Intelligent Service Robotics (2025) 18:1119–1131
https://doi.org/10.1007/s11370-025-00628-8

ORIGINAL RESEARCH PAPER

# A cross-task visuo-tactile representation using point clouds

Giammarco Caroleo[1] · Alessandro Albini[1] · Perla Maiolino[1]

Received: 6 February 2025 / Accepted: 30 June 2025 / Published online: 1 August 2025


**Abstract**
Combining visual and tactile cues has proven effective for object recognition, grasping, and manipulation tasks. However, integrating these modalities is challenging as tactile and visual data convey distinct information and differ structurally. Researchers have addressed this problem by proposing approaches that either do not consider mechanical properties conveyed by tactile sensors or cannot be deployed directly for diverse tasks. In this paper, we propose a cross-task visuo-tactile representation that encodes both the geometrical and mechanical properties of objects in a point cloud (PC) data structure. By physically exploring different areas of a given item, we collect tactile information to estimate the local compliance of the surface, encoding it as the color information of the PC in the probed areas. This color information is extended to the entire object assuming that neighboring points share the same mechanical properties. We apply the proposed PC to six real-world objects showing that it can be effectively used to encode their shape along with their information on the local compliance. Further, we show that the augmented PC can be used for different tasks by exploiting this in three robotic tasks—a visuo-tactile object classification, a path following and a reaching in clutter.

**Keywords** Force and tactile sensing · Sensor fusion · Point clouds

## 1 Introduction

The interaction of visual and tactile stimuli has proven to be a facilitator for early sensory processing in humans, i.e., the presence of a stimulus in one modality enhances the perception for the other [1]. Similarly, the combination of vision and touch has shown great enhancing potential in diverse robotics tasks, such as object recognition [2–5], shape reconstruction [6–9] and manipulation [10–12]. Despite the advantage of exploiting these two modalities together, combining visual and tactile information is not straightforward—raw data acquired from each sensor have diverse structure and resolution. To address this challenge, various solutions have been proposed. Some works encode visuo-tactile data using a point cloud (PC) PC data structure. This can be effectively employed to represent objects or the robot's environment by converting camera depth images and contact points into Cartesian coordinates, thus serving as a common structure to capture visual and tactile information. In this respect, visuo-tactile PCs have been utilized for cross-modal object recognition [2–4, 13] and object pose estimation [14, 15]. Additionally, other works demonstrate the potential of processing PCs further to create mesh-based object representations [7, 8] or continuous models using Gaussian Process Implicit Surfaces (GPIS) [12, 16, 17].

Although the previously mentioned works showed the effectiveness of combining vision and touch to represent the geometrical features of objects, their mechanical properties (such as friction or stiffness) are not considered. Researchers have addressed this aspect by proposing processing pipelines that encode visuo-tactile data with two distinct representations, which are then combined using machine learning (ML) solutions [18–22]. For instance, Liu et al. [18] process raw tactile data and RGB images for a visuo-tactile object recognition task. In [21, 22], vision-based tactile sensors are used in combination with RGB images. In particular, in [21], learning is adopted to achieve better object clustering performance in a lifelong fashion. Instead, Yang et al. [22] use latent space binding of visual and tactile cues to accomplish diverse tasks such as cross-modal recognition and image generation. To

✉ Giammarco Caroleo
giammarco@robots.ox.ac.uk

Alessandro Albini
alessandro@robots.ox.ac.uk

Perla Maiolino
perla@robots.ox.ac.uk

[1] Oxford Robotics Institute, University of Oxford, Banbury Road, Oxford OX2 6NN, UK

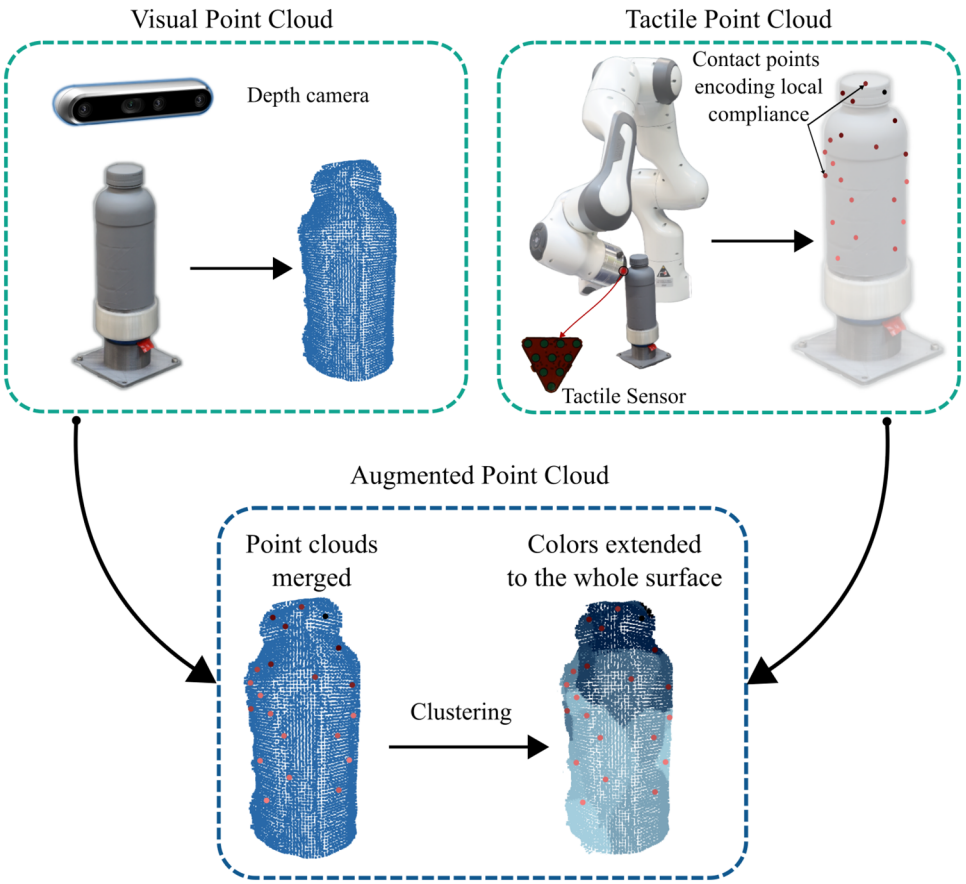


**Fig. 1** Overview of the proposed approach. A high-resolution point cloud is used to encode the shape of the object. Tactile sensing is used to create a lower-resolution point cloud where a color linked to the mechanical property at the contact point is associated with the geometric coordinate. We measure the local compliance of the object. The two point clouds are merged, and the properties encoded with the color are extended to the high-resolution point cloud through a clustering procedure

navigate in clutter with a robot arm, in [19, 20], the authors use spatio-temporal sparse haptic measures coming from a tactile-sensing sleeve and RGB information from a depth camera to train ML algorithms to achieve a dense haptic map of the workspace. Even though these approaches successfully integrate mechanical and geometrical information to perform a specific task, they lack flexibility for other applications. Each new task requires gathering additional data and training new ML models for each case, thus limiting their flexibility and applicability.

In this paper, we propose a cross-task representation that encodes both geometrical and physical information from cameras and tactile sensors. This representation is based on PC data structures and *augmented* with the measured mechanical properties encoded as a color map. The embedding of physical properties in PCs was successfully used in [23–25]. However, with respect to [23], we propose two methods to propagate the tactile measures across the object's surface, avoiding time-consuming full tactile exploration by assuming that neighboring points share similar properties. Also, different from [24, 25], we do not rely on camera-provided color information to extend physical properties, as it fails in correctly extending tactile cues when an object has a homogeneous color but is made of different materials [24].

To validate our method, we exploit the augmented PC in three tasks. First, we perform a visuo-tactile object classification task, demonstrating that the augmented PC can be directly fed into a simple PointNet model [26] without further processing or custom pipelines, unlike in [5, 18, 27, 28]. This approach allows distinguishing objects with the same visual features but with different mechanical properties. Second, our representation aids a robot in a tactile-based path-following task, adjusting contact forces based on local surface compliance. Third, the augmented PC helps the robot avoid collisions with stiffer objects while accommodating contact with compliant ones during a reaching in clutter task.

The paper is structured as follows: Sect. 2 outlines the proposed approach. Section 3 details the experimental setup. Section 4 describes the three validation tasks. Section 5 presents and discusses the results, followed by the Conclusion.

## 2 Augmented point cloud

An overview of the proposed approach is shown in Fig. 1. We adopted a representation based on PC due to its simplicity and computational advantages with respect to methods using Gaussian Process [29] as those proposed in [30, 31].

A robot equipped with a tactile sensor probes onto an object of interest to compute its local compliance. Multiple tactile interactions are performed in different locations of the object to generate a tactile PC. This way, the compliance estimated at each contact point is encoded as a *tactile color* and then extended to the points belonging to the visual PC (i.e., acquired with a depth camera) with two different methods: a cluster-based and a smooth interpolation.

For the proposed method, first, we describe the estimation of the local compliance at the contact location and how this attribute is encoded as a color. Then, we explain how to extend local properties to the whole object's surface by combining information from multiple contacts.

### 2.1 Estimation of the local compliance

To estimate the compliance of the objects, we assume the robot to be equipped with a tactile sensor integrated on its end-effector and to be controlled to reach and probe the surface of the object by exerting a force on it. The method can be applied to different types of tactile sensors, raising an attribute $c \in \mathbb{R}^d$, with $d$ being its dimension. We consider a tactile sensor array that is composed of a set of transducers providing a value that is a function of the normal component of the force applied on top of them. Therefore, at a given time instant, the sensor provides a set of raw measurements $P = \{p_1, p_2, \ldots, p_M\}$ related to the force applied while in contact with the object, with $M$ the number of transducers.

The shape of the object is known in the form of a PC composed of $K$ points, namely $\mathbb{V} : \{\mathbf{x}_k | \mathbf{x}_k \in \mathbb{R}^3\}$, with $k = 1, 2, \ldots, K$. During the probing operation, the end-effector is commanded to reach a point $\mathbf{v} \in \mathbb{V}$ belonging to the visual PC. Then the robot starts applying a controlled force in closed loop on the object that does not undergo any rigid motion. More specifically, we controlled the average values of the sensor measurements $\bar{p} = \frac{1}{M} \sum_1^M p_j$ to reach a target value $\bar{p}^*$. The value $\bar{p}^*$ is empirically chosen to have just local elastic deformation of the objects when the robot is touching them and to avoid any buckling or nonlinear effect. Moreover, in order to prevent unwanted sliding motions during the probing phase, the end-effector is aligned with the normal at the point $\mathbf{v}$ (which can be computed from the PC as described in [32]).

At the steady state, we compute the one-dimensional attribute $c$, namely a coefficient that is related to the compliance of the surface at the contact point:

$$c = \frac{\bar{p}}{\|\mathbf{v}_{ss} - \mathbf{v}\|}, \tag{1}$$

with $\mathbf{v}_{ss}$ the position of the end-effector at the steady state. Therefore, $c$ represents a term inversely proportional to the displacement created when applying a force $\bar{p}$, i.e., the larger the displacement, the lower $c$ is at the given contact point. The value of $c$ is dimensionless since directly computed from the tactile sensor's raw measurements. Indeed, to retrieve the absolute force value from tactile sensors, characterization and calibration procedures (usually hardware dependent [33]) are required. Within the scope of this paper, this aspect is of low interest since we aim to retrieve a qualitative indication of the mechanical property that is then represented as a color map for the resulting PC. The attribute $c$ obtained in this way is linearly mapped to a color scale common to all the objects.

### 2.2 Augmented PC generation

The *tactile color* value $c$ can be computed for different areas of interest on the object. In this respect, we can apply the procedure described in Sect. 2.1 considering different contact points $\mathbf{v}_i \in \mathbb{V}$. As a result, we obtain a set of values $c_i$ with $i = 1, 2, \ldots, N$ and $N$ the number of contact points. This allows for defining a low resolution PC, $\mathbb{T} : \{v_i, c_i | \mathbf{v}_i \in \mathbb{V}, c_i \in \mathbb{R}\}$, where $c_i$ represents the *tactile color* of each point $\mathbf{v}_i$.

The next step consists of combining $\mathbb{V}$ and $\mathbb{T}$ to create an augmented representation of the whole object. Formally, we want to define the scalar field $c : \mathbb{V} \rightarrow \mathbb{R}$ knowing the values of the field at a discrete set of points $L \subseteq \mathbb{V}$ as provided by $\mathbb{T}$.[1] In particular, the color is extended to nearby points with the rationale that nearby points share similar local properties by using both a cluster-based and a smooth interpolation method. The cluster-based approach allows for fast computation, while the smooth one provides a more realistic representation of the object's properties at the cost of a higher computational burden.

With both methods, the resulting augmented PC is composed of the same points of $\mathbb{V}$ with additional information related to the *tactile color* and its generation can be performed iteratively. When the number of touches is equal to one, the color property is extended to the whole surface. By increasing the number of touches, we obtain a more accurate indication of how the compliance of the object is distributed across the whole surface. An example of the updating process on a bottle is shown in Fig. 2.

### 2.3 Cluster-based interpolation

This method uses a clustering algorithm to extend the mechanical properties to the entire surface of the objects, resulting in a sharp interpolation.

Considering the contact points $\mathbf{v}_i$ as the centroid of the clusters, we aim at minimizing the sum of Euclidean distances from each point $\mathbf{x}_k \in \mathbb{V}$ to the nearby cluster

[1] We refer the reader to [34] for a formal mathematical treatment of scalar fields on point clouds.


**Fig. 2** Augmented PC representation at different number of touches. Starting from the visual PC (left), i.e., no tactile information, the colormap is updated by increasing the number of touches, thus obtaining a more accurate representation of how the compliance varies across the different areas of the object. The picture shows the augmented PC generated with 0, 2, 13, 22 touches, respectively, with the smooth (top row) and the cluster-based (bottom row) interpolations

centroids[2]. This corresponds to the minimization of the following cost function:

$$J = \sum_{k=1}^{K} \sum_{i=1}^{N} w_{k,i} \|\mathbf{x}_k - \mathbf{v}_i\|^2. \tag{2}$$

where $w_{k,i} = 1$ if $\mathbf{x}_k$ belongs to the $i$-th cluster and 0 otherwise. The augmented PC can be defined as $\mathbb{A} : \{\mathbf{x}_k, c_k | \mathbf{x}_k \in \mathbb{V}, c_k \in \mathbb{R}\}$, where $c_k = c_i$ when $w_{k,i} = 1$. Given that the augmented PC updates when a new tactile measurement is retrieved according to Eq.(2), no fixed cluster size is defined beforehand and the number of clusters is given by the number of tactile explorations.

### 2.4 Smooth interpolation

To get a smoother *tactile color* distribution, we use an interpolation based on a softmax function by assuming that the color of a point is given by a linear combination of the *tactile colors* of the contact points. The weights depend on the distance between the visual point and the contact points and are computed as follows:

$$\sigma_{i,j} = \frac{e^{-\beta d_{i,j}}}{\sum_{j=1}^{N} e^{-\beta d_{i,j}}}, \tag{3}$$

where $d_{i,j}$ is the Euclidean distance between the visual point $\mathbf{v}_i$ and the $j$-th contact point and $\beta$ is an empirically chosen scaling factor that controls the smoothness of the interpolation, set to $-300$ for the specific application. Higher absolute values of $\beta$ lead to a representation similar to the cluster-based one, while smaller values would excessively smoothen the color distribution, yielding a PC that poorly represents the local properties of the object. The *tactile color* of the $k$-th visual point is then computed as:

$$c_k = \sum_{j=1}^{N} \sigma_{k,j} c_j. \tag{4}$$

The definition of $\mathbb{A}$, the augmented PC, is the same as before.

The complexity of this method is $O(NM)$, where $M$ is the number of points in the visual PC and $N << M$. Although asymptotically equivalent to cluster-based interpolation, it involves the computation of Eqs. (3) and (4), thus requiring more operations. Nonetheless, both proposed methods have lower complexity compared to GP-based approaches, which have the additional cost of the matrix inversion being in this case of $O(N^3)$ complexity [29].

## 3 Experimental setup

The approach was validated using the experimental setup shown in Fig. 3.

[2] A comprehensive analysis on clustering methods can be found in [35].

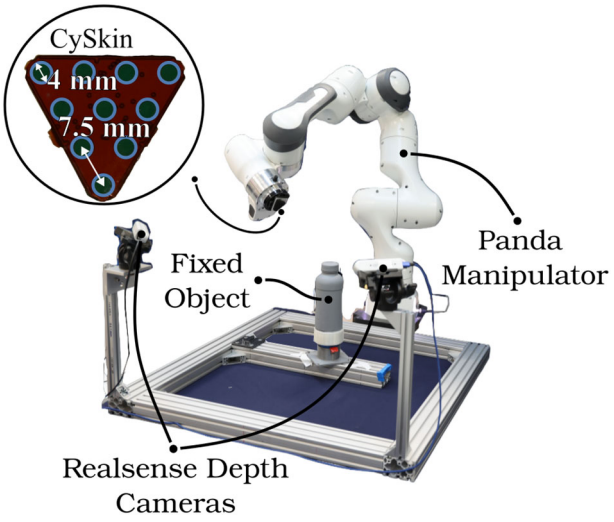


**Fig. 3** Experimental setup: **a** CySkin, the tactile sensor used; **b** positioning of the cameras and of the object with respect to the robot

Two Intel Realsense Depth Cameras D415 were used to get visual PCs, while tactile PCs were acquired using the CySkin [36] sensor, integrated into the end-effector of a 7-DoF Franka Emika Panda robot arm. The tactile sensor (see Fig. 3), composed of 10 capacitive transducers (4 mm in diameter, spaced 7.5 mm apart), provides 16-bit raw values proportional to the normal force on the transducers. Notably, as explained in 2.2, our approach stems from the assumption that a tactile sensor array retrieves an estimation of the local compliance; thus, the method applies without modifications if the adopted tactile sensor is one of those commonly used in previous works [2, 6, 13, 14, 18, 20, 23, 33].

For the object recognition task, to demonstrate the value of encoding physical properties in one representation, we selected objects with similar shapes but different compliance values. This set is shown in Fig. 4 and includes two bottles of identical shape but different compliance (one empty, one filled), a tennis ball, a sponge ball, and two cubes of the same size wrapped in rubber layers with different shores. The balls and cubes are made of a single material, while the bottles have stiffer plastic caps.

The visual PCs acquired using the two Realsense cameras are transformed with respect to a common reference frame, set to be the robot base and merged into a single PC. The resulting visual PCs have a number of points varying from 7863 for the sponge ball to 33676 for the bottle and represent a partial view of the objects.

Regarding the tactile data collection procedure, objects have been placed in front of the robot and fixed with a stand in a certain position as commonly done in literature [2, 18, 24]. Since we aim to capture objects' mechanical properties, in particular their compliance, the robot is commanded to probe the surface of each object as described in Sect. 2.1 with controlled force on different points of the objects. For each contact point, the robot approaches the object with the normal to the sensor oriented as the normal to the object surface at that point. Once contact is detected, the indentation process begins and the sensor feedback is recorded; the measurements obtained at the steady state are averaged to obtain $\bar{p}$ used in Sect. 2.1. Contact points are selected by randomly sampling points from the visual PC of the object. The number of measurements (i.e., $N$ in Sect. 2) is chosen to cover the whole object's surface as much as possible so as to detect compliance in different areas of the objects. Thus, 24 contact points are acquired for the two bottles (being the larger objects) and 17 for the others.

The target tactile sensor response reached at the steady state and used to compute the color has been calibrated to get an elastic deformation and to avoid motions of the object. For the given sensor, we selected a value of 4000 (being dimensionless as explained in Sect. 2.1).

## 4 Validation

To validate the cross-task nature of the proposed representation, we conduct an object classification and two control tasks.[3] We show that the representation is suitable to be used with a simple PointNet model [26] without requiring any custom learning architecture specific for visuo-tactile data.

In one of the control tasks, we show that the augmented PC can be used by the robot when commanded to follow a path defined on the object's surface. In this task, the robot pushes with a desired force that is tuned according to the local compliance of the object. In the other control task, the robot has to reach in clutter and exploits the augmented PC to change the path to follow in order to avoid stiffer obstacles.

### 4.1 Visuo-tactile object classification

We performed a visuo-tactile object classification task on the set of six objects shown in Fig. 4. As previously explained, some of the objects have similar shapes; therefore, an object classifier trained just on depth data cannot properly recognize objects on the basis of their geometrical characteristics alone. The same applies to objects having the same color and the same shape, e.g., the two bottles in our dataset. On the contrary, when considering also tactile information the classifier can take into account the compliance of the object to discriminate the items. Furthermore, as previously discussed, increasing the number of tactile interactions leads to a more accurate representation of the mechanical properties of the

[3] A video showing the robot performing the tasks is provided as supplementary material.

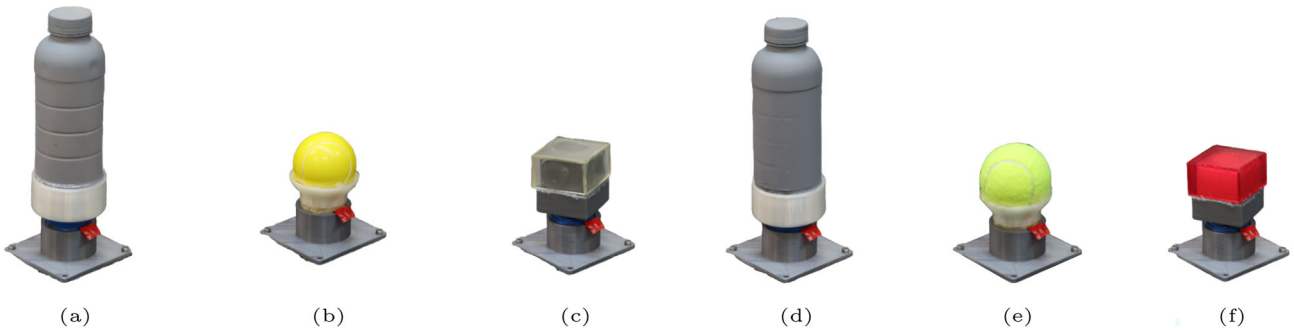


**Fig. 4** The six objects in our dataset: **a** empty bottle, **b** sponge tennis ball, **c** cube with a low-shore wrapping, **d** full bottle, **e** tennis ball, **f** cube with a high-shore wrapping

object. Therefore, we also analyze how this aspect impacts classification accuracy.

The classifier takes advantage of a simple PointNet architecture. In particular, we trained two networks, one on geometrical data and the other on visuo-tactile data, to show that the first has low prediction accuracy without compliance information. As explained in Sect. 3, we collected a single high-resolution visual PC and a low-resolution tactile PC for each object. Following the same notation used in Sect. 2, we refer to them as $\mathbb{V}_o$ and $\mathbb{T}_o$ where $o = 1, 2, \ldots, O$ and $O$ the number of objects. We generated the dataset to train and validate PointNet from these PCs.

We proceeded as reported in [2] to increase the dimension of the dataset. For the geometrical dataset, we sub-sampled 15 unique PCs starting from the single $\mathbb{V}_o$. Each one is composed of 512 points selected such that the different PCs do not contain the same points. The number of points has been chosen according to the default size of the input layer of PointNet.

In the case of $\mathbb{T}_o$, having a number of points ranging from 17 to 24 we created different tactile PCs having a number of points $G < |\mathbb{T}_o|$. The maximum number of different *augmented* PCs we can generate is given by the binomial coefficient $\binom{|\mathbb{T}_o|}{G}$. Therefore, by augmenting the visual PCs with tactile information, for each object, we can create a maximum number $N_{\mathbb{A}}$ of unique augmented PCs $\mathbb{A}_o$, with $N_{\mathbb{A}} = 15 \cdot \binom{|\mathbb{T}_o|}{G}$. From this set, we selected 500 PCs for each object as the dataset for the object classification task. In addition to this, to validate the effect of the number of touches on the classification accuracy, we generated four different versions of the dataset with $G = \{3, 7, 10, 13\}$. The dataset is then split into three parts: 70% is used for training, 10% for validation and 20% for testing. We also generated the same versions of datasets and trained another PointNet using the smooth representation to assess whether this leads to sensibly different results.

In every experiment, we trained PointNet for 20 epochs in batches of 16. During training, we performed dataset augmentation by implementing random rotations and by adding Gaussian noise as commonly done in literature [26].

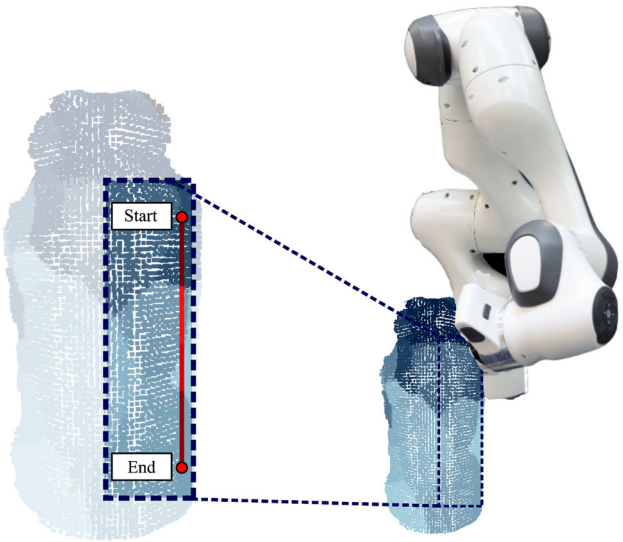


**Fig. 5** Path-following task. The robot is commanded to follow a straight line path defined on the bottle while adjusting the contact force depending on the local compliance of the object. The line that is superimposed on the augmented PC displays a different color according to the local compliance; a lighter color means higher compliance

## 4.2 Path following

In this validation, we propose a task in which the physical content of the augmented PC is used to control the robot in a path-following task operated at a controlled pushing force. In particular, the end-effector of the robot slides on the empty bottle (see Sect. 3) as if in the process of accomplishing an inspection task similar to [37]. The geometric information enables controlling the position of the end-effector, while the tactile information encoded in the PC is used to provide a different target force to be applied depending on the local compliance of the surface. Figure 5 shows the robot in contact with the bottle and the desired path of length 0.065 m

superimposed on the augmented PC. Similarly to the probing operation described in Sect. 2.1, we control the applied force on the basis of $\bar{p}$ being the mean response of the tactile sensors. In this respect, let $\mathbb{A}_{bottle}$ be the augmented PC of the bottle and $\mathbf{x}_r \in \mathbb{R}^3$ the current position of the robot end-effector pushing on the surface. While the robot is following the path, we can retrieve the local compliance value by querying $\mathbb{A}_{bottle}$ at the contact point *on the fly*. In particular, we define the attribute $c(\mathbf{x}_r)$ as the *tactile color* of $\mathbb{A}_{bottle}$ at the contact point $\mathbf{x}_r$. Then, given a maximum pushing force $\bar{p}^*_{max}$, we define the desired force applied by the robot as:

$$\bar{p}^* = c(\mathbf{x}_r)\frac{\bar{p}^*_{max}}{c_{max}}, \tag{5}$$

where $c_{max}$ is the color value corresponding to the stiffest points in $\mathbb{A}_{bottle}$, those that deform the least. Therefore, the robot applies a force inversely proportional to the measured compliance of the object. Particularly in this task, we expect the smooth representation of the augmented PC to be advantageous, preventing abrupt variations when the robot moves between areas with different compliance values. As a consequence, we also execute the path-following task using the smoothened version of $\mathbb{A}_{bottle}$. The main difference lies in the fact that in this case the target value of $\bar{p}^*$ is expected to vary without big steps along the path.

### 4.3 Reaching in clutter

When reaching in clutter, contact between the robot and the environment is inevitable and can be harmful for the robot or the environment itself. Being able to selectively avoid or exploit contacts can be beneficial for the robot to prevent damage or to improve performance in stowing tasks. For this validation, we propose a toy example in which the robot has to reach a target point while moving in clutter and the augmented PC is used to change the path to avoid stiffer obstacles.

In particular, as shown in Fig. 6, the robot needs to move from point A to point B in the presence of a stiff metal pole and a soft prism made of sponge occupying fixed positions in the environment and it cannot avoid both of them because of their configuration. The robot is equipped with an extra link mounted on the end-effector which is fully covered with the tactile sensors described in Sect. 3, similarly to what was used in [38].

The augmented PC of the two objects is uniform in the plane of the task execution. As a consequence, using the smooth interpolation for the augmented PC does not raise any different behavior in this task.

The minimum distance path the robot can follow is given by the line connecting the two points and the underlying control scheme is the one introduced in [39]. However, during the

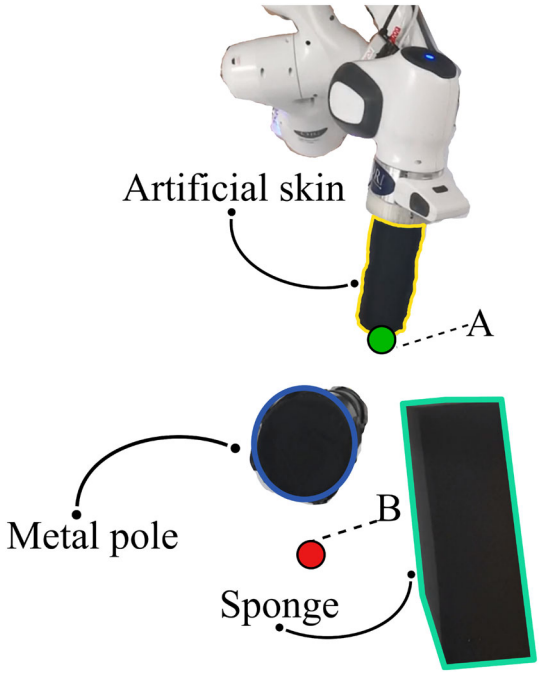


**Fig. 6** Reaching in clutter task. The experimental setup for the task execution is shown. The top view shows the robot equipped with an artificial skin end-effector that is commanded to reach a target point in the presence of a metal pole and a sponge prism whose profiles are highlighted in dark blue and light green, respectively. The starting and ending points of the task are highlighted as well in green and red, respectively

motion it is computed the distance between the end-effector position $\mathbf{x}_{ee}$ and the object centroid $\mathbf{x}_i$, with $\mathbf{x}_{ee}$ and $\mathbf{x}_i \in \mathbb{R}^3$ and $i \in \{\text{pole, prism}\}$. The robot deviates from the minimal distance path when this distance is smaller than a threshold value $\epsilon_i$, which is set differently for each object based on their *tactile color* and is given by:

$$\epsilon_i = c_i \frac{r_{max}}{c_{max}}, \tag{6}$$

with $r_{max}$ being the maximum threshold distance allowed between the robot and the centroid of the stiffest object, $c_{max}$ the attribute associated to the lowest compliance of the objects in the environment and $c_i$ that of the object $i$. For this task, we set $r_{max} = 0.07\ m$.

The deviation $\mathbf{x}_{dev}$ from the current robot position $\mathbf{x}_{ee}$ is proportional to the *tactile color* of the objects, is tangent to the object's surface, lays on the trajectory plane and is computed as:

$$\mathbf{x}_{dev} = \gamma c_i \frac{\mathbf{x}_{ee} - \mathbf{x}_i}{\|\mathbf{x}_{ee} - \mathbf{x}_i\|} \times \mathbf{e_z}, \tag{7}$$

where $\mathbf{x}_{dev} \in \mathbb{R}^3$, $\mathbf{e_z}$ is the normal versor to the trajectory plane and $\gamma$ is a scaling factor. The behavior of the robot when exploiting the augmented PC is compared with respect

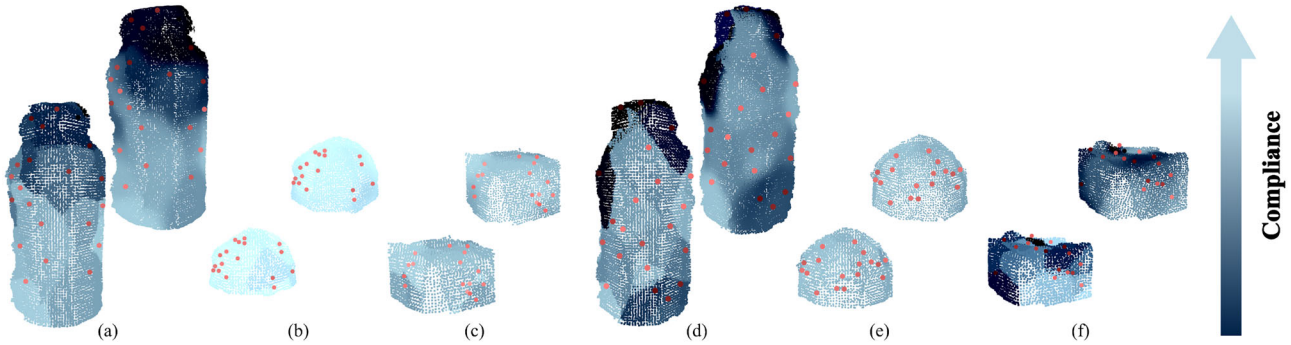


**Fig. 7** The augmented PCs for the six objects in our dataset with both the cluster-based (bottom row) and the smooth (upper row) interpolations: **a** Empty bottle, **b** sponge tennis ball, **c** cube with low-shore wrapping, **d** full bottle, **e** tennis ball, **f** cube with a high-shore wrapping. Red dots represent the contact points. For visualization purposes, we filtered the PCs with a Voxel filter in order to reduce the number of points

to two baselines: the reactive control proposed in [39] and a joint impedance control.

## 5 Results

Figure 7 shows the augmented PCs of the six objects in Fig. 4 with both the interpolation strategies. The color map in the figure is represented such that darker colors are related to stiffer areas. As is visible, the representations correctly grasp qualitative insights on the compliance distribution in the different parts of the objects. In particular, cubes are stiffer toward the edges and the vertices and this is more evident for the one wrapped with high-shore rubber. The empty bottle is overall more compliant than the filled one, as can be deducted from the lighter color of the body associated with the PC. Also, the compliance over the body of the bottles is not uniform and decreases close to the curvatures. In both the augmented PCs of the bottles, this compliance distribution is not grasped everywhere due to the limited number of touches.

The tennis ball is correctly represented as stiffer than the sponge ball. It must also be noted that the compliance of the balls is mostly similar across the whole surface since the local shape and curvature of the objects are everywhere the same. Both strategies yield similar results, but smooth interpolation better represents the actual compliance distribution by avoiding uniform segments and providing more accurate transitions. However, smooth PC generation takes an average of 3.1 sec, i.e., 30 times longer using an Intel Core i7-12700 H CPU.[4]

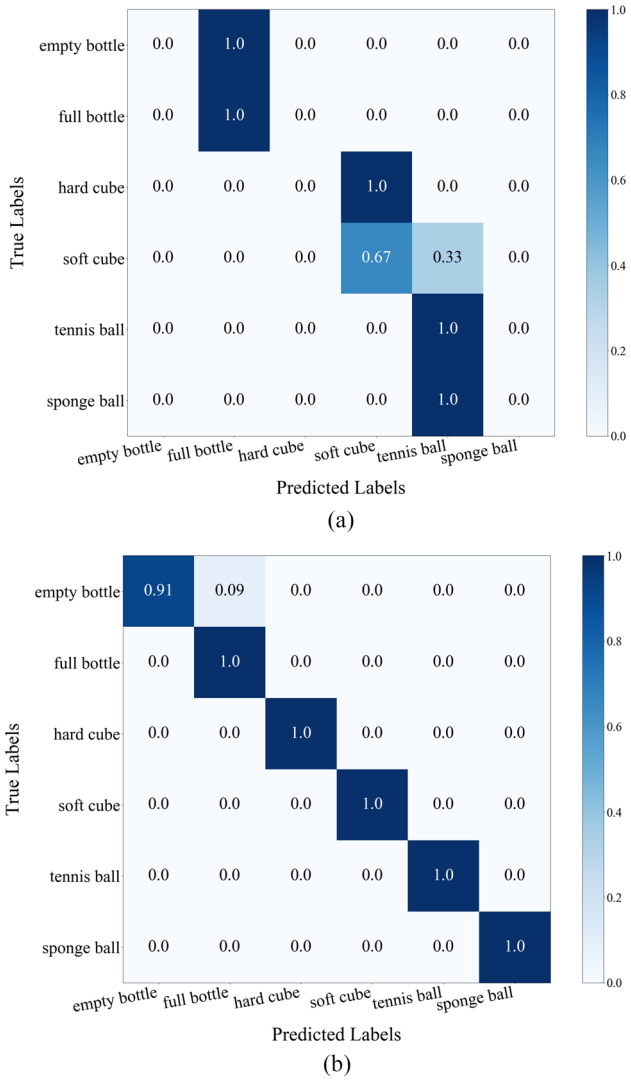


**Fig. 8** Confusion matrices obtained: **a** with training PointNet with just visual PCs, **b** with training PointNet with the augmented PCs considering 13 touches and using the cluster-based interpolation

[4] Experimental analyses conducted with GP-based representations yielded worse performance both time-wise and in the representation. Results are presented in the Appendix.

We note that extending tactile information based on object color, as in [19, 24, 25], cannot allow a proper encoding of the compliance of the object. For objects made of a single material and uniform color, compliance can indeed vary across the surface due to local geometry, as demonstrated by the bottles and cubes in our dataset.

### 5.1 Object classification

In Fig. 8 we present the confusion matrices computed by applying PointNet to the test set of visual PCs and visuo-tactile augmented PCs with the cluster-based interpolation. It can be seen that the accuracy sensibly improves when tactile cues are considered. PointNet correctly distinguishes the objects in less than 45% of the cases if just the geometrical content of the objects is available. From the confusion matrix in Fig. 8a, it is clear that objects having similar shapes are not correctly classified. As an example, the hard cube and the sponge ball are wrongly classified; similarly, the empty bottle is always confused with the filled one. On the contrary, by using the augmented PC with both the interpolation strategies, an almost perfect classification score can be achieved when considering the mechanical properties obtained with 13 touches. As a matter of fact, in this case, tactile information is used by the network to discriminate objects having similar shapes. These results are reported in Fig. 8b for the PointNet trained with the augmented representation that exploits the cluster-based interpolation.

We also evaluated how the classification accuracy is affected by the number of touches. As explained in Sect. 2.2, increasing the number of touches leads to a more accurate representation of the compliance distribution over the surface. Therefore, we expect that adding more tactile information directly improves the performance of PointNet. The result of this analysis is presented in Table 1, where it can be seen how the classification accuracy increases with the number of touches when using the two different interpolation methods. With respect to the score obtained with 0 touches (i.e., visual PCs only), a significant change is already reached when considering 3 touches as more than 25% improvement in prediction ability is gained when considering the dataset with the smooth representation.

In this task, cluster-based interpolation performs better than smooth interpolation. This is because the sub-sampled PCs in this representation have less variation in *tactile color*, making it easier to learn how to distinguish the objects.

Table 2 compares our method with three state-of-the-art approaches to visuo-tactile object recognition. In [28], the authors use a dataset from [40]—tactile data from a glove paired with images—and an ML pipeline that fuses these features in latent space, achieving high accuracy even with limited tactile input. Similarly, Wei et al. [5] employ contrastive learning to classify fabrics, achieving their best performance on the dataset from [41], which combines RGB images with GelSight tactile data [42]. Liu et al. [18] use joint kernel sparse coding to classify five object classes using 20 images and 10 tactile samples per object. Unlike these works, we use a smaller dataset, as our goal is to demonstrate the effectiveness of the augmented representation across tasks, not solely classification. Despite using a simpler tactile sensor (fewer transducers than [18, 40] and lower resolution than GelSight in [41]), our pipeline achieves comparable accuracy. Moreover, our representation supports both classification and control tasks, a level of cross-task generality not directly attainable with latent space encodings used in prior methods.

**Table 1** Accuracy of PointNet with an increasing number of tactile samples

| # Of Touches | 0 | 3 | 7 | 10 | 13 |
|---|---|---|---|---|---|
| Cluster | 44.44 | 79.34 | 94.84 | 97.35 | 98.55 |
| Smooth | 44.44 | 70.62 | 86.37 | 91.28 | 97.78 |

### 5.2 Path following

Figure 9 shows the results of the path-following task executed by using the two representations. In particular, the top row in Fig. 9 highlights at which points of the path the compliance changes (emphasized with the yellow dots), while in the bottom one the variation does not present similar sharp changes since the smooth representation is adopted. The $x$-axis reports the length of the path, while the output $\bar{p}$ of CySkin sensor and the desired $\bar{p}^*$ are reported on the $y$-axis. For this experiment, as in Sect. 3, we want $\bar{p}^*_{max}$ to be such that a limited elastic deformation is imposed over the object and is set to 2000. Also, the same control gains were used in the two experiments.

As correctly encoded in the augmented PC shown in Fig. 7, the bottle is stiffer closer to the cap because of the curvature, more compliant in the central part of the body and slightly stiffer toward the bottom where it is constrained. As visible from the two plots in Fig. 9, the robot starts from the area having the smallest compliance, thus applying the maximum pushing force. Then, while sliding at 1.5 $\text{mms}^{-1}$, the information on the local compliance is updated and used to adjust the setpoint according to Eq. (5). After 0.006 m, in Fig. 9 (top), a change of compliance is detected and the robot reduces the pushing force. Similarly, at 0.023 m the robot moves to a slightly more compliant area of the object. In the final part of the path, at approximately 0.06 m, the robot is sliding to a stiffer region (visible from the darker color in the augmented PC), thus increasing the force setpoint which is tracked by the robot. With the smooth representation, the changes in the compliance are less abrupt, hence produc-

**Table 2** Comparison of different visuo-tactile object recognition approaches from the literature and ours

| Method | # of Objects | max # of Touches | Accuracy |
|---|---|---|---|
| Babadian et al. [28] | 26 | 8 | ∼100 |
| Wei et al. [5] | 118 | 10 | ∼91 |
| Liu et al. [18] | 18 | 10 | ∼91 |
| **Ours** | 6 | 13 | ∼99 |

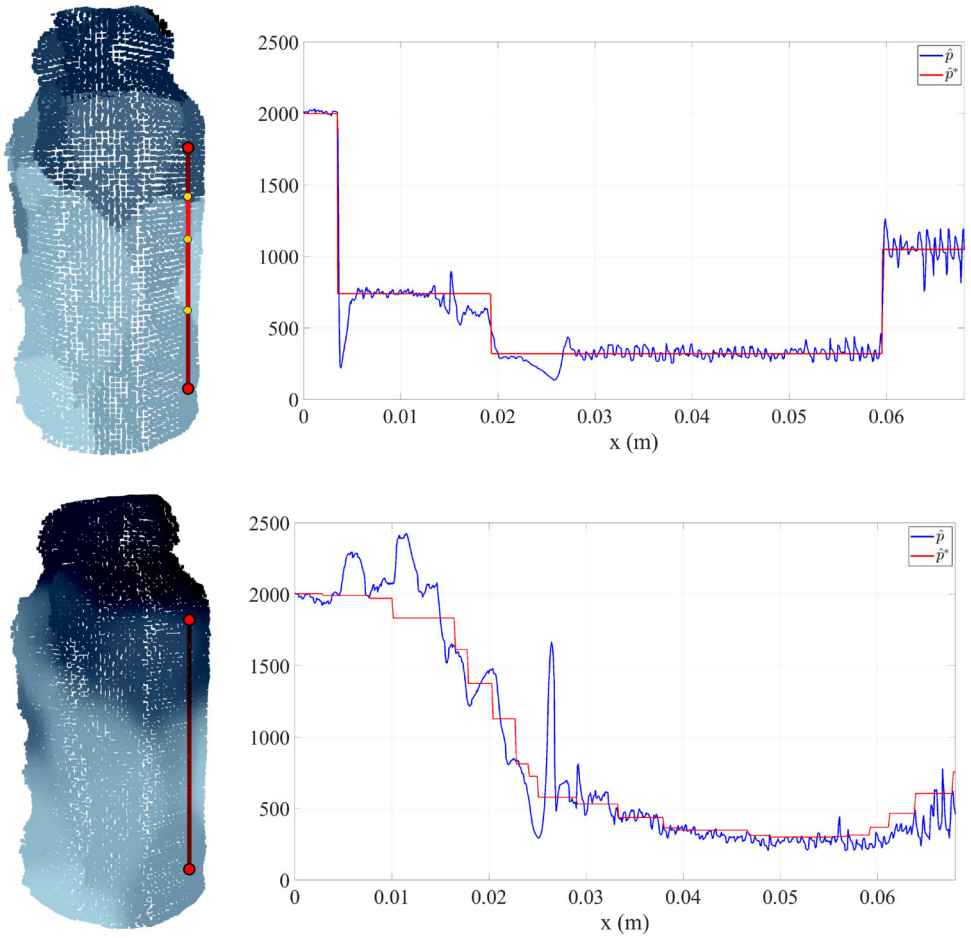


**Fig. 9** Results of the path-following task. The figure shows the desired path superimposed on the augmented PCs of the bottle obtained with the cluster-based (top) and the smooth (bottom) interpolation. The corresponding plots show the evolution of $\bar{p}$ (measured) and $\bar{p}^*$ (desired) along the path when the two representations are used

ing a more realistic force profile. In terms of force tracking, the smooth representation with respect to the cluster-based one leads to higher average error over the path, respectively, 120.86 and 52.92 sensor units, but smaller overshoots since the controller needs to adapt more frequently though to smaller quantities. The only overshoot visible at 0.025 m in the bottom plot is due to the fact that the robot loses contact with the bottle because of its shape at that point.

The augmented representation in this task allows for an *on the fly* tuning of the setpoint, thus avoiding the need for a pre-planned force profile.

### 5.3 Reaching in clutter

As can be seen in the subplots (b) to (e) of Fig. 10, the robot manages to reach the target point in all the analyzed cases, but the followed path is different. As a matter of fact, when using the augmented PCs, the robot is able to reach the target point without touching the metal pole. The robot deviates from its linear path because of the deviation induced by the proximity to the pole as given by Eq. (7); the deviation induced by the sponge prism is such that its contribution is negligible, and as a result, the robot avoids the pole and collides just with the prism. Conversely, with both the other controllers, the robot collides with the pole, raising undesired and potentially harmful contacts. As a consequence, when the augmented representation is not used, the mean exchanged contact inten-

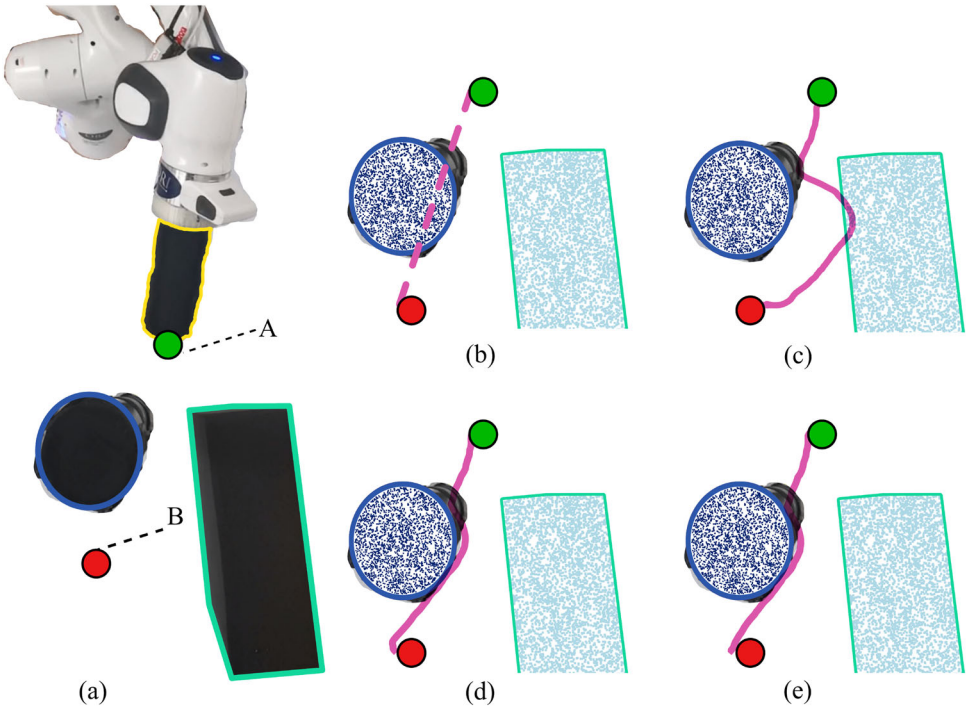


**Fig. 10** Results of the reaching in clutter task. **a** The experimental setup is shown and the starting and ending points are highlighted in green and red, respectively. **b** The minimum distance path. **c** The path followed by the robot when using the proposed method. **d** The path followed by the robot when using the reactive controller proposed in [39]. **e** The path followed by the robot when using the joint impedance controller with very low gains. The point clouds of the objects are overimposed and their color follows the same scale as in Fig. 7

sity as measured by the tactile sensors is 55.8% higher when using the reactive tactile-based control and 33.5% higher when the joint impedance controller with very low gains is adopted. However, by accommodating for safe interaction with the environment, the robot is in contact with the sponge for longer thus the task is 40% slower with respect to the other cases.

In this task, due to the simplicity of the objects involved, we would like to remark that using the smooth representation does not present any advantage with respect to the other one. Moreover, as can be drawn from these results, without the augmented representation to completely avoid the pole, the entire path should be planned in advance.

## 6 Conclusion

In this letter, we introduced a cross-task representation utilizing the PC data structure to encode visuo-tactile cues. Specifically, a PC acquired with cameras is used to represent the geometry of the objects, while tactile sensing is used to augment it with physical information via color mapping, capturing compliance distribution across the object's surface. We showed how the proposed representation can be used to enhance robot abilities across different tasks, such as object classification, by using a state-of-the-art classifier like PointNet directly, and control tasks including path following and reaching in clutter. As shown in Fig. 7, a limitation of this representation is the fact that the color extension relies on the explored regions. Future research will focus on selecting the best points to explore, aiming for a more accurate object representation with few touches.

The proposed methodology can be extended to embed other physical properties that may be relevant for other tasks, e.g., friction or temperature, as additional color channels. This aspect will be investigated in future extensions of the work. It is important to note that this paper considers a single force and a simple model for calculating local compliance. Clearly, increasing the pushing force introduces nonlinear behaviors (such as saturation) which depend on the object's material and local shape. Nonetheless, the PC can still be used as the estimation of the local compliance can be improved and the property extended as explained in Sect. 2.2.

## Appendix

### Gaussian process based representation

In this section, we will discuss the Gaussian Process (GP) based representation of the data. As a matter of fact, GP-based representations are widely used [12, 16, 17] and could serve as a way of combining mechanical and visual properties [30, 31]. For a formal introduction to GP, we refer the reader to [29].

Having a tactile PC, we train a GP to learn a mapping between the geometric feature and the tactile color. Having this model, we predict the tactile color for the points belonging to the visual PC. Given the limited number of tactile points ($N \leq 24$), the resulting augmented representation fails

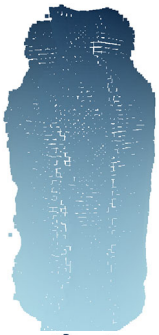

**Fig. 11** GP-based representation of the empty bottle

**Fig. 12** GP-based representation of the empty bottle using an artificially increased tactile PC

to extend the tactile color. As a matter of fact, the resulting PC has a tactile color that smoothly varies from the plastic cap to the bottom in a linear fashion as if the model overfits the geometrical feature rather than fitting the tactile properties. In Fig. 11, it can be seen that the tactile color is spread such that points on the top are darker and points on the bottom have a lighter color associated with them. As expressed in Sect. 3, since a stand is used to keep the object in place, the bottle is expected to be less compliant also at the bottom. This entails that the trained GP model disregards the tactile readings.

This issue is likely related to the limited amount of data the model is trained on. Indeed, to further prove this, we artificially increased the tactile PC and re-trained the model. To this end, we searched for the points in the visual PC belonging to a 3 cm neighborhood of each point of the tactile PC. This way we increase the number of points of the tactile PC and obtain a bigger number of training samples. We chose this size for the radius as it is approximately the size of the tactile sensor and, for this analysis, we assumed that all the points touched by the sensor have the same physical properties. Then, we generated the GP-based augmented representation and got a result that is very similar to the smooth representation, as can be seen in Fig. 12.

The generation of the augmented representation in this case is highly computationally intensive and heavily depends on the number of points of the tactile PC. In this case, once the neighboring points were found, the artificially increased PC was downsampled otherwise the training of the GP model would have been unfeasible (given the capabilities of the laptop used for this work, i.e., Intel Core i7-12700 H CPU). The voxel size was varied, and the best representation (Fig. 12) was obtained with a voxel size of 0.004 m. This way, the computation takes ∼132 s and yields a result that is comparable to the smooth representation, which conversely takes ∼6.5 s and is way less memory-intensive. The computation efficiency could certainly be improved if a smaller radius is used for the neighbor search; however, it is evident that this approach requires further analyses since more hyperparameters are involved and is not justified from a computational point of view even considering that the results are similar to the ones obtained with the other more efficient representations.

In conclusion, even though the GP-based representation may also serve as a tool to encode the local compliance of the object's surface, we found out that for our application it is not suitable because of its greater computational cost; more intensive engineering burden and the retrieved results are not significantly better than the ones obtained with the smooth representation.

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1007/s11370-025-00628-8.



## References

1. Porcu E, Keitel C, Müller MM (2014) Visual, auditory and tactile stimuli compete for early sensory processing capacities within but not between senses. Neuroimage 97:224–235
2. Parsons C, Albini A, De Martini D, Maiolino P (2022) Visuo-tactile recognition of partial point clouds using pointnet and curriculum learning: enabling tactile perception from visual data. IEEE Robot Autom Magaz 30(3):69–78
3. Falco P, Lu S, Cirillo A, Natale C, Pirozzi S, Lee D (2017) "Cross-modal visuo-tactile object recognition using robotic active exploration," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 5273–5280
4. Murali PK, Wang C, Lee D, Dahiya R, Kaboli M (2022) Deep active cross-modal visuo-tactile transfer learning for robotic object recognition. IEEE Robot Autom Lett 7(4):9557–9564
5. Wei F, Zhao J, Shan C, Yuan Z (2022) "Alignment and multi-scale fusion for visual-tactile object recognition," in *2022 International joint conference on neural networks (IJCNN)*. IEEE, 1–8
6. Björkman M, Bekiroglu Y, Högman V, Kragic D (2013) "Enhancing visual perception of shape through tactile glances," in *2013 IEEE/RSJ International conference on intelligent robots and systems*. IEEE, 3180–3186

7. Smith E, Calandra R, Romero A, Gkioxari G, Meger D, Malik J, Drozdzal M (2020) 3d shape reconstruction from vision and touch. Adv Neural Inf Process Syst 33:14193–14206
8. Smith E, Meger D, Pineda L, Calandra R, Malik J, Romero Soriano A, Drozdzal M (2021) Active 3d shape reconstruction from vision and touch. Adv Neural Inf Process Syst 34:16064–16078
9. Suresh S, Si Z, Mangelson JG, Yuan W, Kaess M (2022) "Shapemap 3-d: Efficient shape mapping through dense touch and vision," in *2022 International conference on robotics and automation (ICRA)*. IEEE, 7073–7080
10. Bimbo J, Seneviratne LD, Althoefer K, Liu H (2013) "Combining touch and vision for the estimation of an object's pose during manipulation," in *2013 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 4021–4026
11. Pecyna L, Dong S, Luo S (2022) "Visual-tactile multimodality for following deformable linear objects using reinforcement learning," in *2022 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 3987–3994
12. Ottenhaus S, Renninghoff D, Grimm R, Ferreira F, Asfour T (2019) "Visuo-haptic grasping of unknown objects based on gaussian process implicit surfaces and deep learning," in *2019 IEEE-RAS 19th international conference on humanoid robots (Humanoids)*. IEEE, pp. 402–409
13. Falco P, Lu S, Natale C, Pirozzi S, Lee D (2019) A transfer learning approach to cross-modal object recognition: from visual observation to robotic haptic exploration. IEEE Trans Rob 35(4):987–998
14. Murali PK, Dutta A, Gentner M, Burdet E, Dahiya R, Kaboli M (2022) Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter. IEEE Robot Autom Lett 7(2):4686–4693
15. Villalonga MB, Rodriguez A, Lim B, Valls E, Sechopoulos T (2021) "Tactile object pose estimation from the first touch with geometric contact rendering," in *Conference on robot learning*. PMLR, 1015–1029
16. Dragiev S, Toussaint M, Gienger M (2011) "Gaussian process implicit surfaces for shape estimation and grasping," in *2011 IEEE international conference on robotics and automation*. IEEE, pp. 2845–2850
17. Rustler L, Lundell J, Behrens JK, Kyrki V, Hoffmann M (2022) Active visuo-haptic object shape completion. IEEE Robot Autom Lett 7(2):5254–5261
18. Liu H, Yu Y, Sun F, Gu J (2016) Visual-tactile fusion for object recognition. IEEE Trans Autom Sci Eng 14(2):996–1008
19. Bhattacharjee T, Shenoi AA, Park D, Rehg JM, Kemp CC (2015) "Combining tactile sensing and vision for rapid haptic mapping," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, pp. 1200–1207
20. Shenoi AA, Bhattacharjee T, Kemp CC (2016) "A crf that combines touch and vision for haptic mapping," in *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2255–2262
21. Liu Y, Cong Y, Sun G, Ding Z (2022) Lifelong visual-tactile spectral clustering for robotic object perception. IEEE Trans Circuit Syst Video Technol 33(2):818–829
22. Yang F, Feng C, Chen Z, Park H, Wang D, Dou Y, Zeng Z, Chen X, Gangopadhyay R, Owens A et al (2024) "Binding touch to everything: Learning unified multimodal tactile representations," In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, , pp. 26 340–26 353
23. Rosales C, Ajoudani A, Gabiccini M, Bicchi A (2014) "Active gathering of frictional properties from objects," in *2014 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, pp. 3982–3987
24. Le TN, Verdoja F, Abu-Dakka FJ, Kyrki V (2021) Probabilistic surface friction estimation based on visual and haptic measurements. IEEE Robot Autom Lett 6(2):2838–2845
25. Yao S, Hauser K (2023) "Estimating tactile models of heterogeneous deformable objects in real time," in *2023 IEEE international conference on robotics and automation (ICRA)*, pp. 12 583–12 589
26. Qi CR, Su H, Mo K, Guibas LJ (2017) "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660
27. Rouhafzay G, Cretu A-M, Payeur P (2020) Transfer of learning from vision to touch: a hybrid deep convolutional neural network for visuo-tactile 3d object recognition. Sensors 21(1):113
28. Babadian RP, Faez K, Amiri M, Falotico E (2023) Fusion of tactile and visual information in deep learning models for object recognition. Inf Fusion 92:313–325
29. Williams C, Rasmussen C (1995) "Gaussian processes for regression," *Advances in neural information processing systems*, 8
30. Caccamo S, Güler P, Kjellström H, Kragic D (2016) "Active perception and modeling of deformable surfaces using gaussian processes and position-based dynamics," in *2016 IEEE-RAS 16th international conference on humanoid robots (Humanoids)*. IEEE, pp. 530–537
31. Salman H, Ayvali E, Srivatsan RA, Ma Y, Zevallos N, Yasin R, Wang L, Simaan N, Choset H (2018) "Trajectory-optimized sensing for active search of tissue abnormalities in robotic surgery," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 5356–5363
32. Border R, Gammell JD, Newman P (2018) "Surface edge explorer (see): Planning next best views directly from 3d observations," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 6116–6123
33. Kangro J, Traversaro S, Pucci D, Nori F (2017) "Skin normal force calibration using vacuum bags," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 401–406
34. Chazal F, Guibas LJ, Oudot SY, Skraba P (2011) Scalar field analysis over point cloud data. Discret Comput Geomet 46(4):743–775
35. Ikotun AM, Ezugwu AE, Abualigah L, Abuhaija B, Heming J (2023) K-means clustering algorithms: a comprehensive review, variants analysis, and advances in the era of big data. Inf Sci 622:178–210
36. "Cyskin," https://www.cyskin.com/
37. Sadeghi Aval Shahr M, Abbasimoshaei A, Schwartz C, Kern TA (2024) "Designing, fabricating, and analyzing the whisker sensor for autonomous surface defect detection," in *EUROSENSORS XXXVI*
38. Caroleo G, Giovinazzo F, Albini A, Grella F, Cannata G, Maiolino P (2024) "A proxy-tactile reactive control for robots moving in clutter," in. IEEE/RSJ international conference on intelligent robots and systems (IROS) 2024:733–739
39. Albini A, Grella F, Maiolino P, Cannata G (2021) Exploiting distributed tactile sensors to drive a robot arm through obstacles. IEEE Robot Autom Lett 6(3):4361–4368
40. Sundaram S, Kellnhofer P, Li Y, Zhu J-Y, Torralba A, Matusik W (2019) Learning the signatures of the human grasp using a scalable tactile glove. Nature 569(7758):698–702
41. Yuan W, Wang S, Dong S, Adelson E (2017) "Connecting look and feel: Associating the visual and tactile properties of physical materials," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5580–5588
42. Johnson MK, Adelson EH (2009) "Retrographic sensing for the measurement of surface texture and shape," in *2009 IEEE conference on computer vision and pattern recognition*. IEEE, pp. 1070–1077