MORSE: A 3D Object Recognition System Based on Geometric Invariants

J.L. Mundy[*], **C. Huang**, **J. Liu** and **W. Hoffman**
Box 8
G.E. Corporate Research and Development
Schenectady, NY 12309

D.A. Forsyth
U.C. Berkeley
Berkeley, CA

C.A. Rothwell
INRIA
Sophia Antipolis, FR

A. Zisserman and **S. Utcke**
Department of Engineering Science
Oxford University, Oxford, UK

O. Bournez
ENS de Lyon
Lyon, FR

Abstract

MORSE is an object recognition system, based on geometric invariants of 3D structures taken from a single 2D intensity view. The system exploits the geometric constraints inherent in object classes such as polyhedra, rotational symmetry, bi-lateral symmetry and extruded surfaces. Invariants have been used in the past to index many of these classes, but MORSE is designed to treat multi-class recognition in a unform system architecture. The class constraints are also used to drive image feature extraction and grouping.

1 Invariant Representation

The computer recognition of objects has attracted considerable research effort over the last 25 years. It is now widely accepted that object recognition, in the setting of real world scenes and based on a single perspective view, is a difficult problem and cannot be achieved without the use of object models to guide the processing of image data and to confirm object hypotheses. It is also accepted that the most reliable information which is available in a scene is derived from a geometric description of the object based on its projection in the form of 2D geometric image features, as opposed to, for example, its intensity shading. Thus, object recognition systems draw on a library of geometric models, which usually contain information about the shape and appearance of a set of known objects, to determine which, if any, of those objects appear in a given image or image sequence. Recognition is considered successful if the geometric configuration in an image can be explained as a perspective projection of a geometric model of the object.

A major constraint underlying the work presented here is that recognition is based on one uncalibrated view of a scene. Our motivation is that this restriction applies in many of the current and future applications for object recognition, such as aerial surveillance, image database query processing, image-hypertext editing, and scene construction for virtual reality.

Even if more images are available, for example in the case of video processing, camera calibration will not generally be known initially. Any grouping, recognition hypothesis, or object recovered up to some ambiguity from a single image, can be propagated to advantage to subsequent views.

This paper describes a number of examples of 3D object *classes* that can be recognized from a single image. These classes are defined geometrically, as opposed to *function*[16] or other ontological categories. For example a surface of revolution is a geometric object class,

[*]The research reported here is funded in part by DARPA Contract #MDA972-91-C-0053

as opposed to a type of vase, which would be functional. Rather than identifying a particular model directly, recognition is class based, proceeding by first classifying based on image curves, and subsequently identifying a particular object.

We present here a prototype class based recognition architecture, which integrates these ideas. Class informs each level of the architecture from image grouping through to organization of the model base and finally the constraints imposed by object classes on the 3D scene configuration.

2 The MORSE System

These ideas about 3D invariant class representation and recognition from a single view form the basis for a new object recognition system called MORSE[1]

2.1 The Architecture

Control Representation is organized into a number of layers as illustrated in figure 1. These stages of representation are not very different from other recognition architectures, however the main principles of *class* and *global consistency* provide a framework for control and geometric data query management.

Segmentation and grouping The key to successful recognition is efficient and robust feature segmentation and grouping. There are four levels of image feature representation and grouping:

Level I: **Pixel-level features** are defined with respect to an image coordinate system and reflect the quantized nature of pixel coordinates. Typically, features will be produced using an edge operator with subpixel accuracy, and the resulting edgels linked into list. This level is topological - linked curves and vertices are represented.

Level II: **Geometric features** curves from level I are described in terms of geometric primitives, where appropriate. For example: algebraic curves, smooth curves, concavities.

Level III: **Generic Grouped features** This level of grouping is applied to all features produced at level II. The output is a number of groupings and databases which are used by the class-based groupers described below. Generic grouping includes: near incidence (jumping small gaps, completing corners and junctions); collinearity; marking bitangent and other distinguished points; affine or projective equivalence of curve segments (e.g. concavities). Pairing up concavities uses the distinguished points provided by bitangents and associated cast tangents.

Level IV: **Class based grouping** Each class has an associated "class based grouper" or agent that interrogates the level III groupings and databases, and attempts to form groups appropriate for its class. If successful, class based invariants are extracted from the grouped features and directly index the relevant class library. An example of the class specific grouping mechanism is described in section 3.

Indexing and Hypothesis Combination Indexing is handled by a series of hash-tables, one per class, that take the invariants of a system of generalized features and associate with them models in the modelbase. For complex objects, there may be many feature groups that index to the object, leading to a situation where a single instance could cause many verifications. The number of potential hypotheses can be reduced by forming joint hypotheses (cliques), based on either topological or invariant geometric relations between feature groups that have indexed to the same model instance.

The Modelbase This has three components:

- *Model class properties*, which implements the class constraints to enable grouping, correspondence and indexing. The invariant properties of the classes to be implemented in MORSE are summarized in table 1.
- *Object property model base*, which contains information on object properties that follow from its identity as an object, such as orientation with respect to other objects; - it is at this level that *scene-level knowledge* is stored.
- *Retrieval mechanisms*, which will normally consist of hash-tables, used to associate feature groups with objects.

The Scene An additional source of constraints and parameters is the 3D scene, which can also be viewed as a database which reflects the current configuration of the world and cameras. It provides a representation of all information currently available about the common Euclidean frame in which objects reside.

Verification Verification proceeds at many levels of the formation of a hypothesis. Many model-based vision systems only apply verification at the final object instance hypothesis level. In MORSE, intermediate stages of representation and grouping are also verified with respect to information at lower levels of representation. For example, a set of polyhedral face hypotheses can be refined and verified by applying a "snake" defined by polyhedral incidence and projection constraints. This idea can be extended even to modeling and verifying local image intensity surface events, such as corners.

2.2 Model Acquisition

In MORSE models are directly acquired from multiple views[2]. The fact that such models can serve as sufficient representations for recognition is a major advantage of the invariant approach. We expect that only a small number of views will be required for most objects and that these views will be defined by the extraction of a sufficient set of stable features over a wide range of viewpoints.

[1] The acronym is Multiple Object Recognition System by Scene Entailment and is named after an Oxford detective character.

[2] For some classes, such as a rotationally symmetric object, a single view will suffice.

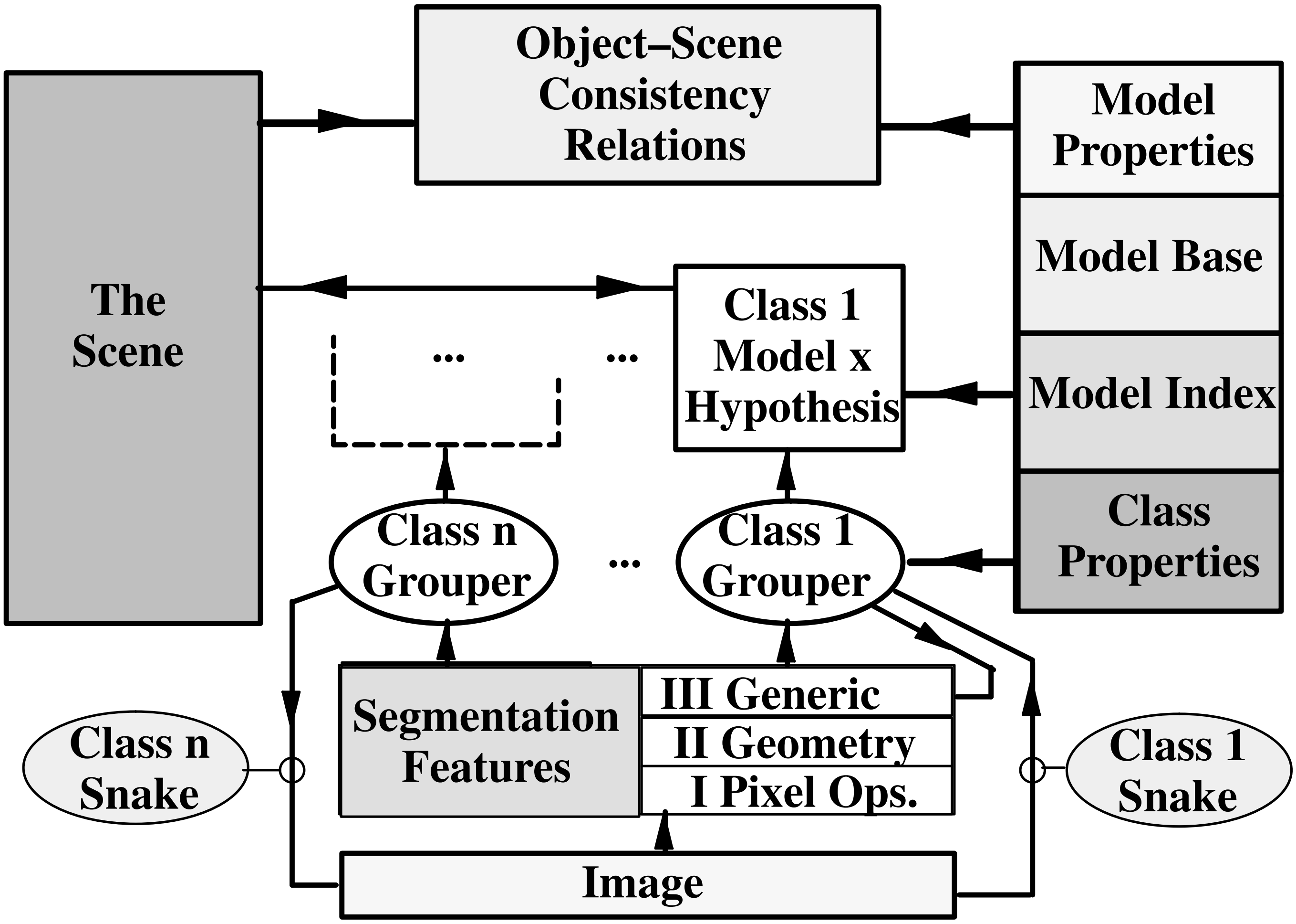


Figure 1: The proposed architecture for object recognition. The architecture is organized around geometric *classes* which define grouping and indexing mechanisms as well as 3D scene constraints.

Euclidean object descriptions are required for full scene consistency techniques to work. One approach is to derive the Euclidean properties from self-calibrated camera views. Recent work on self-calibration [3, 6] demonstrates that it is possible to derive the internal camera parameters based on three or more general views with a single camera. Other assumptions about camera motion between views can also be used to constraint internal parameters. With a sufficient number of views, scaled Euclidean reconstruction appears to be quite practical, even with no a priori information about the camera acquisition viewpoints and 3D world geometry.

2.3 Implementation

The system is being implemented in C++ using a class hierarchy closely related to the Image Understanding Environment (IUE) [7], called Target Jr. The IUE is an ARPA funded project to produce an object-oriented programming environment for vision research. A central object hierarchy in the IUE is the *spatial-object* which incorporates many of the descriptive requirements described in the previous sections. The IUE also has an extensive set of classes for object and image transformations which are a central issue in MORSE. Target Jr. has a similar class hierarchy and has been under development at GE-CRD over the last 5 years. MORSE will be ported to the IUE when the initial implementation is available.

3 Class Driven Grouping

We illustrate the idea that class constraints allow specific grouping mechanisms by describing in detail grouping for surfaces of revolution. Other grouping mechanisms are then sketched briefly at the end of this section.

3.1 Surfaces of Revolution

The profile (image outline) of a surface of revolution can be separated into two 'sides' by the projected symmetry axis. The two sides are tightly constrained - it can be shown that they are related by a particular four degree of freedom projective transformation—a planar harmonic homology [17]. This relationship is exact. Further, under general imaging conditions, this projective relation can be approximated by a three degree of freedom affine transformation. This is an excellent approximation (as shown empirically below), degrading only as the field of view increases [8].

We next summarize the mathematical properties of the affine transformation [9], and then show how these are used to simplify and improve curve grouping.

1. Corresponding points $\mathbf{x}'$ and $\mathbf{x}$ on the two profile sides are related by the affine transformation $\mathbf{x}' = \mathtt{A}\mathbf{x} + \mathbf{b}$. (Corresponding points arise from the same circular surface cross-section). The transformation is an involution, so $\mathbf{x} = \mathtt{A}\mathbf{x}' + \mathbf{b}$.
2. The transformation has three degrees of freedom—two to specify the "symmetry" line (the projection of the 3D axis of revolution), and one for the correspondence direction. See figure 2a.
3. The matrix $\mathtt{A}$ satisfies $\mathtt{A}^2 = \mathtt{I}$. It has eigenvectors $\mathbf{a}$ and $\mathbf{b}$ ($\mathbf{b}$ as above) with eigenvalues $+1$ and -1 respectively. Vector $\mathbf{a}$ is parallel to the symmetry axis, vector $\mathbf{b}$ is parallel to $\mathbf{x}' - \mathbf{x}$ - i.e. the correspondence direction.
4. The equation of the symmetry line is: $2a_y x - 2a_x y - a_y b_x + a_x b_y = 0$

The transformation between the two sides is determined using a type of *affine snake*. This is computed in two stages: first, an approximate solution is determined by matching a number of distinguished points (such as bitangent contact points); second, the approximate solution is used to transform a number of sample points from one profile side to the other. The squared distance between the transformed sample points and the other profile side is then minimized numerically over the the three parameters of $\{\mathtt{A}, \mathbf{b}\}$. Effectively this uses one side of the profile to define a snake, and then determines the affine transformation (constrained to have the above properties) which most closely aligns it with the other side. Ten sample points are usually sufficient and the minimization converges within ten iterations generally to an average distance of less than a pixel. A typical example is shown in figure 2b. The transformation is parameterized by: correspondence angle, θ_c; symmetry line angle, θ_s; and distance of the symmetry line from the origin, r_s.

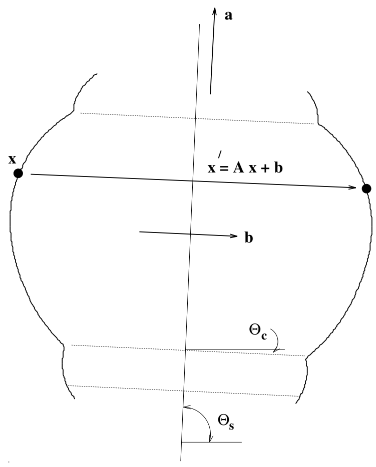
After generic grouping, the class based grouper has the task of determining which curves could have arisen from a surface of revolution, and repairing, where possible, missing curve segments due to occlusion and the usual problems of segmentation. In particular profile curves are often broken into a number of portions. Since the image may well contain several surfaces of revolution the grouper must also partition the profile curves into those arising from different surfaces.

Grouping proceeds in two stages:

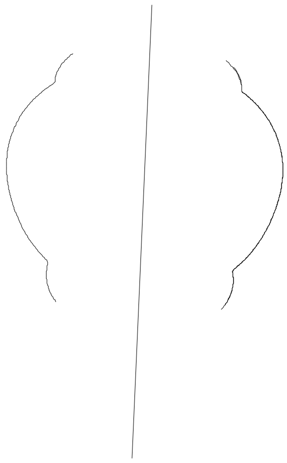
1. **Group conjugal curve fragments** Initially data consists of a set of curve fragments with bitangents identified. The two sides of the profile are affine related, and consequently corresponding profile curves have the same affine invariants. Associating conjugal (corresponding) curve fragments on both side is then a matter of matching curve segments with the same affine invariants. This is an intra-image version of the inter-imaging invariant matching used in model based vision [10]. Currently, area, which is a relative affine invariant, is used to evaluate pairings. This matching has complexity n^2 in the n concavities, where typically $n = 25$ for a cluttered scene containing two surfaces of revolution. A number of these putative matches can then be eliminated because the affine transformation between the curves does not have the additional properties listed above. The transformation is determined using the affine snake. The conclusion of this stage of grouping is a set of paired curve fragments with the three parameters of the transformation $(\theta_c, \theta_s, r_s)$ known in each case.
2. **Group profile fragments** It then remains to

Class	Invariant Structure
Polyhedron[14]	When faces have four or more vertices, 3D invariants are derived from the face incidence structure. Incidence relations are used to eliminate projection parameters.
Rotational Symmetry[8, 18]	A restricted planar projective homology exists between opposite image curves. The transformation is approximately affine and defines the central axis in the image. Invariants are cross-ratios of distinguished points on the axis.
Bilateral Symmetry[14, 18]	Distinguished points on the bi-lateral symmetry plane define planar invariants. The points are constructed by establishing symmetrical feature correspondences.
Extruded Structures	Invariants are constructed on the line of intersection of the terminal planar crosssections of the extruded surface. Corresponding points on the crosssectional curves intersect at a common vertex.
Canal Surfaces[12]	A canal surface is formed by sweeping a sphere along an axis curve. The projection of the axis curve is the symmetry set of the projected surface outline. Invariants are defined by projecting the axis curve into a canonical frame.
Repeated Structures[11]	General symmetries can be viewed as a repeated structure. Symmetrical feature correspondences define 3D invariants of the structure through epipolar reconstruction.
Algebraic Surfaces[5]	Three dimensional projective invariants of algebraic surfaces of degree 3 or higher are constructed from the image projective invariants of the surface outline curve
Ruled Surfaces	Two independent ruling systems are constructed from tangents to the image outline. An invariant projective coordinate frame for surface markings is established with respect to the rulings.

Table 1: The set of 3D surface classes for which geometric invariants can be computed from a single view.



a



b

Figure 2: *(a) The two sides of the profile of a surface of revolution are related by a particular type of affine transformation (an involution). Vector* $\mathbf{a}$ *is parallel to the skewed symmetry axis, vector* $\mathbf{b}$ *is parallel to* $\mathbf{x}' - \mathbf{x}$ *- i.e. the correspondence direction. A point on the left side is transformed to the right by translating in the correspondence direction twice the distance between the left side and the symmetry axis. (b) A typical example of the accuracy of this transformation on real images. The transformation is computed using an affine snake, as described in the text, and is used to map the left side onto the right. The two curves (original right profile, and transformed left) are indistinguishable. The parameters of the transformation also define the symmetry line shown. In this example the three parameters are* $\theta_c = 2.7$, $\theta_s = 87.4$ *and* $r = 329.8$.

#'s	areas pixels^2	av. error pixels	parameters $\theta_c/^o$ $\theta_s/^o$ r/pixels
1 6	181.7 190.8	0.23	5.2 96.5 260.1
1 9	181.7 197.9	no match	
2 5	428.6 442.8	0.43	8.5 98.7 271.3
3 8	242.9 263.1	no match	
4 8	275.1 263.1	no match	
4 10	275.1 277.0	no match	
6 9	190.8 197.9	no match	
8 10	263.1 277.0	0.48	6.5 96.1 257.7

Table 2: Concavities are matched on area (which is a relative affine invariant), followed by the affine snake which enforces involution. Only the symmetry related concavities have a low error. The transformation parameters computed from each matched concavity pair are very similar. These parameters are used to group the concavities.

group curve fragments which may have arisen from the same profile curve. Grouping is based on the similarity of $\{\theta_c, \theta_s, r_s\}$ and corresponding pairs of matched concavities are aligned along their common central axis. The associated outline curve fragments are joined using existing local edgel chain topology and smooth curve continuation.

An example is shown in figure 3 and table 2 of the concavities and matching process. The transformation parameters extracted from different concavity pairings on the same profile are very stable, varying by less than 5^o over the three concavities.

Indexing for surfaces of revolution proceeds by hashing based on invariants [8], and then verifying first by alignment on the axis distinguished points (generated by bitangent intersections), then on the profile. The number of distinguished points is restricted to prevent a combinatorial explosion in the number of invariants. The restriction is achieved by only considering bitangents between convex curve portions (convex defined to be facing out from the symmetry axis). This condition is preserved under perspective projections. Restricting to only convex-convex bitangents has an additional advantage - due to self-occlusion concave-concave and concave-convex bitangents may be missing, since concave profile portions are the first to be self-occluded. Consequently distinguished points formed from these bitangents may well be missing. Typically a model has 5-10 distinguished points from convex-convex bitangents out of a total of 25-50 distinguished points of all types. This bitangent classification scheme has been successfully tested on object model outline curves, which are well formed and do not have missing distinguished points. Work is still in progress on applying the new indexing approach to complex scenes, where only a subset of the distinguished points in a full model can be recovered. .

3.2 Polyhedral grouping

In this initial implementation of MORSE, we have implemented class-based grouping mechanisms for two subclasses of polyhedra,

polyhedra which are projectively equivalent to a cube,

polyhedra projectively equivalent to a triangular prism.

The grouping process is based on the assumption that the interior boundaries of a polyhedron are only partially recovered. The outer, occluding boundaries of an object usually have more contrast against the background than the interior boundary edges. For example, the contrast across a polyhedral edge is small if the light source direction is oriented approximately equally with respect to the face normals on each side of the edge.

The initial stages of grouping are typical, including collinear edge extension and vertex definition by intersection of nearly incident edge segments. Hypothesized polyhedral outlines are verified and geometrically refined by class-specific snakes. For example, only the hexagonal outline of a cube may be recovered as shown in Figure 4a. This outline then generates a cube-class polyhedral snake which incorporates the complete polyhedral structure, including the missing interior boundaries. The snake then attempts to optimize on interior boundaries, using the extremal outline as a constraint, as well as the projective projection constraints for a cube[3]. The final, converged, state of the cube-class snake is shown in Figure 4b. The fitted polyhedral vertices are then available for use in establishing the camera model for the scene and to recover specific polyhedral model instances which are projectively equivalent to one of the generic classes, but differ in their Euclidean geometry.

4 Scene Constraints

In the current implementation of MORSE, we exploit two universal scene constraints which do not depend on object class or on assumptions about the scene (e.g. objects have coplanar bases).

Viewpoint Consistency - The image is formed by a single, uncalibrated camera. Therefore the camera parameters recovered for a set of objects should be the same for any combination of objects in the scene.

Euclidean Consistency - All objects in the scene share a common Euclidean frame. That is, the scene is constructed by placing Euclidean models of specific objects. As a consequence, the transformation between the exterior orientations of two cameras derived from two different objects should be related solely by a 3D Euclidean transformation.

Up to this point objects of each class have been recognized using projective invariants. However, using projective invariants alone it is not possible to distinguish objects that are projectively equivalent. For example, the two cuboids shown in figure 4 are Euclidean *inequivalent* (the sides on one object are a different length to the sides on another), but projectively *equivalent*.

[3] For example, a cube has three major directions which define a triple of vanishing points in the image. All edges aligned with a major direction must pass through the same vanishing point.

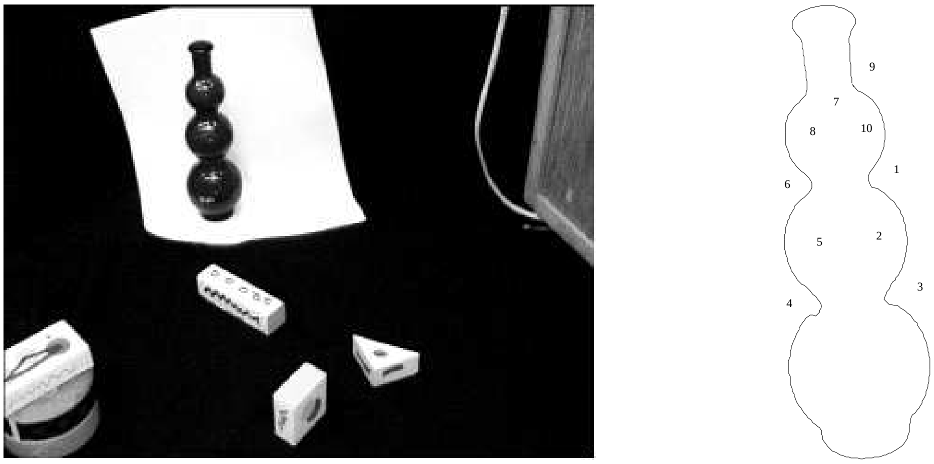


Figure 3: *Example of class based grouping for surfaces of revolution. (a) Original image, 11 concavities are identified. (b) Detail of edges, showing numbering for concavities. The matched concavities and their parameters are given in table2.*

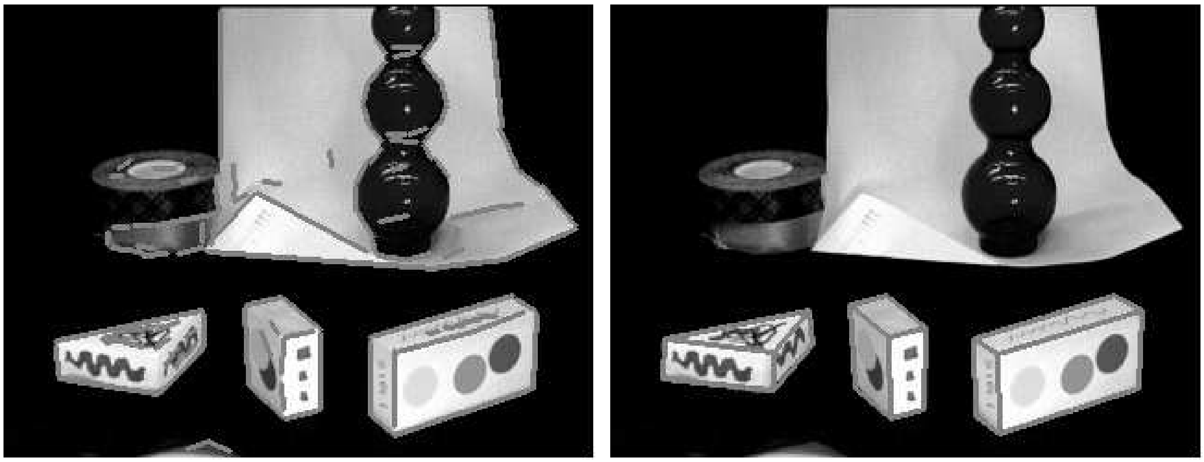


Figure 4: (a) The straight line segmentation found by edge detection and line fitting. (b) The final state of the class-specific snake. Note that all the interior boundaries are now recovered.

model and correspondence	snake 2	snake 3
1, 1	1.66	0.42
1, 2	0.68	0.56
1, 3	4.73	1.04
1, 4	0.49	1.78
1, 5	2.94	1.22
1, 6	0.80	1.81
2, 1	0.89	0.38
2, 2	5.56	1.54
2, 3	0.49	2.72
3, 1	2.38	0.66
3, 2	0.76	0.94
3, 3	1.49	1.15

Table 3: Aspect ratios for cameras computed using specific model-to-snake correspondences. The snakes are numbered from the left; snake 1 is a prism, and so no effective camera estimate can be made. For snakes 2 and 3, the camera is computed using each of the set of possible model face to image face correspondences for each object. Only model-snake pairs with an aspect ratio that lies between 1.5 and 0.7 are considered further.

For each pair of projectively equivalent objects, specific Euclidean instances are hypothesized for each of the two objects. Fully-calibrated cameras are derived for each object, using the hypothesized Euclidean models. If these cameras have identical interior calibration parameters, then the objects are related in 3D by a Euclidean transformation, and are thus a consistent interpretation of the scene.

The intrinsic parameters for each camera are computed by QR decomposition (the product of an upper triangular and rotation matrix). The aspect ratio is the ratio of the first and middle diagonal element of the upper triangular matrix. The consistency of interior parameters are shown in Table 3. We base consistency primarily on the pixel aspect ratio, since this parameter is more stablily recovered, with respect to image feature location errors. Model hypothesis pairs are then tested for consistency with one another by testing whether $\mathtt{C}_a^{-1}\mathtt{C}_b$ is nearly Euclidean, where $\mathtt{C}_i$ is the left 3×3 block of a 3×4 camera matrix. Pairs that pass this test are used to estimate a joint camera. If the joint camera yields an image projection with an average error of less than one pixel[4], the reconstruction is considered consistent and further hypothesis formation can proceed. From the experiments, it appears that the aspect ratio computation is often quite unstable. This instability may be due to the limited number of points, e.g., 7 for a cube, used in the camera computation. We are currently investigating this issue in an attempt to discover a more reliable measure of consistency. The Hessian of the camera optimization function can also be used an expected variance in the calibration parameters and define a consistency threshold.

[4]In the case shown, the average error is 0.64 pixels

The individual camera models are used as an initial guess for a more accurate joint camera resection solution which is carried out on both objects simultaneously. The resection optimizes image projection error subject to the constraint that the two cameras have identical interior parameters.

5 Experimental Results

In these experiments, two model classes have been implemented: polyhedral and surface of revolution. The system is still in the initial stages of implementation and integration, so a number of steps are carried out by coupling to external code implementations by file exchange. Even so, examples of many of the basic components of the MORSE architecture are available and can carry out automatic scene processing. The polyhedral section is implemented largely in Lisp and Mathematica procedures with file transfers to access Target Jr's scene data structure. The rotational symmetry portion is more closely integrated, in C++, with some external procedures for optimizing the planar affine transformation between symmetrical features.

An example is shown in figure 6. Euclidean consistency is used to determine the Euclidean cuboid models and hence the camera in figure 4. Using the Euclidean models and camera, the pose is recovered for all the objects in the scene, and images rendered from a different viewpoint to illustrate the 3D scene configuration. Properties of the surface of revolution class have been successfully used to automatically identify and group the corresponding sides of the boundary and to find the central axis. However in this result, the final placement of the object is based on known correspondences on the axis. Work is still in progress to integrate the model indexing process for rotationally symmetric objects.

As a further illustration of grouping for surfaces of revolution, Figure 5 shows the extraction of two profile curves from a typical image. The matched concavities are partitioned into sets, and the profile curves corresponding to each set grouped. The entire process is automatic and relies only on the affine properties of the homology between symmetrical portions of the outline.

6 Conclusions and Future Extensions

Even at this early stage of implementation, the notion of class-based grouping and scene consistency analysis appears promising. The constraints provided by class, particularly in the case of surfaces of revolution, provide strong grouping mechanisms. We have successfully demonstrated that the association of corresponding bitangent concavities can be reliably carried out using simple affine invariants. This result is encouraging, since bitangents can be robustly constructed and classified.

The current polyhedral object recognition rate is low due to a combination of fragile polyhedral face grouping and the instability of scene camera recovery. It is clear that in the case of polyhedra, it is necessary to exploit sub-classes, such as rectangular or triangular prisms to gain a sufficient number of constraints to recover or predict internal boundaries. The experiments have con-

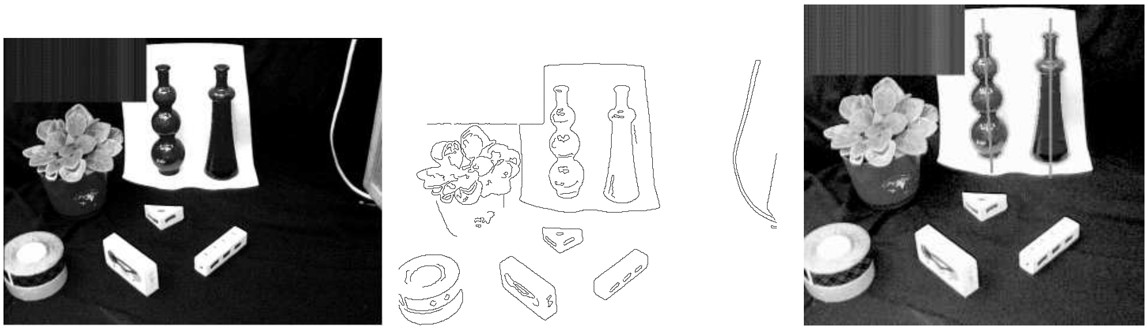


Figure 5: *Class based grouping used to automatically extract two profiles of the surfaces of revolution. (a) Original images. (b) Canny edges. (c) Extracted grouped profiles and skewed symmetry axes. 24 concavities are extracted. Matching on area gives 48 possible pairings. Of these only two groups also satisfy the affine involution, one group with two members, the other with three. The axes and outline curves are indicated by a light grey overlay.*

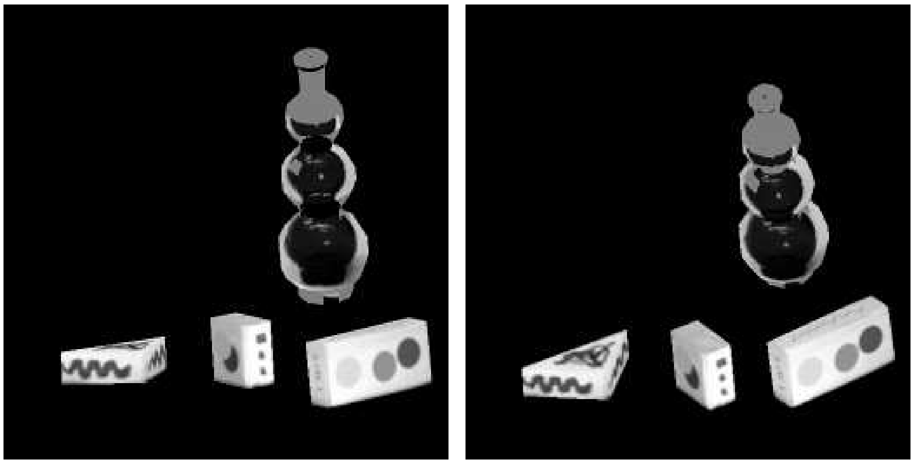


Figure 6: *Graphic renderings showing recognized Euclidean objects and surface of revolution from the image of figure 4. The polyhedral objects are recognized and their pose recovered. A consistent camera is automatically derived for the scene and used to place the rotationally symmetric object, based on manually determined correspondences for distinguished points on the symmetry axis. Texture is rendered from the original image. Note that part of the surface texture of the rotationally symmetric object is missing, since it was clipped in the source image of figure 4.*

firmed that occluding boundaries are much more reliably recovered than internal edges. The use of polyhedral snakes proved successful in acquiring the complete boundary of a polyhedron which is difficult to construct by fully bottom-up methods.

At the scene level, the recognition of even a few simple polyhedra can be used to construct a camera calibration which in turn greatly reduces the complexity of search for additional hypotheses. For example, in the case of figure 6, the triangular prism was not recognized in the first pass of hypothesis formation. However, given the camera derived from the other two rectangular prisms, the triangular prism could be recovered. A single, rotationally symmetric, object hypothesis does not provide enough information to fully determine the camera and thus place the object in 3D space.

A calibrated camera is necessary to place rotationally symmetric object models in the 3D scene, given distinguished points on the axis. It should be noted that all of the analysis reported here is carried out without any use of observations of the crosssection curve of the rotationally symmetric object. The crossection curve, a circle in 3D, does provide additional calibration and placement constraints.

In the near future we will carry out the following extensions.

1. Exploit the local topology of edgel chains to provide more complete curve topology in support of feature extraction for both classes.
2. Improve the scene consistency analysis by providing tolerances on internal camera parameters and explore the use of more stable consistency measures.
3. More fully integrate the components for rotationally symmetric and polyhedral classes into Target Jr.

References

[1] Binford T.O. and Levitt T.S., 'Quasi-Invariants: Theory and Explanation'. *Proc. Darpa IUW*, pp. 819–829, 1993.

[2] Faugeras, O., "What can be Seen in Three Dimensions with an Uncalibrated Stereo Rig?" *Proc. ECCV*, LNCS 588, Springer-Verlag, p.563-578, 1992.

[3] Faugeras, O., Luong, Q.T. and Maybank, S.J. "Camera Self-Calibration: Theory and Experiments", *Proc. ECCV*, LNCS 588, Springer-Verlag, 1992.

[4] Forsyth D.A., Mundy J.L., Zisserman, A. and Rothwell C.A., 'Recognizing rotationally symmetric surfaces from their outlines'. *Proc. ECCV*, LNCS 588, Springer-Verlag, 1992.

[5] Forsyth, D.A., "Recognizing Algebraic Surfaces from their Outlines," To appear *International J. of Computer Vision*, 1994.

[6] Hartley, R.I., "Euclidean Reconstruction from Uncalibrated Views", in *Applications of Invariance in Computer Vision*, Mundy, J.L., Zisserman, A. and Forsyth, D.A. (eds), LNCS 825, Springer-Verlag, 1994.

[7] Kohl, C. and Mundy, J.L., "The Development of the Image Understanding Environment," *Proc. CVPR*, 1994, p.443.

[8] Liu J.S., Mundy J.L., Forsyth D.A., Zisserman A. and Rothwell C.A., 'Efficient Recognition of Rotationally Symmetric Surfaces and Straight Homogeneous Generalized Cylinders', *Proc. CVPR*, 1993.

[9] Mukherjee D.P., Zisserman A. and Brady J.M., 'Shape from symmetry—detecting and exploiting symmetry in affine images'. To appear, *Proc. Royal Soc.*, 1994.

[10] Mundy J.L. and Zisserman A.P., *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[11] Mundy J.L. and Zisserman A. 'Repeated Structures: Image Correspondence Constraints and 3D Structure Recovery.' In *Applications of Invariance in Computer Vision*, Mundy, J.L., Zisserman, A. and Forsyth, D.A. (eds), LNCS 825, Springer-Verlag, 1994.

[12] Pillow N., Utcke S. and Zisserman A. 'Viewpoint-Invariant Representation of Generalized Cylinders Using the Symmetry Set', To appear, *Proc. British Machine Vision Conference*, 1994.

[13] Rothwell C.A., Zisserman A., Forsyth D.A. and Mundy J.L., 'Canonical Frames for Planar Object Recognition'. *Proc. ECCV*, LNCS 588, Springer-Verlag, 1992.

[14] Rothwell, C.A., Forsyth, D.A., Zisserman, A. and Mundy, J.L. "Extracting Projective Structure from Single Perspective Views of 3D Point Sets", *ICCV*, 573-582, 1993.

[15] Rothwell C.A., Zisserman A., Forsyth D.A. and Mundy J.L., 'Planar Object Recognition using Projective Shape Representation'. To appear, *IJCV*, 1994.

[16] Stark, L. and Bowyer, K., "Generic Recognition Through Qualitative Reasoning About 3D Shape and Object Function," *Proc. CVPR*, 1991, p.251.

[17] Springer C.E., *Geometry and Analysis of Projective Spaces*. Freeman, 1964.

[18] Zisserman A., Forsyth D., Mundy J., Rothwell C. and Liu J. '3D Object Recognition using Invariance', *Oxford University Engineering Report*, OUEL 2027/94, 1994.