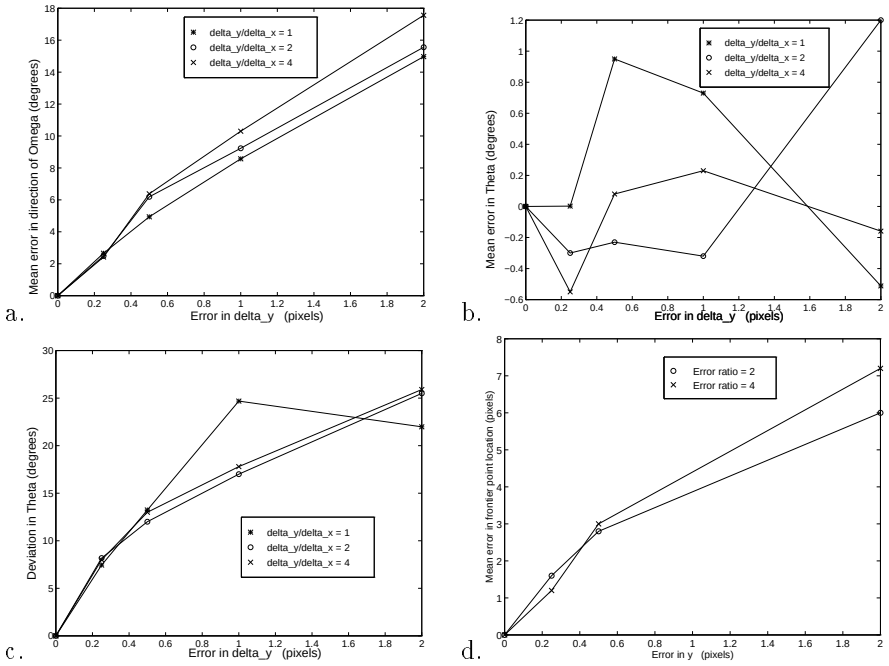

Figure 7: **Perturbation analysis**: Independent uniformly distributed noise $U(-\Delta x, +\Delta x)$ and $U(-\Delta y, +\Delta y)$ was added to the feature coordinates in the tangential and normal directions. The three curves in each plot represent three ratios of Δx and Δy (e.g., $\Delta x = 2\Delta y$). **a**. Mean error in the direction of $\boldsymbol{\Omega}$ compared to its ground truth; **b-c**. Mean and standard deviation of θ; **d**. Mean error in the location of frontier points.

[3] R. Cipolla, K. Aström, and P. Giblin. Motion from the frontier of curved surfaces. In *Int. Conf. on Computer Vision*, pages 269–275, 1995.

[4] R. Cipolla and A. Blake. Surface shape from the deformation of the apparent contour. *Int. J. Computer Vision*, 9(2):83–112, November 1992.

[5] O. Faugeras. *Three Dimensional Computer Vision*. MIT Press, 1993.

[6] P. Giblin, F. Pollick, and J. Rycroft. Recovery of an unknown axis of rotaytion from the profiles of a rotating surface. *J. Opt. Soc. Am.*, A11:1976–1984, 1994.

[7] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. In *Int. Conf. on Computer Vision*, pages 136–144, London, U.K., 1987.

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

[9] B. Horn. *Computer Vision*. MIT Press, Cambridge, Mass., 1986.

[10] T. Joshi, N. Ahuja, and J. Ponce. Structure and motion estimation from dynamic silhouettes under perspective projection. In *Int. Conf. on Computer Vision*, pages 290–295, 1995.

[11] J. J. Koenderink. *Solid Shape*. MIT Press, Cambridge, MA, 1990.

[12] K. Kutulakos and C. Dyer. Global surface reconstruction by purposive control of observer motion. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, 1994.

[13] A. Morgan. *Solving Polynomial Systems using Continuation for Engineering and Scientific Problems*. Prentice Hall, Englewood Cliffs, 1987.

[14] J. H. Rieger. Three-dimensional motion from fixed points of a deforming profile curve. *Opt. Lett.*, 11:123–125, 1986.

[15] R. Szeliski and R. Weiss. Robust shape recovery from occluding contour using a linear smoother. Technical Report CRL-93/7, DEC, December 1993.

[16] C. Taylor and D. Kriegman. Structure and motion from line segments in multiple images. *IEEE Trans. Pattern Anal. Mach. Intelligence*, Oct. 1995.

[17] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *Int. J. Computer Vision*, 9(2):137–154, 1992.

[18] R. Vaillant and O. Faugeras. Using extremal boundaries for 3D object modeling. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 14(2):157–173, February 1992.

[19] B. Vijayakumar, D. Kriegman, and J. Ponce. Invariant-based recognition of complex 3D curved objects from image contours. In *Int. Conf. on Computer Vision*, 1995.

[20] B. Vijayakumar, D. Kriegman, and J. Ponce. Structure and motion of curved 3d objects from monocular silhouettes. Technical Report 9514, Yale Center for Systems Science, 1995. Available via anonymous ftp on daneel.eng.yale.edu.

[21] A. Zisserman, D. Forsyth, J. Mundy, and C. Rothwell. Recognizing general curved objects efficiently. In Mundy and Zisserman, editors, *Geometric Invariance in Computer Vision*. MIT Press, 1992.