IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

Tactile-based Task Definition through Edge Contact Formation Setpoints for Object Exploration and Manipulation

Zhanat Kappassov, Juan Antonio Corrales Ramon and Véronique Perdereau

***Abstract*—In autonomous robot tasks involving physical contacts with the environment, it is still challenging to perform dexterous manipulation. Force control approaches and force sensors are usually used to control the actions of a robot. However, the spatial resolution of the force sensors is limited when exploring and manipulating an object through the tracking of salient tactile features, such as edges, while touching the surface of the object. In fact, the exploration or manipulation can be implemented via tactile servoing approaches that use the parameters of those edges. These parameters, obtained by an array of tactile sensors, are used for generating setpoints driving a robot arm to minimize the gap between the desired and current parameters of a given edge. This paper describes a new common strategy for defining tactile setpoint signals for tactile servoing approaches in order to implement different contact-based tasks. These setpoints represent artificial constraints which comply with natural constraints on the force and position of the robot end-effector imposed by the physical contact between the robot and the object. The sequence of setpoints for three different tasks are given as examples: alignment with an object, exploration of a linear object with variable stiffness and manipulation by rolling of ellipsoidal objects. These tasks are validated with real experiments using a KUKA LWR 4+ robot arm and a Weiss WTS-0614 piezoresistive sensor. The arm controller runs at 1 kHz and the tactile servoing control runs at 100 Hz, which is limited by the sampling rate of the sensing array.**

***Index Terms*—Tactile servoing, robotic manipulation, force and tactile sensing.**

I. Introduction

A new paradigm of robot arm control, called tactile servoing, surged after the introduction of tactile sensing arrays [1]. The control scheme was based on a hybrid position/force approach [2]. Following the artificial and natural constraints paradigm [3], the setpoints were divided into two modes: force control and position control. Force setpoints ensured continuous (in time) contact with an edge of an object while sliding over it. Position setpoints were used to control the orientation of the edge with respect to a planar sensing array. The array's data was interpreted as a gray scale image whose moments – principle components – represented geometrical features of the contact, including the orientation of the edge projected on the contact area. In order to map these features to the control inputs of the robot end-effector, an inverse tactile Jacobian was introduced in [4] and extended in [5]. In our previous work [6], we modified it further so that its matrix elements were dependent on the contact formation (CF): vertex-face (point-contact) or edge-face (edge-contact). The point-contact appears when the area of contact is spread equally along vertical and horizontal axes of a planar sensing array, e.g., when it interacts with a vertex of an object. The edge-contact appears when the area of contact is spread mostly along one direction of a sensing array, e.g., when the array touches a cylindrical body or an edge of a square box.

This paper presents our results in tactile object manipulation and exploration tasks that involve the edge-contact type. In previous works, we tackled the problems of feature extraction [7] and tactile control in order to generate actuation commands [8]. The present paper describes the setpoints for the tactile servoing controller. The setpoints define the motions to be executed in different tasks for every CF. In fact, each contact-based task requires a different sequence of setpoints. For instance, Figure 1 (top row) shows a sequence of CFs, consisting of a vertex-plane and edge-plane CFs, when a robot arm increases the area of contact between its sensing array and the edge of an object. Our contribution to tactile servoing is the definition of contact-based robotic tasks based on tactile interaction data obtained from the edges of the touched objects. The task descriptions developed in this paper allow an application designer to obtain a solution for contact-rich task specification.

Unlike previous works, our approach is essentially based on a force control approach, in which force setpoints prevail over the position setpoints [9]. This approach requires the robot end-effector (with a tactile sensor) to always be installed next to an object in order to steer the contact with a contact force threshold. In a non-contact situation, the robot is guided by using pseudo tactile feedback based on the continuous monitoring of its pose. The pose of the robot end-effector is controlled by using the Cartesian Impedance control approach. The natural and artificial constraints formalism using results of [10] and [11] is implemented to specify which directions need to be compliant (force controlled) or active (position controlled) and Projection matrix elements are specified in order to vary control directions with specified control gains. This strategy results in a set of predefined controllers allowing reactive motion based on edge contacts.

Manuscript received: November 23, 2021; Accepted February 21, 2022.

This paper was recommended for publication by Editor Dan Popa upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by Kazakh Ministry of Education and Science under AP09058050, by Nazarbayev University under Faculty Development Competitive Research Grants Program 11022021FD2923, by the European Unions Horizon 2020 research and innovation program under grant agreement n 869855 (project SoftManBot). JACR was funded by the Spanish Ministry of Universities through a 'Beatriz Galindo' fellowship (Ref. BG20/00143) and by the Spanish Ministry of Science and Innovation through the research project PID2020-119367RB-I00.

V. Perdereau is with Sorbonne Université, CNRS, Institut des Systmes Intelligents et de Robotique (ISIR), 4 Place Jussieu, 75252 Paris, France veronique.perdereau@sorbonne-universite.fr

J.A. Corrales is with Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, 15782, Santiago de Compostela, Spain juanantonio.corrales@usc.es

Z. Kappassov is with Roboics Department, Nazarbayev University (NU), Nur-Sultan, Kazakhstan zhkappassov@nu.edu.kz

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

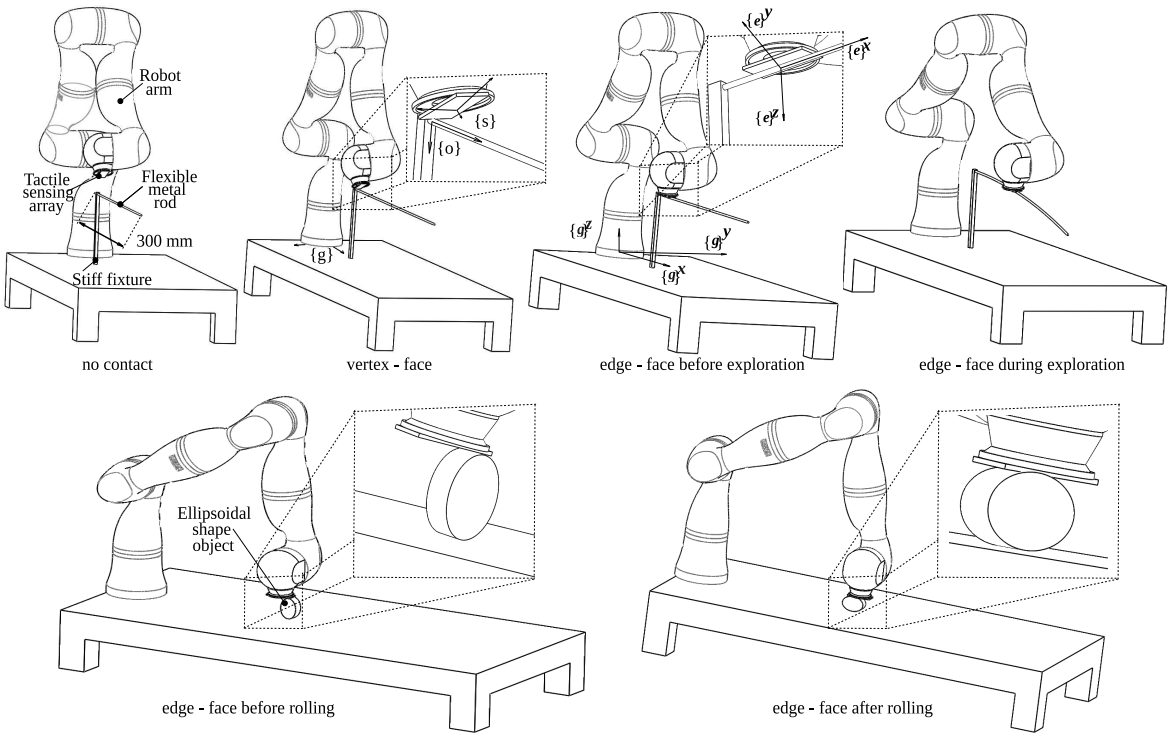

Fig. 1: Contact-Rich Tasks: (top) exploration and (bottom) manipulation by rolling. The robot equipped with a tactile sensor interacts with environment objects: a deformable rod, an ellipsoidal shaped object and a tray. During the exploration, a sequence of CFs (vertexface contact and edgeface contact) is executed. The position and orientation of the contact $\{o\}$ w.r.t. the sensor frame $\{s\}$ is kept by moving the end-effector $\{e\}$ w.r.t. the robot base frame $\{g\}$.

In this connection, previous works [12], [1], [4], [5] were benchmarked in the edge-following task. There were two major assumptions. Firstly, it was assumed that the length of the edge captured by the sensing array is infinite with respect to the sensing array itself. Secondly, the objects used for the task were rigid. In our experiments, the tactile servoing controller is applied to perform several edge-following tasks without these two assumptions. Deploying the tactile servoing controller in manipulators free of these assumptions increases their dexterity in physical interaction tasks involving both rigid and soft object.

The contribution of this letter is related with the fine planning of a single contact point between an edge of an object and a robot arm. Specifically, we define control inputs (setpoints) for the robot arm driven by the aforementioned tactile servoing controller. These setpoints were not fully described or experimentally evaluated in our previous works.

The remainder of this letter is organized as follows. After notations in Section II and a brief state-of-the-art review in Section III, a tactile servoing controller is described in Section IV. The control setpoints and details of experiments are presented in Section V. Finally, conclusions are drawn in Section V-C.

II. Setup and Notation in Tactile Servoing

Figure 1 (top-left) presents a typical setup for object exploration. The tactile servoing platform consists of a tactile sensor, an object and a robot arm. Figure 1 (vertex-face and edge-face) shows the corresponding CFs while the robot arm explores a flexible metal rod. The coordinate frames attached to the robot base, end-effector, sensor and contact with an object are denoted as $\{g\}$, $\{e\}$, $\{s\}$ and $\{o\}$, respectively. The basis vectors of the frames are denoted as ${}_{\{g,e,s,o\}}x, y, z$. The ${}_{\{s\}}z$ -axis coincides with the last joint axis of the robot and it is identical to the sensing surface normal pointing out of the sensor. Figures 1 (bottom) present a typical setup for object manipulation by rolling a cylinder on a rigid surface. Thereby, the cylinder on the plane forms the edge CF [13]. The sensor is attached onto the robot end-effector in the same coinciding configuration.

The following notations are used. x, y, and z are distances in millimeters. If used as subscripts, x, y, and z denote the frame axes. α is an angle of rotation (rad) quantifying the orientation of an edge CF. Both d and des used as superscripts denote a desired setpoint value. $\boldsymbol{s}$ is a tactile feature vector. $\mathbf{Ad}_{\{\mathbf{g}\}}\mathbf{T}^{\{\mathbf{s}\}}$ is the adjoint matrix derived from the forward kinematics that includes the translational plus rotational transformation from $\{s\}$ to $\{g\}$. CoP and CoC are the center of pressure and center of contact, respectively. f is the force in (N). $DZMP$ is the distance to zero moment point that appears at the presence of any moment of force μ in (Nm). υ (mm/s) is the linear velocity. $\boldsymbol{u}$ denotes the controller input. $\boldsymbol{K}_p$ is a diagonal matrix with proportional control gains. q (rad) is the robot joint position. $\boldsymbol{x}$ denotes the Cartesian pose, including translations x, y, z and rotations w_x, w_y, w_z.

III. Related work

Control paradigms for robotic arms that interact with a dynamic environment are usually combined within a control architecture with two main hierarchical levels: a globally-

desirable action control layer that generates the desired setpoints and a locally-feasible motion control layer that generates the actuation commands for tracking the desired setpoints [14]. The local control layer is usually implemented using one of the following approaches: an impedance control approach [15] that defines a mechanical response behavior of the robot end-effector during physical interactions, a hybrid control [2] that aims to control force and position in reciprocal directions or a force control approach in which position setpoints are generated by a proportional-integral force regulator [9].

The global control layer is usually formulated as a constraint-based task [11] and is responsible for achieving a complex goal in an optimal fashion. This goal may involve two or more simultaneous tasks that can be solved by using the robot's null space [16] or by converting these tasks into weighted quadratic functions [14]. The tasks are usually specified by using coordinates of contact locations or force vectors. Force sensors at the robot end-effector and torque sensors at the robot joints provide a mapping so that a task can be specified in the same units as the measurements. Tactile servoing approaches, on the other hand, require the interpretation of sensor signals [12]. Hand-crafted tactile features [8] or deep learning methods [17] are used for fine reconstruction of the geometry of a contact area. In this connection, the reference setpoints generated by the global control layer should be specified in the same units as these features.

One such tactile feature is the orientation of an edge of an object with respect to a flat pressure sensing array (i.e. a tactile sensor). Volumetric shapes in the surrounding world can be considered as a combination of vertexes, planes and edges, amongst which edges are one of the most detectable features [18]. For example, concave and convex surfaces can be discriminated using tactile perception of edges, which is not possible using a conventional vision camera [19]. Indeed, thanks to soft padding at fingertips, humans can detect edges [13] and compare them [20]. Similarly, modern artificial tactile sensors incorporate deformable media allowing the detection of different protrusions, including edges, on the sensing surface [21].

For the first time, edge-contact was investigated in a line-following task in [12]. In order to set in motion to the robot end-effector, the axes of the end-effector pose were divided into feedback and feedforward controlled loops following the natural and artificial constraints approach introduced in [3]. A modified Hough Transform was used to obtain the orientation of the edge detected by a planar tactile sensor, and force was estimated using the number of nonzero taxels. These two features were incorporated within the aforementioned hybrid force-position controller.

Alternatively, edges had already been applied in visually guided robot control approaches [22]. The errors between the desired and current visual features were mapped into robot motions using inverse Jacobian and Interaction matrices. Nowadays, computer vision is still ahead of tactile sensing in terms of technology and algorithms [7]. In fact, the methods introduced for visual servoing were later adopted in tactile servoing. Similar to visual servoing, tactile servoing was introduced for the first time in [1] and described in [4] with more details. An Inverse Tactile Jacobian was developed in order to obtain the corrective motions regulating the error between the desired and current physical contact states. Following the hybrid control approach, the visual and tactile control loops were merged into one control architecture using a Projection matrix followed by a modified inverse Jacobian [23], [24]. By fine tuning the tactile controller gains and selecting the matrix elements without interfering with the visually guided components, an autonomous system with a tactile sensor was able to increase the area of contact with an object's edge and align with it afterwards [5].

Instead of treating the problem of physical interaction from the control point of view, some researchers dealt with this problem by using machine learning techniques (perception-based approaches). In these perception-based approaches, robots either accumulate tactile signals from independent observations, e.g. supervised learning [17], [25], or learn step by step actions, e.g. reinforcement learning [26]. In this connection, the desired states of a contact point, which set in motion joints of a robot arm, are given implicitly for perception-based approaches.

In contrast to the aforementioned literature including our previous work [8], the desired states of the contact point – the one that appears between the robot end-effector and an edge of an object – had not been defined explicitly for the cases when a robot explores or manipulates objects. In this letter, we focus on the use of edge CFs in modulation of the robot pose.

IV. TACTILE SERVOING CONTROLLER

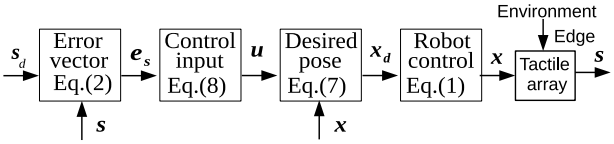

Fig. 2: Tactile servoing block diagram for edge-plane contact formations.

The tactile servoing controller is designed to prevail over the Cartesian position controller of the robot arm. Specifically, during constrained motion the robot control architecture is regarded as two interacting control loops. One loop accepts Cartesian pose commands and yields the motion of the robot arm. The other loop transforms task requirements into the end-effector pose errors (Fig. 2).

In the first control loop, the variation of the robot joint configuration with respect to the end-effector Cartesian pose and force are obtained by using the inverse robot Jacobian $\boldsymbol{J}_q^{-1}$ and the robot Jacobian transpose $\boldsymbol{J}_q^T$, respectively, where $\boldsymbol{J_q} = \frac{d(\boldsymbol{x}^{des}-\boldsymbol{x})}{d\boldsymbol{q}}$. A Cartesian control with gravity compensation is then possible by applying the following $n \times 1$ joint torques $\boldsymbol{\tau}^{des}$:

$$\boldsymbol{\tau}^{des} = \boldsymbol{J}_q^T(\boldsymbol{K}_{Pos}(\boldsymbol{x}^{des} - \boldsymbol{x}) - \boldsymbol{K}_D\dot{\boldsymbol{x}}) + \boldsymbol{g}(\boldsymbol{q}) \qquad (1)$$

where $\boldsymbol{K}_{Pos}$ and $\boldsymbol{K}_D \in \Re^{n\times n}$ are positive-definite gain matrices; $\dot{\boldsymbol{x}}$ is the motion twist of the end-effector; n is the

number of joints in the robot arm; $\boldsymbol{g(q)}$ is the sum of the torques exerted by gravity.

In the second control loop, the variation of the end-effector pose with respect to an $l \times m$ tactile sensing array output is obtained by using the inverse tactile Jacobian $\boldsymbol{J}^{-1}$, Projection matrix $\boldsymbol{P}$, and the tactile feature error-vector

$$\boldsymbol{e}_s = \boldsymbol{P} \cdot (\boldsymbol{s}^{des} - \boldsymbol{J}^{-1}\boldsymbol{s}), \tag{2}$$

where $\boldsymbol{s}^{des}$ is the desired feature vector that represents the desired location and orientation of the contact coordinate frame $\{o\}$ w.r.t. $\{s\}$. $\boldsymbol{s}$ is the current tactile feature vector derived using geometric moments [8]

$$\boldsymbol{s} = [CoP_x, CoP_y, f, CoC_x, CoC_y, \alpha_z]^T, \tag{3}$$

where f is the force causing deformation, d, of the sensing surface along z-axis, since the latter is elastic. The applied stress at the points of contact diffuses uniformly (mathematically it can be viewed as a point-spread function [27]). Given a compliance constant of the elastic surface C, $d = Cf$. This compliance constant also defines $\boldsymbol{K}_{Pos}$ and $\boldsymbol{K}_D$ in eq. (1). Specifically, the inherent mechanical compliance of the tactile sensor should not affect the end-effector resting pose when $\boldsymbol{x}^{des} = \boldsymbol{x}$.

The inverse Tactile Jacobian $\boldsymbol{J}^{-1}$ in edge CF gets the following matrix form [8]:

$$\boldsymbol{J}^{-1} = \begin{bmatrix} 1&0&0&0&0&0 \\ 0&1&0&0&0&0 \\ 0&0&1&0&0&0 \\ 0&1&0&0&-1&0 \\ 1&0&0&-1&0&0 \\ 0&0&0&0&0&1 \end{bmatrix}, \tag{4}$$

where the first three rows map positional deviations of a contact into corrective translational motions of a sensor; the fourth and fifth row are defined to map the difference between the geometrical center and center of pressure of the contact into rolling motions around the x- and y-axes; the last row maps an error in the orientation of the edge into a rotational motion around the z-axis.

The $\boldsymbol{P}$ matrix selects and maps the feature errors to the task-related feature error $\boldsymbol{P} =$

$$\begin{bmatrix} P_{CoP_x,x} & 0 & 0 & P_{\mu_x,x} & 0 & 0 \\ 0 & P_{CoP_y,y} & 0 & 0 & P_{\mu_y,y} & 0 \\ 0 & 0 & P_{f_z,z} & 0 & 0 & 0 \\ 0 & P_{CoP_y,w_x} & 0 & 0 & P_{\mu_y,w_x} & 0 \\ P_{CoP_x,w_y} & 0 & 0 & P_{\mu_x,w_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & P_{\alpha_z,w_z} \end{bmatrix} \tag{5}$$

where $P_{i,j} \in 1, 0$ is a binary value that selects and assigns the Cartesian direction for every feature error depending on the task. Specifically, the error in CoP_x can be transformed into both the translational ($P_{CoP_x,x} = 1$) and rotational ($P_{CoP_x,w_y} = 1$) errors in the motion twist of the robot end-effector. More than one activated element cannot appear in one row. For example, if $P_{CoP_x,x} = 1$, then $P_{\mu_x,x} = 0$. Otherwise, two contradicting tasks activated simultaneously will multiply the Cartesian error along the corresponding axis; in the given example, it is the x-axis.

By substituting eq.(4) and eq.(3) in eq.(2), the feature-error vector $\boldsymbol{e}_s$ is following

$$\boldsymbol{e}_s = \boldsymbol{P} \cdot (\boldsymbol{s}^{des} - \begin{bmatrix} CoP_x \\ CoP_y \\ f \\ DZMP_x \\ DZMP_y \\ \alpha_z \end{bmatrix}) \tag{6}$$

In order to overlay the first control loop with $\boldsymbol{e}_s$, the pose command $\boldsymbol{x}^{des}$ in eq. (1) takes the following form:

$$\boldsymbol{x}^{des} = \boldsymbol{x} + \boldsymbol{u} \tag{7}$$

where $\boldsymbol{x}$ is the current pose and $\boldsymbol{u}$ is the proportional regulator's control input, which depends on the task to be performed and on the contact coordinate frame error in the Cartesian space $\boldsymbol{e}_s$. Initially given in $\{s\}$, this error is transformed to be expressed into $\{g\}$ and multiplied by a proportional gain:

$$\boldsymbol{u} = \mathbf{Ad}_{\{\mathbf{g}\}\mathbf{T}\{\mathbf{s}\}}(\boldsymbol{K}_P \cdot \boldsymbol{e}_s) \tag{8}$$

where $\boldsymbol{K}_P$ denotes a diagonal matrix of the proportional regulator's gains.

This control input is fed into the robot position controller that generates corrective motions. For this tactile servoing control architecture, which transforms tactile feature errors into corrective Cartesian pose errors, the edge exploration action could be specified in the following way: the end-effector axis should coincide with the edge direction and keep a desired constant CoP. Due to the properties of the control architecture, the constant feature error CoP (the friction force while sliding is neglected) is transformed into a constant Cartesian error whenever the exploration of the edge is not accomplished. The impact of the friction force is usually not

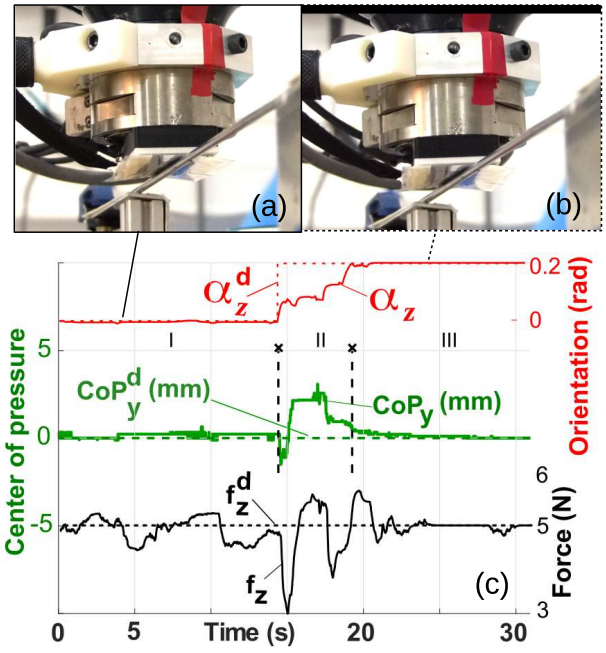

Fig. 3: Edge servoing without sliding: a) initial state, b) final state, and c) edge-plane contact formation parameters. Three parameters are controlled simultaneously: angle α_z of the edge w.r.t ${}_{\{s\}}x$, force f_z, and the center of pressure along y-axis CoP_y. The end-effector rotates around ${}_{\{s\}}z$ until the edge in contact is at α_z^d (rad).

trivial to model [4], but it can be mitigated by using deep convolutional neural network-based techniques [28].

Figure 3 illustrates the controller response over a fixed metal rod. The following corresponding elements of the Projection matrix (5) are activated: $P_{CoP_y,y} = P_{f_z,z} = P_{\alpha_z,w_z} = 1$ to control the physical interaction. The edge CF setpoints, including $\alpha_z^d = 0.2$ rad, $f_z = 5$ N and $CoP_y^d = 0$ mm, steer the contact between the tactile sensor and the object from the initial state (Fig. 3a) to the desired state (Fig. 3b). The center of pressure (CoP_y) is deviated from the desired location during the transition phase (II). Then it is returned to the desired location. The force f_z applied to the bar is maintained at the desired level. In this connection, the errors between the desired (dashed lines in Fig. 3c) and feedback (solid lines) parameters of the edge converge to zero in the steady state phase (III); and this tactile servoing controller can be used in manipulation and exploration tasks described in the following section.

V. Task Descriptions

All of the tasks described in this section are validated using a KUKA LWR 4+ robot arm and a Weiss WTS-0614 piezoresistive tactile sensing array attached onto the robot end-effector. Equation (1), which is used to control the robot joint torques, is computed at 1 kHz and the error given by eq. (2) is computed at 100 Hz, which is limited by the sampling rate of the tactile sensing array. It consists of 84 sensing cells organized in a 6×14 matrix. Every cell with the size of 3.4×3.4 mm detects only normal force.

Every task, related with either manipulation or exploration, requires a predefined set of parameters, including $\boldsymbol{s}^{des}$ and $\boldsymbol{P}$ elements (Fig. 4). This paper focuses on the edge contact type. However, the point contact should be mentioned as it appears when the edge is not parallel w.r.t. the sensing array (see Fig. 1, vertex-plane). In this case there is a vertex-plane CF, which has different constraints than the edge-plane CF. Therefore, $\boldsymbol{J}^{-1}$ is unique for each CF [8]. In the following case-study, the edge aligning task (or increasing the area of contact) is considered. The task is to drive the robot end-effector from the vertex-plane to edge-plane CF as illustrated in Fig. 1 (vertex-plane) and Fig. 1 (edge-plane before exploration), respectively. The comparison with the state-of-the-art approach [5] in edge aligning task is considered.

A. Increasing the area of contact

In order to explore an object whose location is not precisely given and its visual information is occluded or not available, the robot can move its end-effector with a tactile array until the appearance of a contact (Fig. 1, no contact). At the presence of that contact (Fig. 1, vertex-face), the tactile servoing begins.

The strategy to define the setpoints is as follows. The end-effector should roll around the point of contact until vertex-face CF transforms to edge-face CF. While previous approaches defined the setpoints to increase the area of contact only [5], the setpoints proposed in our work are more suitable for the edge-face CFs. Using these setpoints, we realize the rotational motion around ${}_{\{s\}}x$- or ${}_{\{s\}}y$- axis in order to align with the edge of the object as exemplified in Fig. 5.

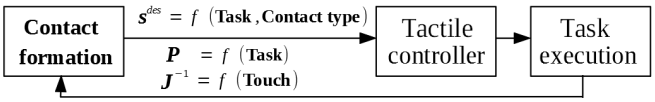

Fig. 4: Functional block diagram of a physically interacting robot arm. The contact formation layer defines tactile controller setpoints and the control layer sets in motion the robot joints. While executing the task, the contact state changes. This change is monitored by the tactile servoing controller.

In this example, it is assumed that the edge is expected to appear along ${}_{\{s\}}x$-axis. Then, the feature error given by current CoP_x and $CoP_x^d = 0$ (the geometrical center of the sensing array) is transformed into the rotational motion around ${}_{\{g\}}y$- axis in order to increase the area of contact. In the tactile servoing literature, the expected edge is assumed to be large enough to cover the whole sensing surface. When CoP_x^d is set to be in the geometrical center of the sensing surface, the pseudo injected error $CoP_x^d - CoP_x$, which is transformed into the rotation around ${}_{\{s\}}y$ with $P_{CoP_x,w_y} = 1$, was sufficient to align with an infinite edge of an object w.r.t. the sensor (Fig. 1, vertex-face before exploration). When an edge is not infinite – it is not covering the sensor from one side to the opposite side –, the previously proposed approach [5] fails to do this alignment (Fig. 5c).

The initial contact configuration is given in Fig. 5b. At the initial state, there is an angle ϕ between ${}_{\{o\}}x$ and ${}_{\{s\}}x$ since the object is not aligned with the sensing surface. During phase I (0 sec to 40 sec), the end-effector is set in rotational motion to reach $CoP_x^d = 0$. This rotation continues during phase II in

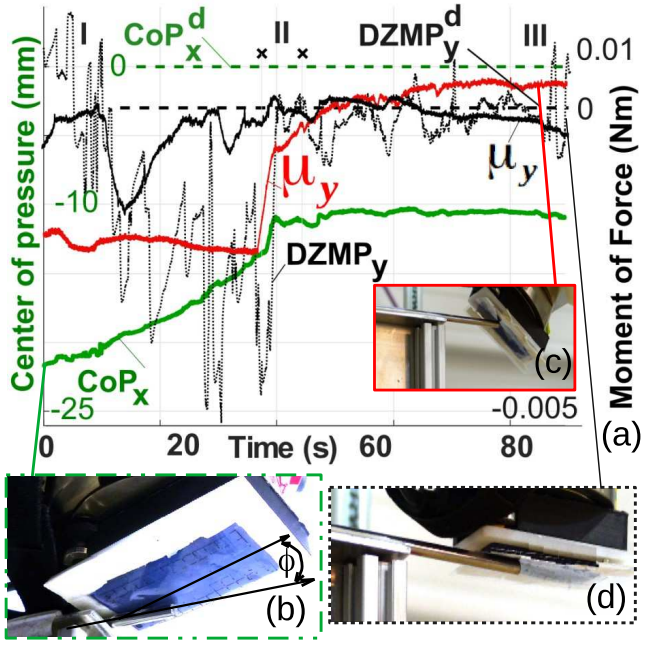

Fig. 5: Physical interaction task 1: rotate around the contact point to align with the edge that is not infinite in the sensing surface. a) Dashed lines depict the desired parameters; feedback CoP_x is given in green; the history of feedback $DZMP_y$ is represented by black dots; red line represents the ground-truth measured moment of force about sensor y-axis for the failed alignment; black line depicts the latter for the successful alignment. The sensor and a metal rod in b), c), and d) exemplify the mutual orientation between the sensor and object at the initial state, final failure state [5], and the successful alignment, respectively.

order to align the sensor with the object by rolling the contact along it. Even when the contact reaches the vertex of the object (phase III) the end-effector continues to rotate around the object but CoC_x is not converging to CoC_x^d (Fig. 5a, solid green line). The end-effector keeps rotating around the vertex of the bar until it reaches the joint limits.

In order to enhance the performance of tactile servoing in this task, the moment of force applied to the object must be tracked. In this connection, the rotation about ${}_{\{s\}}y$ is represented by the error $CoC_x^d - CoC_x$ at the beginning and later by the error in $DZMP_x^d - DZMP_x$ when the edge appears in the sensing surface. The algorithm for the edge detection is based on principal component analysis as described in [8]. $P_{\mu_x,w_y} = 1$ ($P_{CoP_x,w_y} = 0$) in the Projection matrix P projects the latter error to the rotation about ${}_{\{s\}}y$ in order to eliminate the moment of force applied to the object. In addition to CoP_x and $DZMP_x$, the normal force f_z, α_z and CoP_y are controlled simultaneously to keep the contact within the sensing area. The CF setpoint for the point- and edge- contacts are $\boldsymbol{s}^{des} = (CoP_x^d, CoP_y^d, f_z^d, -, -, -)^T$ (with $P_{CoP_x,w_y} = P_{CoP_y,y} = P_{f_z,z} = 1$) and $\boldsymbol{s}^{des} = (-, CoP_y^d, f_z^d, DZMP_x^d, -, \alpha_z^d)^T$ (with $P_{CoP_y,y} = P_{f_z,z} = P_{\mu_x,w_y} = P_{\alpha_z,w_z} = 1$), respectively. Fig. 5d exemplifies the successful alignment with the edge.

B. Tracking a flexible metal rod

In the two previous scenarios, the lateral propagation motions, which appear in object exploration, were neglected. The aim of the following task is to implement this exploration using the tactile servoing and predefined edge CF setpoints providing a guarded motion (i.e. to approach and touch an object without producing excessive force after contact is made).

The strategy to define the setpoints is based on the idea of applying corrective motions while sliding over an unknown edge in order to accumulate its point cloud. While previous approaches defined the setpoints to perform the sliding motion over a planar surface [5], the setpoints proposed here are designed for the sliding motion in a non-planar surface. The following experiment aims at tactile object exploration of a flexible object that bends under the applied normal force (Fig. 6).

In this experiment, the task is to track the unknown shape of a flexible rod fixed from one side only (Fig. 6, b). The rod is flexible and it bends under external forces (Fig. 6, c). In this connection, the sensing surface must be aligned with the local orientation of the rod at every time instant. During the exploration, the robot should maintain the contact within the sensing area while regulating the contact force. This exploration begins from the fixed side of the bar. The robot is already in contact and aligned with the edge. The current orientation of the object coincides with ${}_{\{s\}}x$. In order to follow the metal rod in space, CoP_x^d (activated with $P_{CoP_x,x} = 1$) is set to impose an external tangential motion onto the end-effector along the ${}_{\{s\}}x$-axis. The metal rod bends, causing deviations of the moment of force that is regulated to zero with $DZMP_x^d$ (activated with $P_{\mu_x,w_y} = 1$). The angle α_z between the edge and ${}_{\{s\}}x$

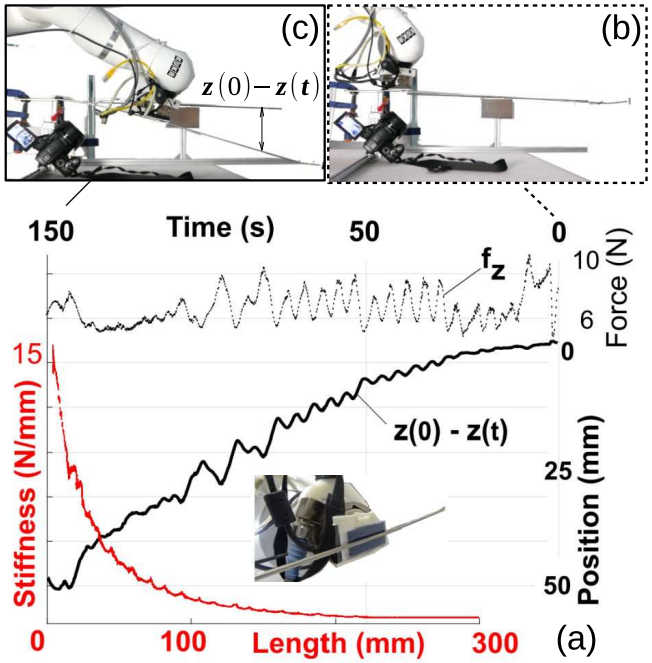

Fig. 6: Tracking the unknown shape of a flexible rod fixed from one side only: a) the time axis propagates from the right-hand side to the left; the amount of displacement versus this time is shown as the black solid line; the force history during the exploration is depicted in black dots; the stiffness along the length of this rod is represented in red solid line (from left to right); b) the rod is not bent at the beginning; c) the metal rod bends under the applied force while exploration.

is maintained at zero. Accordingly, the edge CF setpoint is the following $\boldsymbol{s}^{des} = (CoP_x^d, CoP_y^d, f_z^d, DZMP_x^d, -, \alpha_z^d)^T$ (with $P_{CoP_x,x} = P_{CoP_y,y} = P_{f_z,z} = P_{\mu_x,w_y} = P_{\alpha_z,w_z} = 1$). The tangential forces are assumed to be negligible and thus not affecting the sliding motions.

With these edge CF setpoints, the robot end-effector was able to adapt to the varying curvature of the rod with variable stiffness[1]. The rod deformed under the loads applied at the remote locations from the fixed side. The end-effector traveled 300 mm at a constant velocity $v_x \simeq 300mm/110s \simeq 2.7mm/s$). The farther the end-effector is from the initial location along ${}_{\{s\}}x$-axis, the more it deviates in ${}_{\{s\}}z$-axis (black solid line in Fig. 6). The force history is depicted with dashed black line in the figure. This feedback force varies as the metal rod bends. By sliding over the unknown surface of this object, the robot accumulates a dense shape model. As a result of this process, the stiffness of the metal rod decreases with the traveled distance, which corresponds to the time t from the beginning of the exploration:

$$k_i = f_z(t)/(z(0) - z(t)) \tag{9}$$

where $z(0)$ and $z(t)$ are the coordinates of the end-effector along ${}_{\{g\}}z$ at the beginning and at the time t. The stiffness values along the metal rod k_i with $i \in [0, ..., 300]$ versus the length of the rod are illustrated in Fig. 6, red line. The stiffness decreases from $k_0 = 15$ N/mm down to $k_{300} = 0.15$ N/mm.

[1]video https://youtu.be/-0JCF1vt5A8

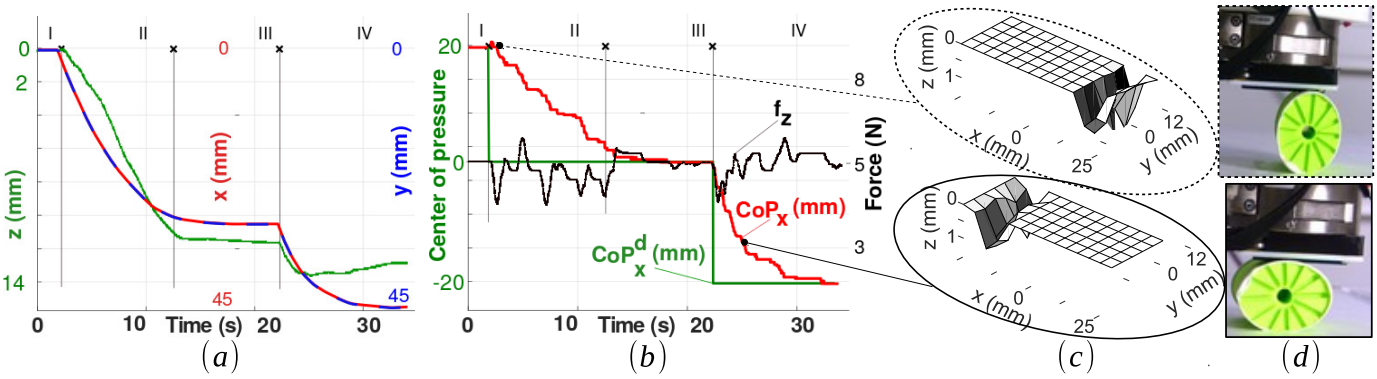

Fig. 7: Edge CF setpoints in manipulation by rolling. (a) Time history of the end-effector position along z-, x-, y-axes in green, red, blue lines, respectively. (b) Tactile features steering the edge CF. (c) Deformation of the sensing surface. (d) The object with ellipsoidal shape.

C. Manipulation by rolling an ellipsoidal object

The above experimental scenarios did not address the problem of object manipulation. The tactile feedback allows the robot arm to manipulate objects even without visual information (Fig. 1, bottom row).

When a cylindrical object rolls over the sensing surface, the location of the center of pressure changes. In order to control this rolling motion, the robot arm presses the object over a stiff surface and follows the setpoints that pull the robot end-effector to comply with the shape of the object. The object can have an ellipsoidal shape as in the following example.

In this case-scenario, we demonstrate that an object can be explored by manipulating it. For example, the major and minor axes of an ellipsoid can be unknown. The object is long enough and its axis of rotation is aligned with the ${}_{\{s\}}y$-axis. The edge CF appears in the tactile sensor image. Then, the tactile servoing can be used to roll this ellipsoidal object in the direction perpendicular to the edge. The force can be controlled and, therefore, the end-effector reacts to the changes in force and moves up and down along the ${}_{\{g\}}z$-axis (green line in Fig. 7, a) during the rolling action (red and blue lines in Fig. 7, a). The tangential motion in the ${}_{\{s\}}x$-axis that rolls the object is imposed by CoP^d_x (activated with $P_{CoP_x,x} = 1$). f^d_z (activated by $P_{f_z,z} = 1$) and $DZMP^d_y$ (activated by $P_{DZMP_y,w_x} = 1$) are also activated in order to adapt to irregular surfaces[2]. Force and CoP_x are depicted in Fig. 7 b. The modeled deformations of the sensor surface at the beginning and end of manipulation are illustrated in Fig. 7 c top and bottom, respectively. The manipulation platform is shown in Fig. 7 d. The maximum size of the sensor along its x-axis is 50 mm and the origin of the contact sensing frame is at the geometrical center of the sensor. The end-effector travels ∼24+24 mm in order to rotate the object. The contact force is kept at $\simeq 5$ N.

The parameters of the ellipse can be computed using the arc length and the difference between the maximum and minimum coordinates of the end-effector along ${}_{\{e\}}z$-axis (Fig. 8). The total arc length is given by:

$$C = \pi\big(3(a+b) - \sqrt{10ab + 3(a^2+b^2)}\big) \qquad (10)$$

[2]video https://youtu.be/imrqMHRmTSs

where a and b are the major and minor axes of the ellipse, respectively, and C is the total length of the ellipsoid. From the minimum to the maximum position along its z-axis, the end-effector travels $\frac{1}{4}$ of the total length of the ellipsoid. Hence, a and b can be derived from this equation given the obtained traveled distances of the end-effector Δx_z and $|CoP^d_x - CoP_x|$. The black cross-markers in Fig. 8 depict the minimum and maximum values of the end-effector's vertical displacement versus the rolling distance (horizontal displacement). Using the Euclidean distances along both axes of coordinates between these extreme points, the ellipsoid was reconstructed from eq. (10) and derivations described in [29] with the accuracy of 98% (black dashed ellipse in Fig. 8). The real values of the major and minor axes were 60 mm and 22 mm, respectively (red ellipse in Fig. 8).

The edge CF setpoint for manipulation by rolling a cylindrical/ellipsoidal object along ${}_{\{s\}}x$-axis is the following: $\boldsymbol{s}^{des} = (CoP^d_x, -, f^d_z, -, DZMP^d_y, \alpha^d_z = \frac{\pi}{2})^T$ (with $P_{CoP_x,x} = P_{f_z,z} = P_{\mu_y,w_x} = P_{\alpha_z,w_z} = 1$).

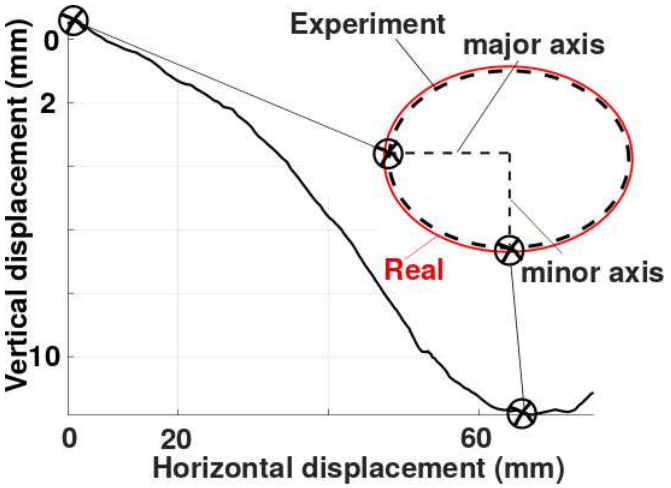

Fig. 8: Reconstruction of an ellipsoidal object (red line for the real shape). The object with unknown major and minor axes lies on a flat stiff surface. The robot end-effector pushes the object with a constant force and rolls it over the surface. Tracking the applied force results in the deviations of the end-effector (i.e. vertical and horizontal displacements). The ellipsoidal shape is reconstructed (dashed black line) using the distances between the extreme points (black crosses) of the deviations.

CONCLUSION

This paper describes an intermediate tactile-based task definition layer between the contact-based task planning and the tactile servoing control. It considers physically constrained robot end-effector motions that are confined by edge-plane contacts.

Firstly, the paper introduces the possible physical interaction scenarios when the edge-plane contact must be finely controlled (Section II). These scenarios include exploration by an edge-following task and manipulation by rolling an object with a generalized cylinder shape– both involving edge-plane contacts. The tactile servoing architecture for steering this edge-plane contact was regarded as two interacting control loops: motion control and tactile edge servoing (Section IV).

Finally, the paper demonstrates the use of edge contact formation setpoints in the aforementioned tasks. For the first time, the tactile exploration over a flexible object (Section V-B) is performed. The setpoints are defined following the ideas on natural and artificial constraints in the Task Frame Formalism [10]. An improvement of the state-of-the-art approach [5] is demonstrated in the edge-aligning task (Section V-A). The edge-aligning task and exploration can be completed using the same controller as was shown in [6]. The edge contact formation setpoints are also specified in order to manipulate by rolling an ellipsoidal shaped object whose parameters were not fully known (Section V-C). The main limitation of these setpoints is that they are applicable only for the contacts that appear on a planar surface.

The task descriptions developed in this paper allow an application designer to obtain a solution for reactive control of complex tasks, including uncertainty and multiple contact point control [16], [30], [31]. Among the presented applications, the setpoints given in Section V-B are the most suitable for these tasks, since they allow the robot to apply fine corrective motions to maintain uncertain contact points.

ACKNOWLEDGEMENTS

The authors are thankful to Vincent Padois for discussions on torque controlled robots, especially robot dynamics, to Youcef Mezouar and Almas Shintemirov for discussions on implementation of controllers.

REFERENCES

[1] P. Sikka, H. Zhang, and S. Sutphen, "Tactile servo: Control of touch-driven robot motion," in *Exper. Robotics III*, 1994, pp. 219–233.
[2] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *J. of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.
[3] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Trans. Syst., Man, Cybern*, vol. 11, no. 6, pp. 418–432, 1981.
[4] H. Zhang and N. Chen, "Control of contact via tactile sensing," *IEEE Trans. on Rob. and Autom.*, vol. 16, no. 5, pp. 482–495, 2000.
[5] Q. Li, C. Schürmann, R. Haschke, and H. Ritter, "A control framework for tactile servoing," in *Robotics: Science and Systems*, 2013.
[6] Z. Kappassov, C. Juan, and P. Veronique, "Zmp features for touch driven robot control via tactile servo," in *Intern. Symp. on Exper. Rob.*, vol. 21, 2016.
[7] Z. Kappassov, J.-A. Corrales-Ramon, and V. Perdereau, "Tactile sensing in dexterous robot hands Review," *Rob. and Auto. Sys.*, 2015.
[8] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Touch driven controller and tactile features for physical interactions," *Rob. and Auto. Syst.*, vol. 123, p. 103332, 2020.
[9] V. Perdereau and M. Drouin, "A new scheme for hybrid force-position control," *Robotica*, vol. 11, no. 5, p. 453464, 1993.
[10] H. Bruyninckx and J. De Schutter, "Specification of force-controlled actions in the "task frame formalism"-a synthesis," *IEEE Trans. on Rob. and Autom.*, vol. 12, no. 4, pp. 581–589, 1996.
[11] T. Lefebvre, H. Bruyninckx, and J. D. Schutter, "Task planning with active sensing for autonomous compliant motion," *Intern. J. of Rob. Res.*, vol. 24, no. 1, pp. 61–81, 2005.
[12] A. D. Berger and P. K. Khosla, "Using tactile data for real-time feedback," *Intern. J. of Rob. Res.*, vol. 10, no. 2, pp. 88–102, 1991.
[13] J. Platkiewicz, H. Lipson, and V. Hayward, "Haptic edge detection through shear," *Sci. Rep.*, vol. 6, p. 23551, 2016.
[14] V. Padois, S. Ivaldi, J. Babi, M. Mistry, J. Peters, and F. Nori, "Whole-body multi-contact motion in humans and humanoids: Advances of the codyco european project," *Rob. and Auton. Syst.*, vol. 90, pp. 97 – 117, 2017.
[15] N. Hogan, "Impedance control: An approach to manipulation," in *1984 American Control Conference*, 1984, pp. 304–313.
[16] N. Sommer and A. Billard, "Multi-contact haptic exploration and grasping with tactile sensors," *Rob. and Auton. Syst.*, vol. 85, pp. 48–61, 2016.
[17] 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 Robot. Automat. Lett.*, vol. 4, no. 2, pp. 2101–2107, 2019.
[18] S. J. Lederman and R. L. Klatzky, "Relative availability of surface and object properties during early haptic processing," *J. of Exper. Psych. Human Percep. and Perform.*, vol. 23, no. 6, pp. 1680–1707, 1997.
[19] P. K. Allen, "Integrating vision and touch for object recognition tasks," *Inter. J. of Robot. Res.*, vol. 7, no. 6, pp. 15–33, 1988.
[20] M. Nakatani, A. Sato, S. Tachi, and V. Hayward, "Tactile illusion caused by tangential skin srain and analysis in terms of skin deformation," in *Proceedings of Eurohaptics*. Springer-Verlag LNCS 5024, pp. 229–237.
[21] D. Baimukashev, Z. Kappassov, and H. A. Varol, "Shear, torsion and pressure tactile sensor via plastic optofiber guided imaging," *IEEE Robot. and Autom. Lett.*, vol. 5, no. 2, pp. 2618–2625, 2020.
[22] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robot.s and Autom.*, vol. 8, no. 3, pp. 313–326, 1992.
[23] Q. Li, C. Elbrechter, R. Haschke, and H. Ritter, "Integrating vision, haptics and proprioception into a feedback controller for in-hand manipulation of unknown objects," in *IEEE/RSJ Intern.l Conf. on Intell. Rob. and Syst.*, 2013, pp. 2466–2471.
[24] Q. Li, R. Haschke, and H. Ritter, "A visuo-tactile control framework for manipulation and exploration of unknown objects," in *IEEE-RAS Humanoids*, 2015, pp. 610–615.
[25] N. Lepora and J. Lloyd, "Pose-based tactile servoing: Controlled soft touch using deep learning," *IEEE Robotics Automation Magazine*, pp. 2–15, 2021.
[26] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 818–824.
[27] Z. Kappassov, J.-A. Corrales-Ramon, and V. Perdereau, "Simulation of tactile sensing arrays for physical interaction tasks," in *IEEE/ASME Intern.l Conf. on Adv. Intell. Mechatr.*, 2020, pp. 196–201.
[28] K. Aquilina, D. A. W. Barton, and N. F. Lepora, "Shear-invariant sliding contact perception with a soft tactile sensor," in *Intern Conf on Rob. and Automat.*, 2019, pp. 4283–4289.
[29] Z. Kappassov, "Touch driven dexterous robot arm control," Ph.D. dissertation, University Pierre et Marie Curie, France, 2017.
[30] S. Soltan, A. Oleinikov, M. F. Demirci, and A. Shintemirov, "Deep learning-based object classification and position estimation pipeline for potential use in robotized pick-and-place operations," *Robotics*, vol. 9, no. 3, 2020.
[31] P. Vulliez, J. P. Gazeau, P. Laguillaumie, H. Mnyusiwalla, and P. Seguin, "Focus on the mechatronics design of a new dexterous robotic hand for inside hand manipulation," *Robotica*, vol. 36, no. 8, p. 12061224, 2018.