HOT CURVES FOR MODELLING AND RECOGNITION OF SMOOTH CURVED 3D OBJECTS *

Tanuja Joshi and Jean Ponce
Beckman Institute
University of Illinois, Urbana, IL 61801

B. Vijayakumar and David J. Kriegman
Center for Systems Science
Yale University, New Haven, CT 06520

Abstract: *We represent arbitrary smooth curved 3D shapes by a discrete set of HOT curves where a surface admits High Order Tangents. These curves determine the structure of the image contours and its catastrophic changes, and there is a natural correspondence between some of them and monocular contour features such as inflections and bitangents. We present a method for automatically constructing the HOT curves from continuous sequences of video images and describe an approach to object recognition using viewpoint-dependent monocular image features as indices into a database of models and as a basis for pose estimation. We have implemented both the methods and present results obtained from real images.*

1 Introduction

While implemented recognition systems based on parametric shape representations such as algebraic surfaces or superquadrics have demonstrated their usefulness, the ultimate utility of a representation is limited by its scope. This suggests looking for a more general representation capable of handling *arbitrary* curved objects. Accordingly we describe smooth curved surfaces by *HOT* curves, a discrete, non-parametric representation anchored in differential geometry. These curves determine the structure of image contours and its catastrophic changes, and there is a natural correspondence between some of them and contour features such as inflections and bitangents. We present in Sect. 3 a technique for automatically constructing the proposed representation from sequences of video images. We propose in Sect. 4 a new method to object recognition (pose computation and indexing) from a single video image.

We have implemented both methods, and present results obtained on real images in Sects. 3-4. We briefly sketch future research directions in Sect. 5.

*This work was supported in part by the National Science Foundation under Grant IRI-9224815. T. Joshi was supported in part by the Defense Advanced Research Projects Agency and the National Science Foundation under Grant IRI-8902728. D. Kriegman was supported in part by a National Science Foundation NYI Grant IRI-9257990.

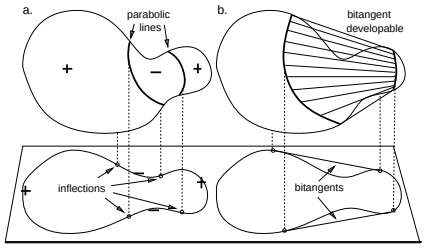


Figure 1: Parabolic curves, limiting bitangent developables, and their projections.

2 HOT Curves

We say that a tangent vector at some point *has contact of order n with the surface* (or more concisely *is of order* n) when the derivative of order i of the surface equation in the direction of the tangent is zero for all $i < n$, and non-zero for $i = n$. While all surface points have an infinity of tangents of order two (or greater) in their tangent plane, only hyperbolic points may have third order tangents. Contact of order four or higher only occurs along certain surface curves (i.e. parabolic and flecnodal curves), and fifth order contact only occurs at isolated points along these curves [9].

Additionally, there are other surface curves where a line grazes the surface in multiple discrete points with at least second order contact in some exceptional manner: the *limiting bitangents*, the *asymptotic bitangents*, and the *tritangents*. A limiting bitangent touches the surface at two points sharing a common (bi)tangent plane; an asymptotic bitangent is an asymptotic direction at one of the two contact points. Each of these curves has a corresponding ruled surface which grazes the original surface along two curves. Finally, a tritangent grazes the surface in three distinct points and has a corresponding ruled surface.

The parabolic, flecnodal, limiting and asymptotic bitangent, and tritangent curves form the basis for a shape representation aimed at automatic model construction and object recognition. We propose to maintain an explicit discrete representation of these five *HOT curves* (where the surface admits *High Order Tangents*), recording the position of each curve point on the surface, the direction of its surface normal, and the direction of its "special" (bi)tangent.

There is a close relationship between the three-dimensional HOT curves and the two-dimensional contour inflections and bitangents (Fig. 1). Koenderink [4] characterized the relationship between the curvature of an image contour, the two principal curvatures, and the viewing direction under orthographic projection (See [12] for an extension to the perspective case). A consequence of this relationship is that the image of a parabolic point is generally an inflection. This defines a natural correspondence between observed inflections and parabolic points (Fig. 1.a). Similarly, it is easy to see that a contour bitangent is the projection of a limiting bitangent point (Fig. 1.b). Contour bitangents have been used in invariant-based recognition of 2D objects [7, 10] and solids of revolution [13]. Interestingly, the contour structure of a generic smooth surface changes when the viewpoint coincides with one of the higher-order tangents or bitangents along the parabolic, flecnodal, limiting bitangent, asymptotic bitangent, and tritangent curves [3]. These *accidental* viewpoints are called *visual events*. This is the basis for the *aspect graph* representation [5].

3 Automatic Construction of HOT Curve Models

In [6], we presented an implemented method for reconstructing the limiting bitangent and parabolic curves from a sequence of images: relying on the direct correspondence between image features and these surface curves, we reformulated the methods proposed by Giblin and Weiss for curve reconstruction rather than surface reconstruction [1]. Here, we extend our previous results in two ways.

First, reconstructing a parabolic point alone is insufficient for determining the corresponding visual event (beak or lip) which only occurs when the viewing direction is an asymptotic direction at that point. Thus, we present a method for computing the asymptotic direction from image measurements.

Second, we have observed that the reconstructed parabolic points are noisier than the reconstructed points on the bitangent curve. The inflection points are relatively easy to detect as zero-crossings of the curvature, but because a curve is locally flat near an inflection, they are very difficult to localize accurately. For the same reason the tangent line at an inflection can be measured very accurately. Using a temporal sequence of image tangent lines as input, we present a method for reconstructing the asymptotic line (a line through the surface point in the asymptotic direction) at each point on a parabolic curve. Essentially the same idea can be applied to computing the limiting bitangent lines from the measured bitangents in an image sequence.

We assume a pinhole projection model and a moving calibrated camera. See Fig. 2. Thus, given a point on the image contour, the viewing direction $\mathbf{v}$ can be determined. We consider the image contour to be parameterized by arc length s. Since the camera is moving, the image contour is a function of time t.

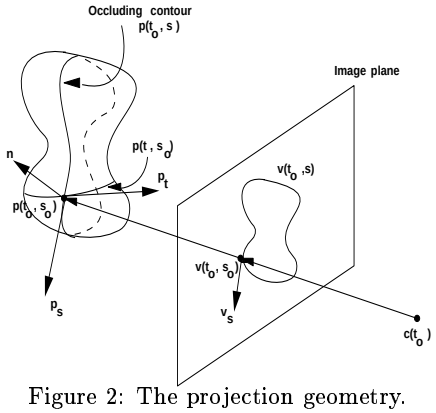


Figure 2: The projection geometry.

In the following, we consider a spatio-temporal parameterization of a 3D surface $\mathbf{p}(s,t)$ in terms of the spatial parameter s and the time t:

$$\mathbf{p}(t,s) = \mathbf{c}(t) + \lambda(t,s)\mathbf{v}(t,s). \tag{1}$$

We know the camera motion ($\mathbf{c}(t)$), and we can determine $\mathbf{v}(t,s)$ from image measurements and the known camera motion. $\lambda(t,s)$ is the unknown distance from $\mathbf{c}$ to $\mathbf{p}$. We only require that $\mathbf{v}(t,s)$ be continuous in the neighborhood of interest and that for a fixed $t = t_0$, $\mathbf{v}(t_0,s)$ parameterize a single image contour. Note that $\mathbf{p}(t_0,s)$ is a parameterization of the 3D occluding contour. As shown in [6], the depth parameter λ is given by

$$\lambda = -(\mathbf{n}\cdot\mathbf{c}_t)/(\mathbf{n}\cdot\mathbf{v}_t) \tag{2}$$

where the subscripts denote the partial derivative of a vector with respect to some variable. Thus, using (1) and (2), we can reconstruct the curve.

We now address the problem of reconstructing the asymptotic tangents from a sequence of image tangents at a parabolic point. We will first show how to reconstruct a particular tangent line (described by an inflection and the tangent vector to the image contour). We then show that this is indeed the asymptotic line.

The tangent plane for every point on the occluding contour from a single image is given by:

$$\mathbf{n}\cdot(\mathbf{p}-\mathbf{c}) = 0. \tag{3}$$

where $\mathbf{n}$ is the surface normal at a point $\mathbf{p}(t,s)$ which can be determined by $\mathbf{n}(t,s) = \mathbf{v}(t,s)\times\mathbf{v}_s(t,s)$. Here $\mathbf{v}_s(t,s)$ is the tangent to the image contour. Differentiating (3) with respect to t and using $\mathbf{n}\cdot\mathbf{p}_t = 0$ yields

$$(\mathbf{n}_t\cdot\mathbf{p}) - (\mathbf{n}_t\cdot\mathbf{c}) - (\mathbf{n}\cdot\mathbf{c}_t) = 0. \tag{4}$$

Equations (3) and (4) define two planes containing $\mathbf{p}$, whose intersection is a line which we call the *tangent line*. Its direction is along $\mathbf{T} = \mathbf{n}\times\mathbf{n}_t$. A point on the tangent

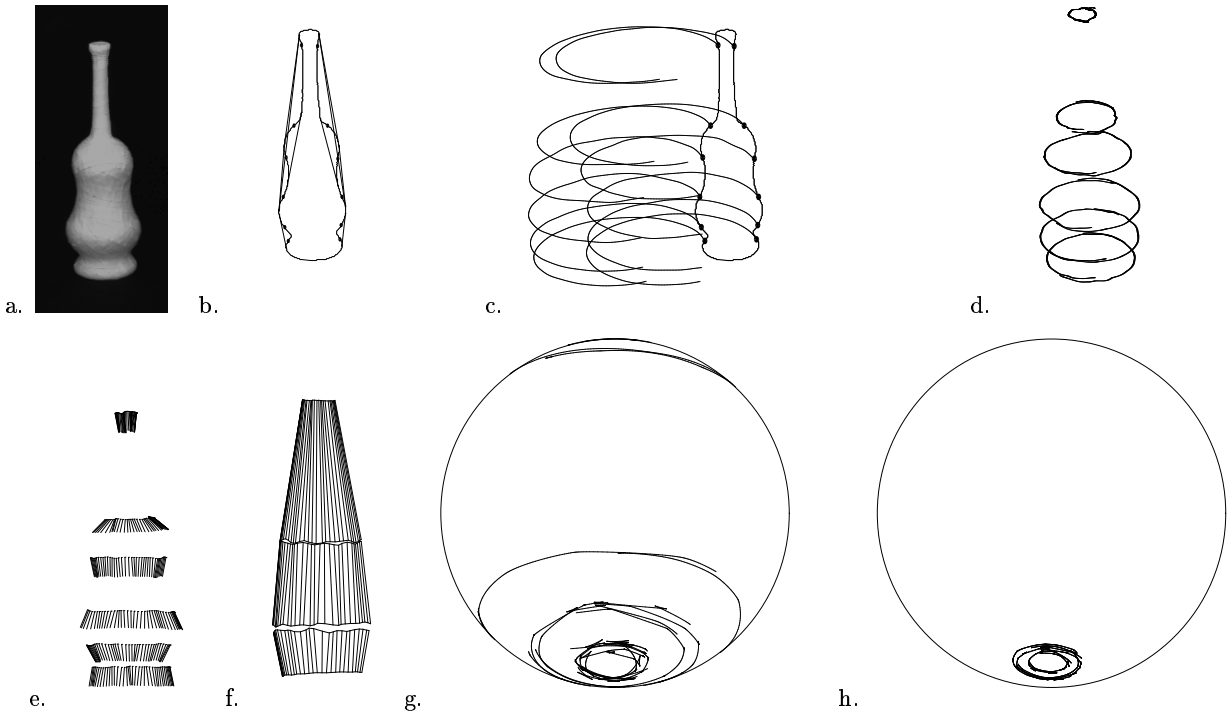


Figure 3: HOT curve reconstruction: a. An image. b. Extracted features. c. Tracked inflections. d. Reconstructed parabolic curves. e. Asymptotic directions along the parabolic curve (from a side view). f. Reconstructed limiting bitangent developable surface that lies on the convex hull (from a side view). g. Beak and Lip transitions on the view sphere. h. Tangent crossing events.

line can be found by intersecting it with a third independent plane such as

$$\mathbf{v}_s \cdot (\mathbf{p} - \mathbf{c}) = 0. \tag{5}$$

A point $\mathbf{p}$ that satisfies (3), (4) and (5) along with the line direction $\mathbf{T}$ fully defines the tangent line. We show in [2] that at a parabolic point, the asymptotic direction is tangent to the occluding contour, and the direction $\mathbf{T}$ is the asymptotic direction.

Thus, the tangent line and the asymptotic direction can be computed directly from the measured inflection tracks.

3.1 Implementation and Results

We have implemented the algorithm for reconstructing the asymptotic lines at parabolic points and the bitangent line of the limiting bitangent developable surface. We place the object to be modelled on a turntable; this scene is viewed from a nearby camera in general position (i.e. the image plane is not parallel to the axis of rotation). Although the equations in Sect. 3 were derived for the case of moving camera, they can be rederived for the case of moving object without difficulty.

In the example shown, we have chosen to use an Israeli glass bottle (which is approximately a solid of revolution), since the parabolic and bitangent developable lines are readily interpreted. In particular, these curves should be circles centered on the object's axis while the corresponding visual event curves should sweep lines of latitude on the view sphere. Our methods do not make use of the fact that this object is a solid of revolution. 280 images were taken at one degree increments. As described in [6], the inflection points and bitangents are found using scale space methods, and feature points are tracked through the sequence using first order prediction.

Figure 3.a–c shows a cropped image of the bottle, the detected contours, inflections and bitangent lines, and the temporal feature tracks. Figure 3.d shows the recovered parabolic curves. In Fig. 3.e the asymptotic directions are drawn along the parabolic lines. Fig. 3.f shows the recovered limiting bitangent developable surface that lies on the convex hull. The recovered asymptotic directions, which correspond to lip and beak events in an aspect graph, are drawn on the view sphere in Fig. 3.g. The tangent crossing events (the direction of limiting bitangent lines) are shown in Fig. 3.h.

4 Object Recognition

In the polyhedral domain there is a pointwise correspondence between viewpoint-independent object features such as vertices or edges and image features such as line segments and corners. The situation is more complicated for curved objects since most image features (e.g., contour

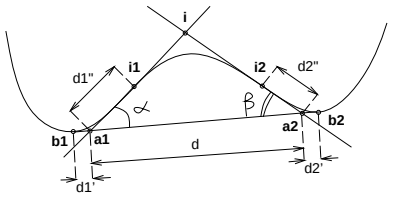


Figure 4: A bitangent and the two inflections in the corresponding pocket.

segments or their inflections) depend on viewpoint. In this section, we use the natural correspondence between the parabolic and limiting bitangent curves of a surface and the inflections and bitangents of its contour for pose computation and indexing into a database of models.

We assume scaled orthographic projection. The key observation is that two surface points with different surface normals completely determine the viewing direction from which they are simultaneously observable. This viewing direction is given as the intersection of their tangent planes. Thus given a pair of points on the parabolic curves (in $\mathbb{R}^3$) or on the limiting bitangent curves (in $\mathbb{R}^6$) we can compute the viewing direction from which they are simultaneously observable. We can also predict the position of the contour features as well as the contour tangent at those features. Using these, we can predict a number of image observables that are independent of the other viewing parameters (rotation, translation, and scaling in the image plane). These scale-independent image observables are the angles between the tangent lines and the ratios of distances between the features. We then use these observables as table indices in a table whose entries record the corresponding viewing directions and scale factors. This table can then be used for pose estimation. Note the interesting fact that by simply adding the identity of the model to each entry in the table, it can also be used to retrieve candidate matches from a database of object models.

A pair of parabolic points yields two such scale-independent parameters, implying that the indexed table will be two-dimensional. Since the entries in this two-dimensional table form a two-dimensional surface parameterized by the position of two points along parabolic curves, this table is likely to be densely populated. A pair of limiting bitangents yields four scale-independent parameters. This yields a four-dimensional table, but the entries corresponding to a given model still form a two-dimensional surface in this four-dimensional space. This makes the table sparse, reducing the likelihood of getting a large number of candidates for a single pair of features.

Note that we can further consider a piece of contour lying between two bitangent points which forms a pocket with at least two inflections if there are no intervening occlusions (Fig. 4). This case yields six scale-independent parameters as described next.

As shown in Fig. 4 the three tangent lines form a triangle, which can be represented by two angles α, β, and the length d of the edge between them. We can measure four more quantities from the image: the distances d'_1, d'_2 between the two contacts $\mathbf{b}_1, \mathbf{b}_2$ of the bitangent with the contour and the vertices $\mathbf{a}_1, \mathbf{a}_2$ of the triangle, and the distances d''_1, d''_2 between $\mathbf{a}_1, \mathbf{a}_2$ and the inflections $\mathbf{i}_1, \mathbf{i}_2$ (Fig. 4). We obtain six scale-independent parameters: $\alpha, \beta, d'_1/d, d'_2/d, d''_1/dand d''_2/d$. Note that although three curves are involved in the construction of a pocket, the viewing direction corresponding to a particular pocket is completely determined by two surface points only. In other words, the entries corresponding to a given object model still form a two-dimensional surface, but this time it is embedded in a six-dimensional space. This is why we have chosen to use pockets formed by a pair of inflections and a bitangent in our implementation.

4.1 Implementation and Results

Two of the observables associated with a pocket (namely d''_1/d and d''_2/d) depend on the position of the inflections along the contour. To avoid mistakes due to poor localization, we use the remaining four parameters for indexing. The two parameters associated with the position of the inflections are used for verification only.

For each model in the database, we consider all pairs of discrete parabolic curves, and all pairs of points on them. For each pair of parabolic points, the viewing direction from which both the points will appear on the contour is computed as the cross-product of the surface normals. All the limiting bitangents that are seen from this direction are found by searching the corresponding curves for a point whose surface normal is coplanar with the normals at the two parabolic points.

For each triple formed by these parabolic points and a bitangent, the six scale-independent parameters are computed. Of these, the four more reliable ones $(\alpha, \beta, d'_1/d, d'_2/d)$ are used to index the table. We store in the corresponding table entry the viewing direction $\mathbf{v}$, the scale d, the two ratios $d''_1/d, d''_2/d$, and the identity of the model.

We have a discrete representation of the HOT curves as well as of the parameter space. To avoid missing any possible viewpoint, we use interpolation whenever neighbouring parabolic point pairs do not map onto neighbouring cells of the discretized parameter space.

At the time of recognition, the contour inflections and bitangents are found in the image. For each triple of the features, the six scale-independent parameters are computed. Four of these are used to index the four-dimensional table, and entries from the indexed cell are retrieved. The remaining two parameters are used to find the closest entry among the retrieved ones. For complex objects, several triples of features are observed. In this case, the table is independently indexed for each triple. The candidates corresponding to all the triples are sorted using the difference between scale-independent parameters of the candidate pocket and those of the detected pocket in the image. The sorted candidates are considered in turn. Using the

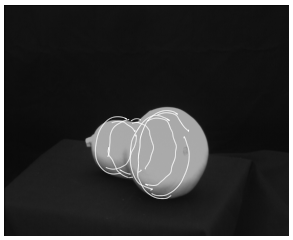
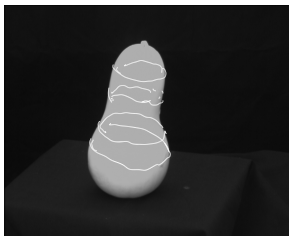
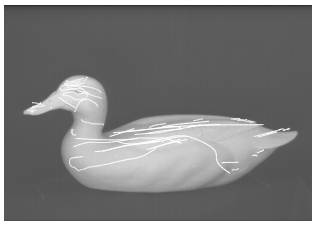
a.  b.  c. 

Figure 5: Result: recognized object models overlaid onto the original image using the estimated pose.

viewpoint and the object model associated with the current candidate, we predict the image position of the rest of the features associated with that model. We use the average distance from predicted features to detected image features to prune false matches. The candidate which gives the shortest average distance is declared to be the final recognized object.

We have implemented the recognition algorithm on a Sun Sparc 10. Depending on the complexity (number of parabolic curves) of the object, it takes from 10 to 25 minutes to add an object to the database. The recognition takes about 30 seconds. Out of these 30 seconds, the crucial step of hypothesizing the candidate models and viewpoints takes only about 5 seconds. The rest of the time is spent in the verification step.

At the time of recognition we always used the same database consisting of the models of four objects: a squash, a pear, a banana and a duck decoy. The HOT curves for the squash and duck model were constructed using the technique given in Sect. 3. The pear's and the banana's HOT curves were computed analytically from quartic surface models constructed using the methods of [11]. Fig. 5 shows the result of recognition on a variety of input images of the squash and the duck. In all cases, the majority of the retrieved entries corresponded to the correct object model. In all cases, the verification step pruned out any false candidates.

5 Discussion

The results shown in Sects. 3-4 offer a preliminary demonstration of the power of the HOT curve representation. We must conduct more experiments with more complicated objects, but we believe that it would have been quite difficult to model and recognize our duck decoy using a parametric shape representation such as superquadrics or algebraic surfaces.

Since the localization of the inflections is not very accurate, it might be better to use only the tangent lines instead of the point positions for recognition. In this case we would only need to store the tangent plane information along the HOT curves of each model, which would simplify the modelling process.

We are exploring the automatic construction of the aspect graph of an object [5] from a sequence of images. We already have two of the ingredients necessary for this construction: the asymptotic tangents at parabolic points and the limiting bitangents. We are investigating the computation of the asymptotic tangents at flecnodal points and the asymptotic bitangents and tritangents. Once those have been computed, a simple algorithm based on plane-sweep techniques (such as the one described in [8]) should be sufficient for computing the partition of the view sphere into an aspect graph.

References

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

[2] T. Joshi, J. Ponce, B. Vijayakumar, and D.J. Kriegman. Hot curves for modelling and recognition of smooth curved 3d shapes. Technical Report UIUC-BI-AI-RCV-93-08, Beckman Institute, University of Illinois, 1993.

[3] Y.L. Kergosien. La famille des projections orthogonales d'une surface et ses singularités. *C.R. Acad. Sc. Paris*, 292:929–932, 1981.

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

[5] J.J. Koenderink and A.J. Van Doorn. The singularities of the visual mapping. *Biological Cybernetics*, 24:51–59, 1976.

[6] D.J. Kriegman, B. Vijayakumar, and J. Ponce. Reconstruction of HOT curves from image sequences. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 20–26, 1993.

[7] Y. Lambdan and H. Wolfson. Geometric hashing: A general and efficient model-based recognition scheme. In *Proc. Int. Conf. Comp. Vision*, pages 238–249, 1988.

[8] S. Petitjean, J. Ponce, and D.J. Kriegman. Computing exact aspect graphs of curved objects: Algebraic surfaces. *Int. J. of Comp. Vision*, 9(3), 1992.

[9] O.A. Platonova. Singularities of the mutual disposition of a surface and a line. *Russian Mathematical Surveys*, 36(1):248–249, 1981.

[10] C.A. Rothwell, A. Zisserman, D.A. Forsyth, and J.L. Mundy. Canonical frames for planar object recognition. In *Proc. European Conf. Comp. Vision*, pages 757–772, 1992.

[11] S. Sullivan, L. Sandford, and J. Ponce. On using geometric distance fits to estimate 3D object shape, pose, and deformation from range, CT, and video images. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 110–116, New York City, NY, June 1993.

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

[13] A. Zisserman, D.A. Forsyth, J.L. Mundy, and C.A. Rothwell. Recognizing general curved objects efficiently. In J. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pages 228–251. MIT Press, Cambridge, Mass., 1992.