In Proceedings of ICCV '95

Invariant-Based Recognition of Complex Curved 3D Objects from Image Contours

B. Vijayakumar D. J. Kriegman
Center for Systems Science, Yale University
New Haven, CT 06520-8267

J. Ponce
Beckman Institute, University of Illinois
Urbana, IL 61801

Abstract

To recognize three-dimensional objects bounded by smooth curved surfaces from monocular image contours, viewpoint-dependent image features must be related to object geometry. Contour bitangents and inflections along with associated parallel tangents points are the projection of surface points that lie on the occluding contour for a five-parameter family of scaled orthographic projection viewpoints. An invariant representation can be computed from these image features and used for modeling and recognizing objects. Modeling is achieved by moving an object in front of a camera to obtain a curve of possible invariants. The relative camera-object motion is not required, and 3D models are not utilized. At recognition time, invariants computed from a single image are used to index the model database. Using the matched features, independent qualitative and quantitative verification procedures eliminate potential false matches. Examples from an implementation are presented.

1 Introduction

One of the primary goals of computer vision is object recognition; that is, given image data as input, determine the identity of objects in a scene. In its purest form, 3D objects are observed from a single arbitrary viewpoint; the recognition system is given a database of object models and must determine which of these models is most compatible with some subset of the image. Ideally, object models are acquired directly from images rather than being hand coded.

Many approaches to recognition have been proposed. Broadly speaking, there are three classes of geometric representations: 3D descriptions such as structural descriptions of volumetric primitives, polyhedra, collections of 3D points, and algebraic surfaces explicitly encode the object's geometry. Alternatively, objects are represented by their appearance in the form of a multiple-view representation such as an aspect graph. Finally, it may be represented by a collection of geometric invariants computed from image features. Depending on the representation, only certain image features (vertices, straight lines, the silhouette, etc.) can be used for recognition. Specific algorithms are a consequence of these choices and may tradeoff computational time, memory, and robustness.

In this paper, we present a new invariant-based representation that can be employed to recognize complex curved 3D objects in a single image. Recently, invariant-based recognition has received a great deal of attention [12]. Most of this work deals with 2D (laminar) objects since the perspective projection image is a projective transformation of the original shape. More recently, geometric constructions have been used to arrive at a projective transformation relating object to image features for certain restricted classes of 3D objects (e.g. surfaces of revolution, generalized cylinders, etc.) [15]. One of the advantages of geometric invariants is that they can be computed without using any specific object information; hence, they can index directly into a database of models. Using hash tables or trees, indexing can occur in time that is sublinear with respect to database size [10]. However, some of the enthusiasm for invariant-based recognition was dampened by the observation that there are no nontrivial invariants for the image of an arbitrary 3D point set [1, 5, 11]. Consequently, a 3D point set cannot be represented by a single vector which is invariant from all viewpoints.

For smooth curved 3D objects, the line drawing is the projection of visible points on the *occluding contour* (surface points where the line of sight lies in the tangent plane). Because the features themselves depend on viewpoint, it is not surprising that a curved object cannot be represented by a single vector of invariants. In recent work on "HOT curves" [6, 8], it was shown how a particular collection of 3D surface curves could be related to image features; taken over all viewpoints, the coordinates of the corresponding image features define a surface in some higher dimensional space. Unfortunately, these surfaces cannot be directly computed from images measurements – a 3D description of the surface curves is required, and so a method for reconstructing these curves from an image sequence with known camera-object motion was developed [8] based on the work of [3, 4].

In this paper, we extend the HOT curve approach by exploiting some of the same image features (bitangents and inflections) and surface features (bitangent developable and parabolic curves). For each of these features, we can define another set of features called *parallel tangent* points. Up to occlusion, these features are mutually visible over a five-parameter family of scaled orthographic projection viewpoints. From these features, a function can be evaluated which is invariant over this entire five-parameter family. An object can be modeled by moving it in front of a camera to reveal new feature points and obtain a curve of invariants. To completely model an object using these features, the object only has to be observed from a trajectory (curve) of viewpoints rather than from all viewpoints. Objects are then represented as collections of invariant curves. To recognize an object in an image, an invariant is computed and used as an index into a database of curves. Using the same features, independent qualitative and quantitative verification procedures eliminate potential false matches. Note that 3D object models are never used.

In [13], Murase and Nayar also use curves defined in high-dimensional spaces as a basis for object recognition. In their case, the curves are splines representing the possible appearances of an object as a function of viewpoint and illumination. They are embedded in the space defined by the principal components determined from a set of images of the object. Matching is achieved by finding the closest model curve to a new point computed by projecting the image into this space. This approach has only been implemented for a single degree of pose freedom.

In the next section, we introduce the set of image features that are used to construct these invariant curves and show their relationship to the corresponding 3D surface curves. In Section 2.2, we then present a simple function of the coordinates of these features whose result is invariant for all viewpoints where the features are visible. From a moving camera, a curve of invariants is obtained and used to model an object. From an image, a combination of indexing, constrained search and an independent verification procedure is used to recognize an object. The algorithm has been implemented and examples are presented. More details can be found in [14].

2 Object Representation

The approach for recognizing curved objects is based on using feature points of the image contour extracted from 2D images and computing measures that are invariant under viewpoint variations. The process is divided into two stages, *off-line* and *on-line*. During the off-line modeling process, a sequence of images of the object is taken either by moving an uncalibrated camera about the object or moving the object in front of a camera. The images are processed to extract features, invariants are computed, and retained in a model base. During the on-line process, similar features are extracted from just one image of the curved object, and the invariants are used to index model database. Returned matches are then verified using additional quantitative and qualitative constraints. Let us now consider the specific features. We assume scaled orthographic projection in the subsequent discussion.

2.1 Image Features

Establishing a correspondence between image and object features is a matter of combinatorics for polyhedra because the image features are viewpoint independent and correspond to object edges and vertices. For a smooth curved surface, the occluding contour is viewpoint-dependent, and so point-to-point correspondences cannot be established. However a point-to-curve correspondence can be determined for two

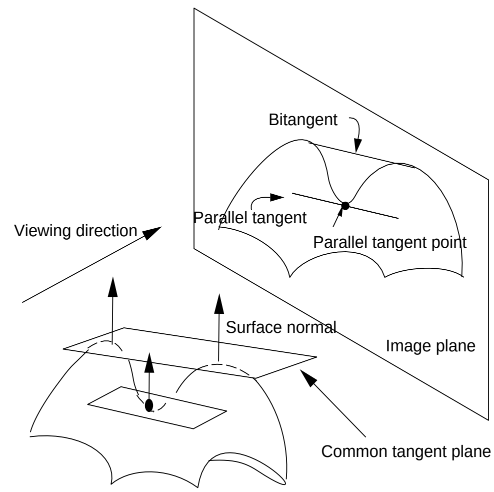

Figure 1. Surface points whose tangent planes are parallel to the tangent plane at a limiting bitangent project onto points with parallel image tangents.

sets of surface curves (parabolic curves and limiting bitangent developables) and their projections (inflection points and bitangents resp.) [6, 7, 8].

Now, consider parallel projection of a single limiting bitangent developable. This bitangent will fall on the occluding contour for any viewing direction that lies in the tangent plane, and the set of such viewing directions forms a great circle on the view sphere.[1] In addition, all surface points whose tangent planes are parallel to the tangent plane of the limiting bitangent will also be part of the occluding contour, up to occlusion. Since all of these tangent planes are parallel, the tangents to the image contour at the projection of these surface points will also be parallel. (See Figure 1). We call the image points *parallel tangent points* and will use them along with the corresponding image bitangent as features for recognition. These features are stable and easy to detect in images. From all viewpoints on the great circle, the set of tangent planes of the feature points project onto a set of parallel lines in the image; for different viewpoints on the circle, the set of lines differ merely by a similarity transformation when the camera is calibrated or an affine transformation otherwise. Such properties of parallel tangent features in a different context have been exploited by Kutulakos and Dyer [9]. Parallel tangent points can also be used as "interest points" in an affine-invariant based recognition scheme for 2D objects as proposed by Lambdan *et. al.* in [10].

When the entire limiting bitangent curve is considered, the set of parallel tangent points defines a family of surface curves. When modeling the object, any camera motion outside of the tangent plane will reveal a new pair of points on the limiting bitangent curve, and consequently the entire family of curves can be observed from a trajectory (a curve) of viewpoints. The same arguments apply to parallel tangents defined

[1] This is true up to occlusion by an opaque object. Note that when the limiting bitangent lies on the convex hull, it is visible over the entire great circle of viewpoints.

with respect to parabolic points; our subsequent discussion will concentrate on bitangents, since they can be measured very accurately in images and computed efficiently [15].

2.2 Invariants from Parallel Tangents

We now derive a simple affine invariant from a measured bitangent and corresponding parallel tangent points. Consider a bitangent and two parallel tangent points. These points and the common tangent direction define three parallel lines in the image which can be expressed as $\mathbf{n} \cdot \mathbf{x} + d_i = 0$, i=1,2,3, where $\mathbf{n}$ is the common unit normal direction, and d_i is the distance of line i from the origin. Define the distance vector to be $\mathbf{d} = (d_2 - d_1, d_3 - d_1)$. We now show that the projective coordinates of $\mathbf{d}$ are invariant under an affine transformation of the image plane.

Let an affine image transformation be given by: $\mathbf{x} = \mathbf{A}\mathbf{x}' + \mathbf{b}$ where $\mathbf{A}$ is a nonsingular 2x2 matrix and $\mathbf{b} \in \mathbb{R}^2$. After applying this transformation, the lines can be written as:

$$\frac{1}{|\mathbf{n}'|}\mathbf{n}' \cdot \mathbf{x} + \frac{\mathbf{n} \cdot \mathbf{b} + d_i}{|\mathbf{n}'|} = 0$$

where $\mathbf{n}' = \mathbf{A}\mathbf{n}$, and $d_i' = \frac{\mathbf{n} \cdot \mathbf{b} + d_i}{|\mathbf{n}'|}$ is the distance of the i-th transformed line from the origin. The transformed vector of distances between the lines is $\mathbf{d}' = \frac{1}{|\mathbf{n}'|}(d_2 - d_1, d_3 - d_1)$. Since $\mathbf{d}' = \lambda \mathbf{d}$ with λ being a non-zero scalar, the direction of the vectors $\mathbf{d}'$ and $\mathbf{d}$ are unaffected by an affine transformation. Equivalently, treating $\mathbf{d}$ and $\mathbf{d}'$ as homogeneous coordinates, both $\mathbf{d}$ and $\mathbf{d}'$ represent the same point in a one-dimensional projective space $\mathbb{P}^1$. Thus the projective coordinates of distances, $(d_2 - d_1, d_3 - d_1)$ are invariant up to an affine transformation of image coordinates. Another perspective on this invariant is that the ratio of the distances given by $\frac{(d_2 - d_1)}{(d_3 - d_1)}$ is also invariant under an affine transformation.

The implications of affine invariance follow. Since image plane rotation, scaling and translations are a subgroup of affine transformations, the distance vector $\mathbf{d}$ is also a similarity invariant. The invariance property holds even if the camera is uncalibrated (that is, when the aspect ratio of the pixels are unknown).

In general, a single image bitangent and the n corresponding parallel tangent points at distances d_i, $i = 1, \cdots, n$ from the developable define a point $(d_1, \cdots, d_n)$ in an $n-1$ dimensional projective space, $\mathbb{P}^{n-1}$. To visualize a point in $\mathbb{P}^{n-1}$, it can be seen as a point on an $n-1$ dimensional sphere embedded in an n dimensional Euclidean space given by $\frac{1}{|(d_1, \cdots, d_n)|}(d_1, \cdots, d_n)$. The invariance of this measurement with respect to 3D rotation of the object about an axis aligned with the surface normal is the basis for modeling 3D curved objects.

2.3 Representing Objects with a Curve of Invariants

In the previous section, we considered a measurement which is invariant over 3D rotations about the surface normal. We now show how a curve of invariants can be obtained from a sequence of images.

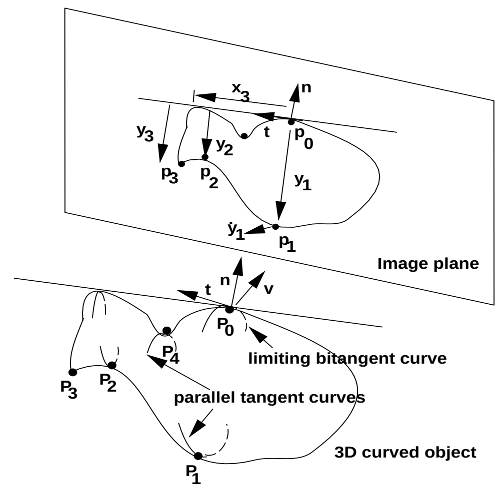

Figure 2. The geometry of parallel tangent curves.

A camera measures a feature point P_0 (bitangent endpoint or inflection) of the silhouette and n other silhouette points P_i, $(i = 1, \cdots, n)$ with parallel tangents. If $\mathbf{t}$ and $\mathbf{n}$ are the unit tangent and normal vectors to the image contour at P_0, $(\mathbf{t}, \mathbf{n})$ form a basis of the image plane. We can then construct a right-handed orthonormal coordinate frame $(\mathbf{t}, \mathbf{n}, \mathbf{v})$, where $\mathbf{v}$ is the viewing direction. Let $(\mathbf{v}, \mathbf{t})$ form a basis of the tangent plane $\mathbf{T_P}$ that contains the limiting bitangent. For $i = 1, \cdots, n$, we write

$$P_i - P_0 = x_i \mathbf{t} + y_i \mathbf{n} + z_i \mathbf{v} \qquad (1)$$

where $P_i - P_0$ denotes the vector joining P_i and P_0. x_i and y_i are the image coordinates of the i-th feature written in the $(\mathbf{t}, \mathbf{n})$ basis. In (1) the quantities x_i, y_i can be computed from image measurements. On the other hand z_i, and the coordinates of $\mathbf{t}, \mathbf{n}, \mathbf{v}$ in a world coordinate system are unknown.

So far, we have considered invariants extracted from a single image. For a camera or object moving along a trajectory parameterized by time t, the bitangent developable curve and family of surface points with parallel tangent planes will be observed. From the observed feature point $P_0(t)$ and parallel tangent points $P_i(t) (i = 1, \cdots, n)$, we define a curve $\Gamma^n(t)$ as the trace of $(x_1(t), y_1(t), \cdots, x_n(t), y_n(t)) \in \mathbb{R}^{2n}$ and the *invariant curve*, $\Psi^n(t)$ as the projection of $\Gamma^n(t)$ defined by $(y_1(t), \cdots, y_n(t)) \in \mathbb{P}^{n-1}$.

An object can now be simply modeled. It is moved in front of an uncalibrated camera or equivalently an uncalibrated camera around a stationary object. The bitangents and corresponding parallel tangents are detected in each image. To expose a new point on the limiting bitangent developable or parabolic curve, the camera is moved by a small amount in any direction except where the change in viewing direction lies in the common tangent plane of the features. In this manner, the limiting bitangent curves are fully revealed; the set of measured feature points are tracked through the sequence of images yielding the curve $\Gamma^n(t)$ where n is the number of parallel tangent points that are tracked through a temporal sequence. The projection of $\Gamma^n(t)$ then defines the invariant curve $\Psi^n(t) \in \mathbb{P}^{n-1}$, and a collection of such curves can be used to model an object. Note that this representation is built without knowing the actual camera motion or reconstructing the 3D surface.

Until now, we have largely neglected occlusion; we now show how this representation methodology can be extended to allow for both self-occlusion during the model building procedure as well as occlusion (or missing features) during object recognition.

To account for occlusion during modeling and on-line recognition, we represent the invariant curve models in multiple dimensions of the projective space. Consider for example, an invariant curve Ψ_1^4 constructed from four parallel tangent points. An image with three corresponding parallel tangents must be matched to invariant curves in $\mathbb{P}^2$ defined from three of the four parallel tangents tracks. There are $\binom{4}{3} = 4$ such combinations corresponding to projecting from four homogeneous coordinates to three coordinates. In general, from an invariant curve $\Psi_j^n(t) \in \mathbb{P}^{n-1}$, we can define n invariant curves, $\Psi_i^{n-1}(t)(i = 1, ..., n) \in \mathbb{P}^{n-2}$ which are the projections of $\Psi^n(t)$ along the n homogeneous coordinate axes of $\mathbb{P}^{n-1}$. This interdependence of invariant curves across the various dimensions of projective space can be exploited in implementing an efficient and fast indexing scheme during the recognition phase as will be explained in Section 3. The relationship of invariant curves across dimensions can be visualized as a tree structure with a node indicating a specific invariant curve (i.e., for a specific j) $\Psi_j^i(t) \in \mathbb{P}^{i-1}$ its parents representing curves in $\mathbb{P}^i$ whose projections onto $\mathbb{P}^{i-1}$ in one of the i directions is $\Psi_j^i(t)$ and its children being the $i-1$ curves in $\mathbb{P}^{i-2}$ which $\Psi_j^i(t)$ projects onto.

Note that for each bitangent, the corresponding parallel tangents can be uniquely ordered by increasing signed distance during both modeling and on-line recognition. Thus, there is no need to consider all permutations of the parallel tangents during recognition.

In summary, an object is modeled as a *collection* of invariant curves, $\Psi_j^i(t) \in \mathbb{P}^{i-1}(i = 3, \cdots, n)$ and $(j = 1, \cdots, \binom{n}{i})$ where n is the maximum number of parallel tangent features extracted in a single image.

3 Object Recognition

The computed invariant curves can be used for on-line recognition, and here we present a simple algorithm. Given an arbitrary image, some edge detection and linking method is used to compute a line drawing. From the image contours, first bitangents (or inflections) and then the corresponding parallel tangents are determined.

From a bitangent and its n corresponding parallel tangents extracted on-line, the recognition process begins by indexing into the database with all $\binom{n}{3}$ combinations of invariants in $\mathbb{P}^2$. The indexing procedure

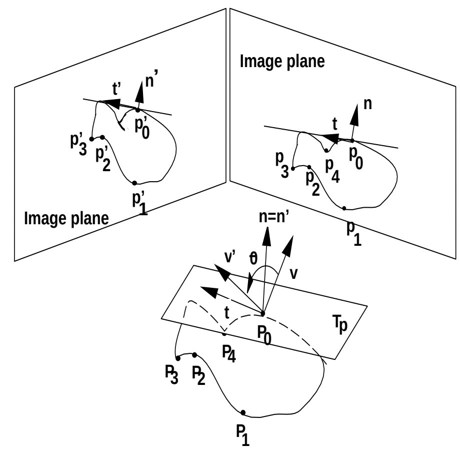

Figure 3. An illustration of motion in the tangent plane. Note that the point p_4' corresponding to the viewing direction $\mathbf{v}'$ has been omitted deliberately because such points are good candidates for self-occlusion.

in $\mathbb{P}^2$ returns all invariant curves $\Psi_j^3(t)$ that intersect a ball about the measured invariant whose radius is determined from measurement uncertainty. Note that we only need to consider indexing into a 2-dimensional space. Using the tree structure of invariant curves constructed during the modeling stage, all valid matches in $\mathbb{P}^2$ are tallied to determine the invariant curve in $\mathbb{P}^3$ that could have given rise to a specific combination of matches in $\mathbb{P}^2$. For example, if a match to a point on $\Psi_j^4(t) \in \mathbb{P}^3$ is expected, it is necessary and sufficient to have matches to 4 invariant curves, $\Psi_i^3(t) \in \mathbb{P}^2, (i = 1, \cdots, 4)$. In general, for a measurement to match a curve $\Psi_j^n(t) \in \mathbb{P}^{n-1}$, it is necessary and sufficient to have matches on all the $\binom{n}{3}$ corresponding curves $\Psi_i^3(t) \in \mathbb{P}^2$. The tallying process is recursively applied to increasing dimensions.

There is no guarantee that the result of a single match will be correct. Some matches may be missed and there may be false positives. In particular due to the combinatorics of matching, there will be many attempts to match sets of features which are not in the database. In order to reduce the number of false positives, all returned matches are subjected to two independent verification procedures described below.

When individual objects can be segmented in an image, each object in the image may have multiple bitangents or inflections. For each feature and corresponding parallel tangent points, the result of indexing and verification will provide the identity of the object. Over all features for a segmented object, a simple voting scheme can be used to select the object model from the matched features.

3.1 Verification

We now consider verification procedures that can eliminate false matches returned from indexing.

Consider two qualitative constraints. First, the sign of the curvature of the image contour at a feature point is invariant under changes of viewpoint in the common tangent plane as well as affine image trans-

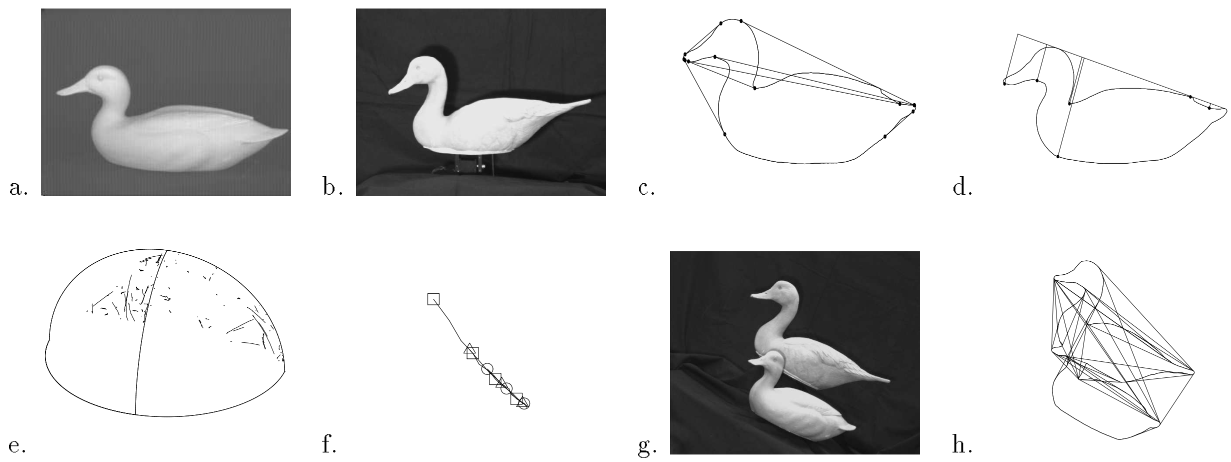

Figure 4. Image processing for modeling and testing. (a) Image of a mallard decoy. (b) a pintail decoy. (c,d) Extracted bitangents and a sample set of parallel tangents of (a). (e) Invariant curves, $\Psi_j^3(t)$ of both decoys drawn on a unit sphere embedded in $\mathbb{R}^3$. (f) Invariant curves on a portion of the unit sphere obtained from three non-overlapping trajectories for the same bitangent; they are labeled with "□", "△" and "○.' (g,h) A test image of the pintail and mallard decoys along with the bitangents extracted.

formations [3, 7]. Second, the direction toward or away from the bitangent of the outward pointing normal at a point on the image contour is also invariant.

An additional quantitative constraint can also be derived from trajectory of feature tracks used to model an object. Reconsider Figure 2. Differentiating (1) with respect to time and taking the dot product of the result with $\mathbf{n}$ yields after some simple manipulation:[2]

$$\dot{y}_i + x_i \dot{\mathbf{t}} \cdot \mathbf{n} + z_i \dot{\mathbf{v}} \cdot \mathbf{n} = 0 \qquad (2)$$

A match between measured and modeled points implies that the viewing directions for the two images must lie in the tangent plane of the matched feature; they differ by a rotation about the surface normal of the feature. This is the situation illustrated in Figure 3. Let the second viewing direction be $\mathbf{v}'$ and construct as before the corresponding coordinate system$(\mathbf{t}', \mathbf{n}', \mathbf{v}')$. Since the camera has remained in $\mathbf{T_p}$, $\mathbf{n} = \mathbf{n}'$. All of the feature point $P_i (i = 0, \cdots, n)$ will be visible in both images up to self occlusion.

Let θ be the angle of rotation about $\mathbf{n}$ which maps $\mathbf{v}$ onto $\mathbf{v}'$. The change of coordinates between the two frames can be written as

$$\begin{cases} x' = x\cos\theta - z\sin\theta, \\ y' = y, \\ z' = x\sin\theta + z\cos\theta. \end{cases} \qquad (3)$$

Combining (2) and the first row of (3) yields

$$(\dot{\mathbf{v}} \cdot \mathbf{n})(x' - x_i \cos\theta) - \sin\theta(\dot{y}_i + x_i \dot{\mathbf{t}} \cdot \mathbf{n}) = 0 \qquad (4)$$

[2] We take advantage of two facts to simplify the expression of the dot product: since the points P_0 and P_i lie on the occluding contour, the tangents to the corresponding surface curves given by $\dot{P}_0$ and $\dot{P}_i$ lie in the tangent plane i.e., $\dot{P}_i \cdot \mathbf{n} = 0$ and secondly, the vector $\mathbf{n}$ is a unit vector and is therefore orthogonal to its derivative.

For every triplet of parallel tangent features P_1, P_2, P_3 to P_0, we define vectors $\mathbf{x} = (x_1, x_2, x_3)^T$, $\mathbf{x}' = (x_1', x_2', x_3')$ and $\dot{\mathbf{y}} = (\dot{y}_1, \dot{y}_2, \dot{y}_3)$. Equation (4) can be rewritten in vector form as

$$(\dot{\mathbf{v}} \cdot \mathbf{n})(\mathbf{x}' - \cos\theta\mathbf{x}) - \sin\theta(\dot{\mathbf{y}} + (\dot{\mathbf{t}} \cdot \mathbf{n})\mathbf{x}) = 0 \qquad (5)$$

This implies that vectors $\mathbf{x}' - \cos\theta\mathbf{x}$ and $\dot{\mathbf{y}} + (\dot{\mathbf{t}} \cdot \mathbf{n})\mathbf{x}$ are linearly dependent or equivalently their cross-product is zero, i.e.,

$$(\dot{\mathbf{t}} \cdot \mathbf{n})\mathbf{x} \times \mathbf{x}' + \cos\theta\mathbf{x} \times \dot{\mathbf{y}} + \dot{\mathbf{y}} \times \mathbf{x}' = \mathbf{0} \qquad (6)$$

In turn, (6) admits a solution in $\cos\theta$ and $\dot{\mathbf{t}} \cdot \mathbf{n}$ if and only if the vectors $\mathbf{x} \times \mathbf{x}'$, $\mathbf{x} \times \dot{\mathbf{y}}$, and $\dot{\mathbf{y}} \times \mathbf{x}'$ are linearly dependent, or equivalently, if the vectors $\mathbf{x}$, $\mathbf{x}'$, and $\dot{\mathbf{y}}$ are linearly dependent. Algebraically, this condition can be written as

$$\mathrm{Det}(\mathbf{x}, \mathbf{x}', \dot{\mathbf{y}}) = 0. \qquad (7)$$

Vector $\mathbf{x}'$ is formed from triplets of components of measured feature points in the tangent direction $\mathbf{t}'$ written in the appropriate $(\mathbf{t}', \mathbf{n}', \mathbf{v}')$ coordinate system, and the vectors $\mathbf{x}$ and $\dot{\mathbf{y}}$ correspond to the hypothesized matches. Only matches satisfying the zero-determinant condition are retained, and the rest are discarded. In practice, the determinant will not evaluate to precisely zero for a correct match due to measurement noise. One approach is to simply check if determinant is within some interval about zero. However, because the entries of the matrix can have a large range of values, a single threshold is inappropriate. Because of noise, the true coordinates (x, y) are related to the measured feature coordinates $(\tilde{x}, \tilde{y})$ by $(\tilde{x}, \tilde{y}) = (x, y) + (n_x, n_y)$ where (n_x, n_y) is the image noise. Assuming only that n_x and n_y are within some

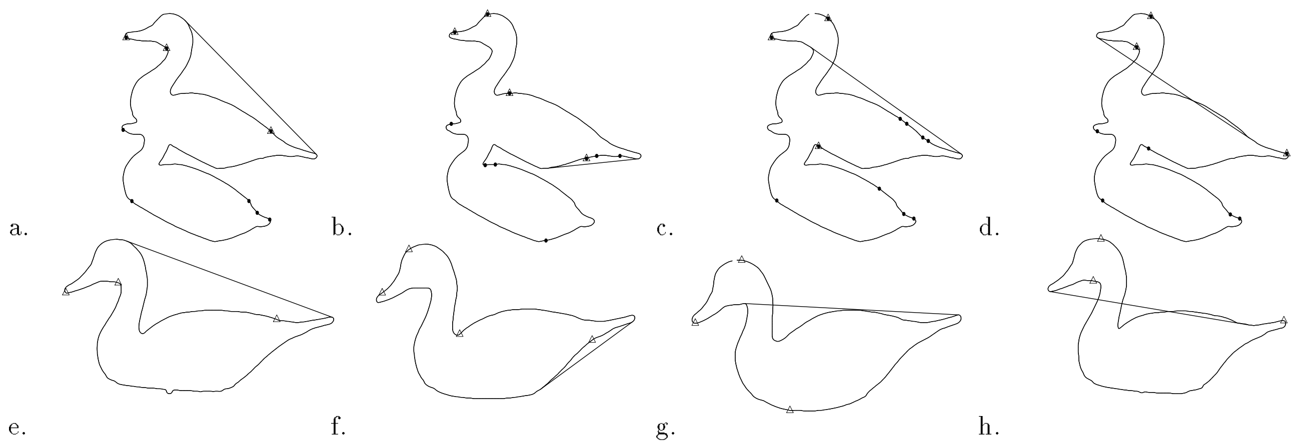

Figure 5. Recognition results: (a-d) Test bitangent and parallel tangent features. (e-h) corresponding matches from the model base after indexing and verifications. Filled circles in the test images indicate all of the parallel tangent features extracted for the indicated bitangent. The triangles in corresponding drawings show the features that were matched.

interval about zero (e.g. plus or minus one pixel), the determinant can evaluate to some interval. If zero lies within the interval, the match is considered to pass the verification test.

4 Implementation and Results

We have implemented the invariant-based modeling and recognition algorithm described in Sections 2 and 3, and will demonstrate it with two duck decoys (a mallard in Fig. 4.a and a pintail in Fig.4.b). The decoys were placed on a pan-tilt stage, and image sequences were acquired by a fixed camera with a 50mm lens under scaled orthographic projection conditions. Sequences of about 100 images were taken of each object. Each image was passed to an image processing routine: The edges were first detected via simple thresholding, the image bitangents were tracked through scale space as in [8], and the corresponding parallel tangent points were detected from zero crossings of the angle between the image contour tangent and the given bitangent. For each decoy, we retained only 9 bitangents some of which are shown in Figure 4.c. Parallel tangent features for a sample bitangent are shown in Figure 4.d. The bitangents and parallel tangent features obtained are then tracked through the image sequence to construct the invariant curves. Figure 4.e. shows the invariant curves in $\mathbb{P}^2$ for all of the bitangents of the two decoys; they are drawn on a portion of the unit sphere embedded in $\mathbb{R}^3$. The maximum dimension represented for both decoys was 5 and the minimum 3. The mallard decoy was represented by 4, 27 and 72 curves in dimensions 5, 4 and 3 respectively and the pintail by 1, 15 and 57 curves in dimensions 5, 4 and 3 respectively.

As an interesting experiment, three different sequences of the pintail duck were obtained by panning the duck at three different tilt angles. The image of the same bitangent developable surface was tracked in all three sequences, and an invariant curve in $\mathbb{P}^2$ was computed for each sequence. According to our development in Section 2, the three curves should coincide and lie on a common curve rather than defining three disjoint curves. This is demonstrated in Figure 4.f. The three overlapping curves were generated using 3 non-overlapping trajectories over a 81° of pan and 36° of tilt. This experiment demonstrates the stability with which these features can be computed and hints at their usefulness for recognition.

Figure 4.g shows an image of both decoys with partial occlusion taken on-line. 33 bitangents were extracted from this test image and are shown along with the detected silhouette in Figure 4.h. Of these 33 bitangents, only six correspond to the projection of bitangent developables used to model the pintail decoy and four to model the mallard decoy. The other bitangents were either not included in the model or have an end-point on each duck. With a model database containing both decoys, the recognition algorithm was applied to all 33 bitangents and their corresponding parallel tangent features. Figure 5 shows all of the matches that were returned. Note that for each entry in the database, we included a pointer to the image used to define the entry, and so the image can be displayed; obviously, neither the pointers nor the actual images need to be retained in an actual recognition system. Figures 5 a,b,c and d, show the matched bitangents along with all of the parallel tangents that were considered for matching. After indexing and subsequent verifications, the model images along with the matched features are shown in Figures 5 e, f, g and h respectively. Note that the four features matched the pintail decoy. The mallard could not be recognized even though four bitangents were detected and are part of the model sequence. Using simple thresholding, the mallard's head was not segmented where it occluded the pintail duck, and so there were insufficient parallel tangents to recognize the mallard.

In contrast to the previous example, a second example in Figure 6 shows an image of the pintail duck taken from a viewpoint which was drastically different than any viewpoint used to model it. Figure 6.c shows a match when indexing the model database using the features defined by the bitangent between the beak and head. Note the significant rotation about the surface normal between the two images. Additional examples have been presented in [14].

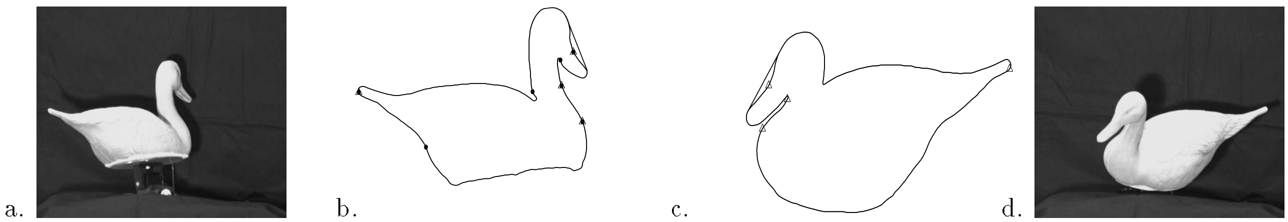

Figure 6. A second recognition example with the test image significantly different from the images used for modeling. (a) Test image. (b) Feature for which match was found. (c-d) Returned match and image used for modeling.

5 Discussion

Though only parallel tangents were considered in this paper, it is easy to show that any viewpoint dependent feature such as a vertex can be used in exactly the same manner; in fact, some of the parallel tangents arise from the image of surface creases. Except for a simply voting on an object's identity, each of the bitangents is treated independently. We are hoping to find a mechanism for determining if a set of matches is mutually feasible; this will offer greater discriminating power than simply voting.

Recognition will only be reliable if the stored representation of a model is complete. Therefore, one needs to extract as much information as possible during the modeling process. Here, this means complete invariant curves. By moving the camera along a trajectory that fully reveals each limiting bitangent curve, entire lengths of the associated parallel tangent curves will also be exposed. During modeling, it will be possible to automatically select the next camera position using image data. Presently, we have used fixed image sequences to model the object and not actively selected new camera positions.

Determining the surface geometry of objects from images is a fundamental problem in computer vision. Methods are available for reconstructing a 3D surface from the image profile when the camera motion is known [4, 2]. Now consider moving an object with unknown motion, perhaps by simply manually turning it in front of a camera. When two viewpoints differ by a rotation about the normal of some bitangent developable, they can be identified using the presented invariant and verification procedure. From this correspondence, constraints on the camera velocity can be determined. It may be possible to fully determine the camera motion from these constraints; existing methods can then be applied to reconstruct the surface.

Finally, the presented method clearly only works for objects which are complicated enough to actually have a bitangent developable and parabolic curves with the corresponding parallel tangents. Thus, the presented invariant should only be considered a component of a complete recognition system.

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. We gratefully acknowledge Tanuja Joshi for some very useful comments.

References

[1] B. Burns, R. Weiss, and E. Riseman. The non-existence of general-case view-invariants. In *Geometric Invariance in Computer Vision*, pages 120–131. MIT Press, 1992.

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

[3] R. Cipolla and A. Zisserman. Qualitative surface shape from deformation of image curves. *Int. J. Computer Vision*, 8(1):53–69, 1992.

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

[5] D. W. Jacobs. Space efficient 3D model indexing. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 439–444, 1992.

[6] T. Joshi, J. Ponce, B. Vijayakumar, and D. Kriegman. HOT curves for modelling and recognition of smooth curved 3D shapes. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, June 1994. In Press.

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

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

[9] K. Kutulakos and C. Dyer. Occluding contour detection using affine invariants and purposive viewpoint adjustment. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 323–329, 1994.

[10] Y. Lambdan, J. Schwartz, and H. Wolfson. Affine invariant model-based object recognition. *IEEE Trans. on Robotics and Automation*, 6:578–589, 1990.

[11] Y. Moses and S. Ullman. Limitations of non model-based recognition schemes. In *European Conf. on Computer Vision*, pages 820–828, 1992.

[12] J. Mundy and A. Zisserman. *Geometric Invariance in Computer Vision.* MIT Press, Cambridge, Mass., 1992.

[13] H. Murase and S. Nayar. Illumination planning for object recognition in structured environments. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 31–38, Seattle, WA, 1994.

[14] B. Vijayakumar, D. Kriegman, and J. Ponce. Invariant-based recognition of complex 3D curved objects from image contours. Technical Report 9411, Yale Center for Systems Science, 1994. Available via anonymous ftp on daneel.eng.yale.edu.

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