Estimating the Friction Parameters of Pushed Objects

Kevin M. Lynch
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In order to plan manipulation of an object by pushing, a robot must have a model of the geometry and the friction properties of the object. This paper presents an approach to estimating the relevant friction parameters by performing experimental pushes and observing the resultant motion. Recognition of objects based on their friction parameters is also explored.

1. Introduction

Pushing is a useful technique for positioning and orienting parts in the plane. In particular, if the object to be placed is too large or heavy for the robot to grasp and carry, pushing provides a natural solution which utilizes the constraint surface inherent in the task. The use of pushing can also reduce the complexity of the manipulator by eliminating the need for a hand or gripper.

Planning of pushing operations depends critically on the parameters describing the frictional contact between the pusher and the object and between the object and the support surface. Most researchers have assumed that the necessary parameters are given [1, 3, 4, 5, 7, 10, 11, 12, 13, 14]. In this paper we describe an approach to estimating these parameters by pushing the object and observing its motion. In particular, we estimate the distribution of support friction, the centroid of the distribution, and the coefficient of friction between the pusher and the sliding object. The estimated parameters are useful for planning pushing operations and recognizing objects which differ only in their friction characteristics.

The basic approach is to push the object with point contact, observe the resulting motion with a vision system, and estimate the friction parameters using the vision data and a model of the mechanics of pushing. Our method is similar to that developed independently by Yoshikawa and Kurisu [16], which additionally utilized a force sensor mounted on the pusher. We will defer a detailed comparison of our methods until Section 7. Related in spirit is the work of Atkeson *et al.* [2] on estimation of the inertial parameters of a grasped object for the purposes of control and object recognition.

In the next section we present the mechanics model used in the parameter estimation procedure. We then describe the estimation algorithm, the experimental setup, and experimental results. We follow with a discussion of the results and open issues. Finally, we provide a brief comparison with previous work and offer some conclusions.

2. Mechanics of pushing

In our analysis we make the following assumptions:

1. Friction forces conform to Coulomb's law. The frictional force at a contact opposes the motion of the contact with a magnitude given by the coefficient of friction multiplied by the normal contact force. If there is no motion at the contact, the frictional force can act in any direction along the tangent with any magnitude less than or equal to the product of the friction coefficient and the normal force.
2. Motions are slow enough that frictional forces dominate inertial forces. This is the *quasi-static* assumption.
3. All motions and pushing forces are in a horizontal plane and gravity acts along the vertical.
4. The support plane is uniform and the object's support friction distribution does not change.

During quasi-static pushing, the force and moment applied by the pusher is always balanced by the support friction. The support friction acting on the object during motion is determined by the motion direction and the support friction distribution $w(\mathbf{p})$, which is the product of the support pressure and the coefficient of friction at each point $\mathbf{p}$ of the support area. In this paper we will approximate the support area by N support points fixed on the object (similar to [16]). In order to estimate the magnitudes $w_j\ (j = 1 \ldots N)$ of the support friction at each point, we must first derive the relationship between the motion of the object and the force and moment of support friction. We will use the following notation (see Figure 1):

N	number of support points
$\mathbf{p}_j$	position of the jth support point
$\mathbf{v}_j$	velocity of the jth support point
$\mathbf{f}_j$	frictional force on the object at the jth support point
w_j	frictional force magnitude at the jth support point, $\lvert\mathbf{f}_j\rvert$
$\mathbf{w}$	N-vector of frictional force magnitudes
$\mathbf{p}_{\mathrm{CF}}$	position of the support friction centroid
$\mathbf{p}_c$	position of the contact between the pusher and the object
μ_c	coefficient of friction at the pushing contact
$\hat{\mathbf{n}}$	unit contact normal vector away from the object
$\hat{\mathbf{t}}$	unit tangent vector 90 degrees counterclockwise of $\hat{\mathbf{n}}$
$\mathbf{f}$	total force of support friction
m_c	total scalar moment of support friction about $\mathbf{p}_c$

All positions, velocities, and forces are expressed in a coordinate frame fixed with respect to the object. Positions, linear velocities, and linear forces are treated as 3-vectors with zero third components.

The frictional force $\mathbf{f}_j$ opposes the direction of motion of the jth support point and has magnitude w_j, which is the normal support force multiplied by the coefficient of friction at the support. The total frictional force and moment about $\mathbf{p}_c$ are given by the following expressions:

$$\begin{aligned} \mathbf{f} &= \sum_{j=1}^{N} \mathbf{f}_j \\ &= -\sum_{j=1}^{N} \frac{\mathbf{v}_j}{|\mathbf{v}_j|} w_j \end{aligned} \tag{1}$$

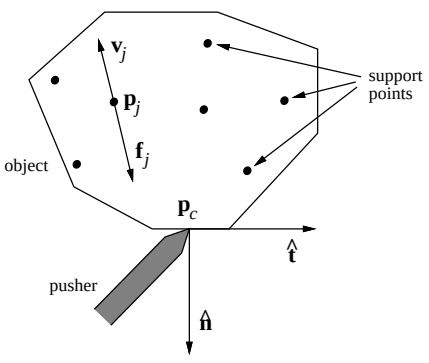

Figure 1: Notation.

$$\begin{aligned} m_c\hat{\mathbf{k}} &= \sum_{j=1}^{N}(\mathbf{p}_j - \mathbf{p}_c)\times\mathbf{f}_j \\ &= -\sum_{j=1}^{N}(\mathbf{p}_j - \mathbf{p}_c)\times\frac{\mathbf{v}_j}{|\mathbf{v}_j|}w_j \end{aligned} \tag{2}$$

where $\hat{\mathbf{k}}$ is the upward-pointing unit vector normal to the support plane. Note that $\mathbf{f}$ and m_c are linear in the $\mathbf{w}$ to be estimated.

During translation of the object, the velocities of all support points are equal. During rotation about a center of rotation at $\mathbf{p}_r$, the velocity of the jth support is given by $\boldsymbol{\omega}\times(\mathbf{p}_j - \mathbf{p}_r)$, where $\boldsymbol{\omega} = (0, 0, \omega)$ and ω is the scalar angular velocity.

The *center of friction* $\mathbf{p}_{\mathrm{CF}}$ is the centroid of the support friction distribution:

$$\mathbf{p}_{\mathrm{CF}} = \frac{\sum_{j=1}^{N} w_j\mathbf{p}_j}{\sum_{j=1}^{N} w_j}. \tag{3}$$

Mason [13] showed that the total support frictional force of a translating object acts through $\mathbf{p}_{\mathrm{CF}}$ and opposes the translation—the support acts like a single point at $\mathbf{p}_{\mathrm{CF}}$. If the coefficient of friction at each support point is the same, $\mathbf{p}_{\mathrm{CF}}$ is simply the center of mass projected to the support plane.

In this paper, the parameters to be estimated are the support friction vector $\mathbf{w}$ for a given set of support points, the center of friction $\mathbf{p}_{\mathrm{CF}}$, and the coefficient of friction μ_c between the pusher and the object. With quasi-static pushing, the coefficient of friction and the normal force at each support point cannot be isolated; only their product w_j can be determined. In the absence of a force sensor, only the relative values of the w_j can be determined, not their magnitudes. The relative values are sufficient for planning pushing operations. The magnitudes could be useful for object recognition.

3. Parameter estimation

A set of M data points is collected experimentally, where each data point consists of the pushing contact location on the object, the contact normal, and the instantaneous velocity direction of the object. Data points for which the center of rotation exactly coincides with a support point are discarded, as the direction of the frictional force at this support is indeterminate. (This never happens with real experimental data.) The data yield a set of constraints on the support friction vector $\mathbf{w}$, described below.

3.1. Moment constraints

The pusher can only apply a force to the object through the contact point. Consequently, the total support frictional force must also pass through the contact point, i.e.,

$$m_c = 0. \tag{4}$$

Given the contact location $\mathbf{p}_c$ and the object motion by the vision system, (4) gives one linear equality constraint on $\mathbf{w}$.

If the pusher contacts a straight edge of the object with uncertainty in the contact location, (4) may be relaxed in order to reflect this uncertainty. The contact location $\mathbf{p}_c$ may be replaced by the interval $[c^-, c^+]$ of possible contact points on the edge. The endpoints of this interval are denoted $\mathbf{p}_{c^-}$ and $\mathbf{p}_{c^+}$, and the frictional moments about these two points are m_{c^-} and m_{c^+}, respectively. The constraint of (4) is replaced by the following two inequality constraints:

$$-m_{c^-} \geq 0 \tag{5}$$

$$m_{c^+} \geq 0. \tag{6}$$

These inequalities constrain the frictional force to pass through or between $\mathbf{p}_{c^-}$ and $\mathbf{p}_{c^+}$ with a nonnegative component along $\hat{\mathbf{n}}$. Note that when there is no uncertainty, these inequality constraints reduce to the equality constraint of (4).

3.2. Force constraints

With a force sensor mounted on the pusher, the support frictional force $\mathbf{f}$ may be directly measured during pushing, as in [16]. Using this sensed force, (1) gives two equality constraints on $\mathbf{w}$.

It is possible to apply force constraints to the relative values of the w_j, if not the magnitudes, even when the pusher is not equipped with a force sensor. The support frictional force must have a nonnegative component along $\hat{\mathbf{n}}$:

$$\mathbf{f}\cdot\hat{\mathbf{n}} \geq 0. \tag{7}$$

This constraint usually provides little or no information for the solution of $\mathbf{w}$. If, however, we know *a priori* that the contact between the pusher and the object is frictionless ($\mu_c = 0$), we know that $\mathbf{f}$ must act along the contact normal:

$$\mathbf{f}\cdot\hat{\mathbf{t}} = 0. \tag{8}$$

More generally, if the coefficient of friction is known to have an upper bound μ_c:

$$\mu_c\mathbf{f}\cdot\hat{\mathbf{n}} - \mathbf{f}\cdot\hat{\mathbf{t}} \geq 0 \tag{9}$$

$$\mu_c\mathbf{f}\cdot\hat{\mathbf{n}} + \mathbf{f}\cdot\hat{\mathbf{t}} \geq 0. \tag{10}$$

This implies that $\mathbf{f}$ must lie within an angle $\tan^{-1}\mu_c$ of $\hat{\mathbf{n}}$, i.e., inside the friction cone. These constraints could be adapted in a straightforward fashion for use with a force sensor that returns only a force angle, possibly with an uncertainty cone.

3.3. Estimation algorithm

Because no force magnitudes are measured, a valid solution to all of the constraints described is $\mathbf{w} = \mathbf{0}$ (zero support friction). To eliminate this solution, we require

$$\sum_{j=1}^{N} w_j = 1. \tag{11}$$

object name	*material*	*dimensions*	$\mathbf{p}_{\mathrm{CF}}$	*support*	*data points*
`delrin_square3`	delrin plastic	7.6 x 7.6	(0, 0)	3 feet	85
`wood_rect3`	wood	12.7 x 6.4	(0, 0)	3 feet	68
`wood_rect4`	wood	12.7 x 6.4	(0, 0)	4 feet	70
`rubber_rect4`	rubber band around wood	12.7 x 6.4	(0, 0)	4 feet	80
`delrin_rect_center`	delrin plastic	8.3 x 5.1	(0, 0)	flat	36
`delrin_rect_right`	delrin plastic	8.3 x 5.1	(0.7, 0)	flat	32

Table 1: Objects used in the experiments.

Another constraint on the solution vector is that all w_j are nonnegative since the coefficient of friction and the support force at each support point must be nonnegative.

The estimated support friction vector should be the $\mathbf{w}$ which best satisfies the linear equality and inequality constraints. Linear programming allows us to use both equality and inequality constraints, and we formulate the estimation of $\mathbf{w}$ as a linear programming problem which minimizes the sum of the error magnitudes of the moment constraints ((4) or (5) and (6)) and the force constraints ((8) or (9) and (10)), while satisfying (7), (11), and the nonnegativity constraints. The residual, or fit error, is the sum of the constraint error magnitudes for the optimal $\mathbf{w}$.

For estimates which are robust to noise, the number of data points M should be much larger than the number of support points N. If moment and force constraints are used in the same linear program, the relative importance of fitting these constraints can be encoded by their weighting in the objective function.

4. Implementation

A camera records the motion of an object as it is pushed by a linear actuator across a glass plate. The linear actuator moves approximately 2 cm/s with a stroke of approximately 5 cm, and the vision system records the position and orientation of the object at about 14 Hz. The object is manually placed in front of the linear actuator before each push.

In order to simplify the vision problem, the pusher and the background are blacked out, and the vision system looks for the bright object in the scene. The position and orientation data for each pushing trajectory are smoothed with an averaging window five samples wide. Data points for the estimation procedure are taken from every fifth frame, resulting in several data points for each push. (A typical push results in three to six data points.) The contact point is determined by intersecting the known pushing line with the perimeter of the object, and the instantaneous velocity direction is found by differencing the smoothed positions of the next and previous frames. Other methods for extracting the data necessary for the estimation procedure are possible, but this approach seemed to yield good results.

In our experiments, all data is acquired off-line by pushing each object at several points around its perimeter. The estimation is then performed using the LINDO optimization package [15]. Alternatively, the estimation can occur on-line by writing the linear program and solving it incrementally (Chvátal [6]) as the data are acquired during pushing.

5. Experimental results

The linear programming estimation procedure was first verified using simulated data [9]. Actual experiments were then performed with six rectangular objects. The origin of the coordinate system

Figure 2: The six objects used in the experiments. The possible support points are shaded (four of the objects have felt feet and two of the objects are flat on the bottom).

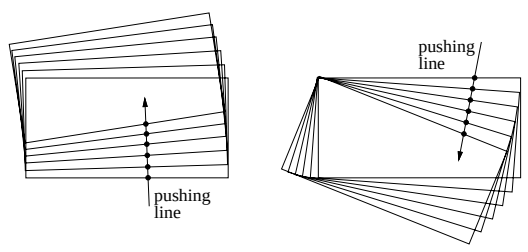

Figure 3: Smoothed trajectories derived from the vision data for two pushes of `rubber_rect4`. The derived contact point for each data point is indicated. There are four observations between each snapshot shown, and the object's velocity at each snapshot is obtained by differencing the position before and after the snapshot. At the beginning of both pushes, the x-axis of `rubber_rect4` points to the right and the y-axis points up.

associated with each rectangle is located at its center, and the x and y axes are aligned with the long and short sides of the rectangle, respectively. A description of each object is given in Table 1.[1]

The supports for the first four objects are small felt feet centered at the listed support points. The last two objects are flat on the bottom, so the support points are unknown. (See Figure 2.) We assume that the coefficient of support friction is uniform in determining the $\mathbf{p}_{\mathrm{CF}}$ listed in Table 1.

`rubber_rect4` is actually `wood_rect4` with a rubber band stretched around its perimeter to increase the coefficient of friction μ_c. `delrin_rect_right` is `delrin_rect_center` with a weight on top in order to offset the center of friction.

Table 1 also shows the number of data points extracted from the vision data for each object. `delrin_square3` was pushed at five

[1] Distances are measured in centimeters, but hereafter distances, forces, and moments will be presented nondimensionally, where the unit distance is a centimeter and the unit force is the magnitude of the support frictional force when the object is translating.

	`delrin_rect_center` *actual* $\mathbf{p}_{CF}$: (0.0, 0.0)		`delrin_rect_right` *actual* $\mathbf{p}_{CF}$: (0.7, 0.0)	
grid size	*estimated* $\mathbf{p}_{CF}$	*percent error*	*estimated* $\mathbf{p}_{CF}$	*percent error*
4 points	(0.41, 1.60)	(4.8%, 31.5%)	(2.13, 0.89)	(17.3%, 17.5%)
9 points	(0.10, -0.28)	(1.2%, 5.5%)	(0.79, -0.48)	(1.1%, 9.4%)
15 points	(-0.08, -0.13)	(0.9%, 2.6%)	(0.71, -0.41)	(0.1%, 8.1%)
24 points	(-0.15, -0.20)	(1.7%, 3.9%)	(0.74, -0.33)	(0.5%, 6.5%)

Table 2: Estimated $\mathbf{p}_{CF}$ for different grid sizes.

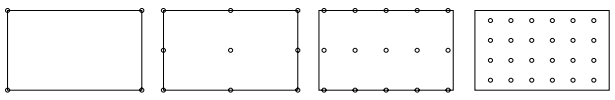

Figure 4: Support grids used in the estimation of $\mathbf{p}_{CF}$ for `delrin_rect_center` and `delrin_rect_right`.

equally spaced points along each edge. `wood_rect3`, `wood_rect4`, and `rubber_rect4` were each pushed at seven points along each long edge. `delrin_rect_center` and `delrin_rect_right` were each pushed at five points along each long edge and two points along each short edge. All pushes began with the pusher moving approximately perpendicular to the pushed edge. The smallest objects, `delrin_rect_center` and `delrin_rect_right`, tended to rotate quickly, resulting in fewer useful data points for each push. Figure 3 illustrates the vision data for two pushing trajectories of `rubber_rect4`.

The experimental results described below utilize only moment constraints. Zero contact uncertainty is assumed unless otherwise noted.

5.1. Estimation

5.1.1. Estimating the center of friction

In the absence of any prior information about the object, the support surface can be approximated by a grid of points inside the perimeter of the object. The estimated center of friction is constrained to lie within the convex hull of these points.

Table 2 shows the estimated $\mathbf{p}_{CF}$ for `delrin_rect_center` and `delrin_rect_right` for grids with 4, 9, 15, and 24 support points, illustrated in Figure 4. The errors in the estimation are given as percentages of the x and y dimensions of the objects. In general, the choice of grid will affect the estimated location of the center of friction. More data justifies the use of more grid points, and more grid points allows a closer approximation to the actual support contacts.

5.1.2. Estimating the support friction distribution

We approximated the support of `wood_rect4` and `rubber_rect4` with a grid of 15 points. The corners of the grid correspond to the actual supports. The estimated $\mathbf{w}$ are shown in Figure 5, where the area of the dot centered at each support is proportional to the support friction magnitude and an "x" indicates zero support friction. The error in the estimated center of friction of both objects is less than 2% of the x and y dimensions of the objects. In addition, the estimated support friction is concentrated at the corners of the grid, where the true support points are. For `rubber_rect4`, only the four corner support points are given weight; the other 11 hypothesized support points did not improve the fit at all, and thus are zero in the optimal solution. The

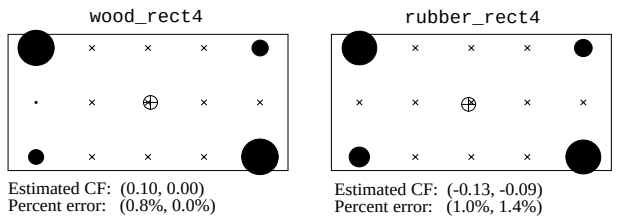

Figure 5: The estimated support friction distributions for `wood_rect4` and `rubber_rect4`. The actual supports are at the corner grid points.

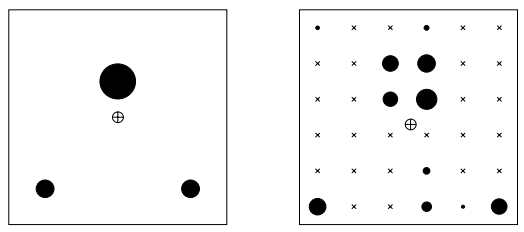

Figure 6: The actual and estimated support friction distributions for `delrin_square3`.

estimated $\mathbf{w}$ for `wood_rect4` gives only a very small weight to one point other than the actual supports.

The support friction distribution of `delrin_square3` is known *a priori*, as only one combination of support forces is consistent with the three supports and the known center of friction. Figure 6 compares the actual support friction distribution with the estimated $\mathbf{w}$ for a 6x6 grid of hypothesized supports. In this example, none of the hypothesized support points coincides with the actual support points. It is interesting to note, however, that the support friction seems to cluster at grid points near the actual supports.

5.1.3. Estimating the coefficient of friction

The estimated $\mathbf{w}$ may be used to obtain a rough estimate of the coefficient of friction μ_c between the pusher and the object. Each data point, along with the estimated $\mathbf{w}$, uniquely specifies a support frictional force. This force must lie within the friction cone of possible forces determined by μ_c. Defining η to be the angle between this line of force and the contact normal $\hat{\mathbf{n}}$, Coulomb's law requires

$$\mu_c \geq \tan \eta. \tag{12}$$

Thus each data point specifies a lower-bound constraint on the value of μ_c. If μ_c is constant along the perimeter of the pushed object, $\mathbf{w}$ is known exactly, and at least one support friction

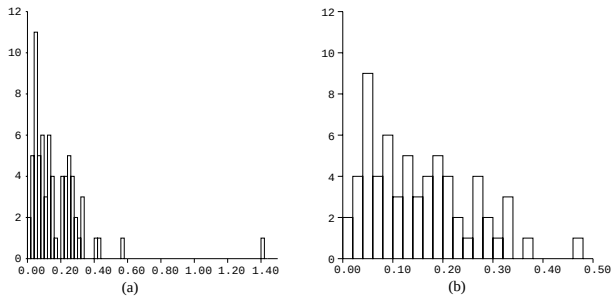

Figure 7: Histograms of lower-bound constraints on μ_c for `wood_rect4` (a) with all data and (b) excluding bad data.

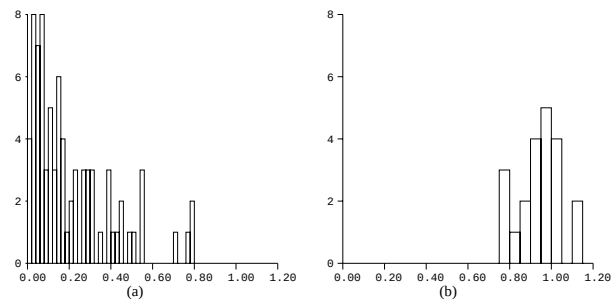

Figure 8: Histograms of lower-bound constraints on μ_c for `rubber_rect4` with (a) original data and (b) test pushes with slipping contact.

force is at the boundary of the friction cone (e.g. slipping pushing contact), then theoretically the highest lower-bound constraint will give the actual value of μ_c.

In practice, however, outliers may be present due to various errors. Figure 7 gives two histograms of the lower-bound constraints on μ_c for `wood_rect4` with $\mathbf{w}$ estimated for four support points hypothesized at the actual supports. A vision error during one of the pushing trajectories was found to be the cause of the distant outlier in Figure 7(a). Removing the six data points of this trajectory and solving again for $\mathbf{w}$ resulted in the histogram of Figure 7(b), which suggests a coefficient of friction of less than 0.5.

Figure 8(a) is the histogram of lower-bound constraints on μ_c for `rubber_rect4` with supports hypothesized at the actual locations. Due to the high coefficient of friction, we did not observe much slipping between the pusher and the object during the initial tests. For this reason, we executed test pushes with sharp pushing angles in order to cause slip and therefore forces at the boundary of the friction cone. Figure 8(b) shows the resulting histogram for these 21 data points and the same $\mathbf{w}$. This histogram should be clustered around μ_c.

Table 3 gives the results of the μ_c estimation for several objects. The estimates are given as μ_c values below which 90% and 100% of the lower-bound constraints lie. The support points used in each estimation are the actual support points, or, in the case of `delrin_rect_center` and `delrin_rect_right`, a grid of 24 supports. These results are compared to μ_c measurements made by the author, which are given as [min, max] intervals measured over a series of trials. Effects such as deformation, nonuniform surfaces, and velocity dependence (static friction vs. kinetic friction) contribute to the noisy measurements.

object	*90% estimate*	*100% estimate*	*author's measurement*
`delrin_rect_center`	0.21	0.34	[0.1, 0.3]
`delrin_rect_right`	0.24	0.36	
`wood_rect3`	0.28	0.41	[0.2, 0.4]
`wood_rect4` (all data)	0.32	1.41	
`wood_rect4` (excluding bad data)	0.30	0.47	
`rubber_rect4`	0.52	0.80	[0.7, 1.1]
`rubber_rect4` (slipping contact only)	range: [0.75, 1.13]		

Table 3: Estimation of μ_c.

5.2. Object recognition

5.2.1. Recognition using the center of friction

To the vision system, `delrin_rect_center`, `delrin_rect_right`, and `delrin_rect_left` (`delrin_rect_right` rotated by 180 degrees) appear identical. With prior knowledge of the location of the center of friction of each object, however, the pushing data may be used to distinguish between these objects. This is a type of model-based object recognition.

A straightforward approach is to estimate the center of friction and determine the model with the closest center of friction. The metric for measuring the distance to a hypothesis should be how well the hypothesis fits the data, however, not the Euclidean distance between the hypothesized and estimated centers of friction.

Our approach is to add to the estimation procedure two equations which require the center of friction to be located at the hypothesized center of friction $\tilde{\mathbf{p}}_{\mathrm{CF}}$:

$$\sum_{j=1}^{N} w_j \mathbf{p}_j = \tilde{\mathbf{p}}_{\mathrm{CF}}. \tag{13}$$

In order to determine which of the object models is correct, we hypothesize the center of friction for each model and perform the estimation. The correct model should minimize the error in the fit.

Tables 4 and 5 give the results for `delrin_rect_center` and `delrin_rect_right`, respectively. Each entry in the tables is the total (moment) error in the fit and the ratio of that error to the error of the best hypothesis. The correct object is found for all grid sizes, but the error ratios of the incorrect hypotheses increase as the grid becomes more refined. We believe this is because as the grid becomes more refined, the error introduced by the grid approximation of the support decreases relative to the error due to an incorrect hypothesis.

By symmetry, we expect that the hypotheses `delrin_rect_left` and `delrin_rect_right` would have approximately the same error when the object is `delrin_rect_center`, and this is supported by the results. Notice also that when the object is `delrin_rect_right`, `delrin_rect_left` appears to be a "worse" hypothesis than `delrin_rect_center`.

5.2.2. Recognition using the support friction distribution

A vision system alone can only determine the orientation of an object up to symmetry in its shape; thus it cannot distinguish between two orientations of `wood_rect3` and four orientations of

	hypothesis		
grid size	`delrin_rect_left`	`delrin_rect_center`	`delrin_rect_right`
4 points	104.14/1.03	101.45/1.00	105.54/1.04
9 points	19.58/1.99	9.86/1.00	19.74/2.00
15 points	17.73/2.61	6.78/1.00	18.85/2.78
24 points	15.29/3.32	4.60/1.00	15.29/3.32

Table 4: Hypothesis matching for `delrin_rect_center` with four grid sizes.

	hypothesis		
grid size	`delrin_rect_left`	`delrin_rect_center`	`delrin_rect_right`
4 points	95.78/1.06	91.21/1.01	90.75/1.00
9 points	30.56/2.27	20.35/1.51	13.46/1.00
15 points	29.62/2.94	18.44/1.83	10.08/1.00
24 points	31.42/5.05	16.54/2.66	6.22/1.00

Table 5: Hypothesis matching for `delrin_rect_right` with four grid sizes.

`delrin_square3`. These objects are also symmetric with respect to their centers of friction. If, however, we have a model of the support friction distribution of an object, we may be able to determine its orientation up to symmetry with respect to the support friction distribution.

For each hypothesized orientation, we use the corresponding hypothesized support points in the estimation procedure. (If the center of friction is also known, this information can be encoded as in Section 5.2.1.) The correct hypothesis should be that which results in the lowest fit error.

Figure 9 shows the results for `wood_rect3` with the correct hypothesis and an incorrect hypothesis of `wood_rect3` rotated by 180 degrees. Figure 9(a) plots the fit error when only the support points are known for each orientation, and Figure 9(b) shows the fit error when the center of friction is also known. Similarly, Figure 10 shows the results for `delrin_square3` with the correct hypothesis and three incorrect orientations. In all cases the true orientation is found, and it is robust to reasonable estimates of the uncertainty in the contact location between the pusher and object.[2] Note that the fit error decreases as the constraints are relaxed to account for contact uncertainty.

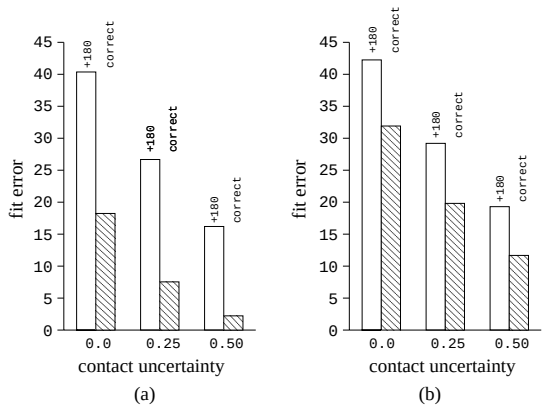

Figure 9: Determining the orientation of `wood_rect3` by finding the minimum-error hypothesis. (a) Hypothesized support locations only. (b) Including the hypothesized center of friction.

It is possible for objects with different support friction distributions to give rise to similar data sets, making it difficult to distinguish these objects by pushing. This emphasizes the need for a "rich" data set consisting of data from pushes at many contact locations and force angles.

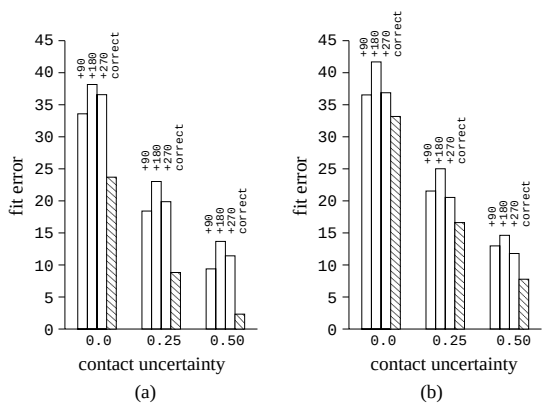

Figure 10: Determining the orientation of `delrin_square3` by finding the minimum-error hypothesis. (a) Hypothesized support locations only. (b) Including the hypothesized center of friction.

6. Discussion

The experimental results show that the approach using moment constraints is fairly successful in estimating the parameters relevant to the quasi-static motion of a pushed object. The estimation procedure finds the correct four support points out of the 15 hypothesized points for `rubber_rect4`, and the error in the estimated $\mathbf{p}_{CF}$ for each object is generally less than a few percent of the object dimensions. Rough estimates of pushing contact friction coefficients are obtained. In addition, the recognition algorithms succeed in distinguishing objects which differ only in their support friction.

[2]The uncertainty is expressed as the half-width of the possible contact interval centered on the nominal contact point.

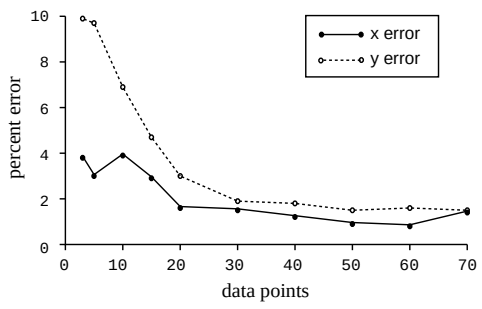

Figure 11: Percent error in the estimated location of $\mathbf{p}_{CF}$ of `rubber_rect4` as a function of the number of data points used in the estimation. Results shown are averaged over 10 random samplings of the 80 data points at each subset size.

Because the estimation procedure uses only vision data, it is applicable in cases where the support friction force is small enough that noise overwhelms force data. A force sensor is necessary if the magnitude of the support friction is of interest; otherwise, we have demonstrated that the friction parameters can be estimated using only vision sensing.

This paper represents a preliminary investigation into the estimation of the friction parameters of pushed objects. Other issues which should be addressed are discussed below.

6.1. Error analysis

Implicit in the parameter estimation and object recognition algorithms is the assumption that the constraint errors from the data are independent. In addition, minimizing the sum of constraint error magnitudes is equivalent to finding the maximum likelihood model for constraint error distributions (around the actual model) of the form $e^{-|z|}$, where $|z|$ is proportional to the magnitude of the constraint error. This error distribution is different than the Gaussian distribution assumption implicit in least-squares fitting. A careful error analysis could allow us to determine the true form of the error distribution. Errors may come from a variety of sources, including vision errors, deviations from the quasi-static model due to inertial forces, approximation of the support area by points, forces out of the plane, and deviations from the Coulomb model of friction due to nonuniform surfaces, velocity dependence, etc. If we can quantify the effects of each of these on the estimation, we might be able to answer questions regarding the confidence of the estimation and recognition results.

To get an idea of the convergence of the estimation as a function of the number of data points, we performed the estimation for `rubber_rect4` using the grid of Figure 5 and random subsets of the 80 data points. Figure 11 shows the average percent error in the estimation of $\mathbf{p}_{CF}$ over 10 trials at each of several subset sizes. The average error decreases little after 20 data points, and the correct four support points are found for nearly all trials with 20 or more data points.[3]

6.2. Using test pushes

The algorithm described here estimates the frictional parameters for an arbitrary set of pushes. If we choose our set of pushes carefully, however, it is possible to obtain a good estimate of $\mathbf{p}_{CF}$ without hypothesizing a set of support points. If the pusher moves

[3]For some trials with three data points, the optimal solution is not unique. All trials with more than three data points had a single optimal solution.

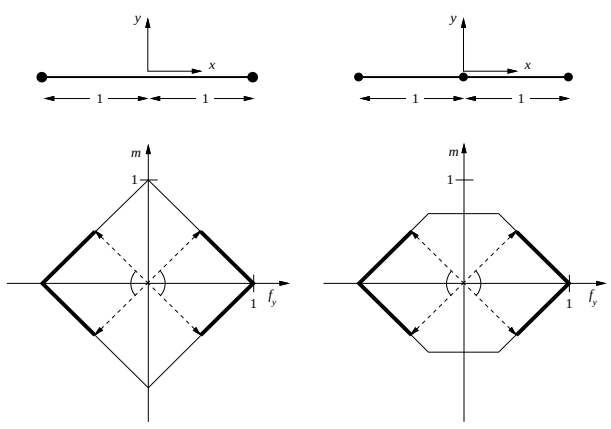

Figure 12: Two rods and the $f_x = 0$ cross-sections of their limit surfaces. The reachable portions of the limit surfaces are bold. These rods behave identically to pushing with zero friction.

opposite to $\hat{\mathbf{n}}$, $\mathbf{p}_{CF}$ must lie on the same side of the pushing line as the center of rotation $\mathbf{p}_r$ [13, theorem 4]. For any push (including nonnormal pushes), $\mathbf{p}_{CF}$ and $\mathbf{p}_r$ must lie on the same side of the *line of motion*—the line along which the contact point on the object moves instantaneously during the push [13, theorem 2]. Thus each push constrains $\mathbf{p}_{CF}$ to lie in a half-plane, and these half-plane constraints limit $\mathbf{p}_{CF}$ to a convex region inside the convex hull of the support area. Pushes should be chosen to maximally reduce the area of this region. If the region collapses due to noise, the estimation could again be turned into a linear programming problem, where the estimated $\mathbf{p}_{CF}$ is that point which minimizes the sum of the errors from the half-plane constraints.

6.3. Distinguishability on the basis of support friction

When is it possible to distinguish objects on the basis of their support friction? The *limit surface* [8] is useful in answering this question. The limit surface is the locus of support frictional forces (f_x, f_y, m) for all possible motions of the object. It is a closed, convex, origin-enclosing surface in the three-dimensional force-moment space. If a slider motion (v_x, v_y, ω) causes a support frictional force $(-f_x, -f_y, -m)$, then (v_x, v_y, ω) is parallel to the outward-pointing normal of the limit surface at (f_x, f_y, m). A support friction distribution uniquely specifies a limit surface. (See [8] for a complete description of the limit surface.)

The set of possible pushing forces during experimentation is limited by the coefficient of friction μ_c and the available contact locations; therefore, the entire limit surface cannot be explored. In order for two objects to be distinguishable by pushing, the reachable portions of their limit surfaces must be sufficiently different, where "sufficiently different" is a function of the noise in the system.

Figure 12 depicts two rods, one with support friction 1/2 at each endpoint (as described in [8]), and one with support friction 1/3 at its endpoints and the center point. These rods can be pushed at points along their length with zero friction, i.e., $f_x = 0$. The $f_x = 0$ cross-sections of the limit surfaces are illustrated in Figure 12, along with the portions of the limit surfaces which can be reached by pushing forces. Although the limit surfaces of the two rods are different, the reachable portions are identical. These two rods behave identically to pushing; therefore, they cannot be distinguished by pushing.

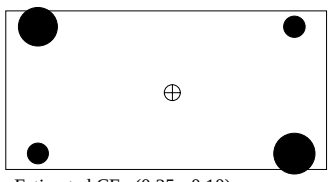

Figure 13: The estimated support friction distribution of `wood_rect4` using least-squares estimation and hypothesized supports at the actual supports.

6.4. Other questions

Given the current data, how should the object be pushed in order to gain the most information about its support friction distribution? How should the hypothesized support points for an unknown object be chosen? How does this affect the estimation of $\mathbf{p}_{CF}$? Is there a better representation of the support area than a grid of support points?

7. Related work

The work of Yoshikawa and Kurisu [16] deserves special mention. They present a strategy for estimating the support friction distribution of an object by performing a series of pushes along a single straight edge. A force sensor measures the force during each push, and a human determines the contact location and the motion of the object by observing the initial and final positions of the object. Each push provides three equality constraints (two force and one moment) for the determination of $\mathbf{w}$. The support is approximated by a grid of points.

The support friction vector $\mathbf{w}$ is solved for using unconstrained least-squares. Thus the quantity to be minimized is the sum of the squared errors, not the sum of the error magnitudes. The important difference, however, is that unconstrained least-squares can result in negative w_j. At these support points, the friction force acts in the same direction as the velocity; in other words, energy is gained, not dissipated, due to friction at these points! Another consequence is that the centroid of the estimated support friction is not necessarily constrained to lie within the convex hull of the support points. These properties of the estimation are unsatisfactory. The experimental results reported in [16] derive from just two data points, resulting in an underconstrained solution to the support friction of the 25 support points.

Figure 13 shows the estimated support friction distribution for `wood_rect4` found by least-squares estimation using singular value decomposition and moment constraints. The four hypothesized support points are at the actual support points. The total magnitude of the support friction is enforced by weighting this constraint heavily. The results for this example are satisfactory; there are no negative w_j. If the support distribution is approximated by a grid of 15 supports as in Figure 5, however, the estimated w_j take on values in the range $[-25.3, 38.2]$ and result in $\mathbf{p}_{CF}$ at $(0.69, -1.80)$, or a percentage error of $(5.4\%, 28.3\%)$. The result found by linear programming is shown in Figure 5.

8. Conclusion

We have presented methods for estimating the frictional parameters of pushed objects and for recognizing objects based on this estimation. Our preliminary experimental results are encouraging. This work is part of an effort to endow robots with the ability to autonomously manipulate objects by pushing. The estimated friction parameters can be used in a mechanics analysis to determine the directions in which the object can be stably pushed with multiple pushing contacts [10]. We can then plan pushing paths which use these pushing directions while avoiding obstacles [11]. While this and most other planning algorithms require only the center of friction and the coefficient of friction between the manipulator and the object, the estimated support friction distribution could be used to further enhance the robot's planning ability.

Acknowledgments

The ideas for this work were developed during discussions with Matt Mason and Tom Mitchell. I thank Eric Krotkov for his advice on this project, and N. R. Natraj, Tamara Abell, Srinivas Akella, and Mark Wheeler for their advice and comments. This work was supported by the Robotics Institute.

References

[1] S. Akella and M. T. Mason. Posing polygonal objects in the plane by pushing. In *IEEE International Conference on Robotics and Automation*, pages 2255–2262, 1992.

[2] C. G. Atkeson, C. H. An, and J. M. Hollerbach. Estimation of inertial parameters of manipulator loads and links. In *International Symposium on Robotics Research*, pages 221–228, 1985.

[3] Z. Balorda. Reducing uncertainty of objects by robot pushing. In *IEEE International Conference on Robotics and Automation*, pages 1051–1056, 1990.

[4] R. C. Brost. Automatic grasp planning in the presence of uncertainty. *International Journal of Robotics Research*, 7(1):3–17, Feb. 1988.

[5] R. C. Brost. Computing the possible rest configurations of two interacting polygons. In *IEEE International Conference on Robotics and Automation*, 1991.

[6] V. Chvátal. *Linear Programming*. W. H. Freeman and Company, 1983.

[7] K. Y. Goldberg. *Stochastic Plans for Robotic Manipulation*. PhD thesis, Carnegie Mellon University, School of Computer Science, Aug. 1990.

[8] S. Goyal, A. Ruina, and J. Papadopoulos. Planar sliding with dry friction. Part 1. Limit surface and moment function. *Wear*, 143:307–330, 1991.

[9] K. M. Lynch. Unpublished results, 1990.

[10] K. M. Lynch. The mechanics of fine manipulation by pushing. In *IEEE International Conference on Robotics and Automation*, pages 2269–2276, 1992.

[11] K. M. Lynch. Planning pushing paths. In *International Conference on Advanced Mechatronics*, 1993.

[12] M. Mani and W. Wilson. A programmable orienting system for flat parts. In *NAMRI XIII*, 1985.

[13] M. T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Research*, 5(3):53–71, Fall 1986.

[14] M. A. Peshkin and A. C. Sanderson. The motion of a pushed, sliding workpiece. *IEEE Journal of Robotics and Automation*, 4(6):569–598, Dec. 1988.

[15] L. E. Schrage. *User's Manual for LINDO.* The Scientific Press, 1981.

[16] T. Yoshikawa and M. Kurisu. Identification of the center of friction from pushing an object by a mobile robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 449–454, 1991.