LOCAL ROTATIONAL SYMMETRIES

Margaret M. Fleck

Artificial Intelligence Laboratory
Massachusetts Institute of Technology

Abstract:

This paper presents a new representation for two-dimensional round regions called Local Rotational Symmetries. This representation is intended as a companion to Brady's Smoothed Local Symmetry Representation for elongated shapes. A local multi-scale algorithm for computing Local Rotational Symmetry regions has been implemented. Results are presented from this implementation and from a re-implementation of Smoothed Local Symmetries.

Introduction

A person looking at a grey-scale camera image of an object can easily produce a rough description of its (two-dimensional) shape. For example, a spoon might be described as a long thin part (the handle) joined to a roundish part (the bowl). Such descriptions can be used for recognition of objects, for practical reasoning about objects and actions, and for representing the meanings of natural language words. It is extremely difficult to get a computer to produce even such basic descriptions of object shape. Most existing systems for representing shape are a poor match to human capabilities. The goal of the work described in this paper is to produce shape descriptions of the sort that people use.

In this paper, a revised and condensed version of my master's thesis,[1] I will present a new representation for two-dimensional round regions, called *Local Rotational Symmetries.* This representation is a companion to Brady's Smoothed Local Symmetries,[2,3] which represent elongated or pointed regions. Smoothed Local Symmetries and Local Rotational Symmetries are computed from the boundaries of regions in the image, rather than directly from the grey-scale image. Figure 1 shows the edges extracted by Canny's[4] edge finder from images of some familiar objects. Figure 2 shows the axes of Smoothed Local Symmetry regions found for these objects.

Smoothed Local Symmetries provide good representations for elongated or pointed regions such as the handle of the teaspoon or the pointed end of the pear. However,

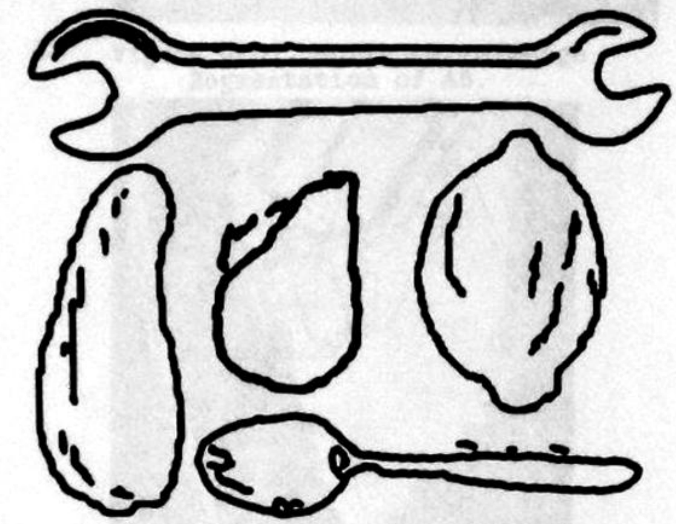

Figure 1. Edges from grey-scale camera images of five familiar objects: a spanner wrench (top), a squash (bottom left), a pear (middle), a lemon (middle right), and a teaspoon (bottom right).

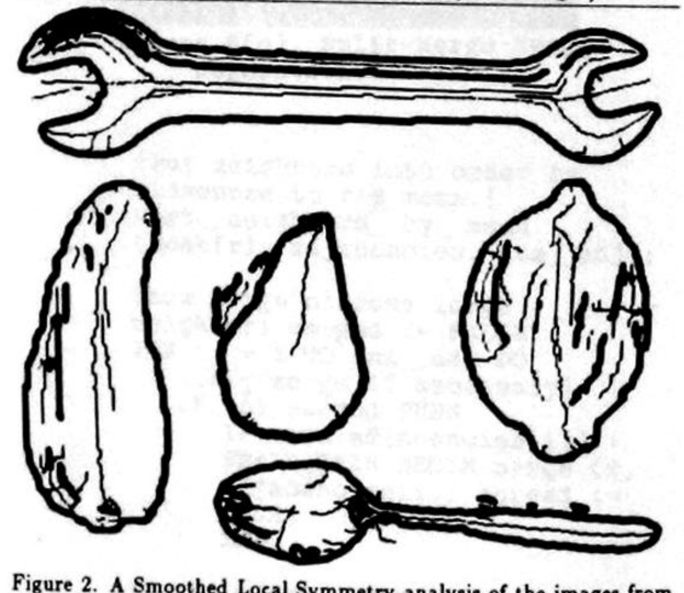

Figure 2. A Smoothed Local Symmetry analysis of the images from Figure 1. The thick lines are the region boundaries from the images. The thin lines are the axes of Smoothed Local Symmetry regions. Smoothed local symmetry regions include elongated regions such as the handle of the teaspoon and pointed regions such as the pointed end of the pear.

CH2290-5/86/0000/0332$01.00 © 1986 IEEE

an axis-based representation will not provide intuitively acceptable analyses for round regions, such as the lemon or the round end of the pear, and it will be unstable on such regions. Such regions are most naturally analyzed in terms of a center. Figure 3 shows a Local Rotational Symmetry analysis of the same objects. This analysis provides representations of round parts of the objects in terms of intuitively plausible center locations. The combination of the two representations provides complete descriptions of object shapes.

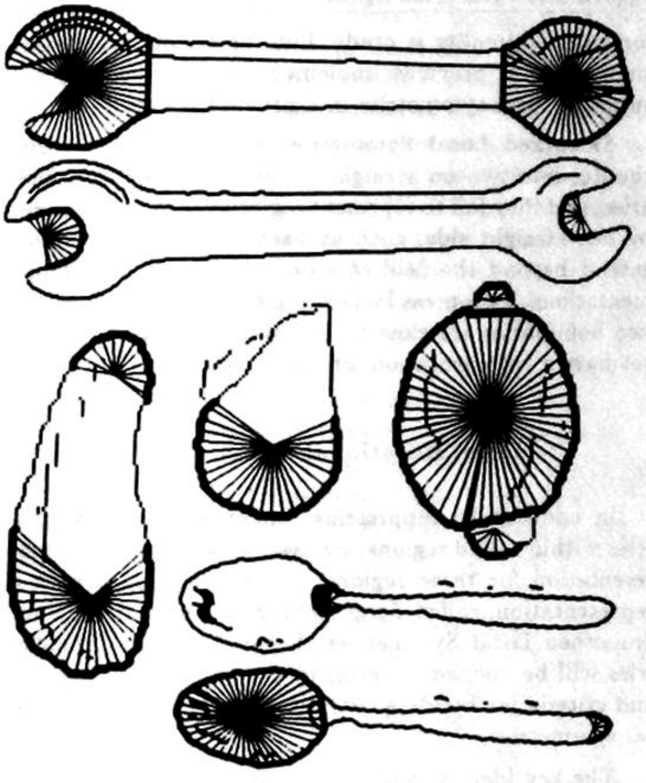

Figure 3. Local Rotational Symmetry analysis of the images in Figure 1. The thin lines show the region boundaries found in the finest-scale version of these images. The boundary of each round region is shown as a thicker line and selected radii from the center of the region to the boundary are also displayed. For example, four regions are found for the teaspoon: the bowl, the round end of the handle, and four small round regions in edges from specular reflections on the bowl.

Smoothed Local Symmetries

Most shape representations describe the shape of an entire region such as a rectangle, a corner, or an ellipse. Local symmetry representations model only local sections of a region, such as two pieces of boundary that are opposite one another in an elongated region. Instances of these local models are found in an image and then joined together to form larger connected regions.

The Smoothed Local Symmetry representation for elongated shapes [2,3,5,6] is based on a local model for two small pieces of boundary opposite one another in an elongated region, called a local (reflectional) symmetry. The key observation is that two such pieces of boundary will be approximately reflections of one another in the perpendicular bisector of the line segment joining them. The perpendicular bisector is a local approximation to the axis of the region. The line segment joining the two pieces of boundary is called a rib and its length provides a local width for the region. The model also specifies which side of each boundary is inside the region. A normal to a boundary can be used to represent a small piece of boundary. The degree of local reflectional symmetry between two small pieces of boundary can then be measured by the angular deviation of their two normals from an exact reflectional symmetry, as shown in Figure 4.

A Smoothed Local Symmetry region is formed by grouping local reflectional symmetry pairs into connected regions with connected boundaries. The current implementation uses a boundary tracking algorithm to build regions, producing an ordered list of symmetry pairs for each region. Adjacent ribs can share one endpoint, but they cannot cross one another. The axis of the region is computed by connecting the midpoints of adjacent ribs. It corresponds well to the perceived axis of the region.

The area occupied by a symmetry region is the area enclosed by the two boundaries and by the first and last ribs. The symmetries in a region must progress in a consistent direction along the axis, so that there is a consistent notion of which side of each boundary is inside. In order for a region to be perceptually salient, it should have a high aspect ratio (ratio of length to width). A region is perceived as several subregions if it contains minima of width (except at its ends) or sharp changes in parameters, such as sharp bends or sharp changes in width or derivative of width.

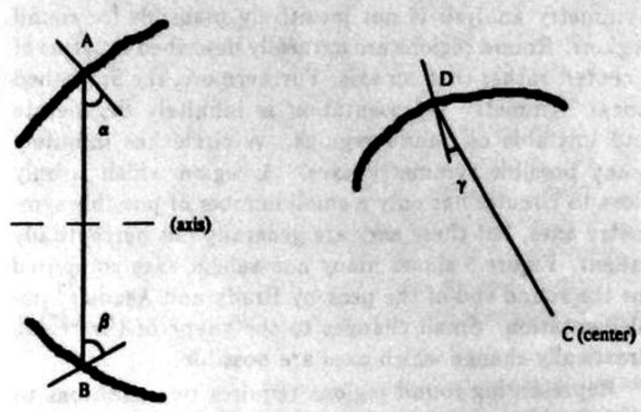

Figure 4. Left: Two small pieces of boundary A and B have a local reflectional symmetry to the extent that the angle α between the normal at A and AB is close to the angle β between the normal at B and AB. Right: A small piece of boundary D has a local rotational symmetry about a center C to the extent that the angle γ between DC and the normal to the boundary at D is small.

A Smoothed Local Symmetry region may be created by joining two regions into one longer region, by creating sections of boundary that are not in the input image. The new region, including points created in the join, should meet all the previous requirements for a good Smoothed Local Symmetry region. Furthermore, the area of the join should be small compared to the width of the region and to the lengths of the boundaries that were actually present in the input.

Smoothed Local Symmetries offer several advantages over previous shape representations. The local symmetry model plus rules for building regions allow one to compute representations for a wide class of regular and irregular shapes using simple algorithms. Programs using whole region models, such as Generalized Cylinders,[7] typically search for only a restricted class of shapes, because the algorithms for fitting a region model to input boundaries are conceptually difficult. The local symmetry representation includes a constructive definition of the axis of an elongated region. Finally, the local symmetry representations take advantage of boundary connectivity information, unlike the Hough Transform.[8]

Problems with Smoothed Local Symmetries

The goal of the current work is to develop a local symmetry representation for round regions. Circles are obviously round regions, but perfect circles rarely occur in natural objects. So I also include regions which are slight deformations of circles, such as ovals. Round bumps or dents and round ends of elongated regions are also intuitively round. I will also consider spirals to be round regions because they are intuitively round and an analysis for them follows automatically from the analysis of other round regions.

Smoothed Local Symmetries provide a good representation for elongated regions, but the Smoothed Local Symmetry analysis is not intuitively plausible for round regions. Round regions are naturally described in terms of a center, rather than an axis. Furthermore, the Smoothed Local Symmetry representation is infinitely degenerate and unstable on round regions. A circle has infinitely many possible symmetry axes. A region which is only close to circular has only a small number of possible symmetry axes, but these axes are generally not perceptually salient. Figure 5 shows many non-salient axes computed for the round end of the pear by Brady and Asada's[3] implementation. Small changes to the shape of the region drastically change which axes are possible.

Representing round regions requires two additions to Smoothed Local Symmetries: a better representation for round regions and a method for suppressing reflectional symmetries within round regions. My implementation does not suppress all reflectional symmetries within round regions, but only symmetry pairs which are not locally optimal. Thus, it does not suppress the reflectional symmetry regions in the corners of a hexagon or the spiral axis between two concentric spiral boundaries. My test

Figure 5. The Smoothed Local Symmetries of the pear, as computed by the Brady and Asada[3] implementation, which did not consistently suppress axes within round regions.

for local optimality is crude, but the results are cleaner than those of previous implementations, which did not suppress these symmetries in a principled way.

Smoothed Local Symmetries are also unstable and counter-intuitive on straight or almost straight boundaries, and they fail to represent regions which are bounded by one straight side, such as background regions which extend beyond the field of view. As in previous implementations, I suppress local symmetry pairs in which the two boundaries are close to collinear. However, I do not yet have a representation for these regions.

Local Rotational Symmetries

In addition to suppressing Smoothed Local Symmetries within round regions, we need a local symmetry representation for these regions. I have developed such a representation, called *Local Rotational Symmetries*. Like Smoothed Local Symmetries, Local Rotational Symmetries will be defined in terms of a local symmetry model and criteria for building connected regions from these local symmetries.

The key idea is that, in a perceptually round region, local sections of the boundary are close to being rotationally symmetric about the perceived center of the region. That is, if the boundary is rotated a small amount about the center, it rotates approximately onto itself. I use the normal at a point on the boundary to represent a small piece of boundary. The angular distance between this normal and the line segment joining the boundary point to a center point can then be used to measure the degree to which the piece of boundary is locally rotationally symmetric about the center, as shown in Figure 4. The distance from the boundary point to the center is a local measure of the radius of the region. The local model also specifies which side of the boundary is inside the region.

A Local Rotational Symmetry region is formed by joining together sections of boundaries with local rotational symmetries to a common center location, to form a connected region with a connected boundary. The area occupied by the symmetry region is enclosed by its boundary and by the line segment joining the ends of the boundary (if the boundary is open). The boundary must proceed in a consistent direction around the center, so that there is a consistent notion of which side of the boundary is inside. Spiral regions do not coherently bound an area. A round region is more perceptually salient if its boundary has a long angular length. That is, all other things being equal, a closed boundary forms a better round region than a half-open region, which is better than a boundary with a shallow curve.

Brady's[2] original definition of Smoothed Local Symmetries required that the two points in a symmetry pair have an exact reflectional symmetry. In round regions, it is not possible to use exact symmetries. Smoothed Local Symmetries relate only two pieces of boundary to each local axis of reflection, whereas Local Rotational Symmetries relate many pieces of boundary to a common center of rotation. Thus, deformation causes points on the boundary of a round region to have inexact symmetries, as much as 30-40 degrees from perfect symmetry in a perceptually round oval. Building rotational symmetry regions requires a compromise between increased length and increased deviation of the boundary points. My use of inexact symmetries is the largest difference between Local Rotational Symmetries and the Symmetric Axis Transform.[9]

Smoothed Local Symmetries can also be made less sensitive to noise by using inexact symmetry pairs. The current implementation uses symmetry pairs which are up to 20 degrees from an exact symmetry and is less sensitive to noise than previous implementations.[3,6] Successful use of inexact symmetries depends on constructing regions by building connected boundaries, rather than by building connected axes.

The connected boundary of a round region may be formed by joining disconnected sections of boundary, since attachment or occlusion may remove part of a region boundary. The new boundary, including points added to fill the gap, must meet all the previous criteria for a good Local Rotational Symmetry region. Joining disconnected pieces of boundary is only perceptually plausible if the length of the gap is small compared to the lengths of the real boundaries. However, the total percentage of real points in the new region does not accurately measure the plausibility of a join. Rather, it is the percentage of real points *locally* which is crucial and there is a maximum angular length (about 40-50 degrees) beyond which it never seems plausible to make a join.

To a local algorithm, considering only a limited range of angles, an open boundary looks just like a gap to be connected and a spiral looks just like a slightly deformed circle. This makes the correct prediction that the ends of an open boundary will be joined under the same conditions as other gaps. In particular, round bumps with a gap larger than 40-50 degrees and round ends of elongated regions are best described using open boundaries, since the area they occupy seems to be bounded by a partial round arc plus the line joining its ends. A second corollary is that spirals will be analyzed as round regions.

Evaluation of a proposed round region involves a tradeoff among the various factors described above. The implemented evaluator uses an empirically determined combination of these factors. Although human evaluation of regions seems to be local, the formula used in the implementation is not local, due to the inevitable lag between theory and implementation. Results of the program suggest that the penalties for joining disconnected boundaries are too low. Fine-tuning the evaluation is a topic for future research.

Since Local Rotational Symmetries are not exact, a typical round region has plausible analyses using centers near the perceived center, in addition to an analysis using the perceived center. Also, a fixed center may generate multiple regions which are similar, but with variations in boundary points and joins. A region which is similar to another region with a better evaluation is not perceptually salient. Thus, picking out just the perceived center of a round region requires finding *locally optimal* pairs of a center and a boundary.

The current implementation first computes the best regions for each center location using a heuristic technique for constructing the best connected regions for a fixed center location. For each region, the program computes a center location and an ordered list of boundary point locations relative to the center. See Fleck[1] for details of the algorithm and its results. This algorithm is not optimal, but the regions that it produces look good. Once candidate regions have been computed for all center locations, the locally best regions are selected from among them. The current implementation suppresses a region when more than 50 percent of its boundary points are the same as the boundary points of a better region.

Computation at multiple scales

The shape representations discussed above represent shapes at a fixed resolution. Consider the boundaries of the key shown in Figure 6 (left). The overall shape of the key is a round part plus a long part, but the key also has small teeth on its side which are crucial in explaining what the key is used for. For such objects, we need to represent shape at multiple scales of resolution. The coarse scales describe the overall shape of the object and the fine scales describe the details. There has been considerable work on multiple-scale representations, including multiple-scale edge finding,[10] multi-grid surface reconstruction,[11] describing curve or surface shape,[12,13] shape representations,[3,5] robot motion planning,[14] and high-level planning.[15]

There are several ways to abstract away detail in an image. Feature dropping techniques take a fine-scale representation of the image and remove regions or other features which are small or "unimportant." Threshold changing involves changing the setting of a threshold that is used to select which features are "salient enough." Such techniques are useful, but they cannot produce qualitative changes in representation between scales, such as between the overall shape of a brush and the shape of its bristles.

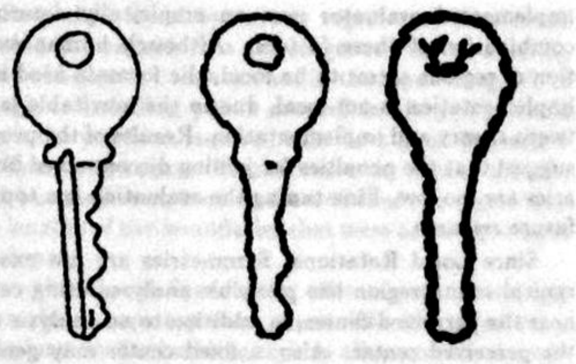

Figure 6. The edges from a image of a key, at three scales (of eight).

Smoothing techniques, on the other hand, can produce such qualitative changes. For two-dimensional images, there are two alternatives: smooth the image and then find region boundaries or extract fine-scale region boundaries and smooth them. In my implementation, the image is smoothed with a Gaussian and sampled at a rate proportional to the amount of smoothing, creating a pyramid structure in which each image is half the area of the preceding one. Figure 6 shows the edges for three scales for a key. The edges at each scale are matched (many to many) to edges at adjacent scales with similar location and orientation.

Previous implementations of local symmetries smooth the boundaries, rather than the image. Smoothing the image is more robust. Current boundary smoothing techniques cannot proceed across gaps in the boundary or across places at which several boundaries meet. Boundary smoothing cannot merge two adjacent regions when there are disconnected boundaries, as in a grating. Deciding whether to keep two regions distinct or merge them requires higher-level knowledge. Image smoothing generally blurs nearby objects into one another, but it is possible to inhibit smoothing across particular boundaries.[13,16] Thus, it gives high-level processing both options. Similarly, image smoothing should allow very thin regions to be either blurred away or selectively retained.

It is not practical to compute local symmetries between all pairs of two boundary points or a boundary point and a center. Such a computation grows as $O(n^2)$ in the area of the image. Data compression techniques, such as the curvilinear boundary approximations used by Brady and Asada,[3] do not change the rate of growth. In order to make the computation efficient, I restrict exhaustive computation to be local at each scale. Smoothed Local Symmetries are only computed exhaustively for boundary points within 15 pixels of one another and Local Rotational Symmetries are only computed for boundary points and centers at most 8 pixels apart.

In order to compute fine-scale representations of larger regions, it is necessary to pass suggestions from coarser scales to finer scales. For each coarse-scale region, the program locates the fine-scale boundary points that match

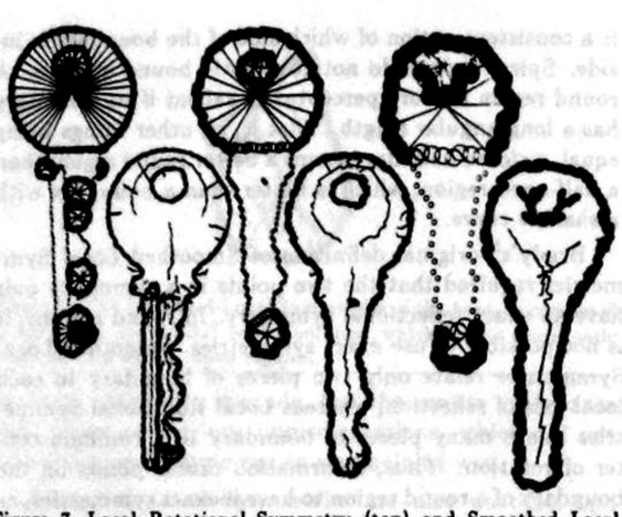

Figure 7. Local Rotational Symmetry (top) and Smoothed Local Symmetry (bottom) analyses of the key at three scales.

the coarse-scale region boundaries and (for round regions) fine-scale center locations that could match the region center. Local symmetries are then computed between these fine-scale points. Region building combines symmetries from the local exhaustive computation and symmetries from suggestions. Thus, the program can build regions which change drastically in width over their length, such as spirals or pointed corners. Figure 7 shows the output of the local multi-scale algorithm on the key boundaries.

The symmetries eliminated by locality seem not to be perceptually salient. For example, Figure 8 shows the Smoothed Local Symmetries of the key at a fine scale, as computed by the Brady and Asada[3] code. Many of the axes in this analysis are between tiny pieces of boundary in unrelated parts of the figure and they strike people as meaningless junk. While these symmetries can be filtered out as having poor aspect ratios,[5,6] locality prevents them from ever being computed.

Regions at adjacent scales which are basically the same can be matched and the multi-scale representation sum-

Figure 8. The Smoothed Local Symmetries of the key at a fine scale, as computed by the Brady and Asada[3] implementation, which was not local.

marized by describing the range of scales at which each distinct region occurs and how each region is replaced by a different set of regions at a finer or coarser scale. This type of summary is what Witkin[17] calls a "scale-space" representation. A rough version of this has been implemented, but more work is needed to handle the different ways that two-dimensional regions can change between scales. For example, a region can break up into multiple regions and a pointed corner elongates as more detail is added at its end. This analysis is more complex than the analysis of Witkin's one-dimensional features.

Conclusions and Future Work

The system described above has been implemented and the Local Rotational Symmetry part has been tested on about 40 images (see Fleck[1] for details). The current implementation does not yet join disconnected Smoothed Local Symmetry regions, although Connell[5] did this for the Brady and Asada implementation. The examples shown are among the simpler images tested, but they are typical of its performance. In general, the program produces an intuitively reasonable set of regions for each figure and it is not sensitive to clutter or distortion.

This implementation is slow. Analysis of the pear figure to the finest scale takes about three hours. However, the current algorithm is far from optimal. Since the shape representation is local, the asymptotic complexity of an optimal algorithm should be linear in the size of the input and it should speed up drastically on parallel hardware.

There are many ways in which to this work could be extended, including building three-dimensional local symmetry representations, using region intensity information to avoid symmetries between boundaries with different interior colors, converting the raw symmetry output into a decomposition of a figure into subregions, and computing symbolic descriptions of the regions.

Acknowledgements

The research described in this paper was done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the Laboratory's Artificial Intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-80-C-0505, the Office of Naval Research under contract number N00014-77-C-0389, and the System Development Foundation. The author is also supported by the Fannie and John Hertz Foundation and by the Bell Laboratories Graduate Research Program for Women. Jon Connell provided helpful comments.

References

[1] Margaret M. Fleck, "Local Rotational Symmetries," MIT AI Lab. TR-852, 1985.

[2] J. Michael Brady, "Criteria for Representations of Shape," pp. 39-84 in A. Rosenfeld and J. Beck, eds., *Human and Machine Vision*, Academic Press, NY, 1983.

[3] J. Michael Brady and Haruo Asada, "Smoothed Local Symmetries and Their Implementation," *International Journal of Robotics Research*, 3/3, pp. 36-61, 1984.

[4] John F. Canny, "Finding Edges and Lines in Images," MIT AI Lab., TR-720, 1983.

[5] Jonathan H. Connell, "Learning Shape Descriptions: Generating and Generalising Models of Visual Objects," MIT AI Lab. TR-853, 1985

[6] Scott S. Heide, "A Hierarchical Representation of Shape from Smoothed Local Symmetries," MS thesis, MIT Dept. of Mech. Eng., 1984.

[7] Rodney A. Brooks, "Symbolic Reasoning among 3-D models and 2-D images," *Artificial Intelligence*, 17, pp. 285-348, 1981.

[8] Dana H. Ballard, "Generalising the Hough Transform to Detect Arbitrary Shapes" *Pattern Recognition*, 13, pp. 111-122, 1981.

[9] Harry Blum and Roger N. Nagel, "Shape Description using Weighted Symmetric Axis Features," *Pattern Recognition*, 10, 167-180, 1978.

[10] V. Torre and T. Poggio, "On Edge Detection," MIT Artificial Intelligence Laboratory, AIM-768, 1984

[11] Demetri Terzopoulos, "Multiresolution Computation of Visible-Surface Representations," Ph.D. dissertation, MIT Dept. of Electrical Eng. and Computer Science, 1984.

[12] Haruo Asada and J. Michael Brady, "The Curvature Primal Sketch," MIT Artificial Intelligence Laboratory, AIM-758, 1984.

[13] Jean Ponce and J. Michael Brady, "Toward a Surface Primal Sketch," Takeo Kanade, ed., *Three-Dimensional Visual Systems*, Kluwer Academic Publishers, Hingham, MA, 1985.

[14] Rodney A. Brooks and Tomás Lozano-Péres, "A Subdivision Algorithm in Configuration Space for Findpath with Rotation," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-15, 1985.

[15] Ramesh S. Patil, "Causal Representation of Patient Illness for Electolyte and Acid-Base Diagnosis," MIT Laboratory for Computer Science, TR-267, 1981.

[16] W. Eric L. Grimson and Theo Pavlidis, "Discontinuity Dectection for Visual Surface Reconstruction," *Computer Vision, Graphics, and Image Processing*, 30, pp. 316-330, 1985.

[17] Andrew P. Witkin, "Scale-Space Filtering," Proceedings of the 8th IJCAI, 1983.