Tactile-based active object discrimination and target object search in an unknown workspace

Mohsen Kaboli, Kunpeng Yao, Di Feng & Gordon Cheng

Autonomous Robots

ISSN 0929-5593

Auton Robot
DOI 10.1007/s10514-018-9707-8

Springer

Autonomous Robots
https://doi.org/10.1007/s10514-018-9707-8

Tactile-based active object discrimination and target object search in an unknown workspace

Mohsen Kaboli[1] · **Kunpeng Yao**[1] · **Di Feng**[1] · **Gordon Cheng**[1]

Received: 31 December 2016 / Accepted: 29 January 2018

Abstract
The tasks of exploring unknown workspaces and recognizing objects based on their physical properties are challenging for autonomous robots. In this paper, we present strategies solely based on tactile information to enable robots to accomplish such tasks. (1) An active exploration approach for the robot to explore unknown workspaces; (2) an active touch objects learning method that enables the robot to learn efficiently about unknown objects via their physical properties (stiffness, surface texture, and center of mass); and (3) an active object recognition strategy, based on the knowledge the robot has acquired. Furthermore, we propose a tactile-based approach for estimating the center of mass of rigid objects. Following the active touch for workspace exploration, the robotic system with the sense of touch in fingertips reduces the uncertainty of the workspace up to 65 and 70% compared respectively to uniform and random strategies, for a fixed number of samples. By means of the active touch learning method, the robot achieved 20 and 15% higher learning accuracy for the same number of training samples compared to uniform strategy and random strategy, respectively. Taking advantage of the prior knowledge obtained during the active touch learning, the robot took up to 15% fewer decision steps compared to the random method to achieve the same discrimination accuracy in active object discrimination task.

Keywords Active tactile object localization · Active tactile object exploration · Active tactile learning

1 Introduction

1.1 Motivation

Touch plays an important role in our daily lives, from perceiving the environment to identifying, learning about and interacting with objects. Compensating for a lack of touch through other senses is difficult. For robotic systems that interact in dynamic environments, recognizing objects via their physical properties (such as surface texture, stiffness, center of mass, and thermal conductivity) is crucial. However, this is difficult to achieve even with advanced vision techniques, which are often marred by the occlusion, poor lighting situations, and a lack of precision.

As an alternative, tactile sensing can provide rich and direct feedback to the robotic systems from abundant simultaneous contact points (Robles-De-La-Torre 2006). Over the past decade, tactile sensing devices have evolved from being located only on the fingertip to covering the full hand, even the whole body, of a humanoid robot. Many tactile sensors with various sensing principles and technologies have been developed, e.g., resistive (Kaltenbrunner et al. 2013; Strohmayr et al. 2013), capacitive (Schmitz et al. 2011; Ulmen and Cutkosky 2010), optical (Ohmura et al. 2006), piezoelectric (Dahiya et al. 2009; Papakostas et al. 2002), acoustic (Denei et al. 2015; Hughes and Correll 2015) and, recently, organic bendable and stretchable (Lee et al. 2016; Nawrocki et al. 2016; Yogeswaran et al. 2015), etc. However, in contrast to the rapid progress of tactile sensor advancement, considerably less attention has been given to research in tactile information processing and modelling (Kaboli et al. 2015a). The performance of tactile systems depends not only on the technological aspects of sensory devices, but also on

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s10514-018-9707-8) contains supplementary material, which is available to authorized users.

✉ Mohsen Kaboli
mohsen.kaboli@tum.de

Gordon Cheng
gordon@tum.de

[1] Institute for Cognitive Systems, Department of Electrical and Computer Engineering, Technische Universität München, Karlstrasse 45, 80333 Munich, Germany

Published online: 19 February 2018

the design of efficient tactile perception strategies, robust feature descriptors, and tactile learning methods (Dahiya et al. 2010).

1.2 Background

Haptically accessible object characteristics can be divided into three general classes: geometric information, material properties, and inner properties (e.f. center of mass). Robots can recognize the geometric properties of objects by perceiving their shapes via either proprioceptive receptors (Jia and Tian 2010; Liarokapis et al. 2015; Liu et al. 2012, 2013) or cutaneous receptors, by exhaustively touching a single object with known orientation and location in the workspace (Jamali et al. 2016; Liu et al. 2015; Nguyen and Perdereau 2013; Yi et al. 2016). The object material can be characterized and identified by textural properties, stiffness, and thermal conductivity. The robot can sense the textural properties of objects using cutaneous tactile receptors by moving fingertips on the objects' surfaces (Chathuranga et al. 2013; Chu et al. 2013; Dallaire et al. 2014; Giguere and Dudek 2011; Jamali and Sammut 2011; Watanabe et al. 2013). The stiffness of objects can also be measured by using fingertips, in this case, by pressing objects (Lederman 1981). The thermal conductivity can be perceived by making lightly contact the finger with the objects' surfaces (Bhattacharya and Mahajan 2003).

The center of mass of rigid objects is an intrinsic physical property of the object (Yao et al. 2017). Consider several rigid objects with the same physical properties, such as shape, stiffness, and textural properties, but different centers of mass. In this case, we can discriminate objects among each other via their centers of mass.

Previous work has been done for estimating the center of mass of objects; however, to the best of our knowledge, no work has applied the center of mass property as an object feature in object classification and recognition tasks.

Atkeson et al. (1985) used a force/torque sensor to estimate the center of mass of a rigid load by solving dynamic equations of the robotic system during a manipulation task. This approach has high computational complexity and requires an accurate dynamic model of the robotic system. Yu et al. (2004, 2005) estimated the center of mass of a target object by determining at least three planes (or lines) that pass through the center of mass of the object. The robot tips the object repeatedly, and in the meantime, it estimates the function of the plane (or the line) using the current fingertip position and measured force signals. However, these approaches require the estimations of the fingertip position and force vectors of high accuracy. In addition, the stability of the target object has to be guaranteed as it is being toppled, which is often hard to satisfy in the real experiment. In this paper, we propose a purely tactile-based approach to explore the center of mass of target object, and formulate the center of mass information as an intrinsic feature of the object, which can be applied in object recognition tasks.

We humans use our sense of touch to actively explore our environment and objects through their various physical properties such as surface texture, stiffness, shape, and thermal conductivity (Kaboli et al. 2017b). To actively learn about objects through their physical properties and efficiently discriminate among them, humans strategically select tactile exploratory actions to perceive objects' properties (e.g. sliding to sense the textural properties, pressing to estimate the stiffness, and static contact to measure the thermal conductivity). Active tactile exploration is a complex procedure which requires efficient active tactile perception and active tactile learning methods.

Previous researchers have used various robotic systems with different tactile sensors to passively explore and classify objects via their physical properties (Friedl et al. 2016; Hu et al. 2014; Lepora et al. 2010; Mayol-Cuevas et al. 1998; Mohamad Hanif et al. 2015; Sinapov et al. 2011; Song et al. 2014). They used a predefined number of exploratory movements to sense physical properties of objects having fixed positions and orientation in a known workspace. Therefore, the autonomy of the robot is limited.

In this sense, active tactile exploration has shown great potential for enabling the robotic system with more natural and human-like strategies (Saal et al. 2010). The autonomous robot should be able to select and execute the exploratory actions that provide the robotic system with the maximum amount of information. In this regard, several approaches were proposed to actively discriminate among objects using their physical properties. For instance, Xu et al. (2013) used the index finger of the Shadow Hand with the BioTac sensor to collect training data by executing three different exploratory actions five times on each experimental object (pressing for stiffness, sliding for surface texture, and static contact for thermal conductivity). In Xu's work, they placed objects under the index finger, and apply a sequence of exploratory movements to construct observation models. The base and wrist of the dexterous robotic hand were fixed on a table, and all joints in the hand and wrist were deactivated (except two joints of the index finger). These physical constraints therefore resulted in an approach which is unnatural and unscalable for robotic tactile exploration. In Lepora et al. (2013), a biomimetic fingertip was controlled to slide along ten different surfaces to perceive their textural properties. In this work, the measurement of surfaces' positions were noisy. In order to actively discriminate among the surfaces under position uncertainty, the authors constructed the observation models as well as the position of the surfaces offline by uniformly sampling the collected training data of each surface texture and each possible surface position under a range of contact depths. In another study (Schneider et al. 2009), the

Weiss Robotics sensor was mounted on the end-effector of a robot arm to classify 21 objects. They created a database of tactile observations offline by grasping each object with a pre-designed trajectory. The authors managed to actively recognize objects task using tactile images, which were produced by strategically selecting the height of the robot finger and grasping the objects. Martins et al. (2014) aimed at developing a general active haptic exploration and recognition strategy for heterogeneous surfaces. The experiments were conducted to search and follow the discontinuities between regions of surfaces with two different materials. However, the experiments were only carried out in simulation using uniformly collected data offline. Tanaka et al. (2014) combined Gaussian process latent variable and nonlinear dimensionality reduction method to actively discriminate among four cups (a paper cup, a steel cup, disposal cup, and ceramic cup). The authors collected 400 training data uniformly using three fingers of the Shadow hand. In this study, the Shadow hand was fixed and the objects were placed on a turntable. The observation model was constructed with action features using the index finger with 2-DOF to generate inflective and horizontal movements on the objects. The authors could discriminate four cups in real experiments and 10 different cups in simulation. Since the proposed method requires a huge amount of training data, the high dimensional action space makes the optimal action search and model learning intractable.

In the above-mentioned work, the location and orientation of the experimental objects in the workspace were known. Moreover, in order to construct the observation models, the training samples were collected uniformly and offline. To increase the autonomy of a robotic system for the tactile-based object recognition, the robot should be able to autonomously explore an unknown workspace, actively detect the number of objects, as well as estimate their positions and orientations in the workspace. Furthermore, the informativeness of the training data collected with each object is different. Some objects have distinctive tactile properties, which makes it easy to discriminate them among each other. Therefore, collecting too many training samples with such objects is redundant; whereas for objects, whose physical properties are similar and thus can be easily confused with other objects' properties, it is necessary to collect sufficient samples to construct reliable and robust observation models. Moreover, in order to efficiently discriminate among objects, the autonomous robot should strategically select and execute the exploratory action that provides the robot with the maximum amount of information.

1.3 Contribution

To tackle the aforementioned problems, we propose a tactile-based probabilistic framework for active workspace exploration and active object recognition. Following our proposed framework, the robotic system autonomously and efficiently explores an unknown workspace to collect tactile data of the object (construct the tactile point cloud dataset), which are then clustered to determine the number of objects in the workspace and estimate the location and orientation of each object. The robot strategically selects the next position in the workspace to explore, so that the total variance of the workspace can be reduced as soon as possible.

Then the robot efficiently learns about the objects' physical properties, such that with a smaller number of training data, reliable observation models can be constructed using Gaussian process for stiffness, surface texture, and center of mass.

Taking advantage of the constructed observation models, the robotic system efficiently discriminate among objects and search for specified target objects by strategically selecting the optimal exploratory actions to apply on objects to perceive the corresponding physical property (sliding to sense textural properties, pressing to measure stiffness, lifting to determine center of mass).

Furthermore, for the first time, the center of mass of rigid objects is considered as an intrinsic property for object learning and discrimination. In this regard, we propose a tactile-based algorithm to enable robotic systems to estimate the center of mass of rigid objects.

Our contribution is summarized in four aspects:

(a) We propose a strategy based on Gaussian process regression method to enable robotic systems equipped with tactile sensors to autonomously explore unknown workspaces, in order to localize and estimate the orientation of objects within it.
(b) We propose a method for autonomous systems to actively learn about the objects based on their physical properties, and to construct Gaussian process based observation models objects with the least required number of training samples.
(c) We propose a method so that the robot can efficiently discriminate among objects, or search for target objects in a workspace that contains unknown objects, by selecting a sequence of exploratory movements to apply on the objects.
(d) We propose a tactile-based method to explore the center of mass of rigid objects.

2 System description

2.1 Robotic gripper

We used the Robotiq 3-Finger Adaptive Robot Gripper which is an under-actuated industrial gripper with three fingers (A,

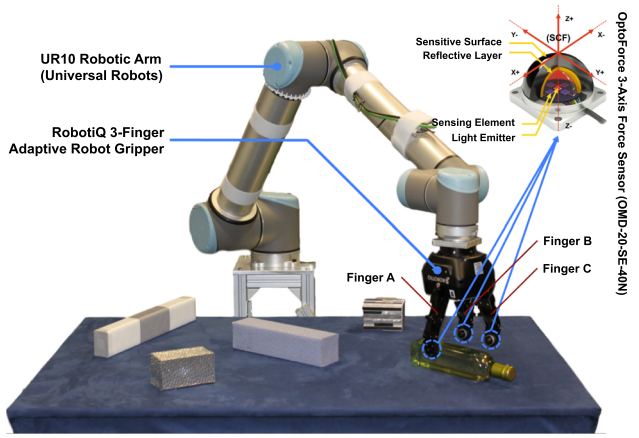


Fig. 1 The experimental setup. A Robotiq three-finger adaptive robot gripper is equipped with OptoForce sensors and mounted on a UR10 robotic arm. In this figure, five experimental objects are selected and placed in the workspace for illustration purpose. In experiments, different experimental objects can be placed in the workspace with various locations and orientations

B, and C) (see Fig. 1). Finger B and C were aligned on the same side of the gripper, and they moved in the opposite direction as finger A. The position range of each of the gripper's fingers was divided into 255 counts, with 0 indicating fully open, and 255 fully closed. Thus, the position of the finger was represented using position counts in this paper.

2.2 Robotic arm

We mounted the gripper at the end of the UR10 (Universal Robots) robotic arm (6-DoF) (see Fig. 1), which was controlled to collaborate with the gripper, in order to explore the workspace and interact with objects.

2.3 Tactile sensors

Three channels of the OptoForce OMD-20-SE-40N 3D tactile sensor set were installed on each fingertip of the gripper. The OptoForce sensor can measure forces in three directions, using infrared light to detect small deformation in the shape of the outer sensor surface. It has a nominal capacity of 40N in Z_{SCF} direction, and $\pm$ 20N in both X_{SCF} and Y_{SCF} directions. In this paper, we discuss forces in two coordinate frames: world coordinate frame (WCF) (see Fig. 3) and sensor coordinate frame (SCF) (see Fig. 1), both of which are standard Cartesian coordinate systems. In SCF, we discuss the tangential force vector $\boldsymbol{f}_{T_i}$ and the normal force vector $\boldsymbol{f}_{N_i}$ exerted on the grasped object. The value of tangential force exerted by the ith finger is calculated as

$$|\boldsymbol{f}_{T_i}| = \left(|\boldsymbol{f}_{x_i}|^2 + |\boldsymbol{f}_{y_i}|^2\right)^{-1/2} \quad (1)$$

The force vectors in SCF are represented as $\boldsymbol{f}_{x_i}$, $\boldsymbol{f}_{y_i}$, and $\boldsymbol{f}_{z_i}$ with the subscript i denoting the index of the finger, $i = 1, 2, 3$, correspond to Fingers A, B, and C, respectively.

3 Proposed framework

Our proposed probabilistic tactile-based framework (Fig. 2) consists of three parts: (1) an active touch approach for exploring the unknown workspace (Fig. 2a), (2) an active touch method for efficiently learning about objects' physical properties (surface texture, stiffness, and center of mass) (Fig. 2b), and (3) an active object discriminating strategy via objects' physical properties and an active target search method (Fig. 2c).

First, taking advantage of the Gaussian process regression (GPR), the robot efficiently explores the workspace from different directions, by strategically selecting the position to explore, so that the total uncertainty of the workspace can be reduced as quickly as possible. The tactile data captured during exploration are then clustered in order to determine the number of objects in the workspace. For each cluster, a minimum bounding box is calculated to estimate the location, orientation, and geometric center of each object. After that, the robot starts learning about objects via their physical properties by strategically selecting the objects to explore and the physical properties to perceive. To do this, the robot moves to the selected object and applies the corresponding exploratory action on the object (sliding to sense the textural properties, pressing to estimate the stiffness, lifting to determine the center of mass (CoM)) in order to perceive its physical property. Later on, the robot exploits the perceived physical property of the explored objects and constructs observation models using the Gaussian process classification (GPC). The constructed observation models will then be used as the robot's prior knowledge for actively discriminating among objects (Fig. 2c-1) and/or searching for the unknown target objects (Fig. 2c-2) in the workspace. Each part of the framework will be explained in detail in the following sections.

4 Active touch for unknown workspace exploration

In this section, we propose a probabilistic active touch method for robotic system with the sense of touch to efficiently explore an unknown workspace.

4.1 Problem formulation

Taking advantage of the proposed approach, the robot strategically executes translational movements from each direction

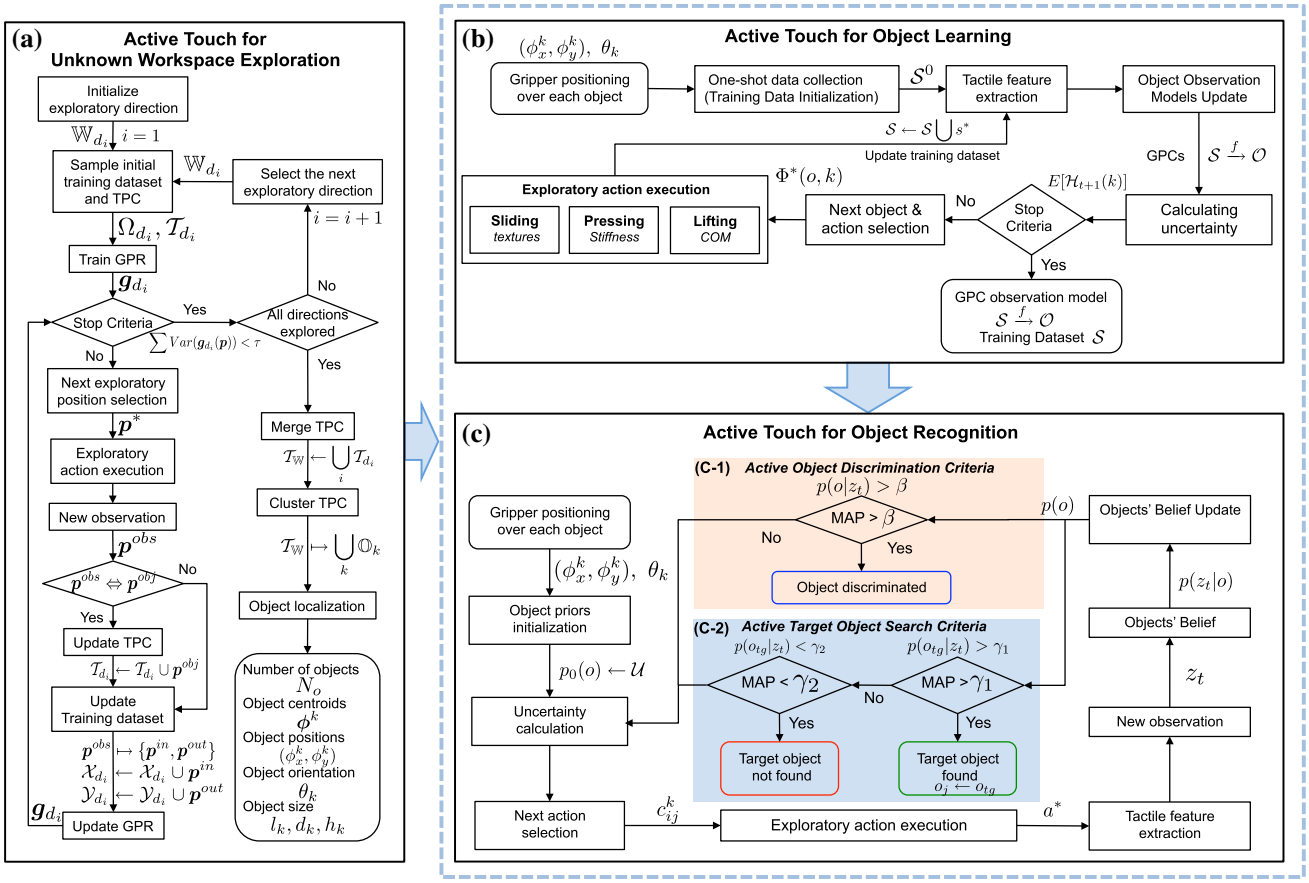


Fig. 2 Our proposed probabilistic tactile-based framework consists of three components: **a** an active touch method for workspace exploration; **b** an active touch algorithm for efficient learning about objects' physical properties (texture, stiffness, center of mass); **c** and active touch for recognizing objects, which has two parts: **c-1** an active touch for discrimination among objects based on their physical properties; **c-2** an active touch strategies for searching a target object/s in the workspace that contains unknown objects

into the workspace, in order to efficiently detect contact with objects via the tactile sensors that installed on the fingertips.

The sampled points obtained during each movement are used to construct the probabilistic observation model (constructed using the GPR) of the workspace from the current exploratory direction. Among the collected sampled points, the ones that are detected on the object surface (i.e. the robot contacted the object before it reached the target position of the movement), are registered into the tactile point cloud (TPC) dataset, which is clustered and used to localize objects after exploration.

The GPR model guides the exploration by predicting the next exploratory position, which is the position that has the maximum uncertainty in the current explored workspace. Consequently, the total uncertainty of the workspace can be reduced as quickly as possible during exploration.

After the explorations from all possible directions are completed, the entire TPC dataset is clustered, and then the clustering result is used to localize and map the experimental objects in the workspace.

4.1.1 Tactile point cloud construction

The workspace is defined as a discretized 3D mesh-grid within the reaching capabilities of the robotic hand. Spatial point in the workspace is denoted as $\boldsymbol{p}_n = (x_n, y_n, z_n)$, $\boldsymbol{p}_n \in \mathbb{R}^3$, $n \in \mathbb{N}$,[1] which lies in predefined boundaries in the WCF: $x_n \in [\underline{x}, \overline{x}]$, $y_n \in [\underline{y}, \overline{y}]$, and $z_n \in [\underline{z}, \overline{z}]$ with the underline and overline of x, y, and z denoting the lower and upper boundaries of the corresponding axis of the WCF, respectively. The workspace can be explored from multiple directions d_i, $i \in \mathcal{I}_{DIR}$. In this work, the workspace is explored from four directions: $\{d_i\} \equiv \{X_+, Y_-, X_-, Y_+\}$, $\mathcal{I}_{DIR} = \{1, 2, 3, 4\}$ with the subscript "+" and "−" representing the positive and negative directions of the X and Y directions of the WCF (see Fig. 3). The workspace in the direction d_i is denoted as $\mathbb{W}_{d_i}$. For the exploration in each direction, the start plane and the target plane are one pair of opposite side faces of the workspace, and they are perpendicular to the exploratory axis. In the exploration process, the nth action

[1] $\mathbb{R}$ is the set of real numbers, $\mathbb{N}$ is the set of natural numbers.

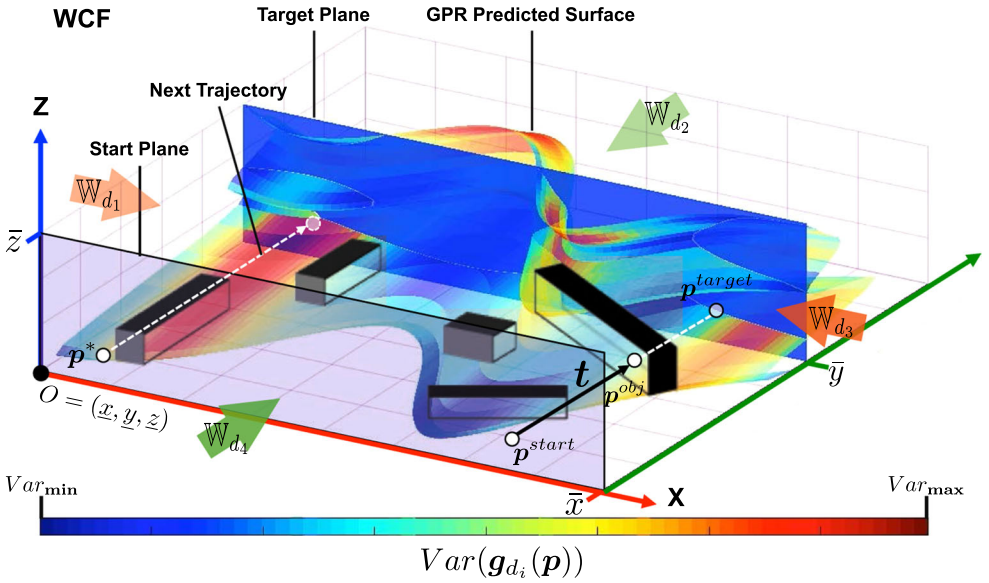


Fig. 3 The illustration of the active exploration process of an unknown workspace $\mathbb{W}_{d_4}$ (workspace in $d_4 = Y_+$ direction). The uncertainties (variance) at different positions are revealed by the color. Red color represents high uncertainty, while blue color represents low uncertainty (Color figure online)

$\boldsymbol{t}_n \in \mathbb{R}^3$ is a translational movement from one position on the start plane $\boldsymbol{p}_n^{start}$ to the corresponding position on the target plane $\boldsymbol{p}_n^{start}$, and the trajectory of $\boldsymbol{t}_n$ is parallel to the exploratory direction. After executing the action $\boldsymbol{t}_n$, an observation $\boldsymbol{p}_n^{obs}$ is obtained.

A light contact is detected as soon as the resultant force $|\boldsymbol{f}^{RES}|$ on the surface of the exploratory finger exceeds a threshold, i.e. $|\boldsymbol{f}^{RES}| > \delta$. The resultant force on the exploratory finger is calculated as:

$$|\boldsymbol{f}^{RES}| = \left(|\boldsymbol{f}_x|^2 + |\boldsymbol{f}_y|^2 + |\boldsymbol{f}_z|^2\right)^{-1/2} \tag{2}$$

with $\boldsymbol{f}_x$, $\boldsymbol{f}_y$, and $\boldsymbol{f}_z$ being the force components in the X-axis, Y-axis, and Z-axis of the SCF, respectively.

During the movement, if the sensor on the exploratory finger has detected a light contact before reaching to $\boldsymbol{p}_n^{start}$, indicating the robot has touched an object on its surface, then the current 3D coordinates of the contact position, $\boldsymbol{p}_n^{obs} = \boldsymbol{p}_n^{object}$, will be recorded. However, if no light contact is detected until the movement is completed, the target position will be returned as the obtained observation, i.e. $\boldsymbol{p}_n^{obs} = \boldsymbol{p}_n^{target}$.

The TPC dataset $\mathcal{T}_{d_i}$ is the set of all the $\boldsymbol{p}_n^{object}$ collected during the exploration of $\mathbb{W}_{d_i}$. The complete TPC of the workspace is denoted as $\mathcal{T}_{\mathbb{W}} = \bigcup_i \mathcal{T}_{d_i}$.

Here we take the exploration of $\mathbb{W}_{d_4}$ as an example to clarify this procedure (see Fig. 3). In this case, the exploratory direction is $d_4 = Y_+$, thus the y coordinates of the spacial points increase along the exploratory direction. The start plane is the X–Z plane that passes through the positions with minimum Y coordinates (points satisfy $y = \underline{y}$), while the target plane is parallel to the start plane and passes through positions with maximum Y coordinates (points satisfy $y = \overline{y}$). Thus $\boldsymbol{t}_n$ starts from $\boldsymbol{p}_n^{start} = (x_n, \underline{y}, z_n)$ and points towards $\boldsymbol{p}_n^{target} = (x_n, \overline{y}, z_n)$. If an observation $\boldsymbol{p}_n^{obs} = \boldsymbol{p}_n^{object}$ is obtained, it will be added into $\mathcal{T}_{d_4}$. Otherwise, $\mathcal{T}_{d_4}$ will not be updated.

4.1.2 Workspace modeling

We use Gaussian Process Regression approach to construct a probabilistic model of the workspace in the current exploratory direction to guide the exploratory process. A brief introduction of the Gaussian process (GP) method (Rasmussen and Williams 2006) can be found in Sect. 11.1.

We explain the application of GP in exploration task through the example introduced in Sect. 4.1.1.

For the exploration of $\mathbb{W}_{d_4}$, all points that trajectory $\boldsymbol{t}_n$ passes through have the same x_n and z_n coordinate, and $\boldsymbol{p}_n^{obs}$ returns the y_n coordinate, i.e. the depth of the object surface from the start point $\boldsymbol{p}_n^{start}$ in this direction.

We considered the single training input of the GPR model $\boldsymbol{g}_{d_4}$, $\boldsymbol{p}_n^{in} = (x_n, z_n)$, $\boldsymbol{p}_n^{in} \in \mathcal{X}_{d_4}$, $\mathcal{X}_{d_4} \subseteq \mathbb{R}^2$, $n \in \mathbb{N}$, and the corresponding training label (output): $\boldsymbol{p}_n^{out} = (y_n)$, $\boldsymbol{p}_n^{out} \in \mathcal{Y}_{d_4}$, $\mathcal{Y}_{d_4} \subseteq \mathbb{R}$, $n \in \mathbb{N}$, with (x_n, y_n, z_n) being the coordinates of the observed point $\boldsymbol{p}_n^{obs}$ of the trajectory $\boldsymbol{t}_n$.

The entire training dataset of $\boldsymbol{g}_{d_i}$ is denoted as $\Omega_{d_i} = \{\mathcal{X}_{d_i}, \mathcal{Y}_{d_i}\}$. Represent the universal set of the 2D coordinate $\boldsymbol{p}_n$ all over the corresponding start plane as $\mathbb{X}_{d_i}$; and $\boldsymbol{p}_n = (y_n, z_n)$ for $i = 1, 3$, whereas $\boldsymbol{p}_n = (x_n, z_n)$ for $i = 2, 4$.

Given N pairs of training data $\{\mathcal{X}_{d_4}, \mathcal{Y}_{d_4}\} = \{\boldsymbol{p}_n^{in}, \boldsymbol{p}_n^{out}\}_{n=1:N}$, the predicted distribution of target function

$\boldsymbol{g}_{d_4}(\boldsymbol{p})$, $\boldsymbol{p} \in \mathbb{X}_{d_i}$ is denoted as $\hat{\boldsymbol{g}}_{d_4}(\boldsymbol{p}) \sim \mathcal{GP}(\hat{\mu}(\boldsymbol{p}), \hat{\upsilon}(\boldsymbol{p}))$, and the corresponding mean function and variance function are calculated as:

$$\hat{\mu}(\boldsymbol{p}) = \tilde{\boldsymbol{k}}^{\mathrm{T}}(\mathrm{K} + \sigma_n^2 \mathrm{I})^{-1}\mathbf{y}, \tag{3}$$

$$\hat{\upsilon}(\boldsymbol{p}) = k(\boldsymbol{p}, \boldsymbol{p}) - \tilde{\boldsymbol{k}}^{\mathrm{T}}(\mathrm{K} + \sigma_n^2 \mathrm{I})^{-1}\tilde{\boldsymbol{k}}. \tag{4}$$

where $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is the covariance function, $\tilde{\boldsymbol{k}}$ is the covariance vector with its nth element indicating the covariance between the test input $\boldsymbol{p}$ and the nth training data point $\boldsymbol{p}_n^{out}$, and $\mathbf{y} \in \mathbb{R}^N$ is a vector of training outputs $\boldsymbol{p}_n^{out}$. The (i, j) entry of the matrix K represents the covariance between ith and jth training inputs, i.e. $\mathrm{K}_{i,j} = k(\boldsymbol{p}_i^{in}, \boldsymbol{p}_j^{in})$.

The predicted target $\hat{\boldsymbol{p}}^{out}$ for the test input $\hat{\boldsymbol{p}}^{in}$ subjects to the Gaussian distribution: $\hat{\boldsymbol{p}}^{out} \sim \mathcal{N}(\hat{\mu}(\hat{\boldsymbol{p}}^{in}), \mathrm{K} + \sigma_n^2 \mathrm{I})$, and the probability of predicted $\hat{\boldsymbol{p}}^{out}$ is denoted as $p(\hat{\boldsymbol{p}}^{out})$.

4.1.3 Next exploratory position selection

The contact positions $\boldsymbol{p}^{obs} \in \mathcal{T}$ are discretely distributed points in the workspace. Hence, a large number of exploratory movements are required to obtain an authentic estimation of the workspace, which is not data-efficient and also time consuming.

A strategy for selecting the next exploratory position is necessary to reduce the total number of non-informative exploratory samples. In order to select the next sample position, several approaches are used, such as Jamali et al. (2016) and Martinez-Cantin et al. (2007). These approaches explore the unknown area and exploit the information from the known region in the workspace. In this paper, we focus on the uncertain region of the workspace during exploration in order to reduce the total uncertainty of the workspace model as soon as possible. Considering the example discussed in Sect. 4.1.2, a GPR model $\boldsymbol{g}_{d_4}$ is trained to make a prediction of $\boldsymbol{p}_n^{out} = y_n$, given input $\boldsymbol{p}_n^{in} = (x_n, z_n)$.

We propose using the variance predicted by the GPR model, $Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$, since it indicates the uncertainty in the current model at input position $\boldsymbol{p}$, $\boldsymbol{p} \in \mathbb{X}_{d_i}$. In addition, the uncertainty of the workspace $\mathbb{W}_{d_i}$ modeled by the GPR model $\boldsymbol{g}_{d_i}$ can be measured by its total variance, defined as $\sum_{\boldsymbol{p} \in \mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$.

To reduce the total variance as soon as possible, we select the next exploratory position $\boldsymbol{p}^* = (x_{n+1}, z_{n+1})$ as the one with the largest variance in the present GPR model:

$$\boldsymbol{p}^* = \underset{\boldsymbol{p} \in \mathbb{X}_{d_i}}{\operatorname{argmax}}\ Var(\boldsymbol{g}_{d_i}(\boldsymbol{p})). \tag{5}$$

In other words, the robot explores the position $\boldsymbol{p}^*$, which the current trained workspace model is mostly uncertain of.

4.1.4 One-shot data collection for initializing the GPR

At the beginning of the exploration process of one direction, the robot first samples a few uniformly located points on the start plane to initialize the GPR model. For example, the training dataset of the GPR model $\boldsymbol{g}_{d_4}$ is denoted as $\Omega_{d_4} = \{\mathcal{X}_{d_4}, \mathcal{Y}_{d_4}\}$, which can be initialized by sampling $M \times N$ points on the start plane, these points are represented as $(\underline{x} + \frac{m}{M-1}(\overline{x} - \underline{x}),\ \underline{z} + \frac{n}{N-1}(\overline{z} - \underline{z}))$, $m = 0, 1, \ldots, M-1$, $n = 0, 1, \ldots, N-1$. In the meantime of collecting Ω_{d_4}, sampled points which satisfy $\boldsymbol{p}^{obs} = \boldsymbol{p}^{object}$ are registered to the TPC dataset $\mathcal{T}_{d_4}$. Then $\boldsymbol{g}_{d_4}$ is trained using the dataset Ω_{d_4}, as described in Sect. 4.1.2.

4.1.5 Updating the total uncertainty of TPC

After initialization, the robot selects the next exploration position based on the GPR prediction. As soon as the next exploratory position $\boldsymbol{p}^* = (x_{n+1}, z_{n+1})$ is determined, the robot moves to the start position $\boldsymbol{p}_{n+1}^{start} = (x_{n+1}, \underline{y}, z_{n+1})$ on the start plane and then executes an exploratory movement towards the corresponding target position $\boldsymbol{p}_{n+1}^{target} = (x_{n+1}, \overline{y}, z_{n+1})$. During the movement, the robot maintains the orientation of the tactile sensor in the d_4 direction.

If a contact on the object surface is detected during this motion, i.e. $\boldsymbol{p}_{n+1}^{obs} = \boldsymbol{p}_{n+1}^{object}$, the current 3D position of the sensor, $\boldsymbol{p}_{n+1}^{object} = (x_{n+1}, y_{n+1}, z_{n+1})$, is registered to $\mathcal{T}_{d_4}$, and the robot immediately retreats back to the start position. If no contact is detected, the observation $\boldsymbol{p}_{n+1}^{obs} = \boldsymbol{p}_{n+1}^{target}$ is returned ($y_{n+1} = \overline{y}$).

The coordinate of the sampled point along the exploratory direction is used as the training output (i.e. label set), while the other two dimensions are used as training input. The observation $\boldsymbol{p}_{n+1}^{obs}$ is added into the training set Ω_{d_4} by appending $\boldsymbol{p}_{n+1}^{in} = (x_{n+1}, z_{n+1})$ into $\mathcal{X}_{d_4}$ and appending $\boldsymbol{p}_{n+1}^{out} = (y_{n+1})$ into $\mathcal{Y}_{d_4}$. The GPR model is updated using the updated Ω_{d_4}, and then the next exploratory position is selected according to Eq. (5).

4.1.6 Stop criteria

The exploratory process in the current direction continues until a stop criterion is satisfied. For example, when the total uncertainty of the model, $\sum_{\boldsymbol{p} \in \mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$, reduces below a tolerable threshold τ. When the exploration process terminates in one direction (d_i), the robot starts the new exploration in the next exploratory direction d_{i+1} by following the same procedure. The entire unknown workspace is completely explored when the exploratory processes are finished for all $\mathbb{W}_{d_i}$.

Algorithm 1 Active unknown workspace exploration

TPC Construction

Require: $[\underline{x}, \overline{x}]$, $[\underline{y}, \overline{y}]$, $[\underline{z}, \overline{z}]$ ▷ workspace description
1: **for** $\mathbb{W}_{d_i}$, $i \in \mathcal{I}_{DIR}$ **do**
2: sample $\Omega_{d_i} = \{\mathcal{X}_{d_i}, \mathcal{Y}_{d_i}\}$ ▷ Sect. 4.1.3
3: initialization: $\mathcal{T}_{d_i}$ ▷ $\mathcal{T}_{d_i} \subseteq \Omega_{d_i}$
4: initialization: train GPR $\boldsymbol{g}_{d_i} : \mathcal{X}_{d_i} \mapsto \mathcal{Y}_{d_i}$ using Ω_{d_i} ▷ Sect. 4.1.2
5: initialization: $\sum_{\boldsymbol{p}\in\mathbb{X}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$ ▷ initialize total uncertainty
6: **while** $\sum_{\boldsymbol{p}\in\mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p})) > \tau$ **do**
7: $\boldsymbol{p}^* \leftarrow \underset{\boldsymbol{p}\in\mathbb{X}_{d_i}}{\text{argmax}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$ ▷ Sect. 4.1.3
8: $\boldsymbol{p}^{obs} \leftarrow$ execute action following $\boldsymbol{t}$
9: **if** $\boldsymbol{p}^{obs}$ is $\boldsymbol{p}^{object}$ **then** ▷ collected on the object surface
10: $\mathcal{T}_{d_i} \leftarrow \mathcal{T}_{d_i} \cup \boldsymbol{p}^{obs}$ ▷ update TPC
11: $\mathcal{X}_{d_i} \leftarrow \mathcal{X}_{d_i} \cup \boldsymbol{p}^{in}$, $\mathcal{Y}_{d_i} \leftarrow \mathcal{Y}_{d_i} \cup \boldsymbol{p}^{out}$ ▷ update Ω_{d_i}
12: $\Omega_{d_i} = \mathcal{X}_{d_i}, \mathcal{Y}_{d_i}$
13: train $\boldsymbol{g}_{d_i}$ using Ω_{d_i} ▷ update the GPR model $\boldsymbol{g}_{d_i}$
14: calculate $\sum_{\boldsymbol{p}\in\mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$ ▷ update total uncertainty

Objects Localization

15: $\mathcal{T}_{\mathbb{W}} = \bigcup_i \mathcal{T}_{d_i}$ ▷ merge TPC
16: $\bigcup_k \mathbb{O}_k, k = 1, 2, \ldots, N_o \leftarrow cluster(\mathcal{T}_{\mathbb{W}})$ ▷ cluster TPC, Sect. 4.2.1
17: **for** $k = 1 : N_o$ **do** ▷ Sect. 4.2.2
18: construct $\mathcal{B}_k$ ▷ construct bounding box
19: calculate $\{\boldsymbol{v}_i^k\}, i = 1, 2, \ldots, N_v$ ▷ construct set of vertices
20: calculate $\boldsymbol{\phi}^k = (\phi_x^k, \phi_y^k, \phi_z^k)$ ▷ estimate center of object
21: estimate l_k,d_k,h_k ▷ estimate object shape
22: estimate θ_k ▷ estimate object orientation

4.2 Object localization and mapping

When the exploration of the unknown workspace is completed, the TPC of the entire workspace, $\mathcal{T}_{\mathbb{W}}$, can be constructed by merging all the TPCs collected in different directions: $\mathcal{T}_{\mathbb{W}} = \bigcup_i \mathcal{T}_{d_i}$. We cluster all the data points in $\mathcal{T}_{\mathbb{W}}$ to obtain the points that belong to the same object into one category, so as to localize each object in the workspace.

4.2.1 Clustering of tactile point cloud

In order to cluster the constructed tactile point clouds we use the mean-shift clustering method (Cheng 1995), which is non-parametric and application-independent. The mean-shift clustering approach does not require prior knowledge of the number of categories, and its performance is independent of the shape of the data clusters. After clustering, $\mathcal{T}_{\mathbb{W}}$ is divided into N_o mutual exclusive subsets, denoted as $\mathcal{T}_{\mathbb{W}} \mapsto \bigcup_k \mathbb{O}_k, k = 1, 2, \ldots, N_o$, where N_o is the estimated number of objects, and $\mathbb{O}_k$ contains all the points in the kth cluster, i.e. belong to the kth object. For the sake of increasing robustness against the noise, a minimum number of data points contained in one category is assigned. If a cluster $\mathbb{O}_k$ contains fewer data points than this lower limit, points in this cluster are considered as noise and this cluster will be discarded.

4.2.2 Object localization

For estimating the location and geometric measurement of each clustered object, a 3D minimum bounding box $\mathcal{B}_k$ is calculated for each point set $\mathbb{O}_k$, $k = 1, 2, \ldots, N_o$. The minimum bounding box is the smallest enclosing volume that contains all the points in the data set. The vertices of each $\mathcal{B}_k$ are represented as: $\boldsymbol{v}_i^k = (v_{i_x}^k, v_{i_y}^k, v_{i_z}^k)$, $i = 1, 2, \ldots, N_v$, where $N_v = 8$ in this work. The geometric center of the kth object, $\boldsymbol{\phi}^k = (\phi_x^k, \phi_y^k, \phi_z^k)$, is calculated as: $\boldsymbol{\phi}^k = (\sum_i v_{i_x}^k/N_v, \sum_i v_{i_y}^k/N_v, \sum_i v_{i_z}^k/N_v)$, therefore the object is located at (ϕ_x^k, ϕ_y^k) on the reference plane (the X–O–Y plane) of the workspace.

The geometric measurement of the object, i.e. length l_k, width d_k, and height h_k can be roughly estimated by calculating the Euclidean distance between vertices on the reference plane (for l_k and d_k, $l_k > d_k$) and in the Z direction (for h_k). The orientation of the kth object is $\theta_k \in [0, \pi]$, defined as the angle that is included between its long edge (the l_k edge) and the positive direction X axis of the WCF. As soon as the geometric information (l_k, d_k, and h_k) of the kth object is determined, an object coordinate frame (OCF) can be defined with respect to the object. As an example, to defined the OCF for a cuboid object, the origin can be assigned as one vertex of the object, and the X, Y, and Z axes of the OCF can be defined as along the length (l_k), depth (d_k), and height (h_k) edge of the object's bounding box.

The entire active exploration process of the unknown workspace is summarized in Algorithm 1.

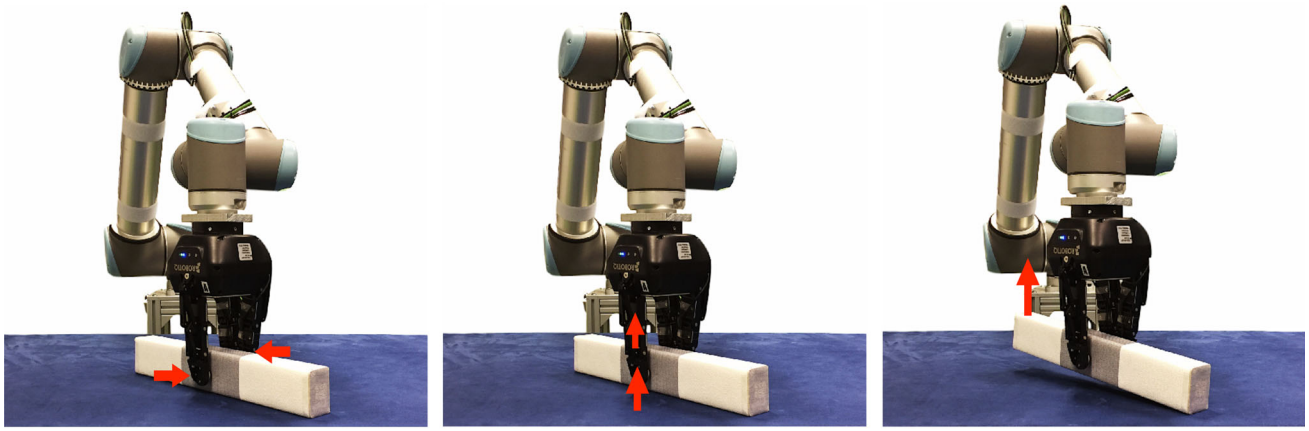
5 Objects' physical properties perception

In tactile object recognition problem, the physical properties of objects are perceived by executing various exploratory actions on the objects. For instance, a robotic system with tactile sensing presses an object to obtain its stiffness, slides on the object's surface to perceive its textural property, and lifts the object at several positions to determine its center of mass. If various exploratory actions are executed on the same objects, multiple physical properties can be sensed by the robot. In this part, we introduce our approaches to perceive the physical properties of the objects, namely stiffness, surface texture, and center of mass.

5.1 Exploratory actions and tactile feature descriptors

5.1.1 Stiffness

The robot perceives the stiffness of objects by pressing the tactile sensors against the objects' surfaces (see Fig. 4a). In



(a) **(b)** **(c)**

Fig. 4 Exploratory actions. **a** Pressing, to measure the stiffness. **b** Sliding, to perceive the textural property. **c** Lifting, to check if the current position is the CoM

this study, by exploiting the geometric information of target object computed during workspace exploration (Sect. 4), the robot first moves to the target object and adjusts the orientation of the gripper to keep the finger facing the object's surface. Then, the gripper establishes a light contact with the object by all three fingertips. The light contact is detected as soon as the measured resultant forces (f_r^{RES}) of all tactile sensors exceed a threshold (f_t), i.e. $f_r^{RES} > f_t,\ r = 1, 2, 3$. Afterwards, the gripper presses the object by closing all its fingers simultaneously for N_ϵ extra position counts. For each finger r, the difference in its resultant forces recorded before and after pressing, Δf_r^{RES}, is used as an indication of the stiffness on the local contact area. The difference value averaged over all fingers serves as a measurement of stiffness of the object.

$$\mathcal{S}_{O_i} = \sum_{r}^{N_r} \Delta f_r^{RES}. \tag{6}$$

with the subscript O_i indicating the ith object, and N_r the total number of tactile sensors in contact with the object.

5.1.2 Surface texture

When the robot slides its fingertips on the surface of an object, it generates vibration (see Fig. 4b). The caused vibration can be measured by each tactile sensor on the fingertip $f_{n_v}(n_v = 1, 2, \ldots, N_v$, is the number of output signals from one tactile sensor) to sense the textural property of the object.

In this regard, we previously proposed a set of novel tactile descriptors to extract the robust tactile information from the output of the tactile sensors (Kaboli et al. 2015b). The proposed descriptor provides the statistical properties of the tactile signals in the time domains. By taking the advantage of our proposed descriptor the robot can extract robust tactile features from the output of the both stationary and non-stationary tactile signals (regular and irregular surface texture). The proposed descriptors are called *Activity*, *Mobility*, and *Complexity*.

The *Activity* (Eq. 7) is the total power of a signal. The *Mobility* parameter (Eq. 8) is the square root of the ratio of the variance of the first derivative of the signal to that of the signal. The *Complexity* (Eq. 9) is the second derivative of the variance and shows how the shape of the signal is similar to a pure sine wave. If the signal is more similar to the sine wave, the complexity value converges to 1.

$$Act(\boldsymbol{f}_{n_v}) = \frac{1}{N}\sum_{n=1}^{N}(f_{n,n_v} - \bar{f}_{n_v})^2. \tag{7}$$

$$Mob(\boldsymbol{f}_{n_v}) = \left(\frac{Act(\frac{df_{n_v}}{dn})}{Act(\boldsymbol{f}_{n_v})}\right)^{-1/2}. \tag{8}$$

$$Com(\boldsymbol{f}_{n_v}) = \frac{Mob(\frac{df_{n_v}}{dn})}{Mob(\boldsymbol{f}_{n_v})}. \tag{9}$$

Tactile object exploration first requires an initiation of a static contact with the surface of objects by a fingertip or in general sensitive skin and then sliding the fingertip or any body part/s (with sensitive skin) across the surface of objects (dynamic motion). The transition from the static state to the dynamic state (and vice-versa) during tactile texture exploration depends very much on the frictional properties of the surface texture of objects. Robotic systems (for instance a fingertip) need to apply more force to transit from the static state to the dynamic state in order to explore the surface of objects with a high frictional coefficient. Such a transition affects the outer layer of the robotic skin (it is usually made of materials such as silicone). This results in deformation of the outer layer of the robotic skin, which generates linear

or/and nonlinear correlation between outputs of tactile sensors in the outer layer of the tactile sensor or robotic skin. Therefore, we proposed in (Kaboli et al. 2014; Kaboli and Cheng 2018) to consider the linear correlation (Eq. 10) and nonlinear correlation coefficients (Eq. 11) between tactile signals/sensors as additional tactile features. These features indirectly provide information about the frictional properties of the surface of objects with the robotic systems during the exploratory procedure.

In all equations, f_{n_v} and f_{n_s} are the input signals, N is the number of data samples, and R_k is the difference between the rank of f_{n_v} and the rank of f_{n_s}.

$$L^{cor}_{\mathbf{f}_{n_v},\mathbf{f}_{n_s}} = \frac{\sum_{n=1}^{N}(f_{n,n_v} - \bar{f}_{n_v}) \cdot (f_{n,n_s} - \bar{f}_{n_s})}{\sigma(\mathbf{f}_{n_v}) \cdot \sigma(\mathbf{f}_{n_s})}. \tag{10}$$

$$N^{cor}_{f_{n_v},f_{n_s}} = 1 - \frac{6\sum_{n=1}^{N}(R_k)^2_n}{N(N^2-1)}. \tag{11}$$

$$\mathbf{A}_{grip} = \left[\frac{1}{N_v}\sum_{n_v=1}^{n_v} Act(f_{n_v})^{F_A}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Act(f_{n_v})^{F_B}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Act(f_{n_v})^{F_C}\right] \tag{12}$$

$$\mathbf{M}_{grip} = \left[\frac{1}{N_v}\sum_{n_v=1}^{N_v} Mob(f_{n_v})^{F_A}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Mob(f_{n_v})^{F_B}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Mob(f_{n_v})^{F_C}\right] \tag{13}$$

$$\mathbf{C}_{grip} = \left[\frac{1}{N_v}\sum_{n_v=1}^{N_v} Com(f_{n_v})^{F_A}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Com(f_{n_v})^{F_B}, \frac{1}{N_v}\sum_{n_v=1}^{N_v} Com(f_{n_v})^{F_C}\right] \tag{14}$$

$$\mathbf{L}_{grip} = \left[\frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} L^{corr}(f_{n_v}, f_{n_s})^{F_A}, \frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} L^{corr}(f_{n_v}, f_{n_s})^{F_B}, \frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} L^{corr}(f_{n_v}, f_{n_s})^{F_C}\right] \tag{15}$$

$$\mathbf{N}_{grip} = \left[\frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} N^{corr}(f_{n_v}, f_{n_s})^{F_A}, \frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} N^{corr}(f_{n_v}, f_{n_s})^{F_B}, \frac{1}{N_s N_v}\sum_{n_s=1}^{N_s}\sum_{n_v=1}^{N_v} N^{corr}(f_{n_v}, f_{n_s})^{F_C}\right] \tag{16}$$

$$\mathbf{D}_{total} = \left[\mathbf{A}_{grip};\ \mathbf{M}_{grip};\ \mathbf{C}_{grip};\ \mathbf{L}_{grip};\mathbf{N}_{grip}\right] \tag{17}$$

In this study, we adapted our proposed tactile descriptors to extract robust tactile information from the output of the three-axis OptoForce tactile sensor (Eqs. 12, 13, 14), and (Eqs. 15, 16). The final proposed feature descriptors for the robotic system with three fingers (F_A, F_B, and F_C) and each fingertip with a three-axis tactile sensor ($n_v = 1, 2, 3$; $N_v = 3$ is the number of output signal from one tactile sensor) can be written as Eq. (17).

The final feature descriptor (Eq. 17) is the concatenation of all extracted features as one feature vector. The final feature vector D_{total} has 15 data samples.

5.1.3 Center of mass

The CoM of a rigid body is a constant position with respect to the object. In this work, we present a tactile-based approach to determine the CoM of the target object via lifting actions.

Think of the process of lifting an object, for example, a steelyard. The steelyard can only maintain its balance (in the equilibrium state) during lifting, if and only if the resultant force's line of application passes through its CoM. In this case, two conditions should be satisfied, i.e. the force condition and the torque condition, which state that both resultant force and resultant torque applied on the lifted object are zero.

We show that in a three-contact-point case, both force and torque conditions can be verified via tactile-based approaches by taking advantage only of the force signals measured on the contact surfaces, and the determination of the CoM of the target object (here we take the 1D CoM as an example) can be formulated as the problem of searching for a lifting position on the object, at which the conditions for equilibrium are satisfied.

A. Linear slip detection for force condition verification

Consider a target object that lying on the reference plane is grasped by the robotic gripper on its side faces and then slowly lifted up for a distance Δh. The applied lifting force (the force component in the gravitational direction) balances the object's weight and the other applied external forces (e.g. support force from the reference plane, if exists). When the object is lifted up and stays at a target height, it will not slip out of the gripper (linear slip does not happen on the contact surface), as long as the resultant force applied on the object is zero, i.e. the force condition is satisfied. The force condition should be satisfied to guarantee that the object can be stably lifted to a target height.

We verify the force condition by detecting linear slip of the object, which can be realized by measuring the increasing rate of lifting force on the contact point (Kaboli et al. 2016b). The lifting force f_L on one contact point is the component of the applied result force that decomposed in the Z direction of the WCF. A linear slip is detected as soon as the value of lifting force applied on the contact surface f_L has increased by a percentage ϵ within a short time period of Δt:

$$|f_L(t + \Delta t) - f_L(t)| > \epsilon \cdot |f_L(t)|. \tag{18}$$

The robotic gripper regulates its applied force by detecting linear slips during the lifting process, in order to satisfy the force condition. In application, the gripper first closes its fingers compliantly until all the fingertips are in light contact with the object's surface. Then the robot slowly lifts the object to a target height. If the object has slid out of the gripper during this process, or if a linear slip is detected after the object has been lifted to the target height, the grasping force is considered insufficient. Then the robot lays down the object, opens the gripper, re-grasps the object with an increased grasping force, and then lifts up the object again. The robot repeats this procedure until the grasped object can be lifted up to the target height and held stably (no linear slip is detected). Then the force condition is considered satisfied, and the robot proceeds to check the torque condition.

B. Object rotation detection for torque condition verification

If the applied resultant torque is not zero, the grasped object will rotate as it is being lifted up, i.e. rotational slip happens on the contact points. However, to the best of our knowledge, rotational slip on one single contact point can hardly be detected based solely on the force signal. Thus, in a two-contact-point grasp case, the rotation of the grasped object cannot be detected based on rotational slips on the contact area. We show that based on force signals, it is possible to detect the rotation of the lifted object with at least three contact points (denoted as A, B, and C), among which two contact points (e.g. B and C) are aligned on the same side of the grasped object and close to each other, while opposite to the other one (e.g. A).

We propose to detect the rotation of the object by measuring the similarity between frictions measured on different contact points during lifting.

The cross-correlation of two jointly stationary series $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as

$$\rho_{\boldsymbol{xy}} = \frac{\mathrm{cov}(\boldsymbol{x}, \boldsymbol{y})}{\sigma_{\boldsymbol{x}}\sigma_{\boldsymbol{y}}} \tag{19}$$

with $\mathrm{cov}(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}[(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{y} - \bar{\boldsymbol{y}})^{\mathrm{T}}]$ being the cross-covariance of $\boldsymbol{x}$ and $\boldsymbol{y}$, $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ being the vectors composed of expected values of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively; $\sigma_{\boldsymbol{x}}$ and $\sigma_{\boldsymbol{y}}$ to denote the standard deviation of $\boldsymbol{x}$ and $\boldsymbol{y}$. The cross-correlation $\rho_{\boldsymbol{xy}}$ is a normalized value within $[-1, 1]$, thus it can also be applied on objects of different textures and stiffness, even if the change of contact properties may result in different absolute values of the friction. The closer $\rho_{\boldsymbol{xy}}$ to 1, the higher the similarity between $\boldsymbol{x}$ and $\boldsymbol{y}$.

In this case, we focus on rigid objects, and we assume that during the lifting process, the three contact positions satisfy the symmetry property that B and C are symmetrical with respect to A, i.e. A, B, and C formulate an isosceles triangle. As a result, same lifting forces should be applied on B and C (or in other words B and C should have balanced the same linear frictions) if the object is lifted at its CoM. We represent the time series of frictions recorded on each contact points of A, B, and C as $\boldsymbol{f}_A$, $\boldsymbol{f}_B$, and $\boldsymbol{f}_C$, respectively. The lifting position is represented by position A. The torque condition is considered to be satisfied if ρ_{BC} is higher than an expected similarity level γ (see Fig. 5a):

$$\rho_{BC} \geq \gamma, \quad \gamma \in (0, 1]. \tag{20}$$

If both force and torque conditions are satisfied, the current lifting position can be estimated as the CoM.

C. CoM exploration

The CoM is a constant 3D position in the corresponding OCF. Here we explain the exploration of the 1D CoM component along the X-axis of the OCF (referred to as the exploratory axis) as an example.

For searching the 1D CoM component, we propose to use the binary search algorithm, which is the 1D optimal search algorithm with a computational complexity of $O(\log_2 N_s)$ for maximum N_s sampling points.

We denote the cross-correlation of $\boldsymbol{f}_A$ and $\boldsymbol{f}_B$ as ρ_{AB}, the cross-correlation of $\boldsymbol{f}_A$ and $\boldsymbol{f}_C$ as ρ_{AC}. At different side of the real CoM, ρ_{AC} and ρ_{AB} show the contrary numerical relationship (see Fig. 5b). It reveals the fact that between two adjacent contact points (B and C), the one that is closer to the real CoM balances larger friction. According to this relationship, the robot determines the exploratory range of the next lifting, thus the sequence of lifting positions is guaranteed to converge to the real CoM of the object. For example, if $\rho_{AB} > \rho_{AC}$, according to the binary search algorithm, the exploratory range of the next action is determined as one of the bisected ranges, which is closer to B while further away from C. The robot then lifts the object at the middle of this exploratory range.

D. CoM feature extraction

The length of the object along the exploratory axis can be segmented by the corresponding CoM component into two

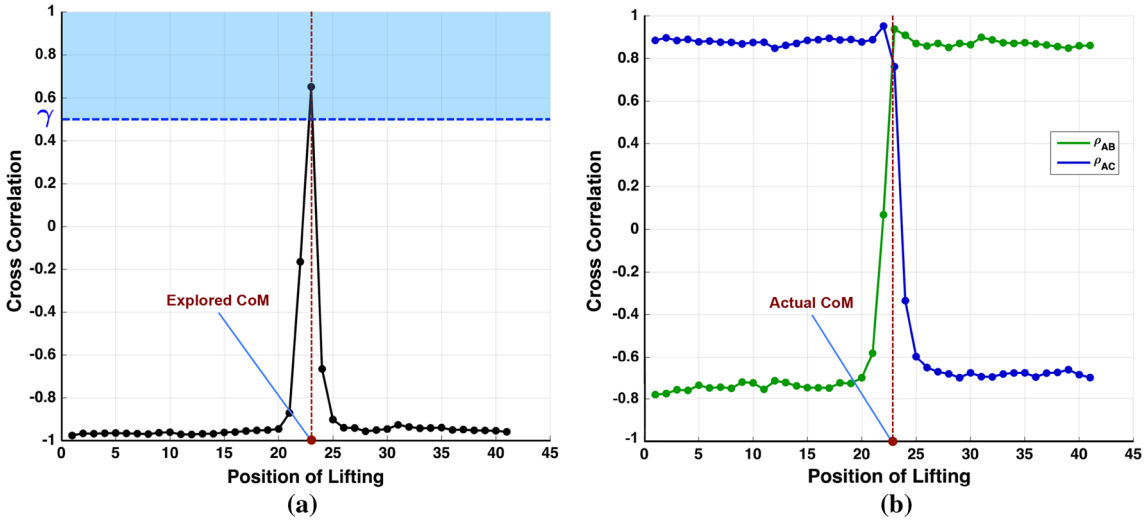


Fig. 5 Illustration of the cross-correlation between force signals sequences recorded during the lifting process at different lifting points. The target object was lifted up for a height $\Delta h = 30$ mm at 41 sequential positions along its length edge, and at each lifting position, the measured sequences of frictions from contact points A, B, and C, were recorded during the lifting process. The abscissa scale denotes the lifting positions sequentially distributed along the object, from one tail (1) to the other tail (41). The ordinate scale represents the calculated cross-correlation of signal sequences. **a** The cross-correlation ρ_{BC} is used to determine if the corresponding lifting position can be determined as the real CoM. γ represents the threshold of similarity, above which the current lifting point can be estimated as the CoM. **b** The relationship of ρ_{AB} and ρ_{AC} is used to guide the selection of the next lifting position

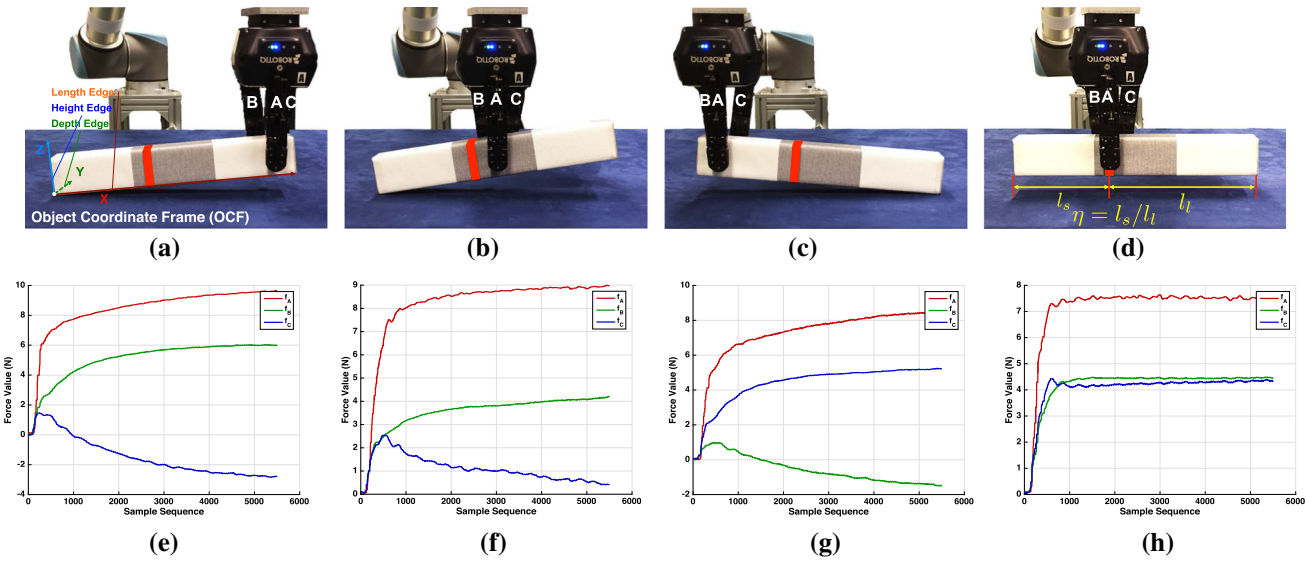


Fig. 6 The analysis of lifting forces at different lifting positions on the length edge of a target object. **a** The X, Y, and Z axes of the object coordinate frame (OCF) are defined along the length edge, depth edge, and height edge of the object, respectively. **a**–**d** The robot lifts object at different positions. The real CoM of the object is marked by the red ribbon in each figure. **e**–**h** The sequence of corresponding lifting force signals from each contact point during lifting process. If the object is lifted almost at its CoM (**d**), the frictions measured on the contact points on the same side are almost the same (**h**), owe to the positional symmetry of contact points B and C

parts. In order to extract the CoM as an object feature that is independent of the position and orientation of the object, we represent the CoM feature as a ratio of the shorter segment (indicated by the subscript s) to the longer part (indicated by the subscript l) (see Fig. 6d). For example, the CoM property of a 3D object can be formulated in the length-depth-height order:

$$\eta = (l_s/l_l,\ d_s/d_l,\ h_s/h_l). \tag{21}$$

Therefore, η is constant for rigid objects, and each component of η lies in the range of (0, 1].

In this paper, only the CoM component along the length edge (1D, along the X-axis of the OCF) is considered. To explore the CoM, the robot moves its end-effector above the centroid of the target object, adjusts its orientation, and lifts the object at determined positions sequentially along its length edge. The robot first regulates its grasping force to satisfy the force condition. As long as the object can be lifted and held stably without linear slip, the robot records the force signals while lifting the object for a distance Δh, and analyzes the cross-correlation of signal sequences to determine if the current lifting position can be estimated as the CoM. If not, the robot selects the next lifting position based on the relationship of signals (Sect. 5.1.3). In this work, the explored CoM feature η is denoted as one single scalar: $\eta = l_s/l_l$.

6 Active touch for object learning (ATOL)

In this section, we describe our proposed probabilistic method for active touch object learning method (ATOL). Our proposed algorithm enables robotic systems to efficiently learn about objects via their physical properties such as surface texture, stiffness, and center of mass properties and to correspondingly construct the observation models of the objects (see Fig. 2b).

6.1 Problem definition

Suppose the robot have explored the unknown workspace and found N objects $O = \{o_i\}_{i=1}^{N}$ and then determined their poses. Now, the robot is asked to learn about the objects via their physical properties. We denote the physical properties of objects by $\mathcal{K} = \{k_j\}_{i=j}^{K}$. These objects might have similar physical properties, for instance similar stiffness, while some might have quite different properties, for example different center of mass and /or texture.

In this situation, the robot's task is to efficiently learn about objects by means of their physical properties with as few training samples as possible and to efficiently construct the reliable observation models of the objects. Since the objects with the similar properties cannot be easily discriminated among each other, the robot should autonomously collect more training samples with these objects.

The active touch-based object learning problem (ATOL) is formulated as a standard supervised learning problem for multi-class classification, where each object o_i is considered as a class; for each physical property k_j, a probabilistic classifier is efficiently constructed by iteratively selecting the "next object to explore" and the "next physical property to learn", in order to collect the next training sample.

In ATOL algorithm the one versus all (OVA) Gaussian Process Classifier (GPC) is used to construct observation models of objects. In this case, the target (or label) set $\mathcal{Y}$ contains integers indicating the labels of input data, i.e. $\mathcal{Y} = \{1, 2, \ldots, N\}$, for N possible target classes in total. Each target label is mapped to a vector $\boldsymbol{v}_y \in \mathbb{R}^N$. In the vector $\boldsymbol{v}_y$, all entries are set to -1 except the yth entry which is set to 1. Then the function relation that maps the input data $\mathcal{X}$ into the classes $\mathcal{Y}$ is learned as: $\boldsymbol{f} : \mathcal{X} \mapsto \mathcal{Y}$.

GPC estimates the probability of each target label $p(y|\tilde{\boldsymbol{x}})$ for a test data point $\tilde{\boldsymbol{x}}$ by $\boldsymbol{f}(\tilde{\boldsymbol{x}})$, and then assigns it to the class with the largest predicted probability:

$$\tilde{y} = \arg\max_{y \in \mathcal{Y}} \ \boldsymbol{f}(\tilde{\boldsymbol{x}}) \tag{22}$$

In this study, we used the RBF as kernel function Eq. (40) in Sect. 11, and the hyper-parameters are selected by cross-validation.

6.2 Methodology

6.2.1 One-shot tactile data collection

To start learning about objects via their physical properties, the robot first constructs a small set of training data $\mathcal{S} = \{\mathcal{S}_{k_j}\}_{j=1}^{K}$ by executing each of the three actions $\mathcal{A} = \{a_{k_j}\}_{j=1}^{K}$ once on each object (One-shot tactile data collection), in order to perceive the object physical property denoted as $k_j \in$ {texture, stiffness, center of mass }, $K = 3$ and $a_{k_j} \in$ { sliding, pressing, lifting }.

Then the autonomous robot iteratively collects new training samples $\mathcal{S}_{k_j}$. At each iteration, ATOL algorithm updates GPCs with the training data set collected hitherto, and estimates the uncertainty in the constructed observation models which guide to next round of tactile data collection.

6.2.2 Objects' uncertainty estimation

In order to estimates the uncertainty in the objects' observation models the ATOL measures the Shannon entropy of each training samples. In this regard, the training dataset of one physical property $\mathcal{S}_k$ is divided into categories $\mathcal{S}_k = \{\mathcal{S}_{o_i}^k\}_{i=1}^{N}$, where each category $\mathcal{S}_{o_i}^k$ has M_i^k number of samples. For each set of training samples, the mean value of the Shannon entropy is measured:

$$\mathcal{H}(o_i, k) = \frac{1}{M_i^k} \sum_{\boldsymbol{s}_{o_i}^k \in \mathcal{S}_{o_i}^k} \mathcal{H}(\boldsymbol{s}_{o_i}^k) \tag{23}$$

$$\mathcal{H}(\boldsymbol{s}_{o_i}^k) = -\sum_{o \in O} p\left(o|s_{o_i}^k\right) \log(p(o|s_{o_i}^k)) \tag{24}$$

with the $p(o|s^k_{o_i})$ being the observation probability predicted by the GPC model. The higher the $\mathcal{H}(o_i, k)$ is, the more uncertain the robot is about the object.

6.2.3 Next object and next physical property selection

We define the object-property pair, $\Phi(o_i, k)$ as a function of the object $O = \{o_i\}_{i=1}^N$ and the physical property k. After selecting $\Phi(o_i, k)$, the robot moves to the object o_i and executes the action a_k to perceive the physical property k. In order to reduce the entropy of the observation models as quickly as possible, the next training sample is generated from the pair $\Phi(o_i, k)$ with the largest entropy. In order to learn about objects efficiently, the robot can greedily sample the next object and the next property which maximize $\mathcal{H}(o_i, k_j)$ of GPCs (exploitation). In this way, the robot autonomously collects more training samples from the objects based on their physical properties which are easily confused. At the end of each iteration, the new training sample will be added to the entire dataset $\mathcal{S} := \mathcal{S} \bigcup s^*$.

The active learning process is repeated until a target criterion is reached, in our case, when there is no perceived reduction of the entropy for the observation models, or the robot collects a certain number of training samples. In order to avoid being trapped in the local maxima, we add an exploration rate so that the robot can randomly select $\Phi(o, k)$ by following the uniform distribution (exploration). We denote p_Φ as a probability, which is uniformly generated with $\mathcal{U}(0, 1)$ at each iteration in the ATOL. Then the next object o^* and next physical property k^* is determined by:

$$\Phi^*(o,k) = \begin{cases} \underset{o_i \in O, k_j \in K}{\text{argmax}} \ \mathcal{H}(o_i, k_j), & \text{if } p_\Phi > \epsilon_\Phi \\ o = \mathcal{U}\{o_1, o_2, \ldots o_N\}, \ k = \mathcal{U}\{k_1, k_2, k_3\}, & \text{o.w.} \end{cases} \tag{25}$$

In Eq. (25), the parameter ϵ_Φ controls the exploration–exploitation trade-off.

7 Active touch for object recognition (ATOR)

Assuming the observation models with the efficient training dataset are constructed during the active learning process (see Fig. 2b), the autonomous robot is faced with the task of recognizing objects in an unknown workspace (see Fig. 2c). The task is divided into two scenarios. In the first scenario (see Fig. 2c-1), the robot is asked to discriminate among objects which have already been learned. However, in this scenario, each object can have various orientations and positions in the workspace. In second scenario (see Fig. 2c-2), the unknown workspace includes both known and unknown objects with different orientations and locations: "known objects" are the objects about which the robotic system has learned before via (ATOL); "unknown objects" are the objects the robot has not been encountered. The task of the robot is to search for a known object or objects in this workspace.

Algorithm 2 Active touch for object learning

Require: $O = \{o_i\}_{i=1}^N$ ▷ N objects to learn, each is regarded as a class
Require: $\mathcal{L} = \{l_i\}_{i=1}^N$ ▷ The locations of the objects
Require: K ▷ object physical properties
initialization: $\mathcal{S}$ ▷ one-shot tactile data collection
initialization: GPCs $f : \mathcal{S} \mapsto O$ ▷ Gaussian Process Classifiers. Sect. 6.2
for $r = 1 : R$ **do**
  $p(o|\boldsymbol{s}) \leftarrow \boldsymbol{f}(\boldsymbol{s})$ ▷ class predictions for training data
  $E[\mathcal{H}_{t+1}(k)] \leftarrow \sum_{o \in O} p(o)\mathcal{H}_{t+1}(o, k)$ ▷ object uncertainty, Sect. 6.2.3
  $\Phi^*(o, k) \leftarrow \underset{o_i \in O, k \in K}{\text{argmax}} \ E[\mathcal{H}_{t+1}(k)]$ ▷ Next object and next physical property selection. Sect. 6.2.3
  $Move\_robot(l_i)$ ▷ move the robot to the object o_i
  $Execute\_action(a_k)$ ▷ Sect. 5.1
  $\mathcal{S} \leftarrow \mathcal{S} \bigcup s^*$ ▷ update training dataset with new samples
  GPCs $\leftarrow \mathcal{S}$ ▷ update the observation models
return GPCs, $\mathcal{S}$ ▷ observation models, training dataset,

7.1 Active touch for object discrimination

7.1.1 Problem definition

The task of the robot is to perform a sequence of exploratory actions ($\mathcal{A} = \{a_k\}_{k=1}^K$) to efficiently discriminate among objects which have already been learned. However, the objects can have different positions and orientations in the unknown workspace. Therefore, using the proposed active touch workspace exploration method the robot first localize the objects in the workspace (see Sect. 4). Then the robot exploits the objects' prior knowledge that efficiently obtained by our proposed active object learning strategy (the observation models and training dataset of objects) (see Fig. 2b), in order to iteratively execute exploratory actions on objects. In this part of study, we propose a method to enable the robotic system to determine the most informative exploratory action at each step, such that the objects can be distinguished with the fewest exploratory actions possible (see Fig. 2c-1).

7.2 Methodology

The object discrimination task is achieved through sequentially executing the exploratory actions on the objects. Firstly, the object's belief $p(o)$ is initialized as being uniformly distributed. Next, an exploratory action is executed on the object in order to perceive an observation z_t. Then, the object's belief is updated which helps to determine the next exploratory movement a^*. This process is repeated until a

target criterion is reached: for example, until the maximum a posteriori (MAP) exceeds a probability threshold, or the maximum times to update the procedure is reached.

7.2.1 Objects' belief updating

Once an action a_k has been performed and the corresponding observation z_t is obtained at time step t, the object posterior distribution can be updated using Bayes' rule:

$$p_t(o|z_t) \propto p(z_t|o)p_{t-1}(o) \tag{26}$$

with $p_{t-1}(o)$ being the posterior distribution from the previous time step, and $p(z_t|o)$ being the observation probability calculated by the observation models.

7.2.2 Next optimal exploratory action selection

When selecting which exploratory action is optimal to recognize objects, we need to predict the benefit of the movement based on the updated object priors $p(o)$ and the prior knowledge (observation models and training dataset). In this work, we propose a method to estimate the expected benefit of a movement, which guides the next action selection called confusion matrix-based uncertainty minimization (CMUM). Our proposed method predicts the benefit of an exploratory action by inferring the resulting confusion between objects. If a movement produces tactile information which is most easily discriminated among objects, then objects can be recognized more quickly by executing such a exploratory action. Conversely, exploratory actions which generates confused observations are not helpful. Therefore, the advantage of selecting a particular exploratory action can be inferred by how much confusion the action results in. To do this, we measure the confusion of an exploratory action by calculating the objects' similarity, and use it to guide the next action selection. Similar work has been done by Fishel et al (Fishel and Loeb 2012). However, their method suffered from the curse of dimensionality and their method could only be tractable with low-dimensional features. In contrast, our proposed method is unrestricted by the feature dimensions, and thus can be applied to high dimensional features, such as surface texture property.

7.2.3 Proposed confusion matrix-based uncertainty minimization (CMUM)

When predicting the confusion c_{ij}^k between objects o_i and o_j for the tactile property k, we calculated the observation probability $p(o_j|\boldsymbol{s}_{o_i}^k)$ for each training sample, which belongs to the object o_i, but is misclassified to the object o_j. Then c_{ij}^k is estimated by the average value of $p(o_j|\boldsymbol{s}_{o_i}^k)$:

Algorithm 3 Active touch for object discrimination

Require: $\mathcal{L}_\mathcal{M} = \{l_m\}_{m=1}^M$ ▷ The locations of M objects in the workspace
Require: GPCs, $\mathcal{S} = \{\mathcal{S}_n\}_{n=1}^N$ ▷ observation models, training data for N classes of objects
for $m = 1 : M$ **do**
  $p_0(o) \leftarrow \mathcal{U}$ ▷ initialize object priors
  $Move\_robot(l_m)$ ▷ move the robot to another object
  while $\nexists n | (p(o_n|z_t^k) > \beta)$ **do**
    $a^* \leftarrow Select\_action(p(o))$ ▷ Sect. 7.2.2
    $Execute\_action(a^*)$
    $p_t(o|z_t) \propto p(z_t|o)p_{t-1}(o)$ ▷ get observation z_t, update object priors: Sect. 7.2.1
  $o_m \leftarrow \underset{o_n \in \mathcal{O}}{\operatorname{argmax}}(p(o))$ ▷ object identified
return $o_{1:M}$

$$c_{ij}^k = \frac{1}{M_i^k} \sum_{\boldsymbol{s}_{o_i}^k \in \mathcal{S}_{o_i}^k} p(o_j|\boldsymbol{s}_{o_i}^k) \tag{27}$$

with M_i^k being the number of training data for object o_i and tactile property k. c_{ij}^k ranges between 0 and 1, where 0 refers to no confusion, and 1 means total confusion.

After obtaining a new observation z_t^k at time step t, the expected confusion $u_{o_i,k}$ between the object o_i and the others is measured:

$$u_{o_i,k} = \frac{\sum_{o_j \in \mathcal{O}, o_j \neq o_i} p(o_j|z_t^k) c_{ij}^k}{\sum_{o_j \in \mathcal{O}} p(o_j|z_t^k) c_{ij}^k}. \tag{28}$$

The expected confusion U_k for property k can be estimated by considering all objects:

$$U_k = \sum_{o \in \mathcal{O}} p(o|z_t^k) u_{o,k}. \tag{29}$$

This value predicts the confusion between objects after executing an exploratory movement. In other words, it measures the expected uncertainty of an action. The next action a^* is selected in order to bring the maximum benefit. In our case, this means minimizing the expected uncertainty:

$$a^* = \underset{k}{\operatorname{argmin}} \, (U_k)^\beta \tag{30}$$

where the discount factor β is used to control the exploration–exploitation trade-off. It is inversely proportional to the number of times an action has been taken.

7.3 Active touch for target object search

The task of the robot is to recognize a target object/s in the unknown workspace includes both known and unknown objects with different orientations and locations. in this scenario the robot should efficiently search for a object in this

workspace. Different from the object discrimination task in which all objects in the workspace should be distinguished (see Fig. 2c-1), in the problem of target object search (see Fig. 2c-2), the robot only needs to recognize the target objects. To do this, we divided the objects which are explored in the active learning into categories of target object and non-target objects $O = O_{tg} \bigcup O_{non\text{-}tg}$, and divide the training dataset correspondingly $\mathcal{S} = \mathcal{S}_{tg} \bigcup \mathcal{S}_{non\text{-}tg}$. Then the multi-class GPCs are reduced to binary classifiers which give the observation probability $p(o \in O_{tg}|\boldsymbol{z})$.

$$p_t(o_{tg}|\boldsymbol{z}_t) \propto p(\boldsymbol{z}_t|o_{tg})p_{t-1}(o_{tg}) \tag{31}$$

in which $p_t(o_{tg}|\boldsymbol{z}_t)$ being the posterior distribution from the previous time step, and $p(\boldsymbol{z}_t|o_{tg})$ being the observation probability calculated by the observation models. The optimal exploratory action is selected by the robot following the procedure described in Sect. 7.2.2. The similarity between object pares is calculated by our proposed CMUM method explained in Sect. 7.2.3. The Algorithm 4 show the proceeder of our proposed active target search more in detail.

Algorithm 4 Active touch for target object search

Require: $\mathcal{L}_{\mathcal{J}} = \{l_j\}_{j=1}^{J}$ ▷ The locations of J objects in the workspace
Require: $o_{tg} \in O$ ▷ define which object to find
1: initialization: $\mathcal{S} = \{\mathcal{S}_{tg}, \mathcal{S}_{non-tg}\}$ ▷ divide training data
2: initialization: binary GPCs ▷ Sect. 7.2.1
3: $p(o_{tg}) \leftarrow \frac{1}{2}$ ▷ initialize target object priors
4: **for** $j = 1 : J$ **do**
5: $Move\_robot(l_j)$
6: **while** $p(o_{tg}) > \tau_1$ *or* $p(o_{tg}) < \tau_2$ **do**
7: $a^* \leftarrow Select\_action(p(o_{tg}))$ ▷ select next exploratory movement
8: $Execute\_action(a^*)$
9: $p_t(o_{tg}|\boldsymbol{z}_t) \propto p(\boldsymbol{z}_t|o_{tg})p_{t-1}(o_{tg})$ ▷ update target object priors
10: **if** $p(o_{tg}|\boldsymbol{z}_t^k) > \gamma_1$ **then**
11: $o_j \leftarrow o_{tg}$ ▷ target object found
12: $l_{tg} \leftarrow l_j$
13: break
14: **if** $p(o_{tg}|\boldsymbol{z}_t^k) < \gamma_2$ **then**
15: continue ▷ leave the robot to another object
return l_{tg}

7.4 Baseline: expected entropy reduction (EER)

We used expected entropy reduction (EER) method as a baseline to compare with our proposed CMUM method. The EER is an approach for estimating the expected benefit of a exploratory action by predicting its entropy (Rebguns et al. 2011; Schneider et al. 2009). The exploratory action which produces lower entropy can better discriminate among objects. To do the comparison, we measured the expected entropy reduction for different action to perceive different physical properties of an object.

Let us denote $\mathcal{H}_{t+1}(k)$ the entropy at the next time step $t+1$ from the action a_k taken to obtain an observation $\boldsymbol{z}_{t+1}^k$, where k refers to the object property. We measure $\mathcal{H}_{t+1}(k)$ by:

$$\mathcal{H}_{t+1}(k) = -\sum_{o \in O} p_{t+1}(o|\boldsymbol{z}_{t+1}^k) \log(p_{t+1}(o|\boldsymbol{z}_{t+1}^k)). \tag{32}$$

Since we do not know which measurements $\boldsymbol{z}_{t+1}^k$ the robot will obtain at time $t+1$, we need to integrate all possible observations. This is approximated through summing up all the samples in the training dataset $\mathcal{S}_k$ for tactile property k, weighted by the object priors $p(o)$:

$$E[\mathcal{H}_{t+1}(k)] = \sum_{o \in O} p(o)\mathcal{H}_{t+1}(o, k) \tag{33}$$

where $\mathcal{H}_{t+1}(o, k)$ is the mean value of the entropy for an object o:

$$\mathcal{H}_{t+1}(o_i, k) = \frac{1}{M_i^k} \sum_{\boldsymbol{s}_{o_i}^k \in \mathcal{S}_{o_i}^k} \mathcal{H}_{t+1}(o_i, k|\boldsymbol{s}_{o_i}^k) \tag{34}$$

$$= -\frac{1}{M_i^k} \sum_{\boldsymbol{s}_{o_i}^k \in \mathcal{S}_{o_i}^k} \sum_{o \in O} p_{t+1}(o|\boldsymbol{s}_{o_i}^k) \log(p_{t+1}(o|\boldsymbol{s}_{o_i}^k) \tag{35}$$

with M_i^k being the number of training data for object o_i and tactile feature k. $p_{t+1}(o|\boldsymbol{s}_{o_i}^k)$ is the object posterior at $t+1$, updated by the training sample $\boldsymbol{s}_{o_i}^k$. Actions have more benefit when the expected entropy is minimized:

$$a^* = \underset{k}{\text{argmin}}\ E[\mathcal{H}_{t+1}(k)]. \tag{36}$$

8 Experimental results

To evaluate the performance of our proposed framework in real time, as well as experimentally validate the efficiency of the suggested approaches for active object learning and active object recognition in an unknown workspace, the robotic system performed experiments in three different scenarios.

At the beginning of all scenarios, the robot did not have any prior knowledge about the location, orientation, and the number of objects. Thus it is necessary for the robot to first explore the entire workspace to gather information about target objects located inside. After exploration, the robot was able to address each object in the workspace and to perform different tasks.

The first task of the robot was to actively and autonomously learn the physical properties of experimental objects in the

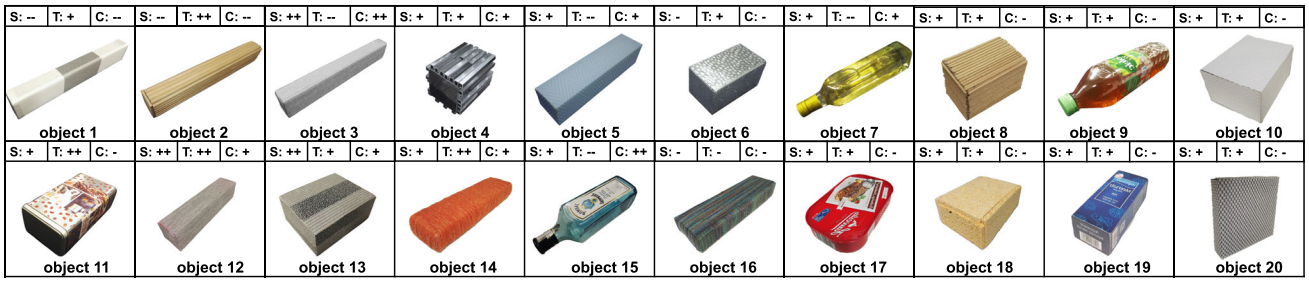


Fig. 7 Experimental objects. The physical properties are evaluated subjectively by human subjects and are indicated in the panel to the upper right of each object (S: stiffness, very soft (- -) and very hard (++) T: roughness of surface textures, very smooth (- -) and very rough (++) C: center of mass, far from the centroid (- -), close to the centroid (++)

workspace, i.e. their stiffness, textural properties, and CoM (see Fig. 8).

In the second scenario (see Fig. 14), the task of the robot was to efficiently discriminate among the objects, taking advantage of the knowledge of objects that was learned in the previous scenario. The objects had different locations and orientations in the workspace to the first scenario.

In the last scenario (see Fig. 16), the robot was asked to search for a specified target object in an unknown workspace that contains objects, some of which were already learned by the robot previously, and some were not (new objects).

In these experiments, the robotic system, i.e. the UR10 robotic arm, the Robotiq gripper, and the OptoForce sensors, was controlled in the framework of ROS. Tactile signals were sampled at a frequency of 333 Hz, and the gripper was controlled at 50 Hz.

8.1 Properties of experimental objects

In order to evaluate the performance of our proposed framework, we deliberately selected 20 objects (see Fig. 7), made of various materials, such as wood, glass, metal, and plastic. The physical properties of these experimental objects vary from relatively similar to quite different. Since the focus of this work is object recognition via surface texture, stiffness, and CoM, the geometrical properties of the objects are out of our scope. Due to the constraints from our hardware (e.g. size of sensor, width and length of robotic fingers), we selected cuboids and objects of bar shape, so that these constraints can be satisfied.

8.2 Active touch for object learning in unknown workspace

In the first scenario (see Fig. 8(WS-1)), the robot started with the active exploration of the unknown workspace to determine the number of objects, as well as their positions, sizes, and orientations. After this, it actively learned about each object using our proposed approach.

8.2.1 Active touch for exploration of unknown workspace

The unknown workspace (see Fig. 3) is a cuboid with the size of 1000 mm(Length) $\times$ 640 mm (Width) $\times$ 100 mm (Height), and the world coordinate frame is defined along its width edge (X), length edge (Y), and height edge (Z). Five objects were selected randomly at uniform from object list in Fig. 7 and put in the workspace.[2] Starting from the origin position, the workspace is discretized by a step size of 40 mm in both X and Y directions, and 15 mm in Z direction, according to the width (40 mm) of the finger and the distance from fingertip to the center of sensor (15 mm). Therefore, the allowed number of sampling points is 25 along the Y axis, 16 along the X axis, and 4 in Z axis, thus the maximum number of sampling for the entire workspace of all four directions counts up to 328 (the total number of mesh grids on four start planes of the workspace). The robot performed exploration clockwise around the workspace, i.e. from $\mathbb{W}_{d_1}$ to $\mathbb{W}_{d_4}$ in sequence.

A. Tactile point cloud construction

We take the exploration of $\mathbb{W}_{d_4}$ as an example, as explained in Sect. 4.1.1. During the exploration process, fingers of the gripper were controlled individually and only one finger was stretched out for exploration. The gripper first stretched out finger A while maintained finger B and finger C closed, and then the UR10 moved its end-effector (the gripper) to the start position on the start plane with the tactile sensor (the fingertip) orienting the d_4 direction, and then started exploration. A contact was detected as soon as the resultant force (Eq. 2) measured on the fingertip of A exceeds $\delta = 0.5\ N$, then the current position of the finger A was returned as an observation point. The observation point is added into the training dataset of the GPR $\boldsymbol{g}_{d_4}$. If this light contact is detected before

[2] Due to the constraints of the workspace, it is difficult for the UR10 robot to explore more than five objects. Therefore, five out of 20 objects were selected randomly at uniform for the evaluation.

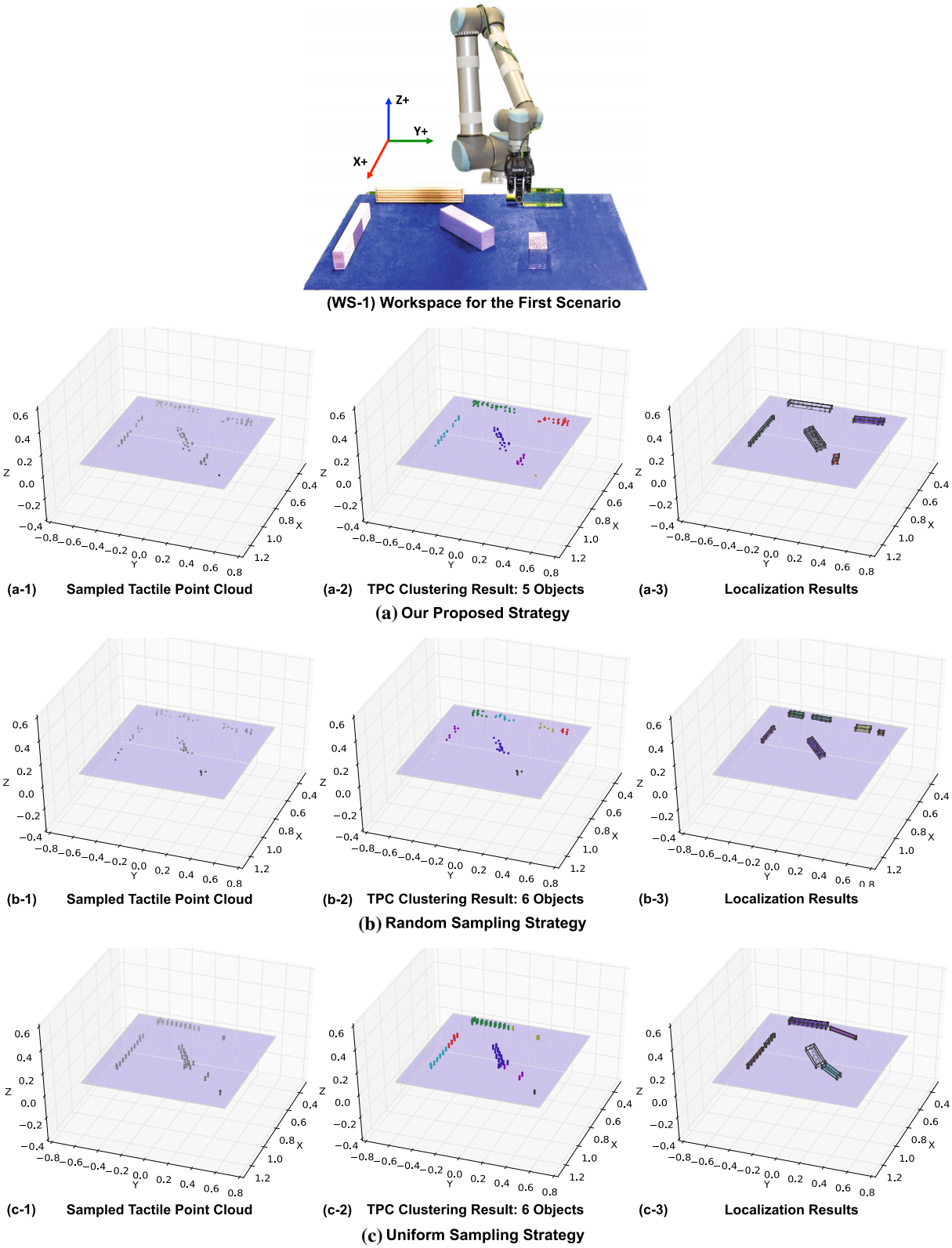


Fig. 8 The active exploration results of the unknown workspace in the first scenario (see Sect. 8.2.2). (WS-1) The layout of the workspace in the first scenario. From left to right, each one of the three sub-figures aligned in a row illustrates the constructed TPC, clustering result of TPC, and result of object localization. **a** The active exploration results of the unknown workspace by applying our proposed strategy. **b** The active exploration results of the unknown workspace by applying random sampling strategy. **c** The active exploration results of the unknown workspace by applying uniform sampling strategy

the robot reached the target point, it is also added to the TPC dataset $\mathcal{T}_{d_4}$.

In order to initialize the GPR model, the robot first uniformly sampled 6($M = 3,\ N = 2$) equally spaced points in total on the start plane as training dataset Ω_{d_4}. After the GPR model was trained, the robot selected the next sampling position according to the predicted variance. The robot continued sampling until the stop criteria is satisfied, which is $\sum_{\boldsymbol{p}\in\mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p})) < \tau$. Then it started the exploration in the next directions. The entire TPC dataset $\mathcal{T}_{\mathbb{W}} = \bigcup_i \mathcal{T}_{d_i}, \mathcal{I}_{DIR} = \{1, 2, 3, 4\}$ was fully constructed (see Fig. 8a-1) after the exploration of the entire workspace was completed.

B. Baseline strategies for comparison

In order to evaluate our proposed active workspace exploration strategy, we selected the uniform sampling strategy and the random sampling strategy as baselines for evaluating the performance. For both baseline strategies, the robot sampled exactly the same initialization dataset as for the active exploration strategy at the beginning of the exploration.

Following the uniform sampling strategy, the robot started from one corner of the start plane, and then sampled over all of the start points on the start plane column-wise. For example, the robot started from one corner (e.g. $(\underline{x}, \underline{z})$) of the start plane of $\mathbb{W}_{d_4}$. The robot sampled from $(x_i, \underline{z})$ to $(x_i, \overline{z})$, and then moved horizontally to the next column (start with $(x_{i+1}, \underline{z})$).

When applying the random sampling strategy, all of the points $\boldsymbol{p} \in \mathbb{X}$ on the start plane have the same probability to be selected, and the robot arbitrarily chose a start point on the start plane, and then executed the translational movement.

Since by following our proposed active exploration strategy, the average value of required sample steps for satisfying the stop criteria ($\sum_{\boldsymbol{p}\in\mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p})) < \tau$) is 60 for $\mathbb{W}_{d_1}$, $\mathbb{W}_{d_3}$ and 40 for $\mathbb{W}_{d_2}$, $\mathbb{W}_{d_4}$, we set these number of sample steps as the stop criteria for both uniform sampling and random sampling in the experiment.

The constructed TPC by following the random and uniform strategies are plotted in Fig. 8b-1 and c-1, respectively.

C. Statistical evaluation of exploration strategies

To statistically compare the performance of three different strategies, the robot explored the unknown workspace in total 30 different scenarios.

For each strategy, a GPR model with the same parameters is trained and updated after each observation point is obtained. When applying the active exploration strategy, the GPR model is used to select the next sample position, as well as measure the uncertainty of the workspace (total variance); while applying uniform and random strategies, the GPR models are trained only to calculate the uncertainty (total variance) of the workspace after each sampling.

In each scenario and for every strategy, the robot first sampled 6 positions ($M = 3,\ N = 2$) for $\mathbb{W}_{d_1}$, $\mathbb{W}_{d_3}$ (4 for $\mathbb{W}_{d_2}$, $\mathbb{W}_{d_4}$, $M = 2,\ N = 2$) to initialize the GPR model at the beginning. Then the robot sampled 60 steps for $\mathbb{W}_{d_1}$, $\mathbb{W}_{d_3}$ and 40 steps for $\mathbb{W}_{d_2}$, $\mathbb{W}_{d_4}$, and recorded the value of the total variance predicted by the trained GPR model after each sample step. A small total variance $\sum_{\boldsymbol{p}\in\mathbb{X}_{d_i}} Var(\boldsymbol{g}_{d_i}(\boldsymbol{p}))$ indicates that the GPR model, which is trained with the dataset sampled so far, can accurately describe the workspace; in other words, the exploration strategy is data-efficient.

The statistical comparison of the results is illustrated in Fig. 9. Since the exploration processes in each direction are independent, here we only compared the exploration performance of $\mathbb{W}_{d_1}$. The result shows that for all the strategies, the uncertainty of the workspace reduces as the number of samples increases. At each sample step, the workspace has the minimal total variance by following the active exploration strategy, and its uncertainty reduces faster than either uniform or random strategy, indicating that the workspace can be much more efficiently sampled by applying the proposed active exploration strategy than the baseline strategies.

D. Object localization and mapping

The sampled TPC in each scenario is clustered using the mean-shift approach (see Fig. 8a-2, b-2, c-2). Clusters with less than 5 data points were considered noise points and therefore discarded. After this, the minimum bounding box was calculated for each cluster to estimate the location, orientation, and geometric center of each object (see Fig. 8a-3, b-3, c-3). As Fig. 8 shows, all of the experimental objects were successfully clustered (see Fig. 8a-2) and correctly localized (see Fig. 8a-3) by employing our proposed strategy. However, by applying random sampling strategy, some regions of the workspace were not sufficiently explored, thus the TPC was incorrectly clustered into 6 objects (see Fig. 8b-2) and the estimated geometric information of the objects was fallacious as a result (see Fig. 8b-3). While using the uniform sampling approach, because of the constraints of sample steps, the robot was not able to complete the exploration. Therefore, only part of the workspace was fully explored. As a result, the TPC dataset was not complete (see Fig. 8c-2) and objects were not able to be correctly localized (see Fig. 8c-3).

We evaluated the performance of mean-shift clustering approach in all 30 scenarios based on the normalized mutual information (NMI). Table 1 shows the NMI values of the clustering result in each scenario. The average NMI of all 30 scenario is 0.92.

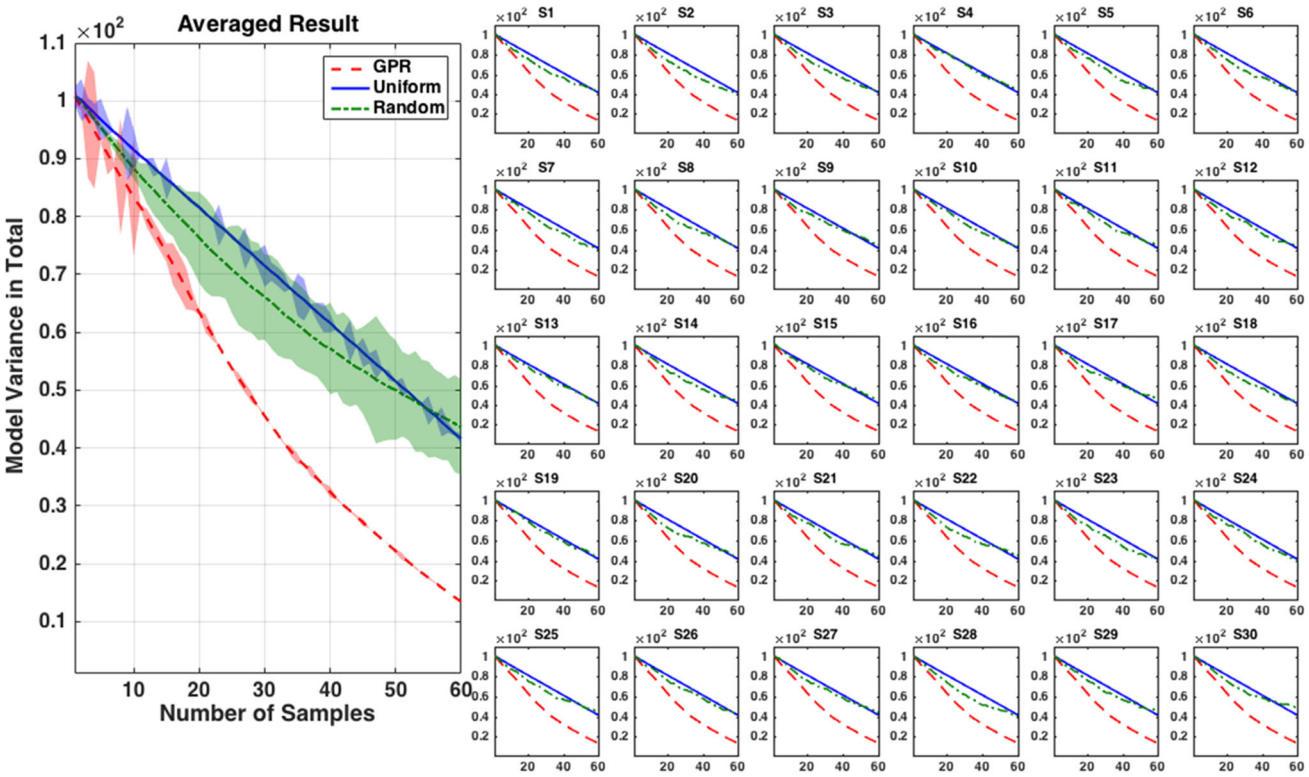


Fig. 9 The statistical comparison of the performance of the proposed active exploration strategy (GPR), uniform strategy, and random strategy for exploring the unknown workspace. Each small sub-figure on the right side named S1–S30 corresponds with one experimental scenario and illustrates the change of total variance in one exploratory direction (here we take $X+$ direction as an example) as the number of samples increases, by applying different exploratory strategies. The large sub-figure on the left side shows the averaged total variance over all the 30 scenarios with the shadowed area denoting the standard deviation

Table 1 Evaluation of clustering performance based on the normalized mutual information (NMI)

Group	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
NMI	1.00	0.94	0.84	0.80	0.86	0.93	0.96	1.00	0.90	0.91	0.87	1.00	1.00	0.91	0.82
Group	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
NMI	1.00	0.85	0.87	1.00	0.95	1.00	0.85	0.91	0.83	0.88	0.90	1.00	1.00	0.93	0.90

In 9 out of 30 scenarios, the TPC data were perfectly clustered ($NMI = 1.00$). In 16 out of 30 scenarios, the sampled data points were clustered ($NMI < 1.00$) followed by a successful localization and mapping of the experimental objects. Although some points were wrongly clustered, due to either the noise data or the connection of adjacent objects; however, by filtering out the noise clusters and constructing the bounding boxes, the localization and mapping results were acceptable for the robot to execute the subsequent tasks. The clustering failed in the remaining 5 scenarios, either because a large part of the TPC dataset were wrongly clustered, or the number of clusters does not match the number of real objects (object number is wrongly estimated). The reason is, multiple experimental objects were densely placed in these scenarios, some of them even connected to each other, thus these objects are occluded and cannot be fully explored.

The active exploration process of the unknown workspace was carried out at the beginning of each scenario, and the obtained information of objects was used for the subsequent procedures.

8.2.2 Evaluation of active tactile object learning (ATOL)

A. Test data collection

The performance of our proposed active object learning (see Fig. 2) was evaluated with a test dataset. The dataset was collected by the robot autonomously, by performing three exploratory actions (pressing, sliding, and lifting) on 20 experimental objects (Fig. 7). The data collection procedure was repeated 20 times for each object and each exploratory action. During executing all exploratory actions, the gripper

is controlled in "pinch" mode, i.e. finger B and finger C of the gripper were arranged next to each other and are controlled to have the same positions. Finger B and finger C move simultaneously and are opposite to the moving direction of finger A. In this configuration, these three contact positions (fingertips) form an isosceles triangle with B and C symmetric with respect to A.

For pressing action, the robot first moved to the object (i.e. let the geometric center of three fingers coincides the geometric center of the target object), it then closed all fingers for $N_\epsilon = 3$ extra position counts after a light contact of $f_t = 0.5\ N$ on each fingertip. When sliding on the surface of objects, the robot slid for 30 mm vertically after contacting the object (light contact force: $f_t = 0.5\ N$). In order to lift the object, the robot exerted an initial contact force of 0.5 N to grasp the object and then lifted it up for $\Delta h = 30$ mm. A linear slip was detected, if the tangential force had increased more than 25% within $\Delta t = 1$ s after the object being lifted to the target height. The expected similarity level for the CoM was set as $\gamma = 0.9$; however, considering the time consumption, the process of exploring the CoM would also be terminated if the distance between two successive lifting positions was less than 0.5 mm.

B. Baselines

The performance of our proposed active learning strategy (ATOL) was evaluated with both random and uniform sampling strategies.

Random learning strategy While applying the random learning strategy, both of the next object $o = \mathcal{U}\{o_1, o_2, \ldots o_N\}$ and the next physical property $k = \mathcal{U}\{k_1, k_2, k_3\}$ subject to uniform distribution ($o \sim \mathcal{U}(1, N)$ and $k \sim \mathcal{U}(1, 3)$), i.e. all of the os (and ks) have the same probability to be selected. The robot arbitrarily determines the next object $o \in \{o_1, o_2, \ldots o_N\}$ and the next physical property to learn $k \in$ {surface texture, stiffness, center of mass}.

Uniform learning strategy Using the uniform learning approach, at each round of the exploration, the robot learned about all three physical properties of each object in the workspace. In other words, the robot moved to each of five object $\{o_1, o_2, \ldots, o_N\}$ and executed all three exploratory actions {a_1: sliding, a_2: pressing, a_3: lifting } on each object in order to learn about {k_1: texture, k_2: stiffness, k_3: center of mass}.

C. Evaluation of active tactile object learning via all physical properties

In this scenario, the task of the robot was to learn five objects in the workspace based on their physical properties (stiffness, surface textures, CoM). To initialize the active learning process, the robot collected small training samples by performing each of three exploratory actions once on each object. Each step when the robot sampled a new training instance, the recognition accuracy of GPCs was measured with the test dataset.

Figure 10 illustrates the distributions of the tactile features extracted from the eight objects (as an example) in test dataset. On the one hand, depending on the physical properties, objects have different degrees of confusion. For instance, Fig. 10b shows that although some objects have similar surface structures, they can be discriminated by their textural property, thanks to our proposed robust tactile descriptors. In contrast, it is difficult to distinguish objects using stiffness, because the stiffness of the objects are very similar (see Fig. 10a). On the other hand, for the same physical property, objects' confusion are different from each other. For example, Fig. 10c clearly shows that objects 1, 2, 3, 4, and 8 can be easily recognized via their CoM, whereas objects 5, 6, and 7 are confused with each other.

At each object learning round 5 objects $\{o_1, o_2, \ldots, o_5\}$ were randomly selected out of 20 objects $O_{1:20}$ (see Fig. 7). Then, the robot learned about objects via their physical properties. In order to have a fair comparison between our ATOL method and the baseline learning strategies the robot executed 45 exploratory actions in total during learning process. This process is repeated 30 time for 30 groups of objects, each group is repeated 5 times.

Figure 11 shows the robot's learning progress measured by the classification accuracy on the test dataset each of 30 groups of experiments as well as the classification accuracy averaged over all 30 groups. Figure 11 demonstrate that ATOL consistently outperforms the baseline methods by obtaining the higher recognition accuracy while performing fewer exploratory actions. For instance, the robot obtained in average more than 81% recognition accuracy when it performed 20 exploratory actions However, using random and uniform strategies the robot achieved 71% recognition rate with the same number of actions. Obviously, the active learner learns about the object more quickly than uniform and random sampling strategies.

Apart from a numerical evaluation of the performance of the proposed method, we also investigated the learning process and decision of the strategy over time. Figure 12 demonstrates one exemplifying result of the learning progress following three aforementioned strategies to select next object and next physical property. The bottom rows with a color code illustrate the selected object and action to perceive physical property at each decision step. Figure 12a shows that following our proposed learning method ATOL, the robot focused on collecting more

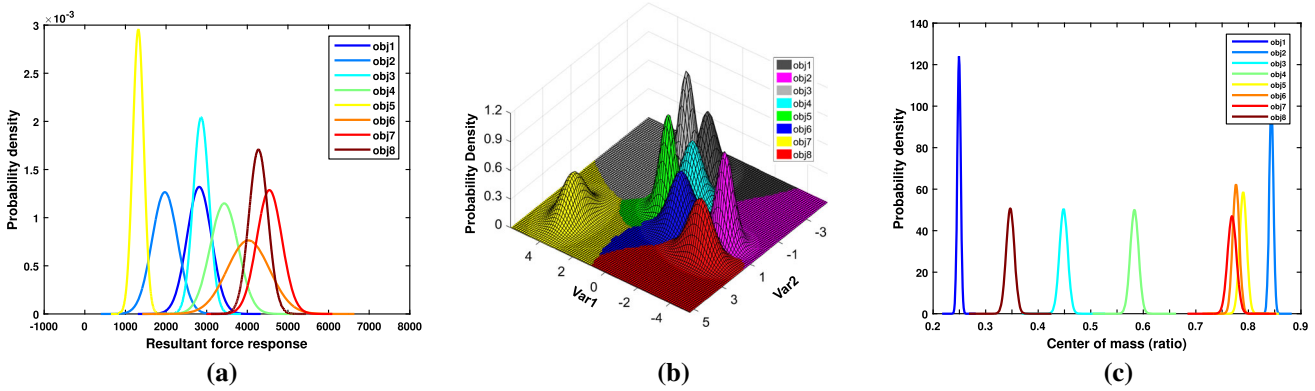


Fig. 10 Distributions of the features extracted from the test dataset. **a** The resultant force response for stiffness. **b** Robust textural descriptors. **c** CoM. The observation distributions for object stiffness and CoM are modeled by univariate Gaussian distribution. To visualize the distribution of textures, we first reduce the 12 dimensional texture descriptor to 2D vector via principle component analysis (PCA). Then we model the distributions of features by multivariate Gaussian distribution

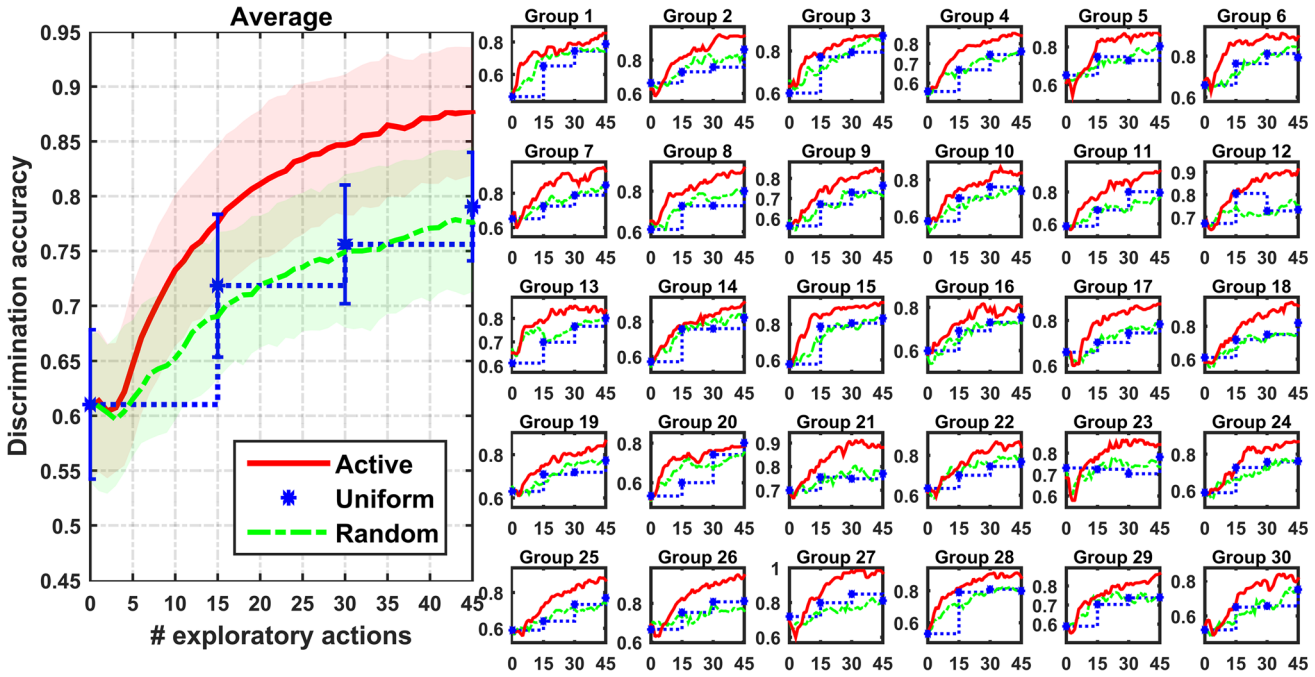


Fig. 11 Active learning about objects based on their physical properties. The horizontal axis represents the number of training data collected thus far. The vertical axis shows the mean value of the classification accuracy of evaluation dataset averaged over 30 runs

training samples for the objects' physical properties that make objects to be more confused (such as stiffness). Moreover, using ATOL the robot sampled less data to obtain the observations with which objects can be quickly recognized (such as surface texture). Conversely, since uniform (see Fig. 12b) and random learning strategies (see Fig. 12c) collected training samples without exploiting their informativeness, the "difficult" objects were insufficiently learned, while the "easy" objects were redundantly observed.

D. Evaluation of active tactile object learning via each physical property

In order to evaluate the robustness of our active learning algorithm, the robotic system was asked to learn about objects via only one of the three tactile properties (stiffness, surface texture, and CoM). Random sampling and uniform sampling serve as baseline. Each method was run 30 times by the autonomous robot. Figure 13 shows the learning performance of the robotic system when it explored the objects. It is evident that the learning progress was

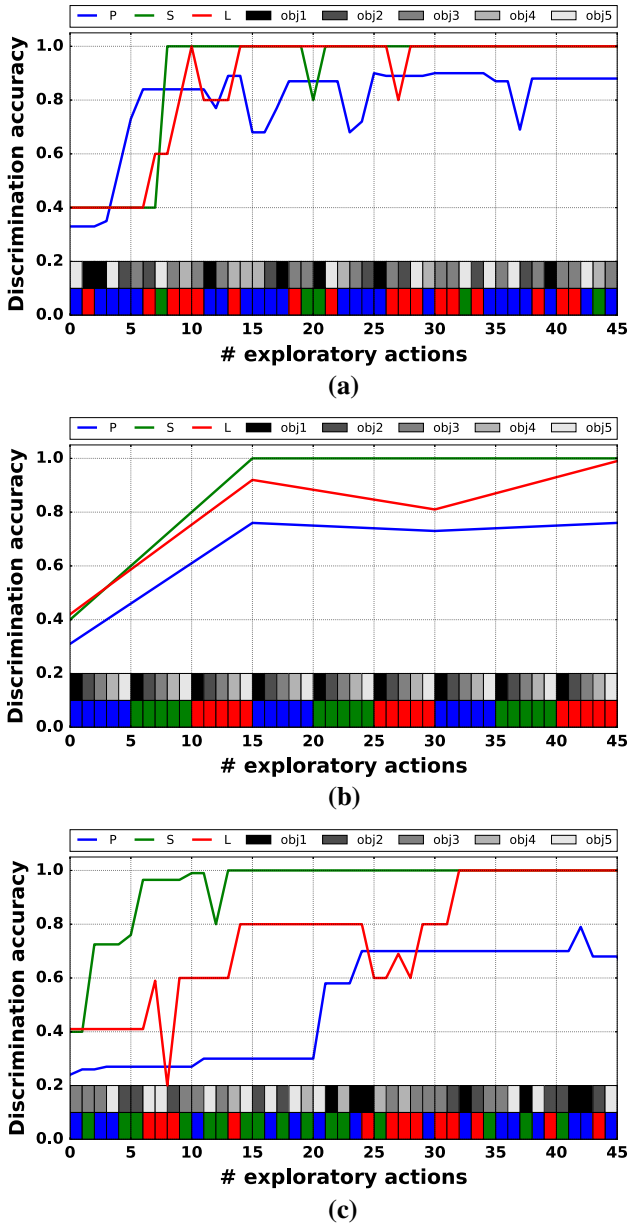


Fig. 12 An example of learning five objects with three physical properties. Three object and exploratory action selection methods are compared. **a** Proposed active learning method ATOL. **b** Uniform sampling method. **c** Random sampling method. The lower bar shows the exploratory actions at each time step ("P" for pressing, "S" for sliding, "L" for lifting). The upper bar shows the object to explore at each step. The vertical axis shows the classification accuracy on the test database

dependent on the distributions of the tactile features. For instance, Fig. 13a shows that learning objects via their stiffness led to low classification accuracy, because object's were confused by their stiffness. It is the same situation for learning objects via their CoM (see Fig. 13c). On the contrary, objects were easily distinguished by using our proposed robust tactile descriptors (see Fig. 13b). Therefore, the learning process for object surface texture was faster and ended with higher recognition rate. In all cases (see Fig. 13), the entropy reduction method outperforms the other two methods by up to 30% of the recognition accuracy with the same number of training samples. Therefore, our active learner is robust to different distributions of the tactile features.

8.2.3 Evaluation of active tactile object recognition (ATOR)

A. Evaluation of active object discrimination

With the reliable observation models constructed by our proposed active touch learning method (ATOL) with 30 groups of objects previously, we evaluated our proposed active object discrimination and action selection strategy. To do this, we compute the decision steps using our CMUM, the expected entropy reduction (EER) (Sect. 7.4) and random strategy $a^* \in \mathcal{U}(1,3)$, $a^* \in \{a_1 : \text{sliding}, a_2 : \text{pressing}, a_3 : \text{lifting}\}$ approaches to discriminate among objects in the workspace (see Fig. 14WS-2).

In this regard, the robot first explored the workspace following the procedure as described in Sect. 8.2. The constructed TPC, clustering results, and localization results are illustrated in Fig. 14.

The robotic system executed a sequence of exploratory movements on an object, until the object MAP exceeded 90%, or the iterations reached seven times. Then, we measured the number of decision steps and compared the MAP results to the true object class. The experiment was repeated 30 times. In each experiment, the robot used three methods to explore each object five trails.

Figure 15 shows the average number of decision steps. The robot discriminated among objects by CMUM and EER more quickly than the random method. Furthermore, the decision accuracy from CMUM are higher than EER and random (CMUM: 99.9%, EER: 92.4%, Random: 93.2%). Therefore, we can conclude that a robotic system that uses our proposed CMUM can discriminate objects quickly and correctly.

B. Evaluation of active target object search

When evaluating the robot's competence of searching for the target objects, we randomly replaced two known objects in the workspace at each of 30 groups of objects with two randomly selected unknown objects (see Fig. 16).

Now the task of the is robot to find the targeted object, or leaves the non-targeted object as quickly as possible while taking the advantage of its already constructed observation models in section as prior knowledge.

To do this, the robotic system explored each of the object in the workspace using our proposed CMUM strategy and the EER and random strategies as baselines. Each strategy was

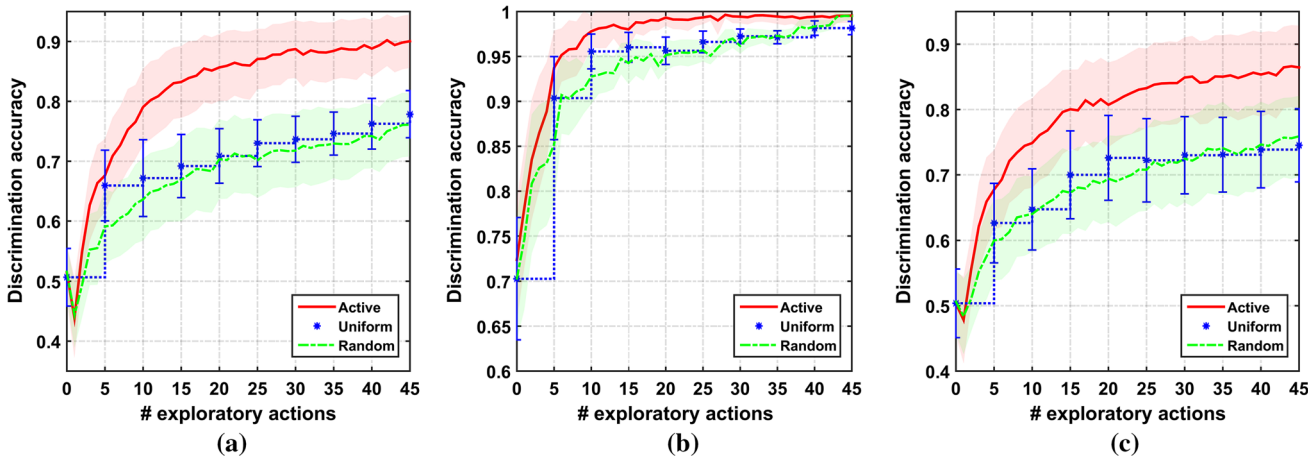


Fig. 13 Active learning each object physical property individually. **a** Learning object's stiffness. **b** Learning object's surface textures. **c** Learning object's CoM. The horizontal axis represents the growing number of training data, and the vertical axis represents the value of classification accuracy on the test data averaged over 30 runs

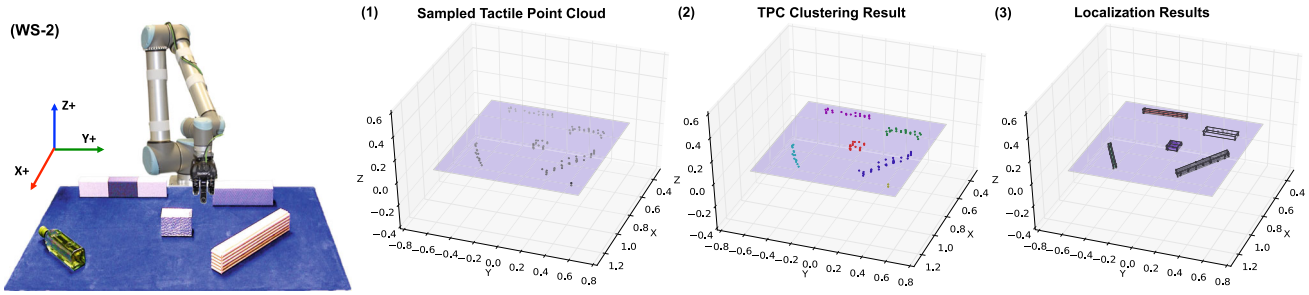


Fig. 14 The active exploration results of the unknown workspace in the second scenario (see Sect. 8.2.3) by applying our proposed strategy. **WS-2** The layout of the workspace in the second scenario. **a-1** The constructed TPC. **a-2** Clustering result of TPC. **a-3** Object localization result

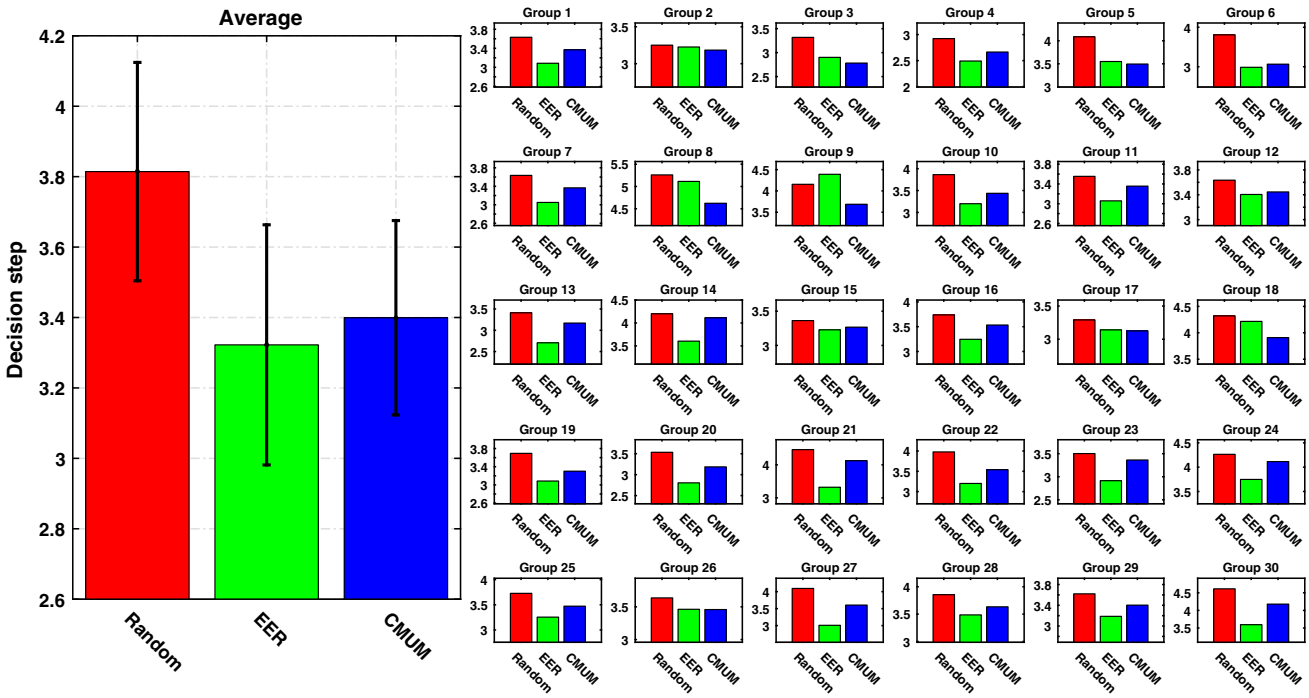


Fig. 15 Evaluating active object discrimination and target object search. Average decision steps the robot takes to discriminate objects

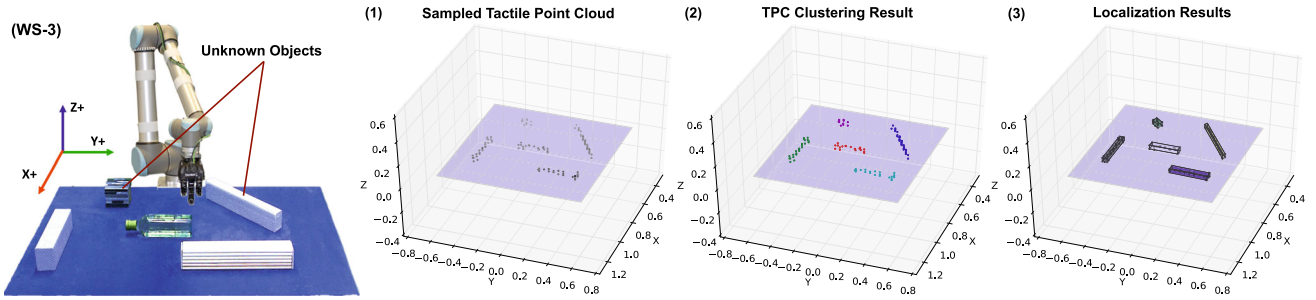


Fig. 16 The active exploration results of the unknown workspace in the third scenario (see Sect. 8.2.3) by applying our proposed strategy. **WS-3** The layout of the workspace in the third scenario. **a-1** The constructed TPC. **a-2** Clustering result of TPC. **a-3** Object localization result

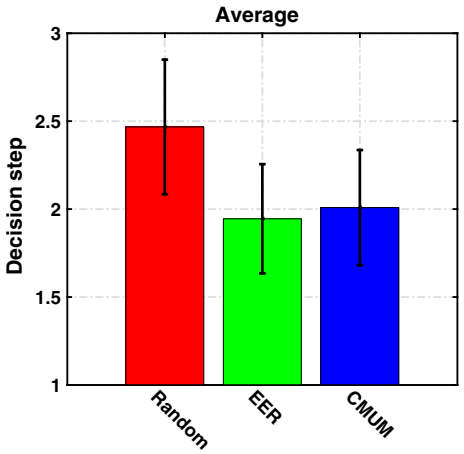


Fig. 17 Evaluating active object discrimination and target object search. Average decision steps the robot takes to find the target objects

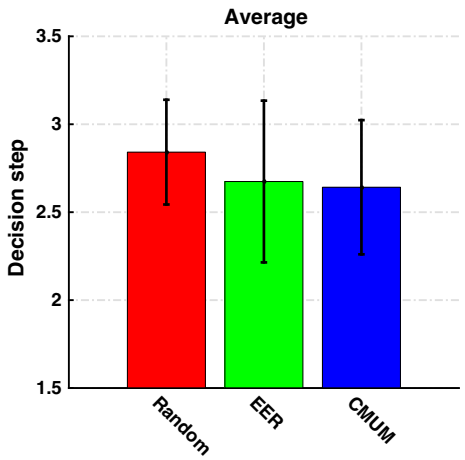


Fig. 18 Evaluating active object discrimination and target object search. Average decision steps the robot leaves the non-target objects

run 30 times with 30 groups of randomly selected objects. In each round, the robot explored each object in the workspace five times. The exploration was run until the object MAP is larger than 90%, or until seven exploratory movements were conducted. As a result, the robot either detected the target object (when $p(o = o_{tg}) > p(o \neq o_{tg})$) or the non-target object (when $p(o = o_{tg}) < p(o \neq o_{tg})$). We recorded the number of exploratory movements the robot executed in order to make a decision, as well as the decision accuracy.

Figure 17 illustrates average decision steps over 30 groups of objects which robot takes to find the target objects. Figure 17 shows that both CMUM and EER take fewer steps than random method to recognize all target objects. The decision accuracy from CMUM is higher than EER and the random selection (CMUM: 99%, EER: 87%, Random: 90%). Figure 17 shows the average decision step the robot leaves the non-target objects. Figure 18 demonstrates the similar results using CMUM, EER, and random strategies, the robotic system efficiently leave the non-target objects with almost the same decision accuracy (CMUM: 97%, EER: 92%, Random: 94%).

9 Discussion

We proposed a probabilistic strategy to enable the autonomous robot to explore its surroundings with high efficiency, in order to localize and estimate the geometric information of the objects in its perimeter. Taking advantage of GP regression, the robotic system explored the workspace so that more tactile points were sampled on the objects and around their perimeter. At each step, the tactile data already collected were used to construct a probabilistic model to guide the next step of workspace exploration. The experimental results show that our proposed approach outperforms both random and uniform data selection. The captured tactile data points were then clustered by the robotic system in order to ascertain the number of objects in the workspace. The minimum bounding box was calculated for each cluster to estimate the location, orientation, and geometric center of each object.

After object localization and workspace mapping, the autonomous robot actively discriminated them from each other based on their physical properties. In this regard, the robot used our proposed GP classification based method to

efficiently learn the physical properties of objects. It used the extended version of our previously proposed tactile descriptor to perceive the textural properties of objects by sliding its fingertips on the surface of the objects (objects with either regular or irregular textural properties). Moreover, the robot employed our suggested tactile-based approach to estimate the CoM of rigid objects. It measured the stiffness of each object by pressing against them with its three fingertips.

In previous studies, the observation models were constructed by the predefined number of training samples for each object, which were collected offline during tactile exploration. Contrary to previous works, using our proposed algorithm, the robotic system sampled the objects such that with a smaller number of exploratory actions it constructed reliable object observation models online. The robot collected more training samples from "difficult" objects which had fewer discriminative tactile properties and thus were confused with other objects. In other words, the robot did not deposit any redundant tactile information. The experimental results illustrate that the robotic system learned about objects based on their physical properties efficiently and achieved higher classification accuracy by using our proposed approach. It proves that our proposed method outperforms random and uniform sampling strategies.

After object learning phase, the autonomous robot efficiently distinguished experimental objects with arbitrary location and orientation from each other. It also found the target object in the workspace quickly. To do this, it used our proposed strategies for active object discrimination and target object search and took the advantage of the reliable observation models constructed during the object learning phase.

In this regard, the robotic system predicted the benefit of each of the exploratory actions (pressing, sliding, and lifting) and executed the one that would obtain the most discriminative properties. The performance of our proposed method was compared with both EER and random exploratory action selection strategies. The experimental results show that by using our proposed method the robotic system discriminated among objects faster than by using random strategy, and reached higher recognition rate than the EER.

It executed fewer exploratory actions to find the target object, while the usage of the random action selection strategy required more exploration. When estimating the benefit of an exploratory movement, CMUM inferred the dissimilarity between objects by building a probabilistic confusion matrix.

The most computational intensive part of the task is the Gaussian process, its computational complexity is $O(N^3)$ with N being the number of training data. The computational complexity of the active touch for unknown workspace exploration is the same as the GPR model, i.e. $O(N^3)$, since the exploration processes of each direction are independent. The computational complexity of active touch learning is $N_p \times N_o \times O(N^3)$, where N_p is the number of object physical properties, N_o is the number of objects, and N is the number of training data points. The complexity active object discrimination is $O(N_o)$, comes from the updating of object belief. For active target object search, the complexity is $O(N^3)$ for training the binary GPCs, then it becomes $O(N_o)$ for the update of object belief during online experiment.

The computation of action selection was proportional to the square of the number of objects ($O(N_o^2)$). However, EER integrated all the training samples to predict the benefit of an exploratory action. As the number of training samples grew, the computation became costly. In this case, for example instead of GPs the sparse approximation of Gaussian processes (Csato and Opper 2002) can be used.

Furthermore, it was found that compared to CMUM, EER would more frequently converge to a incorrect decision when the probability threshold was not set high enough. Its performance could be improved when the threshold was set higher, and more exploratory movements were required.

A limiting assumption of our work is that the experimental objects need to be rigid and should not be immobilized after the workspace exploration phase. Moreover, due to the constraints of our hardware, such as the size of robotic fingers and the spatial resolution of the tactile sensor, we selected cuboid objects to satisfy these constraints.

In the future, in order to evaluate our proposed framework with complex shapes and deformable objects, we will equip a humanoid robot with sense of touch and will extend our framework for dual hand workspace exploration and tactile object recognition. We will also generalize our proposed algorithm for the estimation of the center of mass of objects with complex shape. Moreover, a very interesting future work would be enabling the autonomous robot to learn about unknown objects while searching for the known target object in a workspace (active tactile transfer learning, Kaboli et al. 2016a, 2017a).

10 Conclusion

The probabilistic methods proposed in this study enables the robot with a sense of touch to perform a complete series of tasks: the active exploration of unknown workspaces, active object learning, and the active recognition of target objects. Its effectiveness was demonstrated through online experiments in several distinct scenarios.

Following the active touch for unknown workspace exploration, the robot explored and localized objects in completely unknown workspaces. It was able to reduces the uncertainty of the workspace up to 65 and 70% compared to uniform and random strategies respectively, for a fixed number of samples. The robot then autonomously learned about all objects in the

workspace based on their physical properties. Using our proposed active touch learning algorithm, the robot obtained 20 and 15% higher learning accuracy for the same number of training samples compared to uniform strategy and random strategy, respectively.

By taking advantage of the prior knowledge acquired previously, the robot successfully discriminated among objects. In this scenario, the robot took up to 15% less decision steps compared to the random method to achieve the same discrimination accuracy while using either our proposed CMUM method or the EER action selection method. However, taking advantage of our proposed CMUM action selection method, the robot achieved up to 10% higher decision accuracy in comparison with the EER.

Furthermore, to search for a target object, the robot reduced the decision steps up to 24% to find the target objects by following either the proposed CMUM method or the EER method, compared to the random method, whereas using the CMUM method, the robot reached 14% higher decision accuracy than EER to find the target object.

In addition, our p roposed tactile-based CoM detection approach demonstrated its validity in enabling the robot to find the CoM of a rigid object with a low computational complexity.

11 Appendix

11.1 Gaussian process (GP)

GP (Rasmussen and Williams 2006) is a supervised learning method which describes the functional mapping $\boldsymbol{g} : \mathcal{X} \mapsto \mathcal{Y}$ between the input data set $\mathcal{X}$ and the output data set $\mathcal{Y}$. The GP model is non-parametric and can be completely determined by its mean function $\mu(\boldsymbol{x})$ and covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$:

$$\mu(\boldsymbol{x}) = \mathbb{E}[\boldsymbol{g}(\boldsymbol{x})], \tag{37}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(\boldsymbol{g}(\boldsymbol{x}) - \mu(\boldsymbol{x}))(\boldsymbol{g}(\boldsymbol{x}') - \mu(\boldsymbol{x}'))]. \tag{38}$$

The distribution of $\boldsymbol{g}$ can be denoted as:

$$\boldsymbol{g}(\boldsymbol{x}) \sim \mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')). \tag{39}$$

In a regression task, GP is exploited to approximate the functional relation $\boldsymbol{g}$, in order to predict the output $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x})$ given a test input $\boldsymbol{x}$.

In this work, we used the radial basis function (RBF) as the covariance function:

$$k(\boldsymbol{a}, \boldsymbol{b}) = \sigma_f^2 \exp(-\frac{(\boldsymbol{a} - \boldsymbol{b})^2}{2l^2}) + \sigma_n^2 \delta_{ab}, \tag{40}$$

the hyper-parameters σ_f, σ_n, and l are selected through cross-validation in order to tune the kernel for fitting the training dataset.

11.2 Normalized mutual information (NMI)

In the information theory, the mutual information measures the mutual dependency between two variables, i.e. how much information of a variable can be obtained, when the other variable is known.

Formally, the mutual information of two discrete variables X and Y is defined as:

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \tag{41}$$

The mutual information is a non-negative value, with $I(X, Y) = 0$ indicating that the two variables are independent.

Normalized mutual information normalizes the mutual information to be within 0 and 1, with 1 referring that two random variables are the same. In this work, we used the form proposed in Strehl and Ghosh (2002):

$$\mathrm{NMI}(X, Y) = \frac{I(X, Y)}{\sqrt{H(X)H(Y)}}. \tag{42}$$

This measurement has been widely used to evaluate the performance of a clustering algorithm. Let $\boldsymbol{y}_{\text{true}}$ be the true labels of the data samples, $\boldsymbol{y}_{\text{pred}}$ be their predicted labels. Then the NMI measurement can be computed by:

$$\mathrm{NMI}(\boldsymbol{y}_{\text{true}}, \boldsymbol{y}_{\text{pred}}) = \frac{I(\boldsymbol{y}_{\text{true}}, \boldsymbol{y}_{\text{pred}})}{\sqrt{H(\boldsymbol{y}_{\text{true}})H(\boldsymbol{y}_{\text{pred}})}}. \tag{43}$$

Acknowledgements This work is supported by the European Commission under Grant Agreements PITN-GA-2012-317488-CONTEST.

References

Atkeson, C. G., An C. H., & Hollerbach, J. M. (1985) Rigid body load identification for manipulators. In *IEEE conference on decision and control* (pp. 996–1002).

Bhattacharya, A., & Mahajan, R. (2003). Temperature dependence of thermal conductivity of biological tissues. *Physiological Measurement*, *24*(3), 769.

Chathuranga , D. S., Wang, Z., Ho, V. A., Mitani, A., & Hirai, S. (2013). A biomimetic soft fingertip applicable to haptic feedback systems for texture identification. In *IEEE international symposium on haptic audio visual environments and games* (pp. 29–33).

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *17*(8), 790–799.

Chu, V., McMahon, I., Riano, L., McDonald, C. G., He, Q., et al. (2013). Using robotic exploratory procedures to learn the meaning of haptic adjectives. In *IEEE international conference on robotics and automation* (pp. 3048–3055).

Csato, L., & Opper, M. (2002). Sparse on-line gaussian processes. *Neural Computation*, *14*(3), 641–668.

Dahiya, R. S., Metta, G., Valle, M., Adami, A., & Lorenzelli, L. (2009). Piezoelectric oxide semiconductor field effect transistor touch sensing devices. *Applied Physics Letters*, *95*(034), 105.

Dahiya, R. S., Metta, G., Valle, M., & Sandini, G. (2010). Tactile sensing—From humans to humanoids. *IEEE Transactions on Robotics*, *26*(1), 1–20.

Dallaire, P., Giguere, P., Edmond, D., & Chaib-draa, B. (2014). Autonomous tactile perception: A combined improved sensing and bayesian nonparametric approach. *Robotics and Autonomous Systems*, *6*(4), 422–435.

Denei, S., Maiolino, P., Baglini, E., & Cannata, G. (2015). On the development of a tactile sensor for fabric manipulation and classification for industrial applications. In *IEEE international conference on intelligent robots and systems* (pp. 5081–5086).

Fishel, J. A., & Loeb, G. E. (2012). Bayesian exploration for intelligent identification of textures. *Frontiers in Neurorobotics*, *6*, 4.

Friedl, K. E., Voelker, A. R., Peer, A., & Eliasmith, C. (2016). Human-inspired neurorobotic system for classifying surface textures by touch. *IEEE Robotics and Automation Letters*, *1*, 516–523.

Giguere, P., & Dudek, G. (2011). A simple tactile probe for surface identification by mobile robots. *IEEE Transactions on Robotics*, *27*(3), 534–544.

Hughes, D., & Correll, N. (2015). Texture ecognition and localization in amorphous robotic skin. *Bioinspiration and Biomimetics*, *10*(055), 002.

Hu, H., Han, Y., Song, A., Chen, S., Wang, C., & Wang, Z. (2014). A finger-shaped tactile sensor for fabric surfaces evaluation by 2-dimensional active sliding touch. *Sensors*, *14*, 4899–4913.

Jamali, N., Ciliberto, C., Rosasco, L., & Natale, L. (2016). Active perception: Building objects models using tactile exploration. In *IEEE international conference on humanoid robots*.

Jamali, N., & Sammut, C. (2011). Majority voting: Material classification by tactile sensing using surface texture. *IEEE Transactions on Robotics*, *27*(3), 508–521.

Jia, Y., & Tian, J. (2010). Surface patch reconstruction from 'one-dimensional' tactile data. *IEEE Transactions on Automation Science and Engineering*, *7*, 400–407.

Kaboli, M., & Cheng, G. (2018). Robust tactile descriptors for discriminating objects from textural properties via artificial robotic skin. *IEEE Transaction on Robotics*, *34*(2), 1–19.

Kaboli, M., Feng, D., & Cheng, G. (2017a). Active tactile transfer learning for object discrimination in an unstructured environment using multimodal robotic skin. *International Journal of Humanoid Robotics*, *15*(1), 18–51.

Kaboli, M., Feng, D., Yao, K., Lanillos, P., & Cheng, G. (2017b). A tactile-based framework for active object learning and discrimination using multimodal robotic skin. *IEEE Robotics and Automation Letters*, *2*(4), 2143–2150.

Kaboli, M., Long, A., & Cheng, G. (2015a). Humanoids learn touch modalities identification via multi-modal robotic skin and robust tactile descriptors. *Advanced Robotics*, *29*(21), 1411–1425.

Kaboli, M., Mittendorfer, P., Hugel, V., & Cheng, G. (2014). Humanoids learn object properties from robust tactile feature descriptors via multi-modal artificial skin. In *IEEE international conference on humanoid robots* (pp. 187–192).

Kaboli, M., Rosaand, A. D. L., Walker, R., & Cheng, G. (2016a). Re-using prior tactile experience by robotic hands to discriminate in-hand objects via texture properties. In *IEEE international conference on robotics and automation* (pp. 2242–2247).

Kaboli, M., Walker, R., & Cheng, G. (2015b). In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors. In *IEEE international conference on humanoid robots* (pp. 2242–2247).

Kaboli, M., Yao, K., & Cheng, G. (2016b). Tactile-based manipulation of deformable objects with dynamic center of mass. In *IEEE international conference on humanoid robots*.

Kaltenbrunner, M., Sekitani, T., Reeder, J., Yokota, T., Kuribara, K., et al. (2013). An ultra-lightweight design for imperceptible plastic electronics. *Nature*, *499*, 455–463.

Lederman, S. J. (1981). The perception of surface roughness by active and passive touch. *Bulletin of the Psychonomic Society*, *18*, 253–255.

Lee, S., Reuveny, A., Reeder, J., Lee, S., Jin, H., et al. (2016). A transparent bending-insensitive pressure sensor. *Nature Nanotechnology*, *11*, 472–478.

Lepora, N. F., Evans, M., Fox, C. W., Diamond, M. E., Gurney, K., & Prescott, T. J . (2010). Naive bayes texture classification applied to whisker data from a moving robot. In *The international joint conference on neural networks* (pp. 1–8).

Lepora, N. F., Martinez-Hernandez, U., & Prescott, T. J. (2013). Active touch for robust perception under position uncertainty. In *IEEE International Conference on Robotics and Automation* (pp. 3020–3025).

Liarokapis, M. V., Calli, B., Spiers, A., & Dollar, A. M. (2015). Unplanned, model-free, single grasp object classification with underactuated hands and force sensors. In *IEEE international conference on intelligent robots and systems* (pp. 5073–5080).

Liu, H., Greco, J., Song, X., Bimbo, J., & Althoefer, K. (2013). Tactile image based contact shape recognition using neural network. In *IEEE international conference on multisensor fusion and integration for intelligent systems (MFI)* (pp. 138–143).

Liu, H., Song, X., Nanayakkara, T., Seneviratne, L. D., & Althoefer, K. (2012). A computationally fast algorithm for local contact shape and pose classification using a tactile array sensor. In *IEEE international conference on robotics and automation* (pp. 1410–1415).

Liu, H., Nguyen, K. C., Perdereau, V., Bimbo, J., Back, J., Godden, M., et al. (2015). Finger contact sensing and the application in dexterous hand manipulation. *Autonomous Robots*, *39*(1), 25–41.

Martinez-Cantin, R., de Freitas, N., Doucet, A., & Castellanos, J. A. (2007). Active policy learning for robot planning and exploration under uncertainty. *Robotics: Science and Systems*, *3*, 321–328.

Martins, R., Ferreira, J. F., & Dias, J. (2014). Touch attention bayesian models for robotic active haptic exploration of heterogeneous surfaces. In *International conference on intelligent robots and systems* (pp. 1208–1215).

Mayol-Cuevas, W., Juarez-Guerrero, J., & Munoz-Gutierrez, S. (1998). A first approach to tactile texture recognition. In *IEEE international conference on systems, man, and cybernetics* (Vol. 5, pp. 4246–4250).

Mohamad Hanif, N. H. H., Chappell, P. H., Cranny, A., & White, N. M . (2015). Surface texture detection with artificial fingers. In *37th annual international conference of the IEEE engineering in medicine and biology society* (pp. 8018–8021).

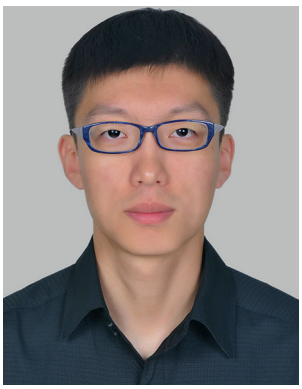
Nawrocki, A. R., Matsuhisa, N., Yokota, T., & Someya, T. (2016). 300-nm imperceptible, ultraflexible, and biocompatible e-skin fit with tactile sensors and organic transistors. *Advanced Electronic Materials*. https://doi.org/10.1002/aelm.201500452.

Nguyen, K. C., & Perdereau, V. (2013). Fingertip force control based on max torque adjustment for dexterous manipulation of an anthropomorphic hand. In *2013 IEEE/RSJ international conference on intelligent robots and systems* (pp. 3557–3563).

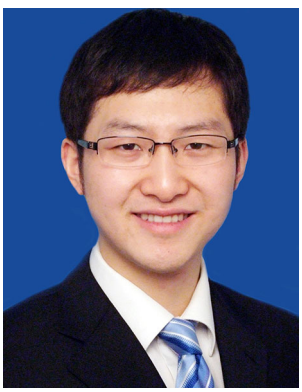
Ohmura, Y., Kuniyosh, Y., & Nagakubo, A . (2006). Conformable and scalable tactile sensor skin for curved surfaces. In *IEEE international conference on robotics and automation* (pp. 1348–1353).

Papakostas, T. V., Lima, J., & Lowe, M. (2002). A large area force sensor for smart skin applications. *IEEE Sensors*, *5*, 1620–1624.
Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.
Rebguns, A., Ford, D., & Fasel, I. R. (2011). Infomax control for acoustic exploration of objects by a mobile robot. In *Lifelong learning*.
Robles-De-La-Torre, G. (2006). The importance of the sense of touch in virtual and real environments. *IEEE MultiMedia*, *13*, 24–30.
Saal, H. P., Ting, J. A., & Vijayakumar, S. (2010). Active sequential learning with tactile feedback. In *AISTATS* (pp. 677–684).
Schmitz, A., Maiolinoand, P., Maggiali, M., Natale, L., Cannata, G., & Metta, G. (2011). Methods and technologies for the implementation of large-scale robot tactile sensors. *IEEE Transactions on Robotics*, *27*(3), 389–400.
Schneider, A., Sturm, J., Stachniss, C., Reisert, M., Burkhardt, H., & Burgard, W. (2009). Object identification with tactile sensors using bag-of-features. In *IEEE international conference on intelligent robots and systems* (pp. 243–248).
Sinapov, J., Sukhoy, V., Sahai, R., & Stoytchev, A. (2011). Vibrotactile recognition and categorization of surfaces by a humanoid robot. *IEEE Transactions on Robotics*, *27*(3), 488–497.
Song, A., Han, Y., Hu, H., & Li, J. (2014). A novel texture sensor for fabric texture measurement and classification. *IEEE Transactions on Instrumentation and Measurement*, *63*(7), 1739–1747.
Strehl, A., & Ghosh, J. (2002). Cluster ensembles—A knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, *3*, 583–617.
Strohmayr, M. W., Worn, H., & Hirzinger, G . (2013). The DLR artificial skin part i: Uniting sensitivity and robustness. In *IEEE international conference on robotics and automation* (pp. 1012–1018).
Tanaka, D., Matsubara, T., Ichien, K., & Sugimoto, K. (2014). Object manifold learning with action features for active tactile object recognition. In *IEEE international conference on intelligent robots and systems* (pp. 608–614).
Ulmen, J, & Cutkosky, M. (2010). A robust low-cost and low-noise artificial skin for human-friendly robots. In *IEEE international conference on robotics and automation* (pp. 4836–4841).
Watanabe, K., Sohgawa, M., Kanashima, T., Okuyama, M., & Norna, H. (2013). Identification of various kinds of papers using multi-axial tactile sensor with micro-cantilevers. In *World haptics conference* (pp. 139–144).
Xu, D., Loeb, G. E., & Fishel, J. A. (2013). Tactile identification of objects using bayesian exploration. In *IEEE international conference on robotics and automation* (pp. 3056–3061).
Yao, K., Kaboli, M., & Cheng, G. (2017). Tactile-based object center of mass exploration and discrimination. In *IEEE international conference on humanoid robotics* (pp. 876–881).
Yi, Z., Calandra, R., Veiga, F., van Hoof, H., Hermans, T., Zhang, Y., et al. (2016). Active tactile object exploration with gaussian processes. In *IEEE international conference on intelligent robots and systems* (pp. 4925–4930).
Yogeswaran, N., Dang, W., Navaraj, W. T., Shakthivel, D., et al. (2015). New materials and advances in making electronic skin for interactive robots. *Advanced Robotics*, *29*(21), 1359–1373.
Yu, Y., Arima, T., & Tsujio, S. (2005). Estimation of object inertia parameters on robot pushing operation. In *IEEE international conference on robotics and automation* (pp. 1657–1662).
Yu, Y., Kiyokawa, T., & Tsujio, S. (2004). Estimation of mass and center of mass of unknown and graspless cylinder-like object. *International Journal of Information Acquisition*, *1*(01), 47–55.

Mohsen Kaboli received his bachelor degree in electrical and electronic engineering and master degree in signal processing and machine learning from the Royal Institute of Technology (KTH) in Sweden, in 2011. In March 2012, he received a scholarship from the Swiss National Foundation for 18 months in order to continue his research as a research assistant at the Idiap lab, École polytechnique fèdèrale de Lausanne (EPFL). April 2013, he was awarded a three-year Marie Curie scholarship. Since that time he has been at the Institute for Cognitive Systems directed by Prof. Gordon Cheng in Technical University of Munich (TUM). In November 2013, he visited the Shadow Robot Company for 2 months. Mohsen has been a visiting researcher at the Human Robotics lab, department of Bioengineering at Imperial College London supervised by Prof. Ettien Burdet from February till April 2014. From September 2015 till January 2016, he spent 5 months as a visiting research scholar at the Intelligent Systems and Informatics lab at the University of Tokyo, Japan.



Kunpeng Yao received his Bachelor's degree in automation, in the School of Electronic Information and Electrical Engineering from Shanghai Jiao Tong University (SJTU), China, in 2013. Currently he is a Master student in the Department of Electrical and Computer Engineering, Technische Universität München, in Munich, Germany. His research interests include machine learning in robotics, tactile sensing, and dexterous manipulation.



Di Feng was born in Kunming, China, in 1991. He is currently pursuing the masters degree in electrical and computer engineering at the Technical University of Munich. He received the bachelors degree in mechatronics with honor from Tongji University, in 2014. During his bachelor study, he received DAAD scholarship and visited Niederrhein University of Applied Science as exchange student from Sept. 2013 till Aug. 2014. His current research is centered on robotic active learning and exploration through tactile sensing. He is interested in perception, lifelong learning machines and cognitive systems.

Gordon Cheng is the Professor and founding director of the Institute for Cognitive Systems, Technical University of Munich, Germany. Prof. Cheng is speaker of the established Elite Master of Science program in Neuroengineering (MSNE) of the Elite Network of Bavaria. He is also involved in a large number of major European Union Projects (e.g. RoboCub, Factory-in-a-Day, CONTEST-ITN, RoboCom-Flagship). He held fellowships from the Center of Excellence and the Science and Technology Agency of Japan. Both of these fellowships were taken at the Humanoid Interaction Laboratory, Intelligent Systems Division at the Electrotechnical Laboratory, Japan. Formerly, he was the Head of the Department of Humanoid Robotics and Computational Neuroscience, ATR Computational Neuroscience Laboratories, Kyoto, Japan. He was the Group Leader for the JST International Cooperative Research Project (ICORP), Computational Brain. He has also been designated as a Project Leader/Research Expert for National Institute of Information and Communications Technology (NICT) of Japan. He received the Ph.D. degree in systems engineering from the Department of Systems Engineering, The Australian National University, in 2001, and the bachelor degrees in computer science from the University of Wollongong, Wollongong, Australia, in 1991 and 1993, respectively. He was the Managing Director of the company G.T.I. Computing in Australia. His current research interests include humanoid robotics, cognitive systems, brain machine interfaces, biomimetic of human vision, human–robot interaction, active vision, and mobile robot navigation. He is the co-inventor of approximately 15 patents and has co-authored approximately 350 technical publications, proceedings, editorials, and book chapters.