Active Learning with Query Paths for Tactile Object Shape Exploration

Danny Driess Peter Englert Marc Toussaint

***Abstract*— In the present work, we propose an active learning framework based on optimal query paths to efficiently address the problem of tactile object shape exploration. Most previous approaches perform active touch probing at discrete query points, which leads to inefficient touch-and-retract motions. In contrast, in this paper we propose to query information efficient sliding paths instead of only touch locations. This is realized by three components: A Gaussian process implicit surface model represents the shape and uncertainty of the object. A compliant task/force controller framework fuses the information of this GP model into the parameterization of its tasks, which enables the robot to slide over the unknown object safely and robustly. Thirdly, we develop two strategies to solve the proposed active path querying learning problem. Sliding along those query paths not only creates more dense data than touch probing, but additionally greatly reduces the uncertainty of the object. We demonstrate the effectiveness of our proposed framework both in simulation and on the PR2 robot platform. Furthermore, it is shown that our methodology can be extended to other learning tasks, such as finding a desired surface normal on an unknown object, e.g. for pushing.**

I. Introduction

The autonomous exploration and manipulation of a priori unknown objects in uncertain environments is still a major challenging problem in robotics. Several studies [1], [2] have shown that humans generate their robustness and dexterity in manipulating objects mainly through tactile perception. The information humans can obtain by the sense of touch is manifold, including the geometric shape of objects, their texture, compliance, weight etc. Using this information alone allows humans to recognize and manipulate objects rapid- and accurately. Therefore, there is high interest to mimic these behaviors on robots. However, transferring tactile perception to robots is not trivial. First, suitable touch sensors are necessary. As tactile perception can only provide local information about properties of the environment, gathering the sensor stimuli is an *active* process. This active exploration through tactile feedback inherently requires contact centered interaction with the environment. Advanced control frameworks are required that are able to safely exploit contacts and focus on this interaction. Finally, learning methodologies are not only crucial to interpret the data from tactile sensors (for example to recognize objects), but are also inherently necessary to actively guide the explorative movements of the robot, e.g. following an information-gain objective, to perceive the tactile information at all.

For manipulation tasks such as grasping, especially knowledge about the shape of an object is important. Most previous research [3], [4], [5], [6] on applying active learning methodologies to the problem of tactile shape reconstruction is based on iteratively selecting discrete locations where the robot should touch the unknown object in order to improve an underlying surface model at most uncertain regions.

Danny Driess, Peter Englert and Marc Toussaint are with the Machine Learning and Robotics Lab, University of Stuttgart, Germany.

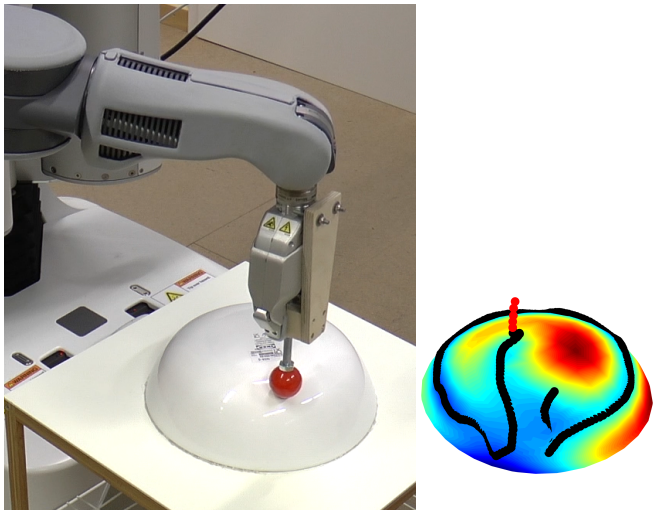


Fig. 1. PR2 robot explores an unknown object, a salad bowl, by sliding along paths obtained by our proposed active learning methodology with the greedy strategy (section VI-A). Right: current Gaussian process implicit surface estimation and exploration path. Red high variance, blue low.

However, we think that point querying has disadvantages. Even with active touch point selection, the touch-and-retract exploration procedure can be inefficient, since the robot has to move a lot. Instead of moving away from the object after each touch, *maintaining* the contact reduces not only the necessary movements, but also the shape uncertainty more efficiently based on tangent information. In addition, staying in contact makes the problem of generating collision free paths to new touch points less reliant on prior knowledge.[1] Therefore, we belief that an efficient tactile exploration strategy should exploit contacts by *sliding* over the surface.

To this end, we propose a novel active learning methodology that queries informative exploration *paths* instead of only discrete points. The unknown object is represented with a Gaussian process implicit surface model, which enables us to take the uncertainty of the surface model into account. Sliding along an unknown surface, however, requires a suitable control concept, which cannot be treated separately. Therefore, we consider how the surface representation and the exploration strategy can be incorporated into the controller such that the exploration is robust and safe.
To summarize, the main contributions are

- A novel active learning formulation that queries complete paths instead of single points while taking a priori unknown constraints into account.

[1] Whereas contact with the touch sensor is desired, collisions between the unknown object and other parts of the robot while probing a new touch point have to be avoided.

- Two algorithms to solve this active learning problem.
- The application of these methods for tactile object shape exploration by sliding along the query paths.
- A compliant task/force controller concept that integrates the surface estimation into its tasks and their parameterizations.
- An acquisition function which enables the robot to find desired surface normals on the unknown object.

After reviewing related work in section II, the background on Gaussian process implicit surfaces is given in section III. The sections IV, V, and VI present the main contributions of our work. First, the active learning with query paths problem is formulated in section IV. Then we discuss how the surface model can be incorporated into the proposed control concept, section V. The actual exploration strategies that solve the active learning problem are presented in section VI. Finally, experiments in section VII demonstrate our approach in simulation and on a real robot, the PR2. Here, we show that our active learning with query path sliding approach is able to outperform existing point querying methods.

II. Related Work

The various aspects of tactile perception and exploration are investigated by different subfields in robotics. So-called tactile servoing deals with tactile sensing and control of a robot for sliding movements [7], [8], [9]. However, here the explorative movements are programmed.

A. Active Tactile Shape Exploration by Point Querying

In the context of active learning, the question of how a robot can autonomously explore an object is addressed. The authors of [3], [4], [5], [6] all deal with active tactile shape exploration by touch probing the unknown object at discrete locations. Dragiev et al. [3] introduced the idea of using the variance of the Gaussian process implicit surface model to guide the robot to locations where the model is most uncertain. Their goal is to use these shape models for grasping. To generate the motions they assume that the robot can sense contact anywhere on its skin and the surface model is initialized with a random sample from the object. Additionally, it is proposed how the surface model can be integrated into joint trajectory optimization. When the robot senses contact, the execution of the trajectory is stopped, the surface model is updated, the trajectory is reversed and process starts again. Björkman et al. [4] initialize the GP implicit surface model with visual data and the robot then enhances this model similarly to [3] by grasping the object at locations where the model has the highest uncertainty. Instead of implicit surfaces, Yi et al. [5] represent the height of an object over a rectangular space. This exploration area must specified by expert knowledge beforehand, which we belief limits the applicability of this approach in uncertain environments. They draw the connection between acquisition functions known from Bayesian active learning and shape exploration. [6] differs from [5] in that they not only use Gaussian process regression for modeling the height of the object, but additionally learn a classifier that distinguishes object/no object to bias the exploration towards the boundary. This classifier also requires a heuristic based on the z-coordinate. A limitation of [6] and [5] is that most objects in the world cannot be represented sufficiently as height maps.

To summarize, existing approaches to tactile shape exploration are based on point queries, require assumptions which we would like to avoid and lead to inefficient repeated touch-and-retract motions. Little effort has been made to incorporate exploration strategies into closed-loop controllers, which we consider is crucial for the success of tactile exploration.

B. Active Path Planning in Field Robotics

In field robotics, a robot often has to travel large distances. Therefore, active learning methodologies in this area address the problem of information efficient path planning with minimal length. Hollinger et al. [10] consider information efficient view planning for underwater inspection. The queried view points are, however, also discrete in nature and no observations along the travel paths are considered.

In contrast, Marchant et al. [11] introduced the idea of continuous informative path planning as an extension to Bayesian optimization. This work is most related to our proposed method. They aim at querying paths that maximize the acquisition function integrated along the path. However, the key difference to our problem of tactile shape exploration is that in field robotics no environmental constraints, especially no unknown ones, are present and therefore the robot can travel along the optimized paths directly, which is not possible in our setting.

In summary, path queries in the context of active learning have rarely been studied. What we propose is, to our knowledge, the first to leverage active learning principles to design sliding paths for tactile object shape estimation.

III. Background on Gaussian Process Implicit Surface Representations

The outcome of the exploration process should be a model that contains information of the object like its shape, surface normals etc. This representation is also relevant to guide the exploration itself. Implicit surfaces are a powerful way to represent arbitrarily shaped surfaces with complex topology. They also provide geometric information like surface normals or curvature easily. An implicit surface $\mathcal{S} \subset \mathbb{R}^3$ is defined as the zero level set of a function $F : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}$

$$\mathcal{S} = \left\{ \mathbf{x} \in \mathbb{R}^3 | F(\mathbf{x}) = 0 \right\}. \tag{1}$$

During the exploration, the robot collects data in form of a tactile point cloud $\mathscr{D} = \{(\mathbf{x}_i, c_i)\}_{i=1}^{w}$ with the position $\mathbf{x} \in \mathbb{R}^3$ where the robot has sensed an object $c = 0$ or not $c = 1$. In contrast to touch sampling approaches, the data collection takes place with a certain sampling rate over the whole exploration process, including both on- and off-surface observations, which provides more dense data than touch probing. Building an implicit surface from this tactile point cloud can be seen as a regression problem. Williams et al. [12] and Dragiev et al. [13] have proposed to use Gaussian processes (GP) [14] as the regression method for implicit surfaces. A GP has the advantage that it not only approximates F, but additionally provides an uncertainty measure of the model given the data in a probabilistic way,

which makes GPs very suitable for the purpose of this project. For a given positive definite, two times differentiable kernel $k : \Omega \times \Omega \to \mathbb{R}$, $\Omega \subset \mathbb{R}^3$ and a constant prior mean $m \in \mathbb{R}$, a GP models the probability $P(F(\mathbf{x})|\mathscr{D})$ of the implicit surface function F conditioned on the tactile data $\mathscr{D}$ as a Gaussian distribution around the mean function

$$\mu_F(\mathbf{x}) = m + \boldsymbol{\kappa}(\mathbf{x})^T \mathbf{b} \tag{2}$$

and variance function

$$\mathbb{V}_F(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \boldsymbol{\kappa}(\mathbf{x})^T \mathbf{G}^{-1} \boldsymbol{\kappa}(\mathbf{x}) \tag{3}$$

with $\boldsymbol{\kappa}(\mathbf{x}) = (k(\mathbf{x}, \mathbf{x}_1), \ldots, k(\mathbf{x}, \mathbf{x}_w))^T$, $\mathbf{G} = \mathbf{K} + \sigma_n^2 \mathbf{I}_w \in \mathbb{R}^{w \times w}$, $\mathbf{K} = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{w}$, $\mathbf{b} = \mathbf{G}^{-1}(\mathbf{Y} - m) \in \mathbb{R}^w$, $\mathbf{Y} = (c_1, \ldots, c_w)^T$. Like [13], we choose $m = 1$ for the prior mean to incorporate the prior knowledge that most of the space is empty. Doing this avoids the need of artificial off-surface points inside and outside the object by expert intervention to obtain a reasonable model, which other authors [4], [10] need to consider or describe to be problematic [5].

In order to estimate the surface based on the learned GP model from tactile data, the zero level set of the GP mean (2) has to be calculated, i.e. $\mathcal{S} \approx \left\{\mathbf{x} \in \mathbb{R}^3 | \mu_F(\mathbf{x}) = 0\right\}$. To realize this, we use the marching cube algorithm [15], which outputs a surface mesh.

Aside from obtaining the surface itself from the GP, with the derivative of the GP mean function

$$\mathbf{g}_F(\mathbf{x})^T = \frac{\partial}{\partial \mathbf{x}} \mu_F(\mathbf{x}) = \mathbf{b}^T \frac{\partial}{\partial \mathbf{x}} \boldsymbol{\kappa}(\mathbf{x}) \in \mathbb{R}^{1 \times 3} \tag{4}$$

the surface normal at location $\mathbf{x}$ on the surface

$$\mathbf{n}_F(\mathbf{x}) = \mathbf{g}_F(\mathbf{x}) / \left\|\mathbf{g}_F(\mathbf{x})\right\|_2 \in \mathbb{R}^3 \tag{5}$$

can on the one hand be estimated. On the other hand, outside the object, i.e. $\mu_F(\mathbf{x}) > 0$, the negative gradient $-\mathbf{g}_F(\mathbf{x})$ points towards the unknown object. This interpretation as a potential field has been introduced in [13] and will be used in this project for the control concept if the contact has been lost, see section V. If tactile sensors are used that provide additional information like normals, those can be integrated into the GP model as well [13], which would yield better estimates of the surface normals.

The gradient of the GP variance function

$$\mathbf{g}_{\mathbb{V}_F}(\mathbf{x})^T = \frac{\partial \mathbb{V}_F}{\partial \mathbf{x}}(\mathbf{x}) = \frac{\partial k}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{x}) - 2 \cdot \boldsymbol{\kappa}(\mathbf{x})^T \mathbf{G}^{-1} \frac{\partial \boldsymbol{\kappa}}{\partial \mathbf{x}}(\mathbf{x}) \tag{6}$$

will play a key role for the development of the greedy exploration strategy presented in section VI-A, because it points in the direction where the current GP surface estimation is most uncertain locally [3].

Regarding the kernel, we consider the strictly positive definite so-called inverse-multiquadric kernel

$$k(\mathbf{x}, \mathbf{x}') = \left(\left\|\mathbf{x} - \mathbf{x}'\right\|_2^2 + l^2\right)^{-\frac{1}{2}} \tag{7}$$

with a length scale parameter l. We observed that this kernel produces more favorable implicit surfaces and their variance compared to the often used squared exponential kernel.

Cholesky updates: A standard GP implementation suffers from the cubic complexity for calculating $\mathbf{G}^{-1}$, which is unfavorable for the purpose of this project, since the model has to be recomputed during the exploration in a 5 Hz loop. However, as the matrix $\mathbf{G}$ is a positive definite matrix, we compute and store the Cholesky decomposition of it. When a new data point is sampled, updating the Cholesky factor only requires quadratic complexity. Since then all involved linear systems are triangular, the complexity of the GP implicit surface model could significantly be reduced.

IV. Active Learning with Query Paths – Problem Formulation

We first describe the motivation for our proposed active learning framework and then formulate it more generally.

The main goal of tactile exploration as considered in this work is to reconstruct the shape of an unknown object with a GP implicit surface model. Therefore, the quantitative objective is to minimize the error between the reconstructed and true surface. Since the true surface is usually not available for an unknown object, this error cannot be used to derive an exploration strategy directly. Instead, the *uncertainty* of the current GP surface model can act as a surrogate measure. As proposed by Dragiev et al. [3], the surface uncertainty for a given tactile dataset $\mathscr{D}$ can be defined as the integral of the GP variance over the surface, normalized by its area (to make it comparable with different objects), which we call the **Uncertainty Measure**:

$$U(\mathscr{D}) = \frac{\int_{\mathcal{S}} \mathbb{V}_F \, \mathrm{d}\mathbf{s}}{\int_{\mathcal{S}} \mathrm{d}\mathbf{s}}. \tag{8}$$

The purpose of an active learning methodology is to generate actions of the robot to interact with the object/environment such that this uncertainty measure is minimized.

In more general Bayesian active learning, this problem is formulated in terms of a general so-called acquisition function $a(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$. This acquisition function encodes regions that are interesting to explore. For example, in so-called Bayesian optimization, the goal is to minimize an unknown and expensive to evaluate function by sampling it at certain locations. Here, the acquisition function tries to trade-off exploiting the current learned model to minimize the function and exploring regions where the function has not been sampled sufficiently to find a better global optimum. For the objective considered in the present work, namely tactile shape exploration, the variance of the GP implicit surface model defines a suitable acquisition function to minimize the uncertainty measure:

$$a(\mathbf{x}) = \mathbb{V}_F(\mathbf{x}). \tag{9}$$

In standard Bayesian active learning, one would maximize the acquisition function and then sample at this location. Exactly this is performed by [3], [4], [5]. However, as discussed previously, this point sampling has disadvantages for robotic applications.

Instead, we aim at maximizing the acquisition function along a *path* lying on the unknown object. These considerations lead to our proposed **Active Learning with Query**

Paths problem formulation:

$$\boldsymbol{\gamma}^* = \underset{\boldsymbol{\gamma}:[0,1]\to\mathbb{R}^3}{\operatorname{argmax}} \int_0^1 a(\boldsymbol{\gamma}(t)) \left\|\dot{\boldsymbol{\gamma}}(t)\right\|_2 \mathrm{d}t \tag{10a}$$

$$\text{s.t.} \quad F(\boldsymbol{\gamma}(t)) = 0 \quad \forall_{t\in[0,1]} \tag{10b}$$

$$\boldsymbol{\gamma}(0) = \mathbf{x}_0. \tag{10c}$$

A similar problem was formulated in [11]. However, they consider only a 2D problem without constraints and execute the optimized path completely, neglecting observations along the path. Especially treating the constraint (10b) is challenging in our problem setting, since it is unknown a priori and has to be explored as well. Therefore, solving this optimization problem (10) is not directly possible. As a side note, a global optimal path in this formulation would also have infinite length. To make this problem tractable, we need

1) A control framework that is able to slide along paths on the unknown object safely and robustly.
2) Further assumptions on the path to solve the active learning path querying problem.
3) An algorithmic realization.

This will be discussed in the next two sections.

V. Control Concept

Tactile exploration requires interaction between the robot and uncertain environments. Whereas point querying can be performed with simple controllers, sliding over a priori unknown surfaces along query paths requires more advanced concepts. Therefore, the controller plays an important role in this project. We belief that without coupling the controller and the exploration strategy closely, tactile exploration cannot work sufficiently.

To provide such a suitable concept, our controller consists of mainly two parts. First, a task description that is parameterized by the GP implicit surface model. Second, a concept to transform these tasks into motor commands of the robot.

A. Tasks and Parameterization

In general, we describe the desired behavior of a robot with n joints in terms of *task maps* $\phi : \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}^d$, $\mathbf{y} = \phi(\mathbf{q})$, which are differentiable functions from the robot configuration $\mathbf{q}$ to a d-dim. space. Common task spaces are the position or orientation of the end-effector, but also tasks like collisions, joint limits etc. are possible. These task spaces are equipped with desired references $\mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}$ and their corresponding importance matrices $\mathbf{K}_p, \mathbf{K}_d \in \mathbb{R}^{d\times d}$, which can be interpreted as stiffness/damping properties in this task space. With $\mathbf{x} = \phi_{\text{pos}}(\mathbf{q})$ we denote the 3D position task map of the end-effector with reference $\mathbf{x}_{\text{ref}}$. The relevant exploration tasks/parameterizations consist of the following three components, which are visualized in Fig. 2:

1) Maintaining contact: During the exploration, the robot should maintain the contact between its end-effector and the unknown object. This is realized by a *velocity* reference $\dot{\mathbf{y}}^{\text{ref}}_{\mathbf{n}_F}$ in the following 1D task space

$$\mathbf{y}_{\mathbf{n}_F} = \phi_{\mathbf{n}_F}(\mathbf{q}) = -\mathbf{n}_F(\phi_{\text{pos}}(\mathbf{q}))^T \phi_{\text{pos}}(\mathbf{q}) \in \mathbb{R}, \tag{11}$$

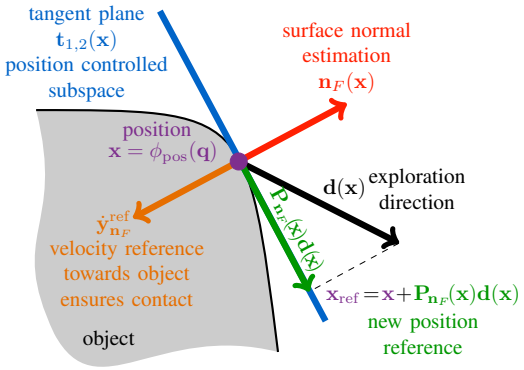


Fig. 2. Control concept: The velocity reference $\dot{\mathbf{y}}^{\text{ref}}_{\mathbf{n}_F}$ ensures that the robot stays in contact by orienting it towards the object based on the normal vector estimation $\mathbf{n}_F$. The desired search direction $\mathbf{d}$ is projected onto the estimated tangent plane, in which position control takes place.

which is the component of the end-effector position in the direction of the estimated surface normal at $\mathbf{x}$ based on the current GP model. With this velocity reference towards the object, the controller inherently includes also a recovery strategy if the contact has been lost unforeseen, since outside the surface, $-\mathbf{n}_F(\mathbf{x})$ will point towards the object and hence the velocity reference in this direction moves the robot to re-establish the contact with the object again.

2) Position task: The duty of the end-effector position task $\mathbf{x}$ is to actually move the robot over the surface. However, not arbitrary movements are possible. Instead, the robot can only change its position locally in the *tangent plane* of the object at its current position. To account for this, we formulate the stiffness/damping matrix of the position task map as a function of the current position of the robot:

$$\mathbf{K}_{p/d}(\mathbf{q}) = \mathbf{V}(\mathbf{q})\boldsymbol{\Lambda}\mathbf{V}(\mathbf{q})^T \tag{12}$$

with the matrix of eigenvectors

$$\mathbf{V}(\mathbf{q}) = \begin{pmatrix} \mathbf{n}_F(\phi_{\text{pos}}(\mathbf{q})) & \mathbf{t}_1(\phi_{\text{pos}}(\mathbf{q})) & \mathbf{t}_2(\phi_{\text{pos}}(\mathbf{q})) \end{pmatrix} \tag{13}$$

and their corresponding eigenvalues

$$\boldsymbol{\Lambda} = \operatorname{diag}\left(0,\ k_{p/d},\ k_{p/d}\right) \in \mathbb{R}^{3\times 3}. \tag{14}$$

Here, the eigenvectors $\mathbf{t}_1, \mathbf{t}_2$ are chosen such that they span the tangent plane, i.e. $(\mathbf{t}_1 \perp \mathbf{t}_2) \perp \mathbf{n}_F$, calculated based on the current surface normal $\mathbf{n}_F$, estimated by GP model. This parameterization, adapting in realtime to the unknown object, enables position control in the tangent space and does not interfere (zero eigenvalue in $\mathbf{n}_F$ direction) with the velocity reference which maintains the contact with the object.

3) Tangent space projection: The last step is to project desired movement references onto the tangent plane. Given an exploration direction $\mathbf{d}(\phi_{\text{pos}}(\mathbf{q})) \in \mathbb{R}^3$ (which could be a function of the current position), the *tangent space projector*

$$\mathbf{P}_{\mathbf{n}_F}(\mathbf{x}) = \mathbf{I}_3 - \mathbf{n}_F(\mathbf{x})\mathbf{n}_F(\mathbf{x})^T \tag{15}$$

projects this search direction onto the tangent plane at $\mathbf{x} = \phi_{\text{pos}}(\mathbf{q})$. Hence, the new position reference is given by

$$\mathbf{x}_{\text{ref}} = \mathbf{x} + \mathbf{P}_{\mathbf{n}_F}(\mathbf{x})\mathbf{d}(\mathbf{x}). \tag{16}$$

If $\mathbf{x}_{\text{ref}}$ is remote, the actual reference is interpolated and always projected to the tangent plane.

B. Closed-Loop Control Framework

Given M tasks maps ϕ_i, their references and parameters, the sake of the actual feedback control framework is to transform these into motor commands (torques) $\mathbf{u} \in \mathbb{R}^n$, such that the robot accomplish the desired task space behaviors. In the following, we describe a compliant task space controller which can achieve reasonable performance without the need of a precise dynamics model of the robot. The controller consists of two nested loops. In the outer, slow loop with 100 Hz, the necessary joint space references $\mathbf{q}, \dot{\mathbf{q}}$ that accomplish the task references are obtained by solving the inverse kinematics optimization problem

$$\min_{\mathbf{q},\dot{\mathbf{q}}} \sum_{i=1}^{M} \left\| \phi_i(\mathbf{q}) - \mathbf{y}_i^{\mathrm{ref}} \right\|_{\mathbf{K}_p^i}^2 + \left\| \dot{\phi}_i(\mathbf{q}) - \dot{\mathbf{y}}_i^{\mathrm{ref}} \right\|_{\mathbf{K}_d^i}^2 . \quad (17)$$

Linearizing the first terms at the current joint configuration $\mathbf{q}_0$, i.e. $\phi_i(\mathbf{q}) \approx \phi_i(\mathbf{q}_0) + \mathbf{J}_{\phi_i}(\mathbf{q} - \mathbf{q}_0)$ with Jacobian $\mathbf{J}_{\phi_i}$, and using $\dot{\mathbf{y}}_i = \mathbf{J}_{\phi_i}\dot{\mathbf{q}}$, the solution of (17) can be obtained

$$\mathbf{q}^* = \mathbf{A}_p^{-1} \sum_{i=1}^{M} \mathbf{J}_{\phi_i}^T \mathbf{K}_p^i \left(\mathbf{y}_i^{\mathrm{ref}} - \phi_i(\mathbf{q}_0) + \mathbf{J}_{\phi_i}\mathbf{q}_0 \right) \quad (18)$$

$$\dot{\mathbf{q}}^* = \mathbf{A}_d^{-1} \sum_{i=1}^{M} \mathbf{J}_{\phi_i}^T \mathbf{K}_d^i \dot{\mathbf{y}}_i^{\mathrm{ref}} \quad (19)$$

with $\mathbf{A}_{p,d} = \sum_{i=1}^{M} \mathbf{J}_{\phi_i}^T \mathbf{K}_{p,d}^i \mathbf{J}_{\phi_i}$. By a regularizing posture task $\phi(\mathbf{q}) = \mathbf{q}$, $\mathbf{y}^{\mathrm{ref}} = \mathbf{q}_0$, $\dot{\mathbf{y}}^{\mathrm{ref}} = 0$, singularity robustness is achieved. Since the optimization problem works on the kinematics level, the task space stiffnesses/dampings correspond to the importance of the task. The higher the eigenvalues of $\mathbf{K}_p^i$, the higher the priority is of achieving the task. In the context of interaction with the environment, the opposite, namely compliance in certain directions is often desired (low eigenvalues). These optimal joint space references (18), (19) are then translated to motor commands in the inner 1 kHz loop by a PD joint space control law

$$\mathbf{u} = \widehat{\mathbf{K}}_p \left(\mathbf{q}^* - \mathbf{q} \right) + \widehat{\mathbf{K}}_d \left(\dot{\mathbf{q}}^* - \dot{\mathbf{q}} \right) + \mathbf{u}_f \quad (20)$$

with joint space stiffness $\widehat{\mathbf{K}}_p \in \mathbb{R}^{n \times n}$ and damping $\widehat{\mathbf{K}}_d \in \mathbb{R}^{n \times n}$ matrix. The term $\mathbf{u}_f$ corresponds to an additional force controller, which is described in the next paragraph V-C.

Instead of using a joint space stiffness matrix that represents the task space stiffnesses and accounts for the dynamics of the robot, we observe that often anisotropic task space stiffnesses are used to realize *compliance* in certain task space directions. This insight allows us to use a hand-tuned diagonal joint space stiffness matrix $\mathbf{K}_p^{\mathrm{base}}$ and modify it

$$\widehat{\mathbf{K}}_p = \mathbf{P}_{\mathbf{p}}^{\gamma}(\mathbf{q}) \mathbf{K}_p^{\mathrm{base}} \mathbf{P}_{\mathbf{p}}^{\gamma}(\mathbf{q}) \quad (21)$$

with the projector

$$\mathbf{P}_{\mathbf{p}}^{\gamma} = \left(\mathbf{I}_n - \gamma \cdot \mathbf{p} \left(\mathbf{p}^T \mathbf{p} \right)^{-1} \mathbf{p}^T \right) \quad (22)$$

to generate compliance (weighted by $\gamma \in [0, 1]$) in the range of $\mathbf{p} \in \mathbb{R}^n$. We observed with our PR2 robot that this enables precises position control with compliance in certain directions. For the purpose of this work, the choice

$$\mathbf{p}(\mathbf{q}) = \mathbf{J}_{\phi_{\mathrm{pos}}}(\mathbf{q})^T \mathbf{n}_F(\phi_{\mathrm{pos}}(\mathbf{q})) \quad (23)$$

generates compliance in the direction of the unknown object, making the interaction safe and possible. In this way, the surface estimation directly affects the joint space stiffness matrix, which connects the closed-loop controller closely with the GP surface model.

C. Limit Force Controller

When the robot explores the unknown object with the proposed controller framework from the last paragraph, contact forces occur. Regulating these forces is important, first not to damage anything, but also to reduce friction, which is favorable for the sliding exploration. To realize this, we use the limit force controller of our previous work [16], which limits forces to a desired reference without the need of switching controllers for the free and constrained motion.

VI. Exploration Strategies

In the last section it is discussed how the robot can slide safely and robustly over the surface of the unknown object. Now the exploration strategies as the solution of the query path active learning problem (10) are discussed. In order to solve (10), we have to impose additional constraints on the path such that it is finite. Since the true on-surface constraint (10b) is unknown, we replace it by constraining the path to lie on the estimated surface based on the current GP model. This yields the slightly modified learning problem

$$\boldsymbol{\gamma}^* = \underset{\boldsymbol{\gamma}:[0,1]\to\mathbb{R}^3}{\operatorname{argmax}} \int_0^1 a(\boldsymbol{\gamma}(t)) \left\| \dot{\boldsymbol{\gamma}}(t) \right\|_2 \mathrm{d}t \quad (24a)$$

$$\text{s.t.} \quad \mu_F(\boldsymbol{\gamma}(t)) = 0 \quad \forall_{t\in[0,1]} \quad (24b)$$

$$\boldsymbol{\gamma}(0) = \mathbf{x}_0 \quad (24c)$$

$$+ \text{ additional assumptions on the path } \boldsymbol{\gamma}, \quad (24d)$$

for which in the following two realizations are presented.

A. Greedy Exploration – Local Optimal Paths as Projected Gradient Methods

The first idea to approach the query path problem (24) is to consider a local neighborhood only. To this end, assume that the query path is of the from

$$\boldsymbol{\gamma}(t) = \mathbf{x}_0 + \mathbf{d} \cdot t, \quad \|\mathbf{d}\|_2 = \alpha \quad (25)$$

with search direction $\mathbf{d} \in \mathbb{R}^3$ and length α, which typically should be small. This local path already fulfills (24c). By using first order Taylor approximations at $\mathbf{x}_0$ for both the acquisition function in the objective (24a) and the GP mean in the surface constraint (24b), solving (24) for (25) reads as

$$\mathbf{d}^* = \underset{\|\mathbf{d}\|_2=\alpha}{\operatorname{argmax}} \int_0^1 \left(a(\mathbf{x}_0) + \nabla a(\mathbf{x}_0)^T \mathbf{d} \cdot t \right) \alpha \, \mathrm{d}t \quad (26a)$$

$$\text{s.t.} \quad \underbrace{\mu_F(\mathbf{x}_0)}_{0} + \mathbf{g}_F(\mathbf{x}_0)^T \mathbf{d} \cdot t = 0 \quad \forall_{t\in[0,1]}. \quad (26b)$$

All parts that are independent of $\mathbf{d}$ do not alter the solution of (26). Therefore, it is equivalent to consider the following

$$\mathbf{d}^* = \underset{\|\mathbf{d}\|_2=\alpha}{\operatorname{argmax}} \nabla a(\mathbf{x}_0)^T \mathbf{d} \quad \text{s.t.} \quad \mathbf{g}_F(\mathbf{x}_0)^T \mathbf{d} = 0. \quad (27)$$

Solving this finite dimensional problem using Lagrange multipliers yields the optimal one step path direction

$$\mathbf{d}^* = \alpha \frac{\mathbf{P}_{\mathbf{n}_F}(\mathbf{x}_0)\nabla a(\mathbf{x}_0)}{\|\mathbf{P}_{\mathbf{n}_F}(\mathbf{x}_0)\nabla a(\mathbf{x}_0)\|_2} \tag{28}$$

with the same tangent space projector $\mathbf{P}_{\mathbf{n}_F}$ (15) as for the controller concept of section V. Not surprisingly, this means that the optimal local path is the projected gradient of the acquisition function onto the current tangent plane estimation with length α. The actual strategy now consists of repeating those local paths, which perfectly fits to the exploration controller. Therefore, this strategy, which we call *greedy exploration*, can also be interpreted in the notion of optimization theory: The robot is solving a constrained optimization problem on the unknown surface by moving along the projected gradient of the objective function, while the velocity reference ensures that after each gradient step the robot fulfills the constraint. For the variance as the acquisition function, the robot slides in the direction where the uncertainty of the current surface model is highest locally.

B. Global Exploration – Geodesics in Uncertainty Warped Space with Receding Horizon Optimal Control

An other idea to address (24) is to add a constraint to the end of the path. Then the query path consists of a geodesic on the surface from the actual position $\mathbf{x}_0$ to the maximum of the acquisition function $\mathbf{x}^* = \operatorname{argmax}_{\mu_F(\mathbf{x})=0} a(\mathbf{x})$. On the path, the robot should move through regions where the acquisition function is high. These considerations lead to the shortest path optimization problem

$$\boldsymbol{\gamma}^* = \underset{\boldsymbol{\gamma}:[0,1]\to\mathbb{R}^3}{\operatorname{argmin}} \int_0^1 \frac{1}{a(\boldsymbol{\gamma}(t))} \|\dot{\boldsymbol{\gamma}}(t)\|_2 \,\mathrm{d}t \tag{29a}$$

$$\text{s.t.} \quad \mu_F(\boldsymbol{\gamma}(t)) = 0 \quad \forall_{t\in[0,1]} \tag{29b}$$

$$\boldsymbol{\gamma}(0) = \mathbf{x}_0 \tag{29c}$$

$$\boldsymbol{\gamma}(1) = \underset{\mu_F(\mathbf{x})=0}{\operatorname{argmax}} a(\mathbf{x}), \tag{29d}$$

where the distances are warped by the acquisition function, e.g. the variance on the surface. We call this the *global exploration strategy*. Since the on-surface constraint (29b) is uncertain, the robot cannot travel the complete optimized path. This is one main difference to the 2D mobile robotics setting considered in [11], where there are no constraints. Furthermore, following the whole path would also not incorporate observations made in between. Therefore, the optimization problem (29) is solved in a receding horizon idea similar to model predictive control. With this we mean the following: First, solving (29) yields a complete path from the actual position to the one where the acquisition function is maximized on the surface. Then the robot travels along the path $\boldsymbol{\gamma}^*(t)$ to a certain extend $t^* < 1$, i.e. $t \in [0, t^*]$, only. Note that the actual path the robot travels until t^* will differ from the optimized one, since it is likely that the surface estimation is corrected by new observations and therefore the sliding controller framework projects the reference on the new estimated tangent plane during the slide. The procedure is repeated by solving (29) again from the current position, integrating the observations made on the traveled path.

To actually solve this receding horizon optimal path querying problem, we utilize the Dijkstra algorithm. This involves as a first step to obtain a surface mesh with the marching cube algorithm from the GP mean function μ_F. This surface mesh can be interpreted as an undirected weighted graph (V, E, W) with vertices V and edges E. The weights W between the vertices are calculated by the line integral $\int_{E_{ij}} a^{-1} \,\mathrm{d}E_{ij}$ along the edge E_{ij} connecting two vertices V_i, V_j. Then obtaining the constraint (29d) as $\mathbf{x}^* = \operatorname{argmax}_{\mathbf{x}\in V} a(\mathbf{x})$ reduces to check all vertices. Finally, the Dijkstra algorithm computes the shortest path from the vertex, which is closest to the current position $\mathbf{x}_0$ of the robot, towards $\mathbf{x}^*$.

The advantage of this method is that the resulting path is globally optimal on the discrete mesh. However, it also has two disadvantages. On the one hand, the discrete nature of the mesh sometimes leads to zigzag paths. Moreover, the computational bottleneck of this is to obtain the surface mesh with the marching cube algorithm, which slows down the exploration process for dataset sizes larger than 2000.

C. Normal Finding

The articulation of external degrees of freedom often requires pushing against the object at a location with a specific surface normal vector. Of course, the robot could first explore the surface sufficiently and then use the model to estimate the location with the desired normal. However, it is questionable whether this is effective. Within our methodology, we can provide a better solution. We propose the *normal finding* acquisition function

$$a(\mathbf{x}) = \frac{\langle \mathbf{n}_d, \mathbf{n}_F(\mathbf{x}) \rangle}{\|\mathbf{n}_d\|_2} \in [-1, 1], \tag{30}$$

which measures the parallelism between the desired surface normal $\mathbf{n}_d$ and the estimated one $\mathbf{n}_F(\mathbf{x})$ at location $\mathbf{x}$ based on the current GP surface model. It turns out (see section VII-B.2) that in order to find such a location with a desired surface normal, no exhaustive shape model needs to be obtained. The normal finding acquisition function combined with the greedy exploration strategy only works for convex objects in general, the global also for non-convex ones. We considered this strategy explicitly for applications where we want the robot to articulate links of a priori unknown shapes.

D. Restricting the exploration range

The exploration strategies presented in this work assume that the robot can move freely and therefore reach every position in Cartesian space. Usually, this is not the case due to kinematic limits of the robot. Even with a limit task in the controller framework, the exploration strategy does not recognize these limits and still tries to explore the unknown object from every pose. To circumvent this problem, we propose to add a potential $\mathcal{P} : \mathbb{R}^3 \to \mathbb{R}$, $\mathcal{P} \in C^1$ to the acquisition function

$$\tilde{a}(\mathbf{x}) = a(\mathbf{x}) + \mathcal{P}(\mathbf{x}). \tag{31}$$

If $\mathcal{P}(\mathbf{x}) < 0$ for locations where the robot should not move, those areas become less interesting for the exploration strategies. Inside the feasible range, $\mathcal{P}(\mathbf{x}) = 0$ should be

ensured. In the notion of optimization theory, restricting the exploration range refers to inequality constraints and $\mathcal{P}$ can therefore be interpreted as penalty function taking these inequality constraints into account. For example, if the robot should not move beyond a certain height z_{limit}, the potential

$$\mathcal{P}(\mathbf{x}) = \begin{cases} -c\left(\mathbf{x}_3 - z_{\text{limit}}\right)^2 & \mathbf{x}_3 < z_{\text{limit}} \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

can be used. One might argue that this incorporates some kind of prior knowledge we would like to avoid. However, restricting the exploration range refers to prior knowledge on the robot and not the object itself.

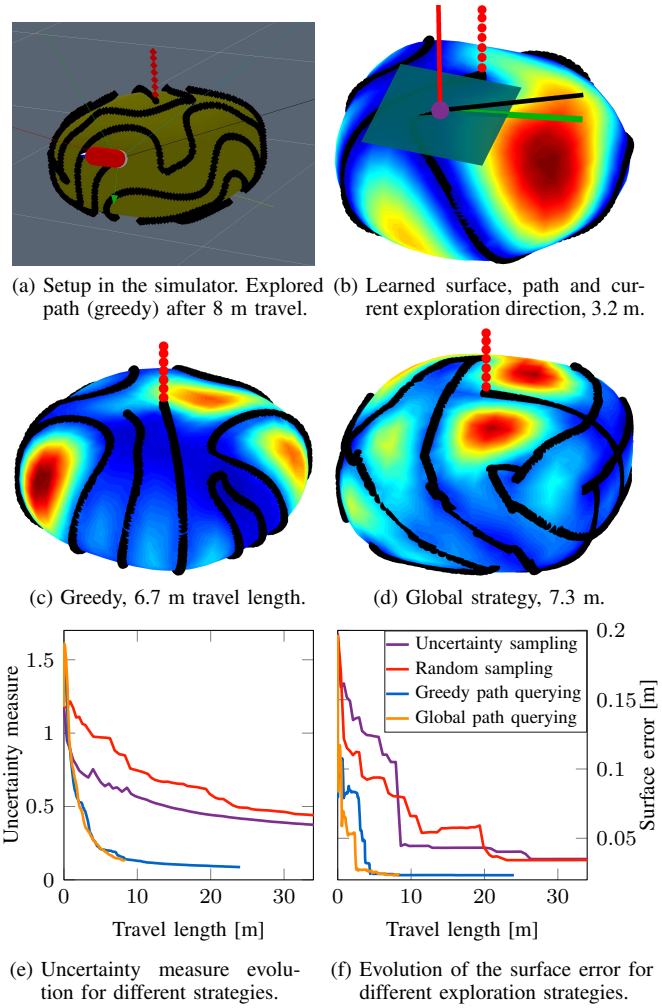
VII. Experiments

We show the effectiveness of our proposed approach both in simulation and on a real robot, the PR2. Refer to the video attachment to see the procedure in action. We compare in simulation random and existing active touch probing strategies [6], [4], [3] with our developed active path querying methodology. The experiments with the real robot include the greedy strategy for shape exploration and normal finding.

The exploration procedure is started as follows. The only prior knowledge the robot has is that it will find the object somewhere in one given direction. By a velocity reference in this direction (together with the limit force controller) the robot moves towards the object. The exact moment of contact is unknown a priori. The 5 Hz loop which collects the tactile data and updates the GP implicit surface model is run right from the beginning. As soon as there is a first surface estimate, which happens exactly at that moment when the robot establishes the contact with the object for first time, the exploration controller and active learning strategy loops are started. Then the exploration is performed until the uncertainty measure (8) is below a certain value, which indicates that the surface is explored sufficiently. The hyperparameters of the GP implicit surface model were $l = 0.2$ for the kernel length scale and $\sigma_n = 0.3$.

A. Simulation

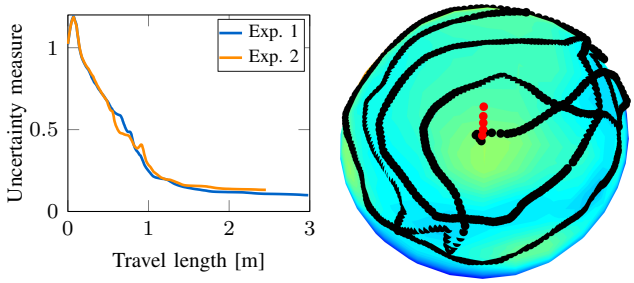
The reason to perform simulated experiments is to focus on our main contribution, the path querying active learning methodology. Here, the robot was a cylinder which can translate and rotate arbitrarily in the 6 degrees of freedom in euclidean space. Therefore, the robot can explore the unknown object from all sides. Nevertheless, the underlying framework (surface representation, control, exploration strategy) is the same as for the real world experiments. The object was a box with 50 cm width/depth and 25 cm height. Fig. 3a shows the experimental setup in the simulator. The red cylinder is the robot. Fig. 3a, 3b, 3c visualize the paths and the learned surfaces at different stages of the greedy exploration procedure. Fig. 3b additionally shows the surface normal (red), the variance gradient (black), which is projected on tangent plane (turquoise) to the green vector defining the exploration direction. In Fig. 3d, the path and learned surface with the global strategy is shown. The colors on the surface in these figures visualize the variance of the surface. Red



(a) Setup in the simulator. Explored path (greedy) after 8 m travel.
(b) Learned surface, path and current exploration direction, 3.2 m.
(c) Greedy, 6.7 m travel length.
(d) Global strategy, 7.3 m.
(e) Uncertainty measure evolution for different strategies.
(f) Evolution of the surface error for different exploration strategies.

Fig. 3. Surface exploration in simulation with greedy (a), (b), (c) and global (d) strategy. The performance between our proposed path querying active learning approach (greedy, global) and existing active/random point sampling methods is compared in (e), (f). Our methodology outperforms point sampling approaches both in uncertainty and surface error reduction.

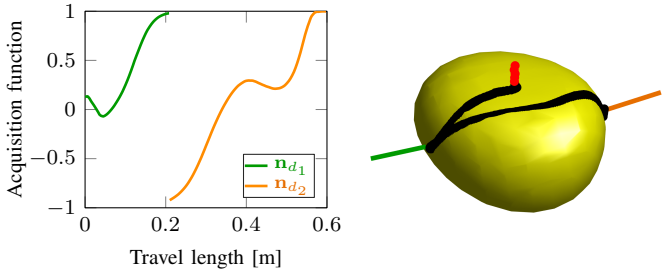
corresponds to high, blue to low variance. In Fig. 3e the performance in terms of the uncertainty measure reduction of the greedy and the global strategy is compared with random and active point sampling strategies. This comparison with the point querying methods was made very fair by assuming that the robot can travel directly from one sampling point to another, i.e. through the object itself, plus an additional travel length of 20 cm for each lift-off and touch-down per sample. In reality, the required travel distance would be higher. Since in simulation an exact model of the object is available, Fig. 3f shows the Hausdorff metric, the worst case error between the GP surface model and the true one. Despite the biasing in favor for the point sampling methods, our path querying active learning framework outperforms those existing methods significantly, both in terms of uncertainty measure and error reduction. With respect to the uncertainty reduction, the greedy and global strategy perform equally. Looking at Fig. 3f, the global strategy is slightly better than the greedy one in terms of fast surface error reduction. However, since the global strategy has computational time issues for large datasets, the greedy strategy is preferred.



(a) Uncertainty measure evolution of two same executions.

(b) Path and learned surface after 3 m travel length.

Fig. 4. Greedy surface exploration strategy with PR2 robot from two experiments. (b) Variance with potential (32), encoded in color, blue low.



(a) Evolution of the normal finding acquisition function.

(b) Normal exploration path on the learned surface.

Fig. 5. Surface normal finding with PR2 and greedy algorithm. The two desired normals were found efficiently without exhaustive exploration.

B. PR2 Robot

The experiments with a real PR2 robot deal with the exploration of a salad-bowl (dimensions 30 cm diameter, 9 cm height), as shown in Fig. 1. The end-effector of the PR2 was equipped with a red ball. Contact is inferred based on force measurements at the wrist of the PR2 exceeding a threshold of 1 N. Since the robot has a limited kinematic range, the exploration is restricted to stay above a certain height, which is realized by the potential (32) with $z_{\text{limit}} = 0.56$ and $c = 100$. Additional tasks to the ones of the exploration controller (see section V-A) include a joint limit and an orientation task that ensures that the robot points downwards. The greedy search direction step size in (28) was $\alpha = 0.04$ m.

1) Greedy exploration: The right of Fig. 1 shows the learned surface model during the exploration with the greedy strategy after 1.3 m travel length. Fig. 4a visualizes the evolution of the uncertainty measure (8) for two runs of the same experiment, which indicates fast uncertainty reduction. The final learned surface is shown in 4b, after 3 m travel length and 720 collected data points. The colors encode the variance of the surface, including the potential (32). The more blue, the less variance or near the restricted z-height.

2) Normal finding: The goal of this experiment is to find two normal vectors on the unknown salad bowl without building a complete surface model first. This was realized with the normal finding acquisition function (30) described in section VI-C and the greedy strategy as the underlying exploration method. The first desired normal was $\mathbf{n}_{d_1} = (0, -1, 0.1)^T$, whereas the second was in the opposite direction, i.e. $\mathbf{n}_{d_2} = (0, 1, 0.1)^T$. Finding the locations on the unknown object with those desired normal vectors corresponds to travel to $\mathbf{x}^*$ on the surface with $a(\mathbf{x}^*) = 1$. After convergence of the acquisition function for the first normal, the second was chosen. Fig. 5a shows the evolution of this acquisition function. The required travel length was approximately the shortest path (obtained with a tape measure). As it can be seen in Fig. 5b, no precise global surface model is required to solve the normal finding task quickly.

VIII. Conclusion

We presented a novel active learning framework based on path querying to efficiently address the problem of tactile shape exploration. An essential aspect was to integrate the surface representation and the controller with the exploration strategy into one system. Our methods showed to outperform previous approaches that are based on point querying.

The experimental setup assumes mainly convex objects. However, by using additional tactile sensor information such as contact point estimation, our methodology directly generalizes to non-convex objects as well. The main limitation of our approach similar to most related work is that the object has to be static. However, we belief that the methods presented in this work are indeed useful for dynamic environments. We especially proposed the normal finding method to effectively find locations to push an unknown object.

References

[1] S. J. Lederman and R. L. Klatzky, "Haptic perception: A tutorial," *Attention, Perception, & Psychophysics*, vol. 71, no. 7, 2009.

[2] R. Deimel, C. Eppner, J. lvarez Ruiz, M. Maertens, and O. Brock, "Exploitation of environmental constraints in human and robotic grasping," in *Int. Symp. on Robotics Research (ISRR)*, 2013.

[3] S. Dragiev, M. Toussaint, and M. Gienger, "Uncertainty aware grasping and tactile exploration," in *Proc. of the Int. Conf. on Robotics and Automation (ICRA)*, 2013.

[4] M. Björkman, Y. Bekiroglu, V. Högman, and D. Kragic, "Enhancing visual perception of shape through tactile glances," in *Proc. of the Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[5] Z. Yi, R. Calandra, F. Veiga, H. van Hoof, T. Hermans, Y. Zhang, and J. Peters, "Active tactile object exploration with gaussian processes," in *Proc. of the Int. Conf. on Intelligent Robots and Systems*, 2016.

[6] N. Jamali, C. Ciliberto, L. Rosasco, and L. Natale, "Active perception: Building objects' models using tactile exploration," in *Proc. of the Int. Conf. on Humanoid Robots (Humanoids)*, Nov 2016.

[7] Q. Li, C. Schürmann, R. Haschke, and H. Ritter, "A control framework for tactile servoing," in *Proc. of Robotics: Science and Systems*, 2013.

[8] J. Back, J. Bimbo, Y. Noh, L. Seneviratne, K. Althoefer, and H. Liu, "Control a contact sensing finger for surface haptic exploration," in *Proc of the Int. Conf. on Robotics and Automation (ICRA)*, May 2014.

[9] H. Liu, K. Nguyen, V. Perdereau, J. Bimbo, J. Back, M. Godden, L. Seneviratne, and K. Althoefer, "Finger contact sensing and the application in dexterous hand manipulation," *Auton. Robots*, 2015.

[10] G. A. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active planning for underwater inspection and the benefit of adaptivity," *The International Journal of Robotics Research*, 2012.

[11] R. Marchant and F. Ramos, "Bayesian optimisation for informative continuous path planning," in *Proc. of the Int. Conf. on Robotics and Automation (ICRA)*, 2014.

[12] O. Williams and A. Fitzgibbon, "Gaussian process implicit surfaces."

[13] S. Dragiev, M. Toussaint, and M. Gienger, "Gaussian process implicit surfaces for shape estimation and grasping," in *Proc. of the Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[14] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[15] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares, "Efficient implementation of marching cubes cases with topological guarantees," *Journal of Graphics Tools*, vol. 8, no. 2, pp. 1–15, 2003.

[16] D. Drieß, P. Englert, and M. Toussaint, "Constrained bayesian optimization of combined interaction force/task space controllers for manipulations," in *Proc. of the Int. Conf. on Robotics and Automation (ICRA)*, 2017.