Object Mapping, Recognition, and Localization from Tactile Geometry

Zachary Pezzementi, Caitlin Reyda, and Gregory D. Hager

***Abstract*— We present a method for performing object recognition using multiple images acquired from a tactile sensor. The method relies on using the tactile sensor as an *imaging device*, and builds an object representation based on mosaics of tactile measurements. We then describe an algorithm that is able to recognize an object using a small number of tactile sensor readings. Our approach makes extensive use of sequential state estimation techniques from the mobile robotics literature, whereby we view the object recognition problem as one of estimating a consistent location within a set of object maps. We examine and test approaches based on both traditional particle filtering and histogram filtering. We demonstrate both the mapping and recognition / localization techniques on a set of raised letter shapes using real tactile sensor data.**

I. INTRODUCTION

Object recognition is a fundamental problem that must be solved for robots to be able to usefully and practically interact with real world environments. Though the problem has received extensive attention in the computer vision domain, research into touch-based algorithms has been less common recently. Previous generations of tactile sensors had extremely limited force and spatial resolution. As a result, early object recognition using tactile sensing focused almost entirely on producing clouds of contact points to constrain the geometry of the object [1]–[5]. Other research investigated methods to infer local geometry, such as surface normals or curvature, associated with each contact point [5]–[8].

Recent advances in capacitive array-type tactile sensors have altered this picture. Despite still exhibiting relatively low spatial resolution (currently available sensors are still a factor of two to four away from human spatial resolution), they now have the force resolution and repeatability necessary for extracting rich surface geometry information from aggregated sensor responses. This *tactile image* information can be used in many ways. For example, in recent work, Schneider et al. have viewed tactile images as characterizing surface "tactile appearance", and they applied a computer vision-style recognition method to these images [9].

In this paper, we investigate an approach to recognition that makes use of tactile images to characterize the geometry of a local surface patch of an object and to combine those surface patches to build up an object-level geometric surface image. As a result, the object representations used for recognition are learned from data, rather than being available from a prior geometric model, as is traditional [6], [10]–[12]. Also, we use a probabilistic approach, allowing better handling of multiple plausible hypotheses and providing a more flexible means of modeling uncertainty, such as from sensor noise. Finally, our algorithm is implemented and tested using data from real-world sensors.

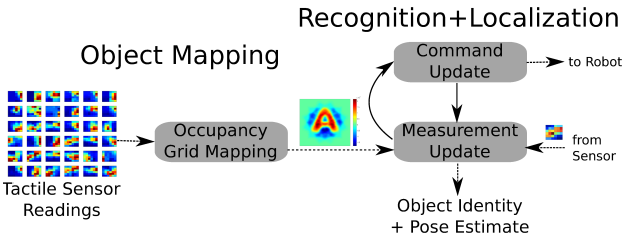


Fig. 1. Data flow for entire model building and recognition process.

Our approach to tactile-based object recognition has its roots in sequential state estimation techniques from the field of mobile robot localization [13]–[16]. Previous authors have used similar techniques to address object localization using force sensing, but assumed the object identity and geometry were known [11], [12], [17]. Briot presents a probabilistic approach to tactile mapping and identification of 3D objects in [18], but objects are restricted to being piece-wise planar, where the entirety of one of these planes must be visible in each sensor reading. We instead consider the object recognition problem to be analogous to a global localization with combined discrete (object identity) and continuous (object pose) state in an environment where a map had previously been learned. Tactile exploration then consists of a series of discrete time steps in which a motion command is sent to a robot hand and a set of tactile sensor readings are received as a result, as illustrated in Fig. 1. Our experiments focus on the case of a single-finger robot equipped with a single planar tactile force sensing array, but the method is equally applicable to multi-fingered robots or even deformable (e.g. skin-like) tactile arrays, provided the geometry of the array is known.

In the remainder of this paper, we focus on the mapping (Sec. II) and recognition and localization (Sec. III) processes, independent of the exploration method. We therefore assume the existence of a robot system capable of pressing tactile sensors against the unknown object under force control to produce consistent sensor readings. We also assume that some exploration algorithm is available to choose which measurements to seek out. In practice, the requirements for the exploration algorithm are modest, since we show that it is possible to achieve good recognition and localization

Z. Pezzementi and G. D. Hager are with the Dept. of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA. {zap,hager}@cs.jhu.edu

C. Reyda is with the Department of Mechanical Engineering, Massachussetts Institute of Technology, Cambridge, MA 02139, USA. reyda@mit.edu

(a) Letters and Sensors (b) Letter Pressing Apparatus (c) Letter-Sensor Contact

Fig. 2. (a) shows the set of raised letters used in our experiments, alongside the PPS DigiTacts sensors, with the sensing area highlighted in blue. (b-c) show the apparatus used for pressing letters against the sensors during data collection. The sensors are covered with a thin layer of black polyurethane, as described in [19].

performance with essentially random exploration.

II. TACTILE MAP BUILDING

In order to perform object recognition, we first need a model of each of the objects to be recognized. Such a model would be constructed during a training session by freely exploring the object. For the purposes of this paper, the information gathered during this session consists of tactile images and the locations, relative to some fixed coordinate system on the object, at which the images were gathered. Following the notation of [13], we view this as a time-series process that provides, at each time step, a tactile sensor measurement, z_t, and an estimate of the location (state), x_t, at which that measurement was collected.

Given a series of measurements and locations, it is possible to build a *tactile mosaic* that represents, in the force domain, the appearance of the object being explored. We use an occupancy grid mapping technique [20], [21] to convert such a collection of tactile force sensor readings into an object-specific *map* that represents object geometry. More specifically a map, $\mathbf{M}$, of the object is generated by dividing the workspace into a set of cells, $\{\mathbf{m}_i\}$, each of which will contain an estimate of the probability that the corresponding location is filled (i.e. generates a force reading) or vacant. We therefore wish to estimate, for each location, the probability $p(\mathbf{m}_i|z_{1:t},x_{1:t})$.

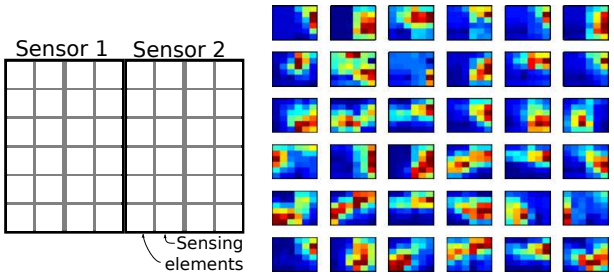
The map estimates are stored in log-odds form,

$$l_{t,i} = \log \frac{p(\mathbf{m}_i|z_{1:t},x_{1:t})}{1-p(\mathbf{m}_i|z_{1:t},x_{1:t})} \quad (1)$$

and we follow the standard method [13] of defining a measurement model to directly estimate $l_{t,i}$ from $l_{t-1,i}$:

$$l_{t,i} = l_{t-1,i} + \log \frac{p(\mathbf{m}_i|z_t,x_t)}{1-p(\mathbf{m}_i|z_t,x_t)} - \log \frac{p(\mathbf{m}_i)}{1-p(\mathbf{m}_i)} \quad (2)$$

$p(\mathbf{m}_i)$ can be estimated from the average object surface area and the area being modeled. The second term on the right hand side of (2) is referred to as the "inverse sensor model", since it inverts the normal forward model of measurement formation to infer map properties, and it must be estimated



(a) Sensor Layout (b) Sample Readings

Fig. 3. (a) Detailed view of the layout of our PPS DigiTacts system. (b) Example tactile images used for recognition and model building, all from the "A" model.

for a particular sensor system or learned from data. A model for our sensor system is derived below.

We illustrate this method in 2D on capital vowels from a child's set of raised letters (from a Leap Frog "Fridge phonics" magnetic alphabet set), shown in Figs. 2a and 2c. The sensor is a DigiTacts sensor system [22] by Pressure Profile Systems. The sensor used here consisted of two individual 6-x-4 element sensors placed side-by-side, as shown in Fig. 3a. The sensors are capacitive in nature, with a force resolution of approximately 0.7 kPa. Each sensing element was 1.8 mm square. We only used the center 6-x-6 element region, so, with the non-sensing material surrounding each sensing element, the total area being used was 11.6-x-11.3 mm, with 89% of that area being sensed. The sensors were covered with a thin layer of polyurethane, which helps to maintain a uniform response by spreading forces across multiple sensing elements, as described in [19].

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 manually press them down with a consistent

force (Fig. 2). Sample tactile readings collected from the "A" model are shown in Fig. 3b. Due to the structured nature of the data collection, we assumed no error in the state estimates during training, $x_{1:t}$, associated with the measurements $z_{1:t}$.

In order to convert the image from the force to the contact domain, we chose to binarize inputs rather than continue with continuous readings. Let the individual sensor elements of reading z_t be denoted $e_{t,j}$ for $j = \{1...n_e\}$. To determine whether element $e_{t,j}$ has detected contact or no contact with the object, with $e_{max} = \max_j e_{t,j}$, the following classification function was found to work well in practice:

$$c(e_{t,j}) = \begin{cases} \text{In Contact} & \text{if } e_{t,j} > T \text{ AND } e_{t,j} > 0.3e_{max} \\ \text{No Contact} & \text{if } e_{t,j} < T \text{ AND } e_{t,j} < 0.3e_{max} \\ \text{Unknown} & \text{otherwise} \end{cases} \tag{3}$$

with other elements left classified as "unknown" and T equal to about 2 kPa for our sensors. Informal evaluation showed this classifier produced the correct classification about 90% of the time and a random value the remaining 10% of the time. This performance was found to be sufficient for our purposes.

The response of these sensors had been characterized in previous work [19] and found to be well-approximated by a displacement map convolved with a symmetric Gaussian point spread function, $G = N(0,\sigma)$. The inverse of this measurement model, G', i.e. the distribution of object locations which might produce a contact reading (or the empty locations which may produce "no contact" readings) was approximated by a Gaussian with half the standard deviation of the point spread function. This value was used as an estimate of the locations where the force response would be more than 30% of the maximum value.

Let $\overline{\mathbf{m}_i}$ denote the location of the center of mass of the map cell $\mathbf{m}_i$ and $\overline{e_{j,t}}$ denote the location of the center of sensor element e_j when the sensor is at the state x_t. We now view each a map cell, $\mathbf{m}_i$, as a binary random variable, and define the probability of occupancy as follows:

$$G'(\mathbf{m}_i; e_{t,j}, x_t) = \frac{1}{\sqrt{\pi\sigma^2/4}} \exp\left(-\frac{\|\overline{\mathbf{m}_i} - \overline{e_{t,j}}\|^2}{8\sigma^2}\right) \tag{4}$$

$$p(\mathbf{m}_i|e_{t,j},x_t) = p(\mathbf{m}_i) + \begin{cases} (1-p(\mathbf{m}_i))\,G'(\mathbf{m}_i;e_{t,j},x_t) & c(e_{t,j}) = \text{In Contact} \\ -p(\mathbf{m}_i)G'(\mathbf{m}_i;e_{t,j},x_t) & c(e_{t,j}) = \text{No Contact} \\ p(\mathbf{m}_i) & c(e_{t,j}) = \text{Unknown} \end{cases} \tag{5}$$

Distinct measurements are assumed to be independent, making map updates order-independent. Once the probabilities in the measurement model are converted to log odds form, map updates are simply additive:

$$l_{t,i} = l_{t-1,i} + \sum_{j=1}^{n_e} \log \frac{p(\mathbf{m}_i|e_{t,j},x_t)}{1-p(\mathbf{m}_i|e_{t,j},x_t)} - \log n_e \frac{p(\mathbf{m}_i)}{1-p(\mathbf{m}_i)} \tag{6}$$

Since the map is built offline and the updates are order-independent, a grid of arbitrary resolution can be efficiently built on a cell-by-cell basis. Visualizations of the reconstructions of these letters are show in Fig. 4.

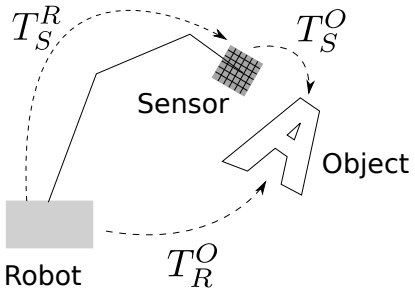


Fig. 5. Reference frames used in tactile exploration of the unknown object

III. RECOGNITION AND LOCALIZATION

We view the recognition task as a sequential state estimation problem, where, at each time step, t, we are given a measurement, z_t, and we wish to estimate the robot state, x_t, at which that measurement was taken. That state consists of the identity of the object, C, along with the pose of the sensor in the object coordinate frame, T_S^O. We assume that we also have, for each measurement, an estimate of the pose of the sensor in the robot frame, T_S^R, from the robot forward kinematics. These frames are illustrated in Fig. 5. Assuming both the robot base coordinate system and the object remain fixed in the world, the problem reduces to estimating the (constant) transformation between robot and object frames, T_R^O since we can recover T_S^O as $T_R^O T_S^R$. For our manual sampling method, T_R^O was defined to be the identity transformation at map building time – i.e. the object frame is simply the robot frame. T_S^O can be decomposed into two translation components and an angle, giving $x = [C, \mathbf{t}_x, \mathbf{t}_y, \theta]$.

Non-parametric estimation methods were chosen because of their strength at representing multi-modal distributions and the relative ease of dealing with combined discrete and continuous state with these methods. The task can then be viewed as an application of either the classic histogram filter [16] or the more widely known particle filter [14], [15] localization techniques from mobile robotics. Each has its relative theoretical merits, so we present both methods for comparison.

Following the notation of [13] again, both filters provide ways of estimating a probability distribution, $p(x_t|u_{1:t}, z_{1:t}, \mathbf{M})$ over state at the current time, t, given a series of commands, $u_{1:t}$, sent to the robot and measurements, $z_{1:t}$, received from the sensors and the map learned in Sec. II.

In the histogram filter, the distribution is estimated using a multi-dimensional histogram. The state space is decomposed into n_H regions, and each histogram bin corresponds to one of these regions, $\mathbf{x}_k$, with $k \in \{1..n_H\}$. The value in that bin, $p_{k,t}$, represents the probability of the state lying within the corresponding region. The particle filter is a Monte-Carlo method, where the distribution is modeled by a set of n_P samples called particles, written as $\mathscr{X}_t = \{x_t^{[1]}..x_t^{[n_P]}\}$.

In both cases, at each time step, we must perform a command update and a measurement update, as illustrated in Fig. 1. The command update requires estimation of $p(x_t|x_{t-1}, u_t)$ for a control input u_t. The control input only

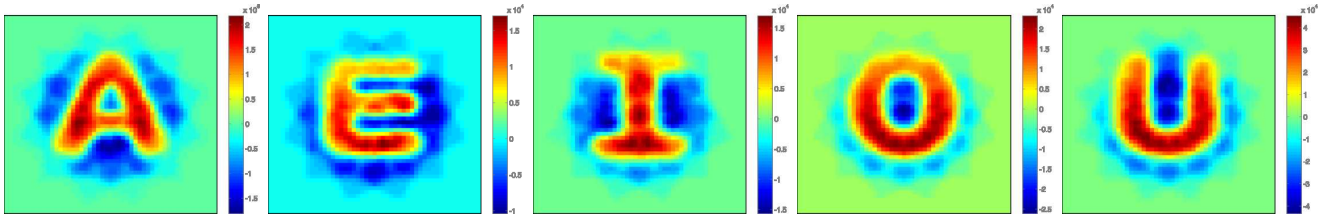


Fig. 4. Reconstruction of letters from tactile images. Pixels are colored by probability of occupancy in log-odds form. This probability can be seen to be high on the letters themselves, to be low in the surrounding area where measurements were taken, and to decay to the prior probability in the surrounding un-sensed area.

affects T_S^R, which is known exactly. Since neither the object identity nor T_R^O is changing from one time step to the next, the command update only requires updating the robot kinematics estimates. It is in the measurement update, therefore, that the distribution is reshaped.

A. Measurement Model

Both particle filtering and histogram filtering implement approximations to a Bayesian filter [13], and thus both require a measurement likelihood model, $p(z_t|x_t,\mathbf{M})$. This can be computed from the measurement model mentioned in Sec. II. For the sensor system used in our experiments, each reading consists of 36 individual element responses, each of which may detect In Contact, No Contact, or Unknown, according to the classifier of (3). The expected measurement of each sensor element $e_{t,j}$ is computed by interpolation of the occupancy grid built in Sec. II. The location of the element in the sensor frame, $pos(e_{t,j})^S$, is transformed to the object frame using the hypothesized object pose to get $pos(e_{t,j})^O = T_S^O pos(e_{t,j})^S$. Then, the occupancy grid is queried for a log-odds value at the location $pos(e_{t,j})^O$ through bilinear interpolation of the grid point estimates to give $p(occ_{t,j}|x_t,\mathbf{M})$. Incorporating the chance of classification failure, γ, gives the following likelihood formulation:

$$p(e_{t,j}|x_t,\mathbf{M}) = \begin{cases} \gamma p(\mathbf{m}) + (1-\gamma)p(occ_{t,j}|x_t,\mathbf{M}) & c(e_{t,j}) = \text{In Contact} \\ \gamma p(\mathbf{m}) + (1-\gamma)p(\neg occ_{t,j}|x_t,\mathbf{M}) & c(e_{t,j}) = \text{No Contact} \\ 1 & c(e_{t,j}) = \text{Unknown} \end{cases} \tag{7}$$

where $p(\mathbf{m})$ is the prior for occupancy of an individual map cell, which effectively ignores Unknown measurements. The element measurements are assumed to be independent, so the overall measurement probability is then taken as the product of the individual element probabilities: $p(z_t|x_t,\mathbf{M}) = \prod_j p(e_{t,j}|x_t,\mathbf{M})$

B. Particle Filter Estimation

For the particle filter, the command update consists simply of copying over the particles from the previous time step, with the components of T_R^O corrupted by a small amount of Gaussian-distributed noise. One could view this as modeling uncertainty in T_S^R or small changes in the pose that might occur from one time step to the next, but, in either case, this injection of randomness is necessary to ensure the particles cover the space, even if the object is completely stationary [14]. Let the components of particle m be $x_t^{[m]} = [C_t^{[m]}, \mathbf{t}_{x,t}^{[m]}, \mathbf{t}_{y,t}^{[m]}, \theta_t^{[m]}]$. To form the updated particle, $\bar{x}_t^{[m]}$, the identity component is copied over without modification. In our experiments, the other components were corrupted with noise sampled from $N(0,\sigma)$, with $\sigma = 1$ mm for $\mathbf{t}_{x,t}^{[m]}$ and $\mathbf{t}_{y,t}^{[m]}$ and $\sigma = 0.1$ radian for $\theta_t^{[m]}$.

During the measurement update, each particle is assigned an importance weight, $w_t^{[m]} = p(z_t|\bar{x}_t^{[m]})$, which is evaluated as described above in Sec. III-A. Then, the particles for the next time step are generated via importance sampling by drawing n_P new particles from $\{\bar{x}_t^{[m]}\}$ with replacement, where particle m is drawn with probability proportional to $w_t^{[m]}$.

At each time step, the current object identity estimate, $\hat{C}$, is taken as the model hypothesized by the most particles, weighted by their importance. Let $Ind(A,B)$ be an indicator function defined

$$Ind(A,B) = \begin{cases} 1 & \text{if } A = B \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Then the current identity estimate is given by

$$\hat{C} = \arg\max_C \sum_{m=1}^{n_P} w_t^{[m]} Ind(C_t^{[m]}, C) \tag{9}$$

The current state estimate, $\hat{x}$, is obtained from the estimated mode of the posterior distribution as the pose corresponding to the maximum value of a kernel density estimate over the particles, weighted by their importance. Then we have

$$cone(d) = \max\left(0, 1 - \frac{d}{\rho}\right) \tag{10}$$

$$Parzen(x) = \sum_{m=1}^{n_P} w_t^{[m]} cone(x - \bar{x}_t^{[m]}) \tag{11}$$

$$\hat{x} = \arg\max_x Parzen(x) \tag{12}$$

$Parzen(x)$ was evaluated on a grid of resolution 0.5mm and 0.04π radians, with ρ set to 1 in both mm and radians.

C. Histogram Filter Estimation

In our experiments, discretization of the state space is accomplished simply by division into n_H equal-volume rectangular regions. The probability for each histogram bin is

estimated based on the centroid $\check{x}_k$ of the corresponding region $\mathbf{x}_k$.

As stated previously, the command update only requires updating the robot kinematics estimates, so no change to the histogram is needed. For the measurement model, each bin is updated as

$$p_{k,t} = p_{k,t-1} p(z_t | \check{x}_k, \mathbf{M}) \tag{13}$$

Letting C_k be the identity corresponding to region $\mathbf{x}_k$, the current identity estimate is given by

$$\hat{C} = \arg\max_C \sum_k p_{k,t} Ind(C_k, C) \tag{14}$$

and the current state estimate is once again taken as the mode of the distribution, i.e.

$$\ell = \arg\max_k p_{k,t} \tag{15}$$

$$\hat{x} = \check{x}_\ell \tag{16}$$

IV. EXPERIMENTS AND RESULTS

The method was tested by repeatedly performing recognition on all of the models in our 5 letter test set under different random 2D transformations. Transformations consisted of a translation selected uniformly at random from the range $[-10, 10]$ mm in both the x- and y-directions and the entire range of rotations in the plane. For each model, performance metrics were averaged across 10 trials.

The measurements used for testing consisted of an alternate set of readings taken in the same positions and orientations as in the training set. In each trial, a sequence of measurements was selected from the test set at random, without replacement. A motion command and measurement were then generated to simulate the situation of a robot exploring the object in the unknown pose. The location information associated with each measurement was transformed according to the random pose generated above, then the command and measurement were presented to the recognition algorithm.

Accuracy in identifying the unknown object was recorded as a function of the number of readings seen. Performance in this metric is shown for the particle filter with 100, 1,000, and 10,000 particles in Fig. 6a. The performance of the histogram filter under the same metric is shown with those same numbers of bins (broken down as shown in Table I) in Fig. 7a. The particle filter's performance leveled out for all numbers of particles after around 10 measurements as the particles prematurely converged to a single (often incorrect) hypothesis, and this approach only achieved an accuracy of 74% with 10,000 particles. The histogram filter, however, was able to achieve 100% accuracy with only 1,000 histogram bins after about 50 measurements and with 10,000 bins with only 9 readings. The computation required for a single histogram bin is roughly equivalent to that for a single particle, so this was a striking difference in performance. Attempts to avoid premature convergence of the particle filter through the injection of random particles, as in Augmented MCL [23] did not improve performance.

Total Bins	x-translation	y-translation	Angle
100	5	5	4
1,000	10	10	10
10,000	25	25	16

TABLE I
HISTOGRAM BIN RESOLUTION IN EACH STATE SPACE DIMENSION

The average distance between the current state estimate and the true state was also computed for each iteration when the state estimate's object identity hypothesis was correct. This distance was computed using the metric from [24] (Eqn. 4), which uses an object's mass and moments of intertia to compute a scalar distance between two transformations, taking into account both translation and rotation. In our case, mass and moments of inertia were estimated from the object maps assuming uniform object density. The units used to report error were normalized with respect to the energy required to translate the object a distance of 1 mm, though a component of the reported values is also due to rotational error. This metric was modified to account for the rotational symmetry of two of the letters: "O" was considered to be fully rotatioally symmetric and to have no rotation error, while "I" was considered to have two-fold rotational symmetry. The angular error for each letter, denoted $dAng_{Let}(\hat{\alpha})$ for letter Let, is given as

$$dAng(\hat{\alpha}) = |mod(\hat{\alpha} - \alpha, 2\pi)| \tag{17}$$

$$dAng_A(\hat{\alpha}) = dAng_E(\hat{\alpha}) = dAng_U(\hat{\alpha}) = dAng(\hat{\alpha}) \tag{18}$$

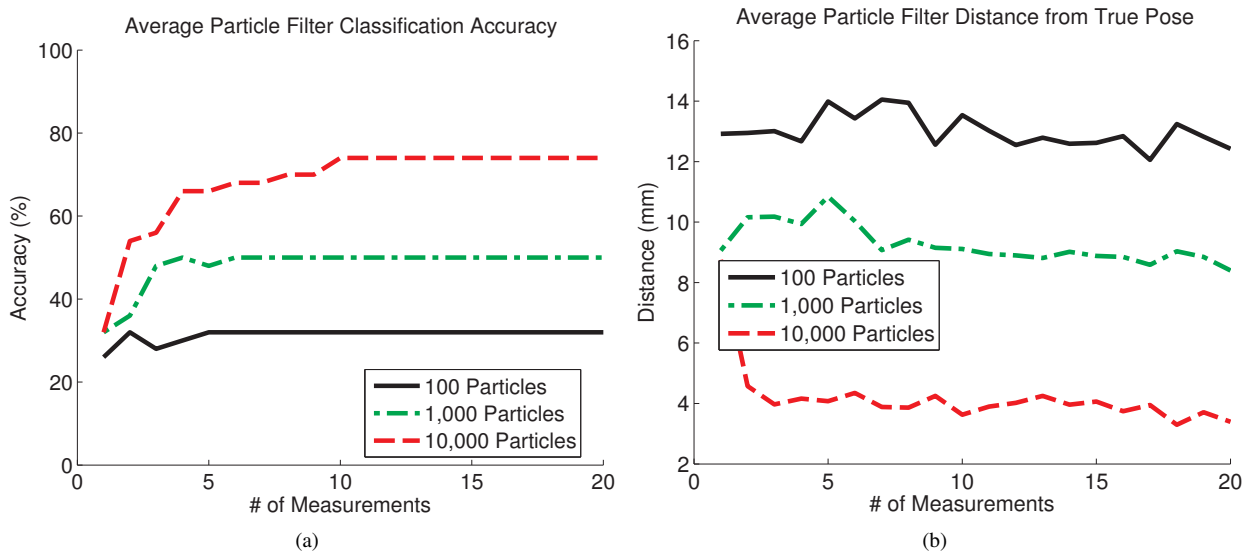
$$dAng_I(\hat{\alpha}) = |mod(\hat{\alpha} - \alpha, \pi)| \tag{19}$$

$$dAng_O(\hat{\alpha}) = 0 \tag{20}$$

Transformation error is shown for the particle filter in Fig. 6b and for the histogram filter in Fig. 7b. Note that the performance of the histogram filter in this metric is inherently limited by the histogram resolution, therefore the particle filter should have a strong advantage. Nonetheless, the histogram filter performed better again for all numbers of particles/bins.

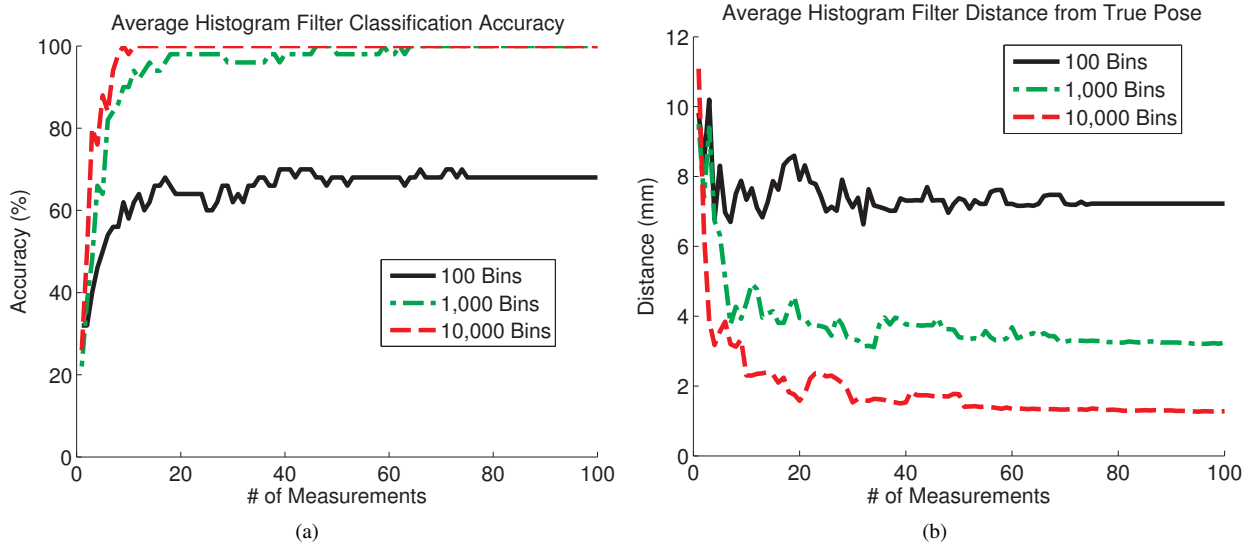
V. CONCLUSIONS AND FUTURE WORK

We have presented a method for generating rich surface models from array-type tactile force sensors and demonstrated its use on a set of real objects. A recognition algorithm was also described, which uses these surface models to estimate an unknown object's identity and pose, using only a small number of measurements. We defined two solutions to our formulation of the task as a sequential state estimation problem and compared their object identification and localization performance. The histogram filtering algorithm was able to achieve 100% accuracy on the test set with as few as 46 sensor readings using 1,000 bins and with only 9 readings using 10,000 bins, while objects were localized to within 1.3 mm of their true positions on average. Histogram filtering was shown to out-perform particle filtering in all cases.



(a) (b)

Fig. 6. Particle filter performance at different resolutions.



(a) (b)

Fig. 7. Histogram filter performance at different resolutions.

In our future work, we intend to extend this framework to incorporate other types of measurements available from tactile sensor readings, such as tactile appearance features or interest points. Tactile features characterizing local appearance combined with geometry may provide better initial recognition performance, but they can be combined with map-based methods to provide good object localization.

There are two other obvious directions for future research. First, in this work, we have focused entirely on the interpretation of whatever static sensor readings were available, but an active and informed exploration process could also greatly improve recognition. In particular, we expect seeking out and matching tactile interest points to be instrumental in reducing the state space that needs to be searched, particularly when moving to the recognition of full three-dimensional objects. Second, there is no reason to separate the imaging process from the interpretation process. Indeed, it is quite likely that processing *image sequences* from a tactile array provides significant and useful information about object properties and identity. Both of these extensions are quite possible within the framework we have described.

VI. ACKNOWLEDGMENTS

This work was supported, in part, by NSF grants MRI-0722943, IIS-0748338 and EEC-0649069, and a Link Foundation Fellowship for Simulation and Training.

REFERENCES

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

[2] R. Bajcsy, "What can we learn from one finger experiments?" in *International Symposium on Robotics Research*, Bretton Woods, NH, 1984, pp. 509–527.

[3] ——, "Shape from touch," in *Advances in Automation and Robotics*, G. Saridis, Ed. Greenwich, CT: JAI Press, 1985, pp. 209–258.

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

[5] P. K. Allen and K. S. Roberts, "Haptic object recognition using a multi-fingered dextrous hand," in *IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, 1989, pp. 342–347.

[6] W. Grimson and T. Lozano-Perez, "Model-based recognition and localization from tactile data," in *IEEE International Conference on Robotics and Automation*, vol. 1, Atlanta, GA, 1984, pp. 248–255.

[7] R. Fearing, "Tactile Sensing Mechanisms," *The International Journal of Robotics Research*, vol. 9, no. 3, pp. 3–23, 1990.

[8] S. Caselli, C. Magnanini, F. Zanichelli, and E. Caraffi, "Efficient exploration and recognition of convex objects based on haptic perception," in *IEEE International Conference on Robotics and Automation*, vol. 4, Minneapolis, MN, Apr 1996, pp. 3508–3513.

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

[10] W. Grimson and T. Lozano-Perez, "Model-based recognition and localization from tactile data," Cambridge, MA, August 1983.

[11] K. Gadeyne and H. Bruyninckx, "Markov techniques for object localization with force-controlled robots," in *Int. Conf. Advanced Robotics*. Citeseer, 2001, pp. 91–96.

[12] A. Petrovskaya, O. Khatib, S. Thrun, and A. Ng, "Touch Based Perception for Object Manipulation," in *Robotics Science and Systems, Robot Manipulation Workshop*, 2007.

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

[14] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *IEEE International Conference on Robotics and Automation*. Citeseer, 1999, pp. 1322–1328.

[15] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," in *Proceedings of the National Conference on Artificial Intelligence*. John Wiley & Sons Ltd., 1999, pp. 343–349.

[16] R. Simmons and S. Koenig, "Probabilistic robot navigation in partially observable environments," in *International Joint Conference on Artificial Intelligence*, vol. 14, 1995, pp. 1080–1087.

[17] S. Chhatpar and M. Branicky, "Localization for robotic assemblies using probing and particle filtering," in *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on*, July 2005, pp. 1379 –1384.

[18] M. Briot *et al.*, "The utilization of an artificial skin sensor for the identification of solid objects," in *9th Int. Symp. on Industrial Robots*, 1979, pp. 13–15.

[19] Z. Pezzementi, E. Jantho, L. Estrade, and G. D. Hager, "Characterization and simulation of tactile sensors," in *Haptics Symposium*, Waltham, MA, USA, 2010, pp. 199–205.

[20] A. Elfes, "Sonar-based real-world mapping and navigation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 3, pp. 249 –265, jun. 1987.

[21] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Mag.*, vol. 9, no. 2, pp. 61–74, 1988.

[22] PPS, "DigiTacts II™, tactile array sensor evaluation kit with digital output," Pressure Profile Systems. [Online]. Available: http://www.pressureprofile.com/UserFiles/File/DigiTactsII%20Evaluation%20Specification%20Sheet.pdf

[23] J. Gutmann, W. Burgard, D. Fox, and K. Konolige, "An experimental comparison of localization methods," in *1998 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1998. Proceedings.*, vol. 2, 1998.

[24] G. Chirikjian and S. Zhou, "Metrics on motion and deformation of solid models," *Journal of Mechanical Design*, vol. 120, p. 252, 1998.