Proceedings of the 2006 IEEE/RSJ
International Conference on Intelligent Robots and Systems
October 9 - 15, 2006, Beijing, China

Surface Recognition by Registering Data Curves from Touch

Rinat Ibrayev Yan-Bin Jia
Department of Computer Science
Iowa State University
Ames, IA 50011, USA
rinat, jia@cs.iastate.edu

Abstract

Model-based recognition of an object typically involves matching dense 3D range data. The computational cost is directly affected by the amount of data of which a transformation needs to be found before carrying out the match against a model. This paper investigates recognition using "one-dimensional" data, more specifically, points sampled along three *concurrent* curves on the surface of an object. The introduced method determines the quality of match against a model in two steps. First, the Gaussian and mean curvatures at the curve intersection point are estimated and used in a table lookup to find multiple candidate points on the model that have similar local geometry. Second, starting at each point, local optimization is conducted to search for a possible location of the curve intersection on the model as well as an orientation that leads to a good match of all data points. The best match between the model and the data curves is chosen over the results obtained from all candidate points. The quality of this match is used for comparison against other models. Simulation and experiment have been conducted to validate the recognition approach.

1 Introduction

Object recognition using range data in two and three dimensions often relies on the recovery of rotation and translation parameters through a least-squares formulation. Curves and curved surfaces have been discretized into points, line segments, and planar facets in advance. Correspondences between these primitives in the range data and those on the model are either implied by the parametrization (in the case of a 2D curve) [21], or determined using a search tree of the hypothesis-and-test type [5].

Efficiency and robustness are the main issues in recovering the transformation parameters over a large amount of range data. To ease the computation, hashing over local geometric features such as curvatures and distances (within point tuples or triples) is conducted to narrow down to a small number of transformations before refining them with a least-squares method or a heuristics-based search [3].

Tactile shape sensing does not possess the capability of global recognition. Nevertheless, it has several advantages over range sensing. First, it can identify the relative position and orientation of an object being manipulated by the robot hand. Second, range images are subject to occlusions of the camera, which is not an issue for the touch sensor. Third, mounted on a high-precision robot (such as the Adept Cobra), even a simple joystick sensor can achieve an accuracy of ± 0.1mm on position, which is higher than those of many range sensors.

The inherent local nature of tactile sensing prompts the use of differential invariants in recognizing planar shapes bounded by low-order algebraic curves [10] and simple curved surfaces [12]. Only minimal tactile data are needed for recognition. These differential invariants, however, must have values independent of point locations on a shape. Their derivations involve meticulous algebraic manipulation of primitive invariants including curvatures and torsions. Such derivations are also very shape specific and appear to be very difficult, if not impossible, to generalize to more complex shapes.

The coefficients of a superquadric fit over sparse tactile data [1] can be directly used for shape matching. Nevertheless, the data points need to be distributed all over the surface to generate a good fitting result. This approach also tends to be more effective at reducing the set of possible models than recognizing a specific one.

In this paper, we present a method that recognizes a surface by registering points acquired along three concurrent curve segments residing on the surface. The idea is to compute the best superposition of these curves onto a surface model. We make use of the local geometry at their intersection point p, and combine table lookup with nonlinear optimization.

With tactile data along three concurrent curve segments, we estimate the Gaussian curvature K and mean curvature H at their intersection point p using data points in its neighborhood as described in [11]. All data points are fixed with respect to a local frame at p whose z-axis is aligned with the surface normal and x-y plane with the tangent plane.

A table is constructed in advance for every surface model to store Gaussian and mean curvatures evaluated at discrete points. We look up the table with the pair (K, H) to find a set of estimated locations for p on the model that have similar local geometry. This has reduced the search for the location of p on the surface model to multiple local searches.

Starting at each estimated location for p on the model, a local search walks a path to a point p^* in the neighborhood that induces the best superposition of the data points onto the model.

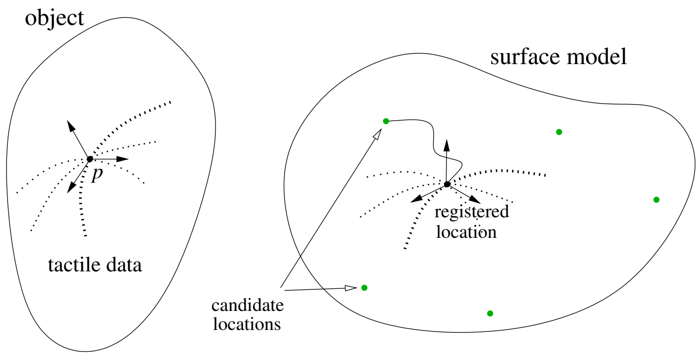


Figure 1: Registering three data curves on a surface. The grey (green) dots represent candidate locations of the curve intersection point p on the surface found in a table lookup. A search path leads from one of these locations to where the curves are "best" superposed onto the surface.

See Figure 1. Matching between the data points and the surface model is done by aligning the two tangent planes, one at p and the other at p^*, and rotating the first one (along with the data curves) to yield the smallest total distance (in the least-squares sense) from all data points to the surface.

The *quality of match* is defined as the minimum aggregated distance over the results from all local searches. The model that yields the best match against the data is then recognized.

Section 2 describes curve registration and model recognition in detail. Section 3 presents simulation results with four classes of shapes. An experiment involving several objects then follows in Section 4. Section 5 summarizes the results and discusses future work.

1.1 Previous Work

There are two primary recognition strategies in model-based vision. The first strategy hinges on the recovery of viewing parameters (and thus the pose) [14]. The second one develops descriptors that are invariant to Euclidean, affine, or projective transformation, or to camera-dependent parameters [15, 22].

Images of 2D objects can be mapped to points in a high-dimensional manifold called the "shape space" [23]. The mapping is invariant to viewing parameters so that recognition reduces to measuring the geodesic distances between these points on the manifold.

In touch sensing, shape recognition has long been based on the notion of "interpretation tree", which represents correspondences between features extracted from the tactile data and those on the model [8, 7, 6]. A volumetric approximation [2] can be built over tactile data to enhance feature selection and prune incompatible models. Constructed over a finite number of features, the interpretation tree method is inherently discrete and its applications have been mostly limited to polyhedral objects.

Geometric hashing and dynamic programming are the two common techniques for matching two plane or space curves. In geometric hashing [13, 9, 16], a curve is discretized, after some smoothing, into equally spaced points. A hash table is usually indexed by the values of some invariants such as curvature, torsion, distance between two points, or angle cosines between their tangents. This table stores the curve model and point location on the model. Voting is done to identify the most promising curve models as well as to set up point correspondences for a least-squares method to find the longest matching segment.

Methods based on dynamic programming [4, 18, 17] tend to segment the curve at special points such as inflections. Consequently, the curve is represented as a "string" of elements corresponding to curve segments. The matching problem then becomes finding the longest common subsequence with low-level comparisons based on features such as total curvature, arc length, etc.

2 Curve Registration

Let S be the surface of an object. Applying the method from [11], we use a touch sensor to sample points along three curves on S, each lying in a different plane. These curves intersect at one point p on the surface. We call them the "data curves" and, in case of no ambiguity, identify them with the data points. The surface normal N and two principal curvatures κ_1 and κ_2 at the intersection point p can be estimated [11]. Thus we have the Gaussian curvature $K = \kappa_1\kappa_2$ and the mean curvature $H = \frac{(\kappa_1+\kappa_2)}{2}$.

The tangent plane at p is orthogonal to the surface normal N. Let us arbitrarily pick two orthogonal tangent vectors $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ at p. Together with N these vectors form a local *data frame* $\mathcal{F}$, with respect to which the curves are fixed. All data points (x_i, y_i, z_i), $1 \le i \le n$, along the three data curves, are converted into local coordinates in $\mathcal{F}$:

$$\begin{pmatrix} x_i' \\ y_i' \\ z_i' \end{pmatrix} = (\boldsymbol{t}_1\ \boldsymbol{t}_2\ N)^T \left(\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} - p \right).$$

To match against a surface model M, we superpose the three data curves onto it by locating their intersection point p on the model. Align p with the found location q and the estimated normal N of S at p with the normal of M at q, and find a rotation of the data points about the coinciding normals that minimize their total distance to M.

2.1 Table Lookup

The surface model M has a preconstructed table which stores the Gaussian and mean curvatures at a set of discretization points. To match the data curves against M, we first look up the table for points whose Gaussian and mean curvatures are close to the estimated pair (K, H) at p. Let $p_1, p_2, \ldots, p_m$ be the points found. We refer to them as the *candidate points*. The next step is to conduct local search, starting at each point p_j, for the best superposition of all data points onto M.

2.2 Local Search

Suppose the surface model M has an implicit form $f(x,y,z) = 0$ and a parameterization $\boldsymbol{\sigma}(u,v)$. Consider the intersection point p of the three data curves coinciding with a point q on M. Let $\boldsymbol{n}$ be the surface normal at q, and $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ be two arbitrarily selected tangent vectors at q such that $\boldsymbol{d}_1 \times \boldsymbol{d}_2 = \boldsymbol{n}$.

We align the xy-plane of the local data frame $\mathcal{F}$, that is, the tangent plane to the data curves, with the tangent plane at q on the model. Let ϕ be the angle of rotation from $\boldsymbol{d}_1$ to the x-axis of $\mathcal{F}$. Every data point (x_i', y_i', z_i') in the frame $\mathcal{F}$ is transformed into some point (x_i'', y_i'', z_i'') in the coordinate system of the surface model:

$$\begin{pmatrix} x_i'' \\ y_i'' \\ z_i'' \end{pmatrix} = (\boldsymbol{d}_1\ \boldsymbol{d}_2\ \boldsymbol{n}) \begin{pmatrix} x_i' \cos\phi - y_i' \sin\phi \\ x_i' \sin\phi + y_i' \cos\phi \\ z_i' \end{pmatrix} + q.$$

This transformation depends on the parameters u and v, which locate the point q on the model, as well as on the angle ϕ. The distance from the transformed data point (x_i'', y_i'', z_i''), $1 \le i \le n$, to M has a first order approximation $\frac{|f(x_i'',y_i'',z_i'')|}{||\nabla f(x_i'',y_i'',z_i'')||}$.

The *registration error* at $q = \boldsymbol{\sigma}(u,v)$ for the data curves is the total distance from all transformed data points to M. It is a function of the parameters u and v and the rotation angle ϕ:

$$E(u,v,\phi) = \sum_{i=1}^{n} \frac{|f(x_i'',y_i'',z_i'')|}{||\nabla f(x_i'',y_i'',z_i'')||}. \quad (1)$$

2.2.1 Error Minimization

Starting at every candidate point $p_j = \boldsymbol{\sigma}(u_j, v_j)$, we search for a local minimum of the function $E(u,v,\phi)$ using the steepest descent method [19, p. 318]. The gradient $\nabla E(u,v,\phi) = (E_u, E_v, E_\phi)$ is obtained by differentiating (1) with respect to u, v, and ϕ, respectively.

The values (u_j, v_j) obtained from the table lookup are used as initial estimates of (u,v), respectively. Let $p_j^{(0)} = p_j$, such that $u_j^{(0)} = u_j$ and $v_j^{(0)} = v_j$. Since $u_j^{(0)}$ and $v_j^{(0)}$ are known, the error E depends on ϕ only. Figure 2 shows how it changes with ϕ when three (synthetic) data curves are superposed onto an ellipsoid. We find $\phi_j^{(0)}$ that minimizes the registration error

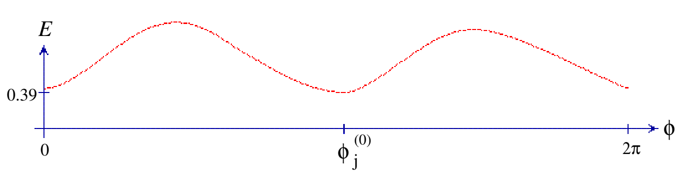


Figure 2: Error (1) of superposing three data curves onto an ellipsoid $(\cos u \sin v, 0.8 \sin u \sin v, 0.5 \cos v)$ over $[0, 2\pi] \times [0, \pi]$. The curves, each consisting of 61 points, were obtained from the ellipsoid where they intersected at $(u,v) = (1.02, 0.69)$. They are now placed at the point $(0.31, 0.63)$ and rotated about its surface normal through an angle ϕ.

and use it as the initial value of ϕ_j.

To compute the initial estimate $\phi_j^{(0)}$, we discretize the domain $[0, 2\pi)$ of ϕ. All local minima of E are bracketed. Within each bracket, bisection is performed to find a local minimum. Then, $\phi_j^{(0)}$ is the angle yielding the smallest of these minima.

2.3 The Registration Algorithm

The input to the algorithm include data points along three curves on a real shape and their intersection point p. It also includes a surface model M with a lookup table recording precomputed Gaussian and mean curvatures at discretization points. The pseudocode of the algorithm is given below.

estimate the normal N of data at p
estimate Gaussian and mean curvatures (K, H) at p
convert all data points into the data frame $\mathcal{F}$
use the lookup table to find candidate points on M
for each candidate point $p_j^{(0)} = \boldsymbol{\sigma}\left(u_j^{(0)}, v_j^{(0)}\right)$
  $\phi_j^{(0)} \leftarrow$ initial estimate of ϕ from bracketing and bisection
  $k \leftarrow 0$
  repeat
    $p_j^{(k+1)} \leftarrow$ steepest descent from $p_j^{(k)}$ on M
      along $-\nabla E\left(u_j^{(k)}, v_j^{(k)}, \phi_j^{(k)}\right)$
    $k \leftarrow k+1$
  until no further minimization is possible
  $p_j^* \leftarrow p_j^{(k-1)}$

Each candidate point p_j will converge to some point p_j^* at which the registration error (1) achieves a local minimum E_j^*, as illustrated in Figure 3. Let p^* be the point among the resulting points $p_1^*, p_2^*, \ldots, p_m^*$ that yields the *minimum registration error*, which is

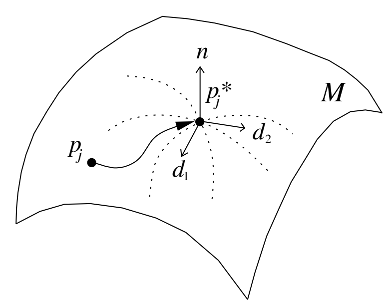


Figure 3: Sliding and rotating three concurrent data curves (as dotted lines) on a surface model to find the best superposition. The point p_j is the initial estimate of the location of the curve intersection while the point p_j^* is the location found through optimization.

$$E_{\min} = \min\{E_1^*, E_2^*, \ldots, E_m^*\}. \quad (2)$$

Then p^* is the estimated location of p on the model M.

2.4 Recognition

Our recognition strategy is based on localization. Suppose we are given an object whose model is known to be from a database. The robot uses a touch sensor to sample data points along three concurrent curves on the object's surface. Estimate Gaussian and mean curvatures at the curve intersection p as described earlier in the section. Then, for each shape model in the database, we run the registration algorithm in Section 2.3 to match the data curves against every model in the database. The model that yields the smallest error $E_{\min}$ defined in (2) is then recognized as the shape of the object.

3 Simulation

Table 1 lists four families of surfaces, in both implicit and parametric forms, which were used in our simulation.

Implicit form	**Parametric form**
$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1$ (ellipsoid)	$(a\cos u\sin v, b\sin u\sin v, c\cos v)$ $(u,v)\in[0,2\pi]\times[0,\pi]$
$\frac{x^2}{a^2}+\frac{y^2}{b^2}=z$ (elliptic paraboloid)	$(av\cos u, bv\sin u, v^2)$ $(u,v)\in[0,2\pi]\times[-1,1]$
$x^3-3xy^2=z$ (monkey saddle)	(u,v,u^3-3uv^2) $(u,v)\in[-1,1]\times[-1,1]$
$x^2y^2=z$ (crossed trough)	(u,v,u^2v^2) $(u,v)\in[-1,1]\times[-1,1]$

Table 1: Four surface families used in the simulation.

3.1 Curve Registration Results

Consider the elliptic paraboloid displayed in Figure 4. We select

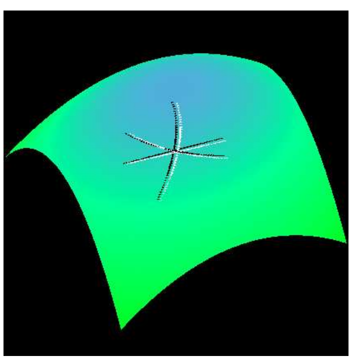


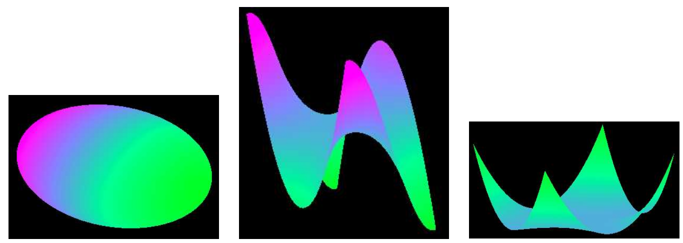
Figure 4: Three data curves (in black color) and their registered locations (in white color) on the surface of an elliptic paraboloid given by $z=\frac{x^2}{1.5^2}+\frac{y^2}{1.1^2}$.

a point p, say, with parameter values $(u,v)=(1.21,0.43)$. Intersect the elliptic paraboloid with three arbitrary planes through p, and generate 61 data points along each of the three intersection curves. Random noises within the range of ± 0.001 are added to the generated data points.

The normal, the Gaussian curvature, and the mean curvature at p are estimated using the method from [11] as $\tilde{N}=(0.1611,0.5764,0.8011)$, $\tilde{K}=0.6082$, and $\tilde{H}=0.7959$.[1] A table lookup finds three candidate points: $p_1=(1.26,0.4)$, $p_2=(1.41,0.4)$, and $p_3=(0.63,0.5)$, all in parameter values. The registration errors (1) at these points are respectively $E_1=0.1625$, $E_2=0.1758$, and $E_3=0.3027$.

Registration starting at the candidate points yields locations $p_1^*=(1.26,0.44)$, $p_2^*=(1.27,0.44)$, and $p_3^*=(1.17,0.44)$. The corresponding registration errors are $E_1^*=0.1211$, $E_2^*=0.1218$, and $E_3^*=0.1256$. The location of p is thus estimated to be $p^*=(1.26,0.44)$, which is very close to its real location $(1.21,0.43)$. The three curves are well registered onto the elliptic paraboloid, as shown in Figure 4.

More registration tests are conducted on the same elliptic paraboloid as well as on three other shapes including an ellipsoid, the monkey saddle, and the crossed trough (see Figure 5). The results are displayed in Figure 6. On each of the



(a) (b) (c)

Figure 5: Three surfaces used in the simulation in addition to the one shown in Figure 4: (a) an ellipsoid with $a=1$, $b=0.8$, and $c=0.5$; (b) the monkey saddle; and (c) the crossed trough.

four surfaces, ten registration instances are performed with randomly generated intersection points p and three cutting planes through each. The real intersection points p of three data curves are drawn as circular dots and their estimated locations p^* as crosses. In the figure every original location p and its estimate p^* lie the closest to each other. Table 2 summarizes the Euclidean distances $\|p-p^*\|$ for the registration instances in Figure 6.

	ellipsoid	elliptic paraboloid	monkey saddle	crossed trough
min	0.0169	0.0148	0.0053	0.0171
max	0.0408	0.0448	0.0433	0.0396
avg	0.0288	0.0297	0.0234	0.0265

Table 2: Minimum, maximum and average Euclidean distance $\|p-p^*\|$ between the real and estimated locations of the curve intersection calculated over the 40 registration instances in Figure 6.

[1] To verify, we also compute their values using the surface equation: $N=(0.1612,0.5827,0.7966)$, $K=0.5915$, and $H=0.7796$, respectively.

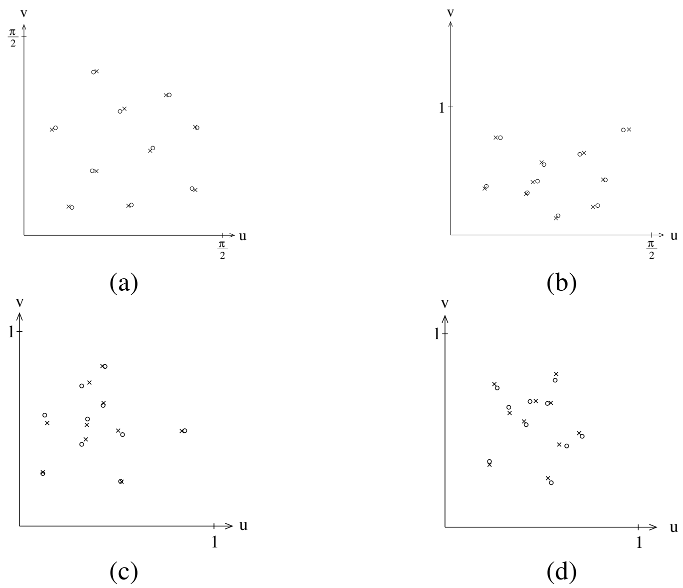


Figure 6: Instances of curve registration on the four shapes displayed in Figures 4 and 5: (a) an ellipsoid; (b) an elliptic paraboloid; (c) the monkey saddle; (d) the crossed trough. In each instance, the intersection point p of three data curves is represented by a circular dot and its estimated location p^* by the closest cross.

3.2 Recognition Tests

As a recognition example we use the same generated data points from the crossed trough, and superpose them onto the four surfaces. The minimum registration error (2) on the crossed trough itself is 0.0720. Using the estimated Gaussian and mean curvature values, we find no candidate curve intersection points on the ellipsoid and elliptic paraboloid after table lookups. The error on the monkey saddle is 0.1867, 159% higher than that on the crossed trough.

For each surface in Figure 6, ten recognition instances, each with a different curve intersection, were carried out. The results

successes	ellipsoid	elliptic paraboloid	monkey saddle	crossed trough
table lookup	2	1	9	0
local search	8	9	1	10

Table 3: Summary of recognition tests, ten on each shape.

are displayed in Table 3. In total, 12 out of 40 tests succeeded after table lookups yielded no candidate points on the wrong models. The other 28 successes involved both table lookups and local optimizations. In these tests, the registration errors on the wrong models exceeded those on the right models by an average of 180%.

4 Experiment

Figure 7 displays the four objects used in our experiment. Data points were obtained using a joystick sensor driven by an Adept robot [11]. By constraining the robot movement in three different planes the sensor sampled points along the corresponding intersection curves with the object. On each curve 41 points including the curve intersection were sampled. So a total of 121 points were acquired on each object.

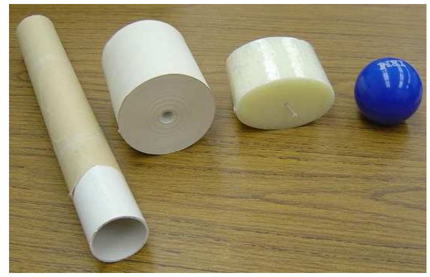


Figure 7: Objects used in the experiment: two regular cylinders with diameters 50.4mm and 94mm, respectively, an elliptic cylinder with semimajor axis 50.8mm and semiminor axis 31.75mm, and a sphere with radius 33mm.

Due to symmetry, curve registration results on the sphere and cylinders were meaningless. But the minimum registration error $E_{\min}$ defined by (2) was still useful for recognition of these shapes. This error was computed for every object.

Each column in Table 4 records the minimum registration error (2), averaged over the number of data points, for the same three data curves onto the four models. The diagonal cells in the

model \ object	cylin. 1	cylin. 2	ell. cylin.	sphere
cylinder 1	**0.033**	0.155	0.446	0.296
cylinder 2	0.182	**0.072**	0.195	1.056
elliptic cylinder	0.230	0.224	**0.156**	0.254
sphere	0.271	0.306	0.929	**0.055**

Table 4: Minimum error (in millimeters) of registering data acquired from the four objects shown in Figure 7 onto their models.

table show the registration errors on the right models, while the non-diagonal cells show the errors on the wrong models. For each object the registration error was the smallest on the right model. As a result, all four objects were correctly recognized.

5 Summary and Future Work

We have presented an approach that recognizes a curved object from a set of surface models based on "one-dimensional" tactile data. More specifically, the data points are sampled along three *concurrent* (but otherwise arbitrary) curves using a touch sensor on the object's surface. The problem of recognition turns into registering these data curves on each model and choosing the one that yields the best matching result.

Finding the rotation and translation typically involves a search in the 6D transformation space. However, our registration task simplifies to finding the location of the intersection point p of the three data curves on the surface model, which is determined by the values of the two surface parameters u and v. At this location, we align the estimated object normal at p with the normal of the model and rotate the data curves about it through an angle ϕ to obtain the best superposition. So there are essentially three degrees of freedom to be determined.

The global minimization over the transformation parameters (u, v, ϕ) is eased with the estimated Gaussian and mean curvatures (K, H) at the point p. We use these two curvatures in a table lookup to locate (discretization) points on the surface model that have similar local geometry to that of p. Search is then carried out in the neighborhood of each point to best superpose the data curves onto the surface model.

The data curves do not need to be planar as long as we can reliably estimate the Gaussian and mean curvatures at the point p. This will facilitate control of the touch sensor in data acquisition. For every shape model, a hash table over (K, H) could be used instead of a 2D array over discrete (u, v) values. This, however, is not expected to improve the recognition time significantly since most of the cost is attributed to local searches.

We have assumed that the surface model has both implicit and parametric forms. However, the method is still applicable if only the implicit form is provided, except that the construction of the lookup table will be more costly. With just a parametric form, the distance from a point to the surface may be found by a search in the parameter space. This will slow down the computation of the registration error.

Experiments with more complex shapes need to be carried out in order to verify the effectiveness of the method. Besides robustness to noise in the position data, we would like to understand how the accuracy of curve registration affects the success rate of recognition. One factor is the "locality" of the data curves — the less local they are, the better they can be registered onto a surface model as we have observed. However, a large number of data points also slows down the computation and is not always necessary for the recognition purpose.

Acknowledgment Support for this research has been provided in part by Iowa State University, and in part by the National Science Foundation through a CAREER award IIS-0133681. We thank Liangchuan Mi and Jiang Tian for generating the tactile data used in our experiments. We also thank the reviewers for their valuable comments.

References

[1] P. K. Allen and K. S. Roberts. Haptic object recognition using a multi-fingered dexterous hand. In *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 342–347, 1989.

[2] G. Beccari, S. Caselli, and F. Zanichelli. Pose-independent recognition of convex objects from sparse tactile data. In *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 3397–3402, 1997.

[3] C. S. Chua and R. Jarvis. 3D free-form surface registration and object recognition. *Int. J. of Comp. Vision*, 17:77–99, 1996.

[4] G. Dudek and J. K. Tsotsos. Shape representation and recognition from curvature. In *Proc. IEEE Conf. Comp. Vision Pattern Recog.*, pp. 35–41, 1991.

[5] O. D. Faugeras and M. Hebert. The representation, recognition, and locating of 3-d objects. *Int. J. Robot. Res.*, 5(3):27–52, 1986.

[6] R. S. Fearing. Tactile sensing for shape interpretation. In S. T. Venkataraman and T. Iberall, editors, *Dextrous Robot Hands*, pp. 209–238. Springer-Verlag, 1990.

[7] P. C. Gaston and T. Lozano-Pérez. Tactile recognition and localization using object models: The case of polyhedra on a plane. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6(3):257–266, 1984.

[8] W. E. L. Grimson and T. Lozano-Pérez. Model-based recognition and localization from sparse range or tactile data. *Int. J. Robot. Res.*, 3(3):3–35, 1984.

[9] A. Guéziec and N. Ayache. Smoothing and matching of 3-D space curves. *Int. J. Robot. Res.*, 12(1):79–104, 1994.

[10] R. Ibrayev and Y.-B. Jia. Semi-differential invariants for tactile recognition of algebraic curves. *Int. J. Robot. Res.*, 24(11):951–969, 2005.

[11] Y.-B. Jia, L. Mi, and J. Tian. Surface patch reconstruction via curve sampling. In *Proc. IEEE Int. Conf. on Robot. Automat.*, pp. 1371–1377, 2006.

[12] D. Keren, E. Rivlin, I. Shimshoni, and I. Weiss. Recognizing 3D objects using tactile sensing and curve invariants. *J. Math. Imaging and Vision*, 12(1):5–23, 2000.

[13] E. Kishon, T. Hastie, and H. Wolfson. 3-D curve matching using splines. *J. Robot. Sys.*, 8(6):723–743, 1991.

[14] D. J. Kriegman and J. Ponce. On recognizing and positioning curved 3-D objects from image contours. *IEEE Trans. Pattern Anal. and Mach. Intell.*, 12(12):1127–1137, 1990.

[15] J. L. Mundy and A. Zisserman. Introduction — towards a new framework for vision. In J. L. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pp. 1–39. The MIT Press, 1992.

[16] T. Pajdla and L. V. Gool. Matching of 3-D curves using semi-differential invariants. In *Proc. IEEE Int. Conf. Comp. Vision*, pp. 390–395, 1995.

[17] E. G. M. Petrakis, A. Diplaros, and E. Milios. Matching and retrieval of distorted and occluded shapes using dynamic programming. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(11):1501–1506, 2002.

[18] A. Pikaz and I. Dinstein. Matching of partially occluded planar curves. *Pattern Recognition*, 28(2):199–209, 1995.

[19] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T Vetterling. *Numerical Recipes in C.* Cambridge University Press, 1988.

[20] A. Pressley. *Elementary Differential Geometry*. Springer-Verlag, 2001.

[21] J. T. Schwartz and M. Sharir. Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. *Int. J. Robot. Res.*, 6(2):29–44, 1987.

[22] I. Weiss. Geometric invariants and object recognition. *Int. J. Comp. Vision*, 10(3):207–231, 1993.

[23] J. Zhang, X. Zhang, H. Krim, and G. G. Walter. Object representation and recognition in shape spaces. *Pattern Recognition*, 36(5):1143–1154, 2003.