Tactile Object Recognition and Localization Using Spatially-Varying Appearance

Zachary Pezzementi and Gregory D. Hager

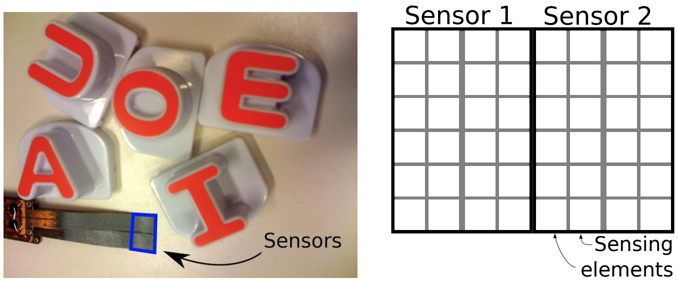
Abstract In this work, we present a new method for doing object recognition using tactile force sensors that makes use of recent work on "tactile appearance" to describe objects by the *spatially-varying appearance* characteristics of their surface texture. The method poses recognition as a localization problem with a discrete component of the state representing object identity, allowing the application of sequential state estimation techniques from the mobile robotics literature. Ideas from geometric hashing approaches are incorporated to enable efficient updating of probabilities over object identity and pose. The method's strong performance is demonstrated experimentally both in simulation and using physical sensors.

1 Introduction

Haptic object recognition has long been a goal of robotics research, and the important role of tactile information has been recognized for decades [3]. Nonetheless, most existing haptic recognition techniques use tactile information only for localizing contact points and estimating local surface normals or curvature to constrain the object geometry [6, 4, 11, 1, 9, 5]. Several researchers have used sequential state estimation techniques to localize the pose of a known object using touch sensing [10, 13, 7]. In recent work [16], we added object identity to the state being estimated in such an approach; this led to a geometry-based object recognition method that used occupancy grid maps as the underlying object representation. This method works very well in 2D, but there are computational challenges in scaling it to 3D, so we wished to incorporate other sources of information.

Only recently (e.g., in [15] and [17]) has the potential to use tactile force sensors to characterize surface textural properties been realized, giving a notion of "tactile appearance" inspired by appearance-based recognition techniques from the com-

Zachary Pezzementi and Gregory D. Hager
Johns Hopkins University, Baltimore, MD, USA, e-mail: {zap,hager}@cs.jhu.edu



(a) Letters and Sensors (b) Sensor Layout

Fig. 1 (a) shows set of raised letters used in the geometry experiments alongside our tactile sensor system, with the sensing area highlighted in blue. (b) shows the layout of sensor elements within that highlighted area. Only the central 6-x-6 element region was used for recognition.

puter vision literature. The work in this paper builds upon the idea of recognition as localization and incorporates appearance information into a new method that characterizes the *spatially-varying appearance* (SVA) characteristics of object surfaces for recognition.

Our approach was inspired in part by geometric hashing techniques, a review of which is provided in [20]. In standard geometric hashing, a basis for 3D space is formed from a set of three 3D points. Our measurements are much more informative than just contact point locations though, so we can take advantage of this extra information to greatly improve efficiency. Though two contact points are not sufficient to define a basis in a 3D space, they can be used to constrain a transformation up to one degree of freedom of uncertainty; our contact points also have associated surface normal estimates, which can in many cases be used to constrain this last degree of freedom. Additionally, we have a tactile image associated with each point, giving it an appearance signature that can be used to distinguish individual points. We therefore extend the geometric hashing algorithm by incorporating this additional information as probabilistic constraints, maintaining the advantage of fast lookup times while reducing space requirements from $O(\binom{N}{3})$ to $O(\binom{N}{2}A^2)$, where A is a factor of the ambiguity of appearance of a surface patch, explained in Sec. 3.2.

1.1 Sensing Tactile Appearance

This work used a capacitive sensor system made by Pressure Profile Systems, consisting of a 6-x-6 array of individual pressure sensors, each of which is square with 2 mm sides, shown in Fig. 1. The physical sensors and a simulation thereof were used in the experiments presented here. Further details on the sensors and the simulator are available in [14].

In [15], the simulation environment from [14] was extended to full robotic exploration using touch sensing, which required the use of a set of surface contact controllers to collect consistent sensor readings. The goal of these controllers is to produce as consistent a tactile image as possible each time the same region of an object surface is sensed, regardless of the angle of approach. The controllers developed for this task control the location and orientation of the sensor up to rotation about the sensor normal (since this rotation can not be controlled for without knowing the object pose), leaving invariance to this final degree of freedom to the appearance representation. In brief, these controllers alternate between pressing the sensor against the object surface under PD control to achieve a target average pressure and reorienting the sensor normal to align with a local estimate of the object surface normal; these steps are iterated until convergence.

Descriptors are extracted from the sensor readings to describe the object surface's local appearance properties. This work uses two descriptors from [15] that account for the aforementioned required invariance to rotation. The first, called moment-normalized (MN), uses spatial moments to determine a major axis for the image; then the image is rotated to bring this major axis to a canonical orientation, and the resulting image is used as a descriptor. The second descriptor, moment-normalized translation-invariant (MNTI), begins with the same steps as MN. Then the result is padded, the 2D Fourier transform is taken, and the magnitudes of the low-frequency coefficients are used as the descriptor. Due to the invariances introduced by discarding phase information, MNTI is more robust to small translations of the sensor with respect to the object surface. More details on the exploration process and the descriptors' formulation and rationale can be found in [15].

2 Spatially Varying Appearance (SVA) and Bayes Filters

A Bayes filtering approach is used to maintain estimates of the unknown object's state, x_t, which consists of the identity and pose of the unknown object, at each time step. During each time step, a command, u_t, is sent to the robot, and a sensor measurement, z_t, is received as a result. Following the notation of [19], the "belief" of the state, $bel(x)$, is updated as

$$\overline{bel}(x_t) = \int \Pr(x_t \mid u_t, x_{t-1}, \mathbf{M}) bel(x_{t-1}) dx_{t-1} \tag{1}$$

$$bel(x_t) = \eta \frac{\Pr(x_t \mid z_t, \mathbf{M})}{\Pr(x_t, \mathbf{M})} \overline{bel}(x_t) \tag{2}$$

$$= \eta \Pr(x_t \mid z_t, \mathbf{M}) \overline{bel}(x_t) \tag{3}$$

where η is a (different) normalization constant in Eqns. 2 and 3. Eqn. 2 is obtained by applying Bayes rule to the standard belief update [19]. In this particular application, the prior on object identity and pose is uniform, so the $\Pr(x_t, \mathbf{M})$ term can be folded into the normalization constant to get Eqn. 3. We consider the explored object to be fixed in space, so the state does not vary over time. Each u_t therefore contributes only kinematic information; since we consider the resolved end effector position to be available in z_t, the commands do not contribute to recognition.

Single sensor readings provide only weak constraints on the possible pose of the object. The pose can be fully constrained by triplets of sensor readings, but this would add extreme space requirements. $\Pr(x_t \mid z_t, \mathbf{M})$ is therefore estimated from the constraints imposed by pairs of sensor readings received so far. The mapping from measurements to states is evaluated as

$$\begin{aligned}\Pr(x_t \mid z_t, \mathbf{M}) &= \Pr(x_t \mid \{\mathrm{pair}(z_i, z_t), \mathrm{pair}(z_t, z_i);\ i = 1,..,t-1\}, \mathbf{M}) \qquad (4)\\ &= \prod_{i=1,\dots,t-1} \Pr(x_t \mid \mathrm{pair}(z_t, z_i), \mathbf{M}) \prod_{i=1,\dots,t-1} \Pr(x_t \mid \mathrm{pair}(z_i, z_t), \mathbf{M}) \qquad (5)\end{aligned}$$

where Eqn. 5 follows from Eqn. 4 by applying Bayes' rule and assuming the constraints of all pairs are conditionally independent given the state. Estimation of the individual probabilities in Eqn. 5 is driven by our maps, $\mathbf{M}$, which contain information about surface patch pairs acquired during training. The training phase consists of collecting a large number of sensor readings that cover the surface of each object to be recognized. The map contains characterizations of pairs of surface patches from each object along with the identity of that object and their location on its surface. During testing, the identity and pose of the unknown object are then constrained by matching observed pairs of surface regions to map pairs, denoted $\mathbf{m}_{[\cdot]}$:

$$\begin{aligned}&\Pr(x_t \mid \mathrm{pair}(z_{a1}, z_{b1}), \mathbf{M}) = \\ &\sum_{\mathbf{m}_{a2}, \mathbf{m}_{b2} \in \mathbf{M}} \underbrace{\Pr\left(x_t \mid \mathrm{match}\left(\mathrm{pair}(z_{a1}, z_{b1}), \mathrm{pair}(\mathbf{m}_{a2}, \mathbf{m}_{b2})\right)\right)}_{\text{match constraint}} \cdot \\ &\underbrace{\Pr\left(\mathrm{match}\left(\mathrm{pair}(z_{a1}, z_{b1}), \mathrm{pair}(\mathbf{m}_{a2}, \mathbf{m}_{b2})\right)\right)}_{\text{match likelihood}} \qquad (6)\end{aligned}$$

First the match likelihood term will be discussed in Sec. 3, then the distributions imposed on the state space by the match constraint term are covered in Sec. 4.

3 Mapping Spatially-Varying Appearance

The purpose of the SVA map is to evaluate $\Pr\left(\mathrm{match}\left(\mathrm{pair}(z_{a1}, z_{b1}), \mathrm{pair}(z_{a2}, z_{b2})\right)\right)$, the probability that two pairs of sensor readings correspond to the same pair of regions on an object surface. This matching, since it makes use of appearance information, depends on the characterization of local appearance to be consistent over a local region; locations nearby a given point on the object surface are assumed to have a similar appearance characterization. This effectively means that the object surface must be sampled densely enough during training for the appearance characterization used to be consistent across regions whose size corresponds to the sampling resolution in the neighborhood.

3.1 Using Surface Patch Pair Statistics

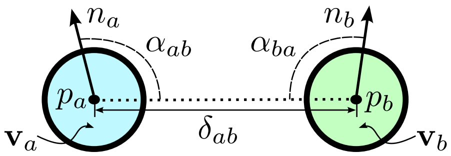


Fig. 2 Illustration of the relevant geometry for dealing with a pair of surface patches, labeled a and b: Each has a centroid $p_{[\cdot]}$, appearance feature $\mathbf{v}_{[\cdot]}$ (visualized by color), and surface normal estimate $n_{[\cdot]}$. δ_{ab} denotes the distance between the patch centroids. The angle between each patch's surface normal and the vector between the patches (dotted line) is marked as $\alpha_{[\cdot]}$.

Each sensor reading consists of a tactile image and a 3D translation and orientation of the sensor in the robot frame. Readings are collected using the controllers described in Sec. 1.1, so rotation about the sensor normal is not controlled for. The geometry of a pair of surface patches a and b is therefore described by the positions of the two patches, p_a and p_b, and their estimated surface normals, n_a and n_b, shown in Fig. 2.

Tactile appearance is described through association with appearance classes in the form of clusters $\mathbf{c}_i$ in the space of appearance descriptors (described in Sec. 1.1). Each cluster corresponds to an appearance class of physical surfaces that gives rise to measurements with certain characteristics picked out by the descriptor being used. We can evaluate the likelihood of a measurement belonging to appearance class i as $\Pr(z_t \mid \mathbf{c}_i)$. $\mathbf{c}(\mathbf{v}_j)$ will be taken to represent the set of those likelihoods for the appearance feature $\mathbf{v}_j$ extracted from measurement z_j.

Unknown objects may be encountered in any pose, so the map of surface patch pairs is indexed by quantities independent of pose. Let $\mathbf{v}_a$ and $\mathbf{v}_b$ be the features describing each patch's appearance. Regardless of the pose of the object, these values and the distance between the points, δ_{ab}, should not change. Pairs are then indexed by $\mathbf{c}(\mathbf{v}_a)$, $\mathbf{c}(\mathbf{v}_b)$, and δ_{ab}, ordered to distinguish pair(z_a, z_b) from pair(z_b, z_a).

3.2 Appearance Class Likelihoods

Now we wish to model $\Pr(z_t \mid \mathbf{c}_i)$, the surface patch measurement likelihoods associated with each appearance class, where appearance classes are defined by clusters in the space of appearance features. A hard clustering method, such as k-means, is excessively restrictive, as the appearance class of many inputs may be legitimately ambiguous, so we opt for a soft clustering approach that only associates a feature with the most likely clusters.

Let the affinity between features $\mathbf{v}_a$ and $\mathbf{v}_b$, $aff(\mathbf{v}_a, \mathbf{v}_b)$ be given by their inner product, $< \mathbf{v}_a, \mathbf{v}_b >$. We use Partitioning Around Medoids [12] to form n_C clusters from the set of all features acquired in training using $aff(\cdot,\cdot)$, each represented by a medoid med_i, such that each feature $\mathbf{v}_j$ is associated with the nearest medoid by its membership m_j. Affinities of members of a cluster to the medoid were assumed to be distributed roughly as a Gaussian with mean 1 and standard deviation, ψ_i, computed for each cluster $\mathbf{c}_i$ as

$$\psi_i = \frac{\sum_j (1 - aff(\mathbf{v}_j, med_i))^2 \operatorname{Ind}(m_j, i)}{\sum_j \operatorname{Ind}(m_j, i)} \tag{7}$$

where $\operatorname{Ind}(i, j)$ is an indicator function equal to 1 if $i = j$ and 0 otherwise. Then the appearance class likelihoods of each feature are given initially by

$$\Pr(\mathbf{v}_j \mid \mathbf{c}_i) = \frac{1}{\sqrt{\pi\psi_i}} exp\left(-\frac{(1 - aff(\mathbf{v}, med_i))^2}{2\psi_i}\right) \tag{8}$$

$$\Pr(\mathbf{c}_i \mid \mathbf{v}) = \eta \frac{\Pr(\mathbf{v}_j \mid \mathbf{c}_i)}{\sum_j \Pr(\mathbf{v}_j \mid \mathbf{c}_j)} \tag{9}$$

where η is a normalization constant. Unlikely matches are then pruned away by setting

$$\text{best}_{i,j} = \max_i \Pr(\mathbf{v}_j \mid \mathbf{c}_i) \tag{10}$$

$$\text{Prune}(\mathbf{v}_j \mid \mathbf{c}_i) = \begin{cases} \Pr(\mathbf{c}_i \mid \mathbf{v}_j) & \Pr(\mathbf{c}_i \mid \mathbf{v}_j) > T_a \text{best}_{i,j} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\Pr(z_j \mid \mathbf{c}_i) = \eta \text{Prune}(\mathbf{v}_j \mid \mathbf{c}_i) \tag{12}$$

In our experiments, T_a was set to 0.75. In the worst case, e.g. if all appearance classes have the same likelihood, this procedure can produce a match to every class, making the appearance ambiguity mentioned in Sec. 1, A, equal n_C in the worst case. In practice, however, many fewer matches are common.

3.3 Matching Distances

Distances were matched using kernelized histograms. The range of possible values was discretized into a set of n_{DB} uniform regions with distance bin centers

$$\text{bin}D_i = \text{minDist} + (\text{maxDist} - \text{minDist}) \frac{i + 0.5}{n_{DB}} \tag{13}$$

and δ_{ab} was associated with the nearest bins through linear interpolation to give degrees of association with each bin, $\Pr(\mathrm{bin}D_i|\delta_{ab})$.

3.4 Putting it all Together

Let *CA* and *CB* be random variables corresponding to the appearance classes of surface patches *a* and *b* respectively for both the prospectively matching pairs and *D* be another random variable corresponding to the distance between points in the pairs. Marginalizing over appearance and distance classes gives

$$\begin{aligned}&\Pr(\mathrm{match}(\mathrm{pair}(z_{a1},z_{b1}),\mathrm{pair}(z_{a2},z_{b2})))\\&\quad=\sum_{i,j,k}\Pr(\mathrm{pair}(z_{a1},z_{b1}),\mathrm{pair}(\mathbf{m}_{a2},\mathbf{m}_{b2}),CA=\mathbf{c}_i,CB=\mathbf{c}_j,D=\mathrm{bin}D_k)\end{aligned} \tag{14}$$

Since the observed measurements are independent of the map measurements given the appearance and distance classes, each of which is independent of the others, this can be manipulated into the form

$$\begin{aligned}&\Pr(\mathrm{match}(\mathrm{pair}(z_{a1},z_{b1}),\mathrm{pair}(\mathbf{m}_{a2},\mathbf{m}_{b2})))\\&\quad=\sum_{i=1}^{n_C}(\Pr(z_{a1}\mid CA=\mathbf{c}_i)\Pr(\mathbf{m}_{a2}\mid CA=\mathbf{c}_i)\Pr(CA=\mathbf{c}_i)\cdot\\&\quad\sum_{j=1}^{n_C}(\Pr(z_{b1}\mid CB=\mathbf{c}_j)\Pr(\mathbf{m}_{b2}\mid CB=\mathbf{c}_j)\Pr(CB=\mathbf{c}_j)\cdot\\&\quad\sum_{k=1}^{n_{DB}}\Pr(\delta_{a1b1}\mid D=\mathrm{bin}D_k)\Pr(\delta_{a2b2}\mid D=\mathrm{bin}D_k)\Pr(D=\mathrm{bin}D_k)))\end{aligned} \tag{15}$$

This provides a way to evaluate the probability of each pair of observed points corresponding to pairs of regions in the object maps. The appearance likelihoods can be computed by evaluating Equation 12, and the distance likelihoods can be obtained from the joint distribution of Section 3.3 as $\Pr(\delta_{ab},\mathrm{bin}D_k)=\Pr(\delta_{ab}\mid\mathrm{bin}D_k)\Pr(\mathrm{bin}D_k)$. In our experiments, the class priors, $\Pr(\mathbf{c}_i)$ for *CA* and *CB* (these distributions are taken to be equal), were assumed to be uniform. The mapping can be efficiently implemented using a hash multi-map, indexed by bin numberings, to support fast lookups without using excessive storage when the space is sparsely covered (particularly, e.g., when there are many appearance classes).

4 Recognition and Localization from SVA Maps

Given a matched pair of surface patches, $\mathrm{pair}(z_{a1},z_{b1})$ and $\mathrm{pair}(z_{a2},z_{b2})$ we now wish to estimate the set of rigid transformations that would align them.

4.1 Initial Alignment of Surface Patch Pairs

We begin with a version of the method of [2] to align 3D point clouds, simplified to the case of two points. Continuing with the notation of Sec. 3, this procedure gives a rotation, R_1, that is effective for aligning the points of contact, p_{a1} with p_{a2} and p_{b1} with p_{b2}, but it leaves rotations about the axis between the points in the pair, $\text{axis}_{a,b} = (p_b - p_a)/||p_b - p_a||^2$, unconstrained. The surface normals associated with each patch are next used to further constrain the aligning transformations.

The normals are limited in their ability to be used in this respect, though, in two ways: Each normal only constrains rotation about $\text{axis}_{a,b}$ if it is not colinear with $\text{axis}_{a,b}$, and the observed sensor surface normals themselves may be unconstrained if the object surface normal in the area is not well-defined, e.g. in the case of an edge or corner. Because of these factors, we have considerably more confidence in the point locations than in the surface normals, so we are comfortable parameterizing the aligning transformation as a rigid transformation based on point locations followed by a rotation about $\text{axis}_{a1,b1}$ by an angle β with uncertainty.

We will first discuss our handling of constraints on the surface normals in Sec. 4.2, then this will be incorporated with the colinearity issue into our full estimate of the axis-angle rotation portion of the transformation with its associated uncertainty in Sec. 4.3.

4.2 Estimating Constraints on Sensor Normals

The surface normal at a surface patch can only be used to constrain rotation if it is itself constrained, so our goal here is to estimate the level of constraint the object surface imposed upon the sensor normal when a reading z_i was taken by looking at the associated tactile image, $\mathbf{I}_i$. Our approach is to infer a rough set of contact points of the sensor with the object surface from sensor elements with non-zero responses. In order for the sensor normal to be well-constrained, the surface should make contact with the sensor in at least three well-separated, non-colinear locations. Algorithm 4.2 quantifies a way of measuring the degree of fulfillment of this requirement, returning $normConf(n_i)$.

A set of 3D contact points are estimated from $\mathbf{I}_i$. Their centroid is subtracted, giving a set of relative positions, which are assembled into a matrix, A, whose singular value decomposition is computed. For edge or point contacts, there should be less than two significant singular values, and in this case the function returns a confidence of zero. Otherwise it returns a value that approaches one as the two largest singular values approach each other, i.e. as the contact type approaches fully planar.

Algorithm 1 Estimate Normal Constraint

```
$pts \leftarrow \emptyset$
$avgPt \leftarrow point3D(0,0,0)$
for all sensor elements $i$ do
    if $val(i) > contactThresh$ then
        $p \leftarrow point3D(getX(i), getY(i), estimateDepth(val(i)))$
        add $p$ to $pts$
        $avgPt \leftarrow avgPt + p$
    end if
end for
if $sizeOf(pts) < 3$ then
    return 0
end if
$avgPt \leftarrow avgPt / sizeOf(pts)$
$r \leftarrow 1$
$A \leftarrow matrix(sizeOf(pts), 3)$
for all points $p$ in $pts$ do
    $rowOf(A, r) \leftarrow p - avgPt$
    $r \leftarrow r + 1$
end for
$S = svd(A)$ {Returns sorted vector of singular values}
return $S[2]/S[1]$ {Ratio of two largest singular values}
```

4.3 Formulating Axis-Angle Uncertainty

Finally, we incorporate the constraint imposed by the normals on about-axis rotation with the uncertainty in the normals themselves to estimate a distribution over possible axis-angle rotations to complete the alignment of the two patch pairs.

First the surface normals associated with the first patch pair, n_{a1} and n_{b1}, and the axis between them, $\text{axis}_{a1,b1}$ are rotated according to the transformation obtained in Sec. 4.1 to be in the same coordinate system as n_{a2} and n_{b2}. Then a projection is computed to project each pair's normals onto the plane normal to $R_1 \text{axis}_{a1,b1}$, giving projected normals $\{pn_{[\cdot]}\}$. Next, a rotation is computed to align these projected normals once again based on the method of [2]:

$$H = pn_{a1} pn_{a2}^T + pn_{b1} pn_{b2}^T \tag{16}$$

$$USV^T = \text{svd}(H) \tag{17}$$

$$R_\beta = VU^T \tag{18}$$

The projection has the effect of scaling each vector by the degree to which it is perpendicular to $\text{axis}_{a1,b1}$ in the rotated space, thereby also scaling its contribution to the least squares error being minimized in the fit. If the determinant of R_β is negative, this generally means S is rank defficient and the sign of one column of U is unconstrained, so R_β is reset to $V\,diag(1,-1)\,U^T$, giving a valid rotation.

Finally, the angle of rotation is extracted from R_β to get $\hat{\beta}$, our estimate of β. The overall confidence in this value is estimated as

$$q_a = ||pn_{a1}||\, normConf(n_{a1})\; ||pn_{a2}||\, normConf(n_{a2}) \tag{19}$$

$$q_b = ||pn_{b1}||\, normConf(n_{b1})\; ||pn_{b2}||\, normConf(n_{b2}) \tag{20}$$

$$alignConf = \frac{q_a + q_b}{2} \tag{21}$$

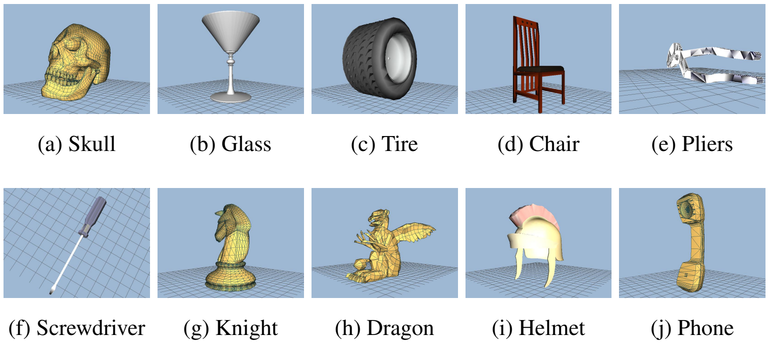
The distribution of possible true values of β is then estimated as a Gaussian with mean $\hat{\beta}$ and standard deviation given by $\sigma_{\text{init}}/alignConf$. In our experiments, σ_{init} was conservatively set to 0.1 radians.

In practice, the distribution over x_t is maintained as a sparse histogram. Except at initialization, when the distribution can be implicitly assumed equal to the prior (e.g., uniform), the likelihood in most bins will be zero. The data structure can be efficiently implemented using a hash map indexed by the histogram bin numbers.

5 Experiments

The approach was tested on 3D models in our simulation environment and on a set of raised letter shapes using our physical sensor system. The simulation experiments are described in Sec. 5.1, then those on physical sensors follow in Sec. 5.2.

5.1 3D Simulation Experiments



(a) Skull (b) Glass (c) Tire (d) Chair (e) Pliers

(f) Screwdriver (g) Knight (h) Dragon (i) Helmet (j) Phone

Fig. 3 The set of models from the Princeton Shape Benchmark [18] used for testing.

To test the algorithm, a set of 10 objects from the Princeton Shape benchmark [18] was used, shown in Fig. 3. The sample objects were selected to span a variety of shapes and local surface characteristics, and all were scaled to the same size.

A set of 1000 sensor readings of each object was collected for training, and a separate 100 readings of each object were collected for testing. All sensor readings were collected by the following process: A location p on the object surface was chosen uniformly at random. The position of the tactile sensor was set to $q = p + dn$, a small distance d away in the direction of the local surface normal, n. The sensor was oriented so that its surface normal was in the direction $-n$ plus a small random perturbation. Then the sensor moved in the direction of the object until the controllers described in Sec. 1.1 converged.

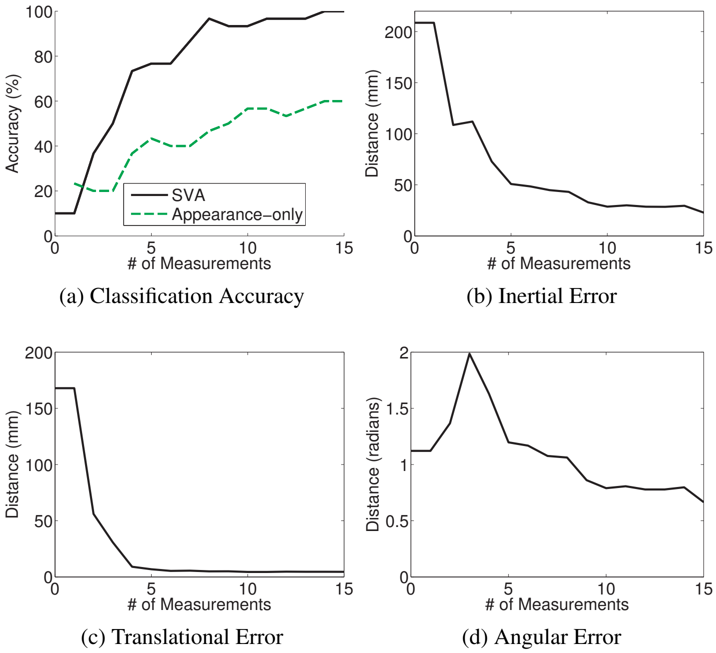
As described in Sec. 3, appearance descriptors were extracted from each sensor reading and these were grouped into 25 clusters using k-medoids, then an SVA map was built of all the objects. The map used 40 bins to discretize the space of inter-patch distances that covered a range from 30 mm to 160 mm. The objects themselves were scaled 80 mm in their largest dimension.

Recognition performance was measured as a function of the number of sensor readings seen so far by averaging results over a number of trials. In each trial, readings were selected uniformly at random from the set of test readings for the selcted unknown object. An object pose was generated and the pose of each sensor reading was transformed according to this unknown pose before it was presented to the recognition algorithm. This pose consisted of an arbitrary 3D rotation and a translation in the range $[-200, 200]$ mm in each direction. One test repetition consisted of one trial of recognition on each object from the set. Performance was averaged over three test repetitions to get the final results shown in Fig. 4. Accuracy of the SVA approach is shown alongside that of the appearance-only approach of [15] for comparison.

The virtual histogram used to maintain pose estimates used 50 bins for each translation dimension and represented rotation by a 3-dimensional vector in Rodrigues form (with magnitude encoding rotation angle) divided into 9 bins per dimension. At each time step, the hypothesized object identity was taken as the object with the most probability weight, summed over all possible poses. The hypothesized object pose was taken as the centroid of the histogram bin with the highest weight.

Performance was measured in terms of classification accuracy and of distance from the estimated pose, $[\hat{R}\ \hat{T}]$, to the true pose, $[R\ T]$, where a pose of $[\mathbf{I}\ \mathbf{0}]$ (with $\mathbf{0} = [0,0,0]^T$) corresponds to the object located at the origin in the pose observed during training. Classification accuracy was taken as the percent of the time the hypothesized object identity was the true identity over all trials. Error in the pose was recorded only when the estimated identity was correct, and it was measured in three ways: Translational error was measured as the distance between the translational components, $\left|\left|\hat{T} - T\right|\right|$. Angular error was taken as the angle of rotation, ϕ_e, required to align the estimate with the true pose, taking into account symmetries in some objects. The Glass, Tire, and Screwdriver objects were all considered to have an axis of rotational symmetry. Let this axis be denoted ζ; then we have

$$\phi_e = \begin{cases} \min_\gamma \left|\left|unskew(logm(R_{R\zeta}(\gamma) R\hat{R}^T))\right|\right| & \text{if symmetric about } \zeta \\ \left|\left|unskew(logm(R\hat{R}^T))\right|\right| & \text{otherwise} \end{cases} \qquad (22)$$



(a) Classification Accuracy

(b) Inertial Error

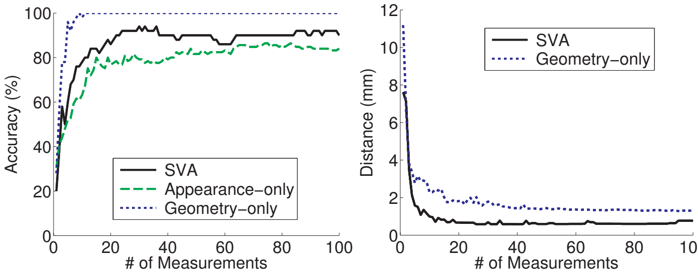
(c) Translational Error

(d) Angular Error

Fig. 4 SVA recognition and localization results on Princeton set.

where *logm* denotes the matrix logarithm, *unskew* extracts the vector v from $sk(v)$, its corresponding skew-symmetric matrix, and $R_\zeta(\gamma)$ is a rotation about ζ by γ. Inertial distance was measured according to the metric of [8, Eqn. 4], where each object's mass and moment of interia was approximated by a solid sphere of radius 40 mm. This metric combines translational and angular error into a measurement of the energy required to align the two transformations. Localization accuracy was not measured for the appearance-only approach, since it does not estimate object pose.

The MNTI descriptor was used to characterize appearance due to its invariance properties' robustness to small translations. Fig. 4 graphs all of the error metrics above: Fig. 4a shows recognition accuracy. Figs. 4b, 4c, and 4d show inertial, translational, and angular error respectively. Classification accuracy climbs to 100% with thirteen sensor readings. Translational error drops to slightly above 4 mm, the lowest expected attainable error using histogram bins of width 8 mm. Angular error remains high, however, most likely due to near-symmetries in the objects; e.g., the knight's base is axially symmetric and the phone nearly has two-fold rotational symmetry. As a result of the angular error, inertial error decreases more slowly.



(a) Classification Accuracy (b) Inertial Error

Fig. 5 SVA recognition and localization results on raised letters

5.2 2D Physical Sensor Experiments

The SVA mapping approach was also tested using physical sensors on capital vowels from a child's set of raised letters (from a Leap Frog "Fridge phonics" magnetic alphabet set), shown in Fig. 1 along with our sensors.

The letters were approximately 2.5 cm per side, so less than a quarter of the letter was visible in any single reading. In order to cover the entire object, readings were collected at 16 planar positions arranged in a 4-x-4 grid with a spacing of 6.8 mm, oriented at 12 evenly-spaced angles at each location for a total of 192 readings. A mechanical system was constructed to position the letters coplanar with the sensors and press them down with a consistent force.

Two readings were taken at each of the 192 poses. The first set of readings at each pose was used for training, while the second set was used for testing.

As before, readings from the unknown object were transformed according to a randomly selected object pose for each trial. This pose consisted of a translation in x and y in the range $[-10, 10]$ mm in each direction and an arbitrary rotation in the plane. This pose space was discretized using 21 bins for each dimension of translation and 9 bins for each dimension of the Rodrigues vector representing rotation. The map used 100 bins for distance, covering a range of 3 mm to 20 mm. Since there was a discrete set of contact locations, invariance to translation was less of a concern, so the Moment-Normalized descriptor was used.

Classification accuracy and inertial distance are shown in Fig. 5. Again, performance of the SVA and appearance-only approach [15] are shown as well as that of the geometry-only approach of [16]. Classification accuracy quickly climbs above 90% within about 30 sensor readings. Inertial distance was measured taking into account symmetries in the letters, so that angular error was measured with respect to the closest of the true pose and its 180° rotation for "I"; "O" was considered fully symmetric, so that angular error was always zero. This metric drops below 1 mm within about 20 sensor readings. This is once again close to the expected optimum given the virtual histogram resolution. While SVA does not achieve the recognition rates of the geometric method in this 2D case, it gives better localization.

6 Discussion

We have presented a method that makes use of both the appearance content of tactile force sensor readings and the geometric information associated with each. The method was demonstrated on both simulated and real tactile data sets, exhibiting strong performance both in recognition accuracy and pose estimation. Performance was not perfect, however, so we provide some analysis of why that may be, ideas for improvement, and guidance on how to apply the method in different situations.

6.1 Analysis of Failure Modes

It is interesting to compare the experimental results of Sec. 5.2 to those of the purely geometric approach of [16] on the same data. The purely geometric approach eventually achieved 100% classification accuracy when using a histogram with 10,000 bins for the pose space, but with a localization accuracy of just over 2 mm; the SVA approach did not quite reach 100% accuracy on the letter set, but its localization accuracy was below 1 mm. Higher pose accuracy is achievable because a higher-resolution histogram can be maintained using a forward mapping from sensor readings to pose and a sparse representation of the probability space. This same difference also makes the SVA approach much more amenable to extension to full 3D. A natural question would be why the SVA method does not achieve perfect classification accuracy. It is instructive to address this question in some detail, as it gives guidance on considerations for applying the method in other situations.

A classification failure must result from the true pose not being among those considered able to explain a patch pair. If the bin of the true pose is assigned zero probability, then the optimal solution will not be found unless all probabilities go to zero and the distribution is re-seeded. There are a few ways this might occur:

1. A sensor reading may be of part of the object surface not observed in training. There would then be no valid match in the map for pairs containing that point.
2. A patch pair may not be matched to the nearest corresponding regions in the map because the estimated appearance and distance do not match well enough.
3. The distribution of aligning transformations computed for a correct (or close) patch pair match may not place significant probability on the true object pose. This might be due to a mis-estimation of the surface patch's locations or (more likely) their surface normals.
4. A combination of the factors above, each acting in part, could push the probability of the true pose below machine precision.

The first situation would not occur with the raised letter data, since sensor readings were taken at set locations, and those locations were the same in training as in testing. The third failure mode is also not likely on the letter data for the same reason, and since the surface normals were all known and equal. The fourth failure mode also did not seem to come into play in our experiments with our chosen parameters. The most likely source of classification failures therefore seems to be the second item, in the appearance classification of surface patch pairs.

6.2 Extensions

Like the geometric method, for which each particle or histogram bin calculation is independent, this approach has great potential for parallelization. With this method, the computations for each surface patch pair can be carried out in parallel. Additionally, the computations for each prospective match for a surface patch pair are also independent, leading to another level of parallelization.

Although this method was developed with tactile force sensors as the intended source of information, it should be noted that it is equally applicable to other sensing modalities. For instance, a stereo vision system could also be used to acquire surface patch information comprising location, surface normal estimates, and appearance. A particularly interesting extension would be to examine what appearance properties can be characterized by both vision and touch; then cross-modality models could be built from one modality and then used in another or with both modalities.

Acknowledgements This work was supported by NSF grants MRI-0722943 and IIS-0748338, and a Link Foundation Fellowship for Simulation and Training.

References

[1] Allen, P., Michelman, P.: Acquisition and interpretation of 3-d sensor data from touch. IEEE Transactions on Robotics and Automation **6**(4), 397–404 (1990)

[2] Arun, K.S., Huang, T.S., Blostein, S.D.: Least-squares fitting of two 3-d point sets. IEEE Trans. Pattern Anal. Mach. Intell. **9**, 698–700 (1987). URL http://portal.acm.org/citation.cfm?id=28809.28821

[3] Bajcsy, R.: What can we learn from one finger experiments? In: International Symposium on Robotics Research, pp. 509–527. Bretton Woods, NH (1984)

[4] Bay, J.: Tactile shape sensing via single- and multifingered hands. In: IEEE International Conference on Robotics and Automation, vol. 1, pp. 290–295. Scottsdale, AZ (1989)

[5] Caselli, S., Magnanini, C., Zanichelli, F., Caraffi, E.: Efficient exploration and recognition of convex objects based on haptic perception. In: IEEE Interna-

tional Conference on Robotics and Automation, vol. 4, pp. 3508–3513. Minneapolis, MN (1996)

[6] Casselli, S., Magnanini, C., Zanichelli, F.: On the robustness of haptic object recognition based on polyhedral shape representations. IEEE/RSJ International Conference on Intelligent Robots and Systems **2**, 2200 (1995)

[7] Chhatpar, S., Branicky, M.: Localization for robotic assemblies using probing and particle filtering. In: Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on, pp. 1379–1384 (2005)

[8] Chirikjian, G., Zhou, S.: Metrics on motion and deformation of solid models. Journal of Mechanical Design **120**, 252 (1998)

[9] Fearing, R.: Tactile Sensing Mechanisms. The International Journal of Robotics Research **9**(3), 3–23 (1990)

[10] Gadeyne, K., Bruyninckx, H.: Markov techniques for object localization with force-controlled robots. In: Int. Conf. Advanced Robotics, pp. 91–96. Citeseer (2001)

[11] Grimson, W., Lozano-Perez, T.: Model-based recognition and localization from tactile data. In: IEEE International Conference on Robotics and Automation, vol. 1, pp. 248–255. Atlanta, GA (1984)

[12] Kaufman, L., Rousseeuw, P.: Finding groups in data: an introduction to cluster analysis, vol. 5. Wiley Online Library (1990)

[13] Petrovskaya, A., Khatib, O., Thrun, S., Ng, A.: Touch Based Perception for Object Manipulation. In: Robotics Science and Systems, Robot Manipulation Workshop (2007)

[14] Pezzementi, Z., Jantho, E., Estrade, L., Hager, G.D.: Characterization and simulation of tactile sensors. In: Haptics Symposium, pp. 199–205. Waltham, MA, USA (2010)

[15] Pezzementi, Z., Plaku, E., Reyda, C., Hager, G.D.: Tactile object recognition from appearance information. IEEE Transactions on Robotics **27**(3), 473–487 (2011)

[16] Pezzementi, Z., Reyda, C., Hager, G.D.: Object mapping, recognition, and localization from tactile geometry. In: IEEE International Conference on Robotics and Automation, pp. 5942–5948. Shanghai, China (2011)

[17] Schneider, A., Sturm, J., Stachniss, C., Reisert, M., Burkhardt, H., Burgard, W.: Object identification with tactile sensors using bag-of-features. In: Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on, pp. 243 –248 (2009)

[18] Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton shape benchmark. In: Shape Modeling International, pp. 167–178. Genova, Italy (2004)

[19] Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics (Intelligent Robotics and Autonomous Agents). The MIT Press (2005)

[20] Wolfson, H.J., Rigoutsos, I.: Geometric hashing: an overview. IEEE Computational Science and Engineering **4**, 10–21 (1997)