Touch-based object localization in cluttered environments

Huy Nguyen, Quang-Cuong Pham
School of Mechanical and Aerospace Engineering
Nanyang Technological University, Singapore
Email: huy.nguyendinh09@gmail.com, cuong.pham@normalesup.org

***Abstract*—Touch-based object localization is an important component of autonomous robotic systems that are to perform dexterous tasks in real-world environments. When the objects to locate are placed within clutters, this touch-based procedure tends to generate outlier measurements which, in turn, can lead to a significant loss in localization precision. Our first contribution is to address this problem by applying the RANdom SAmple Consensus (RANSAC) method to a Bayesian estimation framework. As RANSAC requires repeatedly applying the (computationally intensive) Bayesian updating step, it is crucial to improve that step in order to achieve practical running times. Our second contribution is therefore a fast method to find the most probable object face that corresponds to a given touch measurement, which yields a significant acceleration of the Bayesian updating step. Experiments show that our overall algorithm provides accurate localization in practical times, even when the measurements are corrupted by outliers.**

I. Introduction

Accurate object localization is essential for robots to autonomously operate in cluttered, real-world environments. Yet, localization by visual sensors alone might not provide a sufficient precision for many dexterous grasping or manipulation tasks. Consider for instance the assembly a chair [1], where one sub-task consists in inserting wooden pins into the holes on a wooden stick. While object localization by commercial 3D cameras can provide at best 1-2 mm precision, the tightness of the insertion task here requires sub-millimeter precision.

One principled approach to address this problem consists in refining the pose estimate by *physically interacting* with the object: the robot would touch the object of interest (without moving it) at multiple positions, see Fig. 1. Contact positions and normals are recorded by forward kinematics (some specialized tactile sensors allow detecting the surface normal at contact [2], [3]). The estimation of the object pose from a given a number of such measurements is called the *tactile (or touch-based) localization problem*. Starting from the 1980's, a large amount of literature has been devoted to this problem and efficient methods have been developed, see *e.g.* [4], [5], [6], [2] and our Section II-A.

One major difficulty with these approaches is that, when the target object is placed within a cluttered environment, the robot might touch a different object and/or obstacles instead of the target object, generating thereby an *outlier measurement*. Existing methods are inherently fragile with respect to such outliers.

Here we address the problem of outliers classification by transposing the well-known RANdom SAmple Consensus (RANSAC) method into a Bayesian estimation framework. The algorithm consists in a series of hypothesize-and-verify iterations to select the "best" set of measurements. As these iterations involve (computationally intensive) Bayesian updates, it is crucial to improve these updates in order to achieve practical running times. Our second contribution is therefore a fast method to find the most probable object face that corresponds to a given touch measurement, which yields a significant acceleration of the Bayesian updating step.

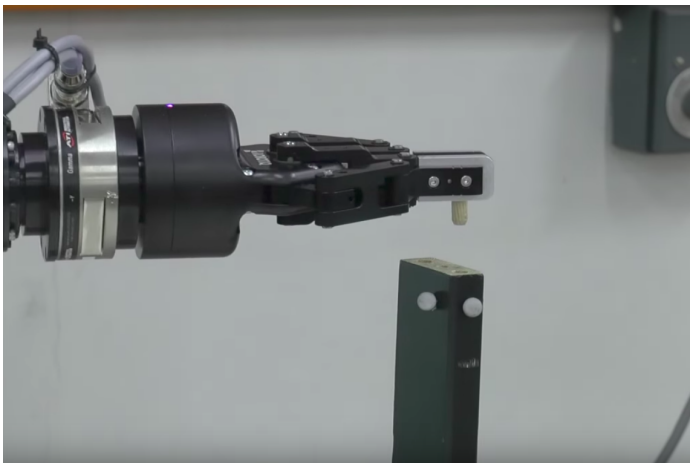

Figure 1: We study the object localization problem via touch. The photo shows the robot interacting with a wooden stick using a pin and its force/torque sensor.

The remainder of the paper is organized as follows. Section II reviews the related literature and provides the necessary mathematical background. Section III and IV present our contributions in detail. Section V reports experimental results, which show that our overall algorithm provides accurate localization in practical times, even when the measurements are corrupted by outliers. Finally, Section VI concludes and sketches some future research directions.

II. Related works and problem setting

A. Related works

Most previous works on touch-based localization are devoted to reducing the computational complexity of problem, which scales exponentially with the number of DOFs and the size of the initial uncertainty region. Based on Bayesian methods, many variants of particle filters have been proposed and proven to well suit the problem [7], [8], [6]. In particular, many approaches are capable of achieving high DOFs localization with large initial uncertainty in a timely fashion. In [2], Petrovskaya *et al.* introduced the Scaling Series method, which

achieved 6-DOF localization with large initial uncertainty of $400mm$ in position and 360 degrees in orientation. In [9], Vezzani *et al.* proposed the Memory Unscented Particle Filter that combines a modified particle filter and the unscented Kalman filter.

In these works, very often, measurements are obtained through a data collection procedure where the robot's end effector, equipped with a tactile or a force/torque sensor, approaches the object from several different directions. Though these actions can be generated randomly [2], [8] or be chosen to maximize the expected information gain [3], [10], there is no guarantee that the set of measurements does not contain *extreme erroneous* measurements, or outliers, which may result from sensor failures or the presence of other objects in the environments. These outliers, however, will shift the distribution of the object states far from the correct state, leading to a significant loss in localization precision.

To mitigate the effect of outlier measurements, one can try to determine whether the received measurement is an outlier. Subsequently, the updating step of the filtering is only performed on a relevant subset of the data. The idea of solving this correspondence problem by classifying measurements into inliers and outliers is not new. There have been many important works on 6-DOF object localization by the computer vision community [11], [12]. However, to our knowledge, 6-DOF touch-based estimation in cluttered environments has not been addressed in prior art.

B. Bayesian estimation

We start out with a quick summary of the problem: one needs to determine the pose $\boldsymbol{X} \in \mathrm{SE}(3)$ of an object $\mathcal{O}$ of known shape based on a set of tactile measurements $\boldsymbol{y}$. The object is typically represented as a polygonal mesh. The measurements $\boldsymbol{y} = \boldsymbol{y}_0, ..., \boldsymbol{y}_n$ are obtained by touching the object with the robot's end effector. Each measurement $\boldsymbol{y}_k := (\boldsymbol{y}_k^{\mathrm{pos}}, \boldsymbol{y}_k^{\mathrm{nor}})$ consists of the acquired contact position $\boldsymbol{y}_k^{\mathrm{pos}}$ and contact normal $\boldsymbol{y}_k^{\mathrm{nor}}$.

Note that we consider here the case when the measurement data sets fully contrain the problem. In other words, we assume enough data has been collected in order to sufficiently disambiguate the object pose.

Hereafter, the tactile localization problem is cast into the Bayesian framework and addressed as a nonlinear filtering problem.

The uncertain knowledge of the object is represented by a probability distribution. The object to be located is assumed to be static during the measurement collection. This assumption is commonly made [8], [2] and is realistic: for instance, the object is heavy or is fixed on a support preventing possible movements, or the contact is very slight. Hence, starting with $P(\boldsymbol{X}_0)$ – the *prior* distribution over the state $\boldsymbol{X}$ – the goal is to recursively updating the following conditional probability

$$P(\boldsymbol{X}_{t+1}|\boldsymbol{y}) = \eta P(\boldsymbol{y}|\boldsymbol{X}_t) P(\boldsymbol{X}_t). \tag{1}$$

Here $P(\boldsymbol{X}_{t+1}|\boldsymbol{y})$ is known as the *posterior*, which represent our uncertain belief about the state $\boldsymbol{X}$ after having incorporated the measurement $\boldsymbol{y}$. On the right-hand side, the first factor $P(\boldsymbol{y}|\boldsymbol{X}_t)$ is the *total measurement probability*, which encodes the likelihood of the measurement given the state (*measurement model*). The second factor $P(\boldsymbol{X}_t)$ is the *prior*, which represents our belief about $\boldsymbol{X}$ before obtaining the measurements $\boldsymbol{y}$. The factor η is a normalizing factor independent of the state $\boldsymbol{X}_t$ and needs not be computed.

As mentioned before, many variants of particle filters have been proposed and proven to well suit the nonlinear and multimodal nature of the problem. This paper builds upon these algorithms and provides an automated method to deal with outlier measurements. To illustrate its performance, we apply our method to the Scaling Series algorithm [2], which is able to solve the 6-DOF localization problem efficiently and reliably. The main idea of the Scaling Series approach is to combine Bayesian Monte-Carlo and annealing techniques. It performs multiple iterations over the data, gradually scaling precision from low to high. The number of particles at each iteration is automatically selected on the basis of the complexity of the annealed posterior.

C. The measurement model

The total measurement probability is commonly computed based on the *proximity measurement model*, where the measurements are considered independent of each other and where both the position and normal components and corrupted by Gaussian noise. For each measurement, the probability is computed based on the distance between the measurement and the object. This model was first introduced authors [2] and became popular in the literature owing to its computational efficiency.

Assume that the target object is represented as a polygonal mesh, containing a set of faces $\mathcal{F}_i$ and their corresponding normal vectors $\boldsymbol{n}_i$. Suppose that the object is at state $\boldsymbol{X}$, then the distance between a measurement $\boldsymbol{y}_k$ and the face i of the object is defined by

$$\mathrm{d}(\boldsymbol{y}_k, \mathcal{F}_i^{\boldsymbol{X}}) := \sqrt{\frac{\mathrm{d}(\boldsymbol{y}_k^{\mathrm{pos}}, \mathcal{F}_i^{\boldsymbol{X}})^2}{\sigma_{\mathrm{pos}}^2} + \frac{\mathrm{d}(\boldsymbol{y}_k^{\mathrm{nor}}, \boldsymbol{n}_i^{\boldsymbol{X}})^2}{\sigma_{\mathrm{nor}}^2}}, \tag{2}$$

where $\mathrm{d}(\boldsymbol{y}_k^{\mathrm{pos}}, \mathcal{F}_i^{\boldsymbol{X}})$ is the shortest Euclidean distance from $\boldsymbol{y}_k^{\mathrm{pos}}$ to any point on the face $\mathcal{F}_i^{\boldsymbol{X}}$, $\mathrm{d}(\boldsymbol{y}_k^{\mathrm{nor}}, \boldsymbol{n}_i^{\boldsymbol{X}})$ is the usual angle between two 3D vectors, and $\sigma_{\mathrm{pos}}, \sigma_{\mathrm{nor}}$ are the Gaussian noise variances of the position and normal measurement components respectively. Next, the distance between the measurement $\boldsymbol{y}_k$ and the object is defined as

$$\mathrm{d}(\boldsymbol{y}_k, \mathcal{O}^{\boldsymbol{X}}) := \min_i \mathrm{d}(\boldsymbol{y}_k, \mathcal{F}_i^{\boldsymbol{X}}). \tag{3}$$

For the whole set of measurements $\boldsymbol{y}$, the total measurement error is defined as

$$u(\boldsymbol{y}, \boldsymbol{X}) := \sum_k \mathrm{d}(\boldsymbol{y}_k, \mathcal{O}^{\boldsymbol{X}})^2. \tag{4}$$

Finally, the total measurement probability can be computed

as follows

$$P(\boldsymbol{y}|\boldsymbol{X}_t) = \eta_{\boldsymbol{y}} \exp\left(-\frac{1}{2}u(\boldsymbol{y}, \boldsymbol{X}_t)^2\right), \tag{5}$$

where $\eta_{\boldsymbol{y}}$ is a constant and will be taken into account during the normalization.

Notice that the considered proximity measurement model assumes in some sense that the closest point on the object causes the measurement. Alternatively, one can consider the contribution from all points to the probability of the measurement [6]. Though such an approach might be more informative, it is much more computationally intensive.

III. Accelerating Bayesian updates by efficient face selection

A. Outline of the algorithm

We note that, to compute the likelihood of a measurement, one needs to look for the face that is the most likely to cause the contact and normal measurement, *i.e.* that minimizes the distance $\mathrm{d}(\boldsymbol{y}_k, \mathcal{F}_i^{\boldsymbol{X}})$. Hence the running time of the updating step depends linearly on the number of faces in the mesh model. A key to improve the speed of the updating step then consists in accelerating the face selection process.

Here we propose to do so by pruning out faces based on a pre-computed offline angle dictionary as follows.

Offline stage: We compute the angles α_i between a reference vector $\mathbf{n}_{\mathrm{ref}}$ (*e.g.* the z-axis in the object reference frame) and the normal vectors $\mathbf{n}_i$ of all faces $\mathcal{F}_i$. The faces are then sorted according to the value of this angle.

Measurement evaluation stage: We first compute the angle $\alpha_{\boldsymbol{y}_k}$ between the measurement normal and the reference unit vector. Then, a binary search is used to find, in the list of the α_i's, the two angles α_L and α_R that best approximate $\alpha_{\boldsymbol{y}_k}$ from below and from above, respectively. Next, a face $\mathcal{F}_i$ is added to the subset for evaluation if its associated angle α_i satisfies following condition

$$\alpha_L - \delta_\alpha < \alpha_i < \alpha_H + \delta_\alpha, \tag{6}$$

where δ_α is a problem-specific threshold. Fig. 2 illustrates the face selection algorithm on a partial sphere mesh. In this case, the number of faces considered in the measurement likelihood evaluation has been reduced down to the number of faces in a ring-like region (shown in red). Finally, we compute the distance $\mathrm{d}(\boldsymbol{y}_k, \mathcal{F}_i)$ for all the faces in the subset and choose the face with the lowest distance as representative of the object in (3).

B. Algorithm parameters

One can see that if δ_α is too large, the algorithm will be too conservative and select a larger number of faces than needed, resulting in a longer running time. However, if δ_α is too small, the algorithm might not be able to return a good subset of faces for evaluation. A reasonable choice is to set $\delta_\alpha = \sigma_{\mathrm{nor}}$ in order to prune out all faces whose normals are farther than about one-standard-deviation from the measured normal.

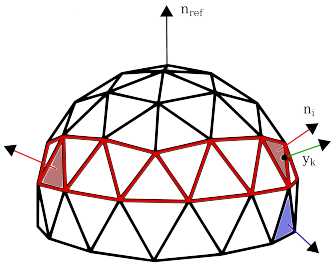

Figure 2: An illustration of the algorithm on a partial sphere mesh. The number of faces need to be considered has reduced to the number of faces on a ring-like region (shown in red).

Another factor that affects the performance of the algorithm is the choice of the reference vector $\mathbf{n}_{\mathrm{ref}}$. A good choice for $\mathbf{n}_{\mathrm{ref}}$ would induce an "even" distribution of the α_i, which can be quantified by the Shannon entropy as follows. The range $[0, \pi]$ is divided into N equal segments. Then the α_i are grouped into N bins depending on their values. The Shannon entropy of the distribution is then given by

$$S := -\sum_{c=1}^{N} p_c \log(p_c), \text{ where } p_c := \frac{\#(\text{bin } c)}{\#\text{faces}}. \tag{7}$$

Fig. 3 illustrates the computation of the Shannon entropy for two different reference vectors on the mesh model of the back of a chair.

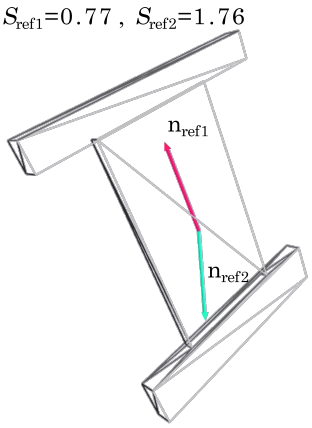

Figure 3: An illustration of the computation of the Shannon entropy for two different reference vectors on the mesh model of the back of a chair.

Finally, to choose the best reference vector, we sample random unit reference vectors, compute the Shannon entropy they induce, and choose the one with the highest Shannon entropy.

The efficient face selection technique described above can be applied equally well in many Bayesian estimations where normal and contact point measurements are available. The main idea is that the normals are discriminative and that the calculations of distances between normals are very fast, as

compared to the calculations of the distances between points and faces.

IV. Outlier classification using RANSAC

A. Outline of the algorithm

The main challenges of the object localization in cluttered environments are: (i) the correspondance problem (whether the measurement belongs to the object), and (ii) computational complexity. In particular, in cluttered environments, the measurement set is usually contaminated by extreme erroneous measurements, or outliers, which may result from sensor failure or the presence of other objects in the environments. When a particle filter receives an outlier measurement, the weight update will shift most of the weights to a few particles that are far from the correct state. This would lead to a significant loss in localization precision. Thus, to mitigate the effect of outlier measurements, one can try to determine via statistical methods whether the received measurement is an outlier or not. Then the updating step of the filtering is only performed on a relevant set of data (*i.e.* the dataset containing only inliers).

Here we adapt the popular Random Sample Consensus (RANSAC) to simultaneously classify the data into inliers (points consistent with the relation) and outliers (points not consistent with the relation). The algorithm consists of a series of hypothesize-and-verify steps and is presented in pseudo-code in Alg.1.

We start by randomly sampling a subset of m measurements called the *hypothetical inliers*. Based on the elements of this sample subset we find a best estimation (hypothesis) using a particle filter. As stated previsouly, we employ in this article the Scaling Series method [2] for its ability to deal with 6-DOF localization with large initial uncertainty in a timely fashion. Once a model has been hypothesized from this minimal subset, the remaining data points are examined to determine which agree with the hypothesis (line 6). This can be achieved by evaluating the Mahalanobis distance as in Eqn. 3 and 2. The points that fit the estimated model well are considered as part of the consensus set. The estimated model is reasonably good if sufficiently many points have been classified as part of the consensus set. Subsequently, the model may be improved by re-estimating it using all members of the consensus set, and a measure of how good the model is can be estimated following Eqn. 9 (lines 9, 10). These measures are stored and used to select the best hypothesis. This process is then repeated util a termination criterion is met.

B. Algorithm parameters

The standard termination criterion for RANSAC is based on the minimum number of samples required to ensure, with some level of confidence, that at least one of the selected minimal subsets is outlier-free. Let K be the number of iterations, K can be chosen as suggested by [cite]:

$$K := \frac{\log(1-p)}{\log(1-w^m)}, \tag{8}$$

Algorithm 1: Outlier classification

Input : $\mathcal{M}$: set of all measurements;
K: max number of iterations;
m: size of initial subsets;
d: minimum size for good subsets;
ϵ: threshold to include in inliers;
δ: threshold to consider as found.

1 **for** $i = 1$ **to** K **do**
2 maybe_inliers $\leftarrow$ m random measurements $\in \mathcal{M}$
3 remainders $\leftarrow \mathcal{M} \setminus$ maybe_inliers
4 pose_hypo $\leftarrow$ EstimatePose(maybe_inliers)
5 **foreach** measurement $\in$ remainders **do**
6 **if** d(pose_hypo, measurement) $< \epsilon$ **then**
7 Add measurement to maybe_inliers
8 **if** # maybe_inliers $>$ d **then**
9 new_pose $\leftarrow$ EstimatePose(maybe_inliers)
10 new_err $\leftarrow$ AverageDistance(new_pose, maybe_inliers)
11 **if** new_err $<$ best_err **then**
12 best_err $\leftarrow$ new_err
13 best_pose $\leftarrow$ new_pose
14 **if** new_err $< \delta$ **then**
15 break

where p is the probability we expect the algorithm to select only inliers from the input data set, w is the probability of choosing an inlier each time a single point is selected. Though this gives an estimate of the required number of iterations, it could result the very large number of iterations when the proportion of inliers is relatively small. Therefore, this should be taken only as the upper limit of iterations. In our algorithm, we terminate the process when the "model goodness" falls below a certain threshold. Here we define the model goodness as follows:

$$G := \frac{1}{I}\sum_{i}^{I} d_i, \tag{9}$$

where I is the total number of measurements in the current consensus set, d_i is the distance between the measurement i in the set and the object at estimated pose (as defined in Eqn. 3,2). In other words, given the measurement consensus set, the proposed model goodness is the *average measurement distance* between each measurement in the set and the estimated object. Since it encapsulates all the distance errors, G provides a good numerical evaluation of the localization. As a further benefit, it can be computed easily on-line at each iteration and could therefore be monitored to understand when to stop the algorithm. Nevertheless, it is worth noticing that when the measurement errors are too large (indicating by the values of σ_{nor} and σ_{pos}), (9) can be non-informative (*i.e.* G might be low even if it is associated to a wrong object pose). In that case, using the maximum number of iterations K is suggested.

Since the presented algorithm requires to perform the

Bayesian update steps many times (as part of EstimatePose), it is critical to use the efficient face selection technique presented in Section III in order to achieve practical running times.

V. Experiments

In order to validate the performance of our proposed methods, a Python implementation has been tested via simulations on differents objects and collection of measurements. All experiments were run on a machine with a 3.40 GHz processor, 4GB RAM. Our implementation is open-source and can be found online at https://goo.gl/uKaH10.

A. *Efficient face selection*

Here we evaluate the proposed face selection algorithm. As mentioned earlier, the proposed improvement procedure for the measurement likelihood evaluation can be applied equally well in the context of particle filter and many of its variants. Here, for the sake of comparison, we apply our procedure to the Scaling Series algorithm [2] and compare it with the vanilla version.

The simulation setup consisted of 3 objects: a rectangle box, the back of a chair, and a simplified mesh of a cash register (Fig. 4).

The performance of the proposed algorithm was assessed in terms of both *reliability* and *execution time*. Reliability was measured in terms of the number of successes in all the trials. A trial was considered as a fail if the estimated pose was far from the real pose. Let $\hat{\boldsymbol{X}} = (\hat{\boldsymbol{R}}, \hat{\boldsymbol{t}})$ and $\bar{\boldsymbol{X}} = (\bar{\boldsymbol{R}}, \bar{\boldsymbol{t}})$ be the estimated and the real poses, respectively. We defined the distance metrics for rotation and translation as follows:

$$\mathrm{d}(\hat{\boldsymbol{R}}, \bar{\boldsymbol{R}}) = \sqrt{\| \log(\hat{\boldsymbol{R}}^{-1}\bar{\boldsymbol{R}} \|^2}, \tag{10}$$

$$\mathrm{d}(\hat{\boldsymbol{t}}, \bar{\boldsymbol{t}}) = \sqrt{\|\hat{\boldsymbol{t}} - \bar{\boldsymbol{t}}\|^2}, \tag{11}$$

where $\|.\|$ denoted the Euclidean norm (refer to [13] for more details). In our experiment, the thresholds of $0.005\,\mathrm{mm}$ in $\mathrm{d}(\hat{\boldsymbol{t}}, \bar{\boldsymbol{t}})$ and $0.05\,\mathrm{rad}$ in $\mathrm{d}(\hat{\boldsymbol{R}}, \bar{\boldsymbol{R}})$ were used to indicate whether or not a trial was successful.

For each object, we ran both methods over 50 trials. The initial uncertainties for all objects were $50\,\mathrm{mm}$ along x, y, z and $0.5\,\mathrm{rad}$ in rotations about x, y, z. At each trial, we randomly selected a pose from the uncertainty region and used it as the ground truth. The measurements set used in the simulation tests were then drawn by sampling random points on 3D model faces. These measurements were perturbed by Gaussian noises with variances $\sigma_{\mathrm{pos}} = 2\,\mathrm{mm}$ and $\sigma_{\mathrm{nor}} = 0.09\,\mathrm{rad}$. For each trial, we drew a sufficient number of measurements to fully constrain the object estimation and used this set of measurements for both methods. The parameters for the Scaling Series algorithm were chosen as suggested in [2] and $\delta_\alpha = 0.09\,\mathrm{rad}$ was used as the threshold of face selection algorithm. After the algorithm terminated at each trial, the pose was estimated by computing the mean of the resulting distribution.

Tables I shows execution times for both methods. For all objects, it shows that the proposed algorithm greatly improves the execution time. The improvements were more significant for complex objects (*i.e.* with a larger number of faces). For example, the time difference ranges from 3.7 times for the box to 6.5 times for the cash register model. This could be expected since our algorithm focuses the computational resources only on a relevant subset of faces during the measurement likelihood evaluation. Note that both methods displayed the same level of reliability since all trials succeeded in both cases.

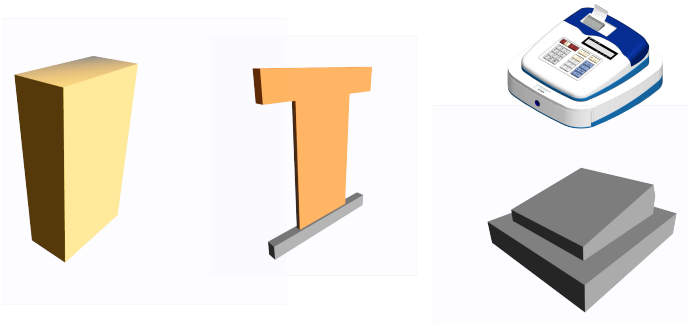

Figure 4: The 3 object models used in our experiments: a box, the back of a chair, and a simplified mesh of a cash register. Model complexity ranges from 12 triangle faces (for the box) to over 44 faces (for the back).

Objects	Face Selection(FS)	Without FS	Improvement
Box	1.06±0.21	3.93±0.79	3.7x
Back	2.71±0.65	12.03±1.61	4.4x
Register	2.07±0.27	13.50±2.27	6.5x

Table I: Execution time, in seconds, for Scaling Series method with and without the proposed face selection.

In this experiment, we did not increase the initial uncertainty to keep the running time of the Python implementation moderate. However, with larger initial uncertainty, the required number of particles and measurement evaluation steps will considerably increase. Hence the improvement on the running time will be even more significant.

B. *Object localization in a cluttered environment*

In this experiment, we consider a scene where the object to be located (the box in red) is placed near two other objects (the back of a chair and a wood stick), see Figure 5.

To sample measurements, we simulated, by a ray tracing method, approaching actions by a manipulator end-effector from random directions. In practice, even when prior information of the box is known and the approaching actions are planned carefully, a number of outliers still appear because of uncertainties arising during the execution. In our experiment, the numer of outliers ranges from 1 to 5 over a total of 15 measurements.

While some of these outliers might have been discarded by considering *e.g.* the initial probability of the object, others were very difficult to be differentiated from the inliers. Note that, in our case, no information about the other objects was

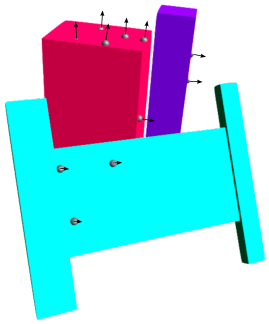

Figure 5: The object to be located (in red) is placed in a cluttered environment. Measurements including outliers are shown as spheres and normal vectors.

Methods	Trans err (mm)	Rot err (rad)
W/ Outlier Classif. (OC)	0.81±0.45	0.015±0.012
Without OC	6.64±5.21	0.1±0.06

Table II: Average distance errors in translation and rotation for both methods.

fed into the algorithm, which brings about the need for an outlier classification procedure.

The initial uncertainty about the object was assumed to be 10 mm along x, y, z and 0.3 rad in rotations about x, y, z. The measurements were generated using the same parameters as last experiment. The parameters for the Scaling Series algorithm were chosen as suggested in [2] and $\delta_\alpha = 0.09$ rad was used as the threshold of face selection algorithm.

We performed 50 trials and recorded the execution time, together with the average distances between the estimated poses and the real ones. Over all trials, the proposed algorithm succeed in locating the object of interest with an average execution time 21.3 ± 17.2 seconds. We also performed the Scaling Series method (without outlier classification) over the same data set. In this case, outliers always shift the estimated pose distribution away from the real pose and cause significantly larger errors, see Table II.

VI. Conclusion

This paper was concerned with the touch-based localization problem in cluttered environments, where outlier measurements can lead to significant loss in precision in existing approaches. Our main contributions consist in applying RANSAC to a Bayesian estimation framework and in proposing a novel face selection procedure to improve the speed of the measurement likelihood evaluation in the Bayesian updating steps. Experiments showed that our algorithm could provide, in a timely fashion, accurate and reliable localization in cluttered environments, in the presence of outliers. Future work includes experiments with real systems and further improvements of the hypothesize-and-verify scheme.

Acknowledgment

This work was supported in part by NTUitive Gap Fund NGF-2016-01-028 and SMART Innovation Grant NG000074-ENG.

References

[1] F. Suarez-Ruiz and Q.-C. Pham, "A framework for fine robotic assembly," in *IEEE International Conference on Robotics and Automation*, 2016.

[2] A. Petrovskaya and O. Khatib, "Global localization of objects via touch," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 569–585, 2011.

[3] P. Hebert, T. Howard, N. Hudson, J. Ma, and J. W. Burdick, "The next best touch for model-based localization," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 99–106.

[4] P. C. Gaston and T. Lozano-Perez, "Tactile recognition and localization using object models: The case of polyhedra on a plane," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 3, pp. 257–266, 1984.

[5] W. E. L. Grimson and T. Lozano-Perez, "Model-based recognition and localization from sparse range or tactile data," *The international journal of robotics research*, vol. 3, no. 3, pp. 3–35, 1984.

[6] C. Corcoran and R. Platt, "A measurement model for tracking hand-object state during dexterous manipulation," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4302–4308.

[7] K. Gadeyne and H. Bruyninckx, "Markov techniques for object localization with force-controlled robots," in *10th Int'l Conf. on Advanced Robotics*, 2001.

[8] S. R. Chhatpar and M. S. Branicky, "Particle filtering for localization in robotic assemblies with position uncertainty," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*. IEEE, 2005, pp. 3610–3617.

[9] G. Vezzani, U. Pattacini, G. Battistelli, L. Chisci, and L. Natale, "Memory unscented particle filter for 6-dof tactile localization," *IEEE Transactions on Robotics*, 2017.

[10] S. Javdani, M. Klingensmith, J. A. Bagnell, N. S. Pollard, and S. S. Srinivasa, "Efficient touch based localization through submodularity," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1828–1835.

[11] R. Raguram, J.-M. Frahm, and M. Pollefeys, "A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus," *Computer Vision–ECCV 2008*, pp. 500–513, 2008.

[12] C. Papazov and D. Burschka, "An efficient ransac for 3d object recognition in noisy and occluded scenes," in *Asian Conference on Computer Vision*. Springer, 2010, pp. 135–148.

[13] F. C. Park, "Distance metrics on the rigid-body motions with applications to mechanism design," *Journal of Mechanical Design*, vol. 117, no. 1, pp. 48–54, 1995.