[11] M. Hashimoto and J. Sklansky, "Multiple-order derivatives for detecting local image characteristics," *Comput. Vision, Graphics, and Image Processing*, vol. 39, pp. 28-55, 1987.

[12] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems.* Englewood Cliffs, NJ: Prentice-Hall, 1974.

[13] H. H. Nagel, ''Displacement vectors derived from second-order intensity variations in image sequences," *Comp. Vision, Graphics, and Image Proc.*, vol. 21, pp. 85–117, 1983.

[14] ——, "Constraints for the estimation of displacement vector fields from image sequences," in *Proc. Eighth Int. Joint Conf. Artif. Intell.* (Karlsruhe, Germany), 1983, pp. 945–951.

[15] ——, "On the estimation of optical flow: relations between different approaches and some new results," *Artif. Intell.*, vol. 33, pp. 299–324, 1987.

[16] H. H. Nagel and W. Enkelmann, "An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 8, no. 5, pp. 565-593, 1986.

[17] T. Poggio, V. Torre, and C. Koch, "Computational vision and regularization theory," *Nature*, vol. 317, pp. 314-319, 1985.

[18] C. Schnörr, ''Untersuchung von modifizierten Ansätzen zur Ermittlung von örtlichen und zeitlichen Ableitungen bei der Schätzung des Optischen Flusses in Bildfolgen," Master thesis, Fakultät für Elektrotechnik der Universität Karlsruhe (TH), Nov. 1987.

[19] ——, "Determining optical flow for irregular domains by minimizing quadratic functionals of a certain class," *Int. J. Comput. Vision*, vol. 6, no. 1, pp. 25-38, 1991.

[20] ——, "Computation of discontinuous optical flow by domain decomposition and shape optimization," *Int. J. Comput. Vision*, vol. 8, no. 2, pp. 153–165, 1992.

[21] M. A. Snyder, "On the mathematical foundations of smoothness constraints for the determination of optical flow and for surface reconstruction," *IEEE Proc. Workshop Visual Motion* (Irvine, CA), Mar. 20–22, 1989, pp. 107–115.

[22] D. Terzopoulos, ''Multilevel computational processes for visual surface reconstruction," *Comput. Vision, Graphics, and Image Processing*, vol. 24, pp. 52-96, 1983.

[23] O. Tretiak and L. Pastor, "Velocity estimation from image sequences with second order differential operators," in *Proc. Int. Conf. Patt. Recogn.* (Montreal, Canada), 30.7.–2.8., 1984, pp. 16–19.

[24] S. Uras, A. Verri, F. Girosi, and V. Torre, "A computational approach to motion perception," *Biol. Cybern.*, vol. 60, pp. 79–87, 1988.

Determining the Axis of a Surface of Revolution Using Tactile Sensing

Matthew D. Berkemeier and Ronald S. Fearing

Abstract—**Dextrous robot hands need to be able to determine the pose of objects to reliably grasp and manipulate them. The first few contacts with an object can be used to provide an initial estimate of this information if we constrain the object to be of a particular class. This paper considers a simple example of exploiting class constraints: finding the axis of an unknown surface of revolution. Three tactile curvature measurements on a surface of revolution with twice-differentiable sweeping rule function are shown to be sufficient for determining the axis except for certain singular configurations. Position and orientation error uncertainties and experimental results are presented for a cylindrical tactile sensor.**

Index Terms—**Class constraint, pose identification, surface of revolution, tactile sensing.**

I. Introduction

Dextrous robot hands need to be able to determine the pose of objects to reliably grasp and manipulate them. Fingertip-mounted tactile sensors are attractive for providing this feedback since, in addition to supplying the contact location and surface normal (in our case, also the curvature directions), the force applied can be determined at each contact.

We take as motivation the problem of finding the position and orientation of an unknown surface of revolution from an initial, attempted grasp with three fingers. The information from these initial contacts can be exploited to obtain an estimate of object pose. This initial, nonrobust object pose estimate could be used to direct further object exploration or to attempt an initial grasp. Concentrating on the information from an initial set of contacts avoids consideration of the problem of the object changing position while the fingers move to explore new contacts. On the other hand, it requires working with very sparse data.

Two methods of determining the shape and properties of an object from sensory data have been used previously: model matching and shape description without specific models.

Model matching can localize and identify objects by comparing relations between sensed features and features on particular object models. For example, Gaston and Lozano-Perez and Faugeras and Hebert [12], [9] determine object shape, location, and orientation by matching features in the world to specific object models. Ellis [8] extends this work by developing a planning system to choose tactile sensor paths, using prior sensed data, that prune an interpretation tree more efficiently.

Shape description uses measurements and geometric constraints to derive a representation of the object, and although specific models are not used, typically, objects are assumed to be of a certain class. Brady *et al.* [5] use range-finder data to describe surfaces in

Manuscript received June 22, 1991; revised August 11, 1992. This work was supported by NSF grant IRI-8810585. Recommended for acceptance by Associate Editor T. C. Henderson.

M. D. Beremeier was with the Electronics Research Laboratory, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720. He is now with the Department of Aerospace and Mechanical Engineering, Boston University, Boston, MA 02215.

R. S. Fearing is with the Electronics Research Laboratory, Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720.

IEEE Log Number 9211832.

0162–8828/93$03.00 © 1993 IEEE

terms of bounding contours, surface intersections, lines of curvature, and asymptotes. Allen [1] describes objects by surface patches and builds a representation of the object with a tactile sensor, in combination with vision, by exploration over the whole object. Cole and Yap [6] assume polygonal objects and describe algorithms that determine the shape using "probes" that determine only the contact location. Rao and Nevatia [16] use a class of linear straight homogeneous generalized cylinders and, with a feature-based stereo system, solve for the representations of cones and cylinders of various cross sections. Allen and Michelman [2] assume that objects can be adequately represented by superquadrics and, using the points of contact from a Utah/MIT hand, determine the superquadric and rotation/position parameters. Printz [15] presupposes cones or cylinders of circular or elliptical cross section and finds the generating axis by analyzing the extended Gaussian image of the object. Grimson [13] assumes that all objects to be encountered are in a model database but allows certain parameters to be "free," e.g., the length of the object or the angle of a joint.

This paper is another example of the shape description method. The particular object class that we have chosen is the surface of revolution. Although it is not a very general class of objects, they are common in man-made environments. Unlike most of the work mentioned above, we assume extremely sparse sensing (three contacts). Nevertheless, our method for determining the axis is applicable to any surface of revolution with a twice differentiable sweeping rule function. This is due to the special properties of this particular class.

In Section II, we describe a method for determining the axis that uses the contact location, surface normal, and principal curvature directions at three contacts. All possible singular configurations are then derived in Section III. Section IV contains both worst-case and average-case error estimations for our tactile sensor [11]. Finally, Section IV contains some experimental results.

Similar ideas were used by Fearing in [10], where it was shown how the axis and orientation of unknown simple cones (linear straight homogeneous generalized cylinders) could be determined from a minimum of three independent curvature measurements. This paper is an extension of [4] with more complete experimental results and analysis.

II. Surface of Revolution Geometry

In this section, we present the basic equations for the features of the surface of revolution that can be determined by our sensor. We also show how these sensed features can be used to determine the surface of revolution axis.

A. Notation

We will use the following notation:

$\vec{p}$ vector to the point of contact on the surface of revolution
$\hat{k}_\theta$, $\hat{k}_z$ unit vectors of the two principal directions of curvature
κ_θ, κ_z magnitudes of the curvature along the two above principal directions, where κ_θ is along $\hat{k}_\theta$, and κ_z is along $\hat{k}_z$
$\hat{n}$ unit surface normal at the contact point.

Our tactile sensor is capable of determining each of these parameters at a point on an object [11].

B. Basic Equations

For simplicity, we choose a coordinate system such that the z axis coincides with the axis of the surface of revolution under consideration. The surface of revolution can then be described by the vector equation

$$\vec{x}(\theta, z) = [r(z)\cos\theta, r(z)\sin\theta, z]^T, \tag{1}$$

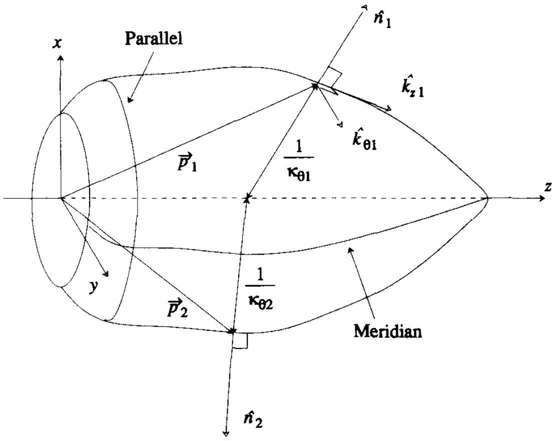


Fig. 1. Surface of revolution.

where $r(\cdot)$ is a function describing the curve rotated about the z axis to generate the surface of revolution.

The unit surface normal directed out of the surface of revolution is calculated as [14] (where, for example, $\vec{x}_\theta = \partial\vec{x}/\partial\theta$).

$$\hat{n} = \frac{\vec{x}_\theta \times \vec{x}_z}{\|\vec{x}_\theta \times \vec{x}_z\|} = \frac{1}{\sqrt{r'^2(z)+1}}\begin{bmatrix} \cos\theta \\ \sin\theta \\ -r'(z) \end{bmatrix}. \tag{2}$$

The two principal curvature directions are

$$\hat{k}_z = \frac{1}{\sqrt{r'^2(z)+1}}\begin{bmatrix} r'(z)\cos\theta \\ r'(z)\sin\theta \\ 1 \end{bmatrix} \tag{3}$$

and

$$\hat{k}_\theta = \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix}. \tag{4}$$

We have used the notation $\hat{k}_\theta$ and $\hat{k}_z$ since the directions are simply $\vec{x}_\theta$ and $\vec{x}_z$, respectively. Although it is possible to determine κ_θ and κ_z from our sensor, the equations are not included here since they are not used to determine the axis.

The lines of curvature that have $\hat{k}_\theta$ as tangent vectors are *parallels*, and *meridians* have $\hat{k}_z$ tangent vectors. Fig. 1 illustrates these curves and the parameters. Notice that $1/\kappa_\theta$ is the distance from the surface to the axis measured normal to the surface (not normal to the axis) as one might first assume.

C. Finding a Plane Through the Axis

Given several contacts on an unknown surface of revolution, we want to use contact information at these points to determine the axis of the surface of revolution. We find a plane that includes the axis by constructing

$$\vec{p} + s\hat{n} + t\hat{k}_z, \tag{5}$$

where s and t are free variables that parameterize the plane. A more useful representation is in the form of the plane equation

$$ax + by + cz = d, \tag{6}$$

where (a, b, c) are the components of a unit vector perpendicular to the plane, and d is the perpendicular distance to the plane from the origin. First, notice that $\hat{k}_\theta$ is a unit vector perpendicular to the plane; thus, we have $a = -\sin\theta$, $b = \cos\theta$, and $c = 0$. Now, substitute (x,

y, z) from (1) into the plane equation (6) to get $d = 0$. Therefore, the plane is given by

$$(-\sin\theta)x + (\cos\theta)y = 0. \tag{7}$$

Clearly, any point on the axis of the surface of revolution (z axis) satisfies this equation. Equation (7) can be used in principle to determine the axis of a surface of revolution from several contacts.

D. *Using the Plane to Determine the Axis*

We now look at how the plane through the axis can determine the axis. An intuitive explanation of the method is the following: From (5), $\hat{k}_z$, along with $\hat{n}$ and $\vec{p}$, determine a plane that includes the axis (equivalently, this plane is given by $\{\vec{p}, \hat{k}_\theta\}$ from (7)). The intersection of two of these planes is a line that is the axis of the surface of revolution. Note that at least three contacts are actually needed to determine the axis. The reason is that *a priori*, we do not know the correct labeling of the curvature data (i.e., we have no way of distinguishing $\hat{k}_\theta$ from $\hat{k}_z$). In order to label or *match* the two directions, we need a third contact so that we can find a common line generated by all pairs of contacts. In this section, we consider only two contacts by assuming that the principal directions are correctly matched. In Section III, we solve for conditions (singular configurations) on the contact placement that will cause incorrect matching.

The axis is determined by the intersection of planes defined by $\{\vec{p}, \hat{k}_\theta\}$. From (7), the two planes can be written as

$$\begin{aligned}(-\sin\theta_1)x + (\cos\theta_1)y &= 0\\ (-\sin\theta_2)x + (\cos\theta_2)y &= 0.\end{aligned} \tag{8}$$

We can represent the intersection line as a point plus a direction multiplied by a free parameter s. Since both planes pass through the origin, $\vec{0}$ will be our point. The direction of intersection is obtained from

$$\begin{bmatrix}-\sin\theta_1\\ \cos\theta_1\\ 0\end{bmatrix} \times \begin{bmatrix}-\sin\theta_2\\ \cos\theta_2\\ 0\end{bmatrix} = \begin{bmatrix}0\\ 0\\ \sin(\theta_2-\theta_1)\end{bmatrix}. \tag{9}$$

Therefore, the line of intersection can be simply written as

$$s\begin{bmatrix}0\\ 0\\ \sin(\theta_2-\theta_1)\end{bmatrix}. \tag{10}$$

The necessary and sufficient condition for being able to determine the axis is thus

$$\sin(\theta_2-\theta_1) \neq 0. \tag{11}$$

III. The Matching Problem

Because we have no way initially of distinguishing $\hat{k}_\theta$ from $\hat{k}_z$, we must consecutively assume particular labels for curvature directions and then check for global consistency with the constraints of the surface of revolution class. In this section, we must consider the intersection of three planes, where each plane is determined from a contact (i.e., $\{\vec{p}, \hat{k}\}$, where $\hat{k}$ is one of the principal curvature directions). Since each contact defines two candidate planes (the desired plane, which is perpendicular to $\hat{k}_\theta$, and the undesired plane, which is perpendicular to $\hat{k}_z$), there will be a total of $2^3 = 8$ different intersections. We show that, assuming contacts are not in particular singular configurations (and assuming perfect data), one of the intersection combinations is a line (which is the axis), whereas the other intersection combinations are not lines. A singular configuration implies that the axis cannot be determined uniquely or, in some cases, that it cannot be determined.

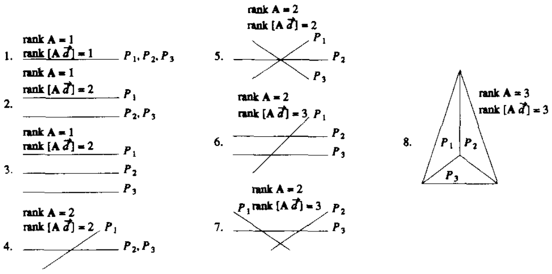


Fig. 2. Possible intersections of three planes.

A. *Intersections of Three Planes*

Our procedure of analysis will be the following: First, we look at the possibilities for the intersections of three correctly matched planes (i.e., planes defined by $\{\vec{p}, \hat{k}_\theta\}$ at each contact). Then, we derive conditions under which a combination of incorrectly and correctly matched planes could also yield these same types of intersections. These, in addition to intersections of correctly matched planes that do not intersect in lines, will be our singular configurations since in these cases, there will be no way to determine the axis of the surface of revolution.

The equations for three planes can be written as

$$\begin{bmatrix}a_1 & b_1 & c_1\\ a_2 & b_2 & c_2\\ a_3 & b_3 & c_3\end{bmatrix}\begin{bmatrix}x\\ y\\ z\end{bmatrix} = \begin{bmatrix}d_1\\ d_2\\ d_3\end{bmatrix} \tag{12}$$

or

$$\mathbf{A}\vec{x} = \vec{d}. \tag{13}$$

Three arbitrary planes in space will intersect in one of eight ways (see Fig. 2, noting that in cases 1 through 7, the planes are viewed edge on). The rank of **A** indicates independence of the plane surface normals. For example, if rank **A** = 1, all three planes are parallel. If rank [**A** $\vec{d}$] $\neq$ rank **A**, then the set of (12) is inconsistent, and there is no set of points common to all three planes, as in cases 2, 3, 6, and 7 of Fig. 2. Using a general position-type argument (and ideal data assumption), the set of planes defined by $\{\vec{p}_i, \hat{k}_{i\theta}\}$ (i = 1, 2, 3) would intersect on the z axis in a line as in case 5.

There are two types of singular configurations: the first where an incorrectly matched set of planes, for example, using planes determined by $\hat{k}_{z1}$, $\hat{k}_{\theta 2}$, $\hat{k}_{z3}$, gives a case 5 type intersection. The second type of singular configuration is where the general position assumption is violated, and correctly matched planes are dependent and intersect as cases 1 and 4.

As an example of a singular configuration, consider the case where two contacts are on the same meridian, but the third is on some other meridian (not 180° from the meridian of the first two contacts). The correctly matched planes associated with the first two contacts are identical, and thus, the intersection of either the correctly or incorrectly matched plane associated with the third contact produces a line intersection. In one case, the true axis is obtained, whereas in the other case, a false axis is obtained. From the available data, it is not possible to determine the axis that is correct.

B. *Intersection of Correctly Matched Planes*

Consider three planes corresponding to correct matches (see (7)):

$$\begin{aligned}P_1:&\quad (-\sin\theta_1)x + (\cos\theta_1)y = 0\\ P_2:&\quad (-\sin\theta_2)x + (\cos\theta_2)y = 0\\ P_3:&\quad (-\sin\theta_3)x + (\cos\theta_3)y = 0.\end{aligned} \tag{14}$$

TABLE I
SUMMARY OF UNIQUE SINGULAR CONFIGURATIONS

Condition(s)	Rank	Correct Planes
$\sin(\theta_1 - \theta_2) = 0$	2	1, 2
$r'(z_2)\cos(\theta_2 - \theta_1) = r'(z_3)\cos(\theta_3 - \theta_1)$ and $r(z_2)r'(z_2) + z_2 = r(z_3)r'(z_3) + z_3$	2	1
$r'(z_1),\ r'(z_2),\ r'(z_3) \neq 0$ and $\sin(\theta_1 - \theta_2) \neq 0$ and other conditions, see [3]	2	—

By using Gaussian elimination, we can determine the rank of **A** and $[\mathbf{A}\ \vec{d}]$.

$$[\mathbf{A}\ \vec{d}] \rightarrow \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & \sin(\theta_1 - \theta_2) & 0 & 0 \\ 0 & \sin(\theta_1 - \theta_3) & 0 & 0 \end{bmatrix}$$

For nonsingular contact configurations, rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 2. Thus, the planes intersect as in cases 4 and 5 of Fig. 2. If $\sin(\theta_1 - \theta_2) = \sin(\theta_1 - \theta_3) = 0$, then all three planes are coincident as in case 1 of Fig. 2 (rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 1). This singular positioning of contacts occurs when all contacts are on the same meridian, or two of the contacts are on the same meridian, and the third is on a meridian displaced 180° from the first.

If all three planes are coincident as in case 1, sufficient information is not available to determine the axis. Only those combinations of candidate planes that result in a case 4 or case 5 plane intersection will be sufficient. Thus, only combinations of planes with rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 2 need to be considered as potential correct matches. In the next section, we show conditions under which incorrectly matched planes could have rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 2.

C. Singular Configuration Summary

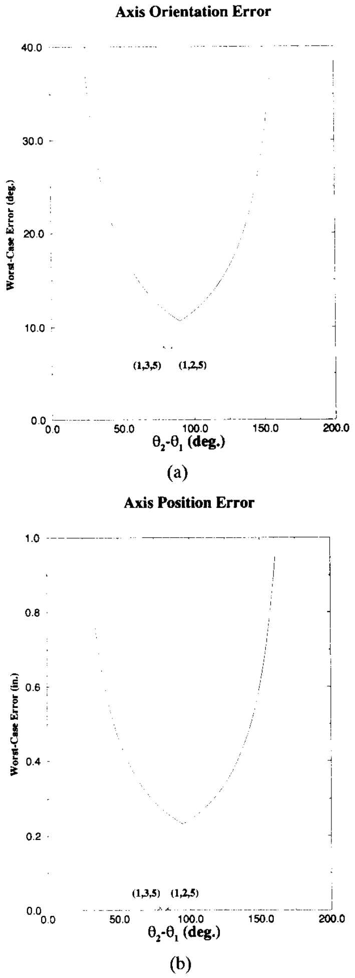
For an arbitrary set of three contacts, there are three possibilities as listed below. Possibilities 2 and 3 are considered singular configurations.

1) Of the eight possible matchings of curvature directions, only one match has rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 2. This match then determines the axis.
2) Of the eight possible matchings, more than one match has rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 2. There will be multiple possibilities for the surface of revolution axis; however, the correct axis will be among the possibilities. These false matches appear when the contact and shape properties meet the conditions in Table I. Details are given in [3]. The **Rank** column contains rank **A** = rank $[\mathbf{A}\ \vec{d}]$, whereas the **Correct Planes** column identifies the correctly matched planes.
3) Of the eight possibilities, at least one has rank **A** = rank $[\mathbf{A}\ \vec{d}]$ = 1.

The most common type of configuration that prevents us from performing the matching consists of contacts on the same meridian or meridians separated by 180°. In addition, a set of three contacts is singular if any of the three is an umbilical point. This is true if $-r(z)r''(z) = r'^2(z) + 1$.

IV. ERROR ANALYSIS

In this section, we consider the effect of errors in the sensed parameters $\{\vec{p}, \hat{n}, \hat{k}_\theta, \hat{k}_z\}$ on the calculated position and orientation of the surface of revolution axis. Specifically, we will consider the axis position error, which is defined to be the minimum distance between the true axis and the calculated axis, and the axis orientation error, which is the angle between the true axis and the calculated axis.



(a)

(b)

Fig. 3. Worst-case axis error.

We take two distinct approaches: The first approach gives worst-case bounds on the position and orientation errors as functions of sensor data obtained from two contacts. It gives a clear (but perhaps simplistic) picture of how errors propagate. The errors predicted by this method can be directly compared with those in our experiments. The second approach gives the standard deviation of the axis position error and angle error as a result of the deviation in the sensed parameters at three contacts. We approximate our calculations of the axis position error and angle error by their linear parts to obtain these estimates. The predicted error cannot be directly compared with our experimental results; however, this method suggests what could be achieved after some work to reduce the large sensor errors.

A. Worst-Case Error Bounds

With correctly matched principal curvature directions, it is possible to calculate the position and orientation of a surface of revolution axis from the sensor data at just two contacts. Considering the sensor data at just two contacts enables us to use simple, intuitive geometry to calculate the resulting error on the estimated axis. This is appealing since it also gives a picture of how errors propagate. Since

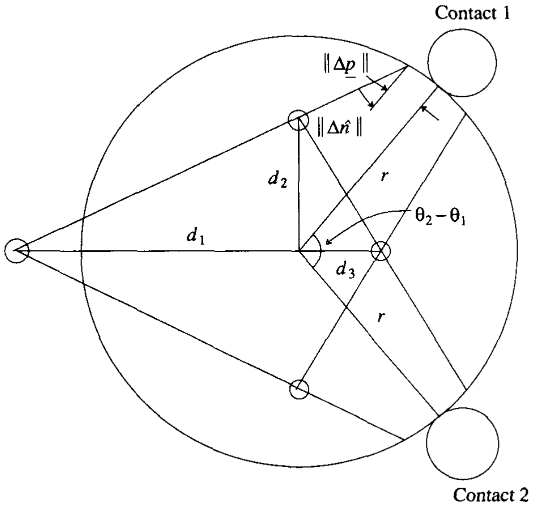


Fig. 4. Geometry for axis position error.

it handles large errors in sensor data (unlike the approach of the next subsection), the results obtained can be directly compared with our experimental results that, unfortunately, were plagued by large sensor errors. In addition, the worst-case error for two contacts is an upper bound on the error for the three contact case.

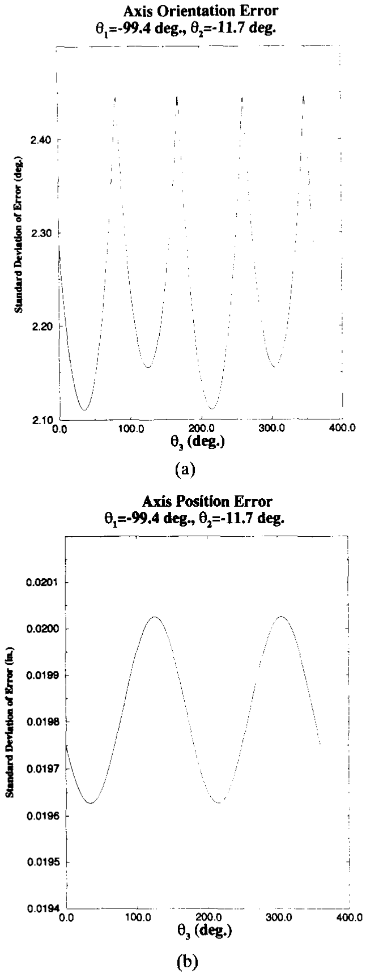
We write the errors in the sensed parameters as $\|\Delta\vec{p}\,\|$, $\|\Delta\hat{n}\|$, $\|\Delta\hat{k}_\theta\|$, and $\|\Delta\hat{k}_z\|$. Since all three of the direction vectors (i.e., $\hat{n}$, $\hat{k}_\theta$, and $\hat{k}_z$) are of unit length and since the orientation uncertainty is small, the uncertainty angle in radians is approximately equal to the magnitudes of the error vectors. Therefore, we will introduce a slight abuse of notation by expressing the magnitude of the error vectors in degrees (e.g., $\|\Delta\hat{n}\|\frac{180}{\pi}$ is approximately the uncertainty angle of $\hat{n}$ in degrees).

1) Axis Orientation Error: Recall that the orientation of the axis is computed from the cross product $\hat{k}_{\theta 1} \times \hat{k}_{\theta 2}$ (see (9)). Ellis [7] has determined the bound on this error by using the Gaussian sphere. $\hat{k}_{\theta 1}$ and $\hat{k}_{\theta 2}$ become points on the Gaussian sphere. The set of vectors perpendicular to $\hat{k}_{\theta 1}$ is a great circle, as is the set of vectors perpendicular to $\hat{k}_{\theta 2}$. Their cross product is then represented by the intersection of these two great circles. When there is uncertainty in $\hat{k}_{\theta 1}$ and $\hat{k}_{\theta 2}$, the points representing the vectors become small circles on the sphere, whereas the great circles become bands of uncertainty. The intersection of the two bands is a curvilinear rhombus. From Ellis [7], the uncertainty is given by

$$\sin^{-1}\left[\frac{\sin\|\Delta\hat{k}_\theta\|}{\sin[(\theta_2-\theta_1)/2]}\right],\ 0<\theta_2-\theta_1<90^\circ. \tag{15}$$

By symmetry, we can reflect the graph about $\theta_2-\theta_1=90^\circ$.

The triples of contacts (1, 2, 5) and (1, 3, 5) from our experimental results produced correct matches for a tolerance of 0.005 (see Table II). Since we only want to consider the errors as functions of two contacts, we will ignore contact 1 data in our error analysis. As was mentioned before, the error as a function of two contacts is an upper bound on the error as a function of three contacts. Therefore, for the pairs (2, 5) and (3, 5), we have the upper bound $\|\Delta\hat{k}_\theta\| = 7.5^\circ$. Using this worst-case bound, we plot the worst-case orientation error as a function of the angle between two contacts in Fig. 3. Notice that the minimum error sensitivity occurs when the contacts are 90° apart. We also include the points corresponding to the actual orientation errors for combinations (1, 2, 5) and (1, 3, 5) from our experimental results.



(a)

(b)

Fig. 5. Standard deviation of axis error.

2) Axis Position Error: Recall from Section II that the axis can be determined by the intersection of the planes at two contacts perpendicular to $\hat{k}_{\theta 1}$ and $\hat{k}_{\theta 2}$. To find an upper bound on the minimum distance between the actual axis and the experimentally determined axis using this method, it is sufficient to examine the cross-section plane at the contact where $r(z)$ is largest. By considering the region in the cross section where it is possible for the surface normals to intersect due to the orientation error $\|\Delta\hat{n}\|$ and the contact position error $\|\Delta\vec{p}\,\|$, we can determine the maximum possible error. The geometry is illustrated in Fig. 4. The small circles mark the vertices of a quadrilateral determined by the angle between the contacts $\theta_2-\theta_1$, the surface normal error $\|\Delta\hat{n}\|$, the contact position error $\|\Delta\vec{p}\,\|$, and the radius of the cross section $r(z)$. Due to the errors, it is possible for the results of experimental data to show that the axis is anywhere within the boundaries of the quadrilateral. Therefore, by computing the distance from the center to the vertex that is the furthest away, we obtain bounds on the position error of the axis determined from experimental data. We can see that a bad situation arises when $\theta_2-\theta_1$ is small, for then, d_1 can become arbitrarily large. As $\theta_2-\theta_1$ increases, d_1 becomes shorter; however, d_2 increases. At approximately $\theta_2-\theta_1=90^\circ$, d_2 becomes longer than d_1 and will continue to increase until it becomes undefined slightly before $\theta_2-\theta_1 = 180^\circ$.

TABLE II
CONTACTS FOR COMPREHENSIVE EXPERIMENT

		$\vec{p}$ (in.)	$\hat{n}$	$\hat{k}_z$	$\hat{k}_\theta$
Contact 1	**Est.**	[35.19 25.36 2.76]	[-.28 -.95 0.14]	[-.44 0.26 0.86]	[.85 -.18 0.49]
θ=−99.4°	**Act.**	[35.16 25.28 2.78]	[-.24 -.96 0.14]	[-.35 0.22 0.91]	[.91 -.16 0.39]
z=5.58 in.	**Err.**	.09	2.7°	5.8°	6.2°
Contact 2	**Est.**	[35.05 25.45 3.23]	[-.26 -.94 0.20]	[-.44 0.30 0.85]	[.86 -.13 0.50]
θ=−95.6°	**Act.**	[35.04 25.40 3.25]	[-.20 -.95 0.23]	[-.35 0.29 0.89]	[.91 -.10 0.40]
z=6.07 in.	**Err.**	.05	3.9°	5.7°	6.9°
Contact 3	**Est.**	[34.92 25.62 3.69]	[-.22 -.93 0.30]	[-.46 0.36 0.81]	[.86 -.05 0.51]
θ=−90.2°	**Act.**	[34.91 25.56 3.71]	[-.14 -.94 0.32]	[-.37 0.35 0.86]	[.92 0.0 0.40]
z=6.5 in.	**Err.**	.07	5.0°	5.7°	7.5°
Contact 4	**Est.**	[36.41 25.86 2.61]	[.84 -.31 0.45]	[-.49 -.05 0.87]	[.24 0.95 0.19]
θ=−14.6°	**Act.**	[36.40 25.86 2.60]	[.83 -.25 0.50]	[-.51 0.03 0.86]	[.23 0.97 0.10]
z=4.93 in.	**Err.**	.01	4.4°	4.9°	5.2°
Contact 5	**Est.**	[35.80 25.95 3.52]	[.76 -.18 0.63]	[-.64 -.03 0.77]	[.12 0.98 0.14]
θ=−11.7°	**Act.**	[35.77 25.95 3.50]	[.75 -.19 0.63]	[-.63 0.06 0.77]	[.19 0.98 0.08]
z=6.00 in.	**Err.**	.04	1.2°	5.0°	5.1°
Contact 6	**Est.**	[36.16 25.93 3.07]	[.81 -.25 0.53]	[-.57 -.14 0.81]	[.13 0.96 0.25]
θ=−10.7°	**Act.**	[36.12 25.94 3.04]	[.80 -.18 0.57]	[-.57 0.04 0.82]	[.17 0.98 0.07]
z=5.45 in.	**Err.**	.04	4.4°	10.2°	10.8°
Contact 7	**Est.**	[35.99 26.78 2.88]	[.54 0.73 0.41]	[-.40 -.21 0.89]	[-.74 0.65 -.18]
θ=46.0°	**Act.**	[35.95 26.74 2.85]	[.55 0.71 0.45]	[-.51 -.14 0.85]	[-.66 0.70 -.29]
z=5.34 in.	**Err.**	.06	2.6°	8.2°	8.2°
Contact 8	**Est.**	[35.69 26.70 3.35]	[.49 0.73 0.47]	[-.46 -.24 0.86]	[-.74 0.64 -.22]
θ=46.6°	**Act.**	[35.65 26.65 3.31]	[.50 0.70 0.51]	[-.55 -.19 0.81]	[-.67 0.69 -.29]
z=5.89 in.	**Err.**	.07	2.9°	6.4°	6.5°
Contact 9	**Est.**	[35.60 26.96 2.93]	[.19 0.93 0.31]	[-.23 -.26 0.94]	[-.95 0.25 -.17]
θ=69.3°	**Act.**	[35.59 26.89 2.9]	[.23 0.91 0.34]	[-.46 -.20 0.87]	[-.86 0.35 -.37]
z=5.53 in.	**Err.**	.08	2.9°	13.9°	14.2°
Contact 10	**Est.**	[35.80 27.04 2.48]	[.21 0.95 0.21]	[-.23 -.16 0.96]	[-.95 0.25 -.18]
θ=70.6°	**Act.**	[35.79 26.98 2.46]	[.24 0.93 0.27]	[-.44 -.14 0.89]	[-.87 0.33 -.37]
z=5.05 in.	**Err.**	.06	3.9°	13.0°	13.0°
Contact 11	**Est.**	[35.80 27.04 2.47]	[.21 0.95 0.22]	[-.27 -.16 0.95]	[-.94 0.26 -.22]
θ=71.2°	**Act.**	[35.78 26.99 2.46]	[.23 0.94 0.27]	[-.44 -.14 0.89]	[-.87 0.32 -.38]
z=5.05 in.	**Err.**	.06	3.0°	10.4°	10.4°
Contact 12	**Est.**	[35.79 27.04 2.47]	[.21 0.95 0.22]	[-.29 -.15 0.95]	[-.93 0.26 -.24]
θ=71.2°	**Act.**	[35.78 26.99 2.46]	[.23 0.94 0.26]	[-.44 -.14 0.89]	[-.87 0.32 -.38]
z=5.05 in.	**Err.**	.06	3.0°	9.2°	9.3°

We again use the triples (1, 2, 5) and (1, 3, 5) from our experimental results. Ignoring contact 1, we have the upper bounds

$$\|\Delta\vec{p}\,\| = 0.07 \text{ in.}, \ \|\Delta\hat{n}\| = 5.0^\circ, \ r(z) = 1.0 \text{ in.}$$

We plot the position error as a function of the angle between two contacts in Fig. 3. Notice that the minimum error sensitivity again occurs when the contacts are 90° apart. We also include the points corresponding to the actual position errors for combinations (1, 2, 5) and (1, 3, 5) from our experimental results.

B. Approximate Standard Deviation of Error

In this subsection, we linearize the calculations for the axis position error and direction error and use these approximations to determine the effect of small deviations in sensor data from their ideal values at three contacts. We assume the small deviations have Gaussian distributions. Currently, our sensor returns large biased errors for reasons explained in the next section. However, we believe these errors can eventually be eliminated, and then, a Gaussian distribution would be more appropriate. Thus, the analysis of this subsection indicates the limits of the algorithm rather than the limits of the experimental setup.

We assume that each of the three components of contact position vectors are corrupted by a random variable with zero mean. The directions, on the other hand, are assumed to be corrupted by two random variables with zero mean, which cause angular deviations. We used $\sigma_\theta = 1.7^\circ$, $\sigma_x = 0.0058$ in from [11].

Suppose the position or orientation error is given by a scalar function $f(\vec{x})$, where about a nominal point, $\vec{x}_0$, $f(\vec{x}_0) = 0$. Then, we may approximate this function $f(\vec{x})$ by $Df(\vec{x}_0)(\vec{x}-\vec{x}_0)$. Suppose $\vec{x} - \vec{x}_0$ is a zero-mean random vector and that each component i is independent of the others with variance σ_i^2. Then, the variance of the error is given by $Df(\vec{x}_0)\Sigma^2 Df(\vec{x}_0)^T$, where Σ^2 is the diagonal matrix of variances $\Sigma_{ii}^2 = \sigma_i^2$.

1) Axis Orientation Error: After the matching has been performed, the estimated direction of the surface of revolution axis can be calculated as

$$\frac{(\hat{k}_{\theta 1} \times \hat{k}_{\theta 2}) + (\hat{k}_{\theta 2} \times \hat{k}_{\theta 3}) + (\hat{k}_{\theta 1} \times \hat{k}_{\theta 3})}{\|(\hat{k}_{\theta 1} \times \hat{k}_{\theta 2}) + (\hat{k}_{\theta 2} \times \hat{k}_{\theta 3}) + (\hat{k}_{\theta 1} \times \hat{k}_{\theta 3})\|}. \tag{16}$$

Notice that the directions obtained by taking cross products of pairs of curvature directions are not normalized. Thus, we are actually weighting these directions by the sine of the angle between the curvature directions. This is done because as two curvature directions become closer to being parallel, their cross product becomes less reliable. The angular error is then the arc cosine of the dot product of the estimated and actual axis directions. Fig. 5 shows a plot of the standard deviation of the orientation error as a function of θ_3 if θ_1 and θ_2 correspond to contacts 1 and 5 from Table II.

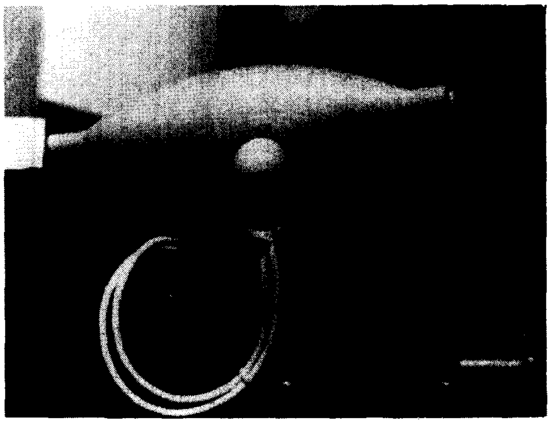


Fig. 6. RobotWorld experimental setup.

2) Axis Position Error: After the matching has been performed, a point on the estimated surface of revolution axis can be calculated as

$$\mathbf{A}^{\dagger}\vec{d} \tag{17}$$

where $\mathbf{A}$ and $\vec{d}$ are as described in Section III, and $\mathbf{A}^{\dagger}$ denotes the pseudo-inverse of $\mathbf{A}$. The pseudo-inverse is used instead of the standard matrix inverse because for correct matches, $\mathbf{A}$ has rank 2 and is singular. To determine the error between the estimated position of the axis and the actual position, we next compute the minimum distance between the estimated and the actual axes. Fig. 5 shows a plot of the standard deviation of the position error as a function of θ_3 if θ_1 and θ_2 correspond to contacts 1 and 5 from Table II.

V. Experimental Results

Experiments were performed on a RobotWorld system. This system consisted of suspended modules with x, y, z, and θ degrees of freedom. Our test object was a wooden surface of revolution cut on a lathe with a sweeping rule modeled by a second-degree polynomial. The object was bolted to the work surface at an angle, and a module equipped with the tactile sensor was commanded to probe the object at various locations (Fig. 6). Recall from Section III that if the contacts do not form a singular configuration, the three planes defined by $\{\vec{p}, \hat{k}_{\theta}\}$, at each contact intersect in a line, and rank $\mathbf{A}$ = rank $[\mathbf{A} \;\; \vec{d}]$ = 2. However, this assumes we have perfect data. With experimental errors, the planes will intersect in a point instead of a line and thus rank $\mathbf{A}$ = rank $[\mathbf{A} \;\; \vec{d}]$ = 3. This problem can be solved by calculating the singular values of $\mathbf{A}$ and $[\mathbf{A} \;\; \vec{d}]$. The number of singular values greater than a preset tolerance determines the rank. Then, we can calculate the estimated axis from equations (16) and (17).

Table I shows the curvature direction information and positions derived from 12 contacts. The curvature direction information for contacts 9 and 10 was badly corrupted. Contacts 11 and 12 were taken at approximately the same point as contact 10 in an attempt to improve the results. Contact 12 had the best overall result of these three; therefore, it was used to represent this position on the object. Notice that groups (1,2,3), (4,5,6), (7,8), and (9,12) have close to the same value for θ. Thus, configurations containing two contacts from any one group will be close to singular.

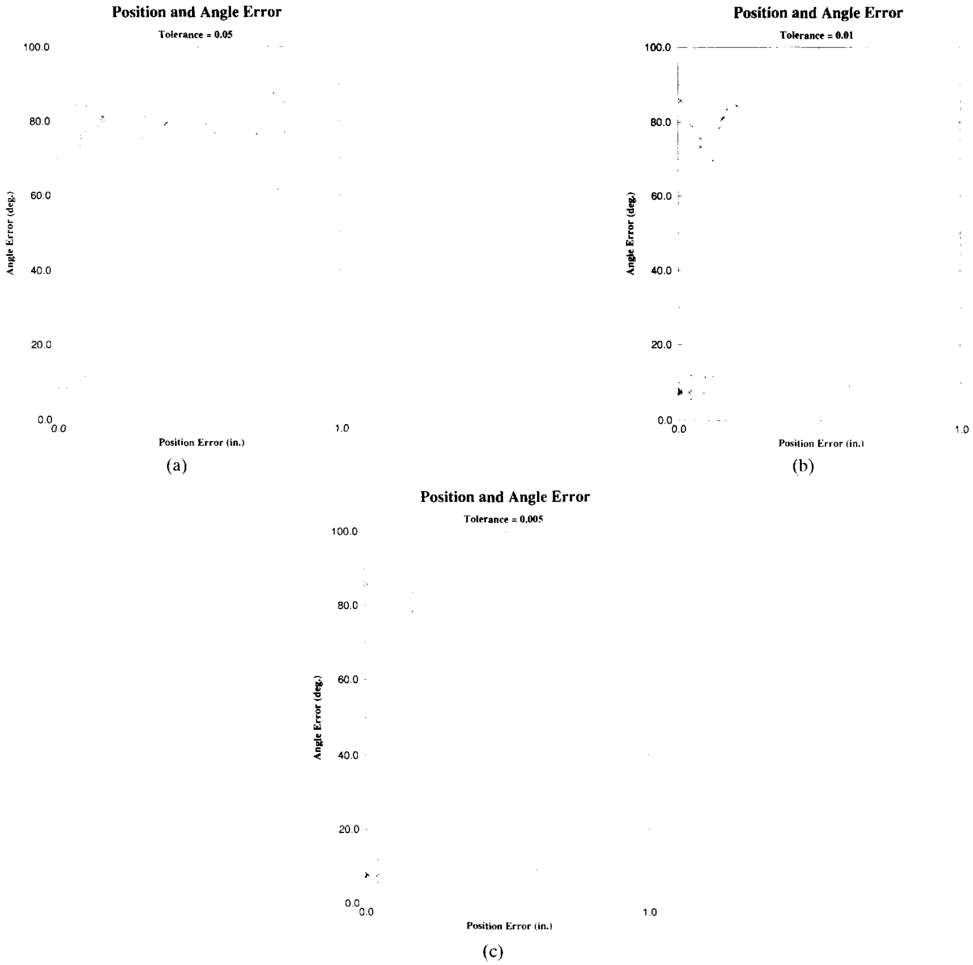
Given ten contacts, there are 120 different combinations of three contact groups. When tested with tolerances of 0.005, 0.01, and 0.05, very few of the combinations were nonsingular. Of those that were nonsingular, many gave a false axis. This poor performance can be attributed to the relatively large orientation errors, the gradual sweeping rule of our object ($r'(z)$ close to zero), and similar values for θ for several groups of contacts. Fig. 7 shows the results of the experiments. Note that there tend to be clusters around 0° and 90° angle error. The cluster around 90° was due to wrongly matched planes (which were almost perpendicular to the axis) intersecting correctly matched planes. For those combinations that gave the correct axis, the angle error was usually below 10°, and the position error was below 0.1 in. In addition, notice the apparent trend: As the tolerance was decreased, the number of nonsingular combinations went down; however, the ratio of correct matches to incorrect matches went up.

Fig. 8 shows the positions of contacts (1,3,6) on our model of the surface of revolution. This particular combination, for tolerances of 0.05 and 0.01, provided a correct matching of the curvature directions and produced a position error of 0.09 in. and an orientation error of 11.7°. The planes corresponding to the contacts are also shown in Fig. 8, where a slice has been cut out of the surface of revolution so that intersection of the planes can be seen. Fig. 9 shows the positions of contacts (5,9,12) on our surface of revolution. For this combination, tolerances of 0.01 and 0.005 produced incorrect labeling of curvature directions. This was because contacts 9 and 12 are close to the same meridian. In addition, the planes associated with contacts 5 and 9 (the plane associated with contact 12 is virtually coincident with the plane associated with contact 9) are in Fig. 9. The false axis is the intersection of these planes. If a singular configuration that gives more than one interpretation for the axis exists, the same data could be from a different surface of revolution. Fig. 10 shows a surface of revolution that could also generate the data for contacts (5,9,12). Error from the tactile sensor during our experiments was greater than in [11]. Some of this error can be attributed to imprecise fixtures, which could cause errors up to a few degrees. Another source of error was in the manner forces were applied to the sensor. Our current model for the sensor assumes that forces are applied normally to the sensor's surface. Unfortunately, with our RobotWorld system, this was not the case since the motions were position controlled rather than force controlled. See [3] for a more complete error discussion.

VI. Conclusion

In general, three contacts with a tactile sensor are sufficient to determine an arbitrary surface of revolution's orientation and location. However, singular configurations that can prevent the axis from being determined uniquely, or possibly from being determined at all, exist. The most common type of configuration that prevents us from performing the matching consists of contacts on the same meridian or meridians separated by 180°.

The bounds on the orientation and position error were calculated as a function of parameters for two contacts. In addition, the deviation of the errors as a function of three contacts was presented. A linear approximation of the error calculation was used to obtain these results. Optimal angular spacing was seen to be 90° between contacts. Experiments showed that our method worked with average position errors around 0.1 in and average angle errors of about 10°.



(a)

(b)

(c)

Fig. 7. Comprehensive experimental results.

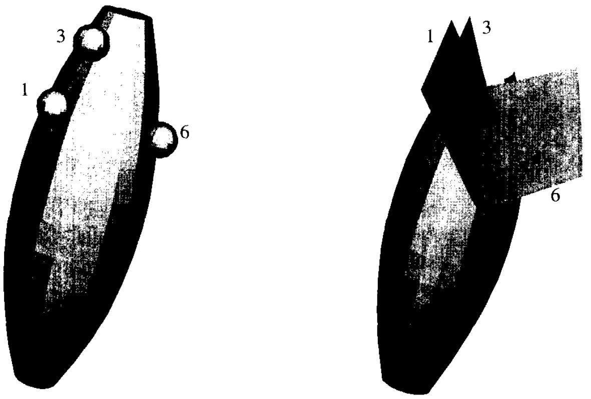


Fig. 8. Positions and planes associated with contacts (1,3,6).

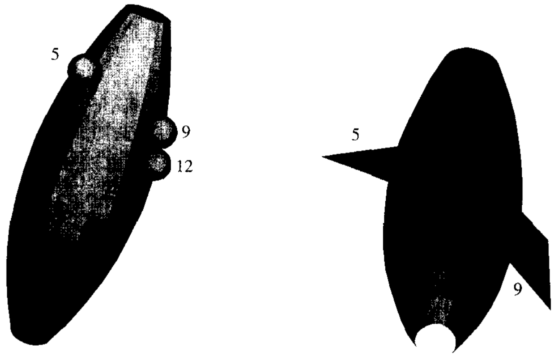


Fig. 9. Positions and planes associated with contacts (5,9,12).

We feel that this method is potentially useful for use with a dextrous hand equipped with tactile sensors. From only the first few contacts, the hand could determine the axis of a surface of revolution, provided that the fingers were not in singular configurations. The experimental results in this paper had relatively high errors, but for tasks that do not require a high degree of precision (such as setting an object on its flat end), the 10° error obtained would be acceptable. In addition, information from subsequent touches could be used to obtain better results by using a least-squares fitting algorithm.

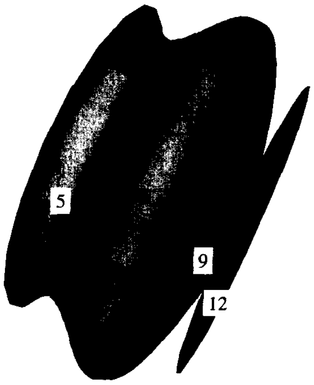


Fig. 10. False surface of revolution associated with contacts (5,9,12).

Acknowledgment

We would like to thank R. Murray and J. Ponce for several helpful discussions and E. Nicolson for a faster converging curvature estimation program.

References

[1] P. K. Allen, "Sensing and describing 3-D structure," in *Proc. IEEE Int. Conf. Robotics Automat.*, 1986, pp. 126–131.

[2] P. K. Allen and P. Michelman, "Acquisition and interpretation of 3-D sensor data from touch," *IEEE Trans. Robotics Automat.*, vol. 6, no. 4, pp. 397–404, Aug. 1990.

[3] M. D. Berkemeier and R. S. Fearing, "Determining the axis of a surface of revolution using tactile sensing," Tech. Rep. UCB/ERL M89/117, UC Berkeley, Electron. Res. Lab., 1989.

[4] ——, "Determining the axis of a surface of revolution using tactile sensing," in *Proc. IEEE Int. Conf. Robotics Automat.*, 1990, pp. 974–979.

[5] M. Brady, J. Ponce, A. Yuille, and H. Asada, "Describing surfaces," in *Robotics Research: The Second International Symposium*(H. Hanafusa and H. Inoue, Eds.). Cambridge, MA: MIT Press, 1985, pp. 5–15.

[6] R. Cole and C. K. Yap, "Shape from probing," *J. Algorithms*, vol. 8, pp. 19–38, 1987.

[7] R. E. Ellis, "Uncertainty estimates for polyhedral object recognition," in *Proc. IEEE Int. Conf. Robotics Automat.*, 1989, pp. 348–353.

[8] ——, "A tactile sensing strategy for model-based object recognition," COINS Tech. Rep., Univ. of Massachusetts, Amherst, 1987.

[9] O. D. Faugeras and M. Hebert, "The representation, recognition, and positioning of 3-D shapes from range data," *Int. J. Robotics Res.*, vol. 5, no. 3, pp. 27–52, 1986.

[10] R. S. Fearing, "Tactile sensing for shape interpretation," in *Dextrous Robot Hands*, S. T. Venkataraman and T. Iberall, Eds. New York, Springer-Verlag, 1990.

[11] R. S. Fearing and T. O. Binford, "Using a cylindrical tactile sensor for determining curvature," *IEEE Trans. Robotics Automat.*, vol. 7, no. 6, pp. 806–817, Dec. 1991.

[12] P. C. Gaston and T. Lozano-Perez, "Tactile recognition and localization using object models: the case of polyhedra on a plane," AI memo, Mass. Inst. of Technol., 1983.

[13] W. E. L. Grimson, "On the recognition of parametrized objects,' in *Robotics Research: The Fourth International Symposium*, R. C. Bolles and B. Roth, Eds. Cambridge, MA: MIT Press, 1988, pp. 245–253.

[14] M. M. Lipschutz, *Differential Geometry*. New York: McGraw-Hill, 1969.

[15] H. Printz, "Finding the orientation of a cone or cylinder," in *Proc. IEEE Comput. Soc. Workshop Comput. Vision*, 1987.

[16] K. Rao and R. Nevatia, "Computing volume descriptions from sparse 3-D data," *Int. J. Comput. Vision*, vol. 2, no. 1, pp. 33–50, June 1988.

On the Calculation of Fractal Features from Images

Susan S. Chen, James M. Keller, *Senior Member, IEEE*,
and Richard M. Crownover

***Abstract*—Fractal Geometry is becoming increasingly more important in the study of image characteristics. There are numerous methods available to estimate parameters from images of fractal surfaces. A very general technique to calculate numerous fractal features involves the estimation of the mass density function by box counting. In this correspondence, we analyze the box-counting method, establish a lower bound for the box size, and indicate how algorithms can be improved to give better estimates of fractal features of images. This provides a theoretical basis for a heuristic approach employed by Pickover and Khorasani.**

***Index Terms*— Box dimension, fractional Brownian motion, fractal geometry, image features, mass density function, scale-insensitive measurements.**

I. Introduction

For segmentation and recognition of regions and objects in natural textured scenes, there is always a need for features that are invariant, or at least insensitive, to scene perturbations, and at the same time provide a good set of descriptive values for the regions. Fractal geometry, which was popularized by Mandelbrot [1], [2], has received increased attention in recent years in this regard. Pentland has shown that the image of a fractal surface is also a fractal, providing a theoretical underpinning for this activity [3]–[5].

There are numerous fractal features that can be generated from an image. These include various moments and texture measures studied by several researchers, including the authors [6]–[12]. However, fractal dimension remains the primary characteristic calculated from image surfaces. Mathematically, it is invariant to changes in scale and can characterize the roughness of the surface.

There are several methods available to estimate features of a fractal surface. Some rely on a specific fractal model, such as fractional Brownian motions, whereas others are applicable to a wider class of fractals. In a previous paper [10], we utilized a method called "box counting" to estimate dimension and lacunarity (texture-related features) of an image. All of these features rely on computing the mass probability density function for the fractal set. In this correspondence, we analyze the box-counting methodology from the standpoint of computing fractal features from real images. We focus on fractal dimension, although the analysis can extend to the other features as well.

It has been shown that the number of boxes of side L needed to cover a fractal set obeys the power law

$$N(L) = KL^{-D},$$

Manuscript received August 19, 1991; revised March 16, 1992. This work was supported by the Air Force Office of Scientific Research Grant AFOSR-87-0226 and Emerson Electric Electronics and Space Division Contract A75786-67. Recommended for acceptance by Associate Editor T. C. Henderson.

S. S. Chen is with Allied Bendix/King Radio Corporation, Olathe, KS, 66062.

J. Keller is with the Department of Electrical and Computer Engineering, University of Missouri-Columbia, Columbia, MO 65211.

R. Crownover is with the Department of Mathematics, University of Missouri-Columbia, Columbia, MO 65211.

IEEE Log Number 9211834.

0162–8828/93$03.00 © 1993 IEEE