

Tactile-Reactive Roller Grasper

Shenli Yuan , Shaoxiong Wang , Radhen Patel, Megha Tippur, Connor L. Yako, Mark R. Cutkosky , *Fellow, IEEE*, Edward Adelson, *Life Fellow, IEEE*, and J. Kenneth Salisbury, *Life Member, IEEE*

Abstract—Manipulation of objects within a robot’s hand is one of the most important challenges in achieving robot dexterity. To address this challenge, Roller Graspsers use steerable rolling fingertips. The fingertips impart motions and exert forces to achieve six degree of freedom mobility and closed-loop grasp force control. The design reported here uses image processing from cameras placed inside steerable compliant rollers to track contact conditions and locations. Integration of this data into a controller enables a variety of robust in-hand manipulation capabilities. We demonstrate that the same information can be used to reconstruct object shape. In addition, we show that by converting in-hand manipulation from a discontinuous process, with fingers frequently attaching and detaching from the object surface, to a continuous process, we can implement a convergent control loop that minimizes errors that otherwise accumulate during large object motions. The difference is apparent when comparing the results of an object rotation using a discontinuous finger-gaiting approach, as would be required without rolling fingertips, to the results obtained with continuous rolling. The results suggest that hybrid rolling fingertip and finger-gaiting approaches to manipulation may be a promising future research direction.

Index Terms—Dexterous manipulation, in-hand manipulation, robot end-effectors, tactile sensing.

I. INTRODUCTION

SIMILAR to how the dexterity of human hands allows us to accomplish a variety of everyday tasks, the manipulation capabilities of robots are necessary to accomplish a wide range of complex tasks in different environments. Out of all the manipulation tasks a human hand could perform, in-hand manipulation requires the most dexterity [1]. To achieve such dexterity, it is only natural that researchers sought to create and

Received 15 June 2024; revised 12 December 2024; accepted 25 January 2025. Date of publication 19 February 2025; date of current version 14 March 2025. The work of Shenli Yuan was supported by Stanford Interdisciplinary Graduate Fellowship. This work was supported by Toyota Research Institute. This work was supported by funding from the Toyota Research Institute. This article was recommended for publication by Guest Associate Editor K. Althoefer and Editor M. Yim upon evaluation of the reviewers’ comments. (*Shenli Yuan and Shaoxiong Wang contributed equally to this work. Edward Adelson and Kenneth Salisbury provided equal advising (Corresponding author: Shenli Yuan.)*)

Shenli Yuan, Connor L. Yako, Mark R. Cutkosky, and J. Kenneth Salisbury are with Stanford University, Stanford, CA 94305 USA (e-mail: shenliy@stanford.edu; clyako@stanford.edu; cutkosky@stanford.edu; kenneth.salisbury@gmail.com).

Shaoxiong Wang, Radhen Patel, Megha Tippur, and Edward Adelson are with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: wang_sx@mit.edu; radhen@mit.edu; mhtippur@mit.edu; adelson@csail.mit.edu).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TRO.2025.3543324>, provided by the authors.

Digital Object Identifier 10.1109/TRO.2025.3543324

use devices of similar function and construction to the human hand. Recent research has shown this to be a promising direction, with a simulated 20 degree-of-freedom (DoF) Shadow Hand [2] able to manipulate in-hand over 2000 objects in a variety of palm poses [3]. However, while having high dexterity is desirable for robot hands, performing in-hand manipulation with robot hands or fingers based on their human counterparts may not be optimal in all situations. To draw inspiration from an analogous development in legged robots, the quadruped ANYmal [4], and its predecessor from company Swiss-Mile, have demonstrated both efficient locomotion on smooth surfaces and a great ability to navigate rough terrain using a combination of legs and wheeled feet. A similar hybrid of finger mechanisms and rolling fingertips may be efficient for dexterous in-hand manipulation.

This mindset led to the development of the Roller Graspsers [5], [6], which introduced a way to manipulate objects within the hand. They have demonstrated success in various in-hand manipulation applications. These include manipulation tasks of various objects such as a piece of paper, a 6-sided die, and a 3-D printed cube using continuous rolling, completed through both open-loop control and closed-loop control with the help of QR tags. However, one of the major limitations of the previous works is the lack of local contact information. Tactile sensors are a crucial component for successful robot in-hand manipulation, as has been shown with various linkage-based robot hands [7], [8], [9].

This article details our investigation into the utility of in-hand manipulation through closed-loop tactile-guided rolling. We developed the Tactile-Reactive Roller Grasper (TRRG) shown in Fig. 1, a robotic hand that utilizes camera-based tactile sensing (based on GelSight [10]) and steerable rolling fingertips to perform a variety of manipulation tasks. The developed sensor information processing allows us to extract necessary contact information that can be used in real-time control. The advantages of combining the Roller Grasper and the GelSight sensor are two-fold. First, incorporating tactile sensing greatly improves the in-hand manipulation capabilities of the Roller Grasper, by enabling the grasper to detect and exploit local contact information between the rollers and the grasped object to perform stable, and robust in-hand manipulation. Second, the steerable rollers enable the tactile sensor to easily scan potentially large and complex surfaces, leading to efficient and accurate 3-D reconstructions. Our robotic system includes the TRRG, a Universal Robots UR-5e robot arm, and a computer, as shown in Fig. 2. Along with the design of the hand, we present sensor calibration and processing techniques related to the continuous

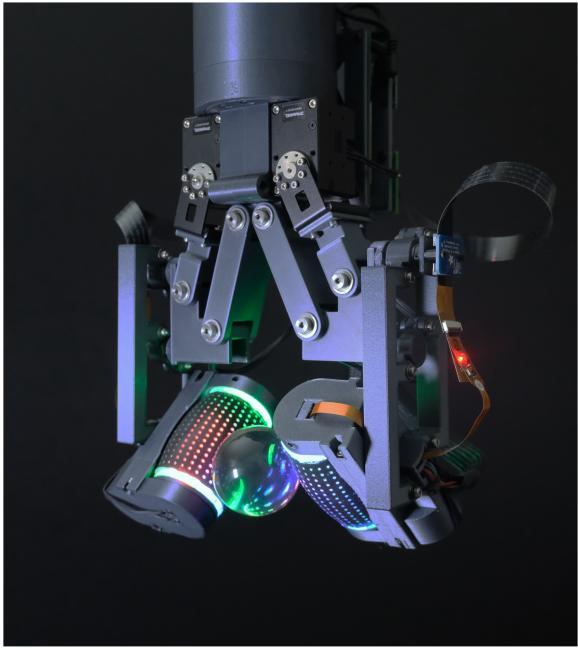


Fig. 1. Physical prototype of the TRRG with rollers in a crossed-axis configuration for imparting screw motions to a clear ball.

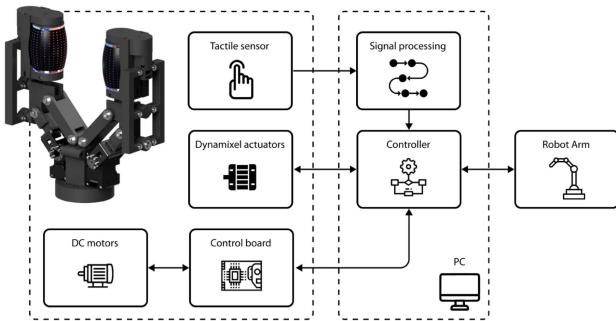


Fig. 2. System setup consisting of the TRRG, a PC, and a UR-5e robot arm. The PC handles the tactile signal processing and controls the TRRG and the robot arm.

rolling nature of the sensor. We also present a control algorithm based on differential geometry and the availability of tactile information. The TRRG completed various tasks with the help of tactile-guided rolling, such as tracing a flexible cable bidirectionally, imparting screw motions to a clear acrylic sphere, performing robust object reorientations, and picking up a single card from a deck of cards. In addition to the demonstrations, we performed experiments to compare manipulations through rolling and finger gaiting with regrasping, with or without tactile feedback. We found that while rolling appears to be more stable in the open-loop case, performing grip adjustments with tactile-guided rolling further promotes stable manipulation for both the regrasping and pure rolling manipulation cases.

II. RELATED WORKS

A. Robot Hand for In-Hand Manipulation

Over the past century, there have been a number of robotic and prosthetic hands designed with in-hand manipulation in

mind [11]. Most of these hands are linkage-based, including anthropomorphic hands [2], [12], [13] and other fully actuated hands [14], [15], many of which have shown promising in-hand manipulation capability. Alternatively, in-hand manipulation can be achieved without intentionally switching contact locations, but this severely limits the object's range of motion. There have also been works that use underactuated linkage-based hands [16], [17] to achieve in-hand manipulation, sometimes exploiting contact with the environment to provide assistance. Another approach toward in-hand manipulation is to use nonanthropomorphic hands [18], [19]. One particular nonanthropomorphic approach is to use active surfaces that allow the hand to move a grasped object without lifting its fingers. Often these designs have fixed conveyor directions [20], [21], [22], [23] but grippers with belts on rotating platforms have also been demonstrated [24]. Previous Roller Grasps, [5] and [6], further developed this concept and incorporated steerable rollers for greater object mobility. However, as mentioned before, the challenge of incorporating traditional tactile sensors into active rolling surfaces has not been addressed and leads us to choose a vision-based imaging approach for this application.

B. Vision-Based Tactile Sensing

Vision-based tactile sensing [25], [26], [27], [28], [29], [30], [31] has become increasingly popular in recent years due to the high resolution, robustness, and ease of communication with small digital cameras. Common designs include a piece of transparent or translucent elastomer, a camera, and a lighting system. When in contact, the sensor captures information about the surface properties of a grasped object and one can compute information relevant to manipulation including the contact kinematic type, area, centroid location, and pressure and shear stresses.

While tactile sensors based on resistance, capacitance, and piezoelectricity [7], [32], [33] can be good options for linkage-based hands, they are difficult to integrate with continuously rolling elements because of the need to route wires to the contact areas. Vision-based tactile sensors, in comparison, can transmit tactile information through light, without requiring an electrical connection to the rotating elements. Based on this realization, there have been previous works that integrate vision-based tactile sensing into passive rollers for inspection tasks [34], [35]. It was demonstrated that rolling action greatly improves the efficiency of inspection, especially when scanning large areas. However, since the rollers in these works are passive, they rely on motions of the robotic arm and cannot impart motions to an object; thus, these systems do not support in-hand manipulation.

In this work, we integrate a category of vision-based tactile sensors, known as GelSight sensors [10], into actively-driven rollers. This combination enables robust in-hand manipulation through closed-loop rolling contact, and facilitates efficient inspection of the geometric properties of a grasped object during manipulation. In addition to the shear forces and 2-D contact geometry, GelSight sensors can provide high-resolution 3-D contact geometry by applying photometric stereo. The 3-D information can be further processed and used to estimate contact

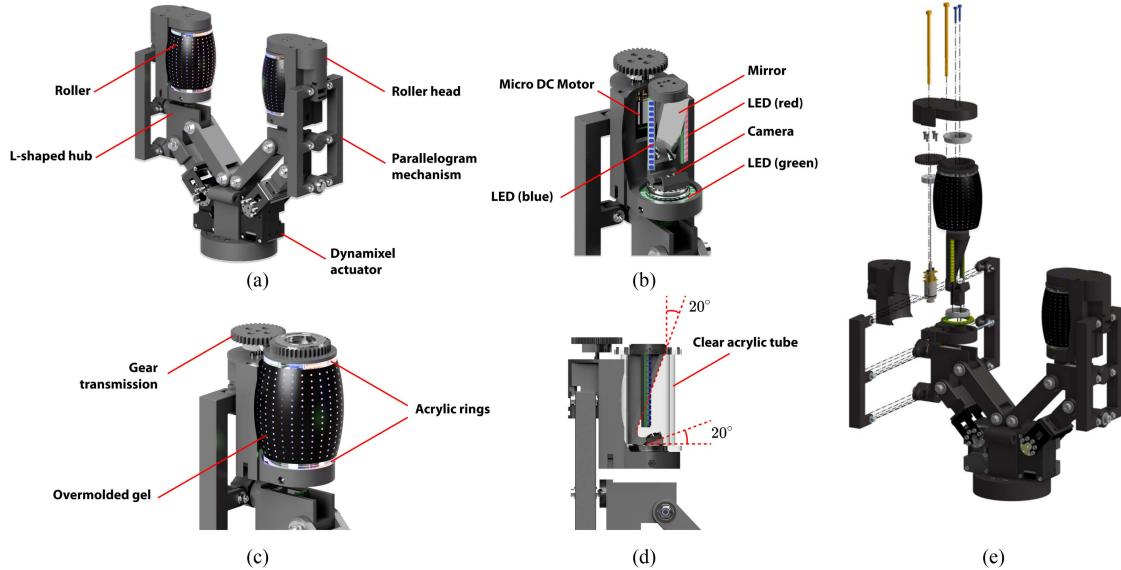


Fig. 3. CAD renderings of the mechanical design of the TRRG. (a) Fully assembled hand. (b) Nonrotating optical components (stator) inside the roller, the rotating part is called the rotor. (c) Roller assembly and drive motor. (d) Camera and mirror in the roller. (e) Exploded view of the roller.

forces and object poses, ultimately leading to surface reconstruction. In comparison to [10], we have redesigned the sensor's form-factor to fit inside driven, steered rollers, and we have designed the corresponding information processing pipeline to accommodate this new geometry.

III. DESIGN

A. Hand Design

The TRRG is a two-fingered grasper with each finger consisting of three actuated DoF. The design of the grasper is shown in Fig. 3. The base DoF is driven by a Robotis Dynamixel XM430-W350 actuator through a four-bar parallelogram linkage. The mechanism enables up to 160 mm opening between the rollers. The finger can exert a normal force of 68.3 N when the input link of the four-bar linkage is vertical (fully closed grip). A micro dc motor embedded in the L-shaped hub controls the second DoF and is capable of pivoting the roller head about a horizontal axis between $\pm 90^\circ$ through a five-bar parallelogram mechanism seen in Fig. 3(a). The mechanism was improved from that in our previous work [5] to allow for a greater range of motion. Another micro dc motor, shown in Fig. 3(b), is embedded at the back of the roller head to drive the roller through spur gears. This motor has a gear ratio of 297.9: 1 with maximum speed of 90 r/min. Combined with the geometry of the roller and its gear reduction, the maximum speed the roller can impart to an object at the contact location is 200.9 mm/s. Unlike the previous generations of the Roller Grasper, the roller motor is located external to the roller to make space for the tactile sensing components. The physical dimensions of the TRRG are presented in Table I.¹

¹The complete CAD assembly can be found through the following link: [Online]. Available: <https://github.com/yuanshenli/Tactile-Reactive-Roller-Grasper>.

TABLE I
PHYSICAL DIMENSIONS OF THE TRRG LABELS SHOWN IN FIG. 4(A)

Properties	L_0	L_1	L_2	L_3	R_0	D_0
Value (mm)	29.75	35.25	60	87.25	100	80

There are multiple reasons the TRRG uses a two-finger design instead of three, as in [5] and [6]. The first reason is to avoid instability for cylindrical rollers when placed on direct drive fingers. This issue is especially problematic when the cylindrical rollers are not on parallel planes, as discussed in [5]. This issue was mitigated in [6] by using spherical rollers on a direct drive finger. However, due to constraints imposed by the sensor design, we were unable to design a spherical roller with the majority of the surface area sufficiently illuminated by the LEDs. Instead, we opted to place each cylindrical roller on a parallelogram mechanism, allowing the rollers to move in parallel planes and avoid the instability discussed in [5]. Even with this mechanism, a three-finger design (with fingers 120° apart like in [6]) would constrain the smallest size of object that can be manipulated, especially when the cylindrical rollers are oriented horizontally. We, therefore, opted for two opposing fingers. While we could potentially add a third finger that operates in a plane normal to that of the opposing fingers in the existing design, it adds complexity and potentially obstructs certain manipulations we would like to perform, as demonstrated in the later sections. We believe the two-finger design at this stage has allowed us to extensively explore rolling manipulation with tactile sensing.

The 3-D kinematics of the TRRG are shown in Fig. 4(b)–(f). In principle, the TRRG is capable of performing full 6-DoF spatial manipulation in its various configurations. Our various experimental scenarios in the later sections utilized one or combinations of these different rotation or translation directions. However, in this work, our emphasis is on how tactile sensing

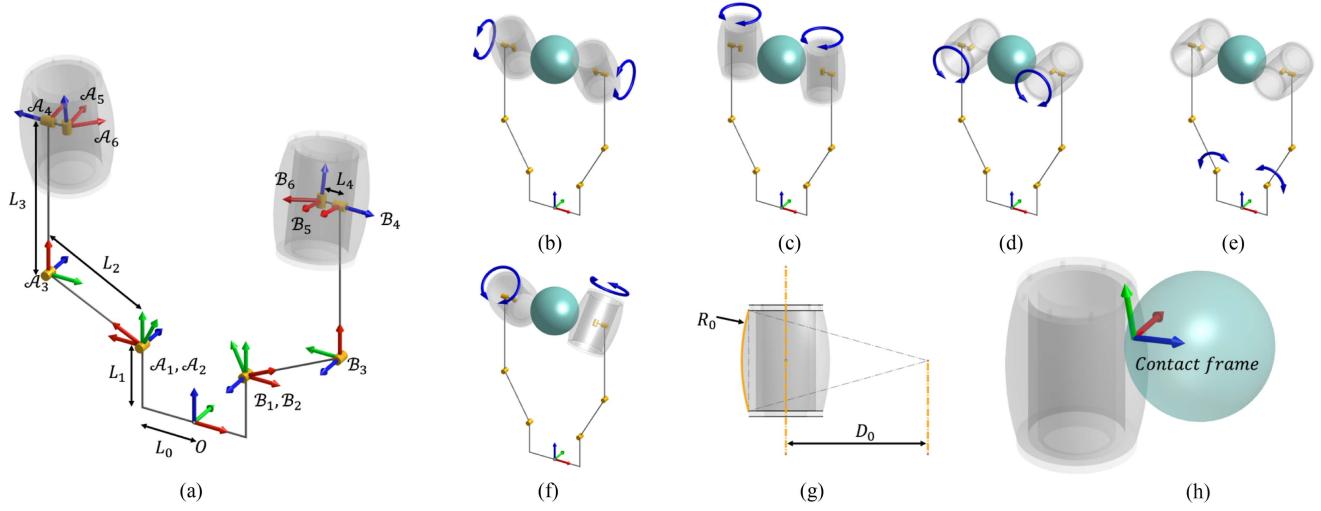


Fig. 4. 3-D kinematics and roller configurations. (a) TRRG frame definitions. \mathcal{A} and \mathcal{B} represent each finger. Frame O is the fixed frame located at the base of the hand. The numerical subscripts represent frames attached at different locations of the hand (e.g., A_1 represents finger \mathcal{A} frame 1). Specifically, frame 1 is attached to the base of the finger, frames 2 to 4 are attached to each joint on the finger, frame 5 is the origin frame of the roller, and frame 6 is attached to the roller. The X , Y , Z axes are represented in red, green, and blue, respectively. Frame O is the world frame with which we reference the manipulation directions. (b) Object rotation about X_O . (c) Object rotation about Z_O or object translation along Y_O , depending on the rolling directions of the two rollers. (d) Object rotation about Y_O or object translation along Z_O , depending on the rolling directions of the two rollers. (e) Any rotation or translation in directions within $Y_O - Z_O$ plane are possible with different pivot positions. (f) Object screw motion (coupled rotation and translation). (g) Roller geometry. (h) Contact frame with the origin at the contact location. Assuming point contact, the contact location corresponds to where the grasped object makes contact with the roller. The X , Y , Z axes are represented in red, green, and blue, respectively.

can be integrated into the rolling fingertip and how it aids rolling manipulation. We do not focus on the details of how individual manipulation directions are achieved, as discussed in [5].

B. Sensor Design and Fabrication

One of the most important considerations for vision-based tactile sensors is designing for clear optical paths. As a result, the mechanical structure of the roller consists of a clear acrylic tube glued with clear acrylic rings at each end, allowing unobstructed light passage between the light source, sensing area, and camera [see Fig. 3(c) and (d)]. A 3D-printed gear is attached to one of the acrylic rings. Eight dowel pins arranged around the perimeter are inserted into both the gear and the acrylic ring to provide torque transmission. A clear gel (elastomer) is molded directly over the acrylic tube to provide a soft, rotationally symmetric fingertip with optical transparency. It is worth noting that the stiffness of the elastomer will affect the performance of the sensor. A more compliant elastomer provides higher sensitivity, suitable for manipulating more delicate objects, while a stiffer elastomer tolerates a larger range of applied normal forces. In this work, the stiffness of the elastomer was chosen to be sufficient across the experiments we conducted. When necessary, the stiffness can be customized for specific applications. In addition to manipulation, the material choice and finger size also affect the range of objects that can be acquired with the grasper. The lower bound on object size depends on the deformation capability of the fingertips, the coefficient of friction, and the size of the fingertips. While the TRRG is not optimized for grasping very small or thin objects (examples in Fig. 12), other grippers have been designed for such tasks. For example, Do

et al. [36] demonstrated a gripper with fingertips made of the same material as the TRRG picking up 4–40 screws, and Zhang et al. [37] demonstrated the use of rotating fingertips with friction to pick up a sheet of paper.

The finger and the elastomeric covering are designed to be compact, to provide a large workspace for manipulating objects. The lower limit of the roller size (the roller's height, and top and bottom radii) is constrained by the components located inside the roller with the fundamental limits being the focal length and size of the camera. As shown in Fig. 3(d), the camera is located at the bottom of the stator and angled at 20° from its horizontal mounting surface. It streams images from the sensing area through a mirror oriented 20° from the rolling axis. The purpose of the mirror is to increase the distance between the sensing surface and the camera, which allows us to use a smaller acrylic tube when the camera focal length is fixed. The thickness of the elastomer is determined based on the following constraints:

- 1) it needs to be close to the width of the LED array to allow unobstructed light transmission;
- 2) it needs to be thin enough so that the directional light can illuminate the necessary surface area at the nominal contact area;
- 3) it needs to be thick enough to ensure the grasped object is always in contact with a compliant surface.

More specifically, if we optimize for grasping performance (as discussed in [5] and [6]), the roller should ideally be spherical to easily adapt to different object geometries and be as small as possible. However, because the sensing surface needs to be fully illuminated by the LEDs, adopting a spherical design would cause the bulge of the roller elastomer to be too significant. This

would result in the widest section of the elastomer not being illuminated, affecting sensor performance. After considering the tradeoffs, the geometry of the roller elastomer was empirically determined with a height of 54 mm, a longitudinal radius of curvature of 100 mm, and a latitudinal radius of curvature between 20 mm (at the two ends) and 23.69 mm (in the middle). This configuration allows the roller to retain the bulged shape, providing more robust object grasping while ensuring the entire sensing area is illuminated by the LEDs. With the design of TRRG, the resulting tactile sensor has a sensing area that forms a 90° angle to the axis of the roller, which is equivalent to a roughly 33 × 55 mm area when projected to a tangent plane.

Lighting: The lighting system was designed to enable accurate 3-D reconstruction based on photometric stereo, while fitting into the compact form-factor of the roller. To satisfy the requirements, we modified the design of the lighting system from the GelSight Wedge sensor [38] to be suitable to illuminate the curved roller surface.

As shown in Fig. 3, two LED bars (one blue and one red, respectively) are located near either vertical edge of the mirror to provide directional light from two different directions toward the sensing area. A green LED ring attached below the roller shines light through the bottom acrylic plate to provide the third color component essential for 3-D reconstruction. Clear Ultraviolet Curing Epoxy Resin, Limino (UV resin) connects the clear acrylic plates to the clear acrylic tube, keeping the interface optically clear for the light from the LED ring.

Camera: We used a Raspberry Pi camera with a 120° field of view (FOV), allowing us to obtain a relatively large sensing area while fitting inside the interior of the roller. The camera was customized with a 200 mm long flex cable, so the bulky connector can be located outside the roller. We streamed the video from the camera through the Raspberry Pi to a PC using *mjpg_streamer*, maintaining a 640 × 480 resolution at 30 Hz. The images were further processed in the PC and we were able to achieve a 30 Hz update rate for the processed sensor signals (further details are provided in Section V-A). The update rate is highly dependent on the number of iterations used for the marker tracking optimization.

Elastomer: We designed and fabricated the seamless elastomer to obtain continuous tactile signals during rolling. In comparison, another fabrication technique is to cast a piece of flat elastomer to be wrapped around the rotor core [34], [35], which would be less durable and result in discontinuous sensing signals at the seam.

Fig. 5(a) shows the sequence of the elastomer fabrication. We first 3D-printed the positive mold and applied a coating layer (XTC-3D, Smooth-On, Inc.) on the curved surface to achieve a smooth finish. A stretchable negative mold was then cast using translucent silicone (Ecoflex 00-30, Smooth-On, Inc.). We applied a layer of mold release (Ease Release 200, Mann Release Technologies) to allow the negative mold to be easily removed from the gel later. The material choice for the negative mold is not critical as long as it can be easily removed. Next, the clear silicone (XP-565, Silicones, Inc.) on the roller was cast together with the acrylic tube. We applied a layer of primer (DOWSIL

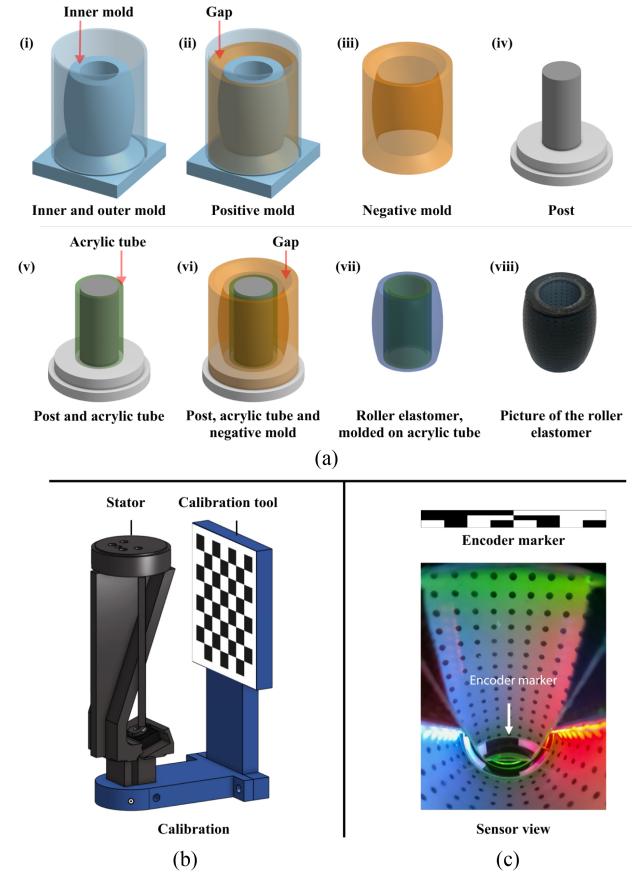


Fig. 5. Sensor construction and calibration. (a) Seamless roller elastomer construction process: We used a two-part mold to create a negative mold, which is then placed around a post along with an acrylic tube. The elastomer is molded between the negative mold and the acrylic tube. Finally, the negative mold and the post are removed to obtain the roller elastomer. Notes: (i) Inner mold has the shape of the roller elastomer. (ii) Fill the indicated gap to create the negative mold. (iii) Remove the inner and outer mold and keep the negative mold. (iv) Prepare a post to locate the acrylic tube for the elastomer to be molded on. (v) Put the acrylic tube on the post. (vi) Place the negative mold, and fill the indicated gap to create the roller elastomer. (vii) Remove the molds and obtain the roller elastomer. (viii) Picture of a fabricated roller elastomer. (b) 7 × 8 Checkerboard mounted on a calibration tool to get the sensor intrinsic and extrinsic matrices. (c) Pattern of the encoder marker to provide precise position encoding, and the corresponding image from the sensor view.

PR-1200, RTV Prime Coat, DOW) on the outside of the acrylic tube before casting to enhance the acrylic-to-elastomer bonding. Finally, we sprayed a layer of opaque gray (Lambertian) silicone ink (Print-On Silicone Ink, Raw Material Suppliers) onto the surface of the roller, allowing the surface to achieve Lambertian reflectance.

Note that the robustness of the sensor elastomer is highly dependent on the use cases. The TRRG allows the elastomer to be used in rolling contacts, making it more robust [39] compared to sliding contacts (such as in [40] and [41]). The robustness of similar sensors has been characterized in previous works such as [29], [42].

Markers: To provide information of shear and torsional forces, we added arrays of markers on the surface of the roller. The markers were created on a lasercutter with a rotary attachment,

which etched away the gray coating at the predefined marker locations. The rollers were then applied with a layer of black silicone ink, resulting in black markers on a gray background in the camera view.

The number of the markers is empirically determined, with 15 different latitudes of markers spread along the longitudinal direction of the roller (along the axis) and 26 markers around the latitudinal direction of the roller (around the roller). Marker tracking is the most computationally intensive part of our sensor signal processing pipeline, and the number of markers used in the TRRG allows the tactile information to be updated at a rate of 30 Hz, which is the maximum update rate achievable because the sensor image is streamed at this rate. The computation time for marker tracking roughly follows $\mathcal{O}(N_{mk}^2 N_{sol})$, where N_{mk} is the number of markers and N_{sol} is the number of solutions sampled in each frame during random optimization (details in Section IV-A). For other task-specific designs, a higher density will lead to higher resolution measurements of the force fields at the cost of higher computational requirements and a larger blocked area for depth feedback.

IV. METHODS

A. Tactile Signal Processing

This section discusses the signal processing techniques for the raw tactile signal. It also addresses the challenges created by the continuous rolling and convex sensing surface, and the corresponding solutions that we developed.

Encoding: In order to achieve 3-D reconstruction and marker tracking, the signal processing algorithms require each image to be compared with a reference image taken in the absence of contact [10]. Unlike GelSight sensors with the conventional form-factor, our sensing area expands to the entire perimeter of the roller and, thus, multiple reference images in correspondence to different roller positions need to be taken in order to properly process the sensor signal. This requires the algorithm to find the correct reference image for each roller orientation. However, due to backlash in the transmission and latency between the actuator and camera, the roller motor encoder cannot be used to correspond a given image to its designated reference. Therefore, we attached an encoder inside the camera FOV, as shown in Fig. 5(c), in order to match a given image with its reference for real-time tracking and minimal sensor noise. The encoder designed with this method can achieve pixel-level precision.

During the calibration process, the roller slowly rotates at a constant speed, allowing the camera to record reference images along with the encoder images in order to construct a lookup table for each frame. During manipulation, we extract the encoder portion of the image and find the L2 distance between the current encoder image and references from the lookup table to determine the corresponding reference image. Finding the correct reference image is a crucial early step toward the successful processing of tactile signals.

Surface projection: Camera matrices are used to calculate the correspondence between the points on the sensor surface in 3-D and the 2-D camera image pixels. The transformation [43] can

be represented as

$$\lambda \begin{bmatrix} v_s \\ u_s \\ 1 \end{bmatrix} = \mathcal{K}[R|t] \begin{bmatrix} X_5 \\ Y_5 \\ Z_5 \end{bmatrix} \quad (1)$$

where $\mathbf{u}_s = (u_s, v_s)^T$ represents the image coordinates of the sensor input; λ is a scale factor; \mathcal{K} is the camera intrinsic matrix; $[R|t]$ is the camera extrinsic matrix, with rotation R and translation t ; $(X_5, Y_5, Z_5)^T$ represents the 3-D coordinates in the sensor frame, shown as Frame $\mathcal{A}_5/\mathcal{B}_5$ in Fig. 4(a). Note that unlike $\mathcal{A}_6/\mathcal{B}_6$, which are fixed to the rollers, $\mathcal{A}_5/\mathcal{B}_5$ are fixed to the cameras or the stator inside the roller, which are not rotating.

The camera was calibrated using a 7×8 checkerboard. The camera, along with the mirror, was first mounted to the 3-D printed housing and calibrated before the stator was assembled with the rest of the roller head. During camera calibration, multiple sensor images were collected with different checkerboard poses, which were later used for providing the camera intrinsic matrix \mathcal{K} . The extrinsic matrix $[R|t]$ was derived by taking the image of the checkerboard and using its known position with respect to frame $\mathcal{A}_5/\mathcal{B}_5$ when it is rigidly mounted on the stator, as shown in Fig. 5(b). We applied OpenCV *calibrateCamera* [44] to the image pixels and their corresponding 3-D positions to get the intrinsic matrix \mathcal{K} and the extrinsic matrix $[R|t]$.

3-D reconstruction: The 3-D positions of the points on the convex sensing area can be projected from the Cartesian space to the 2-D camera image space using the camera matrices. Because the geometry of the roller is known, this projection also allows us to trace the 3-D position of a point given its 2-D coordinate in the image. The mapping of the points on the sensing area between their 3-D positions and 2-D image pixels is saved for 3-D reconstruction when an external object is in contact with the roller.

As shown in Fig. 6, when an object is in contact with the roller, the elastomer on the roller is deformed, creating a shaded image that is recorded by the camera. After unwarping the image into a rectangular shape (with the same pixel density along its horizontal and vertical axes), we applied photometric stereo to create a depth image: each pixel on the depth image will have a corresponding depth value, indicating the offset from its position on the undeformed roller surface. We apply this depth image on top of the mapping described previously to reconstruct the 3-D geometry of the contacted object. This is accomplished by subtracting the offset of each pixel in the depth image from its corresponding 3-D position along the surface normal direction.

The photometric stereo technique used in this work is developed based on previous work in 3-D reconstruction using a planar elastomer. Specifically, we first transformed the shaded image into surface normals, and then applied the fast Poisson solver [45] for integration to produce the depth image. Further details of this method can be found in [10].

Marker tracking: The shear force estimation can be obtained by motion analysis, i.e., analyzing the marker displacement on the sensor in comparison with the reference images. During operation, markers are constantly disappearing and appearing from the boundaries of the sensor image due to the rotation of the

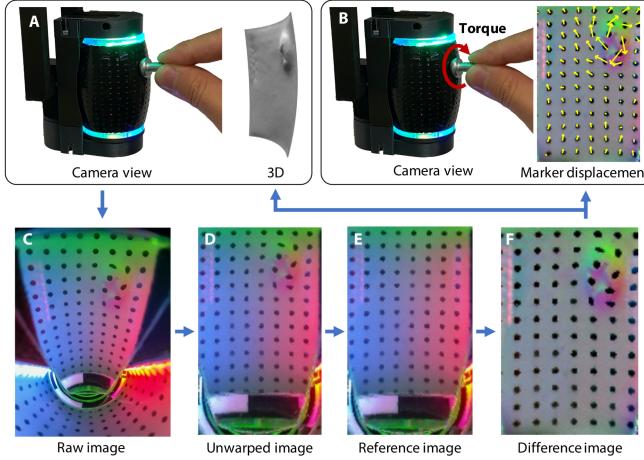


Fig. 6. 3-D reconstruction and marker tracking. (a) Camera view shows a screw head pressing on the roller sensor, and the 3-D view shows the estimated 3-D reconstruction. (b) Camera view demonstrates the torque exerted on the roller sensor, and the marker displacement visualizes the magnified motion of the markers captured from the sensor. (c) Camera inside the roller sensor captures the raw image, and the sensing area is captured in the mirror. (d) Raw image is unwarped into a rectangular image. (e) Reference image is extracted with the encoder marker from the unwarped image. (f) Difference image is calculated between the unwarped image (after contact) and the reference image (before contact). It is further processed to get the 3-D reconstruction and marker displacement.

roller, making the calculation of the marker displacement field difficult. A sensor image might even have a different number of markers compared to its reference image because certain markers are located at the very edge of the sensing area. Such a problem becomes especially prominent when the sensing area is constantly moving during the rolling motion. In these situations, techniques using marker tracking with nearest temporal matching [10], [28] or optical flow [46], [47] tend to generate erroneous results. Instead, we adopted random optimization to reliably track marker displacement during rolling by maximizing for marker flow smoothness [48], which assumes that nearby markers move with similar velocities, as well as minimizing marker mismatch between frames.

More formally, we define marker locations in the reference frame as $\{\mathbf{u}'_1, \mathbf{u}'_2, \dots, \mathbf{u}'_s\}$, and marker locations in the current frame as $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k\}$, where $\mathbf{u}', \mathbf{u} \in \mathbb{R}^2$ represent the markers' pixel coordinates, and s is not necessarily equal to k . The goal of marker tracking is to find the correspondence between $\{\mathbf{u}'_i\}$ and $\{\mathbf{u}_j\}$. For each $\{\mathbf{u}'_i\}$, we define m_i as the index of its corresponding marker in the current frame, i.e., the marker at \mathbf{u}_{m_i} overlaps with \mathbf{u}'_i when the roller is untouched ($m_i > 0$), and $m_i = -1$ when it cannot find a match in the current frame.

The commonly used nearest matching algorithm [10], [28], tends to converge at locally optimal solutions, which causes mismatches for large marker displacements or markers that move out of frame. Instead, we used an optimal matching algorithm that prioritizes the flow smoothness despite the high possibility of marker mismatches introduced by the rolling mechanism. The formulation is described below with a graphic illustration shown in Fig. 7.

For simplicity, we define the indices for the neighboring markers of \mathbf{u}'_i as the set $N_i = \{n_{il}\}$, where each neighboring marker at $\mathbf{u}'_{n_{il}}$ is within a certain distance from \mathbf{u}'_i . Most \mathbf{u}'_i have four neighboring markers ($n_{il} \in N_i, l \in \{0, 1, 2, 3\}$), except that those near the image boundary or those that are cluttered can have different numbers of neighbors. As defined in the following equation, d_i is the displacement between \mathbf{u}'_i and \mathbf{u}_{m_i} , and the displacement of an unmatched marker is calculated by averaging the displacements of its neighboring markers

$$d_i = \begin{cases} \mathbf{u}_{m_i} - \mathbf{u}'_i, & \text{if } m_i > 0 \\ \frac{1}{|N_i|} \sum_{n_{il} \in N_i} \mathbf{d}_{n_{il}} \cdot \mathbb{1}\{m_{n_{il}} > 0\}, & \text{if } m_i = -1 \end{cases} \quad (2)$$

where $\mathbb{1}$ is the indicator function, and $\mathbf{d}_{n_{il}}$ is the displacement of marker i to neighboring marker n_{il} .

We formulate the task as an optimization problem, with the goal to minimize the loss L , which consists of the smoothness term L_{smooth} and the mismatching term L_{mismatch} , where L_{smooth} considers the differences between neighboring marker displacements, inspired by the smoothness objective in optical flow [48], and L_{mismatch} provides penalties for marker mismatch

$$\begin{aligned} L &= L_{\text{smooth}} + L_{\text{mismatch}} \\ L_{\text{smooth}} &= \sum_{i=1}^s \sum_{n_{ij} \in N_i} \|\mathbf{d}_i - \mathbf{d}_{n_{ij}}\| \\ L_{\text{mismatch}} &= \sum_{i=1}^s K_{\text{mismatch}} \cdot \mathbb{1}\{m_i = -1\} \end{aligned}$$

where K_{mismatch} is a hyperparameter to determine the significance of the mismatch objective compared to the smoothness objective.

Minimizing L is a combinatorial optimization problem and is computationally expensive to exhaust all solutions. Therefore, we apply random optimization to heuristically generate several solutions in a limited period, and choose the one with minimal loss. We sequentially assign the corresponding marker for each \mathbf{u}'_i in a stochastic mean. First, we calculate the matching probability p_{ij} for \mathbf{u}'_i and \mathbf{u}_j , which is subsequently normalized and used to sample m_i . To ensure each marker is matched at most once, we set $p_{ij} = 0$ if \mathbf{u}_j is previously matched. The following heuristic is defined to ensure a higher matching probability for closer corresponding markers:

$$p_{ij} = \begin{cases} 1 - \sigma(\|\mathbf{u}'_i - \mathbf{u}_j\| - \mathcal{T}), & \text{if } j > 0 \\ 1 - \sigma(\mathcal{T}), & \text{if } j = -1 \end{cases} \quad (3)$$

where σ is the sigmoid function to map the marker distance to 0-1, and \mathcal{T} is a hyperparameter to determine the marker displacement range.

Based on this formulation, prior to normalization, the matching probability $p_{ij} > 0.5$ when $\|\mathbf{u}'_i - \mathbf{u}_j\| < \mathcal{T}$, and $p_{ij} < 0.5$ otherwise. The condition $j = -1$ represents the situation where no marker is matched. This happens when all \mathbf{u}_j have a distance larger than \mathcal{T} from \mathbf{u}'_i , indicating the corresponding marker is not detected in the current frame. In this work, we selected $\mathcal{T} = 10$ (pixels) based on trial and error. Note that a larger \mathcal{T} allows for \mathbf{u}'_i and \mathbf{u}_j that are further apart to be matched. A larger

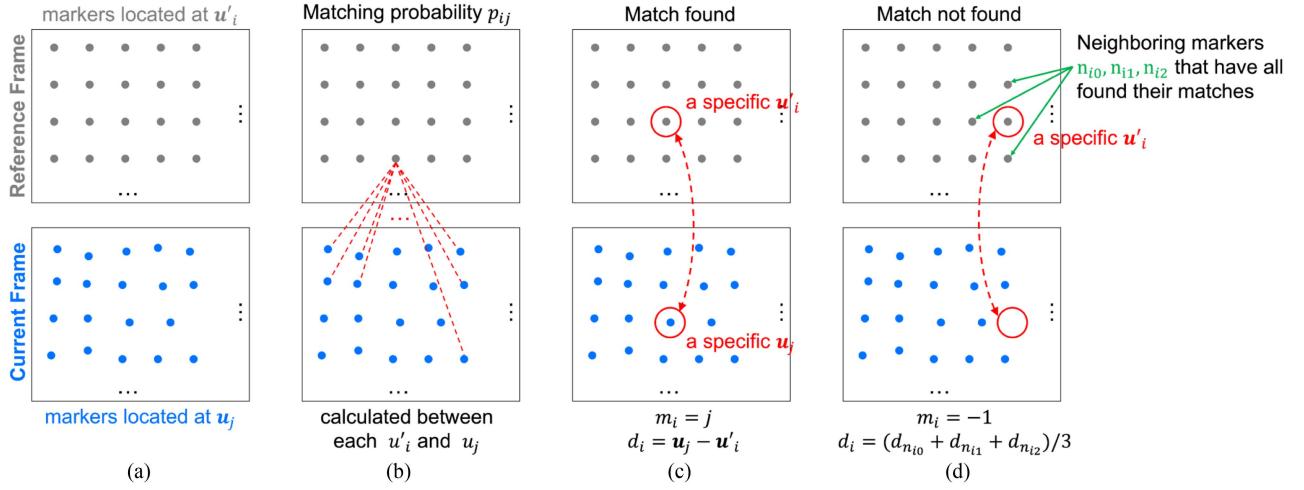


Fig. 7. Illustration of marker displacement calculation. (a) Markers in the reference frame are located at u'_i , and the markers in the current frame are located at u_j . (b) Matching probabilities p_{ij} are calculated for each u_i and u'_j . The matches are selected using weighted random selection. (c) If, for a specific marker i in the reference frame, we find a corresponding marker j in the current frame, we define the corresponding marker index $m_i = j$. The marker displacement is calculated as $d_i = u_j - u'_i$, which is equivalent to $d_i = u_{m_i} - u'_i$. (d) If a specific marker i in the reference frame has no matching marker in the current frame, we define $m_i = -1$. The displacement of this unmatched marker is calculated as the average displacement of its neighboring markers that have matches in the current frame. In this figure, for example, marker i has three neighboring markers n_{i0} , n_{i1} , and n_{i2} . If all three neighboring markers have their matches in the current frame, the displacement of marker i , d_i , is the average displacement of markers n_{i0} , n_{i1} , and n_{i2} : $d_i = (d_{n_{i0}} + d_{n_{i1}} + d_{n_{i2}})/3$.

\mathcal{T} should be selected if we expect the elastomer to experience more shear force, which could occur if the material used is more compliant or if the force applied to the sensor is larger. This heuristic random matching is repeated multiple times to find the minimal loss L .

Unlike exhaustive search, this method is suitable for real-time signal processing. With 200 possible solutions sampled for each frame, the algorithm can achieve real-time marker tracking at the desired frequency of 30 Hz. The number of sampled solutions is empirically determined to provide sufficient tracking accuracy for the tasks performed in this work. Increasing the number of sampled solutions will enhance tracking accuracy but reduce tracking speed. In our case, a reduction in operation speed was not observed until the number of sampled solutions reached 2000.

B. Control Methods for In-Hand Manipulation

We developed a series of manipulation tasks for the TRRG to demonstrate its capabilities. While these demonstrations required various high-level control methods, the low-level joint space control method is consistent across all of them. The base joints used current-limited position control to ensure that the object is being grasped securely without generating excessive internal forces. Position control is used to drive the pivot angle between $\pm 90^\circ$. Smooth rolling motion is achieved through velocity control of the rollers. For each manipulation task presented in this work, we used the tactile sensor to close the control loop. Fig. 8 shows the control diagram. The high-level algorithm can be any heuristic or autonomous planner that outputs the desired object velocity $v_{obj,d}$, which is mapped to the desired pivot angles $\theta_{p,d}$ and roller velocities $\omega_{r,d}$ in the low-level controller. The sensed contact location is sent back to the low-level controller

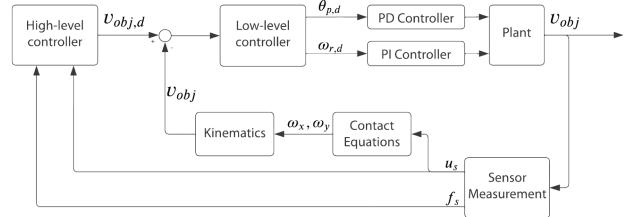


Fig. 8. Control diagram. The heuristic-based high-level controller outputs the desired object velocity, which is mapped to the joint velocities in the low-level controller. Both controllers receive sensor measurements as feedback.

while the high-level controller receives both the contact location and estimated shear force. The surface coordinate information extracted from the sensor measurement is fed through the modified rolling contact equations derived from [49].

The control loop is closed by inferring the object velocity based on the contact coordinates through differential geometry. Consider the shape of the roller shown in Fig. 3(g). We can define the coordinate patch f that maps the open set $U \subset \mathbb{R}^2$ to the roller surface $S \subset \mathbb{R}^3$, i.e., f is a map for $U \rightarrow S \subset \mathbb{R}^3$ for all the points on the surface with coordinate $\mathbf{u} = (u, v) \in U$. The surface S is defined with respect to frames $\mathcal{A}_6/\mathcal{B}_6$. Based on the geometry of the roller and considering the set $U = \{(u, v) | -5\pi/12 < u < -\pi/12, -\pi < v < \pi\}$, the map can be defined as

$$(u, v) \mapsto ((R_0 \cos u - D_0) \cos v - (R_0 \cos u - D_0) \sin v, R_0 \sin u). \quad (4)$$

Note that this formulation can be used for any roller form-factor in the previous Roller Grasps [5], [6]. When $D_0 = 0$, the formulation becomes the mapping for spherical rollers, while

as $u \rightarrow 0$, the formulation becomes the mapping for cylindrical rollers. Based on [49, eqs. (7)–(9)], the curvature form K , the torsion form T , and the metric M become

$$K = \begin{bmatrix} \frac{1}{R_0} & 0 \\ 0 & \frac{\cos u}{R_0 \cos u - D_0} \end{bmatrix} \quad (5)$$

$$T = \begin{bmatrix} 0 & \frac{\sin u}{D_0 - R_0 \cos u} \end{bmatrix} \quad (6)$$

$$M = \begin{bmatrix} R_0 & 0 \\ 0 & R_0 \cos u - D_0 \end{bmatrix}. \quad (7)$$

Assuming a known object geometry around the contact location, an (approximately) point contact between the object and the roller, and rolling without slipping, the rotational velocity of the object can be calculated as

$$\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} = (K + \tilde{K}_{\text{obj}}) M \dot{\mathbf{u}} = (K + \tilde{K}_{\text{obj}}) M (\dot{\mathbf{u}}_s - \dot{\mathbf{u}}_r). \quad (8)$$

We define the contact frame [see Fig. 4(h)] on the roller as right-handed with its origin at the contact point, the z -axis normal to the surface pointing outward, and the x -axis and y -axis pointing in the longitudinal and latitudinal directions of the roller, respectively. ω_x and ω_y are the components of the object's rotational velocity with respect on the contact frame on the roller. $\tilde{K}_{\text{obj}} = R_\psi K_{\text{obj}} R_\psi$ is the curvature of the grasped object relative to the contact frame on the roller, where R_ψ defines the rotation between the contact frame on the roller and that on the object, and K_{obj} is the curvature form of the object. $\dot{\mathbf{u}}_r = [\omega_r, 0]$ is the difference between $\dot{\mathbf{u}}_r$ and $\dot{\mathbf{u}}_s$ caused by the roller speed, where ω_r is the real-time roller speed.

We note that due to deformation of the rollers and unavoidable slippage or creep during rolling, the kinematic relations just derived will have errors. However, as seen in the following section, these errors can be overcome with an appropriate high level control based on tactile information regarding the contact centroid location and velocity.

V. EXPERIMENTS

A. System Setup

The system setup used in the experiments is shown in Fig. 2. The dc motors on TRRG are controlled by a Teensy 4.1 Microcontroller with NXP MC33926 H-bridges. Communication between the Teensy and a computer is facilitated through a bespoke serial communication protocol. The computer hosts four separate processes. The first process manages two separate communication threads to the TRRG, one with the Dynamixel actuators and another with the Teensy. The second process handles communication with the UR-5e robot. The third process is dedicated to sensor image streaming and sensor signal processing. Note that because sensor image streaming and sensor signal processing use separate threads, the update rate of the signal processing results can be as fast as the image streaming rate, provided the processing code is executed faster than the streaming rate. This is why we could achieve a 30 Hz update rate when the input image is streaming at 30 Hz. The last process

TABLE II
MANIPULATION ERRORS: ROLLING VERSUS REGRASPING

Conditions	C1-1	C1-2	C1-3	C1-4
Cylinder (final)	0.16°	6.2°	2.5°	1.8°
hexagonal prism (final)	0.37°	28.7°	5.1°	3.1°
Cylinder (single step)	N/A	0.52°	0.42°	0.45°
hexagonal prism (single step)	N/A	2.4°	0.85°	0.78°

handles all remaining tasks, including the controller and other miscellaneous functions.

B. Rolling Versus Regrasping

We designed experiments to investigate manipulating an object through rolling and through regrasping, with and without the use of tactile sensing. Comparing regrasping and manipulation through rolling with respect to the same task allows us to gain an understanding of the interplay between continuous rolling, breaking and reestablishing contact, and tactile sensing. Regrasping using an external contact was investigated because the TRRG only has two fingers and is not suitable to perform in-hand finger-gaiting. While regrasping is not as common as finger-gaiting, the underlying principle is similar: both involve periodically lifting and replacing the fingers at different locations on the object. However, regrasping requires each finger to be lifted simultaneously from the object while the object rests on a support surface. In finger-gaiting, only a subset of the fingers will be lifted at a given time.

1) *Open-Loop Manipulation*: The first experiment is designed to rotate a grasped object around its vertical axis for 180° using the TRRG and the wrist of a UR-5e. The object is placed on a support surface throughout all experiments and is initially positioned between the two rollers. We tested two objects (a cylinder and a hexagonal prism) under the following four experimental conditions.

C1-1 *Open-loop roll*: The grasper rotates the object by rolling only without tactile feedback.

C1-2 *Regrasp (15° step size)*: The grasper periodically grips the object, rotates the object by rotating its wrist for 15°, then releases the grip and resets to its initial position. The rollers do not operate in this condition.

C1-3 *Regrasp (30° step size)*: Same as C1-2 but with 30° step size.

C1-4 *Regrasp (45° step size)*: Same as C1-2 but with 45° step size.

In each condition, we tracked the orientation of the object throughout the manipulation using the ArUco marker [50] attached at the top of the object.

Fig. 9 and Table II show the results of the experiment. By comparing the final manipulation error, we can see that rolling manipulation results in more accurate outcomes compared to regrasping. For regrasping, error accumulates with each breaking and re-establishing of contact. This is evident from the results in Table II, which show that the more steps involved in regrasping, the larger the final error. We observed no clear relationship

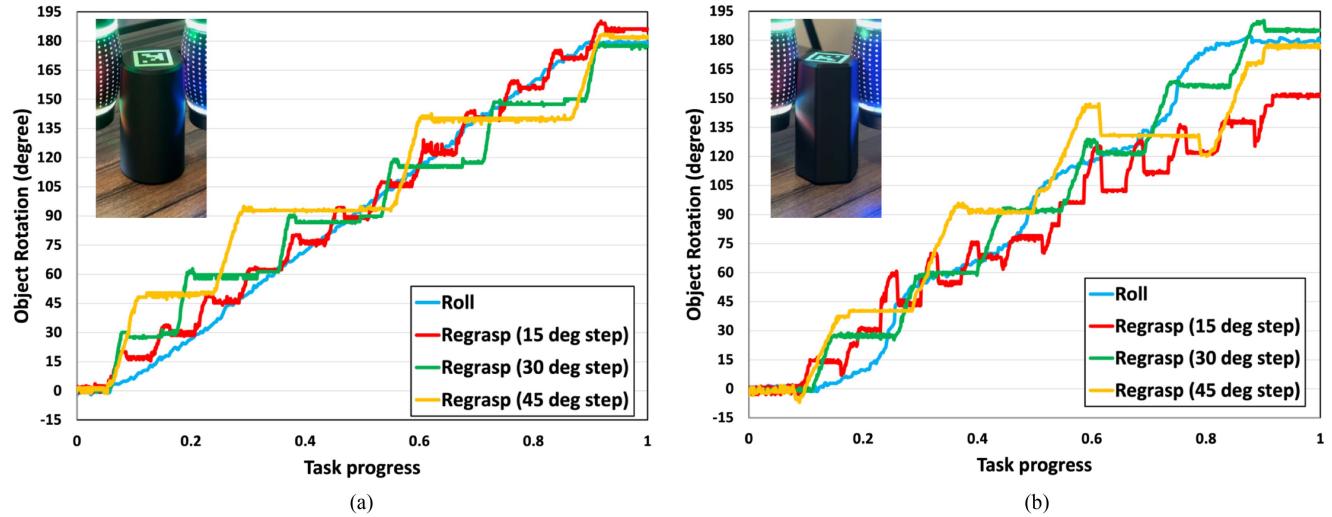


Fig. 9. Orientation tracking of grasped objects undergoing 180° rotation under four different experimental conditions (C1-1 through C1-4in the text) without tactile feedback. Because each condition has a different execution time, we use scaled time (task progress) as the horizontal axis for comparison. (a) Results for a cylinder. (b) Results for a hexagonal prism.

between step size and errors in individual regrasping steps. However, in all regrasping cases, the error accumulated in a single step exceeds the total error observed in rolling conditions. In addition, the orientation tracking in Fig. 9 reveals spikes or dips each time the grasper re-establishes contact, indicating errors occurring during these transient states. We also found that more complex object geometry contributes to higher manipulation errors in both rolling and regrasping situations. However, the impact is significantly greater in the regrasping condition.

2) *Tactile-Guided Manipulation*: The next experiment is designed to understand how contact locations change during manipulation with different methods and how tactile sensing can help correct for that change, thereby stabilizing the manipulation. In this experiment, we rotate a rectangular prism around its vertical axis, similar to the previous experiment but using a different object. We use the tactile sensors to record the contact location between the object and the roller in all of the experimental conditions listed below in order to observe the evolution of the contact location.

We tested the following four experimental conditions.

C2-1 Regrasp open-loop: The grasper periodically grips the object, rotates the object by rotating its wrist for 0.3 radians, then releases the grip and resets to its initial position. The rollers do not operate in this condition.

C2-2 Regrasp closed-loop: In addition to the actions taken in C2-1, while the wrist is rotating, one of the rollers may rotate a small amount based on the sensed contact location to fine-tune the grip pose.

C2-3 Roll open-loop: The grasper rotates the object by rolling continuously without feedback from the tactile sensors.

C2-4 Roll closed-loop: The grasper rotates the object by rolling while simultaneously adjusting the rolling speed based on the contact locations extracted from the tactile sensors.

In each condition, the horizontal contact location in image pixels, v_s , is set to a desired value $v_s = 0$, which represents the contact location being centered on the roller surface from the perspective of the camera.

The results of the experiment can be seen in Fig. 10, which shows the evolution of the horizontal contact location of the object on a roller during manipulation. Top view (a) shows how the horizontal contact location, v_s , can vary when holding a noncircular object. Video recordings of the experiment are provided in the supplementary media. Plots in (b) show variation in v_s under different conditions.

Without tactile feedback the grasper eventually loses control of the object for both the regrasping and the continuous rolling conditions. The results from the regrasp cases (b left) also show that each time the grasper reestablishes contact, the contact location has shifted from its previous position. This uncertainty appears to make the regrasping method less stable compared to continuous rolling when tactile feedback is not used. The open-loop regrasping method (blue dots) loses control of the object after less than 0.25 revolutions. In comparison, the open-loop rolling case (b right, blue dots) is able to rotate the object for 2.5 revolutions.

Regarding the closed-loop experiments, the addition of tactile sensing—plotted as orange dots for both regrasping (b left) and rolling (b right)—allowed both the regrasping and the continuous rolling cases to manipulate the object without losing it. For the pure rolling case, the tactile information is used to continuously monitor and adjust the grasp. However, even in the regrasp case, where the tactile information was only available intermittently, the combination of rolling and tactile sensing allowed the grasper to perform necessary grip adjustments to keep the grasp stable. Thus, whether the object is manipulated by pure rolling or a combination of rolling and regrasping, tactile feedback is beneficial, and possibly necessary, to maintain a stable grasp.

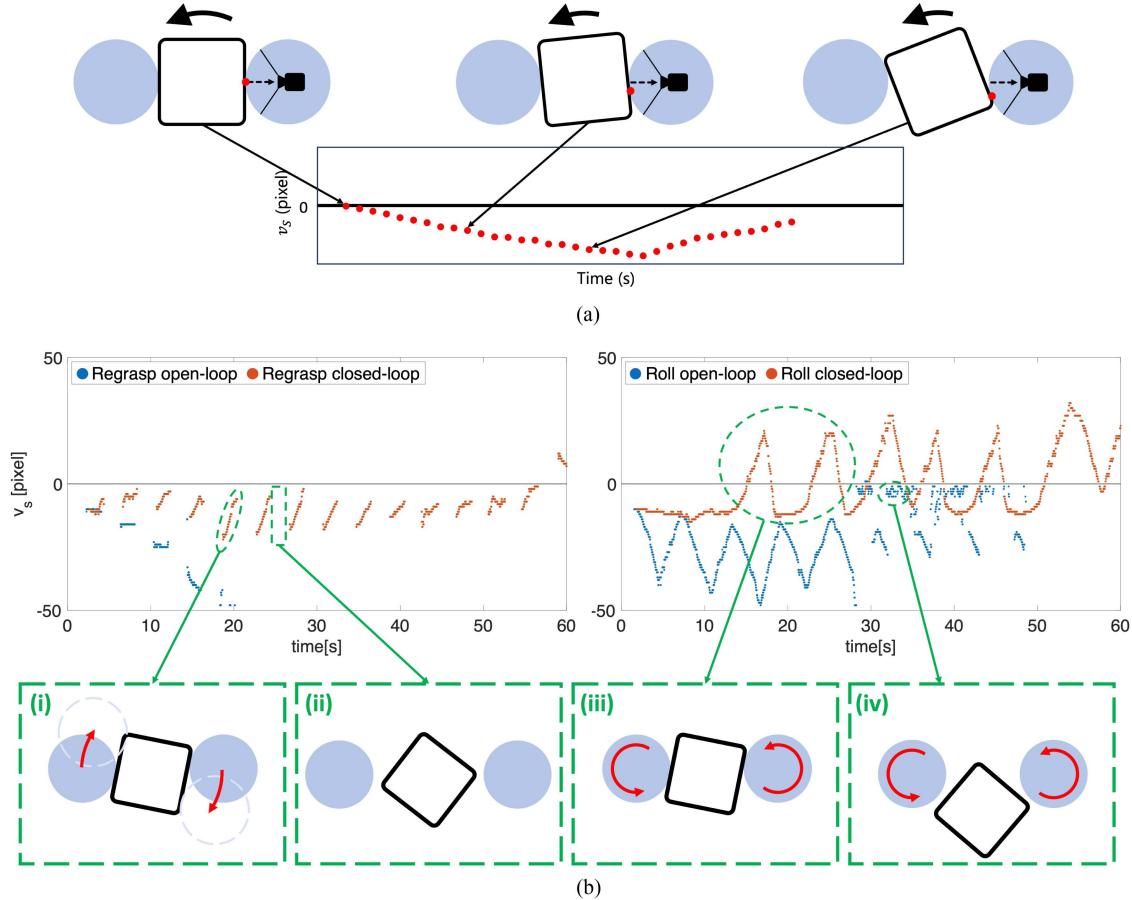


Fig. 10. (a) Top view of rollers grasping an object in different poses, showing how contact location, v_s , varies. The data represent the horizontal coordinate of the contact location recorded by the tactile sensor. (b) Results of experiments comparing regrasping (left) and rolling (right) with and without feedback. In the regrasp closed-loop case (orange dots), v_s is driven toward zero after each new grasp; for the open-loop case, it diverges. For rolling, oscillations arise from the square object cross section, as illustrated in (a). The closed-loop case remains centered around $v_s = 0$ but the open-loop case gradually diverges. Notes: (i) Object manipulated by turning the entire grasper; with contact maintained, system converges to a state of lower error in v_s . (ii) Object is released and grasper resets; a transient occurs and there is no data when there is no contact. (iii) Object is manipulated with rolling; a periodic pattern results from variations in contact locations caused by the noncircular cross section. Since the loop is not closed around the object's position, the variation of the periodic pattern reflects positional drift of the object. (iv) Low-amplitude noise indicates temporary loss of grasp.

Both Figs. 9 and 10 also demonstrate why manipulation through rolling is beneficial. First, because the fingers never lose contact with the object, it eliminates the transient phases associated with breaking or establishing contact, which are common in other in-hand manipulation techniques. These transient phases introduce uncertainties in the system. Second, by maintaining contact with the object, the system operates in a continuous manner, in contrast to the discrete nature of gaiting or regrasping. This continuity facilitates the system's convergence to lower error states.

C. Marker Matching Accuracy

A number of the manipulation demonstrations presented in the following section utilize shear force information extracted from motion analysis, as described in Section IV. To evaluate how the marker matching methods affected tracking results, we designed an experiment to compare the performance of different methods. First, we rotated the rollers for 20 s without any object in contact

TABLE III
MARKER MATCHING RESULTS (AVERAGE OVER EXPERIMENTAL DURATION)

Method	Flow Mean Mag (px)	Flow Mean Ang (°)
Random optimization	0.0104 ± 0.0167	90.03 ± 1.32
Optical flow	0.0103 ± 0.0168	87.63 ± 7.85
Nearest neighbor	0.0139 ± 0.0311	92.56 ± 16.42

and recorded the average marker flow magnitude. Since there is no shear force applied to the roller when no object is in contact, the average marker flow magnitude should remain around zero the entire time. In the next measurement, the TRRG rolled a 3D-printed plate with a 300 g weight attached at the bottom upwards for 20 s, as shown in Fig. 11(c). Because the roller experiences a shear force produced by the weight, the angle of the average marker flow should consistently be 90° throughout the motion.

Fig. 11 shows the data recorded throughout the 20 s duration for the experiments described above, while Table III shows the

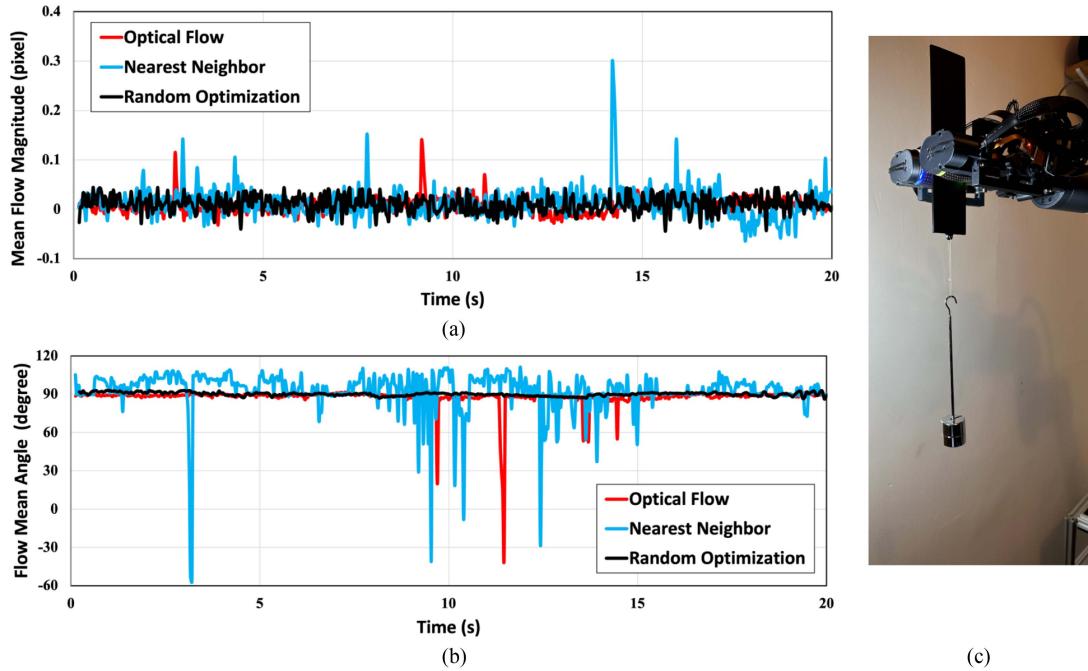


Fig. 11. Comparison of marker tracking accuracy among different methods: random optimization, optical flow, and nearest neighbor/nearest temporal matching. (Mean and standard deviation are presented in Table III). (a) Magnitude of average displacement of the markers from the calibration when no object is in contact. (b) Direction of shear force extracted from the sensor using three different methods. (c) Setup that provides a fixed direction shear force to the rollers, with results plotted in (b).

average values over the experimental duration. As shown in Fig. 11 and Table III, random optimization has the most stable tracking and lowest noise of all three methods. It results in more accurate outcomes compared to optical flow and nearest neighbor in both measurements. While optical flow is comparable to random optimization in flow mean measurement in terms of noise, it occasionally produces a few spikes [see Fig. 11(a)] and performs significantly worse when shear force is present [see Fig. 11(b)]. Among the three methods, nearest neighbor performs the poorest, exhibiting the highest noise and large spikes throughout both measurements. Comparing Fig. 11(a) and (b), we can also see that the benefit of random optimization becomes more significant when a shear force is applied to the roller. In this case, the marker displacements become larger compared to the situation when the roller is not in contact with the object, and the tracking becomes more challenging for optical flow and nearest neighbor.

D. Task Demonstrations

In this section, we describe several manipulation tasks to demonstrate the capabilities of the TRRG. Videos of these tasks are included in the supplementary media.

1) *Cylindrical Object Rotation*: As shown in Fig. 12(a), the controller adjusts the rolling speed of the fingertips to keep the pen rotating at a specified speed. We found that the controller is robust even in the presence of external disturbances. In addition, we can extract the primary and secondary principal axes of the contact area using principal coordinate analysis [9], which, for an object with a relatively consistent contact shape indicate the

orientation of the contact geometry, and subsequently the pose of the grasped object. In this particular example, the principal axis indicates the long axis of the pen.

2) *Planar Object Reorientation*: The planar object reorientation [see Fig. 12(b)] attempts to keep the flat object spinning around a horizontal axis. The planar objects used in this demonstration are 3-D printed with varying radii of curvature to demonstrate that the control method can adapt to complex and unknown 2-D profiles. Tactile sensing was used to adjust the pivot angle of the rollers to stably hold the object during the manipulation. Each of the previous two examples were also run in open-loop without the sensor feedback, and the grasped objects were dropped shortly after the experiment began.

3) *Spherical Object Screw Motion and Trajectory Following*: Unlike the previous two demonstrations with a fixed control target, this task [see Fig. 12(c)] attempts to move the object along predefined trajectories in the operational space. Object position is computed using forward kinematics, and used to close the high-level trajectory following control loop. The screw motions achieved in this demonstration, where the translational and rotational motions are coupled, are difficult or impossible to perform with traditional robot hands. The TRRG easily achieves screw motions by setting the rollers in opposite orientations, forming a cross. Changing the angle between the rollers enables setting the screw pitch from zero to infinity, and anywhere in between.

4) *Cable Tracing*: Unlike stiff objects, cables can withstand substantial tensile load but can buckle under modest compressive loads. This makes a cable a traditionally difficult object for robot manipulation. Tracing along the length of a cable can be achieved

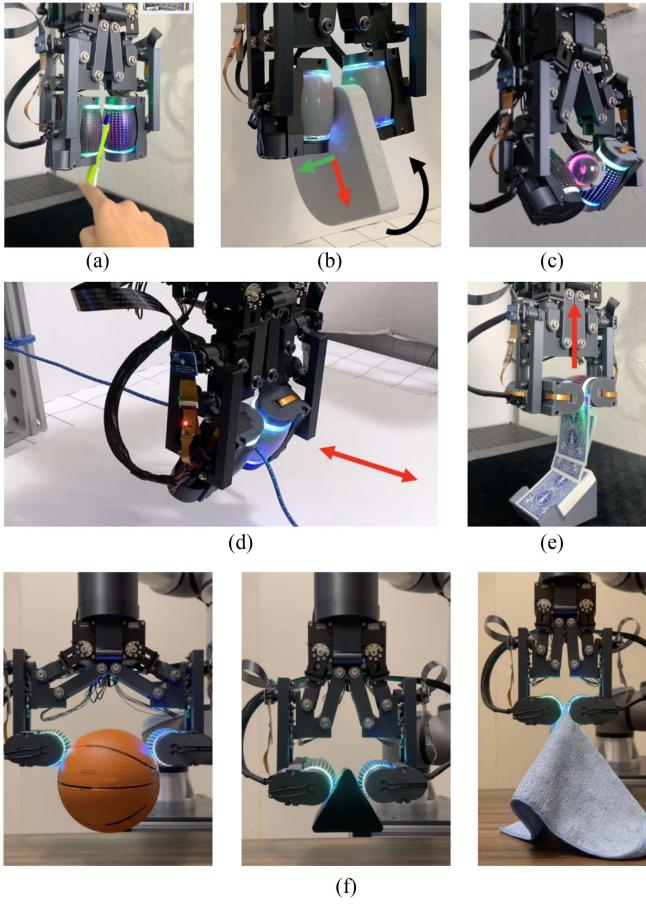


Fig. 12. In-hand manipulation and grasping demonstrations. (Detailed demonstrations are provided in the supplementary video.) (a) Cylindrical object manipulation. (b) Planar object manipulation. (c) Spherical object manipulation. (d) Cable manipulation: The roller grasper can stably traverse bidirectionally along a cable. Closed-loop control for the sensed shear force and the contact location maintain the cable tension and compensate for disturbances from gravitational forces. Without controlling the shear force the cable accumulates slack. (e) Card picking: The TRRG picks up a single card by actuating a single roller and monitoring transient shear force signals. These signals help distinguish between multicard and single-card grasps. (f) Grasping of various objects including a 6-in ball, a prism, and a piece of cloth.

while sliding away from where the cable is anchored in order to maintain the cable tension, as was shown in [9]; moving toward the anchored location, which will cause the cable to buckle due to friction. In our case, the combination of the rolling motion and the ability to control the estimated shear force applied to the cable makes it possible for the rollers to trace along a cable both toward and away from the anchor point while maintaining the cable tension [see Fig. 12(d)]. The contact location can also be computed to prevent the cable from dropping due to gravitational forces. By adjusting the pivot direction the contact location can be kept in the center of the roller. We tested two open-loop cases for this demonstration. The first case did not use any sensor feedback, which resulted in the cable being dropped almost immediately due to gravity. The second case tracked the contact location but ignored the shear information. The rollers were able to move along the cable for longer; however, the cable eventually became slack and was no longer traversable. Using

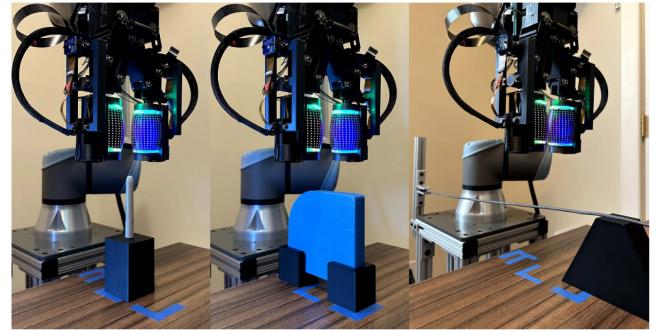


Fig. 13. MTTF measurement. Each object is grasped from a fixed location determined by a fixture. From left to right: a cylindrical object, a planar object with a varying radius of curvature, and a cable.

both the contact location and shear force information, the cable could be traversed without fail. The average positional error with respect to the equator of the roller was 12.8 mm with no feedback, 3.7 mm with contact position feedback, and 3.3 mm with both forms of feedback. Note that although the latter two situations have fairly close positional error, without the shear force feedback, the cable eventually became slack. The slackness of the cable does not reflect in the positional error until the cable actually drops to a lower position.

5) Card Picking: Another way of using the tactile information is to capture the details of transient events—something commonly done by humans in manipulation tasks. In particular, transient events can be useful when interfacing with thin objects, like playing cards. Their extreme aspect ratios increase the likelihood for several cards to stick together; therefore, distinguishing between multicard and single-card grasps can be extremely difficult. The TRRG can circumvent this by using a single roller to actuate one side of a given card while monitoring the observed shear force [see Fig. 12(e)]. When multiple cards are within the hand, relative motion of any two cards reduces the experienced shear force on the roller. However, sudden changes in the shear force indicate that only one card is left in grasp. This is a specific example demonstrating the use of transient shear information, but in practice there are various situations where this approach can be used (e.g., [51], [52], [53]), especially when the state of the hand-object configuration experiences sudden changes.

6) Grasping: Fig. 12(f) as well as the supplementary video demonstrated a few examples of TRRG grasping objects of various geometries, sizes, and compliance. While optimizing for picking up small objects was not a primary objective during the design of TRRG, it is capable of grasping small or thin objects using rolling contact such as picking up cards or a piece of cloth.

7) MTTF for Different Tasks: We compared the MTTF for the tasks described in Sections V-D1, V-D2, and V-D4. In each case, the object was grasped at a specific location determined by a fixture, as shown in Fig. 13. Once the object was grasped, the fixture was removed before the manipulation tasks began. The cylindrical object and the planar object were manipulated under both open-loop and closed-loop conditions, and the cable tracing tasks had three different conditions as described in Section V-D4. Each condition was repeated five times. We

TABLE IV
MTTF OF MANIPULATION TASKS

Task	Open loop (s)	Contact location feedback (s)	Contact location & shear force feedback (s)
Cylindrical object manipulation	67.0 ± 13.54	180	N/A*
Planar object manipulation	7.5 ± 1.11	180	N/A*
Cable tracing	6.1 ± 0.97	141.2 ± 48.09	180

* N/A indicates that no shear force feedback was used in the manipulation task.

recorded the time from the beginning of the task until failure, defined as the loss of grasp. Each task also had a three-minute cutoff time to prevent the manipulation from being carried out indefinitely. The MTTF for the tasks we tested is shown in Table IV. The closed-loop situation significantly outperformed the open-loop cases in manipulations of the cylindrical object and the planar object. For the cable tracing tasks, adding contact location feedback drastically improved the manipulation results, and adding shear force feedback improved them even further.

8) *Discussions:* We developed a series of demonstrations in order to highlight different aspects of the TRRG system. With prior knowledge that manipulation through rolling can provide good dexterity, these demonstrations focus on how the integration of tactile sensing can provide the system with additional capabilities and robustness.

In terms of kinematics, the two-finger design with six total actuated DoF for the TRRG is a significant simplification compared to the previous three-finger Roller Grasper v1, v2 [5], [6]. However, the inclusion of tactile sensing allows the TRRG to stably manipulate objects without the extra redundancies necessary for grasp stability in previous generations. Even with the reduced DoF, the TRRG is capable of translating or rotating the grasped object in each of the X_O , Y_O , and Z_O directions, as defined in Fig. 4(a). By utilizing the combinations of the various manipulation primitives presented in Fig. 4(b)–(f), grasped objects can be manipulated between a wide variety of initial and target poses.

The tactile sensor provides both depth and shear information for an object in contact with the rollers, and the raw sensor data can be further processed to extract higher level information suited for in-hand manipulation. The contact location can be used to close a low-level control loop for determining each finger's rolling speed and direction [see Fig. 12(a)–(c)]. The depth information along with the shear information can also be used to inform the high-level control loop in order for the hand to perform more comprehensive tasks robustly [see Fig. 12(d) and (e)].

The tactile sensor also helps mitigate certain hardware limitations. Specifically, since the TRRG acts as a parallel jaw grasper, it needs to apply sufficient normal forces to stably grasp objects. To ensure that the rollers are still able to roll even in the presence of these grasp forces, we calculated the necessary output torque to the roller based on a highly conservative rolling resistance coefficient of 0.4 (comparable to a wheeled vehicle in soft sand [54]) and the normal force of 68.3 N that the roller can apply to an object when fully closed. With the current design, the required gear ratio for the roller is 331:1, making this DoF mechanically nonbackdrivable. However, the tactile sensor enables force control along the shear direction through active servoing.

Depending on task requirements, the rollers can either not react to the shear force [as shown in Fig. 14(i)], taking advantage of the friction of the transmission for secure grasping, or actively adjust the roller speed based on the shear information [as shown in Fig. 14(ii) and (iii)], allowing for compliant manipulation despite having a nonbackdrivable transmission.

E. Object/Image Reconstruction Using Steerable Rollers

The rolling action combined with tactile sensing also permits efficient surface inspection or reconstruction, compared with previous methods where researchers needed to apply a sequence of discrete touches when inspecting surface roughness [55], detecting defects [56], [57], or reconstructing 3-D shapes [58], [59], [60].

We demonstrated surface scanning ability through the reconstruction of surface geometries for both a credit card (2-D scanning) and a transparent cup (3-D scanning). In the 2-D demonstration, the grasper guided the credit card in between its rollers while scanning the surface texture Fig. 15(a)–(c). The tactile images were then stacked in the time sequence to recover the credit card numbers. We also show the raw stacked images in comparison with the processed images to better visualize the effects of filters and the interpolation required to fill the areas occluded by the black markers.

The 3-D scanning demonstration reconstructed a transparent cup with an embossed logo [see Fig. 15(d)–(f)]. In comparison to [35], the reduced form-factor of our rollers allow us to reconstruct more complex geometries. The cup was mounted on a turntable to enable pure rotation about the vertical axis. The rollers held onto the opposing sides of the cup near its opening, and traversed from the lip to the base. The rollers then tilted a small angle (5°) and rolled upwards until they reached the opening of the cup at adjacent positions to where they started. The slightly tilted angles of the rollers resulted in a screw motion of the cup relative to the grasper: every time the rollers rolled down and up, the cup was rotated by an angle chosen to be sufficiently small such that the entirety of the surface is scanned. To better align the scanned images, we applied cross-correlation to match their correspondence. The resulting images are presented in point clouds in Fig. 15(d)–(f). We chose a transparent cup, which can be challenging to reconstruct using external cameras, to highlight the capabilities of this technique.

The rolling capability of TRRG naturally complements the sensor's scanning abilities for several reasons. First, as mentioned above, it enables more efficient scanning compared to discrete touches due to its continuous rolling, as lifting and repositioning sensors on the object consumes time. Second, it allows

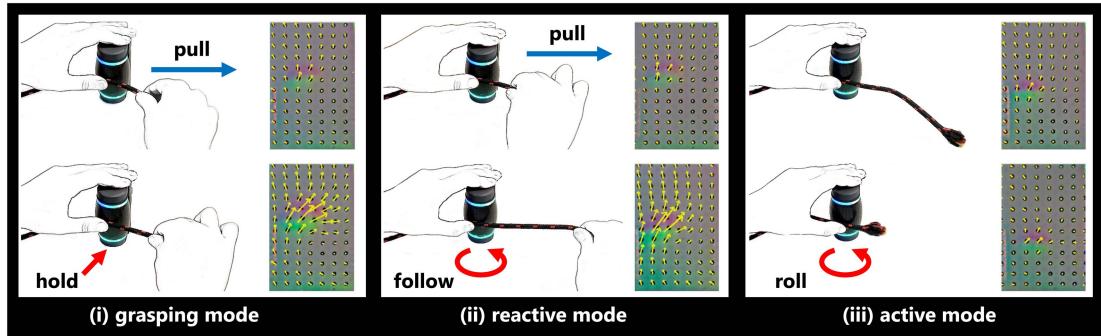


Fig. 14. Different roller actuation modes. (i) Grasping mode: the roller is nonbackdrivable, and will resist external force due to friction. (ii) Reactive mode (software backdriving): The roller can react according to the amount of shear force applied on the roller surface. (iii) Active mode: the roller actively applies motions to the object.

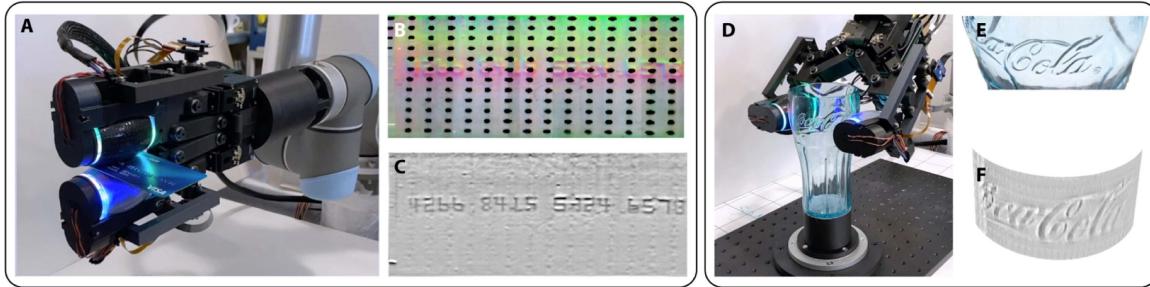


Fig. 15. Surface scanning. (Detailed demonstrations are provided in the supplementary video.) (a) Rolling along a credit card. (b) Stacked tactile images in the time sequence, showing the embossed numbers on the credit card. (c) Processed image with interpolation at the marker region and sharpening filters for better visualization. (d) Rolling along a transparent cup. (e) Embossed characters on the cup. (f) Scanned tactile images stitched in 3-D spaces.

for more accurate manipulation than regrasping or finger gaiting, facilitating image stitching with minimal computation during manipulation. This contrasts with most previous works [55], [56], [57], [58], [59], [60], where sensor images were taken when the object was at a fixed location. The benefits of rolling versus regrasping are discussed in detail in Section V-B. Finally, the rolling method is nonabrasive since it avoids sliding during scanning, enhancing the sensor’s robustness [39], as compared to the methods presented in [40] and [41].

VI. CONCLUSION

A. Contributions

This work investigates how in-hand manipulation can be augmented by use of the tactile sensing in conjunction with rolling contacts. To this end, this article presents the design of the TRRG that has steerable active rollers at fingertips integrated with high-resolution tactile sensors. We developed algorithms to process tactile signals in order to provide real-time feedback for in-hand manipulation and object reconstruction. A generic low-level controller was formulated based on differential geometry to use the tactile information to perform in-hand object manipulation. The approach is suitable for a range of form factors that would support rolling contact, including the previous Roller Graspers. We also demonstrated more comprehensive tasks by closing a higher level control loop using only the

tactile information. The mechanical and algorithmic design of the TRRG demonstrate an ability to perform robust in-hand manipulation for various objects through tactile-guided rolling contact, even with uncertain dynamics. We also demonstrated its unique capability to perform efficient surface inspection and reconstruction of surface geometries during manipulation. This was made possible with the combination of actively driven rolling contact and high-resolution tactile information. In addition to the various demonstrations, we conducted an experiment to investigate manipulation through rolling versus regrasping, including effect of using tactile sensing as feedback. We found that when tactile feedback was used to perform fine grip adjustments, both manipulation through rolling and manipulation through regrasping were more stable and could continue longer without failure. This suggests that even devices such as anthropomorphic hands constrained to pure finger-gaiting could benefit from the inclusion of roller-based fingertips to finely adjust the grasp without the need for lifting and re-placing the fingers on the object.

In summary, we presented the abilities of the TRRG for in-hand manipulation and object reconstruction, as well as its potential to complete complex perception and manipulation tasks in various real-world robotic settings. We hope this work will push the boundaries in both robotic manipulation and tactile sensing, and inform design decisions for future work in tactile-guided rolling manipulation.

B. Limitations and Future Work

While we have demonstrated the abilities of the TRRG, there are different aspects of this work that can be further explored.

In terms of the design of the grasper, although the rollers on the TRRG have a convex curvature, a spherically shaped roller allows for more consistent contact behavior. As noted in [5], when rollers with large radii of curvature are contacting objects with large radii of curvature the contact point can shift unpredictably in the face of minor misalignment.

Another limitation of the current design is that the size of the sensing area is restricted by the 90° camera field of view as well as the mirror size and shape, which could be increased by using a camera with large FOV (e.g., a fish-eye camera) and a convex mirror. In addition, exploring a 360° or omnidirectional tactile sensor design [61], [62], [63], [64] for the fingertips presents a promising opportunity to enhance manipulation capabilities. However, implementing such a design, particularly with cantilevered pivotable rolling fingertips, introduces significant mechanical complexities. Careful consideration is needed to balance the tradeoffs between system capabilities and design complexity.

Our demonstrations presented feedback control methods using only tactile information, however, inclusion of additional sensing modalities, such as force sensing and visual feedback could provide both global object information as well as local contact information.

The TRRG's control pipeline could be augmented through integrating its object geometry reconstruction and in-hand manipulation abilities: while the TRRG can manipulate objects with unknown geometry and dynamics, the geometry of the object reconstructed during manipulation can further be used to improve the manipulation results.

Investigation of how TRRG manipulates various types of soft/compliant objects can be another promising direction. The study of contact dynamics with soft objects, as well as how different parameters of the controller (such as grasp force) affect the sensor performance and manipulation results, will provide additional insights regarding rolling manipulation.

Lastly, while maintaining contact during manipulation demonstrates many benefits, there are certain situations when lifting the finger is desired during manipulation. For example, when the object has sharp protrusions or deep concave features, manipulation through rolling may no longer be an optimal solution. A more comprehensive controller that accounts for such extreme cases would increase the diversity of the objects being manipulated and further improve the robustness of manipulation.

ACKNOWLEDGMENT

The authors would like to thank R. Thomasson for her assistance in clarifying the figures.

REFERENCES

- [1] I. M. Bullock and A. M. Dollar, "Classifying human manipulation behavior," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, 2011, pp. 1–6.
- [2] S. R. Company, "Shadow dexterous hand," (n.d.). [Online]. Available: <https://www.shadowrobot.com/dexterous-hand-series/>
- [3] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Proc. Conf. Robot Learn.*, 2022, pp. 297–307.
- [4] M. Bjelonic et al., "Keep rollin'—whole-body motion control and planning for wheeled quadrupedal robots," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 2116–2123, Apr. 2019.
- [5] S. Yuan, A. D. Epps, J. B. Nowak, and J. K. Salisbury, "Design of a roller-based dexterous hand for object grasping and within-hand manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 8870–8876.
- [6] S. Yuan, L. Shao, C. L. Yako, A. Gruebele, and J. K. Salisbury, "Design and control of roller grasper v2 for in-hand manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 9151–9158.
- [7] H. Yousef, M. Boukallel, and K. Althoefer, "Tactile sensing for dexterous in-hand manipulation in robotics—A review," *Sens. Actuators A: Phys.*, vol. 167, no. 2, pp. 171–187, 2011.
- [8] N. Fazeli, M. Oller, J. Wu, Z. Wu, J. B. Tenenbaum, and A. Rodriguez, "See, feel, act: Hierarchical learning for complex manipulation skills with multisensory fusion," *Sci. Robot.*, vol. 4, no. 26, 2019, Art. no. eaav3123.
- [9] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *Int. J. Robot. Res.*, vol. 40, no. 12–14, pp. 1385–1401, 2021. [Online]. Available: <https://doi.org/10.1177/02783649211027233>
- [10] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.
- [11] C. Piazza, G. Grioli, M. Catalano, and A. Bicchi, "A century of robotic hands," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 2, pp. 1–32, 2019.
- [12] M. A. Diftler et al., "Robonaut 2—the first humanoid robot in space," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 2178–2183.
- [13] M. Grebenstein et al., "The DLR hand ARM system," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3175–3182.
- [14] J. Salisbury, "Kinematics and force analysis of articulated hands," Ph.D. thesis, Mech. Eng., Stanford Univ., Stanford, CA, 1982.
- [15] S. C. Jacobsen, J. E. Wood, D. Knutti, and K. B. Biggers, "The UTAH/M.I.T. dextrous hand: Work in progress," *Int. J. Robot. Res.*, vol. 3, no. 4, pp. 21–50, 1984.
- [16] R. R. Ma and A. M. Dollar, "An underactuated hand for efficient finger-gaiting-based dexterous manipulation," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2014, pp. 2214–2219.
- [17] W. G. Bircher, A. S. Morgan, K. Hang, and A. M. Dollar, "Energy gradient-based graphs for planning within-hand caging manipulation," in *Proc. Int. Conf. Robot. Autom.*, 2019, pp. 2462–2467.
- [18] R. R. Ma, N. Rojas, and A. M. Dollar, "Spherical hands: Toward underactuated, in-hand manipulation invariant to object size and grasp location," *J. Mechanisms Robot.*, vol. 8, no. 6, 2016, Art. no. 061021.
- [19] C. M. McCann and A. M. Dollar, "Design of a stewart platform-inspired dexterous hand for 6-DoF within-hand manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1158–1163.
- [20] N. Govindan and A. Thondiyath, "Design and analysis of a multi-modal grasper having shape conformity and within-hand manipulation with adjustable contact forces," *J. Mechanisms Robot.*, vol. 11, no. 5, p. 051012, 2019.
- [21] V. Tincani et al., "Velvet fingers: A dexterous gripper with active surfaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 1257–1263.
- [22] R. R. Ma and A. M. Dollar, "In-hand manipulation primitives for a minimal, underactuated gripper with active surfaces," in *Proc. Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, 2016, vol. 50152, Art. no. V05AT07A072.
- [23] A. Kakogawa, H. Nishimura, and S. Ma, "Underactuated modular finger with pull-in mechanism for a robotic gripper," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2016, pp. 556–561.
- [24] P. Datsaris and W. Palm, "Principles on the development of mechanical hands which can manipulate objects by means of active control," *J. Mech. Des.*, vol. 107, no. 2, pp. 148–156, 1985. [Online]. Available: <https://doi.org/10.1115/1.3258703>
- [25] B. Ward-Cherrier et al., "The tactip family: Soft optical tactile sensors with 3D-printed biomimetic morphologies," *Soft Robot.*, vol. 5, no. 2, pp. 216–227, 2018.
- [26] A. Alspach, K. Hashimoto, N. Kuppuswamy, and R. Tedrake, "Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation," in *Proc. 2nd IEEE Int. Conf. Soft Robot. RoboSoft*, 2019, pp. 597–604.
- [27] C. Sferrazza, T. Bi, and R. D'Andrea, "Learning the sense of touch in simulation: A sim-to-real strategy for vision-based tactile sensing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 4389–4396.

- [28] A. Yamaguchi and C. G. Atkeson, "Combining finger vision and optical tactile sensing: Reducing and handling errors while cutting vegetables," in *Proc. IEEE-RAS 16th Int. Conf. Humanoid Robots (Humanoids)*, 2016, pp. 1045–1051.
- [29] M. Lambeta et al., "DIGIT: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 3838–3845, Jul. 2020.
- [30] A. Padmanabha, F. Ebert, S. Tian, R. Calandra, C. Finn, and S. Levine, "Omnitact: A multi-directional high resolution touch sensor," 2020, arXiv: 2003.06965.
- [31] B. Romero, F. Veiga, and E. Adelson, "Soft, round, high resolution tactile fingertip sensors for dexterous robotic manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 4796–4802.
- [32] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing—From humans to humanoids," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 1–20, Feb. 2010.
- [33] M. R. Cutkosky and W. Provancher, "Force and Tactile Sensing," in *Springer Handbook of Robotics*. Berlin, Germany: Springer, 2016, pp. 717–736.
- [34] K. Shimonomura, "Tactile image sensors employing camera: A review," *Sensors*, vol. 19, no. 18, p. 3933, 2019.
- [35] G. Cao, J. Jiang, C. Lu, D. F. Gomes, and S. Luo, "Touchroller: A rolling optical tactile sensor for rapid assessment of large surfaces," 2021, arXiv: 2103.00595.
- [36] W. K. Do, B. Aumann, C. Chungyoun, and M. Kennedy, "Inter-finger small object manipulation with densetact optical tactile sensor," *IEEE Robot. Autom. Lett.*, vol. 9, no. 1, pp. 515–522, Jan. 2024.
- [37] X. Zhang, J. Jiang, and S. Luo, "Rotip: A finger-shaped tactile sensor with active rotation," in *Proc. Workshop Robot Embodiment Through Visuo-Tactile Percep.*, 2024.
- [38] S. Wang, Y. She, B. Romero, and E. Adelson, "Gelsight wedge: Measuring high-resolution 3D contact geometry with a compact robot finger," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 6468–6475.
- [39] W. G. Peck, *Elements of Mechanics: For the Use of Colleges, Academies, and High Schools*. New York, NY, USA: AS Barnes, 1872.
- [40] C. Wang, S. Wang, B. Romero, F. Veiga, and E. Adelson, "Swingbot: Learning physical features from in-hand tactile exploration for dynamic swing-up manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5633–5640.
- [41] N. Sunil, S. Wang, Y. She, E. Adelson, and A. R. Garcia, "Visuotactile affordances for cloth manipulation with local control," in *Proc. Conf. Robot Learn.*, 2023, pp. 1596–1606.
- [42] E. Donlon, S. Dong, M. Liu, J. Li, E. Adelson, and A. Rodriguez, "GelSlim: A high-resolution, compact, robust, and calibrated tactile-sensing finger," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1927–1934.
- [43] R. Szeliski, *Computer Vision: Algorithms and Applications*. Berlin, Germany: Springer, 2010.
- [44] G. Bradski, "The opencv library," *Dobb's J. Softw. Tools Professional Programm.*, vol. 25, no. 11, pp. 120–123, 2000.
- [45] J. Doerner, "Fast poisson reconstruction in Python," 2014. [Online]. Available: <https://gist.github.com/jackdoerner/b9b5e62a4c3893c76e4c>
- [46] C. Sferrazza and R. D'Andrea, "Design, motivation and evaluation of a full-resolution optical tactile sensor," *Sensors*, vol. 19, no. 4, p. 928, 2019.
- [47] G. Zhang, Y. Du, H. Yu, and M. Y. Wang, "DeTact: A vision-based tactile sensor using a dense color pattern," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10778–10785, Oct. 2022.
- [48] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, no. 1–3, pp. 185–203, 1981.
- [49] D. J. Montana, "The kinematics of contact and grasp," *Int. J. Robot. Res.*, vol. 7, no. 3, pp. 17–32, 1988.
- [50] F. J. Romero-Ramirez and R. Muñoz-Salinas, "Rafael medina-carnicer, image and vision computing," vol. 76, pp. 38–47, 2018.
- [51] F. Veiga, H. Van Hoof, J. Peters, and T. Hermans, "Stabilizing novel objects by learning to predict tactile slip," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 5065–5072.
- [52] W. Chen, H. Khamis, I. Birznieks, N. F. Lepora, and S. J. Redmond, "Tactile sensors for friction estimation and incipient slip detection—Toward dexterous robotic manipulation: A. review," *IEEE Sensors J.*, vol. 18, no. 22, pp. 9049–9064, Nov. 2018.
- [53] N. Kuppuswamy, A. Alspach, A. Uttamchandani, S. Creasey, T. Ikeda, and R. Tedrake, "Soft-bubble grippers for robust and perceptive manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 9917–9924.
- [54] Tire friction and rolling resistance coefficients, (n.d.). [Online]. Available: <https://hpwizard.com/tire-friction-coefficient.html>
- [55] A. Amini, J. I. Lipton, and D. Rus, "Uncertainty aware texture classification and mapping using soft tactile sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 4249–4256.
- [56] B. Fang, X. Long, F. Sun, H. Liu, S. Zhang, and C. Fang, "Tactile-based fabric defect detection using convolutional neural network with attention mechanism," *IEEE Trans. Instrum. Meas.*, vol. 71, 2022, Art. no. 5011309.
- [57] J. Jiang, G. Cao, D. F. Gomes, and S. Luo, "Vision-guided active tactile perception for crack detection and reconstruction," in *Proc. 29th Mediterranean Conf. Control Autom.*, 2021, pp. 930–936.
- [58] M. Björkman, Y. Bekiroglu, V. Högnan, and D. Kragic, "Enhancing visual perception of shape through tactile glances," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 3180–3186.
- [59] S. Wang et al., "3D shape perception from monocular vision, touch, and shape priors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1606–1613.
- [60] E. Smith et al., "3D shape reconstruction from vision and touch," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 14193–14206, 2020.
- [61] D. F. Gomes, Z. Lin, and S. Luo, "GelTip: A finger-shaped optical tactile sensor for robotic manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 9903–9909.
- [62] O. Azulay et al., "AllSight: A low-cost and high-resolution round tactile sensor with zero-shot learning capability," *IEEE Robot. Autom. Lett.*, vol. 9, no. 1, pp. 483–490, Jan. 2024.
- [63] I. Andrussov, H. Sun, K. J. Kuchenbecker, and G. Martius, "Mindsight: A fingertip-sized vision-based tactile sensor for robotic manipulation," *Adv. Intell. Syst.*, vol. 5, no. 8, 2023, Art. no. 2300042.
- [64] M. H. Tippur and E. H. Adelson, "Gelsight360: An omnidirectional camera-based tactile sensor for dexterous robotic manipulation," in *Proc. IEEE Int. Conf. Soft Robot. (RoboSoft)*, 2023, pp. 1–8.



Shenli Yuan received the dual B.S. degrees in mechanical engineering from Purdue University, West Lafayette, IN, USA, and Shanghai Jiao Tong University, Shanghai, China, in 2015, the M.S. degree in mechanical engineering, the M.A. degree in music science and technology, and the Ph.D. degree in mechanical engineering with a minor in computer science from Stanford University, Stanford, CA, USA, in 2018, 2019, and 2022, respectively.

He is currently a Research Scientist with the Boston Dynamics AI Institute. Prior to this role, he was a Senior Research Engineer at SRI International.



Shaoxiong Wang received the B.E. degree in computer science from Tsinghua University, Beijing, China, in 2017 and the M.S. and Ph.D. degrees in computer science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2018 and 2022, respectively.

He is currently a Senior Robotics Engineer with Dexterity, Inc., Redwood City, CA, USA. His interests include robotic manipulation, tactile sensing, and multimodal robot learning.



Radhen Patel received the B.Tech. degree in electrical engineering from the Sardar Vallabhbhai National Institute of Technology, Surat, India, in 2015 and the M.S. and Ph.D. degrees in computer science from the University of Colorado at Boulder, Boulder, CO, USA, in 2019.

He is currently with BotBuilt, Durham, NC, USA, as a Robotics Software Engineer. His interest lies in the development of intelligent robotic systems that seamlessly bridge the gap between the virtual and physical realms.



Megha Tippur received the B.S. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2020 and the M.S. degree in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2022. She is currently working toward the Ph.D. degree in electrical engineering and computer science with Massachusetts Institute of Technology.

Her research interest include dexterous manipulation and tactile sensing.



Connor L. Yako received the B.S. degree in mechanical engineering from the University of Michigan, Ann Arbor, MI, USA, and the M.S. degree in mechanical engineering in 2018 and 2020, respectively from Stanford University, Stanford, CA, USA, where he is currently working toward the Ph.D. degree in mechanical engineering.

His research is focused on nonanthropomorphic methods for performing in-hand manipulation, such as a gripper that changes its finger dimensions as well as asymmetric vibrations.



Mark R. Cutkosky (Fellow, IEEE) received the Ph.D. degree in mechanical engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 1985.

He is currently the Fletcher Jones Professor in Mechanical Engineering with Stanford University, Stanford, CA, USA. His research interests include bioinspired robots, haptics, and rapid prototyping processes.

Dr. Cutkosky is a Fellow of ASME, and was the recipient of the IEEE-RAS Pioneer in Robotics and Automation Award.



Edward Adelson (Life Fellow, IEEE) received the B.A. degree in physics and philosophy from Yale University, in 1974, and the Ph.D. degree in experimental psychology from the University of Michigan, in 1979. He is currently the John and Dorothy Wilson Professor of Vision Science with the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA, USA, and a member of the Computer Science and Artificial Intelligence Laboratory (CSAIL). He is a member of the National Academy of Sciences. He has authored or coauthored

widely on topics in human vision, computer vision, computer graphics, artificial tactile sensing, and robotics. He is well known for contributions to multiscale image representation (such as the Laplacian pyramid) and basic concepts in early vision such as steerable filters and motion energy models. He introduced the plenoptic function, and built the first plenoptic camera. He has produced some well-known illusions such as the Checker-Shadow Illusion. He is currently work on artificial touch sensing for robotics.

Mr. Adelson was the recipient of the Nakayama Medal of the Vision Sciences Society, two test-of-time awards from the IEEE Computer Society, and the IEEE Lifetime Achievement Award in Computer Vision.



J. Kenneth Salisbury (Life Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical and mechanical engineering from Stanford University, Stanford, CA, USA, in 1975, 1977, and 1982, respectively.

Since then he has held Research Professorships with Massachusetts Institute of Technology, Cambridge, MA, USA, and Stanford University (Stanford), Stanford, CA, USA. During a four-year sabbatical with Intuitive Surgical, he helped develop the da Vinci surgical robot. He has been associated with many fundamental contributions in robotics and haptics, including the PHANToM Haptic Interface, the MIT WAM Arm, and the Willow Garage PR-2. He is currently a Professor Emeritus with Computer Science and Surgery Department, Stanford, conducting research in in-hand manipulation, physical Human/Robot Interaction (pHRI) (in a robotic EMT), and patient-specific simulation of skull-based surgical procedures.

Dr. Salisbury was a recipient of the IEEE-RAS Pioneer in Robotics and Automation Award.