Touch-Based Localization for Object with Coupled Geometric Datums

Shiyuan Chen[1] ,Brad Saund[2], Reid Simmons[3]

***Abstract*— In this paper, we propose a touch-based localization approach for a potentially large and complex object with multiple internal degrees of freedom. Should the robot's task only require a partial localization of the object, our method selects the approriate information gathering actions to register the desired features. We use probabilistic methods to reason over the distribution of the estimated object poses in the 6-DOF configuration space under uncertainty. We introduce the *datum-based particle filter* to handle intrinsic tolerances between each datum of the object while updating our belief of the particular target geometric features. We describe two alternative methods for the particle filter system: one using the full joint belief and the others reasonably simplifying the belief to achieve a better ability to scale. We present simulation results for both proposed methods to show the advantages of our approaches.**

I. Introduction

Many robotic tasks require precisely localizing an object, and tactile sensing is an appealing sensing modality. As motivation, we examine touch localization of a partially manufactured part. In order to handle objects with complex shape, prior information of the object shape is always required, and most previous work assumes that the geometry (CAD) model will match the real object exactly.

However, during the manufacturing and assembly processes, there are tolerances between different datums. A *datum* is defined as a geometric feature in the object that is used as the reference to define the location of other features of the object. The tolerance is the allowed deviation of the actual manufactured dimensions from the nominal designed dimensions. We assume precisely manufactured datums locally, and our method focuses on handling errors due to imprecise machining over larger distances and non-critical components, as well as assembly tolerances.

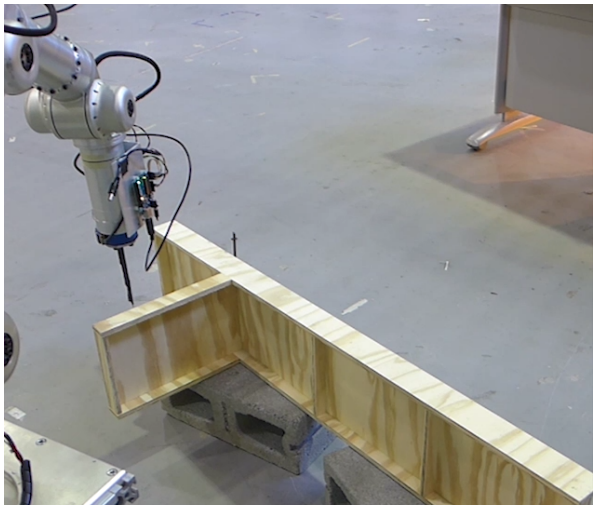
The introduction of tolerance increases the degrees of freedom (DOFs) to the system, as prior to the measurement the real dimension is always unknown. These internal DOFs can be modeled as transformations between corresponding coupled datums with uncertainty.

The goal of the localization is to estimate the pose of the object given multiple measurements obtained from probing. In 3-dimensional workspace, there are a total of 6DOFs for a rigid body. We model this update as Markov process, where we make a single measurement for each update. This model eliminates the needs of storing all of the past measurements but to incorporate them in the current state. We use a *particle filter* to numerically update the belief[1].

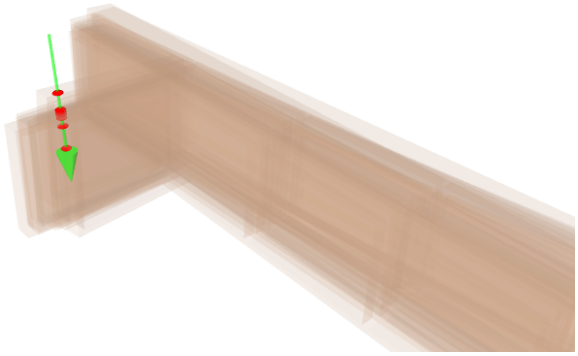
For objects with tolerance, perfectly localizing a single datum will not necessarily reduce the uncertainty of the system sufficiently to perform the desired task. On the other hand, it is usually sufficient to localize only some of the geometric features (target features). Thus we want to localize each referenced datum individually while reducing the uncertainty of the transformation relations, and ultimately localize the target features.

This paper first briefly introduces the framework of our previous work on rigid-body particle filter for high-precision localization(section III), which overcomes the *particle starvation* problem and serves as the basis for this paper[2]. We then present our work to generalize the system for objects with coupled datums (section IV).

[1]Shiyuan Chen, [2]Brad Saund, and [3]Reid Simmons are with the Robotics Institute at Carnegie Mellon University



(a)



(b)

Fig. 1: **(a)**: The robot performing a touch measurement **(b)**: The belief of the part location and measurement NEED TO UPDATE

We propose two related but different approaches for this problem. The first one maintains a single particle filter system to represent the full joint distribution of the coupled datums , while the second one simplifies the relations by assuming independence between the distribution of the internal transformation and the pose of the datums, and models the system using separate coupled particle filters.

We also re-introduce our technique of choosing informative measurement actions to accommodate object with coupled datums. To achieve this we sample many actions for each datum and select the one with the highest expected information gain. Fully predicting this is also computationally expensive, our approach involves a fast approximation for information gain that takes advantage of the discretized belief from the particle filter (section V).

We demonstrate our method both in simulation and on a real robot (section VI). Figure 1 shows our robot equipped with a touch probe, the belief distribution, and the expected points of contact during the probing (red dots).

II. Related Work

Touch is considered one of the most accurate ways of perception. Humans are capable of localizing objects with complex shape using sense of touch. However, for robots, localization using contact sensors is a classic problem, as it requires robots to reason in high-dimensional configuration space.

Recently, there have been a variety of approaches that allow robots to localize objects solely with contact sensors. Different contact sensors have been explored and developed, including basic binary sensors, 6-axis force and torque sensors[3], soft tactile sensor arrays[4], and bio-inspired fingertips[5]. Localization with laser sensors has also been used in the high-precision CNC localization, where 3D point cloud is acquired in order to estimate the transformation between the actual and planned pose [6]. Our localization approach can be generalized to these sensors which can distinguish between contact and free-of-contact states.

Particle filters have been widely used and developed since their introduction. Unlike some other Bayesian estimation approaches such as Kalman filters[7], extended Kalman filters[8] and unscented Kalman filters[9], particle filter can easily model non-Gaussian and multi-modal probability distribution. For touch localization, contact sensors are highly non-linear, and the belief can frequently become multi-modal when multiple configurations are consistent with the measurement at the same time. This advantage makes particle filter a popular approach for touch localization tasks.

However, as we will describe, particle filter will experience *particle starvation* problem for an accurate measurement and for higher-dimensional configuration space[10]. Koval introduced the Manifold Particle Filter to address this issue by implementing different sampling and weighting strategies compared to the traditional particle filters[11][12]. Instead of sampling particles from the process model and weight them based on the observation, samples are directly drawn from the contact manifold given the observation. This provides a good solution for objects with simple shape. for complex object, as in our case, sampling from the manifold will become computational expensive.

Petrovskaya combined Monte Carlo approaches with annealing as smoothing technique[13]. She introduced Scaling Series algorithms for 6-DOF global tactile localization in both full-constrained and under-constrained scenarios to overcome particle starvation by adjusting particle density depending on the complexity of the posterior. Multiple iterations through the measurement data are used and the precision of the belief is scaled from low to high in order to avoid unnecessarily precise estimates in unlikely regions of belief space.

All of the work mentioned above assumes the prior information of the object geometry matches the real piece perfectly. To the best of our knowledge, our proposed method is the first one that handles localization problems for object with coupled datums.

III. Rigid-Body Object Localization

The geometry of the object is stored in a STL file using triangular mesh. The mesh model is defined in a frame, the part frame. The pose of the object can then be defined as the transformation between this part frame and the world frame in the workspace. We assume that the object is fixed in the workspace during measurement and localization, thus the state will not change during the process.

In order to estimate the true distribution of the pose, each particle in the particle filter represents a single potential pose $x_i \in SE(3)$ of the object. For a rigid body, the pose includes both translational dimensions (x, y, z) and rotational Euler angles (α, β, γ). We represent the state as a 6D vector $(x, y, z, \alpha, \beta, \gamma)$ in the configuration space. The robot updates the particle filter based on a set of measurements $M_t = \{m_1, ..., m_t\}$ directly on the object.

A. Measurement Model

We implemented our methods on our robot equipped with both high-precision touch probe and 1D laser range finder[2].

A measurement action, $\mathcal{M}$ is defined by a start point $\mathcal{A}_p$ of the probe and a linear trajectory vector $\mathcal{A}_v$ both in $\mathbb{R}^3$. The measurement value m is the distance the start point travels along the vector until contact is made. The point of contact can then be recovered by $A_p + m \frac{\mathcal{A}_v}{||\mathcal{A}_v||}$. Measurement error exists due to sensor error and robot configuration uncertainty. For the robot arms and sensors used in manufacturing, the uncertainty added typically ranges from 0.1 to 1mm [**?**].

B. Particle Filter

At each time step, we maintain a probability distribution of poses, approximated using a particle filter. This belief distribution is also used to plan measurement operations that maximize the expected information gain. The advantage of particle filter over other Bayes estimation methods, such as original[7], extended[8] and unscented[9] Kalman filters, are that it can model multi-modal non-Gaussian distribution with non-linear measurement model in our system.

The belief of the state at time t is $bel(x_t) = p(x_t|Z_t)$. Our measurement model follows a non-linear probabilistic function of the true state: $m_t \sim p(m_t|x_{actual})$. As in all Bayesian filters, the belief $bel(x_{t+1})$ is calculated recursively as follows [1]:

$$bel(x_{t+1}) \leftarrow \eta \; p(z_t|x_t) \; bel(x_t) \tag{1}$$

with η as a normalization factor. Each new measurement value triggers an update to the belief $bel(x)$.

The traditional particle filter uses importance sampling to update particles[10], where samples are drawn based on the process model and weighted by the observation. However, for precise measurement and higher dimensions, the chance that a particles is consistent with the measurement is extremely low, thus few of them will survive during the update[12]. In order to alleviate this problem, one will require the number of particles to be exponential to the number of dimensions while maintaining high density in the configuration space.

Instead, our approach updates the particles based on rejection sampling[2]. At each update step t, the continuous prior belief is estimated using Gaussian Mixture Models by applying a Gaussian kernel to each particle with the kernel covariance proportional to the covariance of the particles. New samples x_t are then drawn directly from this estimated prior. For each sample, we follow the rejection sampling manner to accept the sample with probability $p(z_t|x_t)$. This probability is defined by the actual distance between the measurement and the object: if the object with pose x_t is sufficiently far away from the measurement, the sample will be rejected.

The computational cost of rejecting samples is balanced by the use of voxelized distance field[14], where the distance between each voxel in 3D workspace and the object is precomputed relative in the particle frame at the beginning of each update. As the measurement m_t is relative in the world frame, during the rejection sampling, the measurement is transformed to the particle frame before lookup to the distance field, as each sample represents a transformation relation $T(x_t)$ from world frame to the particle frame. The minimal unsigned distance $dist_u(m_t, S)$ between each measurement m_t and the object $S(x_t)$ can be obtained directly[2]:

$$dist_u(M_t, S(x_t)) = D_f(T(x_t)^{-1} M_t) \tag{2}$$

$$dist(M_t, S(x_t) = \begin{cases} dist_u(M_t, S(x_t) - r_p, & \text{if } M_t \notin S \\ -dist_u(M_t, S(x_t) - r_p, & \text{otherwise.} \end{cases} \tag{3}$$

where D_f is the precomputed distance field, r_p is the radius of the spherical tip of the touch probe. For 1D range finder, $r_p = 0$.

New samples x_t are accepted based on the signed distance $dist(M_t, S(x_t))$ between the object and the measurement on the object. We continue rejection sampling until we achieve an ideal number of particles for this update. These particles represent updated belief $bel(x_t)$ of the pose.

[1]The full update of a Bayesian filter also includes a process model. Our assumption of a fixed object yields the static process model and this simpler formulation

IV. Datum Based Particle Filter

The particle filter localization method presented above assumes that the object matches its CAD model exactly. However, this is usually not the case, due to tolerances in the manufacturing and assembly processes. To handle this, features are located on parts not with respect to the part frame, but with respect to datums, edges and surfaces on the actual, as built, part. Incorporating the notion of datums, and their relationships, is complicated because the relationship between the datum and the CAD model contains uncertainty. Thus, measuring one section of the assembly provides only uncertain updates to other sections, depending on the specified tolerances. We refer to this as semi-rigid parts. We will also refer to the datums and features as *sections* of the part. The problem is that we want measurements on one section to update the position beliefs of other sections, and also that we want to determine the information gain on a section from measuring a different section.

A. Datum Representation

In general, we treat the overall part as composed of separate sections. The problem is to precisely localize some feature (e.g. a location to drill a hole) with respect to given datums (other sections). For instance, Figure ??? shows a hole feature referenced to the top and right edges datums of our Boeing-supplied piece. The goal is localize each datum in such a way as to localize the feature. For instance, it is necessary to localize the top edge's vertical position and orientation, but not its in-page or horizontal position. The issue is to maintain the transformation relations between sections so that when making one measurement the distributions of all the associated sections are updated.

Instead of using a single CAD model for the whole object, separate mesh files are used for each section under the assumption that each physical section will match its CAD model precisely.

We introduce two approaches; the first explicitly represents the joint probability distribution between the sections, and the second represents separate, independent probability distributions for each section. We continue to assume that the measurements will be made by point-based sensors (touch probe or 1D laser range finder).

Throughout the rest of this paper we will use the following notation. X_t^k is the set of N particles representing the belief of section k at time step t. Frequently t is omitted when implicit. Each particle is a configuration for a single section $X^k = \{{}^j x^k\}_{j=1}^N$. The omission of k indicates all necessary particles to represent the belief of the part: $X = \{X^k\}$ and $x = \{x^k\}$.

B. Geometry Relations

Geometry Relations are defined between two or more datum sections. The existence of the tolerance introduces uncertainty to the relations, where we can model the relation as

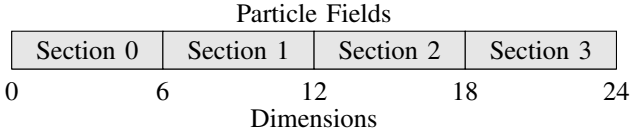


Fig. 2: A 24-dimensional particle for object with 4 sections

a transformation in the configuration space between the pose of each section. More generally, the conditional probability $p(x_t^i|\mathbf{x_t})$ represents our belief of the pose of section i given the poses of its referenced sections $\mathbf{x_t} = \{x_t^k\}_{k\neq i}$.

A measurement is made on a single section once at each step by touch probe or laser sensor. Let the measurement on the section i at step t is $\mathcal{M}_t^i$, the posterior of section i is $p(x_t^i|\mathbf{x_t}, M_t^i)$. We assume that when a measurement was performed, we can detect on which section the measurement was made. This assumption is reasonable, since the uncertainty of the prior belief is very small, otherwise, we can always perform the particle filter for the whole object to get better estimate before considering the object as a combination of coupled sections.

The conditionals are only a prior estimate of the geometry transformations. During updates, both the transformations between sections and the pose of sections itself need to be refined based on the measurement.

C. Full State Representation

The first method we present maintains the full joint probability distribution between sections using a single set of full-state particles: $X_t = \{^j x_t\}$.

Defining transformations of the sections in a graph is natural. The sections which are related by the geometry constraints can be modeled as nodes in the graph, while the constraints serve as edges. An example is Bayesian network, where the geometry relations between a section and its referenced datum are represented by directed edges. The advantage for this representation is that we can restore the full joint distribution using fewer edges within Markov blanket. However, for non-parametric continuous distribution, it is difficult to apply belief propagation approaches to update the whole graph given an observation on a section.

Instead, for the object in 3D workspace with n sections being tracked, we explicitly maintain a full $6n$-dimensional state space. Thus, each particle represents a potential combination of the poses of all of the sections (see Fig. 2), which is drawn from the prior joint distribution $p(\mathbf{x})$.

Although it seems at first that the high dimensional state space will require a huge number of particles to approximate the true distribution, in practice, we have not found this to be the case. Largely, this is because we assume that the tolerances are small compared to the uncertainty of the pose, so the particles tend to cluster in a small subset of the full state space. In addition, before each update, we will manually create the continuous prior by applying Gaussian kernels[2]. Note that if there were no uncertainty in the transformation between sections, then the system reduces to a single 6-DOF state space.

Algorithm 1 Full State Particle Filter

Input: number of particles N and number of sections n
Input: particles $\mathbf{X_{(t-1)}} = \{^\mathbf{j}\mathbf{x_{t-1}}\}_{j=1}^N$ and observation m_t^i
Input: list of mesh files $S = \{S_k\}_{k=1}^n$
Output: particles $\mathbf{X_{(t)}} = \{^\mathbf{j}\mathbf{x_t}\}_{j=1}^N$

```
build distance field D_f(p)
j ← 1
while j ≤ N do
    x ~ p(x|X_(t-1))                          ▷ x = {x^k}_{k=1}^n
    dist ← D_f(T(x^i)^{-1} m_t^i)
    if dist ≤ ξ then
        ^j x_t ← x
        j ← j + 1
    end if
end while
```

At any time step t we have a belief of the state $bel(\mathbf{x_t}) = p(\mathbf{x_t}|Z_t)$, x_t^i represents the pose of section i. As for rigid part localization, the belief is estimated by a single set of particles and updated using rejection sampling.

Instead of applying Gaussian kernel in 6D configuration space, the continuous prior is estimated by Gaussian Mixtures in the full state $6n$-dimensional configuration space with the kernel covariance proportional to the covariance of the sampled states. This is achieved by applying kernel density estimation techniques. One popular method for the bandwidth selection is the Silverman's rule-of-thumb estimator[15]. Other techniques are discussed here in more detail[16].

New full-state samples $\mathbf{x_{t+1}}$ are drawn from the estimated $\hat{bel}(\mathbf{x_t})$ directly. To update the particles based on the measurement M_{t+1}^i on section i, the 6D pose x_{t+1}^i for this section are extracted from each full-state sample $\mathbf{x_{t+1}}$. Equation 2 and 3 are then used for the rejection sampling, except that now we use the extracted x_{t+1}^i to apply the transformation from the world frame to the frame attached on the section: if $dist(M_{t+1}^i, S_i(x_{t+1}^i))$ is sufficiently large, this full-state sample $\mathbf{x_{t+1}}$ is rejected, otherwise accepted. Note that for this time, the full-state sample $\mathbf{x_{t+1}}$ is accepted based on $p(M_{t+1}^i|x_{t+1}^i)$, and the CAD model used for the distance field is the mesh for that particular section S_i (shown in Algorithm 1). For simplicity, the pseudo-code doesn't include the step to adaptively adjust the sample size based on KL-divergence[2][17].

Each time we make a measurement on a section, we update the estimate of the full joint distribution for all sections based on how well the estimate is consistent with that measurement. This allows us to update all sections for one single measurement based on the transformation information in the prior belief without explicitly implementing the transformation.

D. Independent State Representation

The first approach described above keeps track of the updates of both pose of each datum and the their transformations by maintaining a full state representation of the

Algorithm 2 Independent State Particle Filter

Input: number of particles N and number of sections n
Input: particles $\mathbf{X_{t-1}} = \{X_{t-1}^k\}_{k=1}^n$ and observation m_t^i
Input: mesh $S = \{S_k\}_{k=1}^n$, transformations $\{p(T_i^k)\}_{k=1}^n$
Output: particles $\mathbf{X_t} = \{X_t^k\}_{k=1}^n$

build distance field $D_f(p)$ for section S_i
for $k = 1, ..., n$ **do**
  $j = 1$
  while $j \leq N$ **do**
    $x \sim p(x^k | X_{t-1}^k)$
    $T_i^k \sim p(T_i^k)$
    $\tilde{x} \leftarrow T_i^k \times x$
    $dist \leftarrow D_f(T(\tilde{x})^{-1} M_t^i)$
    if $dist \leq \xi$ **then**
      ${}^j x_t^k \leftarrow x$
      $j \leftarrow j + 1$
    end if
  end while
end for

distribution. An alternative is to maintain the probability distribution for each section separately. Instead of using a full high-dimensional particle filter for the object, individual 6-dimensional particle filter is used for each single section under the assumption that the transformations between sections are fixed and independent. While this loses information compared with the full joint belief, in practice this loss is acceptable.

As for the rigid body particle filter, a sample in a particle filter for a section represents a $SE(3)$ pose of this section. The transformation information between different sections are defined explicitly. For a two-section part, our prior belief on the transformation between x_i and x_j is $bel(T_i^j)$, which is a distribution over $SE(3)$ transformations. When there is a measurement on a section, the particle filters for all related sections are updated at the same time. Given a measurement M_i on the section i, the updated belief should become $p(x_j | T_j^i, M_i)$ for a related section j.

The update to the belief $bel(x_i)$ itself given a measurement performed on section i is identical to the particle filter update for the rigid object, since T_i^i is the identity with probability 1, the updated belief can be written as:

$$bel(x_t^i) = p(x_t^i | T_i^i, M_t^i) = p(x_t^i | M_t^i) \tag{4}$$

For a section j that references section i ($i \neq j$), each new sample x_j is drawn from the prior of its corresponding particle filter. x_j is then transformed from the frame of section j to the frame of section i:

$$\tilde{x^j} = T_i^j \times x^i \tag{5}$$

where $T_i^j \sim bel(T_i^j)$is the sampled transformation from the distribution $bel(T_i^j)$. As the measurement was performed on section i, we must use the geometry of i rather than j. Relatively in the frame of section i, the agreement between the measurement M_t and the geometry $S_i({}^j x)$ of the section i is computed. The sampled particle is accepted with probability $p(M_i | x^{h\prime})$. The above process is repeated until the desired number of particles have been accepted (shown in Algorithm 2).

V. Predicting Effective Measurement Actions

Performing measurements is expensive, so we choose the measurement action that provides the most *information gain* on our goal feature. We treat each action as a probabilistic decision over a set of particles approximating our belief of the goal feature. The best measurement may not be on the goal feature, and it may be impossible to even measure the goal feature directly. Our formulation predicts the information gain on the goal feature for measurement directly on the goal feature, and indirectly for measurement on datums or other sections of the part.

A. Information Gain

Given X^G, a set of particles representing the belief of the goal feature, the *information gain* from a measurement action $\mathcal{M}$ is defined as the expected reduction of entropy.

$$IG(X^G | \mathcal{M}) = H(X^G) - H(X^G | \mathcal{M}) \tag{6}$$

Where $H(X^G)$ is the entropy of the particles and $H(X^G | \mathcal{M})$ is the entropy of the particles conditioned on the measurement.

The entropy of a discrete distribution of states depends only on the probabilities of each state occurring.

$$H(X^G) = -\sum_i w_i \log w_i \tag{7}$$

where w_i is the weight of particle i.

To calculate the conditional entropy, $H(X^G | \mathcal{M})$, the measurement action $\mathcal{M}$ is simulated on the part distribution. Performing a measurement action yields a continuous, probabilistic value. Samples are drawn from this distribution:

$$m_{i,j} = Simulate(\{\mathcal{M} + \delta_j\}, {}^i X^G) + \eta_j \tag{8}$$

where δ_j is the deviation from the nominal measurement action, $Simulate$ computes the value for a measurement action applied to the part in a specific configuration, and η_j is measurement noise.

$H(X^G | \mathcal{M})$ is calculated by dividing the continuous values $m_{i,j}$ into discrete bins, b_k. The bin size is chosen as one standard deviation of the measurement error. The conditional entropy of this measurement action is then:

$$H(X^G | \mathcal{M}) = \sum_j p(b_k)\ H(X^G | b_k) \tag{9}$$

Where $p(b_k)$ is the prior probability that this measurement will fall into bin b_k and $H(X | b_k)$ is the entropy of the particles within bin b_k. The likelihood of a bin is computed

by summing the weights of the measurements in that bin. Define the weight of the bin as:

$$W_k = \sum_{i,j} \mathbb{1}(m_{i,j} \in b_k) \cdot w_i \tag{10}$$

$$p(b_k) = \frac{W_k}{\sum_{i,j} w_i} \tag{11}$$

Given a bin, the probability of a specific particle is:

$$p({}^iX|b_k) = \frac{\sum_j \mathbb{1}(m_{i,j} \in b_k) \cdot w_i}{W_k} \tag{12}$$

Then the entropy of the bin can be calculated:

$$H(X|b_k) = -\sum_i p({}^iX|b_k) \log(p({}^iX|b_k)) \tag{13}$$

Adaptations are made to accommodate our two representations as described next:

B. Information for Full State Representation

The Full State Representation does not maintain a set of particles over just the goal feature, but rather each particle, X^F, represents the full $6 \times n$ state. Using these full state particles, X^F, for X^G above provides a good metric for localizing every section of the part, but a poor metric for localizing the goal feature. This metric would often suggest to perform measurements on non-datum features that are irrelevant to the location of the goal feature. The error in this metric is due to

$$H(X^G|\mathcal{M}) \neq H(X^F|\mathcal{M}) \tag{14}$$

To calculate $H(X^G|\mathcal{M})$ using X^F we use the measurement action to sort the particles into bins. Within each bin, if multiple full state particles have similar goal feature states, we treat these particles as identical when computing entropy. NEED TO DESCRIBE HOW WE COMBINE PARTICLES

C. Information for Independent State Representation

The Independent State Representation does maintain the set of goal feature particles, X^G, however additional steps are needed when computing Eq. 8. When simulating $\mathcal{M}$, the robot will measure some section, $\mathcal{S}$, of the part. Simulating the measurement using $X^{\mathcal{S}}$ is straightforward, but leads to computing $IG(X^{\mathcal{S}}|\mathcal{M})$, which is not the desired metric.

A temporary set of particles $\tilde{X}^{\mathcal{S}}$ is created by sampling transforms $T_G^{\mathcal{S}}$ and applying them to X^G. $\tilde{X^{\mathcal{S}}}$ is used in Eq. 8 to generate sample measurement values $\tilde{m}_{i,j}$, which are used in the calculation for bin entropy $H(X|b_k)$. However, measurements $m_{i,j}$ are also calculated using $X^{\mathcal{S}}$. These measurements are used to calculate the bin probabilities $p(b_k)$.

VI. Experiments

VII. Conclusions and Future Work

References

[1] S. Thrun, B. Wolfram, and F. Dieter, "Probabilistic Robotics," pp. 1999–2000, 2000.

[2] B. Saund, S. Chen, and R. Simmons, "Touch Based Localization of Parts for High Precision Manufacturing," vol. 6, no. 1, 2017, pp. 29–33.

[3] A. Del Prete, F. Nori, G. Metta, and L. Natale, "Control of contact forces: The role of tactile feedback for contact localization," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4048–4053.

[4] F. L. Hammond, R. K. Kramer, Q. Wan, R. D. Howe, and R. J. Wood, "Soft tactile sensor arrays for micromanipulation," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 25–32.

[5] J. A. Fishel and G. E. Loeb, "Sensing tactile microvibrations with the biotaccomparison with human sensitivity," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*. IEEE, 2012, pp. 1122–1127.

[6] M. Rajaraman, M. Dawson-Haggerty, K. Shimada, and D. Bourne, "Automated workpiece localization for robotic welding," in *Automation Science and Engineering (CASE), 2013 IEEE International Conference on*. IEEE, 2013, pp. 681–686.

[7] R. E. Kalman *et al.*, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[8] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Journal of basic engineering*, vol. 83, no. 3, pp. 95–108, 1961.

[9] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *AeroSense'97*. International Society for Optics and Photonics, 1997, pp. 182–193.

[10] S. Thrun, D. Fox, and W. Burgard, "Monte carlo localization with mixture proposal distribution," *Proceedings of the National Conference on*, pp. 859–865, 2000. [Online]. Available: http://www.aaai.org/Papers/AAAI/2000/AAAI00-132.pdf

[11] M. C. Koval, N. S. Pollard, and S. S. Srinivasa, "Pre- and Post-Contact Policy Decomposition for Planar Contact Manipulation Under Uncertainty," pp. 1–27, 2011.

[12] M. C. Koval, M. R. Dogar, N. S. Pollard, and S. S. Srinivasa, "Pose estimation for contact manipulation with manifold particle filters," *IEEE International Conference on Intelligent Robots and Systems*, no. Section 3, pp. 4541–4548, 2013.

[13] A. Petrovskaya and O. Khatib, "Global Localization of Objects via Touch," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 569–585, 2011. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5784199

[14] P. F. Felzenszwalb and D. P. Huttenlocher, "Distance transforms of sampled functions," *Cornell Computing and Information Science Technical Report TR20041963*, vol. 4, pp. 1–15, 2004. [Online]. Available: http://ecommons.library.cornell.edu/handle/1813/5663

[15] B. W. Silverman, "Density Estimation for Statistics and Data Analysis," *Biometrical Journal*, vol. 30, no. 7, 1986.

[16] S. J. Sheather *et al.*, "Density estimation," *Statistical Science*, vol. 19, no. 4, pp. 588–597, 2004.

[17] D. Fox, "Adapting the sample size in particle filters through KLD Sampling," *Intl Jour of Robotics Research*, vol. 22, no. 12, pp. 985–1004, 2003.

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

[19] K. Hsiao and L. P. Kaelbling, "Task-Driven Tactile Exploration," *Proceedings of Robotics Science and Systems*, 2010. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.9204{&}rep=rep1{&}type=pdf

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

[21] S. Javdani, A. Krause, Y. Chen, and J. A. Bagnell, "Near Optimal Bayesian Active Learning for Decision Making," vol. 33, 2014.

[22] H. J. Pahk and W. J. Ahn, "Advanced manufacturing Technology," vol. 3, pp. 442–449, 1996.

[23] Z. Xiong, M. Y. Wang, S. Member, and Z. Li, "A Near-Optimal Probing Strategy for Workpiece Localization," vol. 20, no. 4, pp. 668–676, 2004.

[24] H.-T. Yau and C.-H. Menq, "An automated dimensional inspection environment for manufactured parts using coordinate measuring machines," *International Journal of Production Research*, vol. 30, no. 7, pp. 1517–1536, 1992. [Online]. Available: http://www.informaworld.com/openurl?genre=article{&}doi=10.1080/00207549208948105{&}magic=crossref||D404A21C5BB053405B1A640AFFD44AE3

[25] D. Fox, W. Burgard, and S. Thrun, "Active Markov localization for mobile robots," *Robotics and Autonomous Systems*, vol. 25, no. 3-4, pp. 195–207, 1998.

[26] A. Petrovskaya and A. Y. Ng, "Probabilistic mobile manipulation in dynamic environments, with application to opening doors," *IJCAI International Joint Conference on Artificial Intelligence*, pp. 2178–2184, 2007.

[27] C. Stachniss, G. Grisetti, and W. Burgard, "Information Gain-based Exploration Using Rao-Blackwellized Particle Filters." *Robotics: Science and Systems*, pp. 65–72, 2005.