

Article

Towards Haptic-Based Dual-Arm Manipulation

Sri Harsha Turlapati  and Domenico Campolo * 

School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798, Singapore

* Correspondence: d.campolo@ntu.edu.sg

Abstract: Vision is the main component of current robotics systems that is used for manipulating objects. However, solely relying on vision for hand–object pose tracking faces challenges such as occlusions and objects moving out of view during robotic manipulation. In this work, we show that object kinematics can be inferred from local haptic feedback at the robot–object contact points, combined with robot kinematics information given an initial vision estimate of the object pose. A planar, dual-arm, teleoperated robotic setup was built to manipulate an object with hands shaped like circular discs. The robot hands were built with rubber cladding to allow for rolling contact without slipping. During stable grasping by the dual arm robot, under quasi-static conditions, the surface of the robot hand and object at the contact interface is defined by local geometric constraints. This allows one to define a relation between object orientation and robot hand orientation. With rolling contact, the displacement of the contact point on the object surface and the hand surface must be equal and opposite. This information, coupled with robot kinematics, allows one to compute the displacement of the object from its initial location. The mathematical formulation of the geometric constraints between robot hand and object is detailed. This is followed by the methodology in acquiring data from experiments to compute object kinematics. The sensors used in the experiments, along with calibration procedures, are presented before computing the object kinematics from recorded haptic feedback. Results comparing object kinematics obtained purely from vision and from haptics are presented to validate our method, along with the future ideas for perception via haptic manipulation.

Keywords: object pose tracking; haptic; active manipulation; dual-arm manipulation; object localization; object manipulation; haptic manipulation



Citation: Turlapati, S.H.; Campolo, D. Towards Haptic-Based Dual-Arm Manipulation. *Sensors* **2023**, *23*, 376. <https://doi.org/10.3390/s23010376>

Academic Editor: Aiguo Song

Received: 20 November 2022

Revised: 19 December 2022

Accepted: 20 December 2022

Published: 29 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Approaches to Dexterous Robotic Manipulation

Dexterous manipulation of an object is the ability to change the object's position and orientation [1] while grasping it in the hand—an ability that has let humans use tools and arguably develop a superior brain even [2]. Robotics manipulation research has attempted to mimic this human ability by building robot hands with the aim of achieving similar dexterity artificially for both industrial [3] and home [4–6] applications.

Despite impressive results in robotic manipulation showcased for warehouse operations [7], robots still fall short of human-like levels of speed and reliability. Most robot solutions use only one robotic hand with limited gripper orientation [8]. This greatly limits speed, manipulative payload and dexterity, as humans can use two arms.

With dual-arm manipulation, state estimation and control are determined by the contact type between the robot hands and object, i.e., (i) point contact (with and without friction), (ii) sliding contact and (iii) rolling contact [9,10]. With most practical situations having friction between hand and object, point contacts are problematic if the objective is to move the object dexterously without break of contact. For this reason, rolling manipulation increases the dexterity with the hand boundary, allowing for dexterous motion of the object [2,11].

The forward kinematics of rolling-contact dual-arm manipulation was rigorously investigated in the late 1980s [12,13] using geometry. The rolling constraints were used

to derive the relation between robot velocities and object velocities, and a joint torque controller was developed to simulate the contact forces due to rolling contact. Going a step further to kinematic constraints, a dynamical closed loop controller was developed in [14–16]—where the only knowledge needed was sensing of finger–robot states and kinematic parameters in a finger model. This approach was called *blind grasping* and has spawned more research in recent years [17], even with tackling challenges such as unknown object mass, shape, Coriolis terms and hand/object kinematics [18]. Although it is desirable to perform manipulation with as little information as possible as reasoned in [15], it is advantageous in practical situations to choose to use the information which is readily available—robot hand geometry, robot parameters, etc. For instance, contact-point sensing using hand geometry is useful in determining object pose during manipulation in real time, as we will show in this work. Discarding this haptic information could be disadvantageous and would need alternative approaches to manipulation control. In fact, much of the robotic manipulation literature focuses on using vision to determine object pose information, as we discuss next.

Much work regarding vision-based robotic grasping and manipulation for parallel jaw grippers mounted on a single robot arm has been reported in the surveys by [8,19,20]. Such vision-based approaches present fundamental limitations when it comes to dexterous manipulation tasks requiring more accurate and controlled contact interactions, such as object reorientation, object insertion or almost any kind of object use [21]. Many practical applications, especially in home settings, would require robots to have *semi-precise* or *gentle* placement of components, which would require contact information about the object pose and haptic feedback during manipulation. This further motivates the case for using haptic feedback during robot manipulation for perception and control.

Contributions and Organization of Paper

In this work, we propose a simple approach using the concept of the wrench axis, which allows for contact sensing. Tactile sensors [22] are usually placed on the robot's end-effector tips, which enable sensing the point of contact effectively. In this work, we chose to place the force sensors on the robot wrists instead. This choice allows for a measurement of a wrench, which indirectly contains contact point information when viewed with robot hand geometry (which is information that should be available to every robotics engineer). We present a method to estimate online the object pose using (i) an initial kinematic estimate obtained from vision and (ii) real-time haptic feedback from the hands in a dual-arm robotic manipulation setting. We also show that we can reconstruct the normal and tangential forces at contact using the geometry at the contact frame.

The rest of the paper is organized as follows. In Section 2, we present a haptic method to determine contact frame using the wrench axis (estimated using force/torque measurements alone) for a circular robot hand. This haptic method is one of the main contributions of this work. To validate the contact frame information obtained, we demonstrate the computation of object orientation under rolling contact (without slippage) by tracking the contact point on the robot hand. With promising results for estimating kinematics from haptic feedback for a single hand, we present the forward kinematics for dual-arm robotic manipulation in Section 3. The second contribution of our work is the algorithm to compute object position and orientation at any time using (i) information about initial hand–object kinematics and (ii) hand kinematics and haptic feedback at any time. This is presented in Section 4 along with the method to fuse kinematic estimates from two hands. The experimental validation of the proposed algorithm is presented in Section 5, which reports on a human operator performing a haptic demonstration of dual-arm manipulation using a teleoperated robot setup. The object position and orientation were tracked using vision (with a high accuracy motion capture system) and also computed using our proposed algorithm, and the results are compared to validate our method. Finally, the conclusions are presented along with sensor calibration reported in Appendix A.1 and numerical details of the noise present in sensed wrenches in Appendix A.2.

2. Haptic Estimation of Point of Contact in Circular Hands

The basic idea behind our work is to use haptic information from the robot to infer (at least in part) environmental kinematics, such as object position and the orientation of the object being manipulated. In the event that this information is already available, e.g., intermittently from cameras, it is beneficial to fusing such data streams. Although this could be seen as merely algebraic changing of variables, we found the geometric approaches such as those based on wrench axis [23,24] especially insightful for manipulation purposes.

Many contact tasks of interest involve only forces (not moments) at the point of contact. A moment at the point of contact might arise, for example, from twisting the robotic finger/hand about the normal axis of contact, in the presence of dry friction, but we shall exclude this type of manipulation. Moreover, this eventually can only appear in 3D manipulation while this paper focuses on 2D, planar scenarios. However, moments will be generated at frames located away from the contact point. Typically, force/torque sensing units (loadcells) are located at the wrist of the robot, i.e., away from the end-tip where the contact with the environment occurs. Given a force f at the finger-tip, the moment is determined as:

$$\tau = d \times f \quad (1)$$

where d denotes the displacement vector between the point of contact and the origin of the loadcell frame. Therefore, while the force reading from the loadcell provides directly the components of the applied force, the torque readings provide additional information on where the force is applied. Specifically, given force and torque readings f and τ , the point of application lies on the axis (denoted ‘wrench-axis’) characterized, in 3D space, by the linear equations in (1). In 2D space, the cross product has no meaning, and the wrench axis equation simply becomes:

$$\tau = \det([d \ f]) \quad (2)$$

where $[d \ f]$ is the 2×2 matrix with columns $d = [d_x \ d_y]^T$ and $f = [f_x \ f_y]^T$ (see Figure 1a).

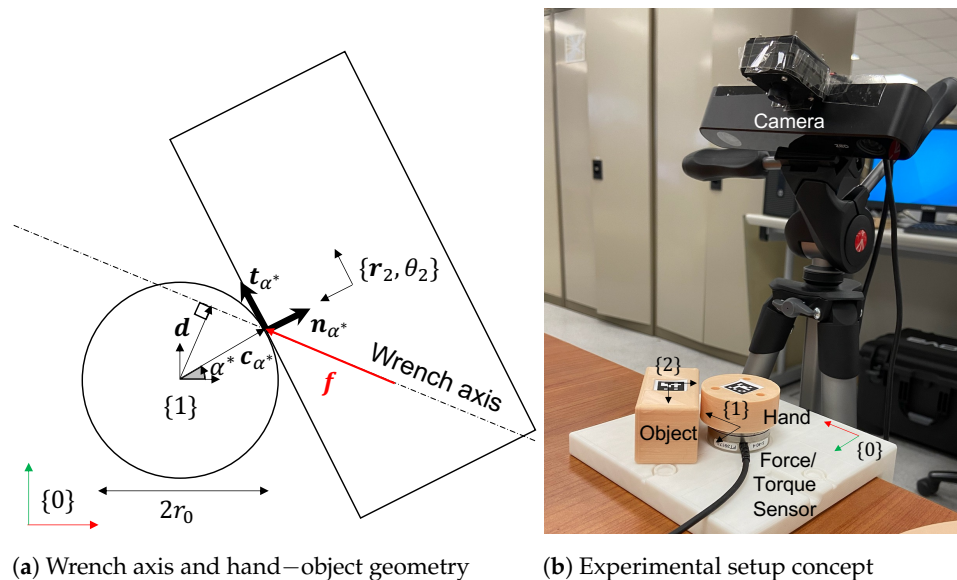


Figure 1. (a) Any point on the wrench axis has the same displacement vector d from frame 1. Using Equation (4), we determine c_{α^*} , the point of pushing. (b) This experimental setup was designed to test the proposed method of estimating the point of contact for circular hands.

Given the geometry of the object on which a force is being applied, the geometric intersection with the wrench axis ultimately defines the point of contact. In summary, haptic measurements from the robot hand/fingers can be used to infer the point of contact.

Additional information such as the geometry of the robotic hand will allow one to determine further variables, such as the normal and tangent vectors at contact, as shown next.

2.1. Contact Frame Estimation from Wrench Axis

In the 2D case, the boundary of a solid object is simply a 2D curve. The boundary of a circular hand can be conveniently parameterized by contact parameter α . Any point on the boundary, c_α is defined as a function of α and hand radius r_0 . With reference to Figure 1a, when a force is acting at a point c_α , the wrench axis equation, Equation (2), can also be expressed as:

$$\tau = \det([c_\alpha \ f]) = r_0(\cos(\alpha)f_y - \sin(\alpha)f_x) \quad (3)$$

Expanding this, we next estimate the parameter α^* obtained from haptic feedback in circular hands as:

$$\alpha^*(f, \tau) := -\arccos\left(\frac{\tau}{r_0\sqrt{f_x^2 + f_y^2}}\right) + \operatorname{atan2}(-f_x, f_y) \quad (4)$$

After the estimation of α^* , the contact point and frame may be computed as:

$$c_{\alpha^*} = r_0 n_{\alpha^*}, \quad n_\alpha = \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} \quad t_\alpha = \begin{bmatrix} -\sin(\alpha) \\ \cos(\alpha) \end{bmatrix} \quad (5)$$

where n_{α^*} and t_{α^*} represent the normal and tangent at the contact point c_{α^*} .

2.2. Experimental Validation of Estimated Point of Contact

With reference to Figure 1a, given the point of contact at c_{α^*} obtained using α^* defined above, in our simplified case, we observe that the tangent t_{α^*} is aligned with the frame $\{2\}$, $t_{\alpha^*} \parallel n(\theta_2)$, i.e.,

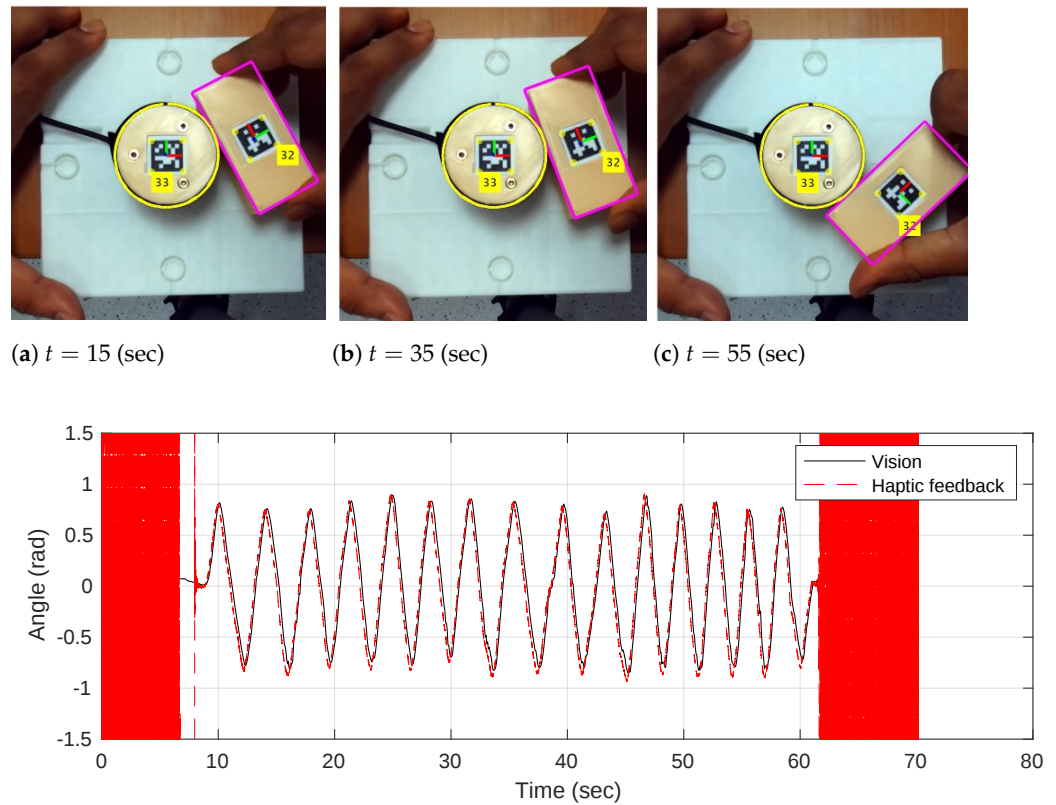
$$\theta_2 = \operatorname{atan2}(t_x, t_y) \quad (6)$$

To validate that the computation of $\alpha^*(f, \tau)$ is accurate, we conducted an experimental trial where a rectangular object was rolled along the circular hand's boundary without slipping manually, as shown in Figure 2. The object orientation θ_2 was obtained from two sources:

- Vision: using Apriltags, as shown in Figure 2.
- Haptics: using the proposed haptic method $\alpha^*(f, \tau)$ in Equation (4).

Note that in Figure 2, during the initial and final phases where there is no contact, the object orientation, as estimated from haptic feedback, is not well defined. Once rolling without slipping is ensured, the haptic-estimated object orientation follows the visual estimate very closely.

With these initial results, we proceed to define the larger problem of using a dual-arm robotic system with circular hands to determine the object pose, i.e., both position and orientation during dexterous manipulation of a planar object in the next section.



(d) The object orientation, as estimated by haptic feedback, closely matches the vision estimate, when the object and hand are in contact.

Figure 2. (a–c) Hand–object configurations during the experiments along with annotated Apriltags for computing the visual estimate of hand–object relative orientation. (d) Rolling contact without slipping was ensured by rubber cladding of the hand. Hand–object configurations were varied manually. Object orientation θ_2 estimated from Apriltags, i.e., vision, plotted against the orientation computed by haptic feedback using Equations (4) and (6).

3. Kinematics of Planar Dual-Arm Manipulator

The planar kinematics (see Figure 3) of the object and the dual-arm robot can be described in terms of rigid planar transformations to and from the following frames:

- Common frame $\{0\}$, a fixed frame attached to the work-space (e.g., the table on which the robot is operating);
- Left-hand frame $\{1\}$, a moving frame attached to the end-effector of robot arm 1;
- Right-hand frame $\{2\}$, a moving frame attached to the end-effector of robot arm 2;
- Object frame $\{3\}$, a moving frame attached to the object to be manipulated.

For each frame, we define its location $\mathbf{r} \in \mathbb{R}^2$ and its orientation θ w.r.t the common frame $\{0\}$, to be composed as a 3-tuple $\{\mathbf{r}, \theta\}$. To compose the transformation of the local coordinates for, say, robot arm 1 at time t , the associated $SE(2)$ transformation denoted by $\Phi_{1t} = [\mathbf{r}_1, \theta_1]_{SE2}|_t$ is defined as:

$$\Phi_{1t} = \begin{bmatrix} \mathbf{R}(\theta_1(t)) & \mathbf{r}_1(t) \\ \mathbf{0} & 1 \end{bmatrix} \quad \mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (7)$$

Similar definitions may be made for the other frames at different times.

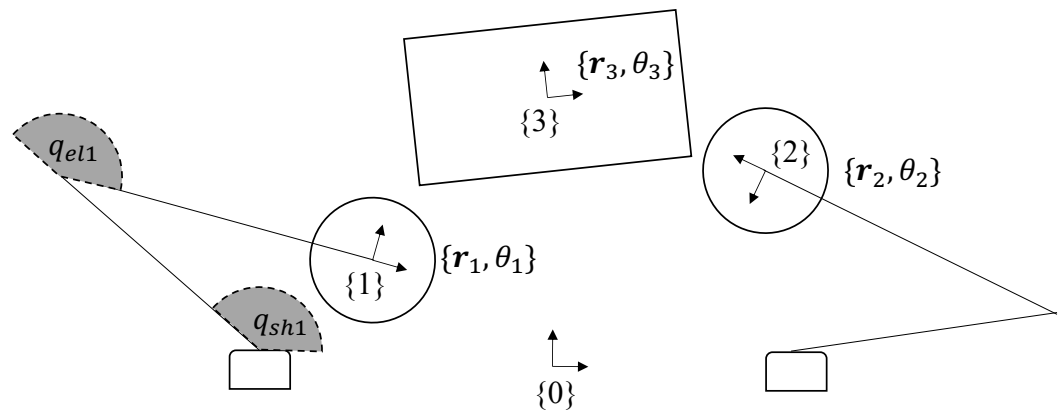


Figure 3. Dual-arm robot kinematic variables.

The task space coordinates of the end-effector frames defined above are related to the joint variables by the robot's forward kinematics as:

$$r_1 = \begin{bmatrix} l_1 \cos(q_{sh1}) + l_2 \cos(q_{sh1} + q_{el1}) \\ l_1 \sin(q_{sh1}) + l_2 \sin(q_{sh1} + q_{el1}) \end{bmatrix} \quad \theta_1 = q_{sh1} + q_{el1} \quad (8)$$

Similar computations were made for the second robot arm.

Hand–Object Surface Parameterization

As defined before in Section 2.1, any point on the robot hand $h = \{1, 2\}$ with parameter α_h and radius r_0 (see Figure 4) can be expressed in local-coordinates as:

$$c(\alpha_h) = r_0 n_{\alpha_h} \quad (9)$$

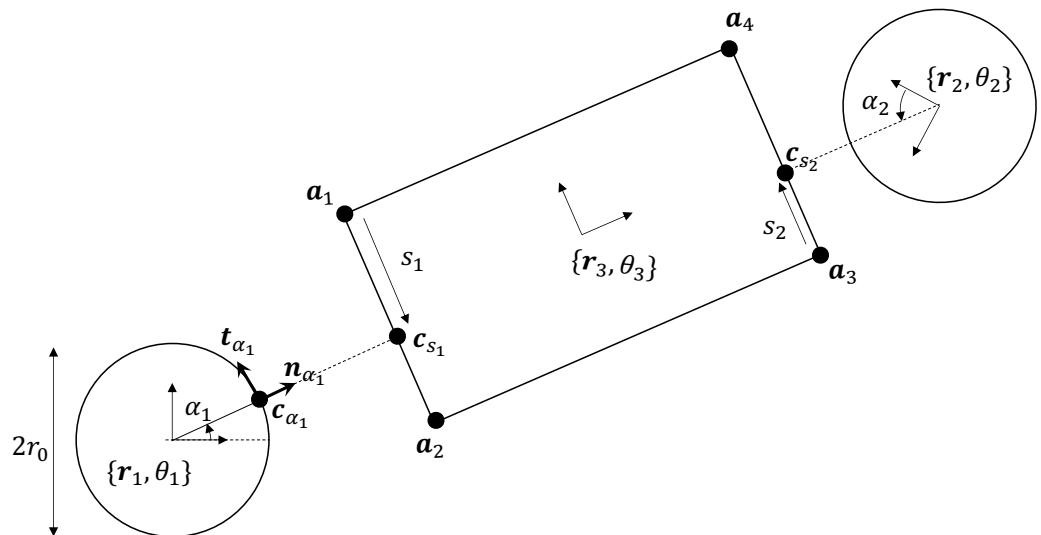


Figure 4. Parameterized surface proxies [25] are virtual points defined as being constrained to be on the surface of objects. By introducing attractive dynamics between these proxies, they can be made to act as the closest points (to the robot hand) on their respective surfaces. In the context of physically interacting objects, the kinematics of such proxies on both surfaces would capture the contact constraints. For instance, with rolling contact between side $a_1 a_2$ and hand 1, the proxies c_{α_1} and c_{s_1} on two rolling bodies would be coincident with no relative velocity. Similarly, the proxies for hand 2 would follow. We define the notation such that, for each point c_{α_h} on hand h , the closest point on the object is defined as c_{s_h} .

Note that n_{α_h} and t_{α_h} represent the normal and tangent at the point parameterized by α_h expressed in local coordinates. We next define the parameter s_1 for any point on the side a_1a_2 of the object expressible as:

$$c(s_1) = a_1 + s_1 \widehat{a_{12}}, \quad a_{12} = a_2 - a_1 \quad \widehat{a} = \frac{a}{\|a\|} \quad (10)$$

Similar definitions can be made for robot hand 2 interacting with side a_3a_4 . These definitions will become important when we discuss the rolling contact constraints between the robot hands and their respective object sides being contacted during manipulation, in the next section.

4. Haptic-Based Tracking of Object Pose

Under the assumptions of rolling with no slipping defined in the previous subsection, we studied two hand–object configurations obtained via rolling contact, as shown in Figure 5a,b. The initial hand–object configuration is defined by the local geometry at the point of contact, which is parameterized by $\alpha_{h0} = \alpha^*(t_0)$ with the normal and tangent in local coordinates also defined as per Equation (5). Similarly, the final configuration was parameterized by α_{ht} . The decomposition of relative motion from the initial to final configuration is shown in Figure 5a,b. The following algorithm summarizes these steps to calculate final object pose given the initial pose and wrench information at any time t .

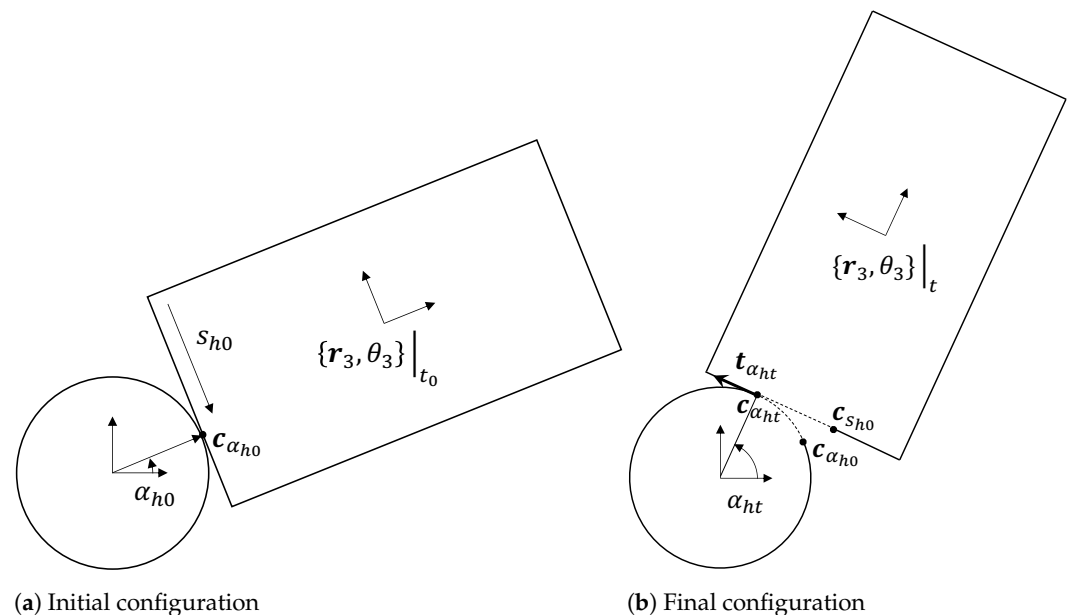


Figure 5. (a) Definition of initial hand–object configuration parameterized by $\alpha_{h0} = \alpha^*(t_0)$. (b) At time t , the object position and orientation can be obtained from $\Delta\alpha_h = \alpha_{ht} - \alpha_{h0}$. This follows from the rolling without slipping constraint that the distance traversed between $c_{\alpha_{h0}}$ and $c_{\alpha_{ht}}$ (along the hand boundary) should be equal and opposite to that traversed on the object boundary, i.e., from $c_{s_{h0}}$ to $c_{s_{ht}}$.

It follows that the displacement of the point of contact on the robot hand is parameterized by $\Delta\alpha_h = (\alpha_{ht} - \alpha_{h0})$ and is given by $r_0\Delta\alpha_h$, for this case of circular hands. We can also infer that the displacement of point of contact on the object is equal and opposite—i.e., a positive value of $\Delta\alpha_h$ will imply a negative value of Δs_h . In other words, since the object has straight sides, we have $\|c_{s_{ht}} - c_{s_{h0}}\| = \|r_0\Delta\alpha_h\|$ (see Figure 5b). In summary, with the hand and object always in rolling contact without slipping or break of contact, at any instantaneous point in time, the relative movement of the object w.r.t. the hand is composed of (i) a pure rotation about the robot hand and (ii) a pure translation about the tangent at the contact point.

Sensor Fusion of Object Pose from Multiple Robot Arms

Assuming that Gaussian white noise is present in the force sensors, we can compute the propagation of uncertainty through Algorithm 1 to obtain $\tilde{r}_3, \Sigma_{r_3}$, i.e., the object position estimate and associated uncertainty, respectively, and $\tilde{\theta}_3, \sigma_{\theta_3}^2$, i.e., the object orientation estimate and associated uncertainty, respectively. We define the covariance matrix for the 2D wrenches Σ_{W_h} (see Appendix A.2 for details) sensed on hand h (in local coordinates) as:

$$\Sigma_{W_h} = \begin{bmatrix} \Sigma_f & \sigma_{f,\tau_h}^2 \\ \sigma_{f,\tau_h}^2 & \sigma_{\tau_h}^2 \end{bmatrix}, \quad W_h = \begin{bmatrix} f'_h \\ \tau'_h \end{bmatrix} \quad (11)$$

Note that Σ_{W_h} is a symmetric matrix obtained from sensor data and denotes the sensitivity of the force/torque sensor's strain gauges; i.e., the more sensitive or accurate they are, the lower the determinant of Σ_{W_h} .

Algorithm 1 Computing final configuration from (i) knowledge of initial configuration and (ii) wrench in final configuration using information from hand h .

Require: $\alpha_{h0}, [r_3, \theta_3]|_{t_0}, [r_h, \theta_h]|_{t_0,t}$ and $[f'_h, \tau'_h]|_t$ $\triangleright$ (') denotes local frame measurements

$\alpha_{ht} \leftarrow \alpha^*([f'_h, \tau'_h]|_t)$

$\Delta\alpha_h \leftarrow \alpha_{ht} - \alpha_{h0}$

$Rot \leftarrow \begin{bmatrix} R(\Delta\alpha_h) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}$ $\triangleright$ Rotation about hand

$Trn \leftarrow \begin{bmatrix} \mathbb{I}_2 & -r_0\Delta\alpha_h \mathbf{t}'(\alpha_{ht}) \\ \mathbf{0} & 1 \end{bmatrix}$ $\triangleright$ Back-shifting along tangent

$\Phi_{30} \leftarrow [r_3, \theta_3]_{SE2}|_{t_0}, \Phi_{h0} \leftarrow [r_h, \theta_h]_{SE2}|_{t_0}$ $\triangleright$ Initial state

$\Phi'_{30} \leftarrow \Phi_{h0}^{-1} \cdot \Phi_{30}$

$\Phi'_{3t} \leftarrow Trn \cdot Rot \cdot \Phi'_{30}$ $\triangleright$ Final state

$\Phi_{ht} \leftarrow [r_h, \theta_h]_{SE2}|_t$

$\Phi_{3t} \leftarrow \Phi_{ht} \cdot \Phi'_{3t}$

return $[r_3, \theta_3]|_t$

Recall that the object position and orientation $[r_3, \theta_3]|_t$ at time t are the outputs of Algorithm 1 with the input parameters $\alpha_{h0}, [r_3, \theta_3]|_{t_0}, [r_h, \theta_h]|_{t_0,t}$ and $[f'_h, \tau'_h]|_t$. To distinguish the noisy object pose information coming from each hand h , we re-define the outputs of the algorithm at time t as $[r_{3,h}, \theta_{3,h}]|_t$. To obtain these estimates, we define the Jacobian of our algorithm as:

$$J_{W_h} = \nabla_{W_h} \begin{bmatrix} r_{3,h} \\ \theta_{3,h} \end{bmatrix} \quad (12)$$

We decompose the Jacobian J_{W_h} to determine the covariances Σ_{r_3} and $\sigma_{\theta_3}^2$ estimated by one hand h , defined as follows:

$$\Sigma_{r_{3,h}} = J_{r_3} \Sigma_{W_h} J_{r_3}^T \quad \sigma_{\theta_{3,h}}^2 = J_{\theta_3} \Sigma_{W_h} J_{\theta_3}^T \quad J_{W_h} = \begin{bmatrix} J_{r_3} \\ J_{\theta_3} \end{bmatrix}_{3 \times 3} \quad (13)$$

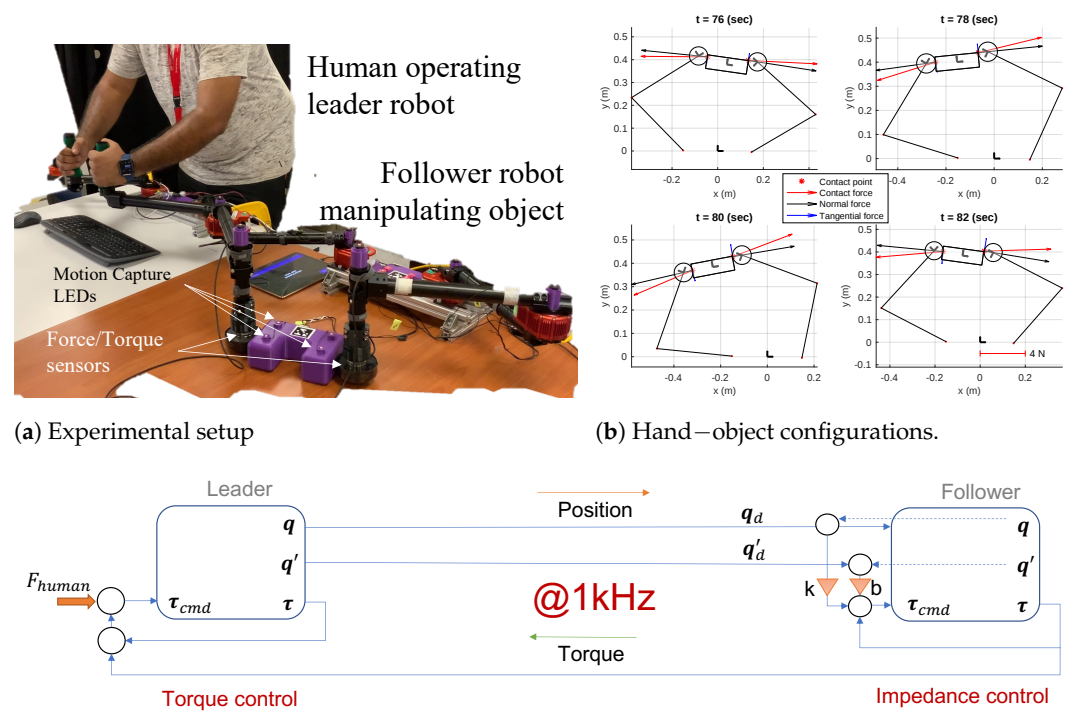
Given multiple hands, one can fuse these probability distributions from each hand h assuming independence, by the product of Gaussians [26]:

$$\Sigma_{r_3} = \left(\Sigma_{r_{3,1}}^{-1} + \Sigma_{r_{3,2}}^{-1} \right)^{-1} \quad \tilde{r}_3 = \Sigma_{r_3} \left(\Sigma_{r_{3,1}}^{-1} r_{3,1} + \Sigma_{r_{3,2}}^{-1} r_{3,2} \right) \quad (14)$$

Similar equations follow to estimate the object orientation as well. We now have all the tools required to compute the estimate for object position and orientation from two hands and their kinematic and haptic feedback. In the next section, we validate our method with our experimentation.

5. Object-Pose Estimation Results

To test the proposed approach of estimating object pose in planar dual-arm manipulation, a robotic test-bed was built, as shown in Figure 6a. Since the scope of this work does not include grasp synthesis and planning, we used human haptic demonstrations of dexterous manipulation performed using a teleoperated dual-arm robot. This choice was in pursuit of a larger goal to understand human haptic strategies in dual-arm manipulation, although it is beyond the scope of this work. A key advantage of our approach to estimating the contact frame is the possibility of reconstructing the normal and tangential forces at contact (see Figure 6b). This adds on to previous research [12] that heuristically set normal forces to control the object pose in simulation. With our approach, we provide an experimental way to generate data from human haptic demonstrations of these internal forces of grasping. Note that although we show results for circular hands, our approach may be extended to any parametric 2D curve that the hand may be shaped as. The teleoperation scheme was designed with consideration for haptic transparency [27] (see Figure 6c), which allows for the impedance control of the robots.



(c) Teleoperation scheme.

Figure 6. (a) Experimental setup with motion-capture LEDs mounted for kinematic tracking and the robot hands equipped with force/torque sensors to sense haptic feedback during manipulation. (b) Hand-object configurations and equilibrium forces during the experiment. The decomposition of the equilibrium force into normal and tangential components at the contact point is also shown. (c) Impedance-control-based bilateral teleoperation scheme for dual-arm manipulation human haptic demonstrations.

Object kinematics $[r_3, \theta_3]_t$ were determined using a motion capture system that tracks the LEDs mounted on the object (see Figure 6a). The motion capture system was also calibrated with the robot joint feedback to obtain robot-hand kinematics $[r_h, \theta_h]_t$ for $h = \{1, 2\}$. We utilized an oracle to guide the calibration procedure and ensure the object kinematics

were consistent with the robot kinematics obtained using on-board sensors. The detailed approach is presented in [28], which was also used in this work with the oracle replaced by a highly accurate motion tracking system, i.e., PTI Phoenix Visualeyez. Furthermore, calibrated wrenches W_{cal} (sensed by the force/torque sensors) are used as $W_h = W_{cal}$ (computing W_{cal} from sensed wrenches is discussed in Appendix A.1). These form the inputs to the Algorithm 1.

We selected an initial configuration where the slave robot established a firm grasp on the object at $t_0 = 36$ s, which also set α_{h0} . Algorithm 1 was evaluated for all time instances in the experiment given this initial condition. To evaluate our estimates, we propagated the wrench uncertainty through Algorithm 1. First, the force/torque sensors were kept static under no load conditions, and readings were taken to compute $\Sigma_{W_h} = \text{cov}(W_h)$, as detailed in Appendix A.2. Using Equations (11)–(14), we then estimated the position and orientation, i.e., $\tilde{r}_3$ and $\tilde{\theta}_3$ estimated at each time instant. The haptic estimated object position and orientation closely follow the motion tracking results, as shown in Figure 7. This validates the proposed method.

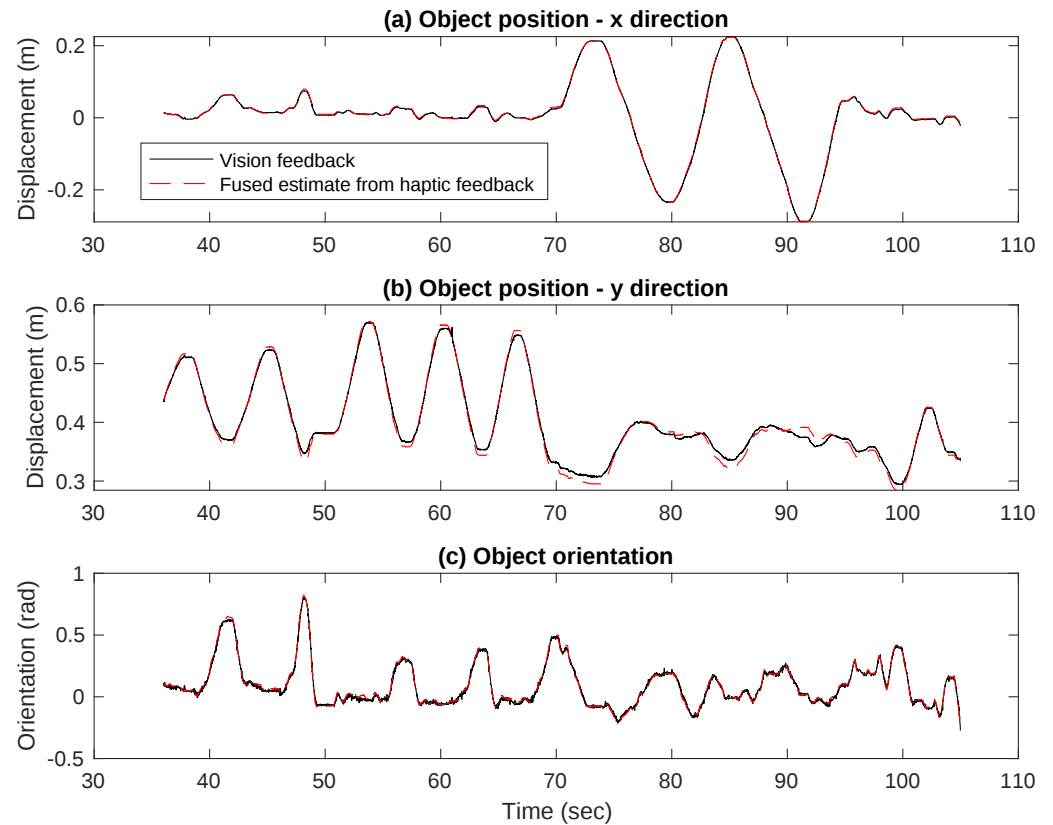


Figure 7. (a) Object position in x direction. (b) Object position in y direction. (c) Object orientation. All measurements are reported in frame $\{0\}$.

Discussion

Among recent works investigating in-hand/dual-arm manipulation for object handling and object-pose estimation, there is a mix of approaches dealing with known and unknown objects. Broadly, these may be classified into global and local methods; each work focused on a different aspect of dexterous manipulation. Among promising recent works in global sensor fusion of tactile information with vision for in-hand localization to estimate the object pose was [29], and subsequent impedance control of object pose in [30]. Their approach evaluates (for known objects) the robot kinematics, possible collisions with the object, contact points and forces, along with visual tracking of object features to refine the in-hand object pose. Among the local methods, the field of blind tactile grasping con-

trols the relative object pose (to the initial grasp) using incremental shifts after estimating contact points via control of the relative orientation object axis to the contact normals [31]. The original idea was derived from an earlier work on the dexterous control of a circular object [32]. Blind grasping methods use the definition of a virtual frame, which is a polygon created by the points of contact on multi-fingered hands. These works [17,18,33,34] present the control of the virtual frame using the standard definitions of object–hand kinematics and dynamics in multi-fingered manipulation. However, blind grasping methods typically rely on additional sensing modalities such as vision or tactile sensing to estimate the actual object pose. Our work directly contributes to this requirement by providing an additional method to sense object pose using haptic feedback during rolling contact.

Despite the example presented in this work being a polyhedral object, the key point to note is to use parameterized curves/surfaces to represent robot and object geometries. This is because robots interact with the world through curves and surfaces. Different representations of geometries have different advantages. The same polyhedral object may be approximated by a differential geometric curve that allows one to define a continuous parameter for every point, which makes it suitable to work with continuous motions that robots make. Our motivation in choosing this approach of utilizing moving-frame method (originally developed in the field of differential geometry) to robots interacting with objects through rolling contact was extensively studied in [35]. In future work, we will study the extension of our method to objects of arbitrary geometry along these lines.

Another consideration when implementing the proposed method is the stability of the teleoperation scheme. In this work, to avoid instability, the robot was moved slowly enough, allowing the human to have a chance to respond to any communication delays. In the event of communication delays, the reader is referred to a recent survey on telerobotic time delay mitigation [36] to take note of the various predictive methods involved in addressing such communication delays.

6. Conclusions

In this work, a haptic method for estimating the contact frame on a circular hand was detailed. Experimental validation of the proposed haptic method was first done for one hand by tracking the object orientation of an object under rolling contact. With this proof of concept for one hand, an algorithm to track the object position and orientation with two robot hands was presented. The kinematics of the dual-arm robot were detailed, along with the mathematical formulation of rolling contact constraints in terms of (i) rotation about robot hand and (ii) translation about the contact tangent. The final object pose was computed as a composition of these two transformations. This algorithm can also be extended to multiple hands, along with the possibility of a probabilistic estimate of the object state. An experimental robotic test-bed was built which allows a human to perform the dual-arm manipulation by teleoperation to test our proposed haptic method. The object pose was tracked using a motion capture system with high accuracy and compared to the pose computed using our method, which validated our approach.

Author Contributions: Conceptualization, S.H.T. and D.C.; Methodology, S.H.T. and D.C.; Software, S.H.T.; Validation, S.H.T.; Formal analysis, S.H.T.; Investigation, S.H.T.; Resources, D.C.; Data curation, S.H.T. and D.C.; Writing—original draft, S.H.T.; Writing—review & editing, S.H.T. and D.C.; Visualization, S.H.T.; Supervision, D.C.; Project administration, D.C.; Funding acquisition, D.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Research Foundation, Singapore, under the NRF Medium Sized Centre scheme (CARTIN). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Appendix A.1. Calibration of Force/Torque Sensors

Before using the force/torque sensors, it is important to determine their error in measurements. This means to compare the (i) measured forces and torques from the sensor given a set of (ii) expected forces and torques. To generate the expected forces, a known calibration mass is mounted on the sensor at a pre-defined location. With knowledge of the kinematics of this calibration mass in the gravity field, one can compute the expected forces. The calibration can then be a least-squares regression between these expected values and the measured ones along with an estimate of the error in the prediction offered by the regression model.

To this end, a calibration setup was built, as shown in Figure A1a, with an IMU and a loadcell in frames $\{IMU\}$ and $\{L\}$, respectively. This setup was held static against gravity using a robot, as shown in Figure A1b, through several poses to obtain a set of expected forces and measured forces. The IMU acceleration measurements $a_i \in \mathbb{R}^3$ allow one to compute the expected forces $\tilde{f}_i = ma_i$, where the m is the calibration mass which was measured on a weighing scale.

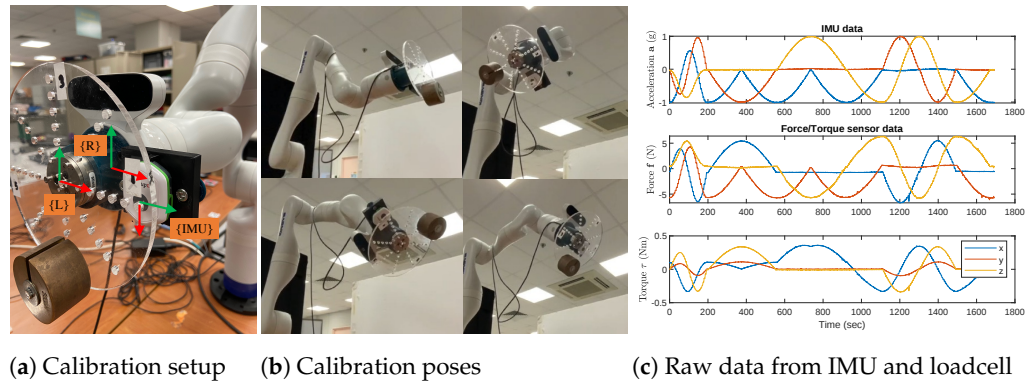


Figure A1. (a) The calibration setup consisted of the loadcell. (b) The calibration setup was held by a robot at several static poses, as shown, to obtain synchronous data from both sensors (c) The IMU data show the true effect of gravity on the calibration rig and was used to compute the expected forces to calibrate the forces recorded by the loadcell, also shown.

Force/torque sensors are typically built as strain gauges [37], with a (i) linear gain term amplifying the deflection of the gauge from (ii) an equilibrium point. Thus, we modeled our force/torque sensor similarly and aimed to regress the gain and the equilibrium bias. To this end, the model we chose to calibrate the forces yielded the following equation for each measurement f_i :

$$\left(\frac{f_{xi} - f_{0x}}{c_x}\right)^2 + \left(\frac{f_{yi} - f_{0y}}{c_y}\right)^2 + \left(\frac{f_{zi} - f_{0z}}{c_z}\right)^2 = \|f_i\| \quad (A1)$$

Here, the $c = [c_x \ c_y \ c_z]^T$ is the gain term in the calibration of forces, and the bias term is given by $f_0 = [f_{0x} \ f_{0y} \ f_{0z}]^T$, both of which are regressed from calibration data by minimizing the following cost function [38]:

$$\min_{c, f_0} \sqrt{(f_{xi} - f_{0x})^2 + (f_{yi} - f_{0y})^2 + (f_{zi} - f_{0z})^2} \cdot \left[1 - \frac{\|\tilde{f}_i\|}{\left(\frac{f_{xi} - f_{0x}}{c_x}\right)^2 + \left(\frac{f_{yi} - f_{0y}}{c_y}\right)^2 + \left(\frac{f_{zi} - f_{0z}}{c_z}\right)^2} \right] \quad (A2)$$

After these terms were determined using non-linear least squares regression (implemented in MATLAB using *lsqnonlin*), the recalibrated forces per each measurement may be computed as:

$$f_{CAL,i} = \text{diag}^{-1}(c) \cdot (f_i - f_0), \quad \text{where, } \text{diag}^{-1}(c) = \begin{bmatrix} 1/c_x & 0 & 0 \\ 0 & 1/c_y & 0 \\ 0 & 0 & 1/c_z \end{bmatrix} \quad (\text{A3})$$

Similarly, expected torques may be computed by $\tilde{\tau}_i = r_{COM} \times \tilde{f}_i$. For each i -th measurement, we have

$$\tilde{\tau}_i = \tau_0 + \text{diag}(g)\tau_i, \quad \text{where, } \text{diag}(g) = \begin{bmatrix} g_x & 0 & 0 \\ 0 & g_y & 0 \\ 0 & 0 & g_z \end{bmatrix} \quad (\text{A4})$$

Here, τ_0 is the bias term and g is the gain term. We rewrote Equation (A4) in matrix form to stack up all measurements and regress the bias and gain:

$$\tilde{\tau}_i = A_i \begin{bmatrix} g \\ \tau_0 \end{bmatrix}, \quad \text{where, } A_i = [\text{diag}(\tau_i) \mathbb{I}_3] \quad (\text{A5})$$

A least-squares regression estimates both g and τ_0 , which are used to calculate the re-calibrated torques $\tau_{CAL,i}$. Together with $f_{CAL,i}$, the re-calibrated wrench is composed as:

$$W_{CAL,i} = \begin{bmatrix} f_{CAL,i} \\ \tau_{CAL,i} \end{bmatrix} \quad \text{where, } \tau_{CAL,i} = \tau_0 + \text{diag}(g)\tau_i \quad (\text{A6})$$

Appendix A.2. Noise in Wrenches Sensed on Both Hands

In this work, we used the ATI Mini40 sensor with SI-80-4 calibration specification. According to the datasheet [37], the resolutions offered by the sensor on channels f_x , f_y and τ_z (the sensors used in our experiment) are shown in Table A1:

Table A1. Comparison of experimentally obtained standard deviation (SD) with sensor resolution reported in the data-sheet.

Quantity Name	Symbol	Value and Units
Resolution of f_x	res_{f_x}	0.02 N
Resolution of f_y	res_{f_y}	0.02 N
Resolution of τ_z	res_{τ_z}	0.0005 Nm
SD of f_x for hand 1	$\sigma_{f_{x,1}}$	0.0034393 N
SD of f_y for hand 1	$\sigma_{f_{y,1}}$	0.0057792 N
SD of τ_z for hand 1	$\sigma_{\tau_{z,1}}$	4.3925×10^{-5} Nm
SD of f_x for hand 2	$\sigma_{f_{x,2}}$	0.019268 N
SD of f_y for hand 2	$\sigma_{f_{y,2}}$	0.01733 N
SD of τ_z for hand 2	$\sigma_{\tau_{z,2}}$	0.00026706 Nm

To validate the sensor accuracy, we placed the Mini40 loadcells used on both hands for the dual-arm manipulation under no-load conditions and recorded data from them for 10 s at 1000 Hz. These data are shown in Figure A2 for both sensors. The covariance matrix W_h was computed from this. They are reported next to detail the noise levels in individual wrench channels, i.e., f_x , f_y , τ_z .

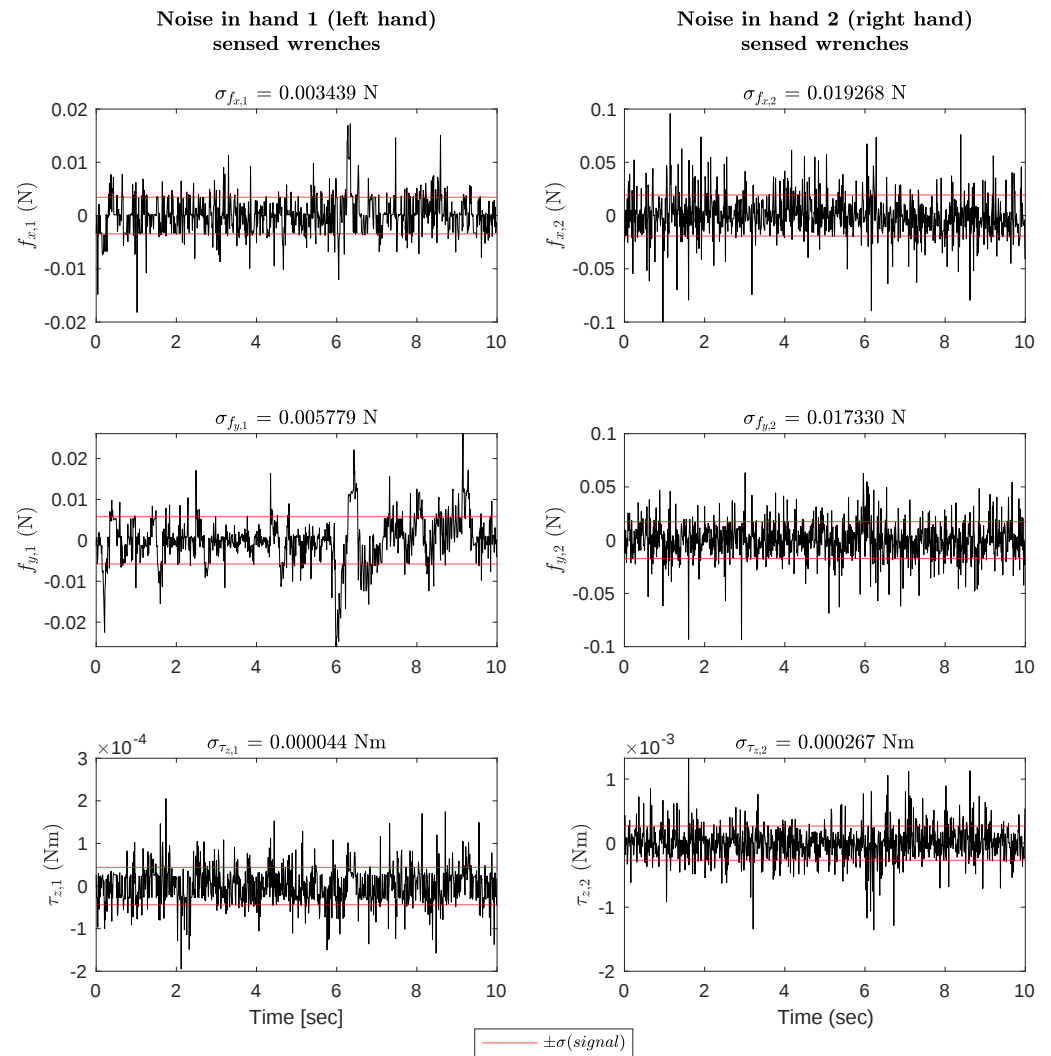


Figure A2. The force/torque sensor on both robot hands was kept static under no-load conditions, and data were collected for 10 s at 1000 Hz. The signals are portrayed here, along with the associated standard deviation per signal computed from the covariance matrix Σ_{W_h} .

The covariance of the force/torque sensor on the left hand is:

$$\Sigma_{W_1} = \begin{bmatrix} 1.1829 \times 10^{-5} & 1.9456 \times 10^{-6} & 4.261 \times 10^{-9} \\ 1.9456 \times 10^{-6} & 3.3399 \times 10^{-5} & 6.4659 \times 10^{-8} \\ 4.261 \times 10^{-9} & 6.4659 \times 10^{-8} & 1.9294 \times 10^{-9} \end{bmatrix} \quad (A7)$$

The covariance of the force/torque sensor on the right hand is:

$$\Sigma_{W_2} = \begin{bmatrix} 0.00037124 & 4.1716 \times 10^{-5} & -1.7737 \times 10^{-6} \\ 4.1716 \times 10^{-5} & 0.00030033 & 3.4014 \times 10^{-7} \\ -1.7737 \times 10^{-6} & 3.4014 \times 10^{-7} & 7.1323 \times 10^{-8} \end{bmatrix} \quad (A8)$$

The square roots of the diagonal elements of these covariance matrices represent the standard deviation of the sensors, reported in Table A1 and also in Figure A2 (in red). The reported standard deviations in wrench sensing (especially for the right hand) correspond closely to the sensor resolution reported in Table A1. It may also be noted that the left-hand sensor is five times less noisy than the right-hand one, both for forces and torques. We identify the study of uncertainty in kinematics of object pose estimated from the wrench uncertainties as future work.

References

- Okamura, A.M.; Smaby, N.; Cutkosky, M.R. An overview of dexterous manipulation. In Proceedings of the IEEE International Conference on Robotics and Automation, Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 255–262.
- Bicchi, A. Hands for dexterous manipulation and robust grasping: A difficult road toward simplicity. *IEEE Trans. Robot. Autom.* **2000**, *16*, 652–662. [\[CrossRef\]](#)
- Suomalainen, M.; Karayiannidis, Y.; Kyrki, V. A survey of robot manipulation in contact. *Robot. Auton. Syst.* **2022**, *156*, 104224. [\[CrossRef\]](#)
- Kemp, C.C.; Edsinger, A.; Torres-Jara, E. Challenges for robot manipulation in human environments [grand challenges of robotics]. *IEEE Robot. Autom. Mag.* **2007**, *14*, 20–29. [\[CrossRef\]](#)
- Srinivasa, S.S.; Ferguson, D.; Helfrich, C.J.; Berenson, D.; Collet, A.; Diankov, R.; Gallagher, G.; Hollinger, G.; Kuffner, J.; Weghe, M.V. HERB: A home exploring robotic butler. *Auton. Robot.* **2010**, *28*, 5–20. [\[CrossRef\]](#)
- Cui, J.; Trinkle, J. Toward next-generation learned robot manipulation. *Sci. Robot.* **2021**, *6*, eabd9461. [\[CrossRef\]](#)
- Correll, N.; Bekris, K.E.; Berenson, D.; Brock, O.; Causo, A.; Hauser, K.; Okada, K.; Rodriguez, A.; Romano, J.M.; Wurman, P.R.; et al. Analysis and observations from the first amazon picking challenge. *IEEE Trans. Autom. Sci. Eng.* **2016**, *15*, 172–188. [\[CrossRef\]](#)
- Kleeberger, K.; Bormann, R.; Kraus, W.; Huber, M.F. A Survey on Learning-Based Robotic Grasping. *Curr. Robot. Rep.* **2020**, *1*, 239–249. [\[CrossRef\]](#)
- Murray, R.M.; Li, Z.; Sastry, S.S. *A Mathematical Introduction to Robotic Manipulation*, 1st ed.; CRC Press: Boca Raton, FL, USA, 1994. [\[CrossRef\]](#)
- Bicchi, A.; Kumar, V. Robotic grasping and contact: A review. In Proceedings of the IEEE International Conference on Robotics and Automation, 2000 ICRA, Millennium Conference, Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 348–353.
- Trinkle, J.C. A quasi-static analysis of dextrous manipulation with sliding and rolling contacts. In Proceedings of the ICRA, Scottsdale, AZ, USA, 14–19 May 1989; pp. 788–793.
- Cole, A.; Hauser, J.; Sastry, S. Kinematics and control of multifingered hands with rolling contact. In Proceedings of the 1988 IEEE International Conference on Robotics and Automation, Philadelphia, PA, USA, 24–29 April 1988; pp. 228–233.
- Murray, R.M.; Sastry, S.S. Control experiments in planar manipulation and grasping. In Proceedings of the 1989 International Conference on Robotics and Automation, Scottsdale, AZ, USA, 14–19 May 1989; Volume 1, pp. 624–629. [\[CrossRef\]](#)
- Arimoto, S.; Yoshida, M.; Bae, J.H.; Tahara, K. Dynamic force/torque balance of 2D polygonal objects by a pair of rolling contacts and sensory-motor coordination. *J. Robot. Syst.* **2003**, *20*, 517–537. [\[CrossRef\]](#)
- Ozawa, R.; Arimoto, S.; Nakamura, S.; Bae, H.J. Control of an object with parallel surfaces by a pair of finger robots without object sensing. *IEEE Trans. Robot.* **2005**, *21*, 965–976. [\[CrossRef\]](#)
- Arimoto, S. A differential-geometric approach for 2D and 3D object grasping and manipulation. *Annu. Rev. Control.* **2007**, *31*, 189–209. [\[CrossRef\]](#)
- Shaw-Cortez, W.; Oetomo, D.; Manzie, C.; Choong, P. Object manipulation using robotic hands with varying degrees of grasp knowledge. *IEEE Trans. Robot.* **2018**, under review.
- Shaw-Cortez, W.; Oetomo, D.; Manzie, C.; Choong, P. Robust object manipulation for tactile-based blind grasping. *Control. Eng. Pract.* **2019**, *92*, 104136. [\[CrossRef\]](#)
- Bohg, J.; Morales, A.; Asfour, T.; Kragic, D. Data-driven grasp synthesis—A survey. *IEEE Trans. Robot.* **2013**, *30*, 289–309. [\[CrossRef\]](#)
- Watt, N. Deep Neural Networks for Robot Vision in Evolutionary Robotics. Master’s Thesis, Nelson Mandela University, Gqeberha, South Africa, April 2021. Available online: <http://hdl.handle.net/10948/52100> (accessed on 8 December 2022).
- Hogan, F.R.; Ballester, J.; Dong, S.; Rodriguez, A. Tactile dexterity: Manipulation primitives with tactile feedback. *arXiv* **2020**, arXiv:2002.03236.
- Yuan, W.; Dong, S.; Adelson, E.H. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors* **2017**, *17*, 2762. [\[CrossRef\]](#)
- Bicchi, A.; Salisbury, J.K.; Brock, D.L. Contact sensing from force measurements. *Int. J. Robot. Res.* **1993**, *12*, 249–262. [\[CrossRef\]](#)
- Tsujimura, T.; Yabuta, T. Object detection by tactile sensing method employing force/torque information. *IEEE Trans. Robot. Autom.* **1989**, *5*, 444–450. [\[CrossRef\]](#)
- Kana, S.; Tee, K.; Campolo, D. Human–robot co-manipulation during surface tooling: A general framework based on impedance control, haptic rendering and discrete geometry. *Robot. Comput.-Integr. Manuf.* **2021**, *67*, 102033. [\[CrossRef\]](#)
- Deisenroth, M.P.; Faisal, A.A.; Ong, C.S. *Mathematics for Machine Learning*; Cambridge University Press: Cambridge, UK, 2021; p. 201.
- Kuchenbecker, K.J.; Fiene, J.; Niemeyer, G. Improving contact realism through event-based haptic feedback. *IEEE Trans. Vis. Comput. Graph.* **2006**, *12*, 219–230. [\[CrossRef\]](#)
- Kana, S.; Gurnani, J.; Ramanathan, V.; Turlapati, S.H.; Ariffin, M.Z.; Campolo, D. Fast Kinematic Re-Calibration for Industrial Robot Arms. *Sensors* **2022**, *22*, 2295. [\[CrossRef\]](#)

29. Pfanne, M.; Chalon, M.; Stulp, F.; Albu-Schäffer, A. Fusing joint measurements and visual features for in-hand object pose estimation. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3497–3504. [[CrossRef](#)]
30. Pfanne, M.; Chalon, M.; Stulp, F.; Ritter, H.; Albu-Schäffer, A. Object-level impedance control for dexterous in-hand manipulation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 2987–2994. [[CrossRef](#)]
31. Montaña, A.; Suárez, R. Manipulation of unknown objects to improve the grasp quality using tactile information. *Sensors* **2018**, *18*, 1412. [[CrossRef](#)] [[PubMed](#)]
32. Ozawa, R.; Arimoto, S.; Nguyen, P.T.A.; Yoshida, M.; Bae, J.H. Manipulation of a circular object in a horizontal plane by two finger robots. In Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics, Shenyang, China, 22–26 August 2004; pp. 517–522.
33. Lee, D.H.; Choi, M.S.; Park, H.; Jang, G.R.; Park, J.H.; Bae, J.H. Peg-in-Hole Assembly With Dual-Arm Robot and Dexterous Robot Hands. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8566–8573. [[CrossRef](#)]
34. Cortez, W.S.; Oetomo, D.; Manzie, C.; Choong, P. Robust object manipulation for prosthetic hand applications. In Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, VIC, Australia, 12–15 December 2017; pp. 5821–5826.
35. Lei, C.; Dai, J.S. *Sliding-Rolling Contact and In-Hand Manipulation*; World Scientific: London, UK, 2020.
36. Farajiparvar, P.; Ying, H.; Pandya, A. A brief survey of telerobotic time delay mitigation. *Front. Robot. AI* **2020**, *7*, 578805. [[CrossRef](#)]
37. ATI Industrial Automation: F/T Sensor mini40. (n.d.). Available online: https://www.ati-ia.com/products/ft/ft_models.aspx?id=mini40 (accessed on 8 December 2022).
38. Campolo, D.; Fabris, M.; Cavallo, G.; Accoto, D.; Keller, F.; Guglielmelli, E. A novel procedure for in-field calibration of sourceless inertial/magnetic orientation tracking wearable devices. In Proceedings of the First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 20–22 February 2006; pp. 471–476.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.