Accepted Manuscript

Dexterous grasping under shape uncertainty

Miao Li, Kaiyu Hang, Danica Kragic, Aude Billard

PII: S0921-8890(15)00196-7
DOI: http://dx.doi.org/10.1016/j.robot.2015.09.008
Reference: ROBOT 2536

To appear in: *Robotics and Autonomous Systems*

Received date: 25 April 2015
Revised date: 25 August 2015
Accepted date: 4 September 2015

Please cite this article as: M. Li, K. Hang, D. Kragic, A. Billard, Dexterous grasping under shape uncertainty, *Robotics and Autonomous Systems* (2015), http://dx.doi.org/10.1016/j.robot.2015.09.008

1. We considered object shape uncertainty in grasp planning and control.
2. We proposed a probabilistic model to solve hand inverse kinematics.
3. Our grasp planning approach is hand interchangeable.
4. We presented a compliant uncertainty-aware controller for finger closing during grasp execution.

Dexterous Grasping under Shape Uncertainty

Miao Li[a], Kaiyu Hang[b], Danica Kragic[b], Aude Billard[a]

[a]*Learning Algorithms and Systems Laboratory(LASA), École Polytechnique Fédérale de Lausanne (EPFL), Switzerland*
[b]*Computer Vision & Active Perception Lab, Centre for Autonomous Systems, Royal Institute of Technology, Stockholm, Sweden*

Abstract

An important challenge in robotics is to achieve robust performance in object grasping and manipulation, dealing with noise and uncertainty. This paper presents an approach for addressing the performance of dexterous grasping under shape uncertainty. In our approach, the uncertainty in object shape is parameterized and incorporated as a constraint into grasp planning. The proposed approach is used to plan feasible hand con guration for realizing planned contacts using different robotic hands. A compliant nger closing scheme is devised by exploiting both the object shape uncertainty and tactile sensing at ngertips. Experimental evaluation demonstrates that our method improves the performance of dexterous grasping under shape uncertainty.

Keywords: Dexterous grasping, Shape uncertainty, Grasp control, Grasp learning

1. Introduction

Dexterous grasping is an essential skill for many tasks that robots are expected to perform, ranging from the assembly of workpieces in a factory setup to advanced manipulation of cutlery in a household environment. The core requirement of a successful dexterous grasping system is to place the ngertips on the relevant locations of an object, applying sufficient contact forces and maintaining grasp stability. To achieve this, a common approach is to address two subproblems: *grasp planning* and *grasp execution*. Considerable progress has been made during the last couple of years and ef cient grasp planning algorithms have been proposed to generate grasps for known, partially known or unknown objects in structured or unstructured environments [1, 2, 3, 4, 5]. Robust and reactive grasp control techniques have been developed and validated on different robotic platforms relying on single or multiple sensory feedback. Despite these achievements, demonstrating robust and exible object grasping and manipulation in natural environments taking into account uncertainties in perception and control remains a challenge.

In this paper, we address the problem of uncertain shape perception in a system considering ngertip grasping. Uncertain shape perception may originate from occlusion, partial view or issue with sensor calibration. We present a system which takes into account shape uncertainty during grasp planning and execution. Shape uncertainty is parametrized using Gaussian Processes (GP) and it is incorporated as a constraint into a contact-level grasp synthesis algorithm. The output of the algorithm is a set of contacts de ning a grasp with an associated shape uncertainty that determines the maximum uncertainty a grasp can withstand, as shown in the left upper part of Fig. 1(1). Given the desired grasping contacts, the feasible hand con guration (hand pose and nger joint con guration), is computed using a probabilistic model. The probabilistic model is learned offline for each hand and is frame invariant thanks to the use of a *Virtual Frame* (VF) approach. The learned model is hence, independent of the choice of hand and object frame. VF relies on a set of parameters de ned to encode grasps as shown in the right upper part of Fig. 1(2). Since a grasp is rst planned in the object frame by generating a set of contact locations, it is not dependent on a speci c hand design. Similarly, the learned probabilistic model for the hand inverse kinematics is not constrained by object shape. Therefore, given a new hand with its learned probabilistic model, the corresponding hand con guration that matches the generated grasping contacts can be obtained in real time. For grasp execution, a compliant nger closing scheme is devised. A parallel

Email address: miao.li@epfl.ch (Miao Li)

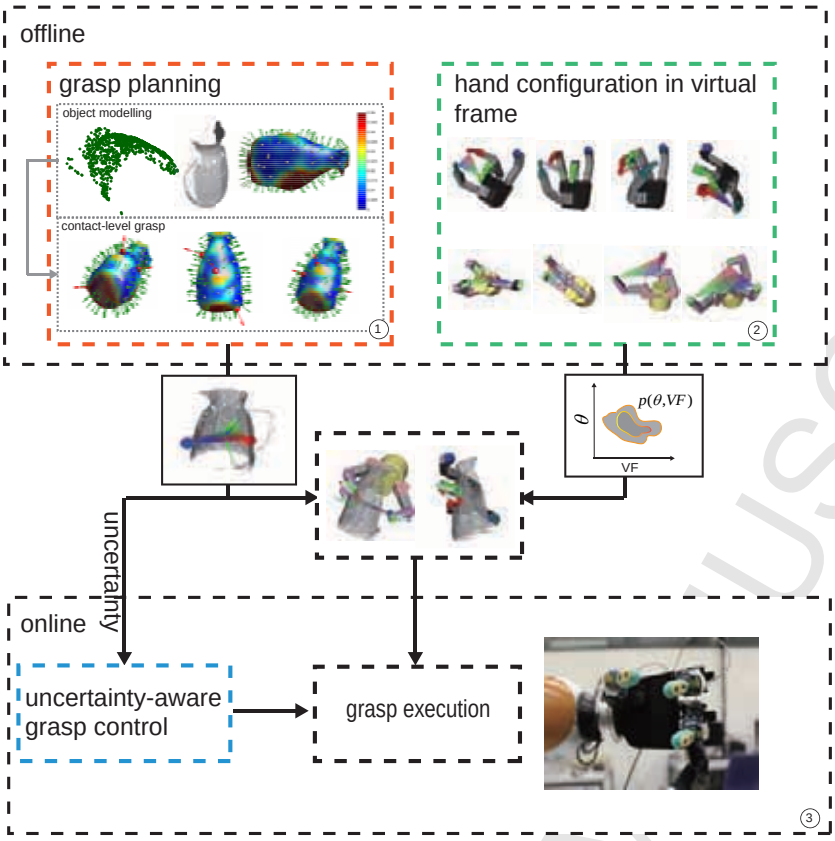


Figure 1: The overview of the proposed approach.

①: The contact-level grasp planning with shape uncertainty. The output is a set of contacts defining a grasp with associated shape uncertainty these can withstand.

②: The probabilistic model for the hand inverse kinematics is learned offline and is frame invariant.

③: Given the desired grasping points and the employed hand, the corresponding hand configuration is obtained in real time.

④: The obtained hand configuration and the uncertainty information are passed to the controller for compliant grasp execution.

position/force (tactile) controller is implemented by exploiting the uncertainty in shape and contact force based on our previous work [6]. An overview of the system is shown in Fig. 1.

The paper is organized as follows: Section 2 provides a review of the related work. Section 3 gives an introduction to object surface modeling using GP, along with its application in grasp planning. Section 4 presents a learning-based approach for the hand inverse kinematics. A compliant finger closing scheme is depicted in Section 5. Implementation details and experimental results are described in Section 6, followed by a discussion and conclusion in Section 7.

2. Related work

We provide an overview of related work considering dexterous grasp planning, control systems for grasping and grasping under uncertainty.

Early work on grasp planning focused on finding the optimal contact points considering force closure as a grasp quality measure [7, 8, 9, 10]. More recently, hand kinematics has been taken into account when estimating the feasible hand configuration for realizing the grasping points [11, 12]. A drawback of this approach is that the valid hand configuration to realize the contacts may not be found. An alternative approach is to optimize the contact locations and the hand configurations simultaneously. Due to the high dimensionality of the problem, the optimization is conducted in a projected space of lower dimensionality using hand synergies [13] or eigen grasps [14]. There are also works that formulate the optimization in the original hand configuration space [15, 16]. However, this is computationally expensive and the obtained grasps are hand-dependent. In this paper, we decouple contact synthesis and hand configuration estimation and rely on an offline learning process to obtain the relevant hand configuration.

Learning-based approaches have been proposed before and most of these use data-driven model to learn "rules" between object shape and feasible hand configurations [1]. In [27, 28, 29], objects are represented as basic shape primitives and then associated with predefined grasp primitives. In [30], a support vector machine (SVM) is used to learn the grasp quality manifold for a specific hand and simple object shapes. The manifold

Table 1: A brief summary of grasping under uncertainty. **Uncertainty type**: The type of uncertainty has been studied; **Rep.**: The uncertainty is represented implicitly (not parametrized) or explicitly (parametrized); **Hand**: The type of robotic hand has been considered; **Planning**: The uncertainty has been considered during the planning; **Control**: The uncertainty has been considered during the control stage and the corresponding strategy used.

Paper	**Uncertainty type**	**Rep.**	**Hand**	**Planning**	**Control**
[17, 18]	Object pose	Implicit	Barrett hand	No	Hand adjustment
[19]	Object pose and shape	Implicit	PR2 gripper	Grasp primitives	Heuristic
[20, 21, 22]	Object shape	Explicit	Dexterous hand	Grasp primitives	iterative exploration
[23]	Object pose	Explicit	Dexterous hand	Grasp Primitives	No
[24]	Contact location	Explicit	No	Yes	No
[12]	Contact location	Explicit	Dexterous Hand	Yes	No
[25]	Contact location and shape	Explicit	No	Yes	No
[26, 6]	Object mass and friction	Implicit	Dexterous hand	No	Finger adaptation
Our work	Object shape	Explicit	Dexterous hand	Yes	Finger adaptation

represents the mapping from grasp parameters and object shape to the grasp quality and new optimal grasps are found through interpolation on the manifold. Most of these methods are either limited to basic shapes or simple grasp primitives and cannot be used to execute a set of speci c contact locations. Along this direction, [31] learns the joint density function of hand pose and nger joints from a large set of grasps. This density function is used later to retrieve nger joints online given any query hand pose. However, learning is conducted in the object frame and with speci c hand-object combination. As a result, a new learning model is required for each new pair of hand-object combination. The authors in [32] learn two separate models, i.e., the contact model to express the relationship between n-gers and object local features, and the hand con guration model to represent whole hand con guration during approach to grasp. They show this approach can generalize to new objects for given grasp types. In this work, we follow a similar principle as [32] to address the grasp planning in two steps. With the help of Virtual Frame, planning of grasping points and learning of hand inverse kinematics are conducted independently, thus allowing for different hands to be used and making it also possible for different contact level grasp planners to be used in the system.

The research on grasp control has mainly focused on the design of control algorithms for achieving the desired grasping force that can either balance the external forces or exert proper ones on the object [33]. The approaches can be classi ed into two groups depending on whether the contact force is explicitly controlled or not. One is the hybrid position and force control (including the grasping force control) [34, 35, 36, 37] which controls the positions in some directions and the force in other directions simultaneously. The other is impedance control which regulates the contact forces implicitly by specifying the desired impedance [38, 39, 40, 41]. Both approaches have their own merits and disadvantages and a detailed comparison has been given in [33]. While these studies focus more on the grasp control after the ngers are closed, rather few works have been reported on devising control algorithms for the nger approaching and closing stage. Previous works usually assume that a position controller is rst used for approaching and then it is switched to a force controller once contact or collision is detected [26, 42]. However, the switching may easily cause the overshooting of the contact force due to the limitation in sensing and low control rate. To alleviate this problem, a parallel position/force controller is adopted as proposed in [43] with a tradeoff for smooth switching from position controller to force controller, depending on the distance to the target hand pose obtained from our learned probabilistic model. Furthermore, the control gain for the position and force controller are determined by the shape uncertainty and the contact forces uncertainty, rather than being hard-coded.

Considering uncertainty in robotic grasping, both for planning and control, has become increasingly important. To deal with various uncertainties in the grasping problem, one approach is to use sensory feedback to perform grasp adjustment locally so as to nd stable grasps near the original planned grasp [17, 19, 18]. For instance, [19] proposes a set of simple and efficient heuristics to reactively correct the alignment error of the PR2 gripper. In [18], a sensor-based grasp primitive of Barrett hand is developed to adapt to the variation of the task conditions. These methods are usually reactive using actual sensing data from force or tactile sensors. The reactive correction strategy is designed to alleviate the need for precise hand-object pose information and hence can be more robust to pose and location uncer-

tainty. The main problem of these methods is that, to design the corrective strategy, the grasp is usually limited to a prede ned set of grasp primitives [18] or only simple hand kinematics is considered [19]. For a more complex dexterous hand with a possibility to execute a large variety of grasps, it becomes more difficult to design such a corrective strategy. In our previous work [6], an object-level grasp adaptation is proposed to deal with physical uncertainties in object mass and friction, focusing mainly on the grasping stage when the object is already in the hand.

Another approach to deal with the uncertainty is to consider uncertainty during the planning process. One way is to incorporate robustness into the planning, preferring grasps that are somewhat insensitive to the uncertainty or search for stable graspable regions on the object [44, 45, 46, 24, 25]. For instance, the concept of independent contact regions (ICRs) is introduced to provide robustness to nger placement error [24] where any grasp with ngertip s positions inside the ICR will achieve a force-closure grasp. The uncertainty information can also be updated online using vision [23, 47] or tactile exploration [20, 21, 22]. However, these approaches usually use a set of prede ned grasps and require several rounds of grasp trials. A comparative study in robotic grasping dealing with uncertainty is given in Table 1. Few works have considered object shape uncertainty by integrating planning and control. However, in real robotic grasping tasks due to the, for example, occlusion problems [48, 49, 22] or non-reachability from tactile exploration [21], object shape uncertainty is inevitable. In this work, we thus integrate object uncertainty in grasp planning and control. In the planning stage, we can explicitly control the level of uncertainty of the nal grasp. During the execution stage, an adaptive nger closing strategy is used to compliantly place the ngers on the desired locations.

3. Grasp planning under shape uncertainty

In this section, we rst describe the GP for object shape modeling as well as the parametrization of associated shape uncertainty. Thereafter, a shape uncertainty constrained contact level grasp planning approach is presented. For clarity of the presentation, the notations adopted in this paper are summarized in Table 2.

3.1. Object modeling for grasping

For an unknown object, due to the limited viewing angle or occlusion, it may be difficult for a robot to observe its complete shape. The unseen parts of the object may be important from a grasping perspective as these may provide better contact locations. To synthesize contacts on unseen parts, we propose to model the whole object surface using GPs.

Notation	De nition
$\mathbf{x} \in \mathbb{R}^3$	a point in $\mathbb{R}^3$
$d \in \mathbb{R}$	signed distance
$\boldsymbol{\omega} \in \mathbb{R}^3$	normal direction
$\mathbf{y} = \{d, \boldsymbol{\omega}\} \in \mathbb{R}^4$	output of GP
$cov(\mathbf{y}^i, \mathbf{y}^j) \in \mathbb{R}^4$	covariance between $\mathbf{y}^i$ and $\mathbf{y}^j$
$E(\mathbf{y}_*) \in \mathbb{R}^{4\times1}$	predicted output of GP
$cov(\mathbf{y}_*) \in \mathbb{R}^{4\times4}$	the covariance with the prediction
$f_{cov}(\mathbf{x}_*)$	shape uncertainty, i.e., $[cov(\mathbf{y}_*)]_{11}$
$\mathbf{p}^i \in \mathbb{R}^3$	position of contact point
$\mathbf{n}^i \in \mathbb{R}^3$	normal direction at contact point
S_{thresh}	threshold on the shape uncertainty
$\mathbf{w}_i^j \in \mathbb{R}^6$	contact wrench
ϕ_j^i	coefficient of contact wrench
$\mathbf{p}^o \in \mathbb{R}^3$	origin of virtual frame
$R^o \in \text{SO}(3)$	orientation of virtual frame
$\Theta \in \mathbb{R}^h$	nger joints
$L \in \mathbb{R}^3$	distance between each ngertip and $\mathbf{p}^o$
$N \in \mathbb{R}^3$	pairwise inner product of $\mathbf{n}^i$
π_i	prior of the ith Gaussian component
$\mathcal{N}(\mu_i, \Sigma_i)$	Gaussian distribution
$T_{Hand}^{Obj} \in \mathbb{R}^{4\times4}$	hand pose in object frame
$\boldsymbol{x}_d \in \mathbb{R}^3$	desired ngertip position
$\boldsymbol{f}_d \in \mathbb{R}^3$	desired contact force
$\boldsymbol{K}_P \in \mathbb{R}^{3\times3}$	position control gain
$\boldsymbol{C}_F \in \mathbb{R}^{3\times3}$	force control gain

Table 2: List of notations

We denote by $\mathbf{x} \in \mathbb{R}^3$ an arbitrary point with normal direction $\boldsymbol{\omega} \in \mathbb{R}^3$, and by $d \in \mathbb{R}$ the relative position of $\mathbf{x}$ with respect to the object surface. We de ne a function

$$g(\mathbf{x}) : \mathbb{R}^3 \quad \mathbb{R}^4 \tag{1}$$

that maps the position of a point to its relative position and its outward normal direction as the basis for estimation of the object shape. In particular, the relative position $d = 0$ when the point is on the object, and $d \in \mathbb{R}^-$ or $d \in \mathbb{R}^+$ when the point is inside or outside of the object respectively.

For training the GP model of the object, the input is a training dataset denoted $\mathbf{X} = \{\mathbf{x}^i \in \mathbb{R}^3\}^{i=1\cdots n_t}$, composed of points on the object surface. It originates from point clouds of partially viewed object, consisting also of points inside and outside the object surface. The latter two kinds of points are included to increase the

accuracy of the GP estimation [50]. In practice, we rst normalize the points on the object surface to range $[-1, 1]$. The origin is then selected as the interior point for training with $d = -1$. For the outside points, 20 points are randomly sampled from a sphere with radius 1.2 with $d = 1$. The output of our training dataset is $\mathbf{Y} = \{\mathbf{y}^i = (d^i, \boldsymbol{\omega}^i) \in \mathbb{R}^4\}^{i=1\cdots n_t}$. The covariance between two outputs $cov(\mathbf{y}^i, \mathbf{y}^j) \in \mathbb{R}^4$ is de ned as:

$$cov(\mathbf{y}^i, \mathbf{y}^j) = \begin{matrix} cov(d^i, d^j) & cov(d^i, \boldsymbol{\omega}_1^j) & cov(d^i, \boldsymbol{\omega}_2^j) & cov(d^i, \boldsymbol{\omega}_3^j) \\ cov(\boldsymbol{\omega}_1^i, d^j) & cov(\boldsymbol{\omega}_1^i, \boldsymbol{\omega}_1^j) & cov(\boldsymbol{\omega}_1^i, \boldsymbol{\omega}_2^j) & cov(\boldsymbol{\omega}_1^i, \boldsymbol{\omega}_3^j) \\ cov(\boldsymbol{\omega}_2^i, d^j) & cov(\boldsymbol{\omega}_2^i, \boldsymbol{\omega}_1^j) & cov(\boldsymbol{\omega}_2^i, \boldsymbol{\omega}_2^j) & cov(\boldsymbol{\omega}_2^i, \boldsymbol{\omega}_3^j) \\ cov(\boldsymbol{\omega}_3^i, d^j) & cov(\boldsymbol{\omega}_3^i, \boldsymbol{\omega}_1^j) & cov(\boldsymbol{\omega}_3^i, \boldsymbol{\omega}_2^j) & cov(\boldsymbol{\omega}_3^i, \boldsymbol{\omega}_3^j) \end{matrix} \tag{2}$$

To compute all the entries in the covariance matrix, the following identities are required, which can be obtained from the selected kernel function $k(\cdot, \cdot)$ and its derivatives [51]:

$$cov(d^i, d^j) = k(\mathbf{x}^i, \mathbf{x}^j) \tag{3}$$

$$cov(\boldsymbol{\omega}_m^i, d^j) = \frac{}{x_m} cov(d^i, d^j) \tag{4}$$

$$cov(\boldsymbol{\omega}_m^i, \boldsymbol{\omega}_n^j) = \frac{^2}{x_m \ x_n} cov(d^i, d^j) \tag{5}$$
$$m = 1, 2, 3, n = 1, 2, 3;$$

For the kernel function, a thin plate kernel [52] is adopted to regularize the rst order continuity, assuming that the normal direction on the object surface is continuous. Given two inputs, the thin plate kernel function is computed by

$$k(\mathbf{x}^i, \mathbf{x}^j) = 2\|\mathbf{x}^i - \mathbf{x}^j\|^3 - 3 \ \|\mathbf{x}^i - \mathbf{x}^j\|^2 + \ ^3 \tag{6}$$

where $\in \mathbb{R}^+$ is the longest pairwise distance among all the training inputs. Note that is the only parameter in the kernel function and it can be easily determined from the training inputs without any optimization procedure involved, such as maximizing the likelihood. This is one of the main reasons for adopting it here in comparison to other kernel functions such as Radial Basis Function [53, 20].

Given the computed covariance matrix and a new data point $\mathbf{x}_* \in \mathbb{R}^3$, we can use GP to predict the function mean value $E(\mathbf{y}_*) \in \mathbb{R}^{4\times 1}$ and its corresponding variance $cov(\mathbf{y}_*) \in \mathbb{R}^{4\times 4}$, [50] by

$$E(\mathbf{y}_*) = E([d_*, \boldsymbol{\omega}_*^T]^T) = K_*[K(\mathbf{X}, \mathbf{X}) + \sigma^2 I)]^{-1}\mathbf{Y} \tag{7}$$

$$cov(\mathbf{y}_*) = K(\mathbf{x}_*, \mathbf{x}_*) - K_*[K(\mathbf{X}, \mathbf{X}) + \sigma^2 I)]^{-1}K_*^T \tag{8}$$

where $K_* = K(\mathbf{x}_*, \mathbf{X})$ denotes the $4 \times 4n_t$ covariance matrix evaluated at all pairs of the testing and training inputs, and similarly for $K(\mathbf{X}, \mathbf{X}) \in \mathbb{R}^{4n_t\times 4n_t}$ and $K(\mathbf{x}_*, \mathbf{x}_*) \in \mathbb{R}^{4\times 4}$. The parameter σ^2 re ects the variance of noise in the output.

Given Eq. (7), we can estimate if the point $\mathbf{x}_*$ is on the object surface or not using $E([d_*])$ as well as predicting the normal direction at this point using $E([\boldsymbol{\omega}_*])$. From Eq. (8), we can compute the uncertainty of our prediction. In our grasp planning method presented in next section, we only consider the shape uncertainty, which is the rst entry of $cov(\mathbf{y}_*)$, i.e., $[cov(\mathbf{y}_*)]_{11}$. As these three quantities only depend on the input $\mathbf{x}_*$, we use the following notations to represent each of them: $f_d(\mathbf{x}_*) = E([d_*])$, $f_\omega(\mathbf{x}_*) = E([\boldsymbol{\omega}_*])$ and $f_{cov}(\mathbf{x}_*) = [cov(\mathbf{y}_*)]_{11}$. Therefore, the nal object shape is expressed as the function:

$$f_d(\mathbf{x}) = 0, \mathbf{x} \in \mathbb{R}^3 \tag{9}$$

3.2. Grasp planning with shape uncertainty constraints

In this section, we rst describe how to explicitly incorporate shape uncertainties as constraints in grasp planning. We then proceed by explaining how to plan dexterous, three- ngered grasps by formulating planning as an optimization problem. We denote a contact point and its corresponding normal direction as $\mathbf{p}^i \in \mathbb{R}^3$ and $\mathbf{n}^i \in \mathbb{R}^3, i = 1, 2, 3$.

Points on surface and normal alignment: To execute the desired grasp, a basic constraint is that the planned contact locations are on the object surface. With the GP representation of the object surface, Eq. (9), this can be expressed as:

$$f_d(\mathbf{p}^i) = 0, \quad i = 1, 2, 3; \tag{10}$$

Besides this constraint, the normal direction of the contact points should be aligned with the object s surface normal which is represented as

$$\mathbf{n}^i = f_\omega(\mathbf{p}^i), \quad i = 1, 2, 3; \tag{11}$$

Shape uncertainty constraint: GP based object surface representation can be used to predict the shape uncertainty and it is taken into account in the grasp planning procedure as

$$f_{cov}(\mathbf{p}_i) < S_{\text{thresh}}, \quad i = 1, 2, 3; \tag{12}$$

where S_{thresh} is a threshold that de nes how much uncertainty a grasp can withstand.

Frictional form closure constraint: A grasp will be said to have frictional form closure if the origin of the

wrench space lies inside the convex hull of the contact wrenches [54]. This is formulated as follows

$$\exists \phi_j^i \in \mathbb{R}, \phi_j^i > 0, \sum_{i,j} \phi_j^i = 1, i = 1 \cdots m, j = 1 \cdots 3;$$
$$\text{s.t.} \sum_{i,j} \phi_j^i \mathbf{w}_i^j = \mathbf{0}; \tag{13}$$

where $\mathbf{w}_i^j \in \mathbb{R}^6, i = 1 \cdots m, j = 1 \cdots 3$ is the i-th primitive contact wrench of the j-th contact point [16], which depends on the contact points, contact normal directions and friction coefficient. Note that in [55] this property is also defined as force closure, here we adopt the definition in [54] where force closure additionally requires that the internal forces are controllable by the hand.

Objective function: Many different grasp quality metrics can be selected as the objective function for the contact level grasp planning [3]. Here, we adopt a simple objective function that minimizes the distance between the center of the contact points and the origin of the object frame as follows [1],

$$\min: \| \frac{1}{3} \sum_{i=1}^{3} \mathbf{p}^i \| \tag{14}$$

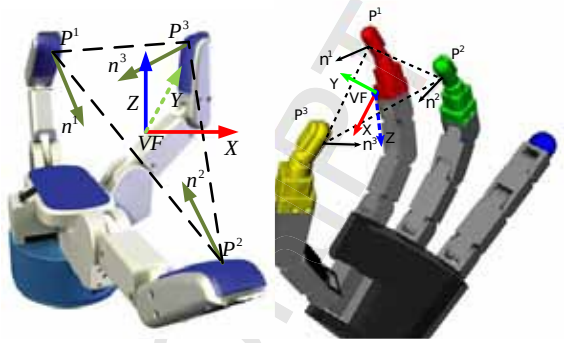
where $\| \cdot \|$ represents the 2−norm.

To generate grasping points, we formulate the grasp planning as a constrained optimization problem subject to the above constraints while minimizing the objective. By using AMPL (A Mathematical Programming Language) and adopting IPOPT (Interior Point OPTimizer) as the optimization solver [56], our method can generate multiple feasible solutions for each object by varying the initial points for IPOPT. More details and examples of grasp optimization are provided in Sec. 6.

4. Hand configuration in virtual frame

In this section, we present our learning-based approach for computing hand configurations needed to realize the planned grasping locations. We first describe our frame invariant encoding of hand configurations in terms of a virtual frame. Thereafter, we introduce a probabilistic model for representing the mapping between contact positions and hand configurations, used to compute hand joint configurations.

[1] Note that if the center of gravity is chosen as the origin of object frame, then this objective function actually attempts to minimize the effect of gravity on the grasp. In this paper, we use the geometric center of the object point cloud as the origin of the object frame



(a) Barrett hand (b) Allegro hand

Figure 2: The examples of Virtual Frame for Barrett hand and Allegro hand.

4.1. Probabilistic model for hand inverse kinematics

We want the mapping between contact positions and hand configurations to be frame invariant. To this end, we follow the idea presented in our previous work [41] that learns a probabilistic representation using the concept of *Virtual Frame*(VF) [57]. A VF defined in the hand frame can be expressed as:

$$T_{VF}^{Hand} = \begin{bmatrix} R^o & \mathbf{p}^o \\ [0,0,0] & 1 \end{bmatrix} \in \mathbb{R}^{4\times4} \tag{15}$$

where $\mathbf{p}^o$ is the origin of the VF with

$$\mathbf{p}^o = \frac{1}{3} \sum_{i=1}^{3} \mathbf{p}^i \tag{16}$$

and $\mathbf{p}^i \in \mathbb{R}^3, i = 1, 2, 3$, is the position of the i-th fingertip. The orientation of the frame is defined by

$$R^o = [\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z] \in \text{SO(3)} \tag{17}$$
$$\mathbf{r}_x = \frac{\mathbf{p}^3 - \mathbf{p}^1}{\|\mathbf{p}^3 - \mathbf{p}^1\|}$$
$$\mathbf{r}_y = \mathbf{r}_z \times \mathbf{r}_x$$
$$\mathbf{r}_z = \frac{(\mathbf{p}^2 - \mathbf{p}^1) \times \mathbf{r}_x}{\|(\mathbf{p}^2 - \mathbf{p}^1) \times \mathbf{r}_x\|}$$

Two examples of the VF for Barrett hand and Allegro hand are shown in Fig. 2. With the definition of the VF, we encode a hand configuration $\mathcal{G}$ as

$$\mathcal{G} = \{\Theta, L, N\} \tag{18}$$

where $\Theta \in \mathbb{R}^h$ is the finger joint. $L = [L_1, L_2, L_3] \in \mathbb{R}^3$ is the distance between each fingertip and the origin of the VF, i.e., $L_i = \|\mathbf{p}^i - \mathbf{p}^o\|$. $N = [N_1, N_2, N_3] \in \mathbb{R}^3$ is the pairwise difference of normal direction in the sense

of inner product, $N_1 = \mathbf{n}^1 \cdot \mathbf{n}^2, N_2 = \mathbf{n}^1 \cdot \mathbf{n}^3, N_3 = \mathbf{n}^2 \cdot \mathbf{n}^3$. Given this encoding, all the variables Θ, L, N are frame invariant, and so is $\mathcal{G}$.

Given a set of hand con gurations $\{\mathcal{G}^i, i = 1 \cdots N_g\}$ we can learn a probabilistic model – Gaussian Mixture Model (GMM) to represent the joint density of $\{\Theta, L, N\}$. This set of hand con gurations can be obtained from simulation by sampling in the joint space, or from human demonstration through demonstration learning. The likelihood of a grasp $\mathcal{G}_* = (\Theta_*, L_*, N_*)$ under a GMM model, denoted by Ω with m Gaussian components is given by

$$p(\mathcal{G}_* \, \Omega) = \sum_{i=1}^{m} \pi_i \mathcal{N}(\mathcal{G}_* \, \mu_i, \Sigma_i) \tag{19}$$

where π_i is the prior of the ith Gaussian component and $\mathcal{N}(\mu_i, \Sigma_i)$ is the Gaussian distribution with mean μ_i and covariance Σ_i as:

$$\mu_i = \begin{matrix} \mu_{\Theta,i} \\ \mu_{L,i} \\ \mu_{N,i} \end{matrix}, \Sigma_i = \begin{matrix} \Sigma_{\Theta\Theta,i} & \Sigma_{\Theta L,i} & \Sigma_{\Theta N,i} \\ \Sigma_{L\Theta,i} & \Sigma_{LL,i} & \Sigma_{LN,i} \\ \Sigma_{N\Theta,i} & \Sigma_{NL,i} & \Sigma_{NN,i} \end{matrix} \tag{20}$$

The number of Gaussian components, i.e., m is determined using Bayesian information criterion (BIC) and the parameters of π_i, μ_i, Σ_i in the model are trained using expectation-maximization (EM) algorithm to maximize the likelihood of all the trained grasps. More details for training and testing the probabilistic model for a speci c hand will be given in Sec. 6.3.

4.2. Online hand con guration query

Given a set of desired grasping points, we can now compute feasible hand con gurations for realizing a grasp using the learned probabilistic model. The key idea here is to de ne two VFs using the ngertips and the grasping points respectively. Thus, the goal of online query is to search for a match of these two VFs. An example of online hand con guration query is shown in Fig. 3. We now explain the details of the query process in detail.

Query hand feasibility: Given a set of desired grasping points, we rst construct a VF, $T_{VF}^{Obj} \in \mathbb{R}^{4\times4}$, similarly constructing the VF for hand con gurations in Eq. (15). We then compute the corresponding L and N for this VF and determine if the current query point, i.e., $q = (L, N)$ is likely enough with respect to the learned model Ω. This step computes the reachability of the hand given the grasping points. For instance, if two points are too far away and the distance between them is larger than the maximal spread length of the nger,

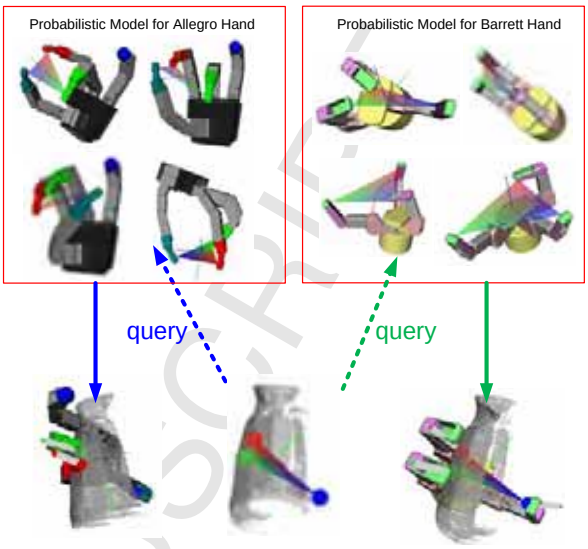


Figure 3: Given a set of grasping points on the object, we rst construct a VF and use it as a key to compute corresponding hand con gurations given the learned probabilistic model. In this gure, we also show a set of grasping points that can be reached by two different hands, which also explains the idea of being able to use different hands to execute grasps.

these grasping points cannot be realized by the given hand. For this purpose, we use the Mahalanobis distance from q to the center of each Gaussian component. The distance to the i-th component is de ned as:

$$f_i(q, \Omega) = \frac{1}{2}(q - \mu_{q,i})^T \Sigma_{q,i}^{-1}(q - \mu_{q,i}) \tag{21}$$

where $i = 1, ..., m$ is the index of Gaussian components, $\mu_{q,i}$ and $\Sigma_{q,i}$ are the corresponding components in Eq. (19) as follows:

$$\mu_{q,i} = \begin{bmatrix} \mu_{L,i} \\ \mu_{N,i} \end{bmatrix}, \Sigma_{q,i} = \begin{bmatrix} \Sigma_{LL,i} & \Sigma_{LN,i} \\ \Sigma_{NL,i} & \Sigma_{NN,i} \end{bmatrix} \tag{22}$$

We consider that the likelihood that a query point q belongs to the learned model is high enough if $\exists i, \; i = 1, \ldots, m, \; f_i(q, \Omega) < 2$. In other words, if the query point is within two standard deviations of any Gaussian component of the model, it is considered to be close enough to the learned model. Otherwise, the grasping points are considered as infeasible for the given hand. Note here when we query the hand feasibility, the possible collisions between the hand and object are not taken into account. However, the collision is checked in simulation (*OpenRave* [58]) before the nal selected grasp is executed.

Query Finger joints: When the current query point q is likely enough under the model, the desired nger joints Θ are obtained by taking the expectation over

the conditional distribution, $p(\Theta\, L, S, \Omega)$, which can be computed as follows [59]:

$$E\{p(\Theta\, L, N, \Omega)\} = \sum_{i=1}^{m} h_i(\mu_{\Theta,i} + \Sigma_{\Theta q,i}\Sigma_{q,i}^{-1}(q - \mu_{q,i})) \quad (23)$$

where $\Sigma_{\Theta q,i} = \begin{bmatrix}\Sigma_{\Theta L,i}\\ \Sigma_{\Theta S,i}\end{bmatrix}$, and $h_i = \frac{\pi_i \mathcal{N}(q\,\mu_{q,i},\Sigma_{q,i})}{\sum_{j=1}^{m} \pi_j \mathcal{N}(q\,\mu_{q,j},\Sigma_{q,j})}$.

Note that for three grasping points, we can have six different query points by permuting the correspondence between grasping points and the nger index. Therefore, given three grasping points, we may nd several different hand con gurations that can realize the grasping points. However, it is also possible that none of the six query points is likely enough under the model (e.g., object too big or too small), which implies that the given grasping points cannot be realized by the considered hand.

Query hand pose: After obtaining joint angles for each nger, we can again construct a VF $T_{VF}^{Hand} \in \mathbb{R}^{4\times4}$ using the ngertips position: This can be obtained using the hand s forward kinematics. Note that this VF is represented in the hand frame. Thus, the desired hand pose in the object frame, i.e., $T_{Hand}^{Obj} \in \mathbb{R}^{4\times4}$, can be obtained as follows:

$$T_{Hand}^{Obj} = T_{VF}^{Obj} T_{VF}^{Hand-1} \quad (24)$$

Due to the probabilistic model we use to compute hand con gurations, there can be errors between the realized ngertip positions and the desired grasping points. This position error is taken care of by our compliant grasp controller presented in the next section.

5. Grasp control under shape uncertainty

Our approach superimposes position and force control, taking both the shape and contact force uncertainty into account. The control scheme for a given nger is represented as

$$\Delta\boldsymbol{\theta} = \boldsymbol{J}^{-1}\left[(1-\lambda)\boldsymbol{K}_P(\boldsymbol{x}_d - \boldsymbol{x}_c) + \lambda\boldsymbol{C}_F(\boldsymbol{f}_d - \boldsymbol{f}_c)\right] \quad (25)$$

where $\boldsymbol{J}$ is the Jacobian of a nger. $\boldsymbol{x}_d$ ($\boldsymbol{f}_d$) $\in \mathbb{R}^3$ and $\boldsymbol{x}_c$ ($\boldsymbol{f}_c \in \mathbb{R}^3$) are respectively the desired and current ngertip positions (contact normal force), both of which are expressed in the hand frame. $\boldsymbol{K}_P$ and $\boldsymbol{C}_F$ are controller gains. $\lambda \in [0, 1]$ is a positional error measure to estimate how close the nger is to its desired position and weighted by the inverse of shape uncertainty as

$$\lambda = \exp\ -\frac{1}{2}(\boldsymbol{x}_d - \boldsymbol{x}_c)^T\Sigma_{cov}^{-1}(\boldsymbol{x}_d - \boldsymbol{x}_c) \quad (26)$$

By changing λ, the position controller is designed to rst dominate and then smoothly switch to the force controller. Hence, a position error will be tolerated along the contact normal direction in order to regulate the contact force. Note that the desired ngertip position $\boldsymbol{x}_d$ is the grasping point, i.e., $\mathbf{p}^i, i = 1, 2, 3$ in Sec. 3.2, but represented in the hand frame. The desired force $\boldsymbol{f}_d$ is estimated as in our previous work [6]. Moreover, the estimation can also provide the variance of the expected force value in the hand frame, i.e., $\Sigma_{f_d} \in \mathbb{R}^{3\times3}$.

The diagonal matrices $\boldsymbol{K}_P \in \mathbb{R}^{3\times3}$ and $\boldsymbol{C}_F \in \mathbb{R}^{3\times3}$ are respectively the gain for position control and force control, which are usually selected heuristically. Here, we use the information from the variance of the desired position and desired force to choose proper parameters. For $\boldsymbol{K}_P$, we set it inversely proportional to the variance of the $\mathbf{p}^i$, i.e., $f_{cov}(\mathbf{p}^i)$ in Eq. (12). The variance of the desired position is along the normal direction $f\ (\mathbf{p}^i)$ (Eq. (11)) in the object frame, which can be transformed to the hand frame, denoted as Σ_{cov}. We have:

$$\boldsymbol{K}_P = \alpha_p \Big|\underbrace{[diag(R_{Obj}^{Hand} f_{cov}(\mathbf{p}^i) f\ (\mathbf{p}^i))]^{-1}}_{\Sigma_{cov}^{-1}}\Big| \quad (27)$$

where R_{Obj}^{Hand} is the rotation from hand frame to object frame, which can be obtained from Eq. (24). $diag(\cdot)$ means the diagonal entries of a matrix and $|\cdot|$ is the absolute value for each entry of a matrix. $\alpha_p \in \mathbb{R}^+$ is a scaling parameter. Similar for $\boldsymbol{C}_F$, we have,

$$\boldsymbol{C}_F = \alpha_f \left|[diag(\Sigma_{f_d})]^{-1}\right| \quad (28)$$

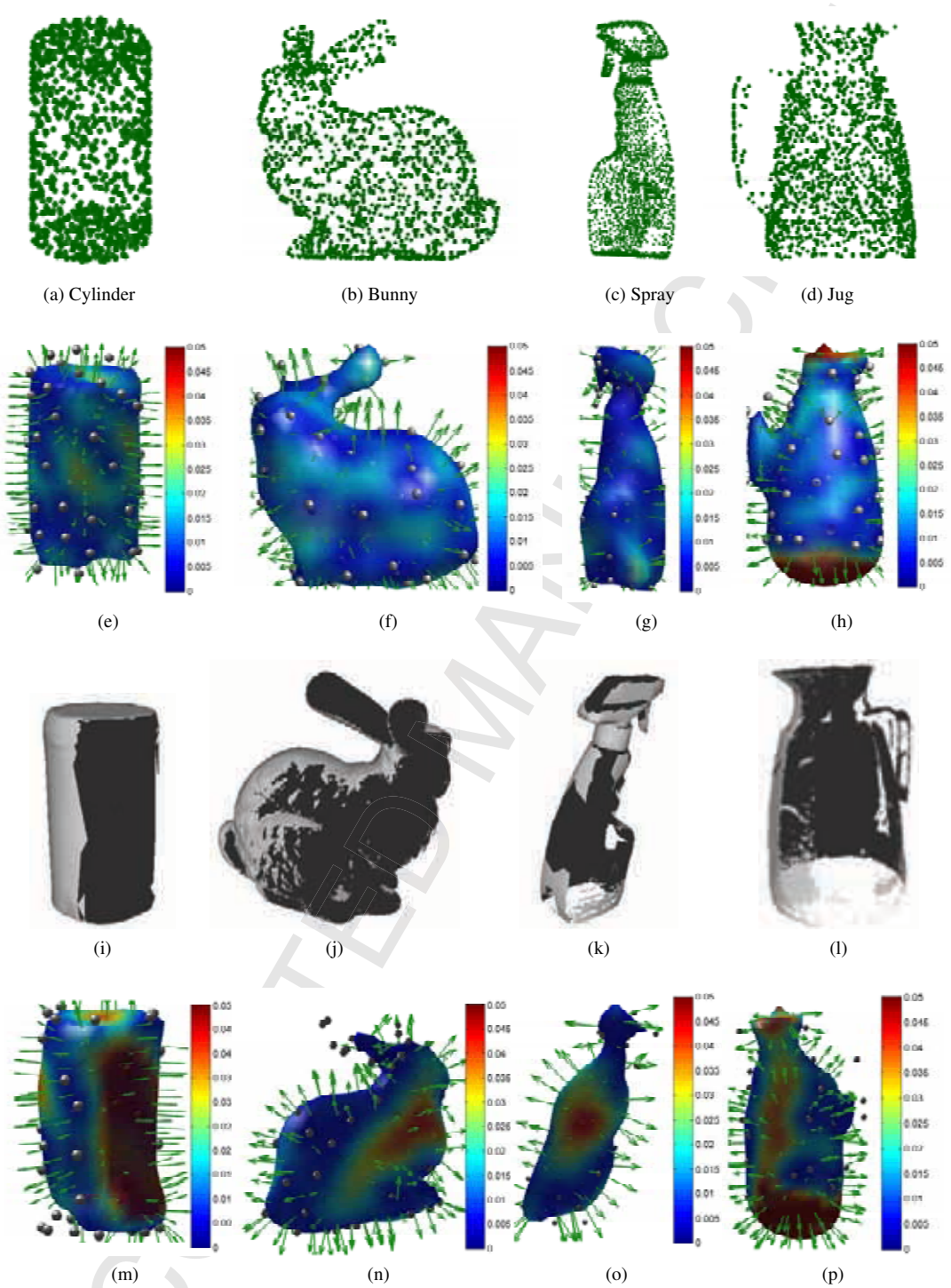
The intuition behind the selection $\boldsymbol{K}_P$ and $\boldsymbol{C}_F$ is that when there is a large uncertainty on the desired contact point location, the nger will move more slowly by choosing a smaller $\boldsymbol{K}_P$. When there is a large variance in the desired contact normal force, we make the contribution of force controller smaller by using a smaller $\boldsymbol{C}_F$. This also implies that the nger will rst contact the grasping point that has smaller uncertainty, and it is demonstrated in the experiments that this scheme can largely improve the grasp success rate.

6. Implementation and experimental results

In this section, we will present our implementation and experimental results for object surface modeling and grasp planning using the Allegro hand.

6.1. Results for object surface modeling

We evaluate our GP based object shape modeling method on four different objects: a cylinder, a bunny



(a) Cylinder (b) Bunny (c) Spray (d) Jug

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

Figure 4: Four 3*D* objects and their corresponding GP representations. The first row show the original 3D point cloud object models, (a) a cylinder, (b) a bunny rabbit, (c) a spray, (d) a jug. The second row shows object shapes modeled by GP with whole object point cloud. Spheres on the object model are the 3D points on the object surface used to train the GP model. The arrow on the surface represent the normal direction predicted by GP on that point. The color of the surface represents the variance of the shape prediction on that point. The third row shows partial object point cloud from a fixed camera. The fourth row shows object shapes modeled by GP with partial object point cloud.

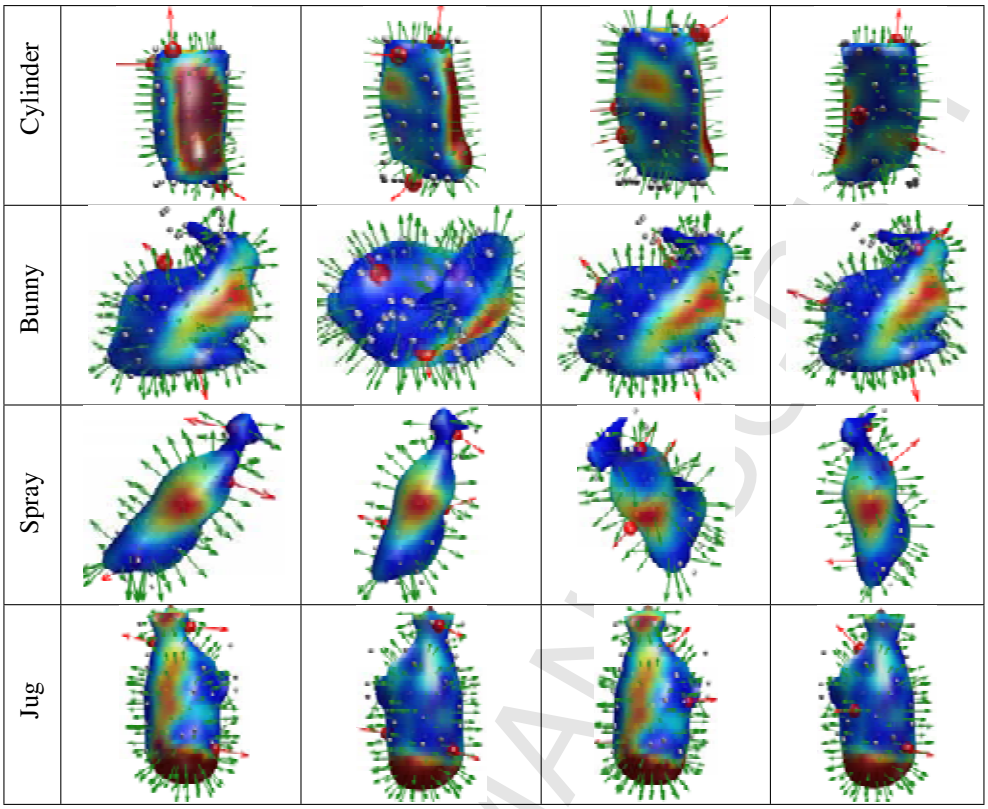


Figure 5: Four of the obtained grasping points (red balls on the surface) for each object and the red arrows represent the normal directions. The grasping points are outside the uncertain region.

rabbit, a spray bottle and a jug, as shown in Fig. 4. The object point clouds are obtained from laser scanner and 1000 data points are randomly sampled from the original point cloud. To speed up the object shape modeling procedure, we further adopt a GP-based filter to select the most informative data points [21] for representing the GP. The filtered data points for GP are shown as spheres on the object surface, Fig. 4(e)-(h). Table 3 shows the number of filtered GP data points and the computation time for filtering and shape modeling [2].

As shown in Fig. 4(e)-(h), when the training data points are sampled from the whole object point cloud, the variance or shape uncertainty is generally very small on the whole surface except the parts with sparse or with even no data points, such as the bottom of the jug. In robotic grasping tasks, due to the occlusion [48, 49] or non-reachability from tactile exploration [21], it is usually the case that some parts of the object are not perceivable and point-clouds exhibit holes. To evaluate our method under missing data points, we use MeshLab [3] to simulate partial view of point clouds with a fixed camera view, and then obtain object point cloud from that virtual camera. The results of object shape modeling with partial object point cloud are shown in the last two rows of Fig. 4. We can see that although the objects are partially viewed, our method can still model the shapes, and importantly, with explicitly computed uncertainties.

Table 3: The number of training data points (Nb.) for GP and the computation time for filtering(Time1) and shape modeling (Time2) on a 8 GB machine with a cpu at 2.4 GHZ.

Object	Nb.	Time1(s)	Time2(s)
Cylinder	59	24.51	0.34
Bunny	71	24.15	0.51
Spray	61	22.62	0.28
Jug	71	16.25	0.31

[2]Note that the main time consumption comes from the filtering, this, however, can be done during the data collection if the object point cloud is collected online using vision or tactile exploration.

[3]http://meshlab.sourceforge.net/

6.2. Results for grasp planning

For each object model shown in the fourth row of Fig. 4, 1000 initial points for each of grasping point $\mathbf{p}^i \in \mathbb{R}^3, i = 1, 2, 3$ are randomly sampled from a sphere with radius 0.1. The coefficient of friction is set to 0.8. In our implementation, we set $S_{\text{thresh}} = 0.03$, which is three times larger than the noise level σ in Eq. (7). The number of final optimal grasps, the average computation time and the grasp quality are shown in Table 4. Some examples of the obtained optimal grasps for each of the four objects are shown in Fig. 5. It can be noticed that all the grasping points are in the area with small uncertainty due to the explicit shape uncertainty constraint.

Table 4: The number of final optimal grasps (Nb.) out of 1000 trials and the average computation time (Time) for each trial and the grasp quality (Q) using Eq. (14).

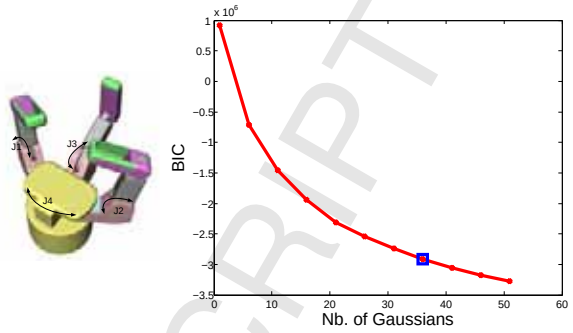
Object	Nb.	Time(s)	Q(cm)
Cylinder	797	17.51 ± 8.03	0.065 ± 0.064
Bunny	864	18.31 ± 10.25	0.0104 ± 0.029
Spray	986	8.46 ± 3.25	0.013 ± 0.32
Jug	914	26.52 ± 13.58	0.38 ± 0.75

6.3. Results for hand configuration query

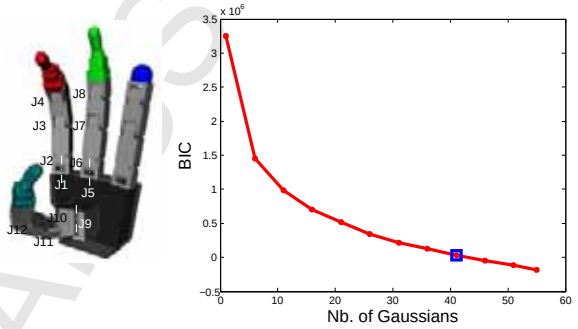
In this evaluation, we use two robotic hands – the 4 DOF Barrett hand shown in Fig. (6a), and the 16 DOF Allegro hand shown in Fig. (6c), as examples to evaluate the effectiveness of the learned probabilistic model described in Sec. 4.1. For each hand, we randomly sample $N_g = 10^6$ self-collision free hand configurations in the finger joint space in *OpenRave* [58], and then use this dataset for further model training and testing.

For the model evaluation on Barrett hand, we use the first 4×10^5 hand configurations for model training and the rest for evaluation. For this hand, each data point is 10-dimensional: $\Theta \in \mathbb{R}^4, L \in \mathbb{R}^3, N \in \mathbb{R}^3$. The number of Gaussians in Eq. (19) is set to $m = 36$, as shown in Fig. (6b)[4]. Using all the data from the test dataset as grasping points queries, we evaluate the accuracy of the model in terms of the Mean Absolute Error (MAE) in radians, as reported in Table. 5. Fig. 7 shows examples of predicted joint angles in comparison with the ground truth joint angles for four different testing cases, the MAE of these four examples are also reported in Table. 5.

[4]The BIC gives a range of the number of Gaussians and the final number is determined by a 10-folds cross validation on a separate dataset.



(a) Barrett Hand (b) BIC of Barrett Hand



(c) Allegro Hand (d) BIC of Allegro Hand

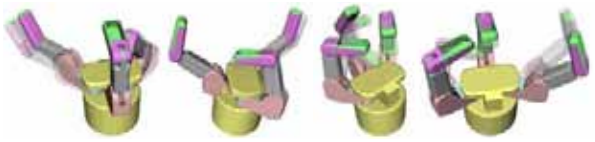
Figure 6: The hands models for the testing examples and the selection of number of Gaussians for training the model using BIC.

To keep the number of grasping points the same for the Allegro hand, we consider only its first three fingers as show in Fig. 7. Therefore, we have $\Theta \in \mathbb{R}^{12}$, and the training data is 18-dimensional. Same as before, we use the first 4×10^5 hand configurations for model training and the rest for testing. The number of Gaussians is chosen as $m = 41$, as shown in Fig. (6d). For the overall evaluation on the Allegro hand, the 6×10^5 data points have been used to test the probabilistic model, and the MAE with the standard deviation for the 12 joint angles is reported in Fig. 8.

From Table 5 and Fig. 8, the average error is around 0.2 rad. Note that the performance of our probabilis-

Table 5: The Mean Absolute Error between the prediction and ground truth joint angles for the four cases shown in Fig. 7 and the average error with the standard deviation over all 6×10^5 testing data for the evaluation on Barret hand. (Unit: Radian)

Joint	case 1	case 2	case 3	case 4	Overall
J1	0.03	0.06	0.27	0.26	0.16 ± 0.01
J2	0.17	0.01	0.27	0.002	0.17 ± 0.01
J3	0.04	0.07	0.25	0.08	0.21 ± 0.007
J4	0.03	0.2	0.06	0.56	0.36 ± 0.02



(a) case 1 (b) case 2 (c) case 3 (d) case 4

Figure 7: The joint angle prediction error of four examples for Barrett hand. The solid color hand shows the ground truth joint angles and the transparent color shows the predicted joint angles.

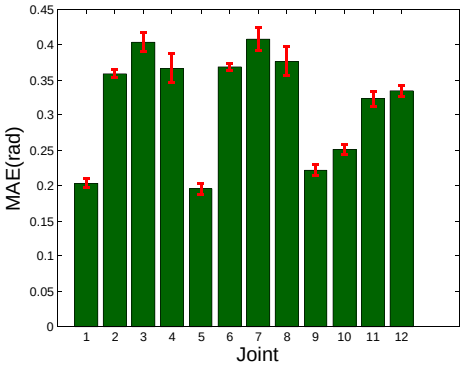


Figure 8: The predition error for the 12 joints of Allegro hand.

tic hand inverse kinematics model depends largely on how we sample the valid hand configurations. In this paper, we sample the hand configurations randomly in the finger joint space with self-collision rejection. We noticed that our dataset includes a lot of hand configurations that are self-collision free, but are unlikely to be valid grasp configurations. This is one cause of deterioration of the model performance. If some other information regarding the infeasibility of some postures (e.g. through models of the finger synergy or through human teaching) was provided to guide the sampling, this would likely improve the performance of the feasible solutions generated by our hand inverse kinematic model. This is one of our future working directions.

At this moment, to improve the performance, we adopt a local derivative-free optimization technique, called Constrained Optimization by Linear Approximation (COBYLA) [60]. This optimization algorithm is based on linear approximation of the optimized objective function and all the constraints and transform the original problem to a linear program to solve. With this optimization technique, we can locally adjust the hand pose as well as the finger joints to improve the performance with respect to the objective function, which is chosen as the sum of the distance between desired grasping points and fingertip positions.

6.4. Results for grasp realization

In this section, we show qualitative examples of grasp execution described in Section 4. Fig. 9 shows that the same grasping points can be realized by two different hand configurations for the same hand. In Fig. 10, we show that the same grasping points can also be realized by two different hands.

(a) (b) (c) (d)

Figure 9: Two examples show that the same grasping points can be realized by the same hand but with two different hand configurations. The frame attached with the object is the obtained VF and the arrows at the contact points are the predicted normal directions.

(a) (b) (c) (d)

Figure 10: Two examples show that the same grasping points can be realized by two different hands

In Fig. 11 and Fig. 12, we show some grasps for each hand using our learned probabilistic model. Note that all the grasps are generated using object models with partial point cloud, see Fig. 4. Collision between the hand and the object has also been checked in this step and invalid grasps due to collision are discarded.

6.5. Implementation on real robotic hand

Using the proposed grasp controller, we demonstrate several of our planned grasps using an Allegro hand mounted on a 7 DOF arm – KUKA LWR. Each fingertip of the Allegro hand is equipped with BioTac tactile sensors [5] to estimate the contact normal force, namely $\boldsymbol{f}_c \in \mathbb{R}^3$ in Eq. (25). After our calibration, we can obtain a force prediction with an accuracy around 0.1N in the normal direction and 0.3N in the tangential directions. In this work, we only control the normal force, i.e., $\boldsymbol{f}_c$ represented in the hand frame. The object to be grasped is placed on the table and the position is obtained using vision tracking system – OptiTrack[6]. Once

[5] http://www.syntouchllc.com/

[6] https://www.naturalpoint.com/optitrack/

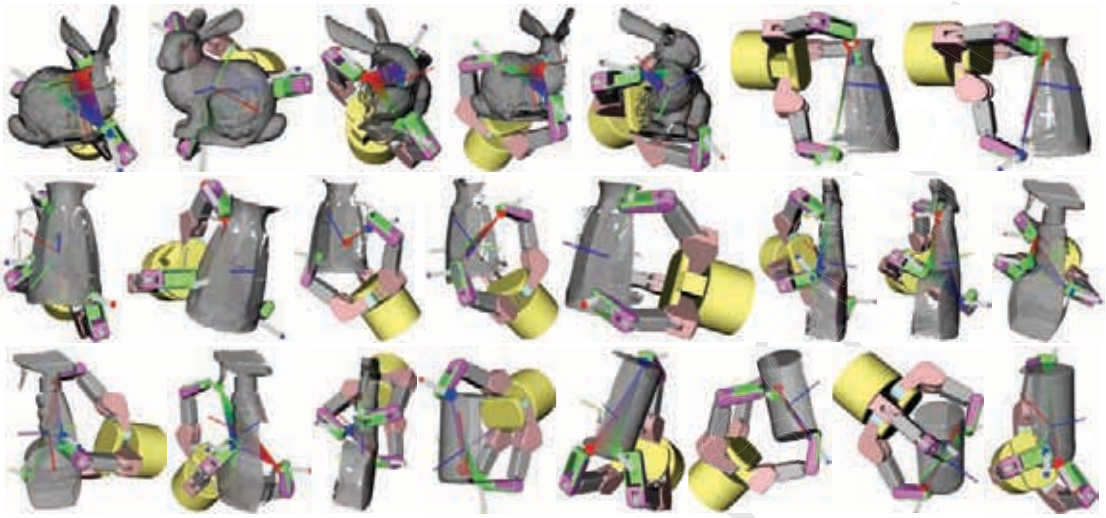


Figure 11: Some example grasps for Barrett hand. The red, green and blue point corresponds to finger 1, 2, 3 respectively. Notice that there are still some position errors between the fingertips and desired grasping points due to the probabilistic model we use.

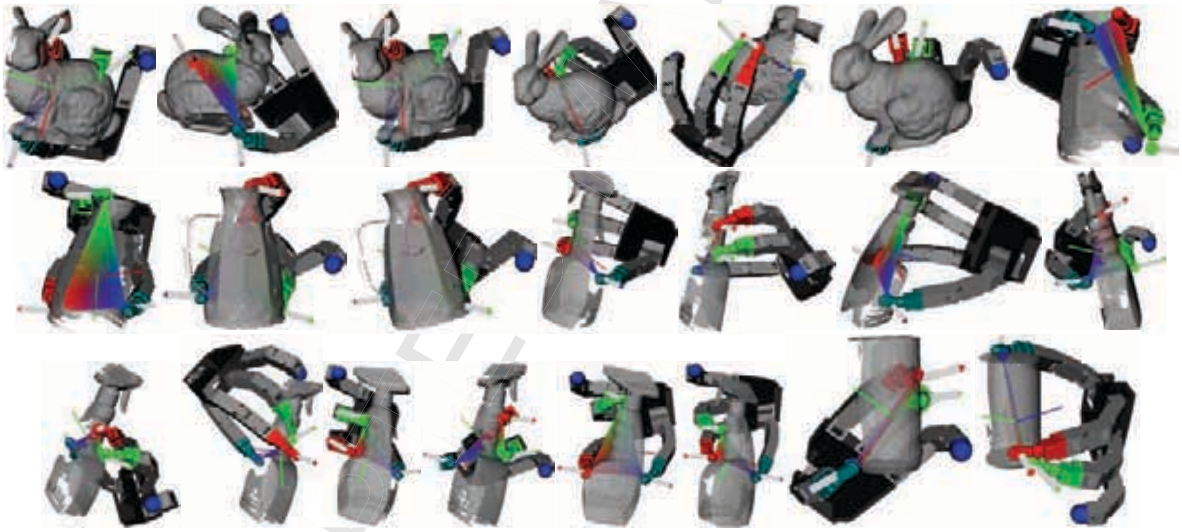


Figure 12: Some example grasps for Allegro hand. The red, green and blue point corresponds to finger 1, 2, 3 respectively.

our grasp controller has assessed that all three fingertips are at the desired positions and that the contact forces are stabilized, we command the robot to lift the object and switch the hand controller to an object-level impedance controller [6], in order to adaptively regulate the grasp stiffness and keep the grasp stable. Some of the realized grasps are shown in Fig. 13.

To evaluate how shape uncertainties affect grasp planning, we relax the upper bound of shape uncertainty S_{thresh} in Eq. (12), and generate grasps with different level of uncertainties. The obtained grasps are ranked according to the sum of uncertainty on each grasping point and the first three grasps with the largest score are selected for implementation on the real robotic hand. When the uncertainty increases, one of fingertips is more likely to contact with the object first and then push

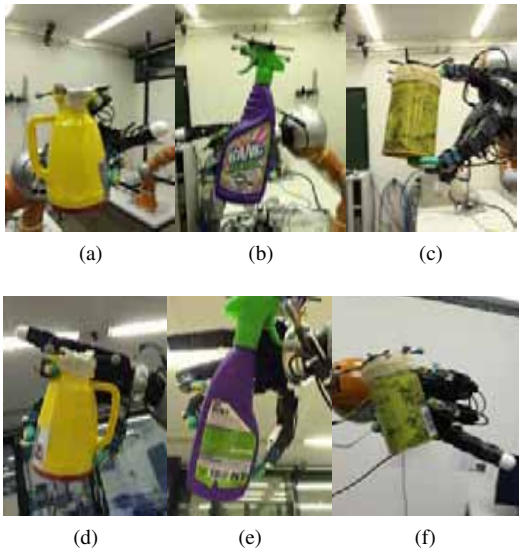


Figure 13: The implementation results of several planned grasps using Allegro hand. For the jug and spray bottle, the objects are placed on the table before lifting. For the tea can, the object is held by a human in order to realize grasp (c) which is unreachable from the bottom.

the object away (or tilt the object), while the other ngertips are still approaching the object. As a result, the other ngertips may end up being far away from their desired positions and thus less likely to achieve a stable grasp.

For each grasp, the test is repeated ten times to capture the uncertain effect from pushing motion of ngertips [7]. The percentage of the grasps that are stable over all the trials and grasps for each object is reported in Table 6. We can see that the percentage of stable grasps after lifting decreases when the grasps become more risk-seeking, i.e., grasping on the object surface with large shape uncertainties.

Table 6: Percentage of stable grasps achieved for each object under with different level of shape uncertainty, averaged over 30 trials (3 grasps x 10 times).

Object	Cylinder	Spray	Jug
$S_{\text{thresh}} < 0.03$	40%	63.3%	56.7%
$S_{\text{thresh}} < 0.06$	43.3%	33.3%	30%
$S_{\text{thresh}} < 0.10$	30%	26.7%	23.3%

To evaluate the performance of the grasp controller for nger closing, we compare it with a position controller for each ngertips with same isotropic gain[8], and the nger stops when the contact force reaches $0.5N$ or the nger reaches desired position. The same grasps in the rst row of Table 6 are tested with two different controllers: uncertainty-aware controller and position controller. The percentage of stable grasp achieved is reported in Table 7, which shows clear improvements given by our uncertainty-aware controller. As we have observed in the experiments, the object is more likely to be moved away when using a position controller with an isotropic gain. This is usually due to the fact that one ngertip would get contact with the object where it has larger uncertainty, and thus the object is shifted before other contacts are made. However, with our uncertainty-aware controller, the position error becomes less important when the ngertip is close to the desired position, and in most cases, the ngertips usually rst get contacts on the object where it has smaller uncertainty, and thus smaller position error and less possibility to shift the object away. Note that due to the imprecise hand dynamics, large friction at nger joints and joint actuator limitation, the total success rate is still far from practical use, which implies a requirement for reliable hand embodiment.

Table 7: Percentage of stable grasps achieved for each object using different controllers: uncertainty-aware controller(**unc.**) and position controller(**pos.**), averaged over 30 trials (3 grasps x 10 times).

Object	Cylinder	Spray	Jug
unc.	40%	63.3%	56.7%
pos.	16.7%	10%	13.3%

7. Conclusion

While dexterous grasping is considered important for in-hand object manipulation, it is still very difficult to realize using real robotic hands. One of the main challenges resides in how to overcome the uncertainties in sensing, actuation and imperfect representation of the environment. This work addressed this challenge and considered shape uncertainty both in grasp planning and control stages. During grasp planning, the uncertainty of the generated grasp can be explicitly determined. Moreover, during grasp execution, the uncertainty of the generated grasp is fed into a compliant grasp closing controller to further improve the grasp stability.

In this work, we proposed an approach for grasp planning and control considering object shape uncertainty.

[7] During the ten trials, the initial condition is the same. However, in practice we notice that the pushing from one ngertip may lead to very different object position and thus the outcome of nal grasps.

[8] In practice, a PI controller is used and the gain is hand tuned to achieve the best performance from our experience.

A probabilistic model for estimation of hand inverse kinematics model is adopted to compute feasible hand con gurations. During grasp execution stage, a compliant nger-closing controller taking into account the uncertainty has been devised to improve and retain grasp stability. Experiments on a real robotic hand demonstrate the effectiveness of our proposed method.

As a nal remark, we list a number of limitations and future research directions. First, our controller considers only one type of uncertainties, namely uncertainties linked to the shape of the object. Other sources of uncertainties (e.g. imprecise nger positioning, inaccurate model of the object s mass and friction coefficient) affect importantly the chance of success of a grasp. The grasp planning and control in this work is limited to shape uncertainty while other sources of uncertainties are not taken into account. However, nding an analytical representation of these uncertainties and considering them in robotic grasping will be a promising direction for the extension of this work.

Acknowledgments

This work was supported by the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement n^{o} 288533 ROBOHOW.COG.

References

[1] T. A. J. Bohg, A. Morales, D. Kragic, Data-driven grasp synthesis a survey, IEEE Transactions on Robotics 30 (2) (2014) 289–309.

[2] C. Goldfeder, P. K. Allen, Data-driven grasping., Autonomous Robots 31 (1) (2011) 1–20.

[3] M. Roa, R. Surez, Grasp quality measures: review and performance, Autonomous Robots (2014) 1–24doi:10.1007/s10514-014-9402-3.

[4] M. Ciocarlie, K. Hsiao, E. G. Jones, S. Chitta, R. B. Rusu, I. A. Sucan, Towards reliable grasping and manipulation in household environments, in: Intl. Symposium on Experimental Robotics (ISER), New Delhi, India, 2010.

[5] A. Bicchi, Hands for dexterous manipulation and robust grasping: a difficult road toward simplicity, Robotics and Automation, IEEE Transactions on 16 (6) (2000) 652–662. doi:10.1109/70.897777.

[6] M. Li, Y. Bekiroglu, D. Kragic, A. Billard, Learning of grasp adaptation through experience and tactile sensing, in: Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, 2014, pp. 3339–3346. doi:10.1109/IROS.2014.6943027.

[7] V. Nguyen, Constructing stable grasps in 3D, in: Robotics and Automation. Proceedings. 1987 IEEE International Conference on, Vol. 4, 1987, pp. 234–239.

[8] C. Ferrari, J. Canny, Planning optimal grasps, in: Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, 1992, pp. 2290–2295 vol.3. doi:10.1109/ROBOT.1992.219918.

[9] B. Mirtich, J. Canny, Easily computable optimum grasps in 2-d and 3-d, in: Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on, 1994, pp. 739–747 vol.1. doi:10.1109/ROBOT.1994.351399.

[10] X. Zhu, H. Ding, Planning force-closure grasps on 3-d objects, in: Robotics and Automation, 2004. Proceedings. ICRA 04. 2004 IEEE International Conference on, Vol. 2, 2004, pp. 1258–1263 Vol.2. doi:10.1109/ROBOT.2004.1307997.

[11] C. Borst, M. Fischer, G. Hirzinger, Calculating hand con gurations for precision and pinch grasps, in: Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on, Vol. 2, 2002, pp. 1553–1559 vol.2. doi:10.1109/IRDS.2002.1043976.

[12] C. Rosales, L. Ros, J. M. Porta, R. Surez, Synthesizing grasp con gurations with speci ed contact regions., I. J. Robotic Res. 30 (4) (2011) 431–443.

[13] M. Santello, M. Flanders, J. F. Soechting, Postural hand synergies for tool use, The Journal of Neuroscience.

[14] M. T. Ciocarlie, P. K. Allen, Hand posture subspaces for dexterous robotic grasping, The International Journal of Robotics Research 28 (7) (2009) 851–867. doi:10.1177/0278364909105606.

[15] S. El Khoury, M. Li, A. Billard, Bridging the gap: One shot grasp synthesis approach, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2012.

[16] S. El-Khoury, M. Li, A. Billard, On the generation of a variety of grasps, Robotics and Autonomous Systems 61 (12) (2013) 1335–1349. doi:10.1016/j.robot.2013.08.002.

[17] H. Dang, P. Allen, Stable grasping under pose uncertainty using tactile feedback, Autonomous Robots (2013) 1–22doi:10.1007/s10514-013-9355-y.

[18] J. Felip, A. Morales, Robust sensor-based grasp primitive for a three- nger robot hand, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2009. doi:10.1109/IROS.2009.5354760.

[19] K. Hsiao, S. Chitta, M. Ciocarlie, E. Jones, Contact-reactive grasping of objects with partial shape information, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2010. doi:10.1109/IROS.2010.5649494.

[20] S. Dragiev, M. Toussaint, M. Gienger, Uncertainty aware grasping and tactile exploration, in: Robotics and Automation (ICRA), 2013 IEEE International Conference on, 2013, pp. 113–119. doi:10.1109/ICRA.2013.6630564.

[21] N. Sommer, M. Li, A. Billard, Bimanual compliant tactile exploration for grasping unknown objects, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2014.

[22] M. Bjorkman, Y. Bekiroglu, V. Hogman, D. Kragic, Enhancing visual perception of shape through tactile glances, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2013., 2013, pp. 3180–3186. doi:10.1109/IROS.2013.6696808.

[23] J.-P. Saut, S. Ivaldi, A. Sahbani, P. Bidaud, Grasping objects localized from uncertain point cloud data, Robotics and Autonomous Systems 62 (12) (2014) 1742 – 1754. doi:http://dx.doi.org/10.1016/j.robot.2014.07.011.

[24] M. Roa, R. Suarez, Computation of independent contact regions for grasping 3-d objects, Robotics, IEEE Transactions on 25 (4) (2009) 839–850. doi:10.1109/TRO.2009.2020351.

[25] V. N. Christopoulos, P. R. Schrater, Handling shape and contact location uncertainty in grasping two-dimensional planar objects., in: IROS, IEEE, 2007, pp. 1557–1563.

[26] T. Takahashi, T. Tsuboi, T. Kishida, Y. Kawanami, S. Shimizu, M. Iribe, T. Fukushima, M. Fujita, Adaptive grasping by multi ngered hand with tactile sensor based on robust force and

position control, in: Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, 2008, pp. 264–271. doi:10.1109/ROBOT.2008.4543219.

[27] A. T. Miller, S. Knoop, H. I. Christensen, P. K. Allen, Automatic grasp planning using shape primitives, in: In Proc. of the 2003 IEEE Intl. Conf. on Robotics and Automation, 2003, pp. 1824–1829.

[28] S. Ekvall, D. Kragic, Learning and evaluation of the approach vector for automatic grasp generation and planning, in: Robotics and Automation, 2007 IEEE International Conference on, 2007, pp. 4715–4720. doi:10.1109/ROBOT.2007.364205.

[29] K. Huebner, D. Kragic, Selection of robot pre-grasps using box-based shape approximation, in: Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on, 2008, pp. 1765–1770. doi:10.1109/IROS.2008.4650722.

[30] R. Pelossof, A. Miller, P. Allen, T. Jebara, An svm learning approach to robotic grasping, in: Robotics and Automation, 2004. Proceedings. ICRA 04. 2004 IEEE International Conference on, Vol. 4, 2004, pp. 3512–3518 Vol.4. doi:10.1109/ROBOT.2004.1308797.

[31] B. Huang, S. El-Khoury, M. Li, J. J. Bryson, A. Billard, Learning a real time grasping strategy., in: ICRA, IEEE, 2013, pp. 593–600.

[32] M. Kopicki, R. Detry, F. Schmidt, C. Borst, R. Stolkin, J. L. Wyatt, Learning dexterous grasps that generalise to novel objects by combining hand and contact models, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 2014, pp. 5358–5365.

[33] T. Yoshikawa, Multi ngered robot hands: Control for grasping and manipulation, Annual Reviews in Control 34 (2) (2010) 199 – 208. doi:http://dx.doi.org/10.1016/j.arcontrol.2010.09.001.

[34] Z. Li, P. Hsu, S. Sastry, Grasping and coordinated manipulation by a multi ngered robot hand., The International Journal of Robotics Research 8 (4) (1989) 33–50.

[35] T. Yoshikawa, X.-Z. Zheng, Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object, The International Journal of Robotics Research 12 (3) (1993) 219–230. arXiv:http://ijr.sagepub.com/content/12/3/219.full.pdf+html, doi:10.1177/027836499301200302.

[36] M. Buss, H. Hashimoto, J. Moore, Dextrous hand grasping force optimization, Robotics and Automation, IEEE Transactions on 12 (3) (1996) 406–418. doi:10.1109/70.499823.

[37] Y. Zheng, M. Lin, D. Manocha, On computing reliable optimal grasping forces, Robotics, IEEE Transactions on 28 (3) (2012) 619–633. doi:10.1109/TRO.2012.2183057.

[38] S. A. Schneider, R. H. Cannon, Object impedance control for cooperative manipulation: theory and experimental results, IEEE Transactions on Robotics and Automation 8 (3) (1992) 383–394.

[39] T. Wimbock, C. Ott, G. Hirzinger, Analysis and experimental evaluation of the intrinsically passive controller (IPC) for multi ngered hands, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2008.

[40] K. Tahara, K. Maruta, A. Kawamura, M. Yamamoto, Externally sensorless dynamic regrasping and manipulation by a triple ngered robotic hand with torsional ngertip joints, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2012.

[41] M. Li, H. Yin, K. Tahara, A. Billard, Learning object-level impedance control for robust grasping and dexterous manipulation, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2014.

[42] J. Romano, K. Hsiao, G. Niemeyer, S. Chitta, K. Kuchenbecker, Human-inspired robotic grasp control with tactile sensing, Robotics, IEEE Transactions on 27 (6) (2011) 1067–1079. doi:10.1109/TRO.2011.2162271.

[43] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics: Modelling, Planning and Control, 1st Edition, Springer Publishing Company, Incorporated, 2008.

[44] Y. Zheng, W.-H. Qian, Coping with the grasping uncertainties in force-closure analysis., International Joournal of Robotic Research 24 (4) (2005) 311–327.

[45] J. Kim, K. Iwamoto, J. Kuffner, Y. Ota, N. Pollard, Physically based grasp quality evaluation under pose uncertainty, IEEE Transactions on Robotics 29 (6) (2013) 1424–1439. doi:10.1109/TRO.2013.2273846.

[46] R. C. Brost, Planning robot grasping motions in the presence of uncertainty, Tech. Rep. CMU-RI-TR-85-12, Robotics Institute, Pittsburgh, PA (July 1985).

[47] J. Laaksonen, E. Nikandrova, V. Kyrki, Probabilistic sensor-based grasping, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), 2012.

[48] J. Bohg, M. Johnson-Roberson, B. Leon, J. Felip, X. Gratal, N. Bergstrom, D. Kragic, A. Morales, Mind the gap - robotic grasping under incomplete observation, in: Robotics and Automation (ICRA), 2011 IEEE International Conference on, 2011, pp. 686–693. doi:10.1109/ICRA.2011.5980354.

[49] I. Gori, U. Pattacini, V. Tikhanoff, G. Metta, Three- nger precision grasp on incomplete 3d point clouds, in: IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 2014.

[50] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2005.

[51] E. Solak, R. Murray-Smith, W. E. Leithead, D. Leith, C. E. Rasmussen, Derivative observations in gaussian process models of dynamic systems, in: S. Thrun, S. Becker, K. Obermayer (Eds.), Advances in Neural Information Processing Systems 15, MIT Press, Cambridge, MA, 2003, pp. 1033–1040.

[52] F. L. Bookstein, Principal warps: Thin-Plate splines and the decomposition of deformations, IEEE Trans. Pattern Anal. Mach. Intell. 11 (6) (1989) 567–585. doi:10.1109/34.24792.

[53] S. Dragiev, M. Toussaint, M. Gienger, Gaussian process implicit surfaces for shape estimation and grasping, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2011.

[54] B. Siciliano, O. Khatib, Springer handbook of robotics, Springer Science & Business Media, 2008.

[55] R. M. Murray, Z. Li, S. S. Sastry, S. S. Sastry, A mathematical introduction to robotic manipulation, CRC press, 1994.

[56] A. Wchter, L. T. Biegler, On the implementation of an interior-point lter line-search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (1) (2006) 25–57. doi:10.1007/s10107-004-0559-y.
URL http://dx.doi.org/10.1007/s10107-004-0559-y

[57] K. Tahara, S. Arimoto, M. Yoshida, Dynamic object manipulation using a virtual frame by a triple soft- ngered robotic hand, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2010.

[58] R. Diankov, Automated construction of robotic manipulation programs, Ph.D. thesis, Carnegie Mellon University, Robotics Institute (August 2010).
URL http://www.programmingvision.com/rosen_diankov_thesis.pdf

[59] G. Mclachlan, D. Peel, Finite Mixture Models, 1st Edition, Wiley Series in Probability and Statistics, Wiley-Interscience, 2000.

[60] M. Powell, A direct search optimization method that models the objective and constraint functions by linear interpolation, in:

Advances in optimization and numerical analysis,eds. S. Gomez and J-P. Hennart, Kluwer Academic (Dordrecht), 1994, pp. 51–67.

Miao Li has received the Bachelor and Master's degree both from the School of Mechanical Science and Engineering, Huazhong University of Science and Technology(HUST), China, in 2008 and 2011 respectively. From Sep. 2011, he joined the Department of Mechanical Engineering at École Polytechnique Fédérale de Lausanne (EPFL) in Switzerland, as a doctoral assistant. His current research interests mainly focus on robust robotic grasping, impedance control of dexterous manipulation using tactile sensing.

Kaiyu Hang is a Ph.D student at the Centre for Autonomous Systems and the Computer Vision and Active Perception Lab at KTH Royal Institute of Technology , Stockholm, Sweden. His current research interests include robotic precision/fingertip grasping, object-hand joint representation, adaptable grasping and grasp manifold learning etc. His interests include robotics, machine learning, AI and computer vision.

Danica Kragic is a Professor at the School of Computer Science and Communication at the Royal Institute of Technology, KTH. She received MSc in Mechanical Engineering from the Technical University of Rijeka, Croatia in 1995 and PhD in Computer Science from KTH in 2001. She has been a visiting researcher at Columbia University, Johns Hopkins University and INRIA Rennes. She is the Director of the Centre for Autonomous Systems. Danica received the 2007 IEEE Robotics and Automation Society Early Academic Career Award. She is a member of the Royal Swedish Academy of Sciences and Young Academy of Sweden. She holds a Honorary Doctorate from the Lappeenranta University of Technology. She chaired IEEE RAS Technical Committee on Computer and Robot Vision and served as an IEEE RAS AdCom member. Her research is in the area of robotics, computer vision and machine learning. In 2012, she received an ERC Starting Grant. Her research is supported by the EU, Knut and Alice Wallenberg Foundation, Swedish Foundation for Strategic Research and Swedish Research Council.

Aude Billard is Professor of Micro and Mechanical Engineering, and the head of the LASA Laboratory at the School of Engineering at the Swiss Federal Institute of Technology in Lausanne. She received a M.Sc. in Physics from EPFL (1995), a MSc. in Knowledge-based Systems (1996) and a Ph.D. in Artificial Intelligence (1998) from the University of Edinburgh. She was the recipient of the Intel Corporation Teaching award, the Swiss National Science Foundation career award in 2002, the Outstanding Young Person in Science and Innovation from the Swiss Chamber of Commerce and the IEEE-RAS Best Reviewer award in 2012. Aude Billard served as an elected member of the Administrative Committee of the IEEE Robotics and Automation society (RAS) for two terms (2006-2008 and 2009-2011) and is the chair of the IEEE-RAS Technical Committee on Humanoid Robotics. Her research interests focus on machine learning tools to support robot learning through human guidance. This extends also to research on complementary topics, including machine vision and its use in human-robot interaction and computational neuroscience to develop models of motor learning in humans.