GP-GPIS-OPT: Grasp Planning With Shape Uncertainty Using Gaussian Process Implicit Surfaces and Sequential Convex Programming

Jeffrey Mahler[1], Sachin Patil[1], Ben Kehoe[2], Jur van den Berg[3], Matei Ciocarlie[4], Pieter Abbeel[1], Ken Goldberg[1]

Abstract— **Computing grasps for an object is challenging when the object geometry is not known precisely. In this paper, we explore the use of Gaussian process implicit surfaces (GPISs) to represent shape uncertainty from RGBD point cloud observations of objects. We study the use of GPIS representations to select grasps on previously unknown objects, measuring grasp quality by the probability of force closure. Our main contribution is GP-GPIS-OPT, an algorithm for computing grasps for parallel-jaw grippers on 2D GPIS object representations. Specifically, our method optimizes an approximation to the probability of force closure subject to antipodal constraints on the parallel jaws using Sequential Convex Programming (SCP). We also introduce GPIS-Blur, a method for visualizing 2D GPIS models based on blending shape samples from a GPIS. We test the algorithm on a set of 8 planar objects with transparency, translucency, and specularity. Our experiments suggest that GP-GPIS-OPT computes grasps with higher probability of force closure than a planner that does not consider shape uncertainty on our test objects and may converge to a grasp plan up to** $5.7\times$ **faster than using Monte-Carlo integration, a common method for grasp planning under shape uncertainty. Furthermore, initial experiments on the Willow Garage PR2 robot suggest that grasps selected with GP-GPIS-OPT are up to 90% more successful than those planned assuming a deterministic shape. Our dataset, code, and videos of our experiments are available at `http://rll.berkeley.edu/icra2015grasping/`.**

I. Introduction

In the past five years, there has been substantial innovation in sensing modalities such as active RGB-D sensors that provide 3D point clouds such as the Kinect. However, these sensors currently do not perform well with transparent and specular objects that distort light projected by the sensor for depth computation. For example, a depth sensor cannot measure the tip of the nail in Fig. 1 due to its size and specularity, leading to uncertainty about its shape. These noisy and missing measurements cam complicate robotic grasping when the shape of objects in the environment is not known a priori. This motivates an object representation for shape uncertainty and a method for computing stable grasps with respect to this representation.

One promising object representation in robotics is the signed distance function (SDF), which is zero-valued at the object surface, positive-valued outside the object, and negative-valued in the object interior [10], [11], [29], [38]. In this paper, we consider representing objects with a Bayesian representation of SDFs known as the Gaussian process implicit surface (GPIS) [2], [10], [11], [19], [32], [41]. We also explore how GPIS representations can be used to plan grasps in the presence of shape uncertainty. The key idea behind GPIS-based representations is to represent shape uncertainty as a distribution over all possible SDFs that could fit sensor measurements of a shape. GPIS-based models can have higher variances in regions where sensor measurements are missing or unreliable and lower variances where sensor measurements are known to be accurate [41].

Our main contribution in this paper is GP-GPIS-OPT, an algorithm to plan grasp contacts for parallel-jaw grippers with high probability of force closure. Our algorithm can handle both (a) shape uncertainty represented as a GPIS and (b) uncertainty in the grasp approach direction. A secondary

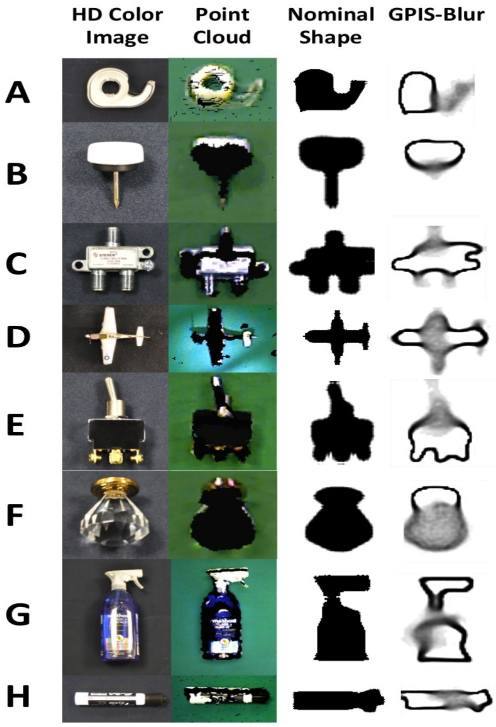

Fig. 1: Eight example objects on a horizontal worksurface illustrating transparency, specularity, and sensor noise: (from top to bottom) (A) plastic tape dispenser, (B) nail, (C) signal splitter, (D) metal toy plane, (E) mechanical switch, (F) crystal knob, (G) squirt bottle, and (H) marker. Displayed from left to right are an HD color image on black background, a point cloud observation from a Primesense Carmine mounted on the head of a PR2, the nominal shape on a 25 × 25 grid, and our GPIS-Blur visualization. The visualization is blurry in areas of larger uncertainty and sharper in areas of low uncertainty. Black areas in the point cloud indicate missing observations due to transparency or specularity, and in transparent areas the sensor sees through the objects and measures the depth of the table.

[1] Department of EECS; {`jmahler, sachinpatil, pabbeel, goldberg`}`@berkeley.edu`
[2] Department of ME; `benk@berkeley.edu`
[1−2] University of California, Berkeley, CA, USA
[3] Google Inc., CA, USA `jurvandenberg@gmail.com`
[4] Department of ME, Colombia University, NY, USA `matei.ciocarlie@columbia.edu`

contribution is GPIS-Blur, a method to visualize 2D GPIS models using shape sampling. We evaluate GP-GPIS-OPT in simulation and on the Willow Garage PR2 on 8 test objects that are challenging to perceive with visual sensors. We compare performance with a grasp planner that assumes a deterministic shape and with ranking grasps using Monte-Carlo integration [7], [22], [23].

II. Related Work

Polygonal mesh models are widely used for representing shapes in robotics, but extensions of mesh models to include uncertainty is difficult. Solutions include assuming independent uncertainty in the vertex locations, but this ignores spatial noise correlations [23]. GPISs are an alternative that represent objects as distribution over SDFs [32], [41]. In robotics, GPIS has been used in online control and active sensing. Dragiev et al. [10] considered the use of GPIS for grasping, using the maximum likelihood SDF to control a grasp to reach a desired location. The authors extended this work for tactile shape exploration during grasping but did not utilize a measure of grasp quality to guide grasping [11]. Hollinger et al. [19] created high resolution GPIS models of ship hulls to guide exploration by an underwater robot to locations with the highest shape uncertainty. Other works have used GPIS to fuse uncertain data from multiple sensors and to guide haptic exploration of object shapes [2], [21]. In this work we consider using GPIS to plan grasp contact points offline.

Choosing grasps given a surface representation of an object typically focuses on finding grasps given an exact object shape by maximizing a grasp quality metric [6], [9]. A common quality measure is the Ferrari-Canny metric, which measures the ability to resist force perturbations [12] and is widely used in grasp software packages such as GraspIt! [27]. Past work on grasping with uncertainty has focused on state uncertainty [14], [15], uncertainty in object pose [8], [24], [40] or uncertainty in the location of contact with the object [33].

Several works have studied the effects of shape uncertainty on grasp quality. Christopoulos et al. [7] sampled spline fits for 2-dimensional planar objects and ranked a set of randomly generated grasps using Monte-Carlo integration, which can be computationally expensive. Kehoe et al. [22], [23] showed that adaptive sampling could be used to select grasps robust to part tolerance for parallel-jaw grippers on extruded polygon shape models. Brook, Ciocarlie, and Hsiao [4], [20] studied a Bayesian framework to evaluate the probability of grasp success given uncertainty in object identity, gripper positioning, shape, and pose by simulating grasps on deterministic mesh and point cloud models. Laaksonen et al. [1], [25] used Gaussian Processes (GPs) to model distributions on grasp stability online from tactile measurements, and selected grasps using Monte Carlo Markov Chain sampling. Panahi et al. [30] presented an algorithm to determine a sequence of pushes to orient 2D polygonal parts with shape uncertainty by computing upper and lower bounds on the orientation of objects after push actions.

Past work on grasp optimization primarily focuses on optimizing contact points on a surface with respect to a grasp quality measure when the surface is known exactly [26], [33]. Chen et al. [5] optimized antipodal grasps to satisfy force closure for parametric surfaces without uncertainty. Ciocarlie et al. [9], [27] use simulated annealing to find grasps with a high Ferrari-Canny metric for the GraspIt! software, using a penalty on signed distance to the object surface to force contact with the object. Simulated annealing is based on random exploration of the input state space, which can avoid local minima but can be less computationally efficient than gradient-based methods [9]. Our approach can be seen as a local optimization similar to the method of Ciocarlie et al. [9] with constraints similar to those used by Chen et al. [5], however our method uses an explicit model of shape uncertainty.

III. Gaussian Process Implicit Surfaces

In this section we review Gaussian process implicit surfaces (GPISs). A signed distance function (SDF) [29] describes the shape of an object by storing the signed distance from every point in space to the nearest point on the surface. SDFs are defined as a real-valued function $f : \mathbb{R}^d \to \mathbb{R}$ such that $f(\mathbf{x}) > 0$ outside the object, $f(\mathbf{x}) = 0$ on the object surface, and $f(\mathbf{x}) < 0$ inside the object. A GPIS is a Gaussian distribution over SDFs formed by Gaussian process regression (GPR) on noisy observations of an SDF [41]. In this work we will use $d = 2$ and restrict evaluations of the SDF f to an $M \times M$ 2-dimensional grid with square cells [2], [10]. In practice M might be set based on the resolution of the sensor used to acquire measurements [29].

A. Gaussian Process Regression (GPR)

Gaussian process regression (GPR) is used in machine learning as a nonparametric regression method for estimating continuous functions from sparse and noisy data [32]. For a GPIS, a training set consists of a set of input spatial locations $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, $\mathbf{x}_i \in \mathbb{R}^d$, and signed distance observations $\mathbf{y} = \{y_1, \dots, y_n\}, y_i \in \mathbb{R}$. In practice, $\mathbf{y}$ can be acquired using KinectFusion, which uses ray tracing to compute an SDF from RGBD point clouds [29], or by segmenting an object from the environment and performing a Euclidean distance transformation [13], [39].

A GPIS is specified by a mean function $m(\cdot)$ and a covariance function $k(\cdot, \cdot)$, also referred to as a *kernel*, which measures the similarity in signed distance between spatial locations. Given a set of training data $\mathcal{D} = \{\mathcal{X}, \mathbf{y}\}$, mean $m(\cdot)$, kernel $k(\cdot, \cdot)$, and measurement noise σ_m, the posterior distribution on SDF f_* at a test location $\mathbf{x}_*$ is [32]:

$$p(f_* \mid \mathbf{x}_*, \mathcal{D}) \sim \mathcal{N}\big(\mu(\mathbf{x}_*), \sigma^2(\mathbf{x}_*)\big)$$

$$\mu(\mathbf{x}_*) = m(\mathbf{x}_*) + k(\mathcal{X}, \mathbf{x}_*)^\intercal (K + \sigma_m^2 I)^{-1} (\mathbf{y} - m(\mathcal{X})) \quad (1)$$

$$\sigma^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - k(\mathcal{X}, \mathbf{x}_*)^\intercal (K + \sigma_m^2 I)^{-1} k(\mathcal{X}, \mathbf{x}_*) \quad (2)$$

where $K \in \mathbb{R}^{n \times n}$ is a matrix with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. $k(\mathcal{X}, \mathbf{x}_*) = [k(\mathbf{x}_1, \mathbf{x}_*), \ldots, k(\mathbf{x}_n, \mathbf{x}_*)]^\intercal$, and $m(\mathcal{X}) = [m(\mathbf{x}_1), ..., m(\mathbf{x}_n)]^\intercal$. This derivation can also be used to predict the mean and variance of the SDF gradient by differentiating the kernel function, which can be used to obtain GPIS surface normals [10], [32], [35].

Following Dragiev et al. [10], we use the squared exponential kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = C \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\ell^2}\right)$$

which specifies the correlation of the signed distance between two spatial points. This kernel depends on a scale $C \in \mathbb{R}$ and bandwidth $\ell \in \mathbb{R}$, which we set using maximum-likelihood estimation [41]. Other common kernels relevant to GPIS are the thin-plate splines kernel [41] and the Matern kernel [2].

B. GPIS Construction from Point Clouds

In this work we consider constructing 2D GPIS models from RGBD point clouds from a single viewpoint of a Primesense Carmine sensor [18]. As seen in Fig. 1, the scenario we consider consists of objects lying flat on a table imaged from above. Our method assumes that the object can be identified in the point clouds and segmented from the background. Furthermore, we assume that missing measurements in the point cloud are caused by surface properties of the object instead of the environment, because in our scenario the table is measured accurately by the Primesense. Thus, we use a constant negative mean function for the GPIS to bias areas of missing measurements to be part of the object.

To construct a GPIS from point clouds, we first combine several point clouds by averaging to remove the effects of small zero-mean noise in the depth values, similar to KinectFusion [29]. Then we create a segmentation mask for the object in both the RGB and depth point clouds. For the objects in Fig. 1 this segmentation is performed by hand, and in our physical experiments we use RGB and depth thresholding. We also create a measurement noise map, which specifies the variance of 0-mean measurement noise, based on a noise model of the Primesense [18]. We then combine the two segmentation masks in an image with each pixel weighted by its inverse variance to form an occupancy grid, and compute an SDF using a Euclidean distance transformation of the occupancy map [18], [39]. Finally, we run GPR on the SDF values and measurement noise map to construct a GPIS. Points with missing measurements (e.g. NaN values in the depth map) are considered part of the object in the occupancy map but are not used for the GPIS construction.

This construction procedure results in high uncertainty in (a) areas where the RGB and depth segmentations disagree (e.g. Object A) and (b) areas with missing measurements (e.g. Object B). GPIS could also be constructed directly from the SDF and confidence weights used in KinectFusion [29].

IV. GPIS-Blur Visualization

Past applications of GPIS in robotics have used visualizations of the mean shape [2], [10], [11], [19], but visualizing shape uncertainty is difficult. This is because GPIS specifies uncertainty in signed distance, which leads to a complicated distribution over the spatial locations of the zero-crossing [32]. Our "GPIS-Blur" method produces visualizations of GPIS uncertainty by sampling SDF zero-crossings, as illustrated in Fig. 2. Let $f_i \sim \mathcal{N}(\mu, \Sigma)$ be a sample drawn from the GPIS on the grid $\mathcal{X}$. The zero-crossing z_i is computed by thresholding f_i:

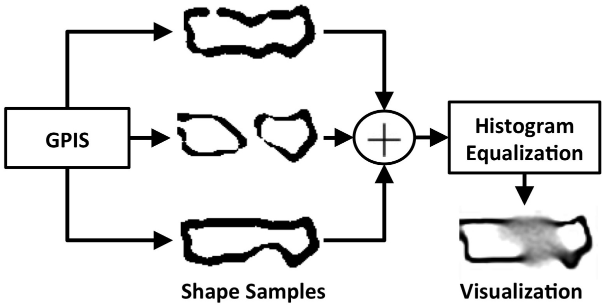

Fig. 2: GPIS-Blur visualization method for a 2D GPIS of Object H from Fig. 1 with shape uncertainty near the object center. (Left to right) We sample 1000 SDFs, threshold for zero crossings, average sample the samples together, and finally enhance contrast for easier perception of the uncertain regions.

$$m_i(\mathbf{x}_j) = \sum_{\mathbf{x}_k \in N(\mathbf{x}_j)} \mathbb{I}(f_i(\mathbf{x}_k) \leq 0)$$

$$z_i(\mathbf{x}_j) = \begin{cases} 1 & m_i(\mathbf{x}_j) > 0 \\ 0 & m_i(\mathbf{x}_j) = 0 \end{cases} .$$

where $N(\mathbf{x}_j)$ is a neighborhood of points around $\mathbf{x}_j$ and $\mathbb{I}$ is the indicator function. This forms a binary "image" of the model. We repeat this procedure for N samples, average the z_i "images" into a single greyscale image, and use histogram equalization to enhance the contrast of the result [37]. The result is that regions of the shape that are more certain appear dark and crisp, while regions of high uncertainty appear grey and blurred. This method can be extended to 3D by using a transparency level based on uncertainty to give uncertain areas a "cloudy" look.

V. Grasping on a GPIS Representation

We consider a rigid object grasped from above on a 2D worksurface using a parallel-jaw gripper. We will use a 2-dimensional GPIS representation of the object contours to select grasps. We assume a known, deterministic coefficient of friction γ between the grasped object and the grippers, as well as hard contacts for the parallel jaws of the grippers. We also assume that the gripper-part interaction is quasistatic [22], [23] and a known, constant mass density.

A. Object Model

We create a 2D GPIS representation that is augmented to predict the gradient of the SDF [35], as described in Section III. We use f to denote a SDF sample of the GPIS over the spatial grid, $f(\cdot) \sim \mathcal{N}(\mu(\cdot), \sigma^2(\cdot))$. We refer to the outward pointing surface normals of f at a spatial location $\mathbf{x}$ as $\mathbf{n}(\mathbf{x}) = \frac{\nabla\mu(\mathbf{x})}{\|\nabla\mu(\mathbf{x})\|_2}$, which is the normalized gradient of f [10]. The SDF f is evaluated over a fixed grid, $\mathcal{X}$, as

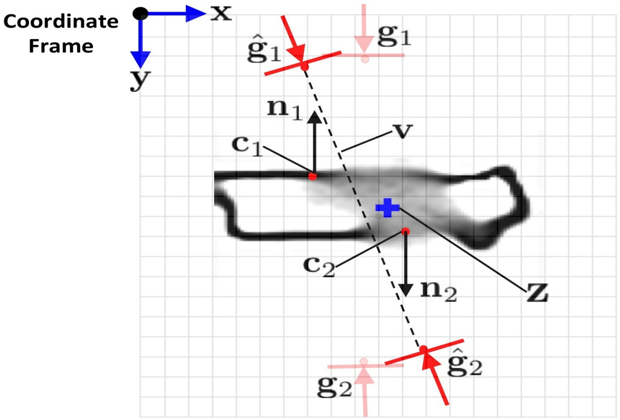

Fig. 3: Illustration of the variables defined for GP-GPIS-OPT on a Object H of Fig. 1 with shape uncertainty near the object center. The GPIS with center of mass $\mathbf{z}$ is specified on a spatial grid with frame of reference at the top left cornet. Jaw placements are illustrated by a direction arrow and jaw. The target grasp $(\mathbf{g}_1, \mathbf{g}_2)$ results in a perturbed grasp $(\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2)$ and approach direction $\mathbf{v}$ due to errors in jaw placement. When the jaws close, the perturbed grasp contacts the surface at locations $(\mathbf{c}_1, \mathbf{c}_2)$ due to the width of the jaws, with outward-pointing surface normals $(\mathbf{n}_1, \mathbf{n}_2)$.

described in Section III. The average center of mass $\mathbf{z} \in \mathbb{R}^2$ is derived from the mean GPIS and object mass distribution.

B. Candidate Grasp Model

Our candidate grasp model is illustrated in Fig. 3. We consider a parallel-jaw gripper with jaws of width $w_j \in \mathbb{R}$ wide and a maximum opening of $w_g \in \mathbb{R}$. We define target grasp points as spatial locations $\mathbf{g}_1, \mathbf{g}_2 \in \mathbb{R}^2$ for each of the parallel jaws with a coordinate frame at the top left of the GPIS grid. Therefore our set of *candidate grasps* is $\mathcal{G} = \{g = (\mathbf{g}_1, \mathbf{g}_2) : \|\mathbf{g}_1 - \mathbf{g}_2\| \leq w_g\}$, and we henceforth refer to a single planned grasp as g.

In practice a robot may not be able to execute a desired grasp g exactly due to errors in trajectory following or registration to the object [20]. To handle this uncertainty, we define a *perturbed grasp* $\hat{g} = (\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2)$ as the location of the grasp points during execution. We model the error in positioning the gripper as zero-mean Gaussian noise with covariance $\sigma_g^2 I$ about the desired location for the first jaw $\hat{\mathbf{g}}_1 \sim \mathcal{N}(\mathbf{g}_1, \sigma_g^2 I)$. The location of the second jaw is conditionally dependent on the first $\hat{\mathbf{g}}_2 \mid \hat{\mathbf{g}}_1 \sim \mathcal{N}(\hat{\mathbf{g}}_1 + (\mathbf{g}_2 - \mathbf{g}_1), \sigma_g^2 I)$. The variance σ_g^2 is set based on repeatability measurements for a robot gripper [28].

Given a perturbed grasp, we also define a *contact point* as the point at which the grasp comes into contact with an object when following the line segment between the two parallel jaws. We will refer to the contact configuration for a perturbed grasp $\hat{g}$ as $c = (\mathbf{c}_1, \mathbf{c}_2)$ where $\mathbf{c}_1, \mathbf{c}_2 \in \mathbb{R}^2$. Formally, a contact point $\mathbf{c}$ is the first zero crossing of an SDF f that the parallel jaws pass over when following approach direction $\mathbf{v} = \hat{\mathbf{g}}_1 - \hat{\mathbf{g}}_2$, where a zero crossing satisfies $|f(\mathbf{c})| < \epsilon$ for some small user-defined $\epsilon \in \mathbb{R}, \epsilon > 0$. In practice the contacts can be found by testing a discrete set of points along $\mathbf{v}$ for the zero crossing condition [29]. We will refer to the surface normals at contact configuration c as $n = (\mathbf{n}_1, \mathbf{n}_2)$ where $\mathbf{n}_1, \mathbf{n}_2 \in \mathbb{R}^2$.

C. Quality Measures

Force closure is a binary quantity that measures whether or not a grasp can resist external wrenches (forces and torques) applied to a grasped object in arbitrary directions when an object shape is known precisely. The the L1 version of the Ferrari-Canny grasp metric Q_F [12], [27] measures the strength of force closure by the relative magnitude of wrenches that the gripper would have to exert to resist external wrenches. Without loss of generality, given a set of contact wrenches $\mathcal{W} \in \mathbb{R}^3$ derived from 2D contact locations c, normals n, and center of mass $\mathbf{z}$, $Q_F(c, n, \mathbf{z})$ measures the size of the largest ball around the origin in wrench space within the convex hull of $\mathcal{W}$ [12]. We will use henceforth use the notation $Q_F(\hat{g}, f)$ as shorthand because c, n, and $\mathbf{z}$ can be derived deterministically from a grasp $\hat{g}$ and an SDF sample f.

In this work we measure the quality of a grasp using the probability of force closure P_F [22], [23], [25], [40] given uncertainty in shape and grasp approach. Since force closure is equivalent to the positivity of Q_F, we evaluate P_F by checking the sign of Q_F over samples from the uncertain quantities, $P_F = P\left(Q_F > 0 \mid g, \mu, \sigma^2\right)$ [40].

D. Grasp Planning Definition

Our goal is to find a grasp g that maximizes P_F given a GPIS mean and variance function μ, σ^2:

$$g^* = \max_{g \in \mathcal{G}} P\left(Q_F(\hat{g}, f) > 0 \mid g, \mu, \sigma^2\right) \ . \tag{3}$$

We can evaluate this probability by integrating over the set of force closure grasps with respect to the distributions on perturbed grasps $\hat{g}$ and SDFs f:

$$P_F(g, \mu, \sigma^2) \quad = \int\limits_{\hat{\mathcal{G}}, \mathcal{F}} \mathbb{I}(Q_F(\hat{g}, f) > 0) p(\hat{g} \mid g) p(f \mid \mu, \sigma^2).$$

where $\hat{\mathcal{G}}$ and $\mathcal{F}$ denote the sets of perturbed grasps and SDFs, respectively.

VI. Grasp Optimization Algorithm

The algorithm GP-GPIS-OPT (Grasp Planner using GPIS OPTimization) finds locally optimal solutions of an approximation to equation 3 using gradient-based optimization methods with finite differences to compute derivatives [31].

A. Quality Approximation

Direct optimization of Equation (3) requires an integration over shape samples to evaluate the objective for each grasp, which may be computationally expensive, especially for high resolution models. However, we show that the probability of force closure on the the mean SDF μ, $\tilde{P}_F$, may be a reasonable approximation of P_F under the following conditions:

1) The center of mass $\mathbf{z}$ is within δ of its expectation $E[\mathbf{z}]$ with high probability, $P\left(\|\mathbf{z} - E[\mathbf{z}]\|_2 > \delta\right) \approx 0$ for some $\delta > 0$.
2) The shape uncertainty along grasp approach $\mathbf{v}$ up to the contact locations $\tilde{\mathbf{c}}_i$ on the mean shape is less than some small $\tau > 0$, $\sigma^2(\tilde{\mathbf{c}}_1) < \tau$.

We refer the reader to the supplementary file for a formal description of the approximation.

Empirically we found that the variance of the center of mass is less than the grid resolution for 7 out of 8 of the objects in Fig. 1, approximately satisfying condition 1. To encourage condition 2 to be satisfied we penalize the uncertainty σ^2 by $\lambda \in \mathbb{R}, \lambda > 0$.

$$\underset{g\in\mathcal{G}}{\text{maximize}}\ \tilde{P}_F(g,\mu) - \lambda(\sigma^2(\tilde{\mathbf{c}}_1) + \sigma^2(\tilde{\mathbf{c}}_2)) \tag{4}$$

High values of λ, which correspond to smaller τ, may increase the accuracy of the approximation for selected grasps but may discard grasps with high P_F. Small values of λ may increase the set of possible grasps but the approximation will become increasingly inaccurate. In practice λ can be set using cross-validation over a set of validation shapes, similar to choosing a regularization penalty in regression models [3]. The λ penalty encourages GP-GPIS-OPT to avoid regions of uncertainty, unlike past algorithms that use GPIS to encourage exploration [11].

B. Grasp Constraints

Each jaw approaches the object from opposite directions, so we constrain the grasp contacts on the mean shape to be antipodal [5]. For unit outward pointing normal vectors n at contact c and approach direction $\mathbf{v} = \hat{\mathbf{g}}_1 - \hat{\mathbf{g}}_2$, c is an antipodal pair if the contacts have (a) opposite normals and (b) normals aligned with the approach direction [5], [10], [11]. We convert these constraints to inequalities because on actual objects there may be very few grasps that satisfy the definitions exactly:

$$\|\mathbf{n}_1 + \mathbf{n}_2\|_2^2 \leq \alpha \quad \mathbf{n}_1^T\mathbf{v} \geq \beta\|\mathbf{v}\|_2^2 \quad -\mathbf{n}_2^T\mathbf{v} \geq \beta\|\mathbf{v}\|_2^2 \tag{5}$$

For 2-dimensional shapes the choice of $\beta = \cos(\arctan(\gamma))$ guarantees that points in the feasible region will be in force closure for the mean SDF [5], [7], and α may be set using cross-validation.

C. Grasp Selection Algorithm

Taking the objective of equation 4 and the constraints of equation 5 yields the optimization objective:

$$\begin{aligned}
\underset{g\in\mathcal{G}}{\text{maximize}} \quad & \tilde{P}_F(g,f) - \lambda(\sigma^2(\mathbf{c}_1) + \sigma^2(\mathbf{c}_2)) \\
\text{subject to:} \quad & \|\mathbf{n}_1 + \mathbf{n}_2\|_2^2 < \alpha \\
& \mathbf{n}_1^T\mathbf{v} > \beta\|\mathbf{v}\|_2^2 \\
& -\mathbf{n}_2^T\mathbf{v} > \beta\|\mathbf{v}\|_2^2
\end{aligned} \tag{6}$$

where $\mathbf{c}_1, \mathbf{c}_2, \mathbf{n}_1, \mathbf{n}_2$, and $\mathbf{v}$ are derived from candidate grasp g on the mean shape μ as described in Section V-B.

This problem is non-convex due to the Q_F quality evaluation and antipodal constraints on the derived contracts and normals. Therefore we can only expect a local solution to this problem for a single random initialization. The GP-GPIS-OPT (Grasp Planner using GPIS OPTimization) algorithm, detailed in Algorithm 1, repeatedly finds locally optimal grasps for equation 6 from a user-specified number of random initializations N_g and selects the grasp with the highest objective value. We use $N_g = 20$, based on the empirical worst-case time to converge to a solution for a validation set of 25×25 GPIS models. The supplementary file shows and discusses plots of the convergence versus iterations.

GP-GPIS-OPT uses Sequential Convex Programming (SCP) to find a locally optimal grasp for Equation (6). SCP iteratively forms a convex approximation to the problem and solves the approximation within a confidence [34], [42]. SCP turns non-convex constraints into penalties and iteratively increasing a penalty coefficient, similar to Interior Point methods [34]. Although P_F is not differentiable everywhere, we use finite differences to approximate the gradients, as has been shown to work well empirically for optimizing Q_F on deterministic shapes in 3D [31].

```
Input: GPIS Model $\mu(\cdot), \sigma^2(\cdot)$, Grid $\mathcal{X}$, and Number
of Initial Grasps $N_g$
Result: Grasp Proposal $g^*$ and Objective Value $V^*$
Initialize mean shape $\mu(\mathcal{X})$;
Initialize grasp count $k = 0$ ;
Initialize $V^* = -\infty$ ;
while $k < N_g$ do
    Sample valid initial grasp $g_{0,k}$ ;
    Use SCP to find a locally optimal grasp $g_k$ for
    Equation  6 with objective value $V_k$ ;
    if $V_k > V^*$ then
        $g^* = g_k$ ;
        $V^* = V_k$ ;
    end
    k = k + 1 ;
end
```

Algorithm 1: The GP-GRASP-OPT Algorithm

VII. Experiments

A. Dataset

We evaluated the performance of several grasp planners on a set of 8 objects. Our test set of objects is displayed in Fig. 1 and the data is available at the URL in the abstract. The objects chosen illustrate properties which lead to missing or invalid measurements with an RGBD sensor. Specularity and transparency can lead to missing measurements, which appear as black regions in point clouds. For example, the specular wings of Object D and the metal parts of Object C and E cannot be sensed by the RGBD camera. Invalid measurements, such as measuring the table behind a transparent region, occur for Objects A and F. The GPIS for each object was constructed on a 25×25 grid, chosen for computational efficiency.

To test our the assumptions of our force closure approximation on this dataset we computed the 95% confidence regions for the center of mass over the GPIS distribution for each object. We found that for 7 of the 8 objects this confidence region was smaller than 1 grid cell, matching our assumptions. However, for Object G the variance was approximately 7 grid cells due to high uncertainty around the body of the squirt bottle.

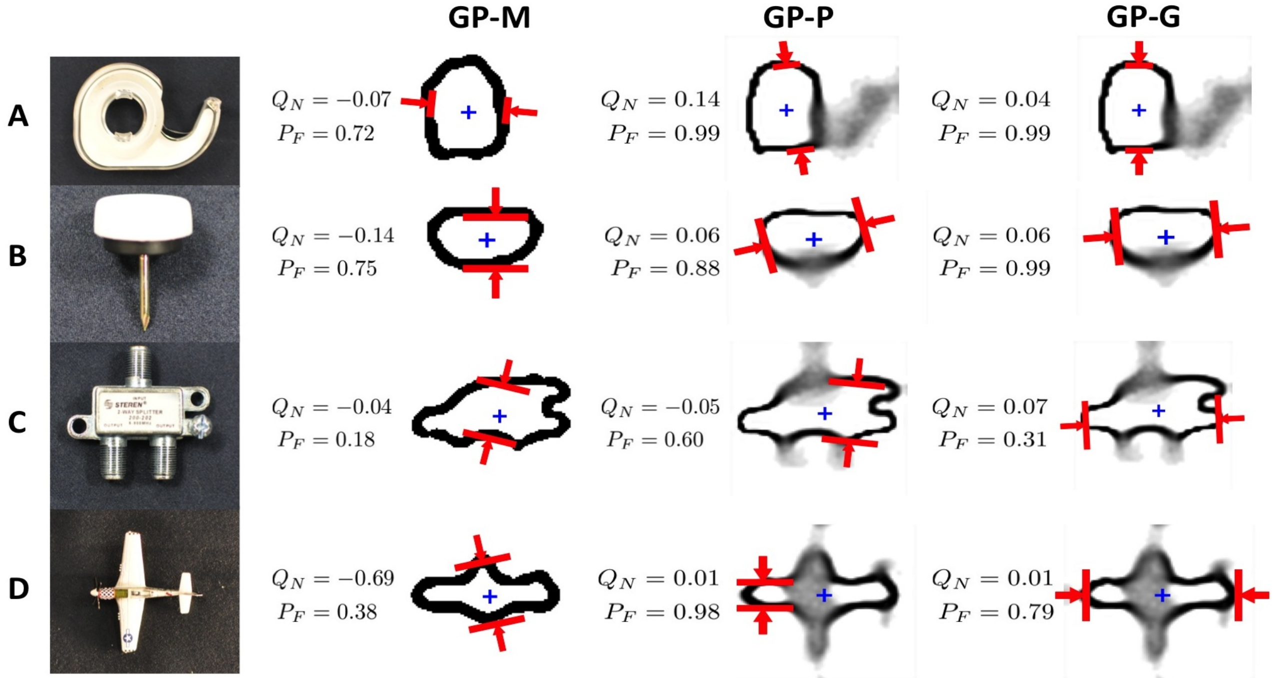

Fig. 4: Visual comparison of grasps on Objects A, B, C, and D (top to bottom) selected by (left to right) (i) GP-M, which chooses the grasp with highest Q_F on the shown mean SDF, (ii) GP-P, which chooses the grasp with the highest probability of force closure P_F from 1000 random grasp samples and (iii) GP-G, our algorithm for selecting grasps using optimization. The Ferrari-Canny quality on the nominal shape Q_N and P_F are listed to the left of each grasp. We see that GP-G outperforms GP-M in terms of Q_N; for example, on Object A GP-M chooses a grasp on the transparent tape dispenser since the mean shape does not contain this region but GP-G avoids this region due to shape uncertainty. Also, grasps chosen by GP-G are comparable to those chosen by GP-P in both P_F and Q_N.

B. Grasp Planning

To evaluate the performance of GP-GPIS-OPT, we compared the probability of force closure P_F and Ferrari-Canny grasp quality on the nominal shape Q_N for 5 competing grasp selection methods:

- **GP-M:** Selects g with the highest Ferrari-Canny quality Q_F on the mean SDF from 1000 random grasp samples.
- **GP-P:** Selects g with the highest P_F from 1000 random grasp samples, evaluating each grasp with Monte-Carlo integration over 1000 shapes [7], [23].
- **GP-D:** Optimizes P_F on the mean SDF μ with finite differences
- **GP-U:** Optimizes P_F on the mean SDF μ with uncertainty penalty λ, as in Equation (4).
- **GP-G:** The GP-GPIS-OPT algorithm, as detailed in Algorithm 1.

GP-M is fast because it ignores shape uncertainty. GP-P is equivalent to evaluating the objective of GP-G using a Monte-Carlo integration to approximate P_F instead of the λ penalty. As a result, it selects grasps with higher P_F but may be slow because it can require many SDF samples to converge [23]. GP-GPIS-OPT is designed to be faster than GP-P and is also to select grasps with high P_F. GP-D and GP-U are of comparable speed and are presented to evaluate the use of the uncertainty penalty and antipodal constraints in GP-GPIS-OPT.

We used a heuristic grasp sampling method for GP-M and GP-P to encourage sampling of antipodal grasps. We first sampled a point uniformly at random from the mean shape surface. We then randomly chose a grasp direction as either the surface normal or direction to the center of mass. Finally, the grasp was set to two points on this line centered around the shape interior that were exactly w_g apart.

	Probability of Force Closure P_F					Runtime (sec)	
Object	GP-M	GP-D	GP-U	GP-P	**GP-G**	GP-P	**GP-G**
A	0.72	0.43	0.13	**0.99**	**0.99**	645.5	**110.9**
B	0.75	0.00	0.02	0.88	**0.99**	955.8	**284.3**
C	0.18	0.22	0.02	**0.60**	0.31	921.9	**253.8**
D	0.38	0.12	0.57	**0.98**	0.79	715.8	**72.4**
E	0.40	0.24	0.32	**0.78**	0.57	1197.3	**253.8**
F	0.50	0.39	0.23	0.64	**0.67**	897.6	**215.3**
G	0.75	0.00	0.01	**0.91**	0.37	848.7	**84.3**
H	0.42	0.47	0.00	**0.99**	0.93	715.0	**85.5**

TABLE I: Comparison of the probability of force closure P_F for (a) GP-M, which selects a grasp based on only the mean SDF, (b) GP-D, which optimizes P_F on the mean SDF, (c) GP-U, which optimizes the P_F with an uncertainty penalty, (d) GP-P, which chooses the grasp with highest P_F from a set of 1000 random grasp samples, and (e) GP-G, GP-GPIS-OPT. GP-G performs as well as GP-P on 3 of 8 objects and selects grasps with higher P_F than methods other than GP-P for all objects. We also compare the worst-case runtime to converge to a grasp plan for GP-G and GP-P, and see GP-G is up to $5.7\times$ faster than GP-P averaged over all shapes.

The parameters of GP-G, GP-D, and GP-U were $N_g = 20$ and a grasp approach uncertainty of $\sigma_g^2 = 0.25$ [28]. The uncertainty penalty of GP-G and GP-U was set to $\lambda = 2.0$ based on a grid search using the set of validation shapes. The sampling-based methods GP-M and GP-P used 1000 random grasp samples, chosen based on the empirical number of random samples required for the methods to converge to a single grasp on the validation set. All experiments were run in Matlab 2014a on a laptop with OS X with a 2.7 GHz Intel

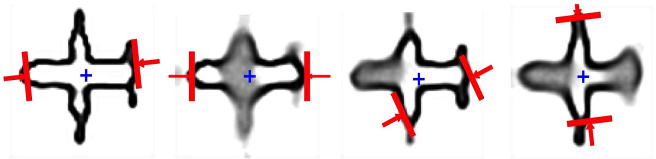

Fig. 5: Comparison of grasps chosen by our algorithm on Object D, the toy plane, with different patterns of shape uncertainty: (left to right) uncertainty in the tips and edges of wings, uncertainty in the entire wings, uncertainty in the cockpit end of the plane, and uncertainty in both the cockpit and rudder end of the plane.

core i7 processor and 16 GB 1600 MHz memory. We used CVX for optimization [17] and our implementations were not optimized for speed.

Table I compares P_F for each of the grasp selection methods and the runtimes for GP-P and GP-G. While GP-M is not designed to optimize P_F, we include its value for comparison. The runtimes reported for GP-P and GP-G are the worst-case runtimes to converge to a grasp plan. For GP-P this is the time to evaluate 753 grasp samples, the worst case number observed over all experiments, and for GP-G this is the time to optimize all $N_g = 20$ grasps.

We see that GP-G chooses the grasp with the same P_F as GP-P on 4 of 8 objects and the grasp with the highest P_F for methods other than GP-P on 7 of 8 objects. Our algorithm selects grasps with considerably lower P_F than GP-P on Object G. This may be explained by the high variance in the center of mass for Object G, which violates the assumptions made in the P_F approximation for GP-G. GP-G also converges faster than GP-P in the worst case for all objects, with up to a $5.7\times$ speedup over GP-P averaged over the objects. See the supplementary file for further details on the convergence and time complexity of GP-GPIS-OPT.

We illustrate grasps chosen by GP-M, GP-P, and GP-GPIS-OPT for Objects A-D and compare each by P_F and Q_N in Fig. 4. We see that GP-M chooses a grasp that is not in force closure on the nominal (true) shape for several objects, illustrating the need to take uncertainty into account. For example, on Object A the grasp chosen by GP-M is not in force closure because it ignores the tape dispenser, which cannot be sensed with the depth sensor. However, both grasps that take into account shape uncertainty are able to avoid the dispenser and choose a grasp with high P_F and Q_N. We also see that grasps chosen by our algorithm have higher Q_N than GP-M on all shapes.

C. Sensitivity to Shape Uncertainty

Fig. 5 compares grasps chosen by GP-GRASP-OPT with different patterns of shape uncertainty on Object D, the metal toy plane. We consider shape uncertainty when Object D has noise only on the edges of the wings, when the entire wings are not observed, when the tip of the plane is not observed, and when both the front and the back are not observed. Uncertainty in the wings leads to a grasp on the endpoints of the plane, while uncertainty on the front end of the plane the algorithm chooses a grasp that leverages a wing and a rear stabilizer. The algorithm selects the tip of the wings when the ends of the plane are uncertain, resulting in an unstable grasp.

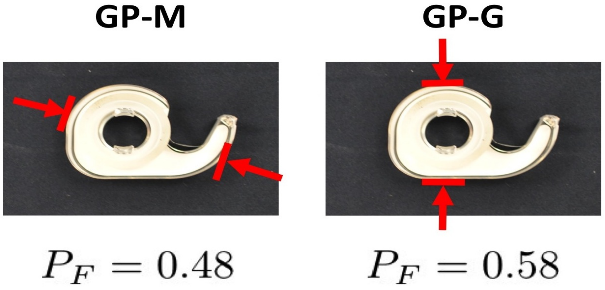

Fig. 6: Grasps selected in physical experiments on the PR2 for Object A, the tape dispenser, overlayed on photos. The grasp planned by $GP - M$ fails to lift the object on 9 of 10 attempts because it collides with the transparent dispenser, but GP-GPIS-OPT avoids the area due to uncertainty in the object geometry near the dispenser.

The planner does not choose the concave regions where the wings meet the body due to the uncertainty penalty, which encourages the optimization to avoid the region.

D. Physical Grasp Experiments with the PR2

We tested grasps for Object A, a roll of Scotch tape, on the Willow Garage PR2. We created a GPIS from a point cloud segmentation of the object from 10 images of a head-mounted Primesense Carmine as described in Section III-B. We compared GP-M and GP-GPIS-OPT with $N_g = 10$ and $\lambda = 2.0$, chosen by grid search. The grasps chosen by these methods are illustrated in Fig. 6. The grasp chosen by GP-GPIS-OPT had $P_F = 0.58$ and achieved force closure on 10 of 10 trials. In comparison, the grasp chosen by GP-M had predicted $P_F = 0.48$ and achieved force closure on only 1 of 10 trials. The failures were due to the gripper contacting and sliding along the transparent outer surface of the tape canister, which could not be sensed reliably in either the depth or color images.

VIII. Discussion and Future Work

We presented GP-GPIS-OPT, an algorithm for selecting parallel-jaw grasps with high probability of force closure P_F on Gaussian process implicit surface representations of shape uncertainty. Our experiments suggest that GP-GPIS-OPT plans grasps with higher P_F than methods that do not model shape uncertainty, and it is faster than common methods for grasp planning with shape uncertainty.

A current shortcoming of GP-GPIS-OPT is the use of an approximation to P_F that introduces a penalty term λ into our objective. In future work, we will study the effect of λ on the quality of selected grasps and consider alternative quality measures. We also plan to study using alternative methods for optimizing over uncertain quantities, such as Multi-Armed Bandits [36].

We will also extend our method to 3D objects. Since GPIS construction scales cubically with the number of grid points used, we will investigate how to construct GPIS models efficiently using subset selections [16] or local GP models [19]. We are also interested in implementing GP-GPIS-OPT on a multicore or Cloud Computing framework to further increase speeds, as it is parallelizable over random initial grasps. Finally, we are interested in extending our method to multi-point grasps on 3D GPIS shape representations by optimizing over joint angles at increasing GPIS resolution and using forward kinematics to determine contact locations [9].

IX. Acknowledgments

This work is supported in part by the U.S. National Science Foundation under Award IIS-1227536, and by grants from Google and Cisco. We thank our colleagues who gave feedback, suggestions, and help on the PR2, in particular James Kuffner, Florian Pokorny, Michael Laskey, Greg Kahn, Sanjay Krishnan, Sylvia Herbert, Zoe McCarthy, Alex Turney, John Schulman, and Dylan Hadfield-Menell.

References

[1] Y. Bekiroglu, J. Laaksonen, J. A. Jorgensen, V. Kyrki, and D. Kragic, "Assessing grasp stability based on learning and haptic data," *IEEE Trans. Robotics*, vol. 27, no. 3, pp. 616–629, 2011.

[2] M. Bjorkman, Y. Bekiroglu, V. Hogman, and D. Kragic, "Enhancing visual perception of shape through tactile glances," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2013, pp. 3180–3186.

[3] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2009.

[4] P. Brook, M. Ciocarlie, and K. Hsiao, "Collaborative grasp planning with multiple object representations," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2011, pp. 2851–2858.

[5] I.-M. Chen and J. W. Burdick, "Finding antipodal point grasps on irregularly shaped objects," *Robotics and Automation, IEEE Transactions on*, vol. 9, no. 4, pp. 507–512, 1993.

[6] J.-S. Cheong, H. Kruger, and A. F. van der Stappen, "Output-sensitive computation of force-closure grasps of a semi-algebraic object," *IEEE Trans. Automation Science & Engineering*, vol. 8, no. 3, pp. 495–505, 2011.

[7] V. N. Christopoulos and P. Schrater, "Handling shape and contact location uncertainty in grasping two-dimensional planar objects," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2007, pp. 1557–1563.

[8] V. N. Christopoulos and P. R. Schrater, "Grasping objects with environmentally induced position uncertainty," *PLoS computational biology*, vol. 5, no. 10, p. e1000538, 2009.

[9] M. Ciocarlie, C. Goldfeder, and P. K. Allen, "Dimensionality reduction for hand-independent dexterous robotic grasping," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2007, pp. 3270–3275.

[10] S. Dragiev, M. Toussaint, and M. Gienger, "Gaussian process implicit surfaces for shape estimation and grasping," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2011, pp. 2845–2850.

[11] ——, "Uncertainty aware grasping and tactile exploration," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2013, pp. 113–119.

[12] C. Ferrari and J. Canny, "Planning optimal grasps," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 1992, pp. 2290–2295.

[13] Y. Furukawa and J. Ponce, "Carved visual hulls for image-based modeling," in *European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 564–577.

[14] K. Y. Goldberg and M. T. Mason, "Bayesian grasping," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 1990, pp. 1264–1269.

[15] K. Y. Goldberg, "Stochastic plans for robotic manipulation," 1990.

[16] A. Gotovos, N. Casati, G. Hitz, and A. Krause, "Active learning for level set estimation," in *Int. Joint Conference on Artificial Intelligence (IJCAI)*, 2013.

[17] M. Grant, S. Boyd, and Y. Ye, "Cvx: Matlab software for disciplined convex programming," 2008.

[18] C. Herrera, J. Kannala, J. Heikkilä, *et al.*, "Joint depth and color camera calibration with distortion correction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2058–2064, 2012.

[19] G. A. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active planning for underwater inspection and the benefit of adaptivity," *Int. J. Robotics Research (IJRR)*, vol. 32, no. 1, pp. 3–18, 2013.

[20] K. Hsiao, M. Ciocarlie, and P. Brook, "Bayesian grasp planning," in *ICRA 2011 Workshop on Mobile Manipulation: Integrating Perception and Manipulation*, 2011.

[21] J. Ilonen, J. Bohg, and V. Kyrki, "Fusing visual and tactile sensing for 3-d object reconstruction while grasping," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2013, pp. 3547–3554.

[22] B. Kehoe, D. Berenson, and K. Goldberg, "Estimating part tolerance bounds based on adaptive cloud-based grasp planning with slip," in *Proc. IEEE Conf. on Automation Science and Engineering (CASE)*. IEEE, 2012, pp. 1106–1113.

[23] ——, "Toward cloud-based grasping with uncertainty in shape: Estimating lower bounds on achieving force closure with zero-slip push grasps," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 576–583.

[24] J. Kim, K. Iwamoto, J. J. Kuffner, Y. Ota, and N. S. Pollard, "Physically-based grasp quality evaluation under uncertainty," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 3258–3263.

[25] J. Laaksonen, E. Nikandrova, and V. Kyrki, "Probabilistic sensor-based grasping," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 2019–2026.

[26] G. Liu, J. Xu, X. Wang, and Z. Li, "On quality functions for grasp synthesis, fixture planning, and coordinated manipulation," *IEEE Trans. Automation Science & Engineering*, vol. 1, no. 2, pp. 146–162, 2004.

[27] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[28] B. Mooring and T. Pack, "Determination and specification of robot repeatability," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, vol. 3. IEEE, 1986, pp. 1017–1023.

[29] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE Int. Symposium on Mixed and augmented reality (ISMAR)*. IEEE, 2011, pp. 127–136.

[30] F. Panahi, M. Davoodi, and A. F. van der Stappen, "Orienting parts with shape variation," 2014.

[31] F. T. Pokorny, K. Hang, and D. Kragic, "Grasp moduli spaces." in *Robotics: Science and Systems*, 2013.

[32] C. E. Rasmussen, "Gaussian processes for machine learning," 2006.

[33] M. A. Roa and R. Suárez, "Finding locally optimum force-closure grasps," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 3, pp. 536–544, 2009.

[34] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization." in *Robotics: Science and Systems*, vol. 9, no. 1. Citeseer, 2013, pp. 1–10.

[35] E. Solak, R. Murray-Smith, W. E. Leithead, D. J. Leith, and C. E. Rasmussen, "Derivative observations in gaussian process models of dynamic systems," 2003.

[36] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2010.

[37] J. A. Stark, "Adaptive image contrast enhancement using generalizations of histogram equalization," *IEEE Trans. on Image Processing*, vol. 9, no. 5, pp. 889–896, 2000.

[38] F. Steinke, B. Schölkopf, and V. Blanz, "Support vector machines for 3d shape processing," in *Computer Graphics Forum*, vol. 24, no. 3. Wiley Online Library, 2005, pp. 285–294.

[39] J. Wang and Y. Tan, "Efficient euclidean distance transform algorithm of binary images in arbitrary dimensions," *Pattern Recognition*, vol. 46, no. 1, pp. 230–242, 2013.

[40] J. Weisz and P. K. Allen, "Pose error robust grasping from contact wrench space metrics," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*. IEEE, 2012, pp. 557–562.

[41] O. Williams and A. Fitzgibbon, "Gaussian process implicit surfaces," *Gaussian Proc. in Practice*, 2007.

[42] S. Wright and J. Nocedal, *Numerical optimization*. Springer New York, 1999, vol. 2.