Pattern Recognition Letters 2 (1984) 235–238
North-Holland
June 1984

*The 3-D Hough shape transform**

Thomas C. HENDERSON and Wu So FAI
Department of Computer Science, University of Utah, Salt Lake City, UT 84112, USA

Received 13 August 1983
Revised 13 January 1984

Abstract: The 3-D Hough shape transform is described which is used for the localization in space of 3-D objects defined in terms of the spatial organization of their features.

Key words: Hough transform, shape analysis, 3-D localization.

1. Introduction

The recovery of groups of features satisfying a certain organization plays a major role in pattern recognition. The original Hough transform was a method by which one could recover the parameters which defined an analytic form characterizing such an organization, e.g., straight lines or circles in Hough (1962). Subsequently, methods have been proposed for detecting sets of features satisfying non-analytic organizations by Merlin (1975), Davis (1980), and Ballard (1981). These methods are most often defined as 2-D shape modeling methods and are applied to features detected in 2-D images; we call this the 2-D Hough shape transform.

With the availability of true 3-D data, e.g., from laser range finders, the 3-D Hough shape transform has been explicitly defined and used in the analysis of that data by Henderson (1981,1982b). In this paper we discuss some of the problems involved in using the 3-D Hough transform and some solutions to these problems; also, see Wu (1983).

The use of the Hough shape transform consists of two steps:

1. Define the Hough shape model.
2. Apply the model to a set of detected features.

* This work was supported in part by the System Development Foundation and NSF grants ECS-83-07483 and MCS-82-21750.

The Hough model is defined in terms of a set, S, of feature points. Then one selects an appropriate reference point which is used to define the spatial relation between the points in S. Next, assuming that the feature points used to form the model have undergone a rigid motion (translation and rotation) and that some (or all) of the points are in the detected feature point set, the Hough shape transform uses the spatial organization of the detected feature points to recover the location of the transformed reference point. If the original reference point is chosen correctly, then the transformation can be determined.

2. Defining the Hough shape model

The feature used to define the shape is crucial to the success of the method. In the 2-D case, boundary edges of a shape are most often used. The corresponding feature in 3-D is a set of surface points. Whereas in 2-D processing the use of the edge feature greatly reduces the amount of computation required (edges occur sparsely in an image), this is not true in 3-D processing because the surface points *are* the data available. Moreover, surface points are dependent on the sampling rate and accuracy. Thus, a more invariant type of feature is required for 3-D shape modeling.

One solution to this problem is the use of control points which are recovered from the surface point data. If these control points are uniquely labeled simply by their nature, e.g., the vertexes of an asymmetric triangle, then the geometric transformation can be determined directly. That is, the correspondence problem between the model features and the detected features has already been solved. One needs then at least four control points to determine the transformation. In general, however, feature detectors are local in nature and no distinction exists between features, other than their locations; for example, the location of a vertex may be computed, but not its degree. In this paper we limit our discussion to the use of a vertex feature detector for polyhedral objects, but the control point method will work in general.

In particular, we define the Hough model in terms of vertexes derived from the detected surface points. First, planes are fit to the data using an efficient method; see Henderson (1982a). These planes are then intersected and the resulting vertexes used as the feature points. Call this set

$$S = \{(x_i, y_i, z_i)\}, \quad i = 1, \ldots, n.$$

Next, a reference point must be selected. In order to recover the orientation and location of an object, it is necessary that the reference point be chosen so that the distances from each of the model feature points to the reference point are unique. This permits a unique labeling to be assigned once the transformed reference point is determined. Thus, the definition of the Hough shape model is obtained as follows:

1. Select $S = \{X_i\}$, $i = 1, \ldots, n$, of model feature points.

2. Choose a reference point X_0 such that for every i and j, r_i is not equal to r_j, where

$$r_i = \text{distance}(X_i, X_0), \quad i = 1, \ldots, n.$$

3. The 3-D Hough shape model, M, is then the set $M = \{r_i\}$, $i = 1, \ldots, n$.

3. Applying the 3-D Hough shape transform

The location and orientation of an object is determined as follows:

1. Detect a set, D, of surface points. For the next steps, treat the connected components, D_i, separately.

2. Fit planes to the points in D_i.

3. Determine a set, V_i, of vertexes from the intersections of the planes.

4. Compute intersections of 3-D spheres of radius r_i, for r_i in M, centered at each vertex in V_i.

5. Determine if the point of maximum intersection of the spheres is the transformed reference point.

The amount of computation will obviously depend on the number of surface points detected and the number of vertexes derived from those points. The vertexes can be recovered in average case time complexity Order($N \log N$); see Henderson (1982a), where N is the number of detected points. The number of vertexes will usually be very small compared to the number of surface points, and therefore the number of vertexes in a connected component of surface points (assuming stable supports like tables and walls are discarded) will be small enough so that the complexity of computing the sphere intersections will not severely limit the system.

We now give in Figure 1 an algorithm based upon the Hough shape model for performing shape recognition.

The procedure DETERMINE ORIENTATION, when given the correspondence between the model reference point and the detected reference point, the model points and the detected vertexes, finds the transformation which maps the model points to the detected vertexes. The procedure INTERSECT returns true if the sphere, centered at a given detected ver-

```
procedure MATCH (detected_object);
  for every model in all_models
    max_count := MAX ACCUMULATOR (detected_object,model);
    if OBJECT IDENTIFIED
      DETERMINE ORIENTATION (detected_object,model);
integer procedure MAX ACCUMULATOR (detected_object,model);
  for every d in all_detected_vertexes
    for every a in all_accumulators
      for every r in all_model_radii
        if INTERSECT (sphere[d,r],a)
          then UPDATE ACCUMULATOR (a);
  return (FIND MAX ACCUMULATOR);
```

Fig. 1. Shape recognition algorithm.

tex with a radius equal to a given model radius, intersects a given accumulator. UPDATE ACCUMULATOR is a procedure that updates information pertaining to an existing accumulator which has been intersected, or creates new accumulators when necessary after intersection of an existing accumulator with a given sphere. FIND MAX ACCUMULATOR is a procedure that returns the maximum of all the accumulators.

3.1. Computing sphere intersections

The number of spheres to be intersected in determining the maximum accumulator can result in large amounts of computation. We outline a method for reducing that significantly. An accumulator can represent a 3-space point, circle or sphere. Whenever a new sphere is to be intersected with the existing accumulators, an analytic method can be used to determine the intersection based on the types of accumulators involved. Moreover, it is not usually necessary to compute the intersections of all the spheres, since reasonable pairs of vertexes can be used to form an initial set of accumulators; then the spheres are intersected with these accumulators only. For example, vertexes linked by an edge probably lie on the same object, and thus, they can be used to form an initial set of accumulators. Thus, at the worst, a linear search for a pair of vertexes on the same object must be performed, i.e., for each detected vertex, pair it with its neighbors, and use that pair to form a set of initial accumulators.

Consider the 3-D Hough shape definition for a cube:

1. $S = \{(-1,-1,1), (-1,1,1), (1,1,1), (1,-1,1), (-1,1,-1), (-1,-1,-1), (1,1,-1), (1,-1,-1)\}$
2. Choose

   $X_0 = (1.5, 1.3, 1.25)$.
3. Then

   $M = \{4.09, 3.43, 3.38, 3.28, 2.54, 2.40, 2.33, 0.65\}$.

Next, the location and orientation are determined:

1. Assume vertexes detected directly.
2. Assume vertexes detected directly.
3. $V_i = \{(4,-1,1), (4,1,1), (6,1,1), (6,-1,1), (4,1,-1), (4,-1,-1), (6,1,-1), (6,-1,-1)\}$.
4. Figure 2 shows the initial circular accumulators generated by the intersections of the first pair of points. The detected reference point is at (6.5, 1.3, 1.25).

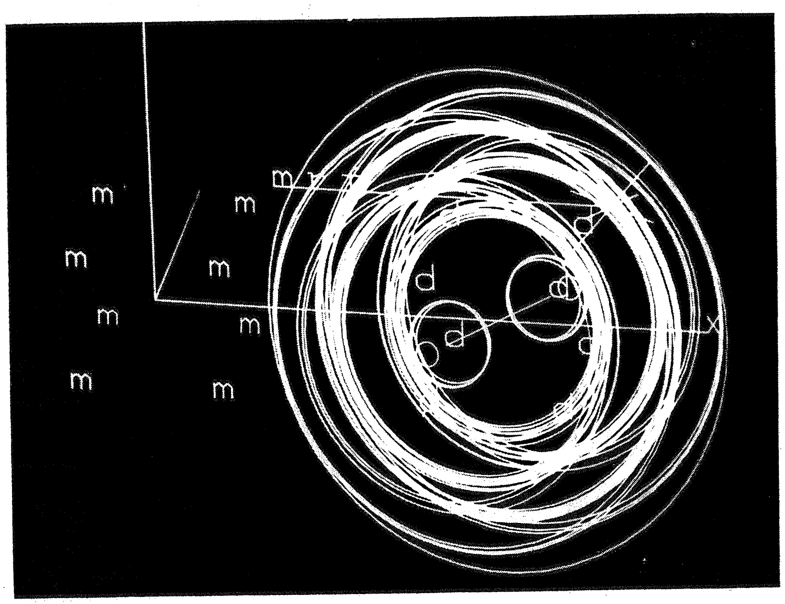

Fig. 2. Initial circular accumulators. m's denote model vertices; mrf denotes model reference point; d's denote detected vertices; * denotes detected reference point.

5. The transformation matrix is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 5 & 0 & 0 & 1 \end{bmatrix}.$$

In this case the detected cube is an instantiation of the model cube which has been translated by five units in the positive direction along the x-axis.

References

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

Davis, L.S. and S. Yam (1980). A generalized Hough-like transformation for shape recognition. Tech. Report TR-134, Univ. of Texas (Feb.).

Henderson, T.C. (1982a). An efficient segmentation method for range data. *SPIE Conference Proceedings*, Arlington, VA (May) 46–47.

Henderson, T.C. and B. Bhanu (1982b). Three-point seed method for the extraction of planar faces from range data. *Proceedings IEEE Workshop on Industrial Applications of Machine Vision*, Research Triangle Park, North Carolina (May) 181–186.

Henderson, T.C. and N. Keskes (1981). L'analyse des objets tridimensionnels. *Proceedings 3rd Conference on Pattern Recognition and Artificial Intelligence*, Nancy, France (Sep.) 277–287.

Hough, P.V.C. (1962). Method and means for recognizing complex patterns. U.S. Patent 3069654.

Merlin, P.M. and D.J. Farber (1975). A parallel mechanism for detecting curves in pictures. *IEEE Trans. on Computers* 24, 96–98.

Wu So Fai (1983). A multi-sensor integration and data acquisition system. Master's Thesis, U. of Utah (June).