Touch Based Perception for Object Manipulation

Anna Petrovskaya, Oussama Khatib, Sebastian Thrun, Andrew Y. Ng
Computer Science Department
Stanford University
Stanford, California 94305, USA
{ anya, ok, thrun, ang }@cs.stanford.edu

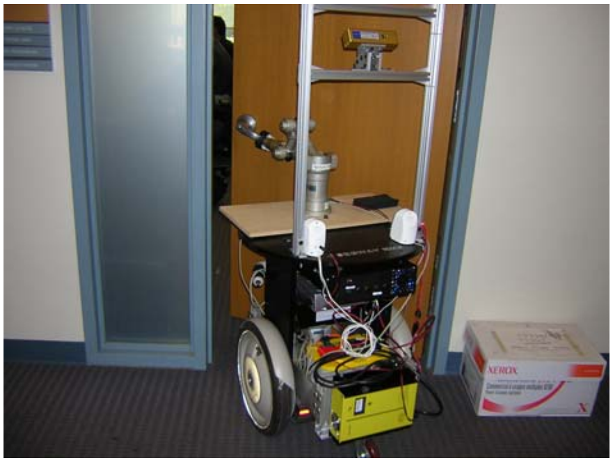


Fig. 1. A mobile robot manipulating a door handle during one of our experiments.

***Abstract*— Humans are capable of manipulating objects solely based on the sense of touch. To study this capability in robots, we focus on touch based object localization. At each stage of exploration our goal is to estimate a Bayesian posterior based on measurements obtained thus far. The state space for object localization is six dimensional: three parameters for position and three for orientation of the object. When initial uncertainty is high (0.5m and 360 degrees), precise estimation of the posterior is computationally expensive. We propose an efficient technique that estimates the posterior in real time. The approach - termed Scaling Series - is based on importance sampling. It performs the estimation using a series of successive refinements, gradually scaling the precision from low to high. Our approach can be applied to a wide range of manipulation tasks. We demonstrate its portability on two applications: (1) picking up a box and (2) operating a door handle.**

I. Introduction

In order to carry out manipulation tasks in real world environments, robots need to perceive objects around them based on sensory information. Although the use of vision for robotic perception has received the most attention in the literature [1], humans rely heavily on the sense of touch for manipulation tasks [2]. To study this capability in robots, we consider touch based object localization. We propose an efficient approach, capable of performing the estimation in real time. Our approach enables robots to carry out manipulation tasks autonomously as we demonstrate on two real life applications: manipulating a box and operating a door handle.

Efficient tactile perception algorithms have been proposed in the past. For example in [3], the authors proposed an efficient method for object identification and localization from tactile data based on interpretation trees. These approaches are very useful in situations where the goal is to estimate a single hypothesis of the state. However, in many situations it can be desirable to estimate the probability distribution over all states. For example this information can enable robots to make better sensing decisions during the exploration process. Probability distributions are typically estimated via Bayesian techniques. These techniques are widely used in mobile robotics [4], motion capture [5], and speech recognition [6]. Recently several approaches applied Bayesian techniques to touch based perception [7]–[9]. The closest prior art to the problem we are approaching in this paper is [9], where authors have advocated the use of particle filters for object localization using a force controlled robot. The localization was restricted to 3 degrees of freedom (DOF), due to computational costs.

In this paper we consider object localization in 6 DOF using touch based exploration. At each stage of exploration, our goal is to estimate a Bayesian posterior based on the measurements obtained thus far. At the final stage of exploration, enough measurements have been obtained to fully constrain the problem, and thus the posterior becomes unimodal. However, during earlier stages of exploration the problem is under-constrained and possible solutions can form entire regions of space of non-zero dimensionality. Estimating this type of posterior in 6DOF precisely is computationally expensive. We propose an efficient approach - termed Scaling Series - capable of estimating the resulting posterior. The approach is based on importance sampling. It estimates the posterior using a series of successive refinements, gradually scaling the precision from low to high. In unimodal cases, precise estimation is possible in real time (1 second). In under-constrained cases, the approach allows a tradeoff between running time and precision, so that coarse estimates can be obtained very quickly.

Our approach is easily applicable to any object represented as a polygonal mesh. We demonstrate its portability on two real life applications. In the first, we localize and manipulate a box. In the second, we localize a door handle, so as to turn the handle and open the door. An earlier version of this paper appeared in [10].

II. Background

Consider a simple example of having measurements from 5 different sides of a rectangular box (see Fig. 2). Let us assume that each measurement contains contact position and

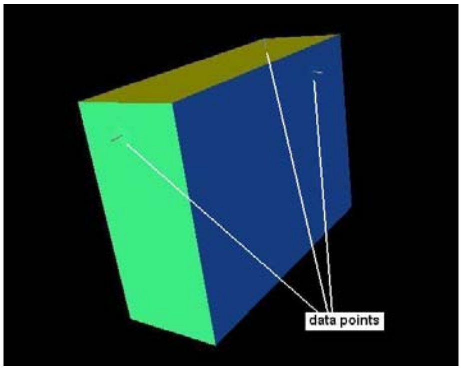


Fig. 2. To estimate position and orientation of a box we take 5 measurements from different sides.

surface normal. How to best estimate position and orientation of the box from these measurements? A simple approach would be to take averages of normals on opposing sides, then fit orthogonal basis to the resulting normals, then perform best fit of corresponding box faces. This approach will work for a box with 6 sides, but will not generalize to arbitrary polygonal meshes of complex objects or if the dataset is incomplete.

Bayesian approach provides the means of parameter estimation for arbitrary objects and datasets. The measurements are considered as being caused by the world with certain probability, called the measurement model $p(Y|X,m)$. Here Y is a measurement consisting of the contact position $Y_p = (x_p, y_p, z_p)$ and the surface normal $Y_n = (n_x, n_y, n_z)$, X is the position and orientation $(x, y, z, \alpha, \beta, \gamma)$ of the object and m is the model of the object (i.e. polygonal mesh). Given a set of measurements, D, the goal is to find the probability distribution of possible states given the measurements and the model. In other words find the posterior distribution $p(X|D,m)$. In the rest of this paper, we will drop the model m from equations for the sake of brevity, although conditioning on the model will be always assumed. It is common to assume that the dependence between measurements $D = \{Y^{(j)}\}$ is based solely on the state X, and that the prior probabilities of state and measurements are uniform. Under these assumptions it can be shown that the posterior is proportionate to the product of measurement likelihoods:

$$p(X|D) = \eta \prod_j p(Y^{(j)}|X) \tag{1}$$

Here η denotes the normalizing constant. One common Bayesian method is importance sampling, where weights are computed according to equation 1 for a number of points (particles) sampled from the state space. The posterior is then represented by these weighted points. See [11] for an overview of importance sampling and other Monte Carlo methods.

III. Measurement model

We represent objects by a polygonal mesh consisting of faces $\{f_i\}$. Based on this object model we compute the likelihood of a measurement as follows. For each face, f_i, we compute the likelihood of the measurement being caused by that face (and a given state X). We assume that the face most likely to cause the measurement was the one that caused it. For convenience, let us introduce correspondence variables $\{c_i\}$. We will assume that $c_i = 1$ when face f_i has caused the measurement, and $c_i = 0$ otherwise. When conditioning, we will write c_i as a shorthand for $c_i = 1$. Thus our measurement model is defined by

$$p(Y|X) = \lambda \max_i \{p(Y|X, c_i)\}, \tag{2}$$

where λ is the normalizing factor given by

$$\lambda = \frac{1}{\int \max_i \{p(Y|X, c_i)\} dY}.$$

Since we do not impose any limitations on the measurement space, λ is independent of the state X. In practice we never need to compute numeric value of this factor as it is taken care of during normalization step.

Recall that each measurement, Y, consists of two parts: contact position, Y_p, and surface normal, Y_n. When computing how likely a measurement to be caused by a face f_i, we consider the two parts of the measurement to be independent. We use state parameters X to transform the measurement into the coordinate system of the object and denote transformed measurement components Y_p^X and Y_n^X respectively. Thus equation 2 becomes:

$$p(Y|X) = \lambda \max_i \{p(Y_p^X|c_i)\ p(Y_n^X|c_i)\}$$

Further, we assume the noise of each measurement component to be Gaussian, with variance err_p^2 for contact position and err_n^2 for surface normal. Thus, the likelihoods can be computed as follows:

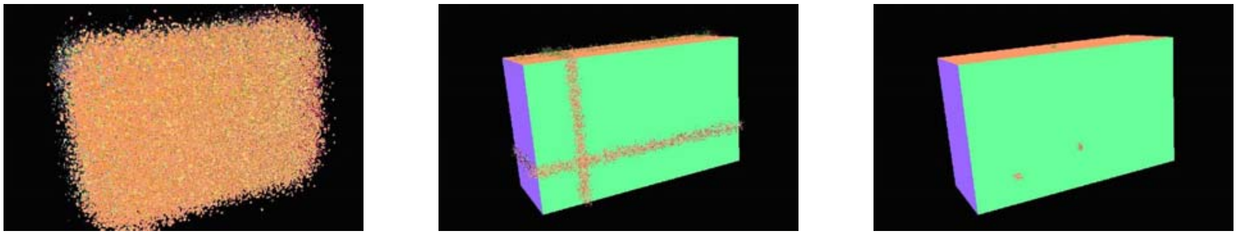
$$\begin{aligned} p(Y_p^X|c_i) &= \frac{1}{\sqrt{2\pi}\ err_p} \exp\{-\frac{1}{2}\frac{d(Y_p^X, f_i)^2}{err_p^2}\} \\ p(Y_n^X|c_i) &= \frac{1}{\sqrt{2\pi}\ err_n} \exp\{-\frac{1}{2}\frac{||Y_n^X - normal(f_i)||^2}{err_n^2}\} \end{aligned}$$

Here $d(Y_p^X, f)$ is the shortest Euclidean distance from Y_p^X to face f_i, and $normal(f_i)$ is the normal vector of face f_i.

IV. Posterior estimation

As we explore an object, more and more measurements arrive and the shape of the posterior changes. See figure 3 for an example of posterior evolution. At early stages, few measurements have been obtained and the problem has not been fully constrained. Thus there are infinitely many possible solutions to the localization problem and the resulting posterior has regions of high likelihood that have non-zero dimensionality. Even though early stages of exploration do not provide sufficient data to fully localize the object, it is useful to estimate the posterior, because this information can be used to make decisions on where to sense next.

Sampling and gridding techniques have been widely used for estimation in multi-modal scenarios. For example in [9], the authors used a particle filtering technique for a similar box localization problem. The main drawback of sampling techniques is that the number of particles required for precise



(a) 1 measurement

(b) 2 measurements

(c) 3 measurements

Fig. 3. During exploration the shape of the posterior evolves as additional measurements arrive. The particles in this figure approximate high likelihood regions of the evolving posterior.

estimation explodes exponentially with the space dimensionality. Large numbers of particles lead to computation times that are unacceptable. On the other hand the problem with using fewer particles is that uniform sampling is extremely unlikely to produce any samples near the actual solution, resulting in high error of estimation. For example, suppose we are performing localization in 40 cm x 40 cm x 40 cm x 360 degrees x 360 degrees x 360 degrees space, with desired deviation of 1mm and 1 degree respectively. If we consider the 6-D sphere around the solution with radius of 1 desired deviation, the volume of this sphere is 3×10^{15} times smaller than the volume of the state space. If we utilize 1,000 particles, we are very unlikely to sample one within desired deviation of the solution.

A. Representing Regions of Space with Particles

Traditionally each particle is seen as a single point in state space, but let us consider what happens if instead each particle represents a region of the space. We will call such particles *broad* to distinguish them from single point particles. For a parameter δ, we will call the 6-D sphere with radius δ around a broad particle a δ-sphere. We will think of each broad particle as representing the entire region within its δ-sphere. If δ is large, it is clearly easy to cover the state space with even a small number of broad particles. For example, if δ is larger than the diameter of the state space, one broad particle would suffice.

Now that our particles are regions of space, we need to understand how to apply the measurement model to compute particle likelihood weights. To parameterize the measurement model relative to δ, we simply update the measurement error based on δ. We set:

$$(err_p, err_n) \leftarrow (\delta, r\delta), \tag{3}$$

where r is the ratio between actual position and normal measurement errors.

The above equations amount to “pretending” that measurement noise is inflated to be δ. Artificially inflating measurement noise is not an uncommon practice, see for example [4]. This technique allows for particles to survive better by making the likelihood weights less discriminative.

B. Scaling Series Approach

Broad particles help us cover the state space with a small number of particles, but the estimates obtained in this manner will be very imprecise as we artificially inflate the measurement noise. Therefore we run a series of successive estimations, reducing the value of δ from one step of the series to the next. The intuition behind this approach is that the first run in the series finds regions of high likelihood at a very coarse resolution. The next run focuses on the smaller subspace found by the previous run and performs estimation at a finer resolution (i.e. reduced δ). In this manner, we can keep reducing δ until it corresponds to the actual noise variance. Thus, the last run will approximate the true posterior.

Reduction in the value of δ during the series progression gradually changes the measurement model from less discriminative initially to more discriminative towards the end of the series. This technique is a variant of annealing, which has been used in other settings for Monte Carlo methods. See for example [5], where the authors applied an annealing particle filter to articulated motion capture from vision data.

C. Algorithm Details

The algorithm consists of a series of importance samplers. We start by running an importance sampler with a large value of δ (i.e. radius of initial state space V). Based on the entire dataset D, the importance sampler produces a set of particles concentrated in the region of high likelihood. This region, denoted V_1, is the union of δ-spheres around the particle set. Since V_1 is smaller than the original state space, we can cover it with smaller particles. Thus we reduce the value of δ and run a second importance sampler, but this time restrict our attention to V_1. The second importance sampler produces a new subspace, V_2, that represents the region of high likelihood for this setting of δ. We repeat the process until we reach the desired value for δ, corresponding to desired precision. Refer to Alg. 1 for a complete listing of the algorithm.

In line 2, the scaling factor $zoom$ is set so that the volume of δ-sphere is halved during scaling. We also take care to maintain a healthy density of particles in each iterative state subspace V_t. This is controlled by the desired number of particles per δ-sphere, M.

During each run of the importance sampler, the importance weights are taken to be the likelihood of measurements, $p(X|D)$, computed as described in section III and parameterized by δ in accordance with equation 3. Line 7 performs a weighted resample of the particle set to remove particles with relatively low weights (see [12] for a listing of a weighted resampling algorithm). At each iteration t we focus on state

```
Scaling_Series($V_0, M, D, \delta_{desired}$)
$\delta \leftarrow radius(V_0)$
$zoom \leftarrow 1/\sqrt[n]{2}$
$T \leftarrow log_2(Vol(V_0)/Vol(S_{\delta_{desired}}))$
for $t = 1$ to $T$ do
    $\{X_i\} \leftarrow$ Uniform_Sample_From_Subspace($V_{t-1}, M$)
    Importance_Sampler($\{X_i\}, D$)
    perform a weighted resample on $\{X_i\}$
    $V_t \leftarrow$ Union_Delta_Spheres($\{X_i\}, \delta$)
    $\delta \leftarrow zoom \cdot \delta$
end for
```

Alg. 1: Scaling Series

subspace V_t, which is the union of δ-spheres centered around the current particle set $\{X_i\}$. Thus, we need an algorithm for sampling uniformly from V_t before each importance sampler run. One of the simplest methods to generate uniform samples from V_t is based on rejection sampling (Alg. 2).

```
Uniform_Sample_From_Subspace($V, M$)
// space $V$ is represented as union of spheres $\{S_i\}$
$X \leftarrow \{\}$
for $i = 1$ to $|\{S_i\}|$ do
    for $j = 1$ to $M$ do
        sample point $x$ from $S_i$
        reject $x$ if it is in union of $S_1 \dots S_{i-1}$
        otherwise add $x$ to $X$
    end for
end for
```

Alg. 2: Uniform Sampling from Subspace

We can view the first $T - 1$ steps of the series as constructing an informed proposal distribution for the final run of importance sampling. The constructed proposal distribution is focused on the region of high likelihood of the posterior, which allows efficient estimation. One simple way to ensure that the estimate converges to the true posterior is to add some number of samples from $V_0 - V_T$ (and adjust importance weights accordingly) for the final step. This forces the proposal distribution to be non-zero everywhere in the state space, which is a sufficient condition for convergence [11]. It can also be shown that the estimates obtained in this manner are unbiased.

V. Experimental Results

We utilized polygonal models of objects. These models were constructed by hand from measurements taken with a ruler. Each model also included optimal grasping points determined by a human. Once localization is performed, grasping configuration is derived from the estimated parameters. We implemented our localization techniques in Java on 1.2GHz laptop computer. We then applied our approach to two different problems: picking up a box and turning a door handle.

A. Application 1: Locating and picking up a box

We applied our approach to the task of localizing, grasping and picking up a rectangular box (see Fig. 4). The manipulator used was a 6 DOF PUMA robot, equipped with a 6D JR3 force/torque sensor. Its end-effector included a gripper and robotic finger configuration. To simplify contact point estimation, touch sensing was performed with the robotic finger that had a spherical end.

For the over-constrained scenario, a simple active sensing procedure (specific to the box) probed 5 different sides of the box recording contact position and surface normal for each data point. Care was taken to make sure the box did not move during sensing as it would introduce considerable noise into measurements.

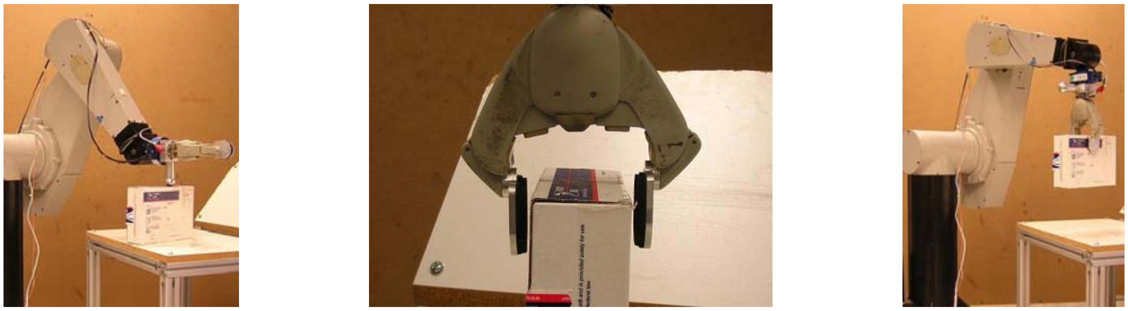
The model of the box was constructed by hand from measurements taken with a ruler. Two grasp points were manually defined on the model. Each grasp point consisted of 3 points: one for each side of the gripper and one for the wrist position. Thus each grasp point fully defined position and orientation of the gripper. After localization, the grasp point with the highest Z-coordinate was selected (Z-coordinates increase vertically upwards). The gripper orientation, position and approach vector were derived from the selected grasp point and estimated parameters. Note the precise fit required for grasping in Fig. 4(b).

The localization was performed in a 40cm x 40cm x 40cm area with unrestricted orientation (i.e. 360 x 360 x 360 degrees). Desired precision was set to 1 mm for position and 2 degrees for orientation. Sensing procedure took 30 seconds. Localization was performed in less than 1 second. We performed 30 trials on the real robot. In our experiments, localization, grasping and manipulation had 100% success ratio on completed datasets. The active sensing strategy had a 70% success ratio. Failures during sensing were due to hardware issues and motion of the object.

We also performed 1,000 simulated trials, where ground truth was easily available for evaluation of localization success and precision. In 99.8% of simulated trials our approach found the solution successfully and had an average running time of about 1 second. Since the object to be localized was symmetric, we added symmetry compensation to rule out symmetric solutions. This allowed for easy automatic identification of correct localization results. Average precision of localization was 2.1mm over the 1000 simulated trials.

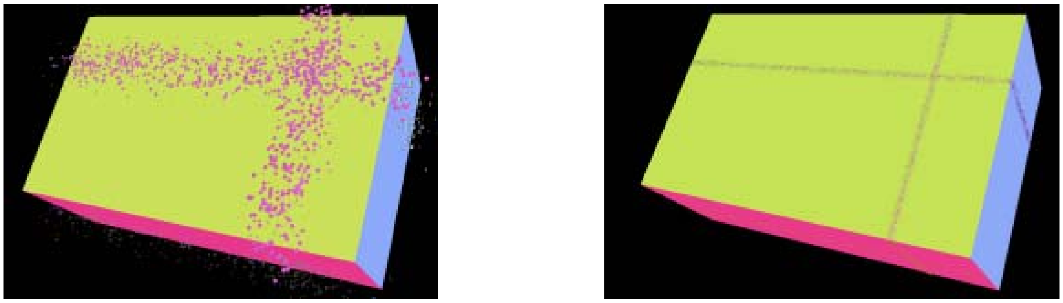
We note that our experiments were performed on a relatively simple object, consisting of only 6 faces. For more complex meshes, measurement likelihood evaluation will be linear in the number of faces. However, it is possible to implement efficiency improvements that only consider a subset of faces during measurement likelihood evaluation.

We also performed experiments for under-constrained scenarios. In this case the datasets consisted of 2 - 3 measurements from different sides of the box. For real robot experiments, we took subsets of measurements from our completed real robot trials. We verified that the estimated region included the true state of the object, as it was estimated from complete datasets. We also examined the estimated region visually to make sure it corresponded to the correct solution region in each under-constrained scenario (Fig. 5). In addition, we performed 100 simulated trials where ground truth was



(a) sensing (b) grasping (c) manipulating

Fig. 4. The stages of our box manipulation experiment. (a) Sensing the box with a robotic finger. (b) Grasping the box. The position and orientation of the box were estimated from the data obtained during sensing stage. The grasping configuration is defined as part of the box model. Note the precise fit required to perform the grasp. (c) Manipulating the box.



(a) 11mm precision (b) 1mm precision

Fig. 5. Examples of under-constrained solution estimation for datasets consisting of 2 measurements (includes symmetry compensation). (a) With δ setting of 11mm, 4,000 particles were generated by Scaling Series (b) With δ setting of 1mm 29,000 particles were generated.

available. The true state was included in resulting solution set in all 100 trials.

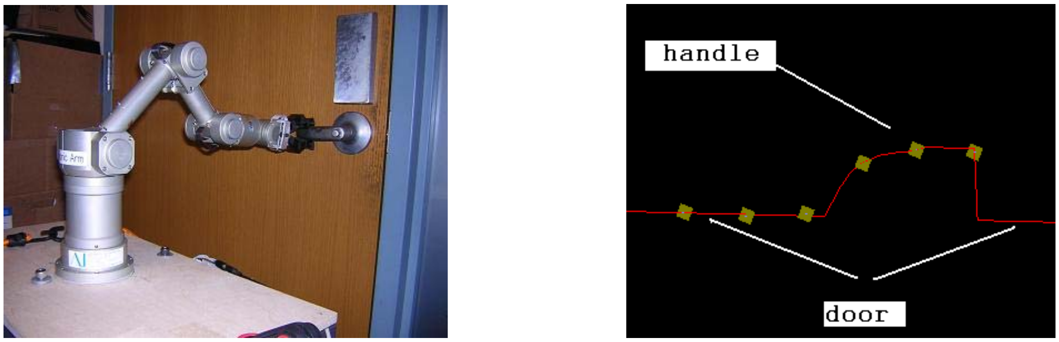
Since the number of solutions is infinite, high precision settings result in large numbers of particles. For example for a dataset consisting of two measurements, Scaling Series generated 4,000 particles for δ setting of 11mm and 29,000 particles for δ setting of 1mm. The running time increases with the number of particles generated. Operations with a few thousand particles take a few seconds, but 29,000 particles take 40-50 seconds to process. Thus it is possible to trade off precision of estimation for running time. As more measurements arrive, the solution region shrinks and higher precision can be achieved with fewer particles.

B. Application 2: Manipulating door handles for building navigation

In a second application, we carried out experiments with door handle manipulation as part of the STanford AI Robot (STAIR) project. The goal of the STAIR project is to build a robot capable of performing a broad range of tasks in home and office environments. Over the long term, the envisioned tasks include fetching a book from an office, showing guests around a research lab, tidying up after a party, and using tools to assemble a bookshelf. In order to carry out these tasks, the robot needs to navigate in home and office environments, which means being able to open doors. We do this by accurately localizing, and then manipulating, the door handle.

Once the robot navigates to the area in front of a door (using its laser sensors for approximate localization), we use tactile feedback to accurately estimate the position and orientation of the door handle. We performed experiments on a 5 DOF Harmonic Arm 6M manipulator, which has about 1mm end-effector positioning precision. (See Fig. 6(a).) The height of the handle as well as 2 orientation angles were fixed, which reduced the localization task to a 3 DOF problem. Our algorithm used a 2D model of the door that was constructed by hand using ruler measurements. Specifically, we took door handle depth measurements every 1cm along its length in a horizontal plane through the center of the handle. This gave a 2D model consisting of line segments (see Fig. 6(b)). The grasping point was defined near the tip of the door handle. The sensing used in this experiment gave only position measurements, and did not include surface normals.

For each experimental trial, the robot took 6 measurements in a 30 degree span (at $0°, 6°, \ldots, 30°$). Each data point thus consisted of range to the contact point and an orientation angle. The sensing procedure took between 1 and 2 minutes. Using these six measurements, our algorithm was able to localize the door handle in a fraction of a second. In these experiments, we restricted the dimensions of the state space (to 6cm x 6cm x 30 degrees) because of the limited operational range of the manipulator. Out of 100 independent trials, our algorithm successfully completed the sensing in 98 trials. In all of these 98 trials, our algorithm then successfully localized, grasped, and turned the door handle, and opened the door. The two failures during sensing were caused by a hardware glitch in communication with the robot.



(a) (b)

Fig. 6. (a) A 5 DOF Harmonic Arm 6M manipulator performing one of our door handle grasping experiments. (b) The 2D model of a door handle was constructed from depth measurements made with a ruler every 1cm along the length of the handle. The squares represent data points from one of our experiments.

VI. Conclusions

We have considered object localization from data obtained via tactile exploration. Bayesian posterior estimation for objects in 6DOF has been known to be computationally expensive [9]. We have proposed an efficient approach, termed Scaling Series, that approximates the posterior by samples. It performs the estimation by successively refining the high likelihood region and scaling granularity of estimation from low to high. Our approach does not utilize any special properties of the manipulated objects and can be easily applied to any object represented as a polygonal mesh. We have demonstrated its portability by applying it to two different tasks: manipulating a box and operating a door handle.

For over-constrained cases the posterior is unimodal. In these cases our approach performs the estimation in real time (about 1 second). For under-constrained cases, running time depends on the precision desired and the size of the high likelihood region. However, it is possible to trade off precision of estimation for running time. Coarse estimates can be obtained quickly when few measurements are available. As more measurements arrive, the high likelihood region shrinks and so more precise estimates can be obtained in a timely fashion.

The presented approach will apply equally well to other types of range data, e.g. data obtained with laser range finders. Also, similarly to [3] our approach can be extended to perform object identification from a set of known objects.

A number of aspects of the presented approach can be improved upon in future work. The running time of the algorithm depends linearly on the complexity of objects (i.e. number of faces in the mesh model). However, it is possible to implement efficiency improvements that only consider a small subset of faces during each measurement evaluation. Our approach rests on the assumption that the object does not move during exploration. Removing this assumption would expand the applicability of the approach, although better hardware is likely to be required. Additional enhancements will be required if the object to be localized is placed into a cluttered environment, where the correspondence problem of measurements to objects has to be solved.

Acknowledgments

We are grateful to Jaeheung Park, Irena Paschenko, Dongjun Shin, and Peter Thaulad for their help with the PUMA robot. We give warm thanks to Jimmy Zhang, David Li, Jamie Schulte, and Francois Conti for help with the STAIR robot. Our sincere thanks to David Stavens, Daphne Koller, Luis Sentis, and Vincent De Sapio for their feedback on earlier versions of this paper. We also thank anonymous reviewers for their insightful comments and one anonymous reviewer in particular for proposing a simpler version of uniform sampling algorithm presented in this version of the paper. This work was supported by the Honda Motor Company and by DARPA under contract number FA8750-05-2-0249.

References

[1] D. Kragic and H. I. Christensen. Survey on visual servoing for manipulation. Technical Report ISRN KTH/NA/P–02/01–SE, Royal Institute of Technology (KTH), 2002.

[2] J. R. Flanagan, M. C. Bowman, and R. S. Johansson. Control strategies in object manipulation tasks. *Current Opinion in Neurobiology*, 2006.

[3] W. E. L. Grimson and T. Lozano-Perez. Model-based recognition and localization from sparse range or tactile data. *Journal of Robotics Research*, 1983.

[4] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. Monte carlo localization: Efficient position estimation for mobile robots. In *Proc. of the National Conference on Artificial Intelligence*, 1999.

[5] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *CVPR*, 2000.

[6] J. Vermaak, C. Andrieu, A. Doucet, and S.J. Godsill. Particle methods for bayesian modeling and enhancement of speech signals. *IEEE Transactions on Speech and Audio Processing*, 2002.

[7] P. Slaets, J. Rutgeerts, K. Gadeyne, T. Lefebvre, H. Bruyninckx, and J. De Schutter. Construction of a geometric 3-D model from sensor measurements collected during compliant motion. In *Proc. of ISER*, 2004.

[8] M. Schaeffer and A. M. Okamura. Methods for intelligent localization and mapping during haptic exploration. In *Proc. of the IEEE International Conference on Systems, Man and Cybernetics*, 2003.

[9] K. Gadeyne and H. Bruyninckx. Markov techniques for object localization with force-controlled robots. In *In Proceeding of the 10th International Conference on Advanced Robotics (ICAR)*, 2001.

[10] A. Petrovskaya, O. Khatib, S. Thrun, and A. Y. Ng. Bayesian estimation for autonomous object manipulation based on tactile sensors. In *Proc. of ICRA*, 2006.

[11] D. J. C. MacKay. Introduction to Monte Carlo methods. In M. I. Jordan, editor, *Learning in Graphical Models*, NATO Science Series, pages 175–204. Kluwer Academic Press, 1998.

[12] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 2002.