Artificial Intelligence Review (1987) **1**, 183–200

Representing three-dimensional structures for visual recognition

R. B. Fisher
Department of Artificial Intelligence, University of Edinburg
Edinburgh EH1 2QL, Scotland

Abstract. Three-dimensional (3-D) geometrical models provide the be representations for 3-D objects. Not all representation schemes are su able, however, for computer-based visual recognition. This survey ar lyses the historical development of recognition-oriented models fro points and lines, to surfaces and volumes. It also considers those aspects the models that successfully promoted recognition, and suggests like areas for future development.

Introduction

Interest in computer-based visual recognition of 3-D objects is increasing. Th stems partly from the need to solve many of the problems associated with quick a competent recognition of 2-D structures, and partly from more applications invol ing 3-D objects (e.g. assembly line, autonomous navigation). As a result, it h become more important to represent 3-D objects for recognition.

Scenes that contain 3-D structures have new problems not present in 2-D scen One problem is occlusion — because objects can lie in front of each other. Th results in some features being hidden. A second problem is that small motions the object or observer might radically alter the object's appearance, and hence t data needed for recognition. Finally, some image features may be ambiguou depending on the type of data used. For example, in a black and white image, image edge could arise from several different scene phenomena (e.g. shadov changes in reflectance, object edges, etc.). Unfortunately, all these appear virtua identical in an image.

One approach to representing a 3-D object is to represent its typical appearar — which is usually a set of 2-D views. This is satisfactory for simple or higl symmetric objects, but is unlikely to provide general capability. Another problem that more complicated objects often have many (e.g. hundreds) topologically d tinct views, so representation by a few views is impossible. Hence, this approa seems unlikely to be useful in the future, and is not considered further here.

In this review, our major concern is with 3-D models, which represent features, such as lines, surfaces or volumes. An example is a 3-D wire-frame mo of a cube (compared with a 2-D drawing of the same cube). An object descript

usually consists of a set of features, with their individual description overall description of how to put them all together (typically relative coordinate system).

The 3-D object model represents aspects of the object's shape, rathe appearance. Since computer vision now knows more about how app related to shape, if we know the shape, it is always feasible to predict tl ance. Further, the model implicitly represents all appearances, rather th them all listed. Moreover, many other properties are usually explicit model, such as the physical sizes of features and how their pos related.

The major advantage that 3-D models offer is that they explicitly rep 3-D structure and feature inter-relationships. This allows (i) direct pairin features to data features (assuming the data has a 3-D character), (ii) dire tion of object position using the correspondence, and (iii) prediction of tl ance of the object from any position, and hence which model features are other words, the use of 3-D models allows for a more complete understanc scene that created the data. Therefore, we proceed with the assumption th want models that represent the 3-D structure of the objects.

The question of what to represent then arises. There are many differ representation schemes, but most of these are designed for object depicti their descriptors are chosen for convenient and completeness of picture rather than for suitability in matching, as required for recognition. The c is that we should only consider representations that represent visual features. As these representations are observed, they should be made pro the model.

Hence, a spline-based surface description is inappropriate for compute represents a complete surface by positions of knot points and the parame contained surface shape. Recognition requires one to deduct where the k lie and the surface parameters, so that they may be compared with tl Unfortunately, these properties are hard to deduce from image data. Alte to represent the surface by patches of nearly constant curvature is a repre appropriate for recognition, because it is possible to segment 3-D depth such patches, and then compare their estimated positions and curvat those of the model.

Any discussion of object representation must consider what the model used for. Here our interest is in object recognition, and the major tasks in this are:

Model invocation. Using some features of the data to index into the mc to select quickly a few likely models for more detailed analysis. becomes increasingly important as vision systems contain more object databases.

Model matching. To find data-feature evidence for model features (not ily of the same type nor visible). The evidence, such as strong image

needed initially to position the model, as well as to help to confirm that t recognition is correct.

Reference frame estimation. To relate data positions to model positions estimate the object's position. The object is usually assumed to be rigid, observed spatial relationships (e.g. an edge in a particular direction, two edg meeting at a given angle) strongly constrain the object's 3-D position.

Hypothesis verification. To ensure that the object hypothesis is valid (to wh ever degree desired) by finding extra supporting evidence. This might entail t use of the oriented model to predict the location of additional features.

Historical progression of 3-D representation

This section briefly surveys the development of 3-D recognition-oriented modelli systems. Table 1 summarizes the systems reviewed. One obvious feature is that time progressed, the complexity of the data and model primitives increased, large due to increased experience with data interpretation and the requirements recognition. For data, initially points were the preferred feature, then for a deca 2-D lines were used. Finally, in the 1980s, with the advent of 3-D data, 3-D lines a surfaces were exploited. Model representation also follows this trend, with or *ACRONYM* out of place. The use of volumetric primitives was advanced, but see now to be probably the wrong choice, with surfaces being favoured.

The rest of this section outlines the basics of these representation metho A good general reference for these methods is (Ballard & Brown, 1982).

Table 1. Historical summary of representation methods

Data	Model		
Primitive	Primitive	Project	Year
2-D points	3-D points	Roberts	1965
2-D lines	3-D lines		1970s
2-D lines	3-D volumes	*ACRONYM*	1981
3-D lines	3-D lines		1980s
3-D surfaces	3-D surfaces	Faugeras	1983
		IMAGINE	1986

Roberts' models

Roberts (1965) developed an early 3-D object recognition program based on g metrically specified 3-D models.

The three primitive objects he used were a cube, wedge and hexagonal pri which could be scaled to an arbitrary size. They were described using the

coordinates of their vertices (unscaled). Thus, the wedge shown in Fig. following vertex description:

Point	Position
A	(0,0,0)
B	(1,0,0)
C	(1,0,1)
D	(0,0,1)
E	(0,1,0)
F	(0,1,1)

This representation is useful to deduce the spatial position of the recogniz (a key process in object recognition). Though the details are not impor model matching established correspondences between 2-D data vertice line drawing of the scene derived from the image) and some of the 3 vertices. From the pairings, an estimate of the 3-D position of the object algebraically deduced, along with object scale. This deduction was only because the model vertices were geometrically specified.

Once a hypothetical object position is deduced, the geometrical mod used to predict the position of other image features (e.g. edges). These are verify the hypothesized identity and position and to decompose more cor objects made from several primitives.

A 2-D image model was also used. It listed the types of polygons see vertex, which are used as an index to select a model to explain the data (an process in object recognition). Hence the two quadrilateral and one tria faces surrounding vertex X in Fig. 2 allow the deduction that the object can be only the wedge. The data vertices can then be paired with corre model vertices and provide the input needed to deduce the geometrical pc described above.

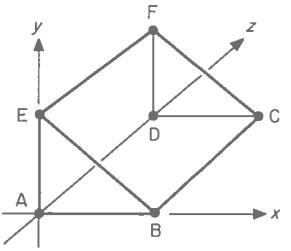

Fig. 1

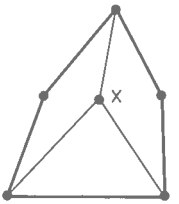

Fig. 2

The three key advantages of Roberts' models were:

1 Its primitives were simple and related closely to easily extractable and n image features.

2 They embodied 3-D feature information to allow the deduction of the ob scene position.

3 The model was specified in a viewpoint independent manner and allow computational transformation of the model's position and thus deduction of appearance.

On the other hand, this modelling approach suffers from two serious problem

1 The variety of objects open to modelling was limited because of the few simp primitives.
2 The main image feature, the vertex, is a highly ambiguous feature which caus combinational model matching and also unstable position estimation.

Wire frame models

The natural generalization of Roberts' point models are the wire frame models (e Falk (1972), Bolles *et al.* (1983), Ballard & Brown (1982), p. 291). These specify t position of surface boundaries in a 3-D coordinate system. The description of t wedge given in Fig. 1 would now be:

Description	Boundary
line: (0,0,0) ↔ (1,0,0)	A ↔ B
line: (1,0,0) ↔ (1,0,1)	B ↔ C
line: (1,0,1) ↔ (0,0,1)	C ↔ D
line: (0,0,1) ↔ (0,0,0)	D ↔ A
line: (0,0,0) ↔ (0,1,0)	A ↔ E
line: (0,1,0) ↔ (0,1,1)	E ↔ F
line: (0,1,1) ↔ (0,0,1)	F ↔ D
line: (0,1,0) ↔ (1,0,0)	E ↔ B
line: (0,1,1) ↔ (1,0,1)	F ↔ C

This model describes the same object as before, but also makes explicit m information: the type of the boundary that passes between the vertices. While above description used only straight lines, curves (e.g. portions of circular ar could also have been used. The difference between the two objects in Fig. 3 co not be expressed in a point-type model, but could be in a wire-frame model.

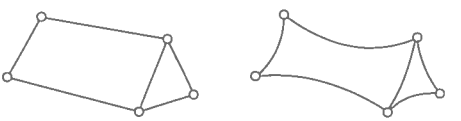

Fig. 3

These edge features can be simply maintained in a list associated with ea model, such as above. Another approach organizes the features according to topological relations between the features. For example, each of the polygo surfaces in the model could link to a list of the edges that bound the surface, and adjacent surfaces at each edge. These edges might also link to each other connecting vertices.

The advantage these representations give is that connectivity is made e: one can directly ascertain which other model features can be matched t feature, rather than having to deduce them as needed. A commonly u structure for edges is the "winged-edge" structure, which also records su vertex information (Baumgart, 1972).

Wire-frame models support the recognition operations introduced ab ings of image edges to model edges support the geometrical calculation u estimate object position. Model invocation can use groups of edges, thou effectively, because of the greater number of models allowed. Geometri formation and projection of the edges produces features to verify hypoth

Wire-frame models are suitable for use with either 2-D (i.e. line drawi (i.e. space-curve) data. In both cases the symbolic correspondences without difficulty, because only lines are matched. The difference li calculations needed to estimate the object's 3-D position, which is easier data.

The main advantages of the wire-frame models are:

1 Edge size and curvature make edges a richer, less ambiguous image fea reducing combinatorial matching.
2 Edge topology is easily represented in a graph-like structure.
3 Edge data for matching is more reliably extracted from 2-D and 3-D dat

Their main disadvantages are:

1 The objects represented are still largely polyhedral, because the re object edges usually coincide with surface orientation discontinuities. shows two objects indistinguishable in a wire-frame model.
2 In general, convex curved surfaces produce edges in images (e.g. when tl is tangential to the line of sight) that do not correspond to any model which move as the relative viewpoint moves.
3 There are still many edges to represent and hence to match.
4 The model makes it difficult to determine which lines are not visible fro point of view, particularly if no information is given about when edg surfaces. (Problems then occur with non-convex objects.)
5 This type of model represents all model features at the same level of de not able to distinguish more significant features or groups of features.

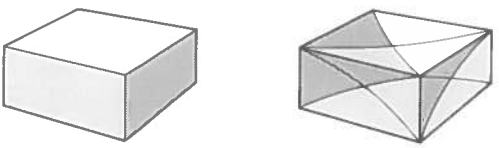

Fig. 4

ACRONYM

ACRONYM (Brooks, 1981) used a volume-based hierarchical represent tem. Volumes are important because they represent the dominant structu

of 3-D objects — their solidity. Hierarchical representations are also importa because they can be used to:

— recursively decompose structures into major subcomponents (e.g. a human h an arm, an arm has a hand, a hand has a finger, . . .) and
— support generalizations, by addition of discriminating details at lower leve (e.g. an L1011 aircraft has one engine on each wing while a B747 has two).

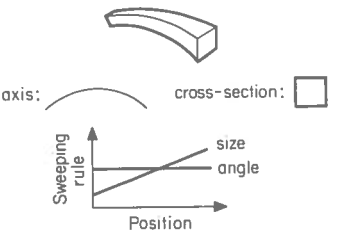

Fig. 5

ACRONYM's representation primitive is the generalized cylinder (Agin, 197 Binford, 1971; Marr, 1976; Hogg, 1983). This is a swept volume suitable represent many elongated structures, such as those created by man-made extrusi or turning processes. The volumes are defined by three aspects: the axis, the cro section and the sweeping rule. Figure 5 shows a curved, truncated pyramid and t definitions that produced it. Its axis is a circular arc, along which a square cro section is swept. The sweeping rule defines the scaling of the cross-section at ea point along the axis and also the angle at which the cross-section meets the ax Here the cross-section is always perpendicular to the axis and its size linea increases as a function of position along the curve. Almost any space curve, cro section and sweeping rule can be used, producing rather exotic volumes, b generally only simple definitions are used. For example, lines and circular arcs a the usual axes. Figure 6 shows some typical generalized cylinders.

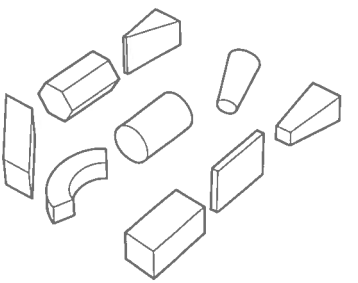

Fig. 6

Large assemblies hierarchically group previously defined assemblies or pri tives by specifying how the subcomponents are placed with respect to the asse bly. Using previously defined subcomponents also allows multiple reference to

same feature, such as when three identical robot fingers are needed as part
gripper. As each object has a locally defined 3-D coordinate system, pla
specified by giving the transformation that relates the main object and s
nent coordinate systems. Figure 7 shows part of a subcomponent hiera
simplified robot gripper. (The geometrical transformations linking the s
nents have been omitted.)

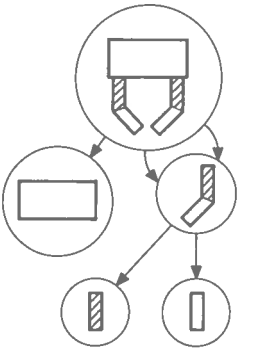

Fig. 7

In *ACRONYM*, translations are specified by the vector position. Rot
specified by giving the vector along which the subcomponent axis lie
rotation of the subcomponent about the axis. Figure 8 illustrates this. He
upper arm is placed with respect to the torso. Figure 9 shows a more co
model made out of several generalized cylinders.

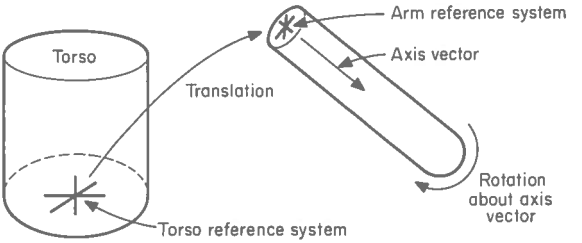

Fig. 8

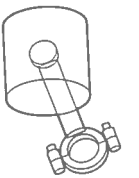

Fig. 9

In most modelling systems, translations are explicitly represented by
Rotations can be expressed using a variety of notations, such as Eul
rotation about the coordinate axes, yaw, pitch and roll, etc. The representa
in *ACRONYM* seems natural for the axis-based generalized cylinder prim

One of *ACRONYM*'s key features was the use of variables. With these
define models with some aspects incompletely specified, such as the curv
generalized cylinder's sweeping axis, or the angle at which a subcompo

placed. Variables are useful to represent both changing features, e.g. the joint ang of a robot arm, or undetermined properties, such as object scale. Anywhere wher numerical value is required, a variable, or an expression containing a variable c be used. As an example, we can define bricks with cross-section "rectan (WIDTH,HEIGHT)" and axis "straight(LENGTH)".

Constraints can then be defined over these variables. A variable or an express containing variables has its range limited by one or more algebraic inequalities. our brick model, two possible constraints on the shape variables might be:

WIDTH > 2 * HEIGHT
LENGTH > 2 * WIDTH

Constraints allow two important model properties:

— limited ranges of variation, and
— introduction of constraint-based object subclasses.

The first is useful to express more detailed relationships than those strictly c tained in the shape definitions, e.g. when angles are not allowed for some rc joint.

In general, the use of variables in the model defines a class of objects, all meeting model definition. Constraints on the variables are then useful to define a specia ation hierarchy, where a subclass of the models has some distinguishing prop (represented by a restricted variable range). Associated with each structural mc is a set of constraints on the variables used in the model. A specialization of model is given by addition of new (i.e. stronger) constraints. As the objects re sented now meet all the specialized constraints, as well as the more general o they are specializations of the original object. An example of this occurs when define a new class called "long-bricks" by adding the constraint:

LENGTH / WIDTH > 4

Model matching in *ACRONYM* was complicated, and is summarized her show how some model features are important. The data primitives were ribbons — elongated regions defined by parallel image edges. These were foun grouping processes in edge-detected 2-D intensity images. Ribbons were matc with simple generalized cylinder primitives, because a ribbon axis correspond the projection of the generalized cylinder axis. The ribbon width relates to cylinder cross-section width. Matching larger assemblies used data relating p tion, orientation and sizes of axes.

The relationships between the camera, the scene, the image and the geomet models provide several geometrical constraints on the position of the object any embedded variables. Other constraints were explicitly given in the mc When enough constraints were obtained, all variables could be fully constrai by using a complex constraint simplification method. To establish precise subc identity involved verification of the additional constraints associated with subclass. In summary, identity was established provided sufficient structural tionships and geometrical constraints were obtained from the data.

ACRONYM demonstrated several significant advances for computer vi
3-D model representation, in particular, it showed that:

1 The generalized cylinder primitive is useful to describe many objects characterize a wide range of solids using only a few parameters.
2 The subcomponent and generalization hierarchies express important m tionships not represented previously.
3 The variable and constraint mechanism allows both bounded object vari incremental object description.

ACRONYM's representational weaknesses include:

1 The need for data primitives for other than elongated shapes. While mar can be represented simply, the generalized cylinder primitive has nc extension to represent exceptions.
2 The fact that volumetric model primitives have no directly correspon primitives, thus this requires matching to be either weakly based on fra evidence, or computationally expensive.
3 The difficulty of deducing what the visible features of a model and thei ance will be from a given viewpoint, except through the equivalent of image generation. This makes matching computationally expensive.

Faugeras

Faugeras and colleagues (Faugeras & Herbert, 1983) investigated object re using surface patches as their model primitive. Surfaces have an obviou advantage, because these are directly visible data features and are also la points and lines, and are thus more reliable and less ambiguous.

Surface patches are also suitable for matching using 3-D data that cont and surface orientation information as well as XY and intensity informati

Work on new sensors and low-level data processing techniques leads to acquisition of 3-D data. Typical sources of such data come from:

— *stereo* where the 3-D location of some feature is deduced by triangulat several independent views,
— *motion* where either observer or object motion give cues to surface feature depth,
— *shading* where patterns in the observed intensity data indicate, or c local surface orientation,
— *direct range sensing* where laser or sonar range finders explicitly mea depth at selected points (to give the 3-D position of the point), or
— *structured light* where the deformation of a known stimulus (e.g. ligh indicates the surface shape, distance and orientation.

Two key advantages of direct 3-D data are:

1 the usual loss of depth information in forming a 2-D projection from a 3-D avoided, and

2 it is now possible to more unambiguously label image features (e.g. a reflectan versus a depth discontinuity edge — which appear identical in a typical 2 intensity image).

Faugeras used range data from laser triangulation both to create models a analyse scenes. The data were segmented into planar patches using several tec niques (Hough transform and local plane fitting with region growing). Each pat was characterized by its approximate patch position and surface orientation. A f 3-D model of the object was defined when enough patches were seen from differ points of view. Thus, the model description for the cube in Fig. 10 might be:

Patch	Position	Orientation
P1	(0.55,0.47,0.03)	(0.02,−0.01,−0.99)
P2	(1.02,0.49,0.53)	(0.99,0.02,−0.02)
P3	(0.48,0.99,0.51)	(0.02,1.01,−0.03)
P4		
P5	...	...
P6		

Object recognition involves matching model to data patches and estimation of t 3-D transformation that relates the two. Since the models and scenes were defin using the same type of raw data and identical segmentation processes were appli the model and data surface correspondences should be close. Further, as both t data and model surfaces were 3-D, estimation of the transformation between t two was easy (a least squares estimation of the transformation).

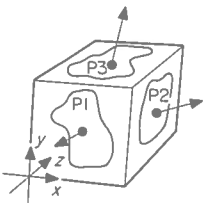

Fig. 10

The advantages of these models are:

1 Surface patches directly correspond to visible image features, simplifying matching process.
2 Complicated objects can be modelled empirically. Figure 11 shows an exampl a model for an irregular cast part.

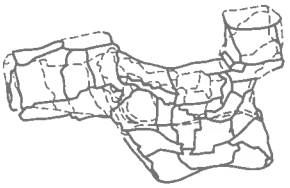

Fig. 11

(Figure reproduced by permission from Faugeras and Hubert 1983.)

Some disadvantages of these models are that:

1 No structure hierarchy was used which makes all object features equiv
2 Only planar features can be modelled reliably. This means that mode associated with curved surfaces may not always be detected or describe ently, causing matching and geometrical transformation estimation Further, this means that some object portions may not be modelled.
3 Only rigid objects can be represented.

IMAGINE

IMAGINE (Fisher, 1986) used a surface-based object modeller. The obje sented were compact, flexibly connected solids with defined surface bou

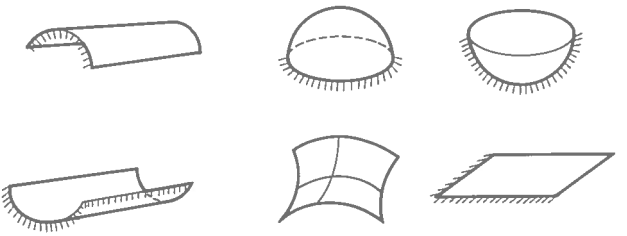

Fig. 12

The surface patch is the model primitive and is described by shape (i.e. curvatures with zero, one or two curvature axes) and extent (i.e. boundar 12 shows the six curvature classes represented in the models. Thes primitives were chosen because they correspond to the symbolic classif data surfaces based on principal curvature.

The boundaries are specified using a polycurve notation, which c selected points on the surface boundary and intervening boundary segm types of segments used are lines and circular arcs (which are projected surface to create the actual segment boundary). As the surface shape is th feature of the patch, minor variations in the patch extent caused b boundary shape are acceptable. A simplified description for the convex c surface patch (shown in Fig. 13) follows:

SHAPE = convex cylinder (radius = 10,axis = (1,0,0))
BOUNDARY = (0,0,−10) ← ARC(radius = 10) →
(0,7,−7) ← LINE →
(20,7,−7) ← ARC (radius = 10) →
(20,0,−10) ← LINE →

Objects are recursively constructed from surfaces or sub-objects using c reference frame transformations. Each structure has its own local refere

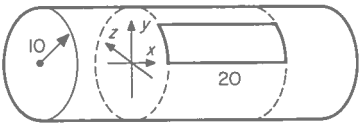

Fig. 13

and larger structures are constructed by placing the subcomponents in the referen frame of the aggregate. Sub-objects can be connected flexibly by using variables the attachment relationship. The geometrical relationship between structures useful to make model to data assignments and to provide the adjacency and relati placement information needed for hypothesis verification.

Figure 14 shows an image of a robot assembly with the surfaces shaded accordi to surface orientation.

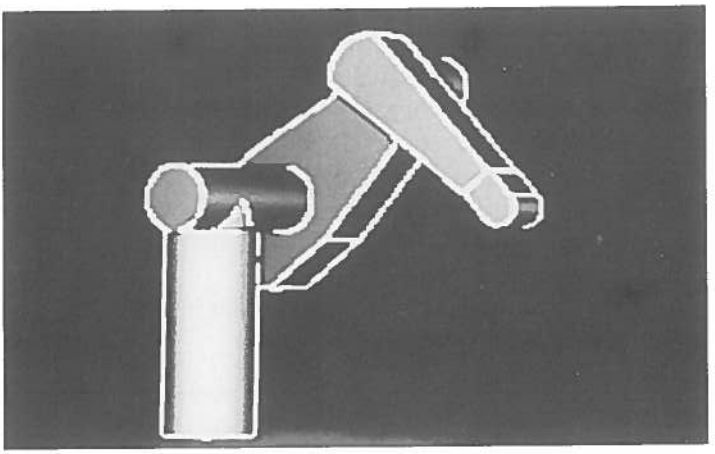

Fig. 14

IMAGINE also represents three types of information useful for selection o model for direct matching:

1 nominal values for properties of boundaries, surfaces and assemblies, such area, curvature and elongation,
2 relationships between different model entities, such as structural (e.g. subcom nent) or generic (e.g. subtype),
3 groups of object features seen from typical significant viewpoints.

Much of this information could have been derived from the geometric model, l some derivations are time-consuming and can be done in advance, so it v decided to represent the results explicitly in the model.

Model matching in *IMAGINE* paired surfaces segmented from surface image d (e.g. a depth image) directly with model surface patches, which explains why surface patches were so defined. Because both the models and data had a character, it is easy to estimate the geometrical relationship between them. Furth

the surface representation makes it easy to predict what model features s visible from a given perspective and where they should be found in t Model invocation used the image feature properties and relationships with the model properties and relationships to select a few candidate mod larger model base for more expensive (computationally) processing.

IMAGINE combines the structural aspects of *ACRONYM* with the surf representation of Faugeras. Because it extends the range of surface shape be represented, objects can be more accurately described. The surfac promote better and easier image understanding, by representing directl able, salient object features.

IMAGINE's variable mechanism is not as powerful as *ACRONYM*'s, and represent other visible features, such as prominent lines.

Observations

One feature common to all these representational systems is the use geometric description of key object features. They differ in the type of fe point, line, surface or volume), and the degree of structure of the mc monolithic, "winged-edge" or hierarchical), but all use geometrically de tures placed with respect to a reference coordinate system. This allows reasoning approach to image understanding, where the program can d appearance and position of features, based on a 3-D understanding position.

Moreover, experience with the different representational techniques several results, given below:

1 Model features should correspond directly with observed, segmented tures. If model surfaces are used, then it is hard to decide what feature a should be matched to (e.g. an orientation discontinuity boundary, a occluding boundary, etc.). Data volumes are better, because a predict surface can be derived from the volume description. However, a data su be directly matched.

2 If we want to represent the visibly salient features, a variety of repres should be used. For some objects the features are strong surface or r edges, for others, the shape of a few key surfaces, or the shape of a swe A similar point can be made about the scale of features. Because an usually observed from several different distances, the salience of feat vary. Hence, it is desirable to provide alternative representations for ranges.

3 Model features should be chosen for visible salience rather than acc metrical object represention. Hence, the models may be more sugges accurate at times. Thus a pencil, while technically six blended planes, more appropriately represented as a cylinder. This also implies that a m be incomplete — there may be portions of the visible surface without tions, e.g. where two surface patches meet without producing a strong

boundary. A model might consist of a few striking surfaces and edges only. Sin the point is recognition rather than image production, this might be sufficient.

4 Structured models are easier to match. They support attachment variation mc easily. Further, by partition of the model features they reduce the features need for matching at any level, thus reducing computational requirements. To defi subcomponent structure independently allows re-use of the definitions f repeated components.

5 Useful models embody both object-centred and viewer-centred informatic (Object-centred means the information is represented in relation to the position the object, rather than to the viewer.) The geometrical models, described abo are mainly used to represent object-centred information. While viewer-centr information (i.e. representations of features as seen by the viewer) can usually derived from the geometrical models, it is often useful to make the informati explicit in the model. This allows the information to be used as indices duri model invocation and also avoids having to compute it during matching (whi can be costly).

To summarize, for 3-D object recognition tasks, 3-D models offer the only r possibility for high competence general object recognition and spatial locatic While the details of the model representations are still uncertain, the most use representations record visibly salient features that suggest and confirm identi rather than define object shape.

Future representation systems

The key deficiency in the models described above was that they could only descri man-made objects, and only those with relatively regular shapes. While represe tation for recognition of objects with wildly varying shapes (e.g. trees) is st distant, representation of objects whose basic shape remains constant, but whc parameters of shape vary slightly, can be expected soon (e.g. bend in a flexil member, elongation of a face, placement of eyes, etc.).

Another likely development is the incorporation of surface shape texture a feature added to the general surface shape and distribution. Multiple alternati representations for scale-dependent data also seems likely. Finally, mixed rep sentation systems are starting to be used.

Of course any new developments in object representation will have to proceed parallel with developments in data acquisition and model matching, both evaluate their effectiveness and enhance their usefulness.

Further reading

Marr (1982) proposed five criteria for object representation, and any serious meth should be judged using them:

— *accessibility*, needed information in a model should be directly available rat than derived through heavy computation,

— *scope*, a wide range of objects should be represented,
— *uniqueness*, an object should have a unique representation,
— *stability*, small variations in an object should not cause large variatic model, and
— *sensitivity*, detailed features should be represented as needed.

Most of this paper ignores property-based representations, because of limitations; however, they can be useful in restricted domains. They defir by properties or constraints (without recourse to an explicit geometrical m satisfaction of which should lead to unique identification. Duda and Ha used properties such as colour and height to analyse scenes. Shirai (19 rough sizes, colours and edge shapes to characterize desk-top objects. Ad used viewer-centred property models to interpret 2-D Peanuts carto scenes. The model primitives are regions with summary properties (e.g. ar larger figures meet adjacency constraints. Falk (1972) used face shape, edg and 2-D edge angles to identify polyhedra. Constraints may also include ships that have to be held with other structures (e.g. Barrow and Tenenbau

Property representations are usually viewer-centred. Minsky (1975) pr frame representation to record features visible from typical distinct vie Various researchers (Hanson & Riseman, 1978; Nagao *et al.*, 1979; Ohta *et* have augmented property representations with weak image shape (e.g. square) and image relations (e.g. above, near).

A more structured property representation is the graph, where object fe nodes, and relationships between the features become arcs. Barrow anc stone (1971) used a viewer-centred graph to represent visible object reg their inter-relationships, e.g. adjacency and relative size. Graph repres have the advantage of adding some structure to the object properties, and common representation method for many problems. One problem is that details tend to be represented at the same level, so the graphs can bec without benefit.

Several other less important geometrical representations are described

Surfaces can be represented by bi-cubic spline patches (York *et al.*, 198 & Brown, 1982, p. 269), where cubic polynomials interpolate the surface fixed points, to give both positional and derivative continuity at the second popular approach uses polygonal planar surface patches (e.g. Boi Faugeras, 1981), with splitting of the patches until arbitrary accuracy is These represent surfaces well, but give no conceptual structure to the su

Space-filling models (e.g. Ballard & Brown, 1982 p. 280) represent c denoting the portions of space in which the object is located. Construc geometry starts from geometrical primitives, e.g. cubes, cylinders or h (Requicha & Voelcker, 1977; Cameron, 1984) and then forms more compl by merging difference and intersection operations. The primitives and r operations are simple, but, unfortunately, this approach makes little in explicit.

Another volumetric representation is that proposed by Shapiro and colleagu (1980). This combines a rough geometrical model based on sticks (1-D), plates (2- and blobs (3-D) with a relational characterization of their structural relationship

Acknowledgments

We would like to thank J. Aylett and M. Orr for comments on this paper.

Figures 6 and 9 are reproduced by permission from North-Holland from Broc (1981). Figure 11 is reprinted with permission from O. D. Faugeras, and M. Hebe A 3-D recognition and positioning algorithm using geometrical matching betwe primitive surfaces. *Proceedings of the Eighth International Joint Conference Artificial Intelligence*, Karlsruhe, West Germany, p. 1002, (c) *1983 Internatio Joint Conferences on Artificial Intelligence*. All proceedings volumes for *IJC* conferences can be obtained from Morgan Kaufman Publishers, Inc. 95 First Suite 120, Los Altos, CA 94022, USA.

References

Adler, M. (1975) *Understanding Peanuts' Cartoons*, Department of Artificial Intelligence Resea Report, **13**, University of Edinburgh.

Agin, G. J. (1972) Representation and description of curved objects, *PhD thesis AIM-1973, Stanf AI Lab.*

Ballard, D. H. & Brown, C. M. (1982) *Computer Vision*, Prentice-Hall, New Jersey.

Barrow, H. G. & Popplestone, R. J. (1971) Relational descriptions. In: *Picture Processing.* (B. Meltzer & D. Michie), *Machine Intelligence* **6**, 377–396.

Barrow, H. G. & Tenebaum, J. M. (1976) MSYS: a system for reasoning about scenes. *Stanf Research Institute Technical Note 121.*

Baumgart, B. G. (1972) Winged edge polyhedron representations, *STAN-CS-320, AIM-179, St ford AI Lab.*

Binford, T. O. (1971) Visual perception by computer, *IEEE Conference on Systems and Contro*

Boissonnat, J. D. & Faugeras, O. D. (1981) Triangulation of 3-D objects. *Proceedings of the International Joint Conference on Artificial Intelligence*, 658–660.

Bolles, R. C., Horaud, P. & Hannah, M. J. (1983) 3-DPO: a three-dimensional part orientation syst *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, 1116–112(

Brooks, R. A. (1981) Symbolic reasoning among 3-D models and 2-D images. *Artificial Int* gence, **17**, 285–348.

Cameron, S. A. (1984) Modelling solids in motion. *PhD Thesis, Department of Artificial Int* gence, *University of Edinburgh.*

Duda, R. & Hart, P. (1970) Experiments in scene analysis. *Stanford Research Institute repor group Technical note 20.* Project 8259.

Falk, G. (1972) Interpretation of imperfect line data as a three-dimensional scene. *Artifi Intelligence*, **3**, 101–144.

Faugeras, O. D. & Hebert, M. (1983) A 3-D recognition and positioning algorithm using geome matching between primitive surfaces. *Proceedings of the International Joint Conference Artificial Intelligence*, 996–1002.

Fisher, R. B. (1986) From surfaces to objects: recognizing objects using surface information object models, *PhD Thesis, University of Edinburgh.*

Hanson, A. & Riseman, E. (1978) *VISIONS*: a computer system for interpreting scenes. In: *Comp Vision Systems*, (eds. A. Hanson & E. Riseman) pp. 303–333. Academic Press, New York.

Hogg, D. (1983) Model-based vision: a program to see a walking person. *Image Computing*, **1**, 5–20.

Marr, D. (1976) Representation and recognition of the spatial organization of three-c objects. *Massachusetts Institute of Technology AI memo 377.*

Marr, D. (1982) *Vision*, W.H. Freeman & Co., San Francisco.

Minsky, M. (1975) A framework for representing knowledge. In: *The Psychology o Vision*, (ed. P. Winston) pp. 211–277. McGraw-Hill, New York.

Nagao, M., Matsuyama, T. & Mori, H. (1979) Structural analysis of complex aerial p *Proceedings of the 6th International Joint Conference on Artificial Intelligence*, 61

Ohta, Y., Kanade, T. & Sakai, T. (1979) A production system for region analysis. *Procee 6th International Joint Conference on Artificial Intelligence*. pp. 684–686.

Requicha, A. A. G. & Voelcker, H. B. (1977) Constructive solid geometry. *University o Production Automation Project memo TM-25.*

Roberts, L. G. (1965) Machine perception of three-dimensional solids. In: *Optical c Optical Information Processing*, (ed. J. T. Tippett) Ch. 9, p. 159–197. MIT Press, MA.

Shapiro, L., Moriarty, J., Mulgaonkar, P. & Haralick, R. (1980) Sticks, plates and blc dimensional object representation for scene analysis, *NCAI-80.*

Shirai, Y. (1978) Recognition of real-world objects using edge cue. In: *Computer Visi* (eds. A. Hanson & E. Riseman) pp. 353–362. Academic Press, New York.

York, B. W., Hanson, A. R. & Riseman, E. M. (1981) 3-D object representation and matcl splines and surface patches. *Proceedings of the 7th International Joint Conference Intelligence*, 648–651.