THE DYNAMIC ANALYSIS OF APPARENT CONTOURS

Roberto Cipolla
Andrew Blake

Reprinted from PROCEEDINGS OF THE THIRD INTERNATIONAL CONFERENCE ON COMPUTER VISION, Osaka, Japan, December 4-7, 1990

IEEE Computer Society
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1264

Washington, DC • Los Alamitos • Brussels • Tokyo

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

IEEE COMPUTER SOCIETY

The Dynamic Analysis of Apparent Contours

Roberto Cipolla Andrew Blake

University of Oxford
Department of Engineering Science
Parks Road
Oxford OX1 3PJ, UK.

Abstract

Robots perambulating and manipulating unmodelled environments need robust geometric cues to recover scene structure. It is furthermore attractive to capitalise on structural information inherent in the evolution of the image under robot motion. However, especially in artificial environments, surface texture may be sparse, and silhouettes or apparent contours are the dominant image features. We develop previous theories of the analysis of deformation of apparent contours under viewer motion. First, earlier results showing how surface curvature can be inferred from *acceleration* of image features are generalised for arbitrary viewer motion and perspective projection. Second, we show that *relative* image acceleration, based on parallax measurements, is robust to uncertainties in robot motion. Thirdly, our theory has been implemented and extensively tested in a realtime (15 frames per second) tracking system based on deformable contours (snakes). We show that focusing attention by means of snakes enables rapid, robust computation of surface curvature, including discrimination of extremal and occluding contours.

1 Introduction

The deformation of an apparent contour (the silhouette of a smooth surface or the image of the extremal boundary) under viewer-motion is a potentially rich source of geometric information for navigation, motion-planning and object-recognition. Barrow and Tenenbaum [1] pointed out that surface orientation along an extremal boundary can be computed directly from image data. Koenderink [15] related the curvature of an apparent contour to the intrinsic curvature of the surface (Gaussian curvature); the sign of Gaussian curvature is equal to the sign of the curvature of the contour. Convexities, concavities and inflections of an apparent contour indicate, respectively, convex, hyperbolic and parabolic surface points. Giblin and Weiss [9] have extended this by adding viewer motions to obtain quantitative estimates of surface curvature. For orthographic projection they show that a surface can be reconstructed from the envelope of all its tangent planes, which in turn are computed directly from the family of silhouettes of the surface, obtained under planar motion of the viewer. In Blake and Cipolla (1989) [3] this was extended to the general case of arbitrary non-planar, curvilinear camera motion under perspective projection.

In this paper we summarise further developments of the theory and describe the implementation and results with a camera mounted on a moving robot arm. We describe a simple, computationally efficient method for accurately extracting image curves from real images and tracking their temporal evolution. This is an extension of tracking with *Snakes* [14] - energy minimising splines guided by image forces - in which we avoid computing the internal energies by representing sections of curves by cubic B-splines and can achieve realtime processing (15 frames per second) by windowing [13]. Experiments show that with adequate viewer motion calibration it is possible to obtain 3D shape measurements of useful accuracy.

A consequence of the theory, representing an important step towards qualitative vision, concerns the robustness of measurements of curvature based on *motion parallax* at two nearby points. Intuitively it is relatively difficult to judge, moving around a smooth, featureless object, whether its silhouette is extremal or not — whether curvature along the contour is bounded or not. This judgment is much easier to make for objects which have at least a few surface features. Under small viewer-motions, features are "sucked" over the extremal boundary, at a rate which depends on surface curvature. Our theoretical findings exactly reflect the intuition that the "sucking" effect is a reliable indicator of relative curvature, regardless of the exact details of the viewer's motion. Relative measurements of curvature across two adjacent points are entirely immune to uncertainties in the viewer's rotational velocity. This is somewhat related to earlier results showing that relative measurements of this kind are important for depth measurement from optic flow [16, 19, 21] and for curvature measurements from stereoscopically viewed highlights [2]. Furthermore, they are relatively immune to uncertainties in translational motion in that, unlike single-point measurements, they are independent of the viewer's acceleration. Only dependence on velocity remains. Experiments show that this theoretical prediction is borne out in practice. Surface curvature estimated from parallax measurements prove to be more than an order of magnitude less sensitive than single-point measurements to errors in viewer-motion calibration.

As an illustration of their power, we show how these motion analysis techniques can achieve something which has so far eluded photometric analysis: namely reliable discrimination between fixed surface features and points on extremal boundaries and the reconstruction of surfaces in the vicinity of their extremal boundaries.

CH2934-8/90/0000/0616$01.00 © 1990 IEEE

2 Theoretical framework

It is now well established that static views of extremal boundaries are rich sources of surface geometry [1, 5, 9, 15]. The physical constraints of *tangency* (all rays at an extremal boundary are in the surface's tangent plane) and *conjugacy* (the extremal boundary is not in general orthogonal to the ray direction but *conjugate* to it) allow the recovery of surface orientation and the sign of Gaussian curvature directly from the image of the extremal boundary, the apparent contour [3].

Moreover, each vantage point, generates a new extremal boundary making it an ideal cue in the active exploration of 3D geometry. We outline below how the deformation of apparent contours under known viewer-motion can be used to recover the position, orientation and full surface curvature (3D shape) of visible surfaces in the vicinity of their extremal boundaries.

2.1 Surface Geometry

Consider a point P on the extremal boundary of a smooth, curved surface which we represent locally by a vector valued function $\mathbf{r}(s,t)$ with parameters s and t. The parametric representation can be considered as covering the surface with 2 families of curves: $\mathbf{r}(s,t_0)$, and $\mathbf{r}(s_0,t)$ where s_0, t_0 are fixed for a given curve in the family. A one-parameter family of views is indexed by the time parameter t and s,t are defined so that the s-parameter curve, $\mathbf{r}(s,t_0)$, is an extremal boundary for a particular view t_0 (figure 1).

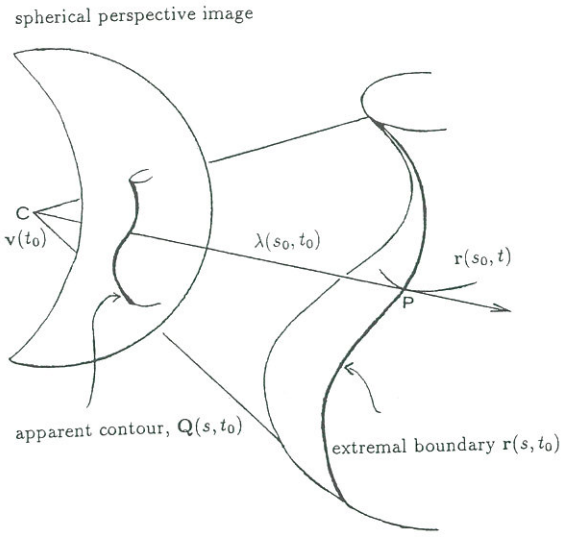

Figure 1. Surface and Viewing Geometry.
P lies on a smooth surface which is parameterised locally as $\mathbf{r}(s,t)$. *For a given vantage point,* $\mathbf{v}(t_0)$, *the family of rays emanating from the viewer's optical centre (C) that touch the surface defines an s-parameter curve* $\mathbf{r}(s,t_0)$ *– the extremal boundary from vantage point* t_0. *The spherical perspective projection of this extremal boundary – the apparent contour,* $\mathbf{Q}(s,t_0)$ *– determines the direction of rays which graze the surface. The distance along the ray, CP, is* $\lambda(s_0,t_0)$.

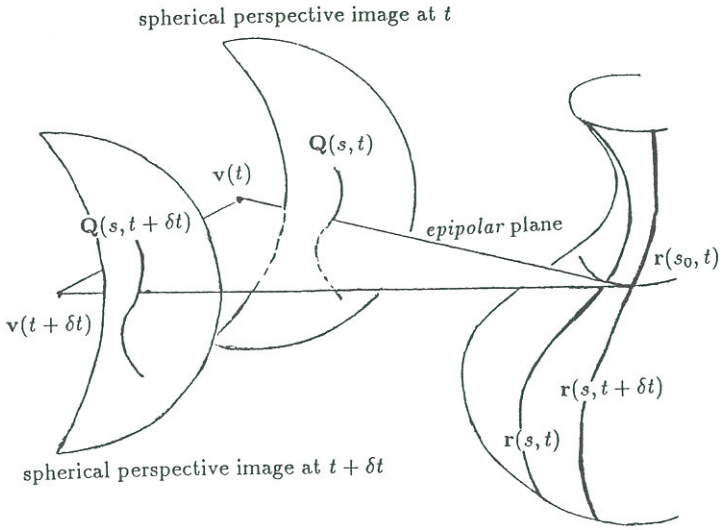

Figure 2. Epipolar parameterisation
A moving observer at position $\mathbf{v}(t)$ *sees a family of extremal boundaries* $\mathbf{r}(s,t)$ *indexed by the time parameter t. Their spherical perspective projections are represented by a 2 parameter family of apparent contours* $\mathbf{Q}(s,t)$. *For the epipolar parameterisation t-parameter curves* ($\mathbf{r}(s_0,t)$ *and* $\mathbf{Q}(s_0,t)$) *are defined by chosing the correspondence between successive contours to be in an epipolar plane which is determined by the translational velocity and the direction of the ray.*

A t-parameter curve $\mathbf{r}(s_0,t)$ can be thought of as the 3D locus of points grazed by a light-ray from the viewer, under viewer-motion. Such a locus is not uniquely defined. The correspondence, as the viewer moves, between "successive" (in an infinitesimal sense) extremal boundaries is not unique. Hence there is considerable freedom to choose a spatio-temporal parameterisation of the surface, $\mathbf{r}(s,t)$. A natural choice of parametrisation, it has been shown [3], is the *epipolar parameterisation* in which points on successive extremal boundaries are matched in the epipolar plane defined by the ray direction and the instantaneous viewer translational velocity (figure 2). The advantage of the parametrisation is clear below, when it leads to a simplified treatment of surface curvature and a unified treatment of the projection of rigid space curves and extremal boundaries.

Surface curvature (3D shape) can be expressed in terms of the coefficients of the first and second fundamental forms $\mathbf{I}, \mathbf{II}$ [8]. For the *epipolar* parametrisation, these are:

$$\mathbf{I} = \begin{bmatrix} 1 & cos\theta \\ cos\theta & 1 \end{bmatrix} \tag{1}$$

$$\mathbf{II} = \begin{bmatrix} \kappa^t & 0 \\ 0 & \kappa^s \end{bmatrix}, \tag{2}$$

where θ is the angle between the ray direction and the extremal boundary; κ^t is the normal curvature of the t-parameter curve $\mathbf{r}(s_0,t)$) and κ^s is the normal curvature of the extremal boundary $\mathbf{r}(s,t_0)$ at P. Equivalently κ^t is the curvature of the normal section at P in the direction of the ray. Note that $\mathbf{II}$ is diagonal, a result of choosing, in the epipolar parametrisation, basis directions that are *conjugate* [7].

In the following we relate the surface geometry at a point P with its spatio-temporal image and then proceed to show how the components of **I** and **II** can be recovered.

2.2 Imaging model and Spatio-temporal image

The imaging model is a spherical pin-hole camera of unit radius. A point on a visible surface , $\mathbf{r}(s,t)$, projects to a (unit) vector $\mathbf{Q}(s,t)$ on the image sphere:

$$\mathbf{r}(s,t) = \mathbf{v}(t) + \lambda(s,t)R(t)\mathbf{Q}(s,t), \tag{3}$$

where, at time t, $\mathbf{v}(t)$ is the position of the viewer, $\lambda(s,t)$ is the distance along the ray from the viewer to the point $\mathbf{r}$ and $R(t)$ is a rotation operator describing the orientation of the camera frame relative to the world frame. The viewer motion is assumed rigid, involving translational motion $\dot{\mathbf{v}}$ and rotational motion $\mathbf{\Omega}$ defined by:

$$(\mathbf{\Omega}\wedge)\cdot = \dot{R}$$

where $\dot{}$ denotes differentiation with respect to time and $\wedge$ denotes a vector product.

For a given vantage position $\mathbf{v}(t_0)$ the apparent contour is a continuous family of rays $\mathbf{Q}(s,t_0)$ emanating from the camera's optical centre which touch the surface so that

$$\mathbf{Q}.\mathbf{n} = 0. \tag{4}$$

where $\mathbf{n}$ is the surface normal (figure 1).

The moving observer at position $\mathbf{v}(t)$ sees a family of apparent contours indexed by time, $\mathbf{Q}(s,t)$, swept out over the image sphere (figure 2). As before the spatio-temporal parameterisation of the family is not unique. The mapping between extremal boundaries, and hence between apparent contours, at successive instants is undetermined. With the *epipolar* parameterisation a natural *correspondence* between points on successive snapshots of an apparent contour can be set up. The lines of constant s on the image sphere are defined so that the tangent to the t-parameter curves, $\mathbf{Q}(s_0,t)$, are given by [7]

$$\dot{\mathbf{Q}} = \frac{(\dot{\mathbf{v}} \wedge \mathbf{Q}) \wedge \mathbf{Q}}{\lambda} - \mathbf{\Omega} \wedge \mathbf{Q} \tag{5}$$

Note that $\dot{\mathbf{Q}}$ is equal to the image velocity of a point on the projection of a static space curve [18]. This is not surprising since instantaneously image velocities are dependent only on depth and not surface curvature. Points on different image contours are matched by moving along great-circles on the image sphere with poles defined by the direction of the viewer's instantaneous translational velocity $\dot{\mathbf{v}}$. If the motion is linear corresponding points on the image sphere will lie on an epipolar great-circle (or an epipolar line for planar image geometry). For a general motion, however, the epipolar structure rotates continuously as the direction of $\dot{\mathbf{v}}$ changes and the space curve, $\mathbf{r}(s_0,t)$, generated by the movement of a contact point will be non-planar.

3 Dynamic properties of the apparent contour

Depth, λ; the normal curvature κ^t along the line of sight; and the curvature of the apparent contour, κ^p are in fact sufficient to compute the full surface curvature at a point [7]. These can all be recovered from the apparent contour and its deformation with viewer motion.

3.1 Depth from image velocities

When the viewer executes a *known* motion then surface depth can, of course, be computed from image velocities [4] [12]. This is correct for static space curves but it also holds for extremal boundaries even though they are not fixed in space. From (5) we see that depth can be computed from the normal component of image velocity, $\dot{\mathbf{Q}}.\mathbf{n}$, if the angular velocity is known:

$$\lambda = -\frac{\dot{\mathbf{v}}.\mathbf{n}}{\dot{\mathbf{Q}}.\mathbf{n} + (\mathbf{\Omega} \wedge \mathbf{Q}).\mathbf{n}} \tag{6}$$

This formula is an infinitesimal analogue of triangulation with stereo cameras. The numerator is analogous to baseline and the denominator to disparity. In the infinitesimal limit stereo will correctly determine the depth of an extremal boundary.

3.2 Normal curvature along the line of sight from image accelerations

The normal curvature at P in the direction of the ray $\mathbf{Q}$, κ^t, can be computed from image accelerations, $\ddot{\mathbf{Q}}$, at the apparent contour if the viewer motion (both translational and rotational velocity and acceleration) is known:

$$\begin{aligned}\ddot{\mathbf{Q}}.\mathbf{n} \;=\; & -\frac{(\dot{\mathbf{v}}.\mathbf{n})^2}{(\lambda)^3}\left[\frac{1}{\kappa^t}\right] - 2\frac{(\mathbf{Q}.\dot{\mathbf{v}})(\dot{\mathbf{v}}.\mathbf{n})}{(\lambda)^2} - \frac{\ddot{\mathbf{v}}.\mathbf{n}}{\lambda} \\ & -(\dot{\mathbf{\Omega}} \wedge \mathbf{Q}).\mathbf{n} - \frac{2(\mathbf{Q}.\dot{\mathbf{v}})(\mathbf{\Omega} \wedge \mathbf{Q}).\mathbf{n}}{\lambda} \\ & +\frac{(\mathbf{\Omega} \wedge \dot{\mathbf{v}}).\mathbf{n}}{\lambda} + (\mathbf{\Omega}.\mathbf{Q})(\mathbf{\Omega}.\mathbf{n})\end{aligned} \tag{7}$$

The magnitude of κ^t can be used to distinguish surface markings or creases from extremal boundaries. A fixed feature is simply a degenerate case of the parameterisation and can be considered as the limiting case of a surface point with infinite curvature ($1/\kappa^t = 0$). The sign of κ^t is an important geometric cue since it determines the "sidedness" of the extremal boundary – on which side of the image contour lies the surface.

3.3 Local Surface Curvature

The remaining the components of **I** and **II** now follow.

The *normal* curvature of the extremal boundary, κ^s, can be recovered from the curvature of the apparent contour, κ^p.

The two curvatures are simply related by [7]

$$\kappa^p = \lambda \frac{\kappa^s}{\sin^2\theta}. \tag{8}$$

where depth, λ and θ are recovered from image velocities along the apparent contour via (6).

The sign and magnitude of the Gaussian curvature, K, can then be computed from (1) and (2) as the product of the normal curvature, κ^t, and the curvature of the apparent contour, κ^p, scaled by inverse-depth [15]

$$K = \frac{\kappa^p \kappa^t}{\lambda} \tag{9}$$

4 Motion parallax and robust estimation of surface curvature

Although it is perfectly feasible to compute surface curvature from the observed deformation of an apparent contour, we have seen that this requires knowledge of the viewer's translational and rotational velocities and accelerations. Moreover the result (see below) is highly sensitive to motion calibration errors. This may be acceptable for a moving camera mounted on a precision robot-arm or when a grid is in view so that accurate visual calibration can be performed [20]. In such cases it is feasible to determine motion to the accuracy of around 1 part in 1000 that is required. However, when only crude estimates of motion are available another strategy is called for. One possibility is refine the estimate of motion using image velocities [10]. The alternative approach is to seek qualitative measurements of geometry that are much less sensitive to perturbations in assumed egomotion.

In this section we show that the difference of the image motion of a pair of nearby points can be much more robust as a cue than the absolute image motion at a point. Differences of measurements at two points are insensitive to errors in rotation and in translational acceleration. Typically, the two features might be one point on an extremal boundary and one fixed surface point. The surface point can be considered as having infinite curvature and therefore acts simply as a stable reference point for the measurement of curvature at the extremal boundary. Intuitively the reason for the insensitivity of relative estimates of curvature is that global additive errors in motion measurement are cancelled out.

4.1 Motion parallax

Consider two visual features whose projections on the image sphere are $\mathbf{Q}(s_i, t)$, $i = 1, 2$ which we will abbreviate to $\mathbf{Q}^{(i)}$, $i = 1, 2$. Think of them as two points on extremal boundaries, which trace out curves with (normal) curvatures κ^{t1} and κ^{t2} as the viewer moves. The relative image displacement $\boldsymbol{\delta}$ of the two points is

$$\boldsymbol{\delta}(t) = \mathbf{Q}^{(2)} - \mathbf{Q}^{(1)}. \tag{10}$$

Parallax is the temporal derivative, $\dot{\boldsymbol{\delta}}$, of $\boldsymbol{\delta}$.

If instantaneously the two points project to the same point on the image sphere, so that

$$\mathbf{Q}^{(1)} = \mathbf{Q}^{(2)} = \mathbf{Q},$$

then, from (5), the parallax $\dot{\boldsymbol{\delta}}$ depends only on their relative inverse-depths and on viewer velocity. It is independent of angular rotation $\boldsymbol{\Omega}$:

$$\dot{\boldsymbol{\delta}} = (\dot{\mathbf{v}} \wedge \mathbf{Q}) \wedge \mathbf{Q} \left[\frac{1}{\lambda^{(2)}} - \frac{1}{\lambda^{(1)}} \right] \tag{11}$$

The use of "Motion Parallax" [11] for robust determination of the direction of translation $\dot{\mathbf{v}}$ and of relative depths from image velocities was described by Longuet-Higgins and Prazdny [16] and Rieger and Lawton [19].

4.2 Rate of parallax

Following from the well known results about motion parallax, we derive the central result of this paper — that the *rate* of parallax is a robust cue for surface curvature. The direct formula (7) for normal curvature κ^t in terms of image acceleration was sensitive to viewer translational acceleration and rotational velocity and acceleration. If instead *differences* of image accelerations are used the undesirable sensitivity vanishes.

Equation (7) expresses the relationship between image acceleration and normal curvature. It can be written for both points $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$. The important point is that the two copies of this equation for the two positions can be subtracted, cancelling off the undesirable dependency on $\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}$ and on $\ddot{\mathbf{v}}$.

Let us define the *differential curvature* $\Delta\kappa^t$ of the feature pair by

$$\frac{1}{\Delta\kappa^t} = \frac{1}{\kappa^{t1}} - \frac{1}{\kappa^{t2}}. \tag{12}$$

Note that it is not an infinitesimal quantity but a difference of inverse curvature. Consider the two features to be instantaneously spatially coincident, that is, initially, $\mathbf{Q}(s_1, t) = \mathbf{Q}(s_2, t)$. Moreover assume they lie at a common depth λ, and hence, instantaneously, $\dot{\mathbf{Q}}^{(1)} = \dot{\mathbf{Q}}^{(2)}$. In practice, of course, the feature pair will only coincide exactly if one of the points is a surface marking which is instantaneously on the extremal boundary. The effect of a small separation is analysed below. Now, taking the difference of equation (7) for the 2 image positions leads to the following relation between the two *differential* quantities; the "rate of parallax", $\ddot{\boldsymbol{\delta}}$ and the *differential* curvature $\Delta\kappa^t$:

$$\ddot{\boldsymbol{\delta}}.\mathbf{n} = \frac{(\dot{\mathbf{v}}.\mathbf{n})^2}{\lambda^3} \frac{1}{\Delta\kappa^t}. \tag{13}$$

Differential curvature $\Delta\kappa^t$ can be computed from depth λ, viewer velocity $\dot{\mathbf{v}}$, and the "rate of parallax". *Dependence on viewer motion is now limited to the translational velocity* $\dot{\mathbf{v}}$. There is no dependence on viewer acceleration or rotational velocity. Hence the relative measurement should be much more robust. (Higher derivatives are generally far more sensitive, computationally, to noise.)

In the case that $\mathbf{Q}^{(1)}$ is known to be a fixed surface reference point, with $1/\kappa^{t1} = 0$, then $\Delta\kappa^t = \kappa^{t2}$ so that the *differential* curvature $\Delta\kappa^t$ constitutes an estimate, now much more robust, of the normal curvature κ^{t2} at the extremal boundary point $\mathbf{Q}^{(2)}$. Of course this can now be used in detecting extremal boundaries; determining the "sidedness" (on which side of the image contour lies the surface) and in equations (1) and (2) to obtain robust estimates of surface curvature. Our experiments confirm this, as shown below.

4.3 Degradation of sensitivity with separation of points

The theory above relating *differential curvature* to rate of parallax assumed that the two points $\mathbf{Q}^{(1)}$ and $\mathbf{Q}^{(2)}$ were actually coincident in the image, and that the underlying surface points were also coincident and hence at the same depth $\lambda^{(1)} = \lambda^{(2)}$. In practice, point pairs used as features will not coincide exactly and an error limit on curvature (or, more conveniently, its inverse) must be computed to allow for this. The limit can be used to provide an error interval for computed curvature. It is also useful as a threshold for the test (see below) for labelling extremal boundaries — that is, to tell how close to zero the inverse normal curvature $1/\kappa^t$ must lie to be considered to be on a surface marking or a crease edge rather than an extremal boundary.

The error in inverse normal curvature, R_{error}, consists of errors due to the difference in depths of the 2 features, $\Delta\lambda$, and due the finite separation in the image, $\boldsymbol{\delta}$. These effects can be easily computed from the difference of equation (7) for the 2 points. For nearby points that are on the same surface and for rotation about a fixation point the dominant error can be conveniently expressed as [7]:

$$|R_{error}| \approx 9\lambda|\boldsymbol{\delta}| + \lambda^3|\boldsymbol{\delta}|\frac{\dot{\boldsymbol{\Omega}}}{(\dot{\mathbf{v}}.\mathbf{n})^2} + \lambda|\Delta\lambda|\frac{\ddot{\mathbf{v}}.\mathbf{n}}{(\dot{\mathbf{v}}.\mathbf{n})^2} \qquad (14)$$

Measurements of curvature using parallax measurements will be robust if the separations between points on nearby contours satisfy

$$|\boldsymbol{\delta}| \ll \frac{1}{9\lambda\Delta\kappa^t} \qquad (15)$$

5 Implementation

We have implemented the theory of deformation of apparent contours using images taken from a camera mounted on a robot arm. Image contours are tracked using a variant of the well-known "snake" of Kass, Witkin and Terzopoulos [14]. The snake is a computational construct, a dynamic curve able to track moving, deforming image features. Since many snakes can be active at once, each tracking its feature contour as a background process, they constitute a versatile mechanism for direction and focus of attention, a second generation of Inoue's window system [13].

We have made a simple innovation that considerably enhances efficiency.

We use a B-spline in place of the original elastic rod simulation, greatly reducing the number of state variables required for a snake of a given length. Since accurate measurements are required to compute image accelerations, care has been taken over sub-pixel resolution. At earlier stages of tracking, when coarse blurring is used, the capture range of the snake is large but localisation is poor — the snake may lag considerably behind the contour. Once the snake has converged onto the contour, standard edge-detection techniques [6] are used to obtain accurate localisation. For relative measurements of curvature, as advocated for robustness, two snakes are used.

5.1 Tracking with the B-spline snake

The B-spline is a curve in the image plane

$$\mathbf{x}(s) = \sum_i f_i(s)\mathbf{q}_i \qquad (16)$$

where f_i are the spline basis functions with coefficients $\mathbf{q}_i$ - the vertices or control points of the curve's "characteristic polygon". The curves may be open or closed as required. The flexibility of the curve increases as more control points are added; each allows either one more inflection in curve or, when multiple knots are used, reduced continuity at one point. The vertices are positioned so that the curve locates the desired image contour. Away from equilibrium, the external force on a point $\mathbf{x}(s)$ is chosen to be

$$\mathbf{F}(s_j) = \nabla\,|\nabla G(\sigma) * I(\mathbf{x}(s))| \qquad (17)$$

so that, at equilibrium (when image forces vanish), $\mathbf{x}(s)$ stabilises close to a contour of high-contrast. There are no internal forces since the B-spline representation maintains smoothness via hard constraints implicit in the representation.

External forces are applied to the curve itself but for iterative adjustment of displacement it is necessary to compute the force transmitted to each control point. This is done via the principle of virtual work. At each iteration the motion of the control points, $\delta\mathbf{q}_i$ is given by [7]:

$$\delta\mathbf{q}_i = \alpha\sum_j f_i(s_j)\mathbf{F}(s_j) \qquad (18)$$

where α, the compliance constant, is chosen so that, in practice, the maximum movement at any iteration always lies within the radius of the pointspread function for the applied smoothing.

As the snake approaches the image contour it "locks on" and tracking is maintained provided the contour does not move too fast. In the present implementation a limited number of snakes were initialised by hand in the first frame near images contours of interest after which they track the image contour automatically (figure 3). We have recently produced a realtime system (15 frames per second) with computer control of camera motion in which snakes just "hang-around" in the image until they are swept by the motion of the camera over a feature for which they have affinity.

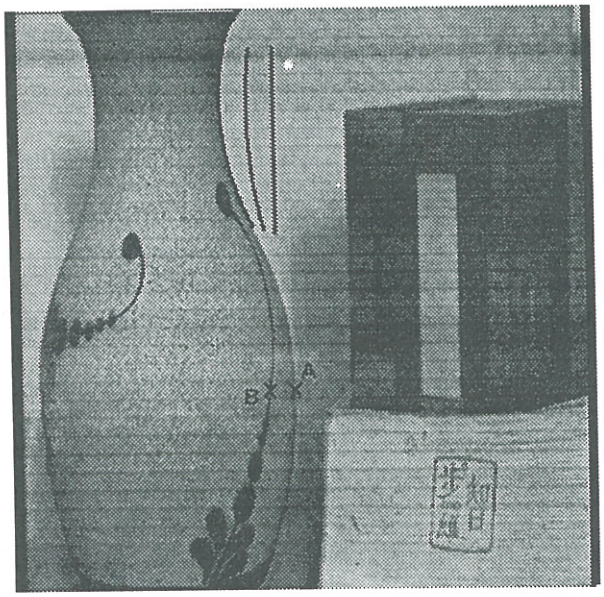

Figure 3. B-spline snake for image contour localisation. The snake "hangs around" until camera motion sweeps it into the vicinity of an image feature. External image forces guide it towards the image contour to which it then locks on to and tracks over the image sequence. The intitial, intermediate and final position are shown.

Figure 4. Estimating surface curvatures from 3 discrete views Points are selected on image contours in view 1, indicated by crosses A and B for points on an extremal boundary and surface marking respectively. For epipolar parameterisation *of the surface corresponding features lie on epipolar lines in subsequent views. Measurement of the 3 rays lying in an epipolar plane can be used to estimate the* osculating circle *in the epipolar plane*

5.2 Numerical estimation of depth and curvature

Computing depth and surface curvatures from equations (6) and (7) is a non-trivial problem since it involves the estimation of second order spatio-temporal derivatives. This is greatly simplified for simple motions – linear viewer motion perpendicular to the optical axis – when the trajectories of fixed features appear as straight lines in the spatio–temporal image [4]. First and second order temporal derivatives of the trajectory can then be used to determine depth and surface curvature respectively.

For arbitrary motions we have developed a simple numerical method for estimating depth and surface curvatures from a minimum of three discrete views, by determining the *osculating* circle in each epipolar plane [7]. The error and sensitivity analysis is greatly simplified with this formulation. Of course this introduces a tradeoff between truncation error and measurement error. We are no longer computing surface curvature at a point. However the computation is less sensitive to edge localisation. It also allows the use of longer "stereo baselines".

6 Experimental Results

6.1 Determining surface curvatures from 3 discrete views

Figure 4 shows 1 view from a sequence of a scene taken from a camera mounted on a moving robot-arm whose motion has been accurately calibrated from visual data [20]. The image contours are tracked automatically and the numerical method described in the [7] is used to estimate the radius of curvature of the *normal section*, R (where $\kappa^t = 1/R$), for a point on an extremal boundary of the vase, A . The method is repeated for a point which is not on an extremal boundary but is on a nearby surface marking, B. This is a degenerate case of the parameterisation. A surface marking can be considered as the limiting case of a point with infinite curvature and hence ideally will have zero "radius of curvature".

The radius of curvature at A was estimated as 42 $\pm$ 10mm. It was measured using calipers as 45 $\pm$ 2mm. For the marking, B, the radius of curvature was estimated as 3 $\pm$ 10mm. The estimated curvatures agree with the actual curvatures. However, the results are very sensitive to perturbations in the motion parameters. The estimate of curvature is affected by errors in image localisation, and uncertainties in ego-motion calibration in a non-linear way. The effect of small errors in the assumed egomotion can, however, be easily computed by first order perturbation analysis (figure 5a and b, curves I).

Differential measurements of curvature are in principle based on measuring the relative image motion of nearby points on different contours (equation (13)). In practice this is equivalent (equation(12)) to computing the difference of radii of curvature (inverse curvature) at the 2 points. The radius of curvature measured at a surface marking is determined by errors in measurement of motion. It can be used as a reference point to subtract the global additive errors due to imprecise motion when estimating the curvature at the point on the extremal boundary. Figures 5 a and b (curves II) show that the sensitivity of the *differential curvature* to error in position and rotation computed between points A and B (2 nearby points at similar depths) is reduced by an order of magnitude. This is a striking decrease in sensitivity even though the features do not coincide exactly as the theory required.

6.2 Discriminating between fixed surface features and extremal boundaries

The magnitude of $R = 1/\kappa^t$ can be used to determine whether a point on an image contour lies on an apparent contour or on the projection of a fixed surface feature such as a crease, shadow or surface marking.

With noisy image measurements or poorly calibrated motion we must test by error analysis the hypothesis that R is not equal to zero for an extremal boundary. The effects of small errors in image measurements, and ego-motion are easily computed and can be conveniently represented by the *covariance* of the estimated curvature.

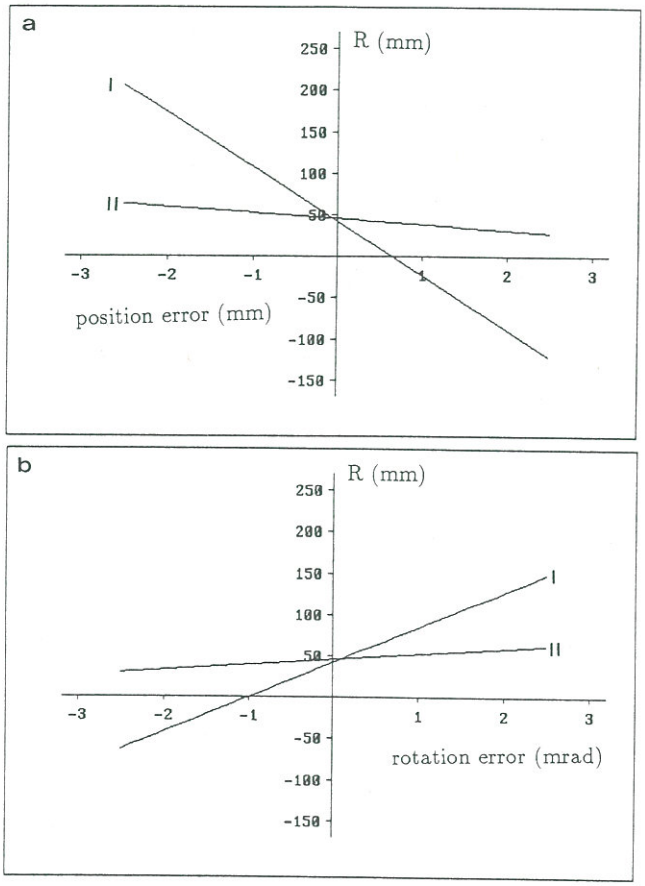

Figure 5 . Sensitivity of curvature estimated from absolute measurements and parallax to errors in motion.
The radius of curvature ($R = 1/\kappa^t$) for a point on the extremal boundary (A) is plotted as a function of errors in the camera position (a) and orientation (b). Curvature estimation is highly sensitive to errors in egomotion. Curve I shows that a perturbation of 1mm in position (in a translation of 100mm) produces an error of 155% in the estimated radius of curvature. A perturbation of 1mrad in rotation about an axis defined by the epipolar plane (in a total rotation of 200mrad) produces an error of 100%. However, if parallax based measurements are used the estimation of curvature is much more robust to errors in egomotion. Curve II shows the difference in radii of curvature between a point on the extremal boundary (A) and the nearby surface marking (B) plotted against error in the position (a) and orientation (b). The sensitivity is reduced by an order of magnitude, to 19% per mm error and 12% per mrad error respectively.

Using absolute measurements, however, the discrimination between fixed and extremal features is limited by the uncertainties in robot motion. For the image sequence of figure 4 it is only possible to discriminate between fixed features and points on extremal boundaries with inverse curvatures greater than 15mm.

By using relative measurements the discrimination is greatly improved and is limited by the finite separation between the points as predicted by (14). For the example of figure 4 this limit is approximately 3mm.

Suppose no known surface feature has been identified in advance. Can the robust relative measurements be made to bootstrap themselves without an independent surface reference? It is possible by relative (two-point) curvature measurements obtained for a small set of nearby points to determine pairs which are fixed features. Once a fixed feature is detected it can act as stable reference for estimating the curvature at extremal boundaries.

In detecting an apparent contour we have also determined on which side is the surface and so can compute the sign of Gaussian curvature from the curvature of the image contour. Figure 6 shows a selected number of contours which have been automatically tracked and are correctly labelled.

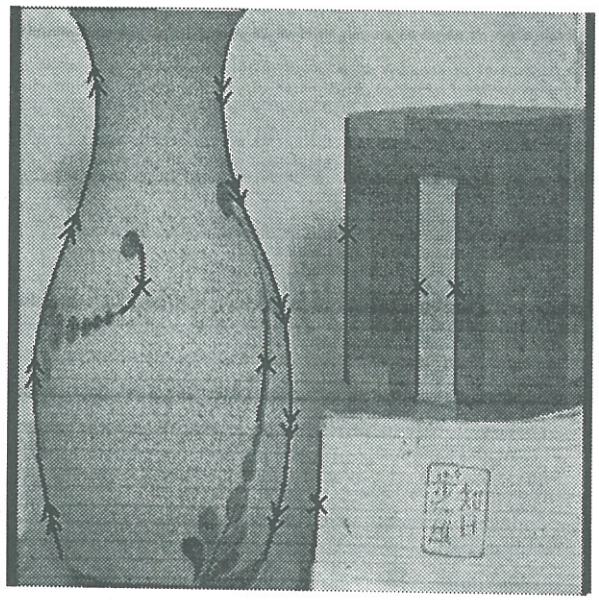

Figure 6. Detecting and labelling extremal boundaries
The magnitude of the radius of curvature ($1/\kappa^t$, computed from 3 views) can be used to classify image curves as either the projection of extremal boundaries or fixed features (surface markings, occluding edges or orientation discontinuities). The sign of κ^t determines on which side of the image contour lies the surface. NOTE: a $\times$ label indicates a fixed feature. A $\twoheadrightarrow$ label indicates an apparent contour. The surface lies to the right as one moves in the direction of the twin arrows. The sign of Gaussian curvature can then be inferred diectly from the sign of the curvature of the apparent contour.

6.3 Reconstruction of surfaces

In the vicinity of the extremal boundary we can recover the 2 families of parametric curves. These constitute a *conjugate* grid of surface curves: s-parameter curves (the extremal boundaries from the different viewpoints) and t-parameter curves (the intersection of a pencil of epipolar planes defined by the first 2 viewpoints and the surface). The recovered strip of surface is shown in figure 7.

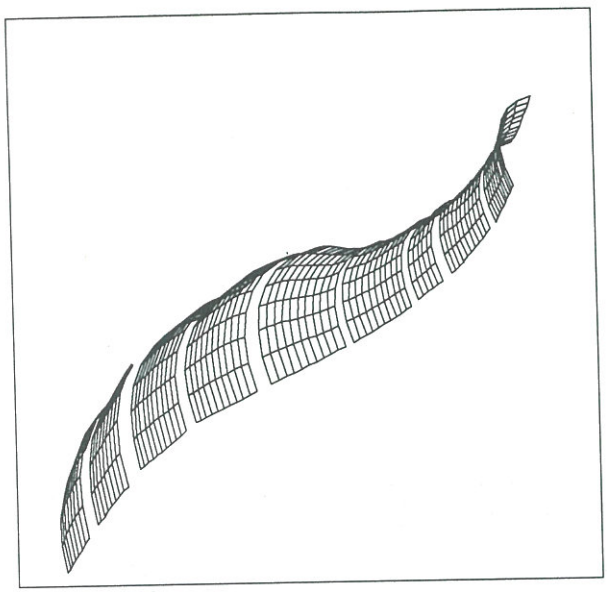

Figure 7. Reconstructed surface in the vicinity of the extremal boundary (A) of the vase shown in figure 4, shown here from a new viewpoint

7 Conclusion

We have presented a computational theory of 3D shape recovery from the deformation of the apparent contour. We have successfully demonstrated initial implementations of this theory in discriminating between the projection of rigid and extremal curves and recovering surface strips in the vicinity of the extremal boundaries. We have found that the apparent contour is a robust indicator of solid shape.

We have also shown that in practice given just one surface reference point, discrimination is easier, and highly robust relative curvature measurements can be made at points on apparent contours. We believe this represents a significant step in the development of practical techniques for robust, qualitative 3D vision.

We are currently working on the realtime implementation of these algorithms for use in tasks involving the active exploration of the 3D geometry of visible surfaces.

References

[1] H.G. Barrow and J.M. Tenenbaum. Interpreting line drawings as three-dimensional surfaces. *Artificial Intelligence*, 17:75–116, 1981.

[2] A. Blake and G. Brelstaff. Geometry from specularities. In *Proc. 2nd Int. Conf. on Computer Vision*, pages 394–403, 1988.

[3] A. Blake and R. Cipolla. In *Proc. 1st European Conf. on Computer Vision*, 465–474, Springer-Verlag, 1990.

[4] R.C. Bolles, H.H. Baker, and D.H. Marimont. Epipolar-plane image analysis: an approach to determining structure. *International Journal of Computer Vision*, vol.1:7–55, 1987.

[5] M. Brady, J. Ponce, A. Yuille, and H Asada. Describing surfaces. *Computer Graphics Image Processing*, 32:1–28, 1985.

[6] J.F. Canny. A computational approach to edge detection. *IEEE PAMI*, 8, 6, 679-698, 1986.

[7] R. Cipolla and A. Blake. The dynamic analysis of apparent contours. Technical Report OUEL 1828/89, University of Oxford, April 1990.

[8] M.P. DoCarmo. *Differential Geometry of Curves and Surfaces.* Prentice-Hall, 1976.

[9] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. In *Proc. 1st Int. Conf. on Computer Vision*, pages 136–144, London, 1987.

[10] C.G. Harris. Determination of ego - motion from matched points. In *Proc. 3rd Alvey Vision Conference*, pages 189–192, 1987.

[11] H. von. Helmholtz. *Treatise on Physiological Optics.* Dover (New York), 1925.

[12] B.K.P. Horn. *Robot Vision.* MIT Press, 1987.

[13] H. Inoue and H. Mizoguchi. A flexible multi window vision system for robots. In *Proc. 2nd Int. Symp. on Robotics Research*, pages 95–102, 1985.

[14] M. Kass, A. Witkin, and D. Terzopoulos. Snakes:active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, pages 259–268, 1987.

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

[16] H.C. Longuet-Higgins and K. Pradzny. The interpretation of a moving retinal image. *Proc.R.Soc.Lond.*, B208:385–397, 1980.

[17] D. Marr. Analysis of occluding contour. *Proc. Royal Society, London*, 197:441–475, 1977.

[18] S.J. Maybank. The angular velocity associated with the optical flow field arising from motion through a rigid environment. *Proc. Royal Society, London*, A401:317–326, 1985.

[19] J.H. Rieger and D.L. Lawton. Processing differential image motion. *J. Optical Soc. of America*, A2(2):354–359, 1985.

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

[21] D. Weinshall. Qualitative depth from stereo, with applications. *Computer Vision, Graphics and Image Processing*, 49:222–241, 1990.

Acknowledgments

The authors acknowledge the generous support of the IBM UK Scientific Centre and the Lady Wolfson Fellowship, Oxford. This work was partially funded by Esprit BRA 3274 (FIRST) and the SERC. We are grateful of discussion with Professor Mike Brady, Dr Andrew Zisserman and Professor Christopher Longuet-Higgins.