Pattern Recognition Letters 12 (1991) 565–573
North-Holland

September 1991

Generalizing the generalized Hough transform

Haim J. Wolfson*

Computer Science Department, Sackler Faculty of Exact Sciences, Tel Aviv University, Israel
Robotics Research Laboratory, Courant Inst. of Mathematical Sciences, New York University, 715 Broadway, 12th fl., New York, NY 10003, USA

Received 10 December 1990
Revised 3 June 1991

Abstract

Wolfson, H.J., Generalizing the generalized Hough transform, Pattern Recognition Letters 12 (1991) 565–573.

We present a new method for model based recognition of articulated objects in cluttered scenes. The objects consist of rigid parts connected by rotary or prismatic joints. Our method is based on an extension of the Generalized Hough Transform paradigm. It is applicable to various viewing transformations. Unlike previous methods no significant degradation is expected in performance for recognition of articulated objects compared with the recognition of rigid objects containing a similar amount of information.

Keywords. Articulated objects, object recognition, generalized Hough transform.

1. Introduction

One of the major problems in machine vision is object recognition. This problem is especially challenging when the objects may partially occlude each other. In *model-based* vision one may assume that the objects in the scene belong to a large library of models which is known in advance. This is the situation in various industrial applications, where a robot is faced with recognition and location of objects in a scene comprised of a subset of the factory tools and products. Most of the work which was done up to date in object recognition concentrated on recognition of rigid objects (see (Besl and Jain, 1985; Chin and Dyer, 1986) for comprehensive reviews and (Lowe, 1985; Thompson and Mundy, 1987; Linnainmaa et al., 1988; Huttenlocher and Ullman, 1988; Lamdan and Wolfson, 1988) for some recent work). However, many of the standard tools, as well as robots possess internal degrees of freedom. Thus in order to have a practical object recognition system, one has to extend it to handle articulated rigid objects, namely, objects having rigid parts which are connected either by rotary (revolute) or prismatic (sliding) joints. This class of objects is of special importance since it includes most of the industrial robots and man-made factory tools.

Up to date very little work was done on articulated object recognition. The first attempt to tackle the problem was probably by Brooks (1981) in his ACRONYM system. Grimson (1987) extend-

* This research was supported by grant No. 89-00481/1 from the US-Israel Binational Science Foundation (BSF), Jerusalem, Israel.

0167-8655/91/$03.50

ed the interpretation tree approach (Grimson and Lozano-Pérez, 1987) to deal with 2-D objects with rotating subparts. Goldberg and Lowe (1987) extended Lowe's SCERPO system to deal with 3-D articulated objects, such as staplers etc.

Typically, there are two major approaches to extend existing rigid object recognition systems so that they could tackle the more complicated articulated object recognition problem. One approach is to incorporate the additional degrees of freedom into the recognition machinery, and attempt to discover the objects modulo all these new degrees of freedom. This usually results in a significant increase in the complexity of recognition and degradation of performance compared with the recognition of rigid objects having a similar amount of visual information. Another approach is to represent an articulated object as a composition of its rigid parts, try to recognize each part individually, and then check some global consistency among the candidate solutions for the separate parts. A variation of this approach (Grimson, 1987) is to attempt to recognize just one part (the more informative one), derive some constraints from this hypothesis, and then try to recognize the other parts under the simplifying assumptions of the derived constraints. Here one is also dependent on the order parts are being recognized. This second type of approach can give reasonable results if the individual parts have enough unoccluded information to be recognized as rigid objects. However, if the unoccluded portions of the parts are not informative enough, the initial hypotheses of the algorithms will not be reliable to proceed to the global consistency check. One can say that this second approach does not exploit forcefully enough the fact that the different parts do belong to the same object.

It would be desirable to design an articulated object recognition approach which incorporates both the more simple rigid subpart recognition techniques and the global consistency checks as an integral part of the recognition process. In this short paper we present such an approach, which is an extension of the Generalized Hough Transform paradigm for rigid object recognition (Ballard, 1981). By introducing a new indexing scheme which exploits reference frames located at the joints we introduce the global consistency check already on the level of the initial local voting of the Hough scheme. In such a way the performance of our algorithm for articulated objects should be quite similar to the performance of the Generalized Hough Transform (GHT) for rigid object recognition, given that one is dealing with objects having the same amount of relevant unoccluded visual feature information. Our approach is general in the sense that it applies to various viewing transformation assumptions. It is applicable for articulated objects with either rotational or sliding joints. Our basic idea also extends to objects with multiple joints.

We have done a preliminary implementation of our algorithm for recognition of rotated, translated and scaled (nearly) 2-D objects with rotary joints. We have tested the performance of the algorithm on a set of standard tools, such as pliers, hammers, scissors etc. The results have been quite reassuring and are reported in (Beinglass and Wolfson, 1991). Since this paper is intended just to present the basic idea, we do not present an analysis of the experimental results. This will be done in a follow up paper.

This paper is organized as follows. In Section 2 we give a short overview of the existing Generalized Hough Transform based rigid object recognition techniques and present a general framework for these techniques under various viewing transformation assumptions. Section 3 presents a general framework for articulated object recognition by extending the GHT approach. It analyses the cases of rotary and sliding joints, and also presents the multiple joint case. Finally, in Section 4 we summarize and discuss our method and outline some future research and implementation projects.

2. The generalized Hough transform for recognition of rigid objects

In this section we briefly survey the existing GHT based rigid object recognition methods, and present a general framework of recognition by a GHT technique using R-tables. This serves as an introduction to our suggested articulated object

recognition method which is presented in the subsequent sections.

The Hough Transform was originally introduced to detect the parameters of line segments in binary image data. It was later extended to detection of other parametrized curves, and even to the detection of arbitrary shapes. For a comprehensive survey of Hough Transform based techniques see (Illingworth and Kittler, 1988).

Ballard (1981) suggested a rigid shape recognition method which incorporates a clever indexing scheme, based on a kind of an associative memory which he called the R-table. His method was originally designed for 2-D objects which have undergone a translation in the plane. It was based on an indexing procedure, where edges which are distinguished by their orientation, vote for the location of a predefined 'object center'. His technique, being a local one, had the advantage of handling partial occlusion. The suggested extension of the method to accommodate for both rotation and scale change was by exhaustive enumeration on all possible (quantized values of) rotations and scaling factors (see (Ballard, 1981), or (Ballard and Brown, 1982, pp. 128–131)), although Ballard mentioned the possibility of using pairs of edges to reduce this huge enumeration space.

A related rigid object recognition method which has also been termed *Generalized Hough* or *pose clustering* (see (Stockman, 1987; Silberberg et al., 1984; Thompson and Mundy, 1987; Linnainmaa et al., 1988)) is usually applied for various viewing transformation assumptions. It does not use an R-table indexing scheme. In this approach, recognition of an object in a scene is achieved by finding a transformation between a model-object and the scene, which maps a large enough number of the model *interest features* into scene *interest features*. The transformation is discovered by voting for its parameters which are consistent with hypothetical pairings of object and image compound *interest features*. Ideally, the compound features are such that they uniquely define the parameters of a candidate transformation. Since this method does not use an R-table, each model-object has to be matched separately against the scene.

We present now a unified approach to rigid object recognition using the Generalized Hough Transform. This method exploits the efficient R-table indexing method, nevertheless it can be still applied to various 2-D to 2-D or 3-D to 3-D transformations since it uses compound *interest features*, which have enough information to define a transformation invariant reference frame, and a transformation invariant quantity which we call *shape signature*. In the original GHT application (Ballard, 1981) an edge with its direction are used to define a translation invariant reference frame, which origin is the location of the edge and x-axis direction is the orientation of the edge. The *shape signature* is the orientation of the edge. Such compound features can be easily defined for any other 2-D object transformation. For example, in the important case of the similarity transformation (rotation, translation and scale), two ordered points (or, equivalently, a line segment) uniquely define a transformation invariant reference frame. The first point is the origin, so it defines the translation, the second defines the unit vector in the x-direction, and thus also the scale, and rotation. However, two ordered points do not define a similarity invariant *shape signature*, thus one needs a third point (or at least a direction). Angles of such a triangle, or ratios of edge lengths supply several possible signatures. For an affine transformation one needs a compound feature of four points in general position. There, the first three points can define an affine frame and the coordinates of the fourth point in this frame define an affine invariant *shape signature* (see (Lamdan et al., 1988)). An analogous discussion applies also to 3-D from 3-D transformations.

The GHT algorithm for any 2-D (or 3-D from 3-D) transformation can now be described as follows.

Preprocessing – R-table formation

The following preprocessing is applied to each model-object.

(a) Extract the *interest features* of the object.

(b) Pick an arbitrary pixel as the object's 'reference center', and define a 2-D coordinate frame centered at this pixel.

(c) For each compound feature (set of basic features) which defines a transformation invariant *shape signature*, compute this signature and the

compound feature based coordinate frame (the compound feature has enough information to define such a frame uniquely using some convention). Use the signature as an address to the R-table, and record in the appropriate entry two pieces of information, namely, the object which gave rise to this signature, and the coordinate transformation between the compound feature based coordinate frame and the object's 'reference center' based coordinate frame.

The preprocessing stage is done only once for each object in the model base, and is fully independent on the scenes to be recognized.

Recognition

Given a composite scene of partially occluding objects the following recognition procedure is applied.

(a) Extract the *interest features* of the scene.

(b) For each compound feature which defines a transformation invariant *shape signature*, compute this signature and use it as an address to the appropriate R-table entry. For each record in that entry compute a candidate 'object center' and reference frame by applying the prerecorded transformation to the compound feature based coordinate frame. Cast a vote for the identity of the object together with the associated object centered reference frame (the same object with different frames are treated as separate entities).

(c) Check the accumulator of votes for high scoring pairs of (*object, reference frame*).

(d) Verify the high scoring candidates (see, for example, (Heller and Stenstrom, 1989)).

This general description of the GHT method using an R-table generalizes Ballard's method for translation invariant recognition, applies to all the 2-D transformations mentioned above and to 3-D from 3-D recognition. While having the advantage of the R-table based indexing, it does not require exhaustive enumeration of some of the parameters for more complex transformations. However, the main advantage from our point of view is its natural generalization to the recognition of articulated objects, as is described in the sequel.

3. Articulated object recognition

In this section we discuss recognition of articulated objects under various viewing transformations. Articulated objects consist of rigid parts which are connected by joints which allow relative motion of neighboring parts. In robotic applications (Craig, 1986) one is usually concerned with rotary (revolute) or prismatic (sliding) joints. Thus articulated object recognition systems have to deal not only with *external* transformations of the whole object, such as rotations, translations, scaling, but also with *internal* transformations of the neighboring parts relative to each other.

Rigid object recognition systems have been designed to deal with *external* object transformations by considering object features which are invariant to these transformations (see Section 2). The main property exploited is that for a given object, all its features have undergone the same transformation. This assumption does not hold for articulated objects because of the *internal* degrees of freedom. In Section 1 we have mentioned that one of the approaches to articulated object recognition has been to try and recognize its individual parts, and then check whether the recognized parts are consistent at the joints. Such an approach does not advance us much beyond the rigid object recognition systems, since individual parts of a given object might not contain enough information to be reliably recognized. This might happen due to considerable occlusion which is especially acute in the articulated object case, since in such objects even different parts of the same object tend to occlude each other. Hence, we suggest an approach which incorporates the joint consistency information in a natural way, and not as a post-processing to the rigid part recognition step. We describe our approach both in the cases of single rotary or prismatic joints, and then we outline a possible method to address the recognition problem of articulated objects with multiple joints.

3.1. Recognition of objects with rotary joints

For the sake of clarity, we first describe our algorithm for the case of objects with two parts

each. The parts are linked by a rotary joint (e.g. pliers, scissors, pincers are objects belonging to this class). We consider the cases (transformations) which occur either in the recognition of 2-D objects from 2-D images, or in the recognition of 3-D objects from 3-D data (e.g. range, tactile). The key point is that, since the joint belongs to both parts, one can gather simultaneous votes from all the parts of an object, if he places his reference frame at the joint location.

Preprocessing – R-table formation

The following preprocessing is applied to each model-object.

(a) Extract the *interest features* of the object.

(b) Pick *the rotary joint position as the object's 'reference center'*, and define a 2-D coordinate frame centered at this joint.

(c) For each compound feature (set of basic features) which defines a transformation invariant *shape signature*, compute this signature and the compound feature based coordinate frame. Use the signature as an address to the R-table, and record in the appropriate entry two pieces of information, namely, the object which gave rise to this signature, and the coordinate transformation between the compound feature based coordinate frame and the object's 'joint centered' coordinate frame.

Recognition

Given a composite scene of partially occluding objects the following recognition procedure is applied.

(a) Extract the *interest features* of the scene.

(b) For each compound feature which defines a transformation invariant *shape signature*, compute this signature and use it as an address to the appropriate R-table entry. For each record in that entry compute a candidate 'object joint location' and reference frame by applying the prerecorded transformation to the compound feature based coordinate frame. Cast a vote for the identity of the object together with the associated location and orientation of the *joint centered* reference frame.

(c) Check the accumulator of votes for high scoring pairs of (*object, joint location*).

(d) Check the high scoring candidates, whether there are only two high scoring orientations at the suggested joint locations.

(e) Verify the high scoring candidates by matching all of their edges. This verification can be done by applying the two candidate transformations to the appropriate model parts, and checking whether a relatively high percentage of the projected model edges are in close vicinity to image edges having a similar direction (for a thorough discussion of such a verification scheme see (Heller and Stenstrom, 1989)).

In this algorithm we have naturally exploited the fact that both parts incorporate the same joint by locating the object's reference frame at the joint. In such a way both parts contribute votes to a reference frame at the same location, although at two different orientations. This two orientation clustering is exploited for an additional consistency check at step (d) of the recognition algorithm. By picking up votes from all of its parts, an object, which is considerably occluded, can still score high, although its individual parts could receive an insignificant score each. Moreover, even if different parts of the same object do appear in the scene, but are not linked by a joint at a correct location, their votes will not combine together.

In practice, one has more consistency constraints than just the joint location. Some of them are due to the viewing transformation, and others are due to the consistency of evidence for the same part of a model. For example, in the case the scaling of rotation, translation, and scale, the scaling factors of all the parts should be nearly identical. This constraint can be checked before the final verification step (e) of the algorithm. See (Beinglass and Wolfson, 1991) for the consistency constraints that we have applied in similarity invariant recognition.

Note that the method presented here is almost as powerful as the GHT method for rigid object recognition. There is almost no expected degradation in performance for the more complicated articulated object case. If one applies the GHT (as in Section 2) for the same object with a fixed joint position (no internal degrees of freedom), the only advantage he has is that he should gather votes not

only for a unique joint location, but also for a unique (rotary) joint based reference frame orientation (instead of the two expected in our case). It seems to us that the expected degradation in performance for articulated objects is small compared with the gain of the extension of the algorithm for this important class of objects. It is also worthwhile to mention that occlusion of the reference joints in the scene has no effect on the performance of the method.

The algorithm described above has been implemented for objects with single rotary joints which have undergone a similarity transformation (rotation, translation and scaling). Due to the succinct nature of this letter, we do not discuss here these preliminary results. They can be found in (Beinglass and Wolfson, 1991).

Our discussion was so far restricted to rotary joints. In the next subsection we describe the method for *prismatic (sliding) joints*.

3.2. Recognition of objects with prismatic joints

In this subsection we discuss our method for *prismatic (sliding) joints*. Conceptually, the basic approach is analogous to the one described earlier. This time, however, the information that is shared by both parts is not the joint location, but its line of slide.

Specifically, let us describe the algorithm, for objects with two parts, which are linked by a prismatic joint (see Figure 1).

For the sake of clarity, let us discuss the case of 2-D objects, which have undergone a similarity transformation (rotation, translation and scaling) in the image. The algorithm follows:

Preprocessing – R-table formation

The following preprocessing is applied to each model-object.

(a) Extract the *interest features* of the object.

(b) Pick a reference frame for the object in such a way, that the *x-axis* of this frame is aligned with the line of slide of the prismatic joint. The origin of the reference frame is chosen in an arbitrary way.

(c) For each compound feature (set of basic features) which defines a transformation invariant *shape signature*, compute this signature and the compound feature based coordinate frame. Use the signature as an address to the R-table, and record in the appropriate entry two pieces of information, namely, the object which gave rise to this signature, and the transformation between the compound feature based coordinate frame and the object's 'line of slide oriented' coordinate frame.

Recognition

Given a composite scene of partially occluding objects the following recognition procedure is applied.

(a) Extract the *interest features* of the scene.

(b) For each compound feature which defines a transformation invariant *shape signature*, compute this signature and use it as an address to the appropriate R-table entry. For each record in that en-

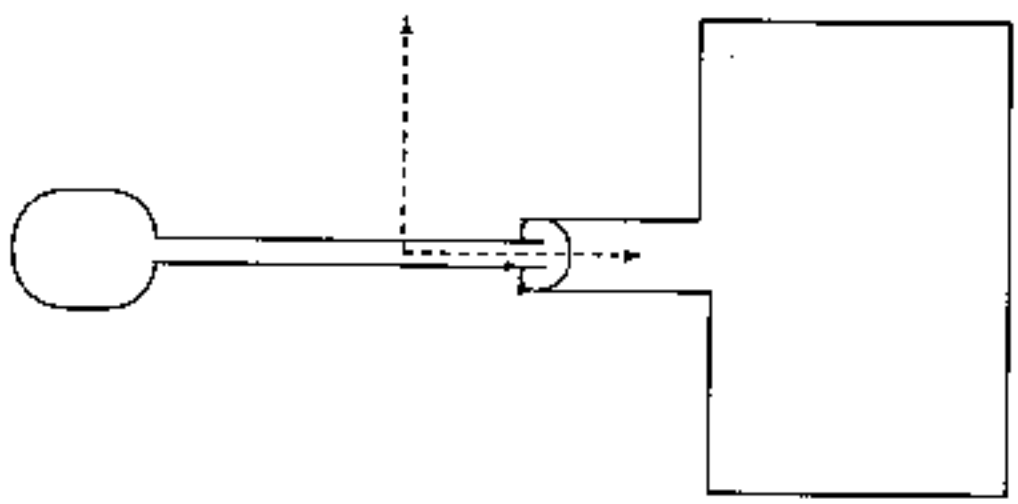

Figure 1. A two part object linked by a prismatic joint with an associated reference frame.

try compute a candidate 'joint line of slide' and a reference frame, having this line of slide as its *x-axis*, by applying the prerecorded transformation to the compound feature based coordinate frame. Cast a vote for the identity of the object together with the associated location and orientation of the reference frame.

(c) Check the accumulator of votes for high scoring pairs of (*object, line of slide*).

(d) Check the high scoring candidates, whether there are only two high scoring reference frame origins on the suggested *x-axis*.

(e) Verify the high scoring candidates by matching all of their edges. This verification can be done by applying the two candidate transformations to the appropriate model parts, and checking whether a relatively high percentage of the projected model edges are in close vicinity to image edges having a similar direction.

The differences between the rotational and prismatic joint cases are as follows. In the *R-table formation* phase for objects with rotary joints we picked the joint location as the origin of the object's reference frame, while the orientation of that frame was selected arbitrarily. In the case of a sliding joint we fix the object's reference frame orientation by aligning its *x-axis* with the joint's line of slide. The origin can be chosen arbitrarily on that axis. Thus, in step (b) of the *Recognition* phase we cast votes for the pair (*object, line of slide*) and expect to get a high voting candidate with only two significant clusters of frame origin location. In a way, the prismatic joint case may be viewed as the dual of the rotary joint case.

Obviously, the same *shape signature* may vote for (several) rotary joints as well as for (several) prismatic joints, the way it may vote for different objects. This information is naturally represented in the R-table.

3.3. Recognition of objects with multiple joints

Although objects with single joints probably comprise the largest group of man-made tools, there are also objects with *multiple joints*. In particular, most of the industrial robots are articulated objects with multiple joints. The extension of our method to handle multiple joint information is quite straightforward. Consider Figure 2. There we have depicted a four part object with two rotary joints and one prismatic joint. Parts B and C are connected to two joints, while parts A and D are connected to a single joint each. In the *R-table for-*

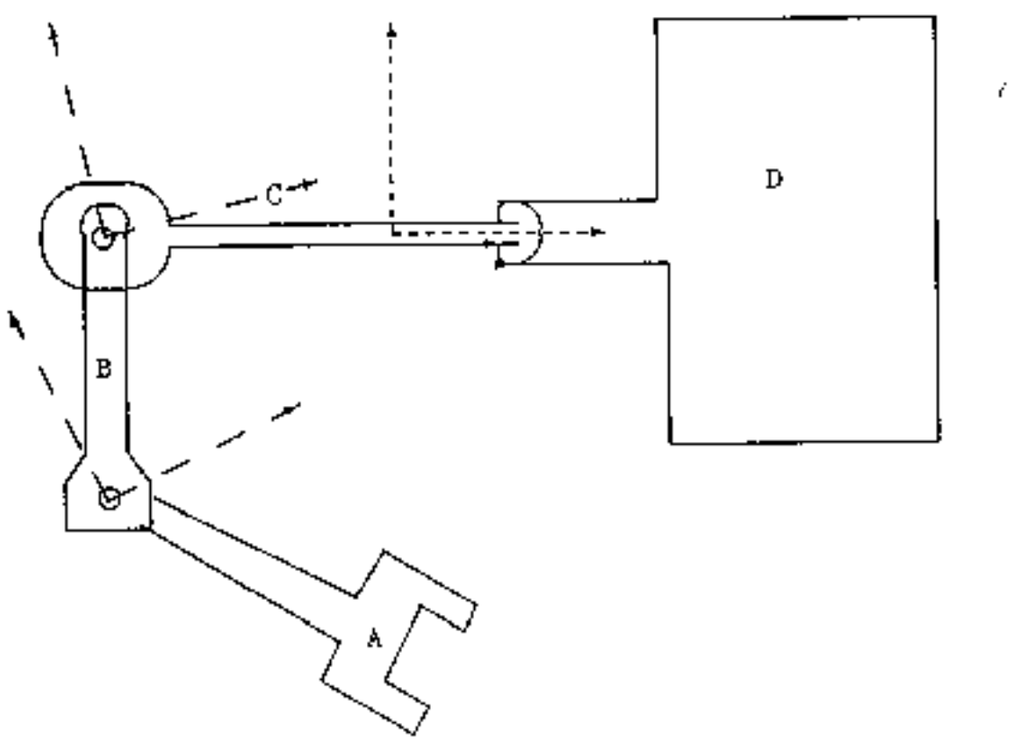

Figure 2. Multiple *joint* object.

mation phase the 'single joint' parts A and D will be processed exactly as explained above. Namely, each *compound feature* of part A will encode into the R-table the information about the object, and the transformation between the feature based reference frame and the appropriate rotary joint based reference frame, while each *compound feature* of part D will encode the transformation between its reference frame and the prismatic joint based frame. However, for parts B and C we will define two reference frames, one at each of its joints. Each *compound feature* of parts B and C will contribute two entries to the R-table, one for each reference frame. Otherwise, everything will be handled as before. In the *Recognition* phase the voting will be exactly the same. Of course, if one gets a significant vote for some part with different joint locations (or orientations), it is possible to introduce yet another consistency check at step (d) of recognition to verify, whether the high scoring joints might be on the same object.

4. Summary and discussion

We have presented an articulated object recognition approach which is based on an extension of the Generalized Hough Transform technique. This approach is quite general and applies to various 2-D to 2-D and 3-D to 3-D external object transformations as well as to internal transformations of objects with rotary and prismatic joints. It can even be extended to accommodate objects with multiple joints.

As we see it, the major advantage of our technique is its straightforward extension of the Hough voting scheme with very little degradation in performance compared to the much simpler rigid object recognition problem. This suggests that application of the GHT technique to the recognition of rigid objects only does not fully exploit the original power of the method. Our method shares the well known limitations of the GHT method, but does not add new ones, as usually happens when an existing object recognition technique is extended to handle articulated objects.

A preliminary implementation of the method for recognition of rotated, translated and scaled 2-D objects with rotary joints is presented in (Beinglass and Wolfson, 1991). Currently we are extending our approach to additional cases, in particular to the recognition of 3-D objects from 2-D images.

References

Ballard, D.H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition* 13(2), 111–122.

Ballard, D.H. and C.M. Brown (1982). *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ.

Beinglass, A. and H.J. Wolfson (1991). Articulated object recognition, or, how to generalize the generalized Hough transform. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Maui, June 1991.

Besl, P.J. and R.C. Jain (1985). Three-dimensional object recognition. *ACM Computing Surveys* 17(1), 75–154.

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

Chin, R.T. and C.R. Dyer (1986). Model-based recognition in robot vision. *ACM Computing Surveys* 18(1), 67–108.

Craig, J.J. (1986). *Introduction to Robotics*. Addison-Wesley, Readings, MA.

Goldberg, R. and D. Lowe (1987). Verification of 3-D parametric models in 2-D image data. *Proc. IEEE Workshop on Computer Vision*, Miami-Beach, FL, 255–257.

Grimson, W.E.L. (1987). Recognition of object families using parametrized models. *Proc. IEEE Intl. Conf. on Computer Vision*, London, England, 93–101.

Grimson, W.E. and T. Lozano-Pérez (1987). Localizing overlapping parts by searching the interpretation tree. *IEEE Trans. Pattern Anal. Machine Intell.* 9(4), 469–482.

Heller, A.J. and J.R. Stenstrom (1989). Verification of recognition and alignment hypothesis by means of edge verification statistics. *Proc. DARPA IU Workshop*, Palo Alto, CA, 957–966.

Huttenlocher, D.P. and S. Ullman (1988). Recognizing solid objects by alignment. *Proc. DARPA IU Workshop*, Cambridge, MA, April 1988, 1114–1122.

Illingworth, J. and J. Kittler (1988). A survey of the Hough transform. *J. Computer Vision, Graphics, and Image Processing* 44, 87–116.

Lamdan, Y., J.T. Schwartz and H.J. Wolfson (1988). Object recognition by affine invariant matching. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Ann Arbor, MI, June 1988, 335–344.

Lamdan, Y. and H.J. Wolfson (1988). Geometric hashing: a general and efficient model-based recognition scheme. *Proc. IEEE Intl. Conf. on Computer Vision*, Tampa, FL, Dec. 1988, 238–249.

Linnainmaa, S., D. Harwood and L.S. Davis (1988). Pose determination of a three-dimensional object using triangle pairs. *IEEE Trans. Pattern Anal. Machine Intell.* 10(5), 634–647.

Lowe, D.G. (1985). *Perceptual Organization and Visual Recognition*. Kluwer, Dordrecht.

Silberberg, T.M., L.S. Davis and D. Harwood (1984). An iterative Hough procedure for 3-D object recognition. *Pattern Recognition* 17, 621–629.

Stockman, G. (1987). Object recognition and localization via pose clustering. *J. Computer Vision, Graphics, and Image Processing* 40(3), 361–387.

Thompson, D.W. and J.L. Mundy (1987). Three-dimensional model matching from an unconstrained viewpoint. *Proc. IEEE Intl. Conf. on Robotics and Automation*, Raleigh, NC, 208–220.