

# A Local Filtering Technique for Robot Skin Data

Alessandro Albini<sup>1</sup>, Giorgio Cannata<sup>2</sup> and Perla Maiolino<sup>1</sup> <sup>\*</sup>

February 2, 2022

## Abstract

Algorithms for tactile data processing heavily rely on the generation of tactile images representing the contact pressure distribution. This approach allows to exploit existing algorithms for image processing, but requires the integration of the tactile elements on a flat surface. Robot skin technologies introduce challenges related to the non-regular distribution of the tactile elements and the non-planar surface over which they are integrated. In this paper, we present a method to address these challenges by developing a local filtering technique directly applicable on large-area tactile sensing systems. The proposed filter can process the contact distribution without the need of intermediate steps that are required in the typical method of generating a tactile image. We particularly focus on the design of a filter to detect sharp variations in the contact distribution, i.e. *edges*. The approach is validated in a task of planar contour following performed using a robot equipped with two different end-effectors (planar and non-planar) sensorized with large-area tactile sensing technology. Additional experiments have been performed to evaluate strengths and limitations of the proposed approach with respect to tactile image-based data processing techniques.

## 1 Introduction

Tactile sensors for robots are crucial in the realization of complex tasks related to manipulation and control [1], objects or textures recognition [2], and touch based control [3].

Technological advancements have recently enabled the realization of complex large-area tactile systems, namely *robot skins*, composed of thousands of distributed transducers which can be conformed to curved surfaces and can potentially cover the whole robot body [4–8]. Despite the large number of tactile sensors technologies available, there is still a lack of standards at hardware level. Indeed, tactile sensing systems mainly differ in terms of: (i) transduction principles; (ii) spatial resolution (possibly

---

<sup>\*</sup>1First Author and Third Author are with the Oxford Robotics Institute (ORI), University of Oxford, UK. [alessandro@robots.ox.ac.uk](mailto:alessandro@robots.ox.ac.uk); [perla@robots.ox.ac.uk](mailto:perla@robots.ox.ac.uk)

<sup>†</sup>2The second Author is with the Department of Informatics, Bioengineering, Robotics and Systems Engineering (DIBRIS), University of Genoa, Italy. [giorgio.cannata@unige.it](mailto:giorgio.cannata@unige.it)

non-regular); (iii) conformability on complex surfaces [9]. This makes it challenging to design tactile data processing algorithms that can be applied on such a large variety of devices.

To cope with the above differences, tactile data processing is usually performed by transforming raw sensor measurements into representations abstracted from the specific sensing technology adopted. In particular, a widely exploited technique consists in mapping raw sensor data into a *tactile image*, i.e. a planar representation of the contact pressure distribution [2]. This transformation is performed by resampling a planar distribution of tactile sensors with a regular grid. The value of each pixel forming the image is then computed by interpolating the raw measurements provided by the tactile elements. The main advantage of this method is that algorithms developed for images can be directly applied to the tactile sensing domain when the shape of the contact distribution needs to be processed [2]. In the literature, several works proposed the use of tactile images generated from planar tactile sensing devices for contact shape classification [2, 10–13] or processing and features extraction [14–18]. The major limitation of tactile images is that they can only be directly generated from sensors distributed on a planar surface.

An example of a robot skin system is given in Figure 1(a). As visible from the picture, the tactile sensing elements are distributed over a curved surface, impeding the ability to *directly* generate a planar tactile image from this sensors distribution.

In our previous work [19], we showed that the application of tactile images is still valid for a robot skin system, after applying a set of geometric transformations creating a flat representation of the skin geometry. Although the method is effective, it requires several processing steps. Furthermore, the planarization of the robot skin system can introduce distortions in the resulting tactile image and further processing may be required to cope with them. The use of tactile images to process robot skin data is necessary since, to the best of our knowledge, there is no direct way to process the tactile measurements when sensors are placed over complex surfaces with non-regular distribution.

The contribution of this paper is to present a technique to directly process the contact pressure distribution captured by a robot skin system without the need of intermediate steps required to generate tactile images. Therefore, the proposed method is independent of the spatial arrangement of the tactile elements and the surface over which they are integrated. Moreover, it is based on the idea of extending the concept of local processing, which is at the core of the majority of filtering techniques for images, to robot skin data. This extension is not straightforward: while in an image the pixel’s neighbourhood has a fixed structure, tactile elements composing a robot skin system do not have this property due to the non-regular spatial arrangement of the sensors. In particular, within the scope of this paper, the specific problem of processing the contact distribution to extract edges is addressed. The detection of this geometric feature is important for dexterous manipulation, as demonstrated by previous works addressing this problem with the use of tactile images [14, 15, 17, 18], by using optical tactile sensing technologies which directly provide an image of the contact distribution [16] or by using raw sensor measurements [20, 21]. However, the latter are based on Bayesian active exploration and the processing of contact shape (which is the focus of this paper) is not considered.

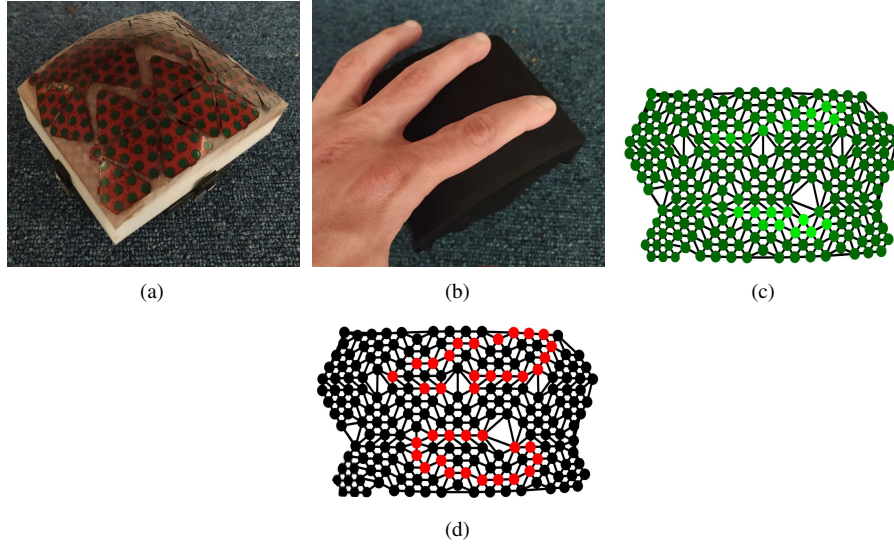


Figure 1: The proposed method of edge detection filtering applied to a non-planar robot skin patch. (a) The CySkin technology integrated on a curved surface. Green circles correspond to independent pressure transducers. (b) A contact with two fingers. (c) Robot skin response (top view); darker dots correspond to lower responses. (d) Filtered robot skin response (top view).

Figure 1 illustrates the outcome of the proposed approach, where the contours of the contact distribution generated by two fingers pressing over a non-planar surface are correctly extracted. The idea behind the approach is to represent the surface covered with the robot skin system as a mesh describing local connections among the tactile elements. A filtering technique is then designed on top of this mesh. It is worth noting that, in the literature, several techniques have been proposed to perform filtering operations on meshes, such as smoothing [22], features extraction [23] or denoising [24]. However, while previous works aimed at filtering the mesh geometry or at finding geometric features in the mesh representing the surface, in this paper the goal is not to filter the mesh geometry itself but a contact pressure distribution applied on it. The proposed technique is validated in a task of planar contour following, where robot skin is integrated on two types of end-effectors: (i) a small flat end-effector with 20 tactile elements; (ii) a large and non-planar end-effector with 211 tactile elements. Furthermore, a comparison between the proposed method and an approach based on tactile images is presented, followed by an analysis of strengths and weaknesses of the proposed approach.

The manuscript is structured as follows. Section 2 introduces some preliminary definitions. Section 3 describes the proposed filtering technique. The edge detection filter is described in Section 4. Section 5 describes the experiments performed to validate the proposed method. Results and discussions are presented in Section 6. Conclusion follows.

## 2 Preliminaries and Problem Description

The robot skin system is composed of  $N$  distributed transducers, called *taxels*, mounted on a rigid non-planar manifold  $S$  with a non-regular spatial distribution. The taxels are assumed to be covered or embedded in a continuous and deformable material (see Figure 2 as a reference). The position of each taxel  $\mathbf{t}_i \in \mathbb{R}^3$  is assumed to be known with respect to a given reference frame. As sketched in Figure 2, a pressure distribution applied on the top surface, namely  $\Gamma$ , produces a deformation. Let us consider a continuous function  $p(\cdot)$  related to the deformation of  $\Gamma$ . Thus, we can define the response of the  $i$ -th taxel  $p_i = p(\mathbf{t}_i)$  equivalent to the spatial sampling of the function  $p(\cdot)$ . The structure of  $p(\cdot)$  is not known; however, its geometrical characteristics can be analyzed through the knowledge of its sampled form.

The taxel responses  $p(\mathbf{t}_i)$ , along with their positions in the space  $\mathbf{t}_i$ , provide a discrete information of the pressure distribution applied on  $S$  at a given time instant. The assumptions introduced above refer to a class of robot skin systems composed of a number of independent tactile elements rigidly attached to the robot surface. Several examples of technologies matching these assumptions can be found in the literature [4–6, 8, 25–27].

The problem addressed in this paper is to introduce a filtering technique to compute the values  $\hat{p}(\mathbf{t}_i)$ , encoding edges in the contact distribution. Within the scope of this work, an *edge* is intended as a sharp variation in the contact shape.

Although edges could be computed using existing methods based on tactile images, the procedure would require to: (i) transform the non-planar skin geometry into a planar structure and map the taxel measurements on it; (ii) resample the planar skin geometry to obtain a regular grid; (iii) compute the pressure value associated to each pixel in the grid; (iv) apply an edge detection filter; (v) invert the transformations by back-projecting the filtered pixels both to the planar and non-planar representation of the skin.

In contrast with existing methods, this paper presents an approach allowing to compute  $\hat{p}(\mathbf{t}_i)$  directly processing the raw measurements  $p(\mathbf{t}_i)$ , without the need of any transformation step. In image processing, edge detection or generic filtering operations are commonly performed exploiting neighbourhood information: pixel values are computed by considering nearby points. The method proposed in this paper follows the same concept: the values  $\hat{p}_i(\mathbf{t}_i)$  are computed by considering the measurement of adjacent taxels.

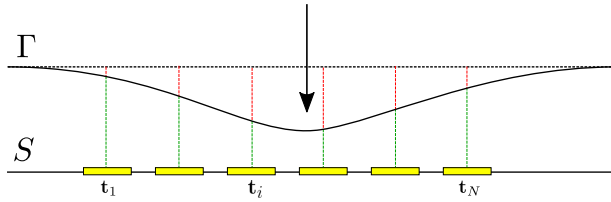


Figure 2: Working principle of the class of robot skin systems considered in this paper.

### 3 Local Processing For Robot Skin

Pixels in images are arranged as a planar grid, making the neighborhood of each pixel well defined. This does not hold for a robot skin system due to the non-regular placement of tactile elements. Therefore, before discussing about local processing, the concept of *neighbourhood* or *adjacency* for taxels must be defined and encoded in a proper data structure. As proposed in [19], a suitable approach to represent taxels positions and their proximity relations is by defining a *skin mesh*. In this representation, taxels are connected by edges (similarly to a graph structure) which provide both topological and geometric information. Let  $T = \{\mathbf{t}_1, \dots, \mathbf{t}_N\}$  be the set containing the position of each taxel. The Delaunay triangulation [28] applied to  $T$  allows to define a  $N \times N$  matrix  $\mathbf{E}$  describing topological relations among adjacent tactile elements:

$$\mathbf{E} = [E]_{ij} = [E]_{ji} = \begin{cases} 1, & \text{if taxels } i \text{ and } j \text{ are connected} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The set  $T$  containing taxel positions and the adjacency matrix  $\mathbf{E}$  can be used to define a mesh  $S^* = (T, \mathbf{E})$ , representing a piecewise linear approximation of the sensorized surface  $S$ .

While  $T$  and  $\mathbf{E}$  provide information on the topological connections among the taxels, the geometric relations, representing the distance among *connected* elements, can be represented with a matrix  $\mathbf{D} \in \mathbb{R}^{N \times N}$ , defined as:

$$\mathbf{D} = [D]_{ij} = [D]_{ji} = \|\mathbf{t}_i - \mathbf{t}_j\| E_{ij} \quad (2)$$

where  $D_{ij} = 0$  for non-connected elements.

The information given by the adjacency matrix  $\mathbf{E}$  can be used to compute the values  $\hat{p}_i(\mathbf{t}_i)$  taking into account the response of nearby sensors. In this paper, the filtered values are computed as:

$$\hat{p}(\mathbf{t}_i) = p(\mathbf{t}_i)w_{ii} + \sum_{j \in \text{adj}(i)} p(\mathbf{t}_j)w_{ji} \quad (3)$$

where  $w_{ji} \in \mathfrak{R}$  are scalar weight coefficients and  $\text{adj}(i)$  is the adjacency list of the  $i$ -th taxel defined as:

$$\text{adj}(i) = \{j\} \forall j \in \{1, \dots, N\} : E_{ij} = 1$$

Equation (3), representing a weighted sum of the taxel responses, can be rewritten in a more compact form by imposing  $E_{ii} = 1$  for  $i = \{1, \dots, N\}$ . Therefore, Equation (3) becomes:

$$\hat{p}(\mathbf{t}_i) = \sum_{j \in \text{adj}(i)} p(\mathbf{t}_j)w_{ji} \quad (4)$$

It is worth noting that if taxels were arranged as a matrix (like pixels in an image) and  $w_{ij}$  were constants, Equation (4) would correspond to a 2D convolution operation performed with a 3x3 squared kernel (see Figure 3(a)). A graphical representation of Equation (4) applied to a generic skin mesh is represented in Figure 3(b).

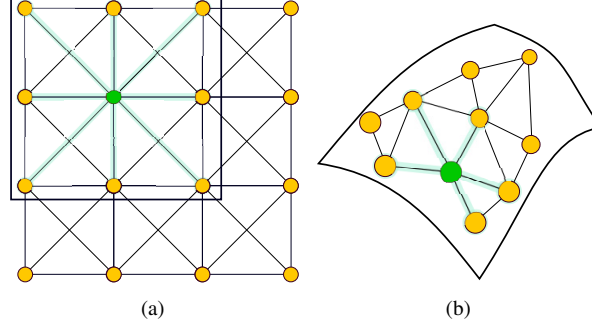


Figure 3: Adjacent taxels considered when computing Equation (3). (a) Taxels arranged as a flat 2D matrix. In this case Equation (3) corresponds to a convolutional operation performed with a  $3 \times 3$  kernel mask. (b) Generic non-regular spatial arrangement of taxels placed over a non-planar surface.

An Equation similar to (4) is used in Graph Convolutional Neural Networks to compute features in the hidden layers with  $w_{ij}$  learned at training time [29]. However, similarly to the image processing domain, it will be shown that a proper choice of the weights  $w_{ij}$  allows to design a filter. In this paper, they are computed to highlight edges in the pressure distribution, as described in the next Section.

## 4 Edge Detection on Robot Skin Data

In image processing, edges can be detected in grayscale images by looking for variations in the luminance of nearby pixels. This is usually performed by designing convolutional masks approximating the first or second order derivative [30]. The concept for tactile data processing is similar. A gradient-based operator can be considered to highlight high-frequency spatial variations in the taxel responses. However, some differences must be considered. Due to the regular structure of an image, the weights approximating the derivative in an edge detection mask: (i) are constant; (ii) they can be computed separately for rows and columns [30]. This does not hold for robot skin due to the non-regular spatial distribution of the taxels. One possibility to overcome this problem is to compute an approximation of the *directional derivative* along the vector  $\mathbf{k}_{ij} = \frac{\mathbf{t}_j - \mathbf{t}_i}{D_{ij}}$ . Therefore the weights  $w_{ij}$  can be computed as:

$$w_{ij} = \begin{cases} \frac{p(\mathbf{t}_i + D_{ij}\mathbf{k}_{ij}) - p(\mathbf{t}_i)}{D_{ij}}, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

It is clear that the weights computed with Equation (5) are not constant for each taxel. Indeed they depend on  $D_{ij}$  and  $\mathbf{k}_{ij}$  that change when different set of taxels are considered. The weights in Equation (5) can be substituted in Equation (4), leading to:

$$\hat{p}(\mathbf{t}_i) = \sum_{j \in \text{adj}(i)} p(\mathbf{t}_j) \frac{p(\mathbf{t}_i + D_{ij}\mathbf{k}_{ij}) - p(\mathbf{t}_i)}{D_{ij}} \quad (6)$$

Therefore,  $\hat{p}(\mathbf{t}_i)$  can be interpreted as the response to the input  $p(\mathbf{t}_i)$  of a spatially varying linear filter.

By computing Equation (6), even small variations on the intensity values among adjacent taxels are detected. Since the goal is to focus on strong variations corresponding to edges, a thresholding operation must be performed on the values  $\hat{p}_i(\mathbf{t}_i)$ , thus obtaining:

$$\bar{p}(\mathbf{t}_i) = \begin{cases} 1, & \text{if } \hat{p}(\mathbf{t}_i) > \varepsilon \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where  $\varepsilon$  is a value that must be properly tuned to highlight edges in the contact shape. Figure 1 shows an example of the result of this filtering applied on human fingers pressing on a non-planar sensorized surface. As it can be seen, the values  $\bar{p}_i(\mathbf{t}_i)$  mapped on the robot skin mesh allow to extract the contour of the fingers. Another example is shown in Figure (4) where a pen is pressed on a small planar surface.

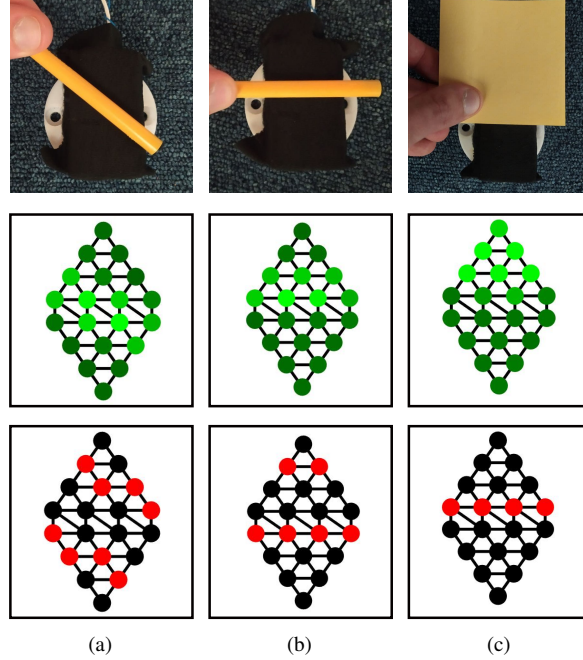


Figure 4: Columns (a) and (b) show a line profile on the skin response obtained pressing a pen on a small flat sensorized surface in two configurations. Column (c) shows a step profile obtained using a note cube pressed on a small flat sensorized surface. The images on the first row show the objects in contact with the sensor. The second row images show the robot skin responses mapped on the skin mesh (darker dots correspond to lower responses). The third row images show the filtered values mapped on the skin mesh. It can be seen that the contour of the line edge and the step edge are correctly extracted. In the case of (a) and (b) (line profile) two contours are extracted since the filter detects a variation on both sides of the pen.

## 5 Experimental Validation

### 5.1 Contour Following

The developed filter was validated in the simple task of planar contour following performed using a robot equipped with a sensorized end-effector. This Section describes

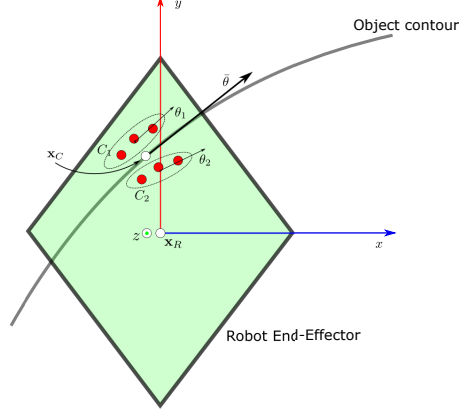


Figure 5: The end-effector in contact with the contour of an object. The green area represents the sensorized surface. The sets of connected taxels representing edges in the contact shape are extracted and their orientation is used to compute the robot motion.

how the planar motions of the robot are computed based on the perception of edges in the contact shape. Consider Figure 5 as a reference. The robot end-effector is in contact with the contour of the object and the  $z$ -axis of its reference frame is perpendicular to the object surface. The end-effector must be commanded to compute a roto-translation movement around the point  $\mathbf{x}_R$ , proceeding along the edge profile. It is assumed that all the geometric vectors are referred to the end-effector frame.

The first step that needs to be performed consists in extracting the components representing edges from the skin mesh and in computing their orientation with respect to the  $x$ -axis. As visible in Figure 4, there could be more than one set of taxels representing edges in the contact shape. Each single edge must be identified and extracted to compute its orientation. This is done by computing the connected components [31] from the skin mesh where  $\bar{p}_i(\mathbf{t}_i) = 1$ . This operation can be performed with a BFS visit [31]. As an outcome  $K$  connected components  $C_k$  with  $k = \{1, \dots, K\}$  are computed, where the  $k$ -th component contains the indexes of the taxels connected by topological edges where  $\bar{p}(\mathbf{t}_i) = 1$ . Therefore, each connected component  $C_k$  represents a set of taxels describing an edge in the contact distribution.

Then, to compute the planar motion of the end-effector, elements belonging to the connected components have to be projected on to the  $xy$  plane described in Figure 5, thus computing:

$$\tilde{\mathbf{t}}_h = (\mathbf{I} - \mathbf{n}^T \mathbf{n}) \mathbf{t}_h \quad \begin{matrix} \forall h \in C_k \\ k = \{1, \dots, K\} \end{matrix} \quad (8)$$

where  $\mathbf{n} \in \mathbb{R}^3$  is a unit vector oriented as the  $z$ -axis and  $\mathbf{I}$  is the identity matrix.

Once the positions of the taxels belonging to  $C_k$  have been projected on the plane, the orientation  $\theta_k$ , related to  $C_k$ , with respect to the  $x$ -axis is computed as described in [32]. The technique presented in [32] assumes to compute the geometric moments of a distribution of pixels. In the proposed case, the moments are computed, for each  $C_k$ , on the sets of projected taxels positions  $\tilde{\mathbf{t}}_h$ .

The last step is to compute centroid  $\mathbf{x}_C$  of the taxels associated to  $\bar{p}(\mathbf{t}_i) = 1$  (see Figure 5). This is done by computing the center of mass of the distribution of points described by Equation 8.

Then, the rotational movement of the end-effector around the point  $\mathbf{x}_R$  is given by  $\bar{\theta}$ , which is computed by averaging all over the  $\theta_k$ . The translational part of the motion is instead computed as:

$$\delta \mathbf{x} = \begin{bmatrix} s \cos(\bar{\theta}) \\ x_{C_2} - x_{R_2} \\ 0 \end{bmatrix} \quad (9)$$

where  $s \in \mathbb{R}$  represents the step size, defining the length of the translation movement of the end-effector along the edge.

## 5.2 Experiments Description

The approach is experimentally validated on the CySkin robot skin [33]. It is a robot skin technology composed of interconnected triangular modules that can be adapted to curved surfaces. Each module contains up to 11 capacitive transducers capturing the pressure distribution generated by a normal force applied on the sensorized surface. The pitch among each taxel is 7.5 mm and tactile measurements are sampled at 20 Hz. To validate the filtering technique proposed in this paper, CySkin was integrated on two different end-effectors (see Figure 6). The first is flat and contains 20 taxels. The second is non-planar and contains 221 taxels. Both end-effectors were fixed on a

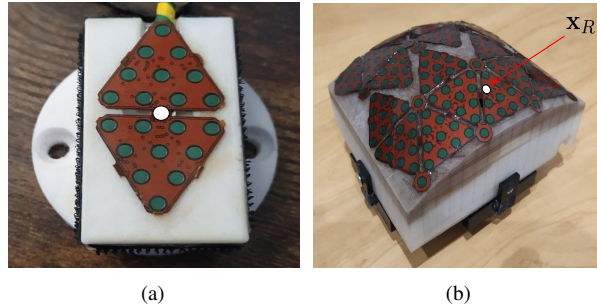


Figure 6: The two end-effectors used to validate the proposed approach.(a) Planar tactile patch: 20 taxels.(b) Non-planar tactile patch: 211 taxels.

Franka Emika arm, which is used to perform the contour following task on two objects: a ruler (a straight line path) and a mug (a circular path). Figure 7 shows the objects and the non-planar end-effector in contact with them. The reference position  $\mathbf{x}_R$  described

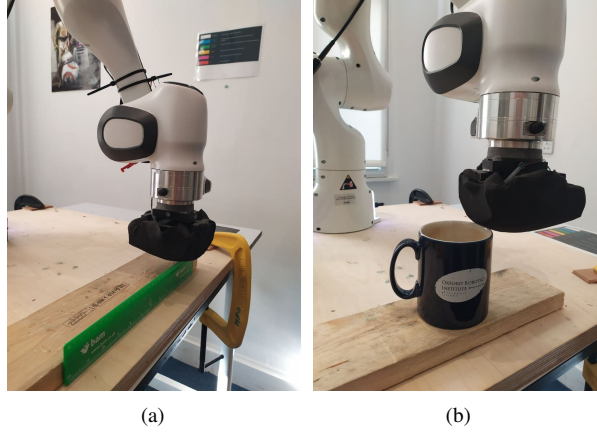


Figure 7: Object used to validate the proposed approach. (a) Robot following a ruler of 25 mm length and 1 mm thickness. (b) Robot following the contour of a mug of 81 mm and 75 mm outer and inner diameters respectively.

in Section 5.1 is highlighted in white in Figure 6. It corresponds to the center of the flat end-effector, while for the non-planar case it coincides with one of the positions of the taxel lying on the curvature.

The proposed technique was compared with a tactile image based approach in the case of the non-planar end-effector. The processing pipeline needed to create a tactile image and summarized at end of Section 2 was used [19]. The tactile image is generated by resampling the flattened skin geometry with a step of 1 mm, generating an image of  $74 \times 132$  pixels. The edges are extracted on the tactile image using the Canny operator [30]. Pixels corresponding to edges are then back-projected on the skin mesh, and the motion directions are computed as described in Section 5.1.

The experiments were performed using the following tapping-based procedure:

1. the robot starts from a random initial configuration where the end-effector is placed over the contour of the object that must be followed;
2. the robot proceeds with constant velocity along the  $z$ -axis until a contact is detected with tactile sensors;
3. the robot is controlled to apply a constant desired force along the  $z$ -axis of 15 N;
4. once the force controller reaches the steady state, the tactile measurements are filtered to extract the edges and the motion command is computed as described in Sections 4 and 5.1;
5. the end-effector is lifted of 1 cm and its position and orientation are adjusted according to what explained in Section 5.1.
6. after the end-effector is correctly repositioned, the robot starts again from step (2) until the object contours are fully explored.

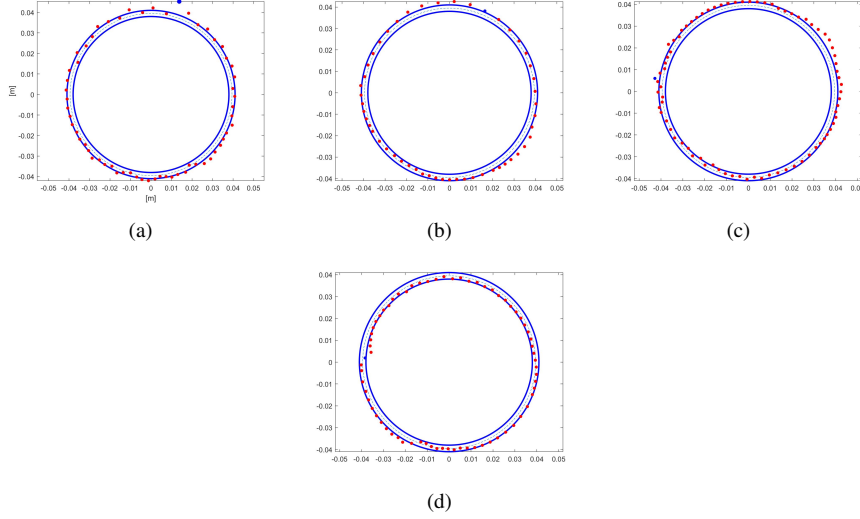


Figure 8: Example of circular contour. Red dots correspond to the position of  $\mathbf{x}_R$  at each contact. The blue lines represent the inner and outer perimeter of the mug. The blue dot corresponds to initial the position of  $\mathbf{x}_R$ . The exploration is performed clockwise. (a) Planar end-effector - proposed method. (b) Planar end-effector - tactile image based (c) Non-planar end-effector - proposed method. (d) Non-planar end-effector - tactile image based.

This procedure has been repeated three times for each object and end-effector using a step size  $s$  of 5 mm.

## 6 Results and Discussions

Figure 8 reports the outcome of one of the three experiments of contour following performed on the mug. Both cases of the planar and non-planar end-effector are provided. Red dots correspond to the position of  $\mathbf{x}_R$  at each contact. The blue dot represents the position of  $\mathbf{x}_R$  at the first contact. The blue lines represent the inner and outer perimeter of the mug.

To evaluate whether the contours of the objects were correctly tracked, the position of  $\mathbf{x}_R$  at each contact was recorded. Let  $\mathbf{x}_l$  be the position of  $\mathbf{x}_R$  at the  $l$ -th contact with the object, with  $l = \{2, \dots, L\}$  and  $L$  is the total number of contacts between the end-effector and the object in the single experiment. The index  $l$  starts from 2 since the initial position of the end-effector was not considered while computing the error.

The tracking error in the case of the circular path was computed as follows. The points  $\mathbf{x}_l$  were centered in the origin and they were transformed in polar coordinates, so that for each point  $\mathbf{x}_l$  a radius  $r_l$  and an angle  $\alpha_l$  were computed. Then the real position of the contour for the angle  $\alpha_l$  is given by  $\bar{\mathbf{x}}_l = \bar{r}[\cos(\alpha_l) \sin(\alpha_l)]^\top$  where  $\bar{r} = 39.5$  mm is the real radius of the mug (the dashed line in Figure 8). For each pair of  $\mathbf{x}_l$ ,  $\bar{\mathbf{x}}_l$  the

Table 1: Results of the contour following task performed with the small planar end-effector.

Object	Method	$\bar{d}(mm)$	$\bar{\sigma}_d(mm)$	$d_M(mm)$
Ruler	Proposed approach	0.97	0.67	2.99
Ruler	Tactile Image	0.89	0.49	2.93
Mug	Proposed approach	1.53	0.87	4.16
Mug	Tactile Image	1.34	0.85	3.53

Table 2: Results of the contour following task performed with the non-planar end-effector.

Object	Method	$\bar{d}(mm)$	$\bar{\sigma}_d(mm)$	$d_M(mm)$
Ruler	Proposed approach	1.12	0.70	4.11
Ruler	Tactile Image	1.12	0.53	2.38
Mug	Proposed approach	1.86	0.96	5.21
Mug	Tactile Image	1.81	0.82	6.12

distance  $d_l = \|\mathbf{x}_l - \bar{\mathbf{x}}_l\|$  was computed.

In the case of the ruler, which was fixed parallel to one of the axes of the robot, we computed  $d_l = \|x_{l_2} - c\|$ , where  $c$  represents the offset of the line with respect to the robot base.

For each one of the three experiments the following statistics were computed: (i) the mean distance; (ii) the standard deviation; (iii) the maximum distance.

Tables 1 and 2 summarized the results obtained in all the three experiments reporting:  $\bar{d}$  the average of the mean distances,  $\bar{\sigma}$  the mean standard deviation and  $d_M$  the maximum distance across all the three experiments.

It must be noted that the results presented in Table II are affected by an uncertainty of the position of the taxels. This uncertainty is due to the procedure (manually performed) required to integrate the tactile sensors over non-planar geometries [34–37]. Since the geometry of the large end-effector used in the experiments is simple, the taxels position has been estimated with a small uncertainty of 1 mm. On the contrary, for flat tactile sensor geometries, the position of the taxels is exactly known from the CAD model and the results showed in Table I are not affected by any uncertainty.

Results show that with both techniques the object contour is tracked with good accuracy. Indeed, the thickness of the ruler and the mug (respectively 1 mm and 3 mm) are lower than the pitch among the sensors, which is 7.5 mm. It can be seen from the Tables that the maximum error is much lower than the pitch. The method based on tactile images performs slightly better, providing a smoother reconstruction of the contour of the object. The mean difference among each  $\bar{d}$  computed with the two methods is 0.295 mm (see Tables 1 and 2).

Indeed, the current limitation of the proposed method is that the contours extracted with Equation (4) exactly correspond to the positions  $\mathbf{t}_i$ . This in general is not true

since the edge may pass through two adjacent taxels (see Figures 1 and 4). This is clearly related to the resolution of the skin mesh. A finer pitch among the sensor would lead to more precise results. On the contrary, the filtering based on tactile images allows to reconstruct edges lying across the vertices composing the mesh. Indeed, with tactile image based methods the skin geometry is usually resampled using a grid with a finer resolution than the pitch among the sensors. This leads to a slightly higher precision in the contour reconstruction. However, to overcome this limitation, we are currently evaluating the possibility of performing the same operation of resampling and interpolation directly on the 3D mesh, thus obtaining a finer mesh [38].

In terms of execution time, the proposed method is significantly faster. The computational time of the proposed method linearly scales with the number of vertices composing the mesh. Indeed, although Equation (4) requires the adjacency list, the maximum number of adjacent vertices is constant and fixed by the Delunay triangulation.

In the case of the curved patch, the code implementing Equation (4) takes an average of 0.075 ms to be executed for all taxels on a computer equipped with an Intel i7-10875H.

The computational time required to generate the tactile image linearly scales with the number of rows and columns. However, the execution time of the whole pipeline depends on the size of the contact area. Indeed, the pixels describing the contours (extracted from the tactile image) need to be back-projected on the skin mesh. The time required by this operation depends on how many pixels must be back-projected. In the case of the contour following, the method based on tactile images takes an average of 3.7 ms to be executed. It is also worth discussing the worst case, when all the pixels forming the image must be back-projected. In this case the pipeline based on tactile images takes 7.2 ms. Although the worst case analysis could not be relevant in the case of the contour following, it becomes interesting when considering filtering techniques requiring the back-projection of all the pixels.

## 7 Conclusion

In this paper a local filtering technique for robot skin data has been proposed. Differently from previous literature, where tactile images were used to process the contact shape, the proposed method can be directly applied even with a non-regular and non-planar arrangement of the tactile sensors, thus avoiding the use of additional processing steps needed to convert the tactile data to images. In particular, it has been shown how the proposed technique can be used to design a filter extracting sharp variations in the contact distribution. The filter was validated in a simple task of planar contour following, where the robot was commanded to follow a straight path and a circular one. As a possible extension of the work, several other filters can be designed to perform lowpass filtering or noise reduction, just to name but a few.

## References

- [1] Z. Kappassov, J.-A. Corrales, and V. Perdereau, “Tactile sensing in dexterous robot hands — review,” *Robotics and Autonomous Systems*, vol. 74, pp. 195–220, 2015. [Online].

Available: <https://www.sciencedirect.com/science/article/pii/S0921889015001621>

- [2] S. Luo, J. Bimbo, R. Dahiya, and H. Liu, "Robotic tactile perception of object properties: A review," *Mechatronics*, vol. 48, pp. 54–67, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0957415817301575>
- [3] S. Denei, F. Mastrogiovanni, and G. Cannata, "Towards the creation of tactile maps for robots and their use in robot contact motion control," *Robotics and Autonomous Systems*, vol. 63, pp. 293–308, 2015, advances in Tactile Sensing and Touch-based Human Robot Interaction. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889014001869>
- [4] G. Cannata, M. Maggiali, G. Metta, and G. Sandini, "An embedded artificial skin for humanoid robots," in *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Aug 2008, pp. 434–438.
- [5] P. Mittendorf and G. Cheng, "Humanoid multimodal tactile-sensing modules," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 401–410, June 2011.
- [6] Y. Ohmura, Y. Kuniyoshi, and A. Nagakubo, "Conformable and scalable tactile sensor skin for curved surfaces," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, May 2006, pp. 1348–1353.
- [7] D. S. Tawil, D. Rye, and M. Velonaki, "Improved image reconstruction for an eit-based sensitive skin with multiple internal electrodes," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 425–435, June 2011.
- [8] D. Um, B. Stankovic, K. Giles, T. Hammond, and V. Lumelsky, "A modularized sensitive skin for motion planning in uncertain environments," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 1, May 1998, pp. 7–12 vol.1.
- [9] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing—from humans to humanoids," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 1–20, 2010.
- [10] A. Drimus, G. Kootstra, A. Bilberg, and D. Kragic, "Classification of rigid and deformable objects using a novel tactile sensor," in *2011 15th International Conference on Advanced Robotics (ICAR)*, 2011, pp. 427–434.
- [11] H. Liu, X. Song, T. Nanayakkara, L. D. Seneviratne, and K. Althoefer, "A computationally fast algorithm for local contact shape and pose classification using a tactile array sensor," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 1410–1415.
- [12] S. Decherchi, P. Gastaldo, R. S. Dahiya, M. Valle, and R. Zunino, "Tactile-data classification of contact materials using computational intelligence," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 635–639, 2011.
- [13] S. Luo, W. Mou, K. Althoefer, and H. Liu, "Localizing the object contact through matching tactile features with visual map," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3903–3908.
- [14] A. D. Berger and P. K. Khosla, "Using tactile data for real-time feedback," *The International Journal of Robotics Research*, vol. 10, no. 2, pp. 88–102, 1991. [Online]. Available: <https://doi.org/10.1177/027836499101000202>
- [15] Q. li, C. Schuermann, R. Haschke, and H. Ritter, "A control framework for tactile servoing," 06 2013.

- [16] N. F. Lepora, A. Church, C. de Kerckhove, R. Hadsell, and J. Lloyd, "From pixels to percepts: Highly robust edge perception and contour following using deep learning and an optical biomimetic tactile sensor," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2101–2107, 2019.
- [17] Ning Chen, R. Rink, and Hong Zhang, "Efficient edge detection from tactile data," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, vol. 3, 1995, pp. 386–391 vol.3.
- [18] T. C. Phung, Y. S. Ihn, J. C. Koo, and H. R. Choi, "An enhanced edge tracking method using a low resolution tactile sensor," *International Journal of Control, Automation and Systems*, vol. 8, no. 2, pp. 462–467, 2010.
- [19] A. Albin and G. Cannata, "Pressure distribution classification and segmentation of human hands in contact with the robot body," *The International Journal of Robotics Research*, vol. 39, no. 6, pp. 668–687, 2020. [Online]. Available: <https://doi.org/10.1177/0278364920907688>
- [20] U. Martinez-Hernandez, G. Metta, T. J. Dodd, T. J. Prescott, L. Natale, and N. F. Lepora, "Active contour following to explore object shape with robot touch," in *2013 World Haptics Conference (WHC)*, 2013, pp. 341–346.
- [21] U. Martinez-Hernandez, T. J. Dodd, M. H. Evans, T. J. Prescott, and N. F. Lepora, "Active sensorimotor control for tactile exploration," *Robotics and Autonomous Systems*, vol. 87, pp. 15–27, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889016303086>
- [22] Y. Ohtake, A. Belyaev, and I. Bogaevski, "Mesh regularization and adaptive smoothing," *Computer-Aided Design*, vol. 33, no. 11, pp. 789–800, 2001. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0010448501000951>
- [23] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, "Surface feature detection and description with applications to mesh matching," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 373–380.
- [24] M. Wei, J. Huang, X. Xie, L. Liu, J. Wang, and J. Qin, "Mesh denoising guided by patch normal co-filtering via kernel low-rank recovery," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 10, pp. 2910–2926, 2019.
- [25] T. Mukai, M. Onishi, T. Odashima, S. Hirano, and Z. Luo, "Development of the tactile sensor system of a human-interactive robot "ri-man"," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 505–512, April 2008.
- [26] T. Someya, T. Sekitani, S. Iba, Y. Kato, H. Kawaguchi, and T. Sakurai, "A large-area, flexible pressure sensor matrix with organic field-effect transistors for artificial skin applications," *Proceedings of the National Academy of Sciences*, vol. 101, no. 27, pp. 9966–9970, 2004.
- [27] T. Minato, Y. Yoshikawa, T. Noda, S. Ikemoto, H. Ishiguro, and M. Asada, "Cb2: A child robot with biomimetic body for cognitive developmental robotics," in *2007 7th IEEE-RAS International Conference on Humanoid Robots*, Nov 2007, pp. 557–562.
- [28] J. E. Goodman and J. O'Rourke, Eds., *Handbook of Discrete and Computational Geometry*. USA: CRC Press, Inc., 1997.
- [29] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, 2021.

- [30] R. Jain, R. Kasturi, and B. Schunck, *Machine Vision*, 01 1995.
- [31] R. Sedgewick, *Algorithms in C, Part 5: Graph Algorithms, Third Edition*, 3rd ed. Addison-Wesley Professional, 2001.
- [32] L. Rocha, L. Velho, and P. de Carvalho, “Image moments-based structuring and tracking of objects.” 01 2002, pp. 99–105.
- [33] “Cyskin,” <https://www.cyskin.com/>.
- [34] A. Del Prete, S. Denei, L. Natale, F. Mastrogiovanni, F. Nori, G. Cannata, and G. Metta, “Skin spatial calibration using force/torque measurements,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3694–3700.
- [35] A. Albini, S. Denei, and G. Cannata, “Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 153–159.
- [36] G. Cannata, S. Denei, and F. Mastrogiovanni, “Towards automated self-calibration of robot skin,” in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 4849–4854.
- [37] P. Mittendorf, E. Dean, and G. Cheng, “3d spatial self-organization of a modular artificial skin,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3969–3974.
- [38] P. Alliez, G. Ucelli, C. Gotsman, and M. Attene, *Recent Advances in Remeshing of Surfaces*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 53–82.