Bimanual Compliant Tactile Exploration for Grasping Unknown Objects

Nicolas Sommer, Miao Li and Aude Billard
Ecole Polytechnique Federale de Lausanne (EPFL)
{n.sommer, miao.li, aude.billard}@epfl.ch

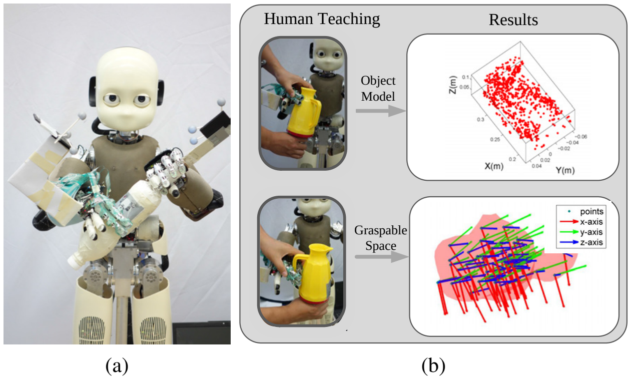
Abstract— **Humans have an incredible capacity to learn properties of objects by pure tactile exploration with their two hands. With robots moving into human-centred environment, tactile exploration becomes more and more important as vision may be occluded easily by obstacles or fail because of different illumination conditions. In this paper, we present our first results on bimanual compliant tactile exploration, with the goal to identify objects and grasp them. An exploration strategy is proposed to guide the motion of the two arms and fingers along the object. From this tactile exploration, a point cloud is obtained for each object. As the point cloud is intrinsically noisy and un-uniformly distributed, a filter based on Gaussian Processes is proposed to smooth the data. This data is used at runtime for object identification. Experiments on an iCub humanoid robot have been conducted to validate our approach.**

I. INTRODUCTION

Tactile exploration is of primary importance for visually impaired people. It is also crucial in non-visually impaired people, in particular when manipulating objects with two hands. In this case, the objects are often obstructed from view by either or both arms and fingers and one can rely only on tactile information to direct the manipulation. We explore the use of tactile information to reconstruct object's shape. We develop a method whereby the two arms move in coordination so as to maximize the surface on the object that can be explored. This is used in conjunction with an algorithm for object recognition.

Bimanual coordination and tactile exploration in robotics have been studied so-far separately. One of the first early attempts to exploit active tactile exploration with passive stereo vision for object recognition was proposed in [1]. There, a rigid finger-like tactile sensor is used to trace along the surface with predefined movement cycles and to provide a limited amount of object surface information. This work was later extended to develop different exploratory procedures (EP) to acquire and interpret 3-D touch information [2]. These EPs were, however, not linked to a fully autonomous system and a human experimenter has to specify the EPs for each exploration. Different strategies were proposed to explore the objects automatically by considering different objective functions for the choice of EP, such as reducing the maximal number of interpretations for the exploration data acquired so far [3] or reducing the uncertainty [4]. However, all of these exploration strategies only work for single-finger touch sensors and polyhedral objects.

More recently, some studies have been done in the research area of dexterous tactile exploration. In [5], a dynamic potential field approach is used as an exploration strategy to guide an anthropomorphic five-finger hand along the surface of previously unknown objects in simulation. In [6], two motion primitives, i.e., explorative and exploitative grasps, are introduced to make a trade-off between grasping known regions and exploring unseen regions. However, the robustness and applicability of these strategies have only been evaluated on simple shaped objects (a cylinder and a sphere) in simulation. Other related works, such as our previous work [7], show that tactile exploration using an anthropomorphic robot hand can successfully recognize several human-like faces. In [8], a probabilistic approach is proposed to represent datapoints, which are collected from predefined grasps using a Schunk hand with tactile sensors at high frame rates.



(a)
(b)

Fig. 1: **Left:** The iCub humanoid robot is exploring an object with its two hands. **Right:** The robot is taught the shape of the object by a human teacher passively guiding the robot's hand along the object, emphasizing the object's part that can be grasped, e.g. the handle on the jar. The object model is stored as a point cloud and the graspable part is modeled using GMM (Gaussian Mixture Model).

All the previous works are limited to the scope of static scenes, which means that the object to explore is fixed during the whole exploration. This assumption is quite restrictive as the robot may be not able to explore some of the unknown but interesting areas due to the limited workspace explorable by the robot's hands.

More complex tactile exploration has been studied in humans: to study dynamic exploration of objects, either using in-hand exploration [9] or dual-hand exploration [10]. None of these works has been used to drive the control of a robotic system. In this paper, we develop a strategy for bimanual compliant tactile exploration of unknown objects. The object is held by one hand while the other hand is exploring it, see Figure 1. The bimanual coordination strategy consists in moving the hand holding the object so as to bring the interesting region on the object into the reachable space of the other hand so as to make it easier for the other hand to either explore or grab the object.

Furthermore, because the data obtained from tactile sensing is intrinsically noisy and un-uniformly distributed (compared with point clouds obtained from vision), it is difficult to find the correct corresponding points in two point clouds, which greatly influences the object identification's accuracy. To deal with this, a filter based on Gaussian Processes (GP) is proposed to process the data. GP has been used before in tactile exploration to define the uncertainty [6]. In [8], the authors utilize a modified Kalman filter to build a probabilistic model of objects, which can greatly reduce the number of points that need to be stored. Their modified Kalman filter is very similar to the proposed GP-based filter but for different purposes. In this work, we aim at uniformly sampling from the object's surface. To this end, we assume that all the collected data can be generated from an implicit surface that can be approximated by GP [11], [12], from which a GP-based filter is designed to smooth the data.

In this work, the objective of tactile exploration is to identify the object and thus to find a suitable hand position and orientation to hold the objects with the exploration hand. An extensive work has been done on finding an optimal or suitable grasp on known, unknown or partially unknown objects [13]. The approaches, such as [14], are advantageous as they provide all possible grasps, whereas the approaches based on learning from demonstration require an expert and provide only a subset of feasible grasps, but with the advantage of being faster.

The rest of the paper is organized as follows: in the next section, we introduce our bimanual control framework and the local finger exploration strategy. In Section III, methods about data smoothing, object identification and grasp selection are presented. In Section IV, we present the experimental setup with our humanoid robot iCub. Further, we present our experimental results and discussion in Section V. Finally, we give a conclusion and an outlook of future work in Section VI.

II. Bimanual compliant tactile exploration

Our objective involves the identification of objects through tactile exploration. However, the workspace of humanoid robots is usually limited. Most daily objects are too large and cannot be explored by a single arm and hand. In order for the robot to gather enough information on the object's shape to allow unambiguous identification, the hand needs to explore a large portion of the object. To this end, we must extend the reachability of the exploring hand relatively to the object. In order to achieve this, we use one hand of the robot to hold an object, while the other hand explores it. This allows to approach and touch the object from different angles and with higher dexterity relatively to the workspace of both arms.

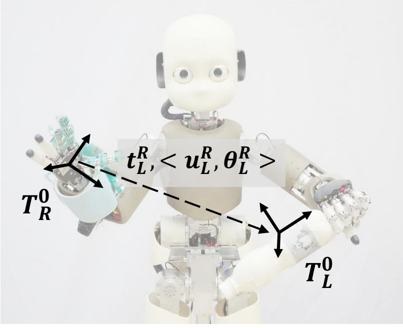


Fig. 2: Schematic of the bimanual constraint. The frames $\mathcal{T}_0^L$ and $\mathcal{T}_0^R$ of the interest points should coincide to satisfy the bimanual constraint. The frame $\mathcal{T}_0^L$ changes depending on where the object should be scanned and the estimation of the object's diameter.

A. Bimanual coordination

Let $\mathcal{T}_R^0$ and $\mathcal{T}_L^0$ be the homogeneous transformations from the robot root frame to the frames R and L attached respectively to the right and left arm's "interest points". In the rest of the paper, we will refer to the "interest points" to denote a) the center of the palm on the exploring hand and b) the point to be reached on the object held by the other hand.

The goal is to have both frames coincide: $\mathcal{T}_R^0 = \mathcal{T}_L^0 \Leftrightarrow \mathcal{T}_R^L = \mathbb{I}$ in which $\mathbb{I}$ is the identity matrix (see Fig. 2).

1) Motion generation: We generate a kinematic constraint from the above static constraint in order for the system to converge to this state. We give the following translational and rotational velocities in Cartesian space until both frames coincide:

$$\mathbf{v_R} = \frac{\mathbf{t_L^R}}{\parallel \mathbf{t_L^R} \parallel} \cdot f(\parallel \mathbf{t_L^R} \parallel) \qquad (1)$$

and $\mathbf{v_L} = -\mathbf{v_R}$. With $\mathbf{v_x}$ being the translation velocity vector of the frame x, expressed in the robot root frame, $\mathbf{t_L^R}$ the translation vector from R to L, and f a function from $\mathbb{R}^+$ to $\mathbb{R}^+$ designed to give a smooth and converging motion. For the rotation, a similar constraint is expressed in the axis-angle notation which defines a rotation with an axis $\mathbf{u}$ and a rotation angle θ around this axis. Given the axis-angle rotation $< \mathbf{u_L^R}, \theta_L^R >$ equivalent to the usual rotation matrix notation R_L^R, the rotational velocity is defined in axis-angle notation: $\mathbf{w_R} =< \mathbf{u_L^R}, \omega >$ and $\mathbf{w_L} =< \mathbf{u_L^R}, -\omega >$ with $\omega = f(\parallel \theta_L^R \parallel)$. We chose to use the function $f(x) = a \cdot \exp(-\frac{w}{x})$, with parameters a and w determining respectively the velocity of the motion far from the target point and a measure of the closeness to the target.

2) Working at the limit of the workspace: Because the workspace of the robot's arms are often limited during bimanual manipulation (e.g. the hands of the iCub robot can barely reach each other), it is important to take into account the non-feasibility of a given inverse kinematics problem. Our IK solver uses a pseudo-inverse of the Jacobian with optimization, this way we can weight some constraints so as to satisfy them in priority. In our application, the position constraints are more important than the orientation constraints, and the orientation of the normal of the palm (the right hand's interest point) is more important than the orientation of the other axes of the interest point's frame. Therefore, we express the orientation Jacobian in the interest point's frame and the desired axes are weighted as indicated previously. These weights are only taken into account when the IK problem has no solution and a compromise between the constraints has to be found, therefore their choice is not very sensitive and they are set empirically.

3) Collision avoidance: During the scanning of objects, the goal is to keep one hand and fingers in contact with the scanned object at all times. However, when both arms are changing configuration to start a new "scan" from a different angle, there is a need for collision avoidance in order for the scanning hand not to hit the object while moving around it. Because we do not know the exact shape of the object held by one of the hands, we assume a cylinder with a sufficiently large diameter, and require the end-effector to move outside this cylinder during the motion, until the hand is aligned in front of the target reaching point, where it is allowed to enter the "collision area" (see Fig. 3). The corrected velocities for collision avoidance are given in Table. I.

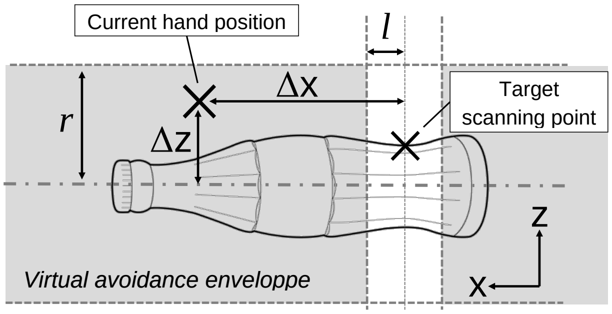


Fig. 3: Scheme of an object and its collision avoidance virtual envelope. A lateral safety zone is delimited by the parameter *l*, in which the exploring hand can enter. The diameter of the virtual envelope is defined by *r*.

	$\|\Delta x\| \leq l$	$\|\Delta x\| > l$
$\|\Delta z\| > r$	$\mathbf{v_L^{R}}' = \mathbf{v_L^R}$	$v_{L\,z}^{R'} = v_{L\,z}^{R} \cdot \exp(\frac{-w_{out}}{\|\Delta z - r\|})$
$\|\Delta z\| \leq r$		$v_{L\,z}^{R'} = -v_{av} \cdot \exp(\frac{-w_{in}}{\|\Delta z - r\|})$

TABLE I: Velocity correction for collision avoidance, with v_{av} a predefined avoidance speed, w_{out} and w_{in} parameters that regulate the transition from collision avoidance to the normal behaviour of reaching the starting scanning point, $\mathbf{v_L^{R}}'$ the modified relative velocity between the right and left frames to avoid collision, and $v_{L\,z}^{R}$ the z component of the velocity, expressed in a coordinate system rotating around the object. *Z* is constrained to be normal to the principal axis of the object (i.e. the scanning direction *x*, see Fig. 3) and oriented around the *x* axis to point from the center of the object towards the other hand.

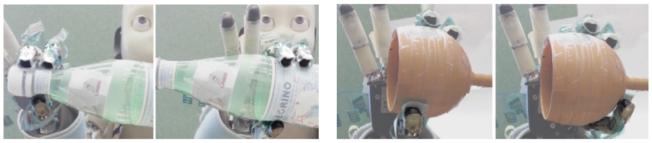
B. Finger exploration strategy

1) Compliant tactile control: During the exploration of the object, the tactile sensors provide contact information. In order to obtain this information, the fingers must apply enough pressure on the object. The tactile response is thus used in a pressure loop designed to apply sufficient force and obtain contact data, while not pressing too hard so as not to damage the object being touched (see Fig. 4a). For n_s tactile sensor patches and n_a actuators, the motors are commanded in current with $\mathbf{u} \in \mathbb{R}^{n_a}$ following:

$$\mathbf{u} = \kappa \cdot \Phi(S, S^*) \tag{2}$$

with $S, S^* \in \mathbb{R}^{n_s}$ respectively the current and desired tactile response, $\kappa \in \mathbb{R}^{n_a}$ a vector of proportional gains for each actuator, $\Phi : \mathbb{R}^{n_s} \rightarrow \mathbb{R}^{n_a}$ a mapping between the tactile sensor patches and corresponding motors. The mapping Φ depends on the architecture of the robot hand – the number of actuated joints, the number and disposition of the sensors – and the desired behaviour[1].

2) Thumb motion: On anthropomorphic robotic hands, the thumb is usually equipped with an additional degree of freedom which enables it to control its opposition to the other fingers. During the scanning of objects, we use this DoF to increase the amount of the object's surface explored by the thumb, especially for reaching areas otherwise difficult to access (see Fig. 4b). A periodic swiping motion is implemented and efficient enough to gather data more efficiently.



(a) Finger's compliance during scanning

(b) Thumb opposition

Fig. 4: **Left:** The fingers adapt to the size of the object in order to follow compliantly the surface. **Right:** Illustration of the advantage of changing the thumb's opposition while scanning a glass: the thumb follows the high curvature of the surface.

3) Detect loss of contact with the object: While scanning, the fingers might slide off the object (for instance when reaching an extremity). In that case, they might touch each other and record the contact as if they were touching the object. When this happens, the distance between the contacts points on the two fingers is close to 0 and this allows us to detect these events and to discard these contact points. This is also used to detect that the exploration has reached the end of the object and decide that the object can be scanned from another orientation.

4) Approaching the object: When the exploring hand comes in contact with the object, we need to detect precisely when the hand touches the object. Tactile sensors seem a good way to detect this contact. However, they should be extremely sensitive and detect very light pressure. Otherwise, when the exploring hand comes into contact with the object for the first time, it may apply too much force on the object – a small force on the object creates a high torque on the hand holding the object. While exploring, we overcome this problem by "pinching" the object so as to apply forces on

[1] In our implementation, each finger is controlled the same way, $n_s = 3$ for each finger as we take directly the average value for each tactile patch as inputs (one per phalanx, each composed of 12 or 16 taxels), and $n_a = 2$: the first actuator of the finger controls the first phalanx and the second actuator controls the second and third phalanx coupled together. For each finger, Φ is defined as follows: $\Phi(s) = \{min(e_0, e_1), e_2\}$, with s the average tactile response for each of the three phalanx (s_0, s_1 and s_2 are the average pressures on respectively the first, second and last phalanx and $e_i = s_i^* - s_i$, with s_i^* the corresponding desired pressures).

both sides of the object. Since our tactile sensors are not sensitive enough, we use force-torque sensors embedded in the robot's arm to detect when the hand touches the object. We use a first order band-pass filter to remove both the low-frequency component of the signal due to the errors of estimation of the robot's limb's own weight and smooth the high-frequency component since the signal is very noisy.

III. Object Identification and Grasping

In order to identify an object, the data collected from tactile exploration is first filtered and smoothed using a GP-based filter. The data can then be aligned with previously known object models and the average distance after alignment is used as criterion for identification. After the identification, a grasp is computed from previously learned grasps.

A. Object identification

1) Data filtering and smoothing: As the data acquired from tactile exploration is noisy and un-uniformly distributed, which is not suitable for object identification, a GP-based filter is proposed in this section to smooth the data. The basic idea is that only points that decrease uncertainty are stored in the training dataset.

Because the tactile sensors can also provide normal information for the points, a GP with derivative observations is adopted here to compute the uncertainty [15] (it requires the derivative of the covariance function with respect to the input). With the same inputs, the GP with derivative observations can provide the predictions on the derivative (the normal information). Given a set of n_t training input points $\mathbf{x} = \{\mathbf{x}^i \in \mathbb{R}^3\}^{i=1..n_t}$, the outputs are augmented with normal information $\hat{\mathbf{y}} = \{\mathbf{y}^i, \boldsymbol{\omega}^i\}^{i=1..n_t}$, where $\boldsymbol{\omega}^i \in \mathbb{R}^3$ is the normal direction of the i-th point. $\mathbf{y}^i$ equals to -1, 0 and 1 when the points are respectively inside, on and outside the estimated implicit surface, on which the collected tactile points are assumed located [11]. In practice, only one inside point is required, which is chosen as the center of all the collected points on the surface. For the outside points, 20 points are randomly sampled from a sphere with its diameter 20% larger than the distance between the center and the farthest collected point on the surface.

In order to compute the prediction, the following identities are necessary to construct the full covariance matrix $K \in \mathbb{R}^{4n_t \times 4n_t}$, which can be computed from the kernel function and its derivatives [15].

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

$$cov(\boldsymbol{\omega}^i_m, \mathbf{y}^j) = \frac{\partial}{\partial x_m} cov(\mathbf{y}^i, \mathbf{y}^j) \tag{4}$$

$$cov(\boldsymbol{\omega}^i_m, \boldsymbol{\omega}^j_n) = \frac{\partial^2}{\partial x_m \partial x_n} cov(\mathbf{y}^i, \mathbf{y}^j) \tag{5}$$

$$m = 1, 2, 3, n = 1, 2, 3;$$

In this work we use a kernel function k that has been derived from the *thin plate spline* in [16], which is widely used in shape analysis and reconstruction [17].

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

Where $\Omega \in \mathbb{R}$ is the parameter that can be readily computed from the training dataset. $\Omega = \max\{\|\mathbf{x}^i - \mathbf{x}^j\|, i = 1..n_t, j = 1..n_t\}$. In other words, Ω is the maximal distance in the training inputs.

With the full covariance matrix, for a new tactile point $\mathbf{x}_*$, we can determine whether the point is on the surface of the object by computing the predicted function value $E(\mathbf{y}_*)$ and its normal direction $E(\boldsymbol{\omega}_*)$ as follows:

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

The points $\mathbf{x}_*$ is said to be on the object surface if $E(\mathbf{y}_*) = 0$. In our GP-based filter, our main novelty is to use the covariance $cov(\hat{\mathbf{y}}_*) \in \mathbb{R}^{4\times 4}$ at the testing points to filter and smooth our collected datapoints. In the GP prediction, the covariance implies the uncertainty about its prediction:

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

where $K_* = K(\mathbf{x}_*, \mathbf{x})$ denotes the $4 \times 4n_t$ vector of covariances evaluated at all pairs of the testing and training inputs, and similarly for $K(\mathbf{x}, \mathbf{x})$ and $K(\mathbf{x}_*, \mathbf{x}_*)$. The parameter σ^2 reflects the variance of noise in the output. This new datapoint $\mathbf{x}_*$ is stored in the training dataset if the uncertainty about the predicted function value is above a given threshold, i.e., $cov(\mathbf{y}_*) > V_{thresh}$, where $cov(\mathbf{y}_*)$ is the first entry of $cov(\hat{\mathbf{y}}_*)$. In this paper, the threshold V_{thresh} is set to the same level as the noise's variance. As the normal direction is also available, a new datapoint is included if the predicted normal direction $E(\boldsymbol{\omega}_*)$ using equation (7) is very different from the measured one $\boldsymbol{\omega}_*$, i.e., $\boldsymbol{E}(\omega_*)^T \boldsymbol{\omega}_* < \theta_{thresh}$[2]. These datapoints usually mean that there is a large change in the curvature of the object surface. Furthermore, as GP is $\mathcal{O}(N^3)$ in computation complexity, the number of training datapoints should be limited. Therefore, the datapoint with the smallest uncertainty in the training dataset is replaced when the size of training dataset comes to its limitation S_{lim}[3], see Algorithm 1. In Fig. 5, we give an example using data generated from an object point cloud. We can see that the filtered points are more uniformly distributed and also that more datapoints are kept in the region of the jug handle where the curvature changes importantly.

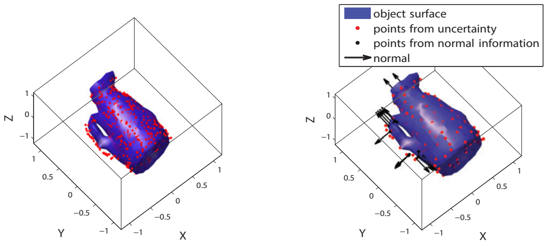
2) Object identification: For each of the objects to explore, we assume that there is already a point cloud model for it, which can be obtained either from a vision scanner or from human demonstrations. In Section IV, we give details on how we collect this point cloud model from human demonstrations. Herein, for the i-th object, the point cloud model is denoted as $\mathcal{O}^i = \{\mathbf{p}^{i,j}\}^{j=1..n_p}$. The object's identification algorithm tries to align the datapoints collected so far with the available object point clouds $\mathcal{O}^i$ and the one with the smallest alignment error is identified as the corresponding object. To this end, after each scanning, the

[2] The value of $\boldsymbol{E}(\omega_*)^T \boldsymbol{\omega}_*$ can vary between -1 and 1. As the normal information is usually not as accurate as position information, we set $\theta_{thresh} = cos(\pi/4) \simeq 0.7071$, which corresponds to a change in curvature of 45 degrees.

[3] The limit of the training dataset is chosen by considering the computation time and model accuracy. In this paper, we set $S_{lim} = 120$.

Algorithm 1: GP-based filter

```
Data: New collected data $\mathcal{X}_* = \mathbf{x}_*, \mathbf{y}_*, \boldsymbol{\omega}_*$
Result: The training dataset $\mathcal{X} = \{\mathbf{x}^i, \mathbf{y}^i, \boldsymbol{\omega}^i\}^{i=1..n_t}$
Compute Uncertainty: $cov(\mathbf{y}_*)$, see equation (8)
if $cov(\mathbf{y}_*) > V_{thresh}$ or $\boldsymbol{E}(\omega_*)^T \boldsymbol{\omega}_* < \theta_{thresh}$, then
    $\mathcal{X} = \{\mathcal{X}, \mathcal{X}_*\}, \quad n_t = n_t + 1;$
else
    Keep the same $\mathcal{X}$
Size Update:
if $n_t > S_{lim}$ then
    Compute the uncertainties for all the datapoints in the
    training dataset and delete the datapoint with the smallest
    uncertainty
```



(a) original data (b) filtered data

Fig. 5: An example to illustrate the performance of the GP-based filter. **Left:** 500 datapoints are randomly sampled from the point cloud of a jug, obtained from a 3D scanner. The surface is reconstructed with the method in [12]. **Right:** With the filter, only 120 datapoints are selected, either according to their prediction uncertainty (red dot) or from their normal information (black dot and arrow).

points gathered so far $\mathcal{X} = \mathbf{x}^j, j = 1...n_x$ are transformed into the most similar pose using the *iterative closest point* (ICP) algorithm.

As described in [18], ICP can compute the optimal transformation $(R, \mathbf{q}_t)$ between two corresponding datasets that minimizes the following distance error:

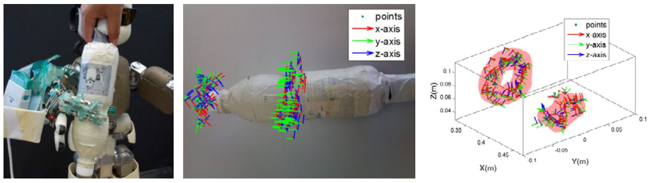
$$Dist(\mathcal{X}, \mathcal{O}^i) = \frac{1}{n_x} \sum_{j=1}^{n_x} \|\mathbf{p}^{i,j} - (R\mathbf{x}^j + \mathbf{q}_t)\|^2 \tag{9}$$

In our work, the correspondence between the measured points and the object point cloud models are chosen with the nearest neighbour match without replacement[4]. In general, this method suffers from local minima. To counter this effect, we run 10 different comparisons with 10 different initializations of the initial points. These initial points are uniformly obtained from different rotations R around the object's principal axis and its normalized translation components are randomly sampled in $[-0.5, 0.5]$.

For each available object model, we compute the minimal distance after alignment, i.e. $Dist(\mathcal{X}, \mathcal{O}^i)$ and the object is identified as the object with the minimal distance.

B. Grasp selection

For each object, we teach a set of suitable grasps by human kinaesthetic teaching, see Fig. 6. We place the object in the iCub's hand and record the position of the fingers relative to the objects as suitable grasps. These grasps are then modelled with Gaussian Mixture Model (GMM), from which one grasp is selected in real time after object identification. This method has been proven to be fast and useful for some real time tasks, such as catching an object in flight [19].



(a) (b) (c)

Fig. 6: (**a**) The feasible grasps for a bottle (around the cap and the neck) are taught by human demonstration; (**b**) The grasps are stored in the object's frame, only the relative hand position and orientation are used; (**c**) These grasps are trained and encoded by GMM, here is shown the trained result in the subspace of hand positions. The red surface corresponds to an iso-surface of the likelihood function, defining the limit of the graspable space.

1) Graspable space modelling: In human kinaesthetic teaching, the position $\mathbf{h} \in \mathbb{R}^3$ and orientation $\mathbf{o} \in \mathbb{R}^6$ (we use the first two columns of the orientation matrix) of the robot hand are recorded in the object frame. With these data, the graspable space can be modeled using GMM as follows:

$$p(\mathbf{h}, \mathbf{o}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{h}, \mathbf{o}, |\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{10}$$

where K is the number of Gaussian components, π_k is the prior of the k-th Gaussian component and $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is the Gaussian distribution with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$. The graspable space is the constitution of all the feasible grasps, i.e., $\mathcal{G} = \{(\mathbf{h}, \mathbf{o}) | p(\mathbf{h}, \mathbf{o}|\Omega) > L_{thresh}\}$. L_{thresh} is chosen such that the likelihood of 99% of the training data is above this value. For more details about learning graspable space using GMM, one can refer to [20]. In Fig. 6, we show the learned graspable space for a bottle used in our experiment.

Note: For the graspable space, we are only interested in the hand position and orientation that can possibly lead to a feasible grasp when the fingers are closed. The stability conditions, such as force closure, are not necessarily ensured. But one can also model the graspable space using grasps that fulfill force closure, from which one can infer grasps that are stable [20]. Note also that the method we propose in this paper could also include the finger joints and tactile response [21]. In principle, this increases the complexity of the model and more training data are required.

2) Grasp selection: To determine the grasp that is closest to the current position of the scanning hand, we find the minimum of $Q(\mathbf{h}^*, \mathbf{o}_1^*, \mathbf{o}_2^*)$:

$$Q(\mathbf{h}^*, \mathbf{o}_1^*, \mathbf{o}_2^*) = \alpha_1 \|\mathbf{h}^* - \mathbf{h}\|^2 + \alpha_2[(\mathbf{o}_1^* \mathbf{o}_1 - 1)^2 + (\mathbf{o}_2^* \mathbf{o}_2 - 1)^2] \tag{11}$$

$\mathbf{h}, \mathbf{o}_1, \mathbf{o}_2$ are the current hand position, the first and second column of the hand orientation respectively. α_1, α_2 are the penalty parameters for the position and orientation. $(\mathbf{h}^*, \mathbf{o}_1^*, \mathbf{o}_2^*)$ are the variables and correspond to the final

[4] The same point in the object point cloud model cannot be the correspondence point for two different measured points.

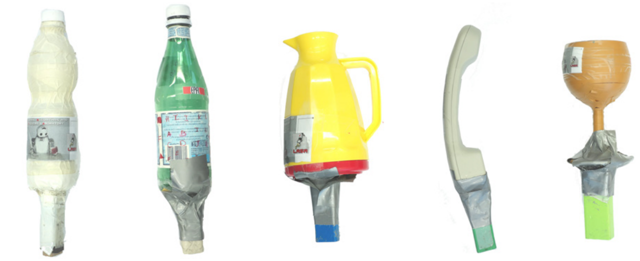
position and orientation at the grasping point. They must lie in the graspable domain that is determined by the likelihood threshold of the GMM. Determining the closest feasible grasp can be formulated as a constrained based optimization problem of the form:

$$\begin{aligned} \text{minimize:} \quad & Q(\mathbf{h}^*, \mathbf{o}_1^*, \mathbf{o}_2^*) \\ \text{subject to:} \quad & (\mathbf{h}^*, \mathbf{o}_1^*, \mathbf{o}_2^*) \in \mathcal{G} \end{aligned} \tag{12}$$

To solve the problem efficiently, a gradient descent method is used here. The initial points are chosen as the centers of the K Gaussian functions (10). Once the grasp is selected, we use our bimanual coordination algorithm with obstacle avoidance to move the hand to the graspable posture.

IV. Experiment

The iCub humanoid robot (Fig. 8a) is used to explore different everyday objects using both arms. We chose five objects: 2 bottles, 1 jar, 1 phone receiver and 1 glass, shown in Fig. 7. The two bottles are very similar and can test the accuracy of the identification method. The phone's profile encompass sharp changes in curvature, a challenge for the compliant control of the fingers. Scanning the glass is even more challenging as it requires to control precisely for the thumb's motion in order to follow the edges. The jar has a much larger diameter and involves two particular features: the handle and the spout. During the exploration, one arm holds the object, while the other arm explores it with its fingers. The collected data are compared with data previously collected manually in order to identify the object. During the exploration, the robot attempts to identify the objects as well as their positions and orientations. Then, from the previously learned grasps, one grasp is selected and adopted by the free hand, on the object (see Alg. 2).

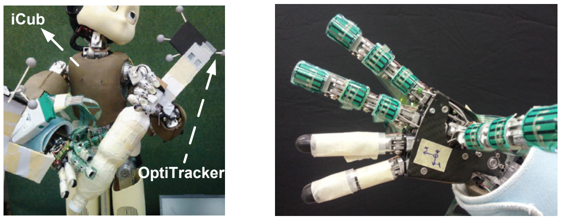


(a) bottle 1 (b) bottle 2 (c) jar (d) phone (e) glass

Fig. 7: Five different everyday objects are used in our experiment. Handles are mounted on the bottom of the objects in order to adapt to the size of iCub hand.

A. Setup

We use both arms of iCub, each of which have 7 degrees of freedom (DoFs). Each hand has 9 DoFs, 3 for the thumb, 2 for the index finger, 2 for the middle finger, 1 for the coupled ring and little finger and 1 for the adduction/abduction. Only the thumb, index and middle finger are equipped with *Tekscan*[5] tactile sensors (see Fig. 8b). The Tekscan sensors have a spatial resolution of $4mm$ ($6.2\ sensors/cm^2$), the fingers are equipped with $3 * 4$ taxels – tactile pixels – per phalanx, and $4 * 4$ taxels on their fingertip, which makes a total of 120 taxels on the hand. A motion capture system – *OptiTrack*[6] – is used to track the position and orientation of both hands to compensate for the inaccuracy of iCub's kinematics and obtain precise measurements. The contact positions are obtained through forward kinematics starting from the motion tracker, and given the geometry of the tactile sensors.



(a) iCub (b) iCub's hand equipped with Tekscan tactile sensors

Fig. 8: An iCub humanoid robot (**a**) is used in our experiment. The thumb, index and middle finger of the right hand are equipped with Tekscan sensors (**b**). During the experiment, the objects are firmly held by the left hand (no relative motion), while the right hand explores the object from different orientations.

B. Manual data collection

Prior to the exploration, we manually collect data from the objects using the same setup with the difference that the object is held by a human demonstrator in place of the robot itself. An optical tracker is attached to the object while the fingers of iCub are pressed against the object to collect point cloud data all over the surface (see the top left image on Fig. 1b). The acquired object point clouds are shown in Fig. 9.

C. Exploratory procedure

For the exploration process, we only assume that the principal axis of the object is available, for instance through basic image processing. However, we know the precise position of the hand holding the object through our motion capture system – instead of using forward kinematics, imprecise because of slack in the joints. This system is also used to track the position of the right hand.

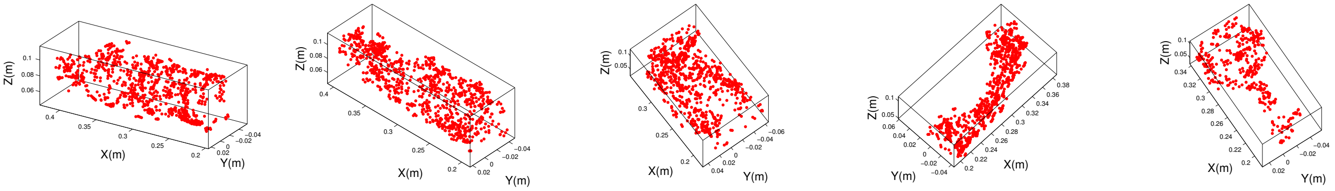
The right hand scans the object from one end to another along this principal axis, and changes the angle of approach iteratively around this principal axis at every scan. The procedure is described in Algorithm 2. During the whole exploration, both arms move simultaneously to achieve the desired relative position and orientation between the interest points (the palm and a point on the object's surface), therefore the indications in Alg. 2 are given in relative terms between these two interest points.

V. Results, discussion

a) Exploration Results: Each object is scanned using Algorithm 2. The acquired point clouds are quite noisy and non-uniformly distributed. As mentioned above, the raw

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

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



(a) bottle 1 (b) bottle 2 (c) jar (d) phone (e) glass

Fig. 9: The object point clouds obtained from human teaching.

```
Algorithm 2: Exploratory procedure
$\theta \leftarrow \theta_{min}$; \\ Angle of approach around the principal axis
while $\theta < \theta_{max}$ do
    Go above initial scanning point;
    while !contact do
        Move hand and object towards each other;
    Close fingers and activate finger compliant control;
    while fingers in contact with object do
        Slide the hand along the object's principal axis;
    Open fingers;
    Try to identify object;
    if object identified then
        Compute and reach a grasping posture;
        Grasp the object;
        return 1;
    $\theta \leftarrow \theta + increment$;
return 0;
```

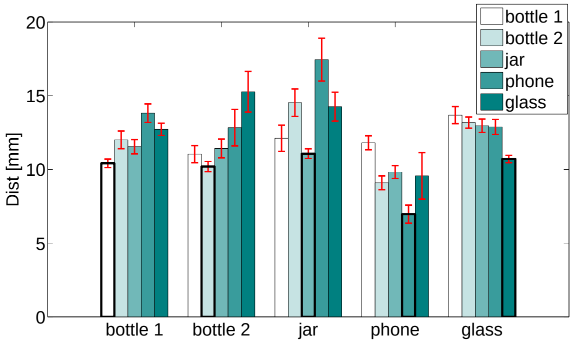


Fig. 12: Comparison of the aligned distance from Eq. (9) for all the objects. The object with the smallest distance is chosen as the identification result. For each object, we ran the identification algorithm 10 times.

point clouds are not suitable for object identification, due to the difficulty in finding the correct corresponding points for the ICP algorithm. With the GP filter in Algorithm 1, the filtered point clouds become smoother, sparser and less noisy, as shown in Fig. 11.

b) Object Identification Results: For each explored object, we chose 10 different initial configurations for the ICP algorithm, where the rotation R in (9) is uniformly sampled around the principle axis of the object. The object is identified as the object with the smallest distance among the 10 different trials. The distance for each trial is shown in Fig. 10 and the points after alignment are shown in Fig. 11. We repeated the identification algorithm 10 times for each object and the success rate of identification is always above 90%. The failure happens when bottle 2 is misidentified as bottle 1 and the jar is misidentified as bottle 1. The statistics for the distances are shown in Fig. 12.

c) Grasp Selection: After the object is identified, a grasp is chosen and the right hand moves to the selected grasp (section III-B.2), as shown in Fig. 13. Note that as the object is firmly fixed on the left hand for experimental purposes, we move the right hand to the chosen grasp without switching the hand holding the object.

d) Discussion: The raw point cloud acquired with the tactile sensors is very noisy and would make the identification of the object very difficult without filtering. In our experiments, we noticed that without the GP filter, it is very hard to identify the *jar* and *phone* from raw data as their surfaces are not as "smooth" as the surfaces of the other objects. Also the selection of the grasp is done according to a general distance criterion (Eq. 11). Other criteria, such as manipulability and grasp stability can also be applied here. Furthermore, in the current work, we did not include the finger joints and tactile response in our graspable space model, which is part of the possible improvements.

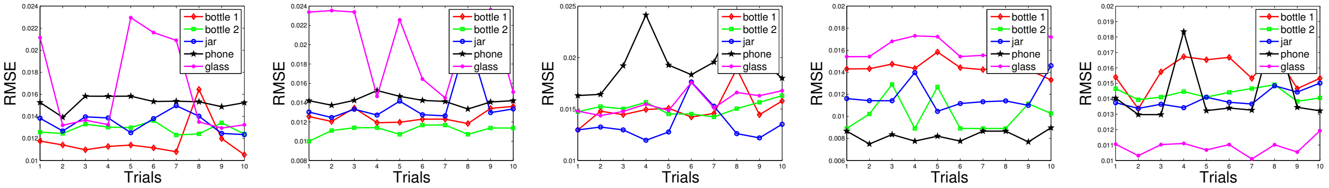
VI. Conclusion

We have presented a general approach for bimanual compliant tactile exploration, with applications to object identification, manipulation and grasping. The kinematic limitations of the system, i.e., workspace limitation and collisions, are considered in this exploration strategy, which is critical in tactile exploration as suggested in [5]. Also, due to the intrinsically noisy and un-uniformly distributed characteristics of tactile datapoints, a GP-based filter is proposed to smooth the data, which can then be more easily used for object identification. In order to grasp the object after identification, we teach grasps for each object by human demonstration and model the graspable space with GMM. A general distance metric is adopted to choose a final feasible grasp given the current hand configuration.

For our future work, we are interested in learning the exploration strategy from human demonstration, where one may often switch the hand holding the object during the exploration. This opens a challenging research perspective for both tactile exploration and robotic grasping.

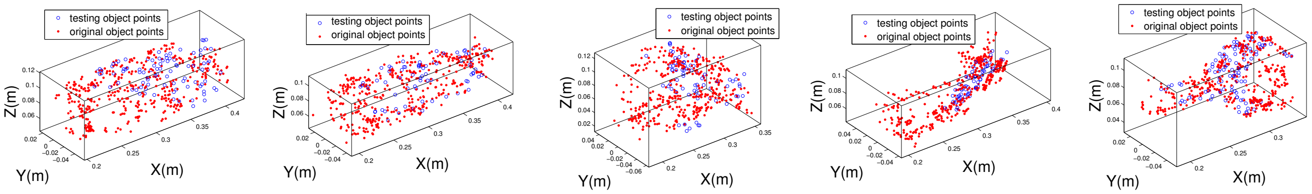
Acknowledgment

This research was supported by the Swiss National Science Foundation through the National Centre of Competence in



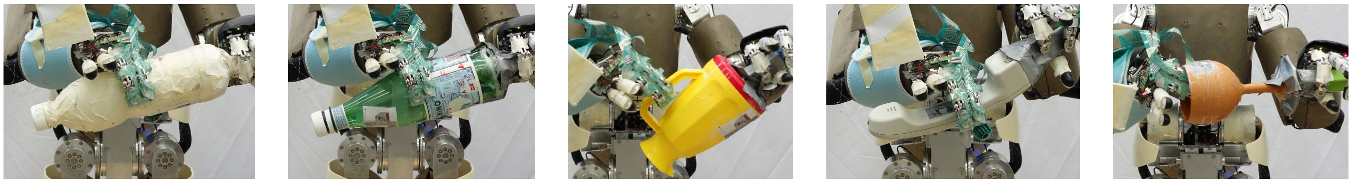
(a) bottle 1 (b) bottle 2 (c) jar (d) phone (e) glass

Fig. 10: Object identification with sparse point cloud starting from 10 different initial configurations.



(a) bottle 1 (b) bottle 2 (c) jar (d) phone (e) glass

Fig. 11: The filtered object point clouds aligned with the trained object points cloud. Only 400 datapoints from the trained object point clouds are displayed.



(a) bottle 1 (b) bottle 2 (c) jar (d) phone (e) glass

Fig. 13: The selected grasp (hand position and orientation) for each explored object after identification.

Research Robotics. The research leading to these results has also received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement n^o 288533 ROBOHOW.COG.

References

[1] P. Allen and R. Bajcsy, "Object recognition using vision and touch," in *Proceedings of the 9th international joint conference on Artificial intelligence (IJCAI), 1985.*

[2] P. Allen and P. Michelman, "Acquisition and interpretation of 3-d sensor data from touch," *IEEE Transactions on Robotics and Automation*, vol. 6, pp. 397–404, Aug. 1990.

[3] K. Roberts, "Robot active touch exploration: constraints and strategies," in *Proceedings of International Conference on Robotics and Automation (ICRA), 1990.*

[4] S. Caselli, C. Magnanini, F. Zanichelli, and E. Caraffi, "Efficient exploration and recognition of convex objects based on haptic perception," in *Proceedings of International Conference on Robotics and Automation (ICRA), 1996.*

[5] A. Bierbaum, M. Rambow, T. Asfour, and R. Dillmann, "A potential field approach to dexterous tactile exploration of unknown objects," in *International Conference on Humanoid Robots(Humanoids), 2008.*

[6] S. Dragiev, M. Toussaint, and M. Gienger, "Uncertainty aware grasping and tactile exploration," in *Proceedings of International Conference on Robotics and Automation (ICRA), 2013.*

[7] N. Sommer and A. Billard, "Face classification using touch with a humanoid robot hand," in *International Conference on Humanoid Robots (Humanoids), 2012*.

[8] M. Meier, M. Schopfer, R. Haschke, and H. Ritter, "A probabilistic approach to tactile shape reconstruction," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 630–635, 2011.

[9] D. Faria, R. Martins, J. Lobo, and J. Dias, "Probabilistic representation of 3d object shape by in-hand exploration," in *International Conference on Intelligent Robots and Systems (IROS), 2010.*

[10] K. Charusta, D. Dimitrov, A. J. Lilienthal, and B. Iliev, "Extraction of grasp related features by human dual-hand object exploration," in *Proceedings of the IEEE International Conference on Advanced Robotics (ICAR), 2009*.

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

[12] S. El-Khoury, M. Li, and A. Billard, "On the generation of a variety of grasps," *Robotics and Autonomous Systems*, 2013. (In Press).

[13] A. Sahbani, S. El-Khoury, and P. Bidaud, "An overview of 3d object grasp synthesis algorithms.," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 326–336, 2012.

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

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

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

[17] F. L. Bookstein, "Principal warps: Thin-Plate splines and the decomposition of deformations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, pp. 567–585, June 1989.

[18] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 239–256, Feb. 1992.

[19] S. Kim, A. Shukla, and A. Billard, "Catching objects in flight.," *IEEE Transactions on Robotics*. (submitted).

[20] B. Huang, S. El-Khoury, L. M., J. J. Bryson, and A. Billard, "Learning a real time grasping strategy," in *Proceedings of International Conference on Robotics and Automation (ICRA), 2013.*

[21] E. L. Sauser, B. D. Argall, G. Metta, and A. G. Billard, "Iterative learning of grasp adaptation through human corrections," *Robotics and Autonomous Systems*, vol. 60, pp. 55–71, Jan. 2012.