2025 IEEE/SICE International Symposium on System Integration (SII)
January 21-24, 2025, Munich, Germany

Online object localization in a robotic hand by tactile sensing

Ali Hammoud[1], Mahdi Khoramshahi[1], Quentin Huet[1] and Veronique Perdereau[1]

Abstract— **Robotic grasping and manipulation mainly rely on vision and tactile sensing. While tactile sensors are frequently proposed for grasp control in the literature, object localization and recognition are typically achieved through vision. Vision-based approaches perform satisfactorily in clear and structured surroundings by providing reliable sensory inputs about the object. However, their performance deteriorates when faced with object occlusion, which is typical of manipulation tasks; for instance, robotic fingers occluding the object during in-hand object manipulation. This work presents an online object pose estimation based on tactile sensing. More specifically, the proposed method finds the wrist-object transformation based on the contact-point positions between fingertips and the object, representing the minimal tactile sensor requirement. We validate our method experimentally using different object geometries during in-hand manipulation tasks. The experimental results demonstrate that our proposed method outperforms the vision-based approaches during in-hand object manipulation due to its inherent robustness to object occlusion.**

I. Introduction

Robot manipulation requires an accurate and robust perception of the object's pose, especially during fine in-hand manipulation. Nevertheless, most robotic applications rely solely on vision sensing to localize objects [1]–[4]. Indeed, vision-based approaches proved effective in providing relevant information for executing in-hand manipulation actions. However, dealing with occlusions is considered the main drawback of such approaches; i.e., the object is not fully visible to the camera. Occlusion is ubiquitous in manipulation tasks, especially during in-hand manipulation when robotic fingers surround the objects. A typical occlusion example is shown in Fig. 1, where the camera viewpoint is displayed with a visualization tool indicating the robot and object poses acquired by the vision system. Comparing the two images, it is clear that the object is not in the correct position with respect to the robot. Humans benefit from vision and tactile sensing to localize objects for manipulation purposes. Similarly, recent studies [6], [7] showed the advantages of other sensory inputs. For instance, multimodal sensory inputs can be fused to overcome occlusion-related problems. For example, Bimbo et al. 2013 [8] used tactile sensing to correct vision-based object localization. Other object localization algorithms have combined tactile sensing and visual features in several studies as presented by Chaudhury et al. 2022 in [9]. A comparison between the fusion of different types of sensory modalities and vision in robotics was made in [10]. This comparison shows that the combination of the three modalities outperforms the other cases.

[1]A. All authors are with the Institute of Intelligent Systems and Robotics (ISIR) at Sorbonne University, Paris, France. ali.hammoud@sorbonne-universite.fr

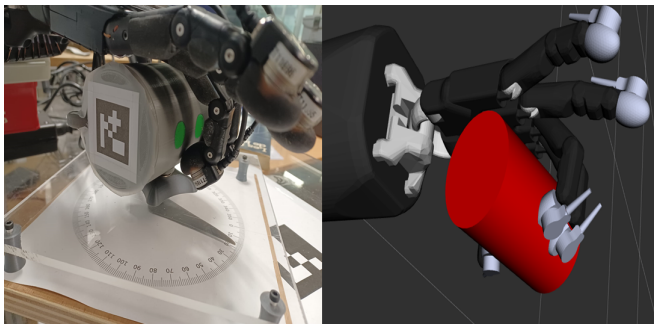


Fig. 1: Camera viewpoint, and visualization of the robot's current posture and visualization of the robot's current posture with the vision-acquired object pose by Garrido-Jurado et al. 2014 [5]. The visualization algorithm starts to lose its accuracy when the ArUco marker is not clear to the camera.

Vision-based approaches are limited when objects are out of sight or partially occluded. This inherent limitation begs for approaches that utilize or combine vision with other sensory modalities. In [11], they proposed a tactile-based approach for object localization for a force-controlled robot. However, their method is limited to three degrees of freedom due to the high computational cost of using a particle filter. Petrovskaya et al. 2006 [12] extended this method to six degrees of freedom using an adaptive particle filter, increasing the precision as the localization process progresses. In [13], [14], the object pose was estimated from the contact information. By comparing the finger measurements to a previously learned model, the algorithm generates hypotheses for possible contact points. These hypotheses were then used to determine which object pose best represented the measurements. Moreover, other works [15], [16] relied on particle filters to track object-hand configuration from tactile sensing. To this end, these methods use measurement models that consider areas on the manipulator that are not in contact with the object.

Most existing approaches to object localization overlook the kinematic or dynamic model of the hand due to the added theoretical and practical challenges. An exception is Zhang et al. (2013) [17], who estimated the object pose and dynamic model using pre-recorded data in an offline manner. Another method by [18] employed an extended Kalman filter (EKF) with finger position and torque measurements for online localization during in-hand manipulation. However, torque measurements are affected by gravity and movement dynamics, leading to varying results for the same object-hand

position depending on hand orientation. In contrast, tactile sensing, which relies on contact points, offers robustness against hand position and dynamics.

Robotic hands, during in-hand manipulation, can move objects through re-grasping or finger gaiting, necessitating continuous object pose estimation. Vision-based methods are effective unless the object is occluded. Bimbo et al. [8] tackled occlusion by integrating vision and touch, estimating the nearest object pose within hand constraints but without supporting finger gaiting or changing contact points during manipulation. Due to high computational demands, their tactile methods have been applied offline. Chalon et al. (2013) [19] developed an online tactile sensing method for object pose estimation, but it was limited to grasping, arm movement, and release, excluding in-hand manipulation. Moreover, methods combining object recognition with pose estimation may compromise pose accuracy due to uncertainties in object position [20], [21]. Furthermore, recent research highlights the ability of neural network-based methods for object localization to overcome occlusion. For example, Azulay et al. (2013) [22] employed Gaussian Processes, Feedforward Neural Networks, and Long Short-Term Memory Networks, achieving outstanding results. Similarly, Goudie et al. (2017) [23], Doosti et al. (2020) [24] and Azulay et al.(2023) [22] focused on Convolutional Neural Networks for 3D hand-object pose estimation, achieving notable accuracy. These studies demonstrate that while neural network-based methods can enhance object localization, their high computational and data requirements related to object shape and training data dimensionality present significant drawbacks.

In this paper, we contribute to this body of literature by proposing an online method for object localization based on tactile sensing, hand kinematics, and object model. Through experimental validation and comparison to vision-based methods, we demonstrate that our proposed method is effective in overcoming the aforementioned limitations; i.e., robustness to occlusion, low computational cost, minimal tactile sensing, and accurate tracking performance. The rest of this paper is organized as follows. Section II formulates the object localization problem using contact points. In Section III, we propose a fast and computationally low-cost solution. In Section IV, we validate our proposed method in simulation. Sections V and VI present our experimental setup and results respectively. Finally, Section VII provides discussions and conclusions.

II. Problem formulation

In this paper, we propose an effective method to solve the following problem: "Online" object localization during in-hand manipulation using "a minimum number of fingers" in contact with the object. Most often, the object localization problems are reduced to an optimization problem to find the geometric transformation between the robotic wrist/palm and the object. In this work, the object pose is represented by a frame $R_b(O_b, i_b, j_b, k_b)$ with respect to a fixed frame attached to the palm of the robotic hand. We also assume that the 3D shape of the object is known. The geometrical model of the object allows us to specify any contact points in the object frame. Furthermore, we assume that the contact points stay fixed during in-hand manipulation; i.e., there is no slippage between fingertips and the object. Moreover, in our work, the positions of the contact points (specified in the palm frame) are available using tactile sensors. Finally, we assume that we have sufficient contact points specified in both frames. It is also important to note that, given the non-slippage assumption, only an initial pose for the object is necessary to compute the contact points in the object frame. Nevertheless, this is not a limiting assumption since one can choose an arbitrary frame for the object based on the initial contact points.

In summary, our assumptions are: **(A1)** A 3D mesh of the object and the hand (palm and fingers) are available. **(A2)** Absence of slippage between the hand and the object. **(A3)** The positions of the fingers and of the contact points on the fingertips are available online. **(A4)** Sufficient contact points specified in both hand and object frame. **(A5)** The initial object pose is provided.
To localize the object, we need to find the object pose (including translation and rotation) with respect to a fixed frame. Conventionally, the position is represented by a 3d Euclidean vector; i.e., 3 unknown parameters. However, several representations can be used to describe the orientation of the object; for example, Euler angles, axis-angle, quaternions, and rotation matrices. Each of these representations leads to a different estimation problem with its related challenges. In the following, we consider two of these possibilities. However, let us first consider $p_b = [x_b, y_b, z_b]^T$ and $p_w = [x_w, y_w, z_w]^T$ as the position of a contact point in the object and world/palm frame, respectively. Moreover, $d = [d_1, d_2, d_3]^T$ represents the position of the object (i.e., O_b) in the world frame.

Objects move in space according to rigid motions, which include only two types of motions: translation and rotation. In robot kinematics, the dual quaternion vector is widely used since it can represent the roto-translation with fewer components than a homogeneous transformation matrix [25], [26]. However, the homogeneous transformation matrix performs better than the dual quaternion in solving our problem because it leads to a direct estimation of the object localization.

Quaternion: The object's orientation is represented by $q = [q_r, q_i, q_j, q_k]$ which allows the following transformation between the two frames:

$$\begin{bmatrix} 0 \\ p_w \end{bmatrix} = q[0, p_b^T]q^* + \begin{bmatrix} 0 \\ d \end{bmatrix} \tag{1}$$

where $q^* = [q_r, -q_i, -q_j, -q_k]$ denotes the conjugate trans-

pose of q. This expression can be expanded into:

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \begin{bmatrix} 0 \\ (q_r^2+q_i^2-q_j^2-q_k^2)x_b + 2(q_iq_j-q_kq_r)y_b + 2(q_rq_k+q_jq_r)z_b + d_1 \\ 2(q_iq_j+q_kq_r)x_r + (q_r^2-q_i^2+q_j^2-q_k^2)y_b + 2(q_jq_k-q_iq_r)z_b + d_1 \\ 2(q_iq_k-q_jq_r)x_b + 2(q_jq_k+q_iq_r)y_b + (q_r^2-q_i^2-q_j^2+q_k^2)z_b + d_1 \end{bmatrix} \tag{2}$$

Given a set of contact points (specified in both frames), the localization problem simplifies into finding these four unknown parameters in each time-step during the in-hand manipulation. However, the non-linearity with respect to these parameters hinders fast and simple methods for achieving precise solutions.

Rotation matrix: Given the rotation matrix $R \in SO(3)$ that represents the object's orientation, we can write:

$$\begin{bmatrix} p_w \\ 1 \end{bmatrix} = \begin{bmatrix} R & d \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} p_b \\ 1 \end{bmatrix} \tag{3}$$

where $0^T = [0,0,0]$. The expansion leads to:

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \begin{bmatrix} r_{11}x_b + r_{12}y_b + r_{13}z_b + d_1 \\ r_{21}x_b + r_{22}y_b + r_{23}z_b + d_2 \\ r_{31}x_b + r_{32}y_b + r_{33}z_b + d_3 \end{bmatrix} \tag{4}$$

where r_{ij} is the element in the ith row and jth column of the rotation matrix R.

As shown in Eq. 2, the expansion of the dual quaternion in Eq. 1 results in a complex non-convex problem that necessitates an optimization process, requiring significant time to resolve. On the other hand, if we use the homogeneous transformation matrix, we have twelve parameters, as we see in Eq. 4, the expansion of the transformation matrix Eq. 3. Having such linear equations allows for obtaining a fast and efficient solution that represents a homogenous transformation matrix. Therefore, the object localization problem is formulated as identifying $d = [x_d, y_d, z_d]^T$, the object position, and R, the object rotation matrix.

III. Proposed method

This section discusses the mechanism for object localization and the minimization of inputs to the method.

A. Object localization method

To find the parameters of the transformation matrix, we need four known vector representations in the world $\mathbb{R}_w$ and the object frame $\mathbb{R}_b$; namely $p_w^{(1)}$, $p_w^{(2)}$, $p_w^{(3)}$, $p_w^{(4)}$ with their respective representation in object frame as $p_b^{(1)}$, $p_b^{(2)}$, $p_b^{(3)}$, $p_b^{(4)}$. Given these points, Eq. 4 can be expanded as:

$$W = BM \tag{5}$$

where

$$W = \begin{bmatrix} x_w^{(1)} & y_w^{(1)} & z_w^{(1)} \\ x_w^{(2)} & y_w^{(2)} & z_w^{(2)} \\ x_w^{(3)} & y_w^{(3)} & z_w^{(3)} \\ x_w^{(4)} & y_w^{(4)} & z_w^{(4)} \end{bmatrix}, \; B = \begin{bmatrix} x_b^{(1)} & y_b^{(1)} & z_b^{(1)} & 1 \\ x_b^{(2)} & y_b^{(2)} & z_b^{(2)} & 1 \\ x_b^{(3)} & y_b^{(3)} & z_b^{(3)} & 1 \\ x_b^{(4)} & y_b^{(4)} & z_b^{(4)} & 1 \end{bmatrix} \tag{6}$$

with the unknown parameters:

$$M = \begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \\ d_1 & d_2 & d_3 \end{bmatrix} \tag{7}$$

The "exact solution" exists if A_b is invertible:

$$M = B^{-1}W \tag{8}$$

However, the resulting solution is not necessarily a rotation matrix, especially when dealing with noisy measurements. In order to satisfy this constraint, we use the Singular Value Decomposition ($R = USV^T$) to derive the following polar decomposition:

$$R = \tilde{R}P \tag{9}$$

where $\tilde{R} = UV^T \in SO(3)$ and $P = VSV^T \in \mathbb{R}^{3\times3}$ is positive-definite matrix which represent scaling part of R. With the modified rotation part (i.e., $\tilde{R}$), we update the translation part. To this end, we expand Eq 5 as follows:

$$W = B_1R + B_2d \tag{10}$$

where $B_1 \in \mathbb{R}^{4\times3}$ and $B_2 \in \mathbb{R}^{4\times1}$ are segments of $B = [B_1, B_2]$. Thus, having updated $\tilde{R}$, we have:

$$\tilde{d} = (B_2)^{\dagger}(W - B_1\tilde{R}) \tag{11}$$

where † denotes the left psuedo-inverse.

Given the quasi-static updates on the two parts, we can perform the estimation process sequentially as follows:

$$\begin{aligned} \Delta_R &= (B_1\tilde{R}_{k-1})^{-1}(W - B_2d_k) \\ \Delta_R &= \tilde{\Delta}_R P \\ \tilde{R}_k &= \tilde{R}_{k-1}\tilde{\Delta}_R \\ \tilde{d}_k &= (B_2)^{\dagger}(W - B_1\tilde{R}_k) \end{aligned} \tag{12}$$

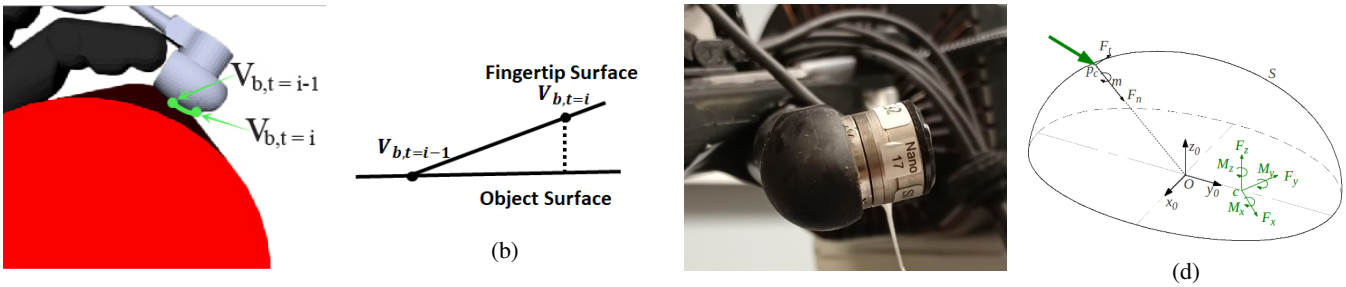
where iteration k utilizes the previous estimation for the rotation $\tilde{R}_{k-1}$. The estimation process can run asynchronously from the task; i.e., in each iteration, we use the most updated values for W and B.

The quality of this parameter identification can be quantified by the following the Frobenius norm:

$$e = \frac{1}{\sqrt{N}}||W - B\tilde{M}||_F \tag{13}$$

where $N = 4$ as we use four contact points in this work. It is important to note that this quantity represents the fitting error and not the localization error for which we need a ground truth.

As we can see, unlike systems based on tactile sensors, we do not need complex optimization procedures. Instead, we use a set of linear systems (Eq. 5) to find the parameters of the object pose followed by a polar decomposition. This method can be efficiently applied to online implementations since it does not require an optimization process that requires an unpredictable processing time. As stated previously, Our



(a) (b) (c) (d)

Fig. 2: Design of the contact sensing fingertip based on force/torque sensors. (a) Point of contact V_b at time i and $i-1$ with respect to the object frame. (b) Projecting the actual point of contact on the object's surface in its previous pose. (c) Detail of the force and torque sensor mounted on a fingertip. (d) The resulting forces and torques on an ellipsoid when a force is acting on point p_c

approach only relies on four position vectors represented in both frames. In the subsection below, we show how we compute these vectors.

B. Object localization input

As previously stated, our approach requires four vector representations in both frames. We consider that the robot manipulates the objects using its fingertips. Thus, the contact points between the object and the fingertips are common in both frames that can be quantified. In our proposed approach, we use such contact points as input variables. Objects can be manipulated by a variable number of fingers depending on the in-hand manipulation scenario and the number of fingers on the robotic hand. The number of fingers in contact with the object can range from two to five. We use feedback from four tactile sensors when we have four or more fingertips in contact with the object. On the other hand, if we have only three or two fingers in contact with the object, we have to find additional vectors representing the object.

In the case of three contact points, we can derive the fourth point from the data acquired by the three sensors. The fourth point must not create a singularity to ensure the invertibility of the matrix B. Therefore, the created point is the centroid of the triangle formed by the three points levitated u(m) with the normal vector $\vec{n}$ to the triangle surface found in the two references as described in Eq. 14. Thus, we can guarantee the invertibility of the matrix B.

$$p_b^{(4)} = \frac{1}{3}(p_b^{(1)} + p_b^{(2)} + p_b^{(3)}) + u\vec{n} \tag{14}$$

If two fingers are in contact with the object, we cannot find the representation of the third and fourth vectors in both frames. From a mathematical perspective, two points can only represent a line in 3D space. Therefore, we cannot create three independent vectors that can represent a reference in the 3D workspace. Therefore, the minimum number of contacts between the object and the hand must be three to apply our method. In this section, we introduced our method concept based on the transformation matrix and contact points between the object and the hand. The roto-translation matrix helped us find a direct method that can be applied online without passing through an optimization process. In addition, we were able to minimize the number of contact points required.

IV. Simulation

To validate the proposed object localization method before real-world implementation, simulations were conducted in controlled environments to assess performance under varying noise conditions. These simulations replicated scenarios in which a robotic system manipulates objects using fingertip contact points for localization.

Contact points were generated in both the world and object frames, with two types of noise introduced: (1) sensor measurement errors and (2) inaccuracies in the estimation of contact points relative to the object frame as described in Eq.15 and 16 respectively. These errors were simulated through iterative updates, incorporating random noise ν_{sensor} based on the step size $\Delta(V_{\text{sensor}})$. Sensory data were propagated through projection, influencing a proportional noise component $\nu_{\text{projection}}$ in the object frame.

$$\hat{V}_{\text{sensor}} = V + \Delta(V_{\text{sensor}})\nu_{\text{sensor}} \tag{15}$$

$$\hat{V}_{\text{object}} = V_{\text{object}} + \Delta(V_{\text{object}})\nu_{\text{projection}} \tag{16}$$

where V_{object} represents the projected contact point of $\hat{V}_{\text{sensor}}$ on the manipulated object. This approach enabled the evaluation of localization performance in noisy conditions.

Simulations were implemented in Python, utilizing NumPy for iterative updates and noise generation. Ten trajectories, each lasting 20 seconds, were generated per noise level going up to 10%, simulating in-hand manipulation. Both noises ν_{sensor} and $\nu_{\text{projection}}$ can reach up to 10%, introducing realistic inaccuracies into the contact points. The results, illustrated in Fig.3, demonstrate that localization errors increase with higher noise levels, reaching up to 6 degrees in orientation and 0.2 cm in translation. The method

maintained robustness up to 8% noise; beyond this threshold, accuracy significantly deteriorated. These findings suggest that the proposed method is effective under controlled noise conditions and is suitable for real-world implementation within specific noise limits. The following section presents experimental results obtained from real-world scenarios.

V. Experimental setup

Two steps are necessary to implement the object localization technique. The first step is to measure the points of contact between the hand and the manipulated object with respect to the world frame and the object frame. The second step is to apply the proposed geometric object localization method.

A. Contact points measurement

During in-hand manipulation, hand configuration changes are performed by the fingers. The contact points between the object and the hand correspond to the fingertips in contact with the object. The minimum number of contacts between the hand and the object is three. Using the tactile sensors positioned at the robot's fingertips, we can capture three contact points between the object and the hand. These measured points of contact must be represented with respect to the world frame and the object frame. For every finger j in contact with the object, using the finger kinematics transformation matrix $T_s^{w,(j)}$ and the jth fingertip sensor measurements $p_s^{(j)}$, we obtain the point of contact with respect to the world frame, as shown in the equation below:

$$p_w^{(j)} = T_s^{w,(j)} p_s^{(j)} \tag{17}$$

We cannot directly measure the points of contact with respect to the object frame since the object is not equipped with sensors to detect the points of contact. Therefore, it will be estimated using the prior object pose and the difference between the prior point of contact on the sensor and the current one. The difference in the contact sensor readings between two consecutive readings is very small. The displacement of the contact point on the sensor surface can be considered as a linear displacement. Thus, the actual point of contact representation with respect to the object frame is found by projecting the actual contact point on the object in its prior pose as shown in Fig. 2. After expressing the inputs of our method in both frames, we now have to use them in the object localization algorithm.

B. Geometric Object localization method

As already explained in sub-section II, we have to solve twelve equations using the data of four vectors represented in the world and object frames to estimate the roto-translation matrix parameters. To implement our geometrical method, we need the prior object pose and the actual points of contact in both the world and the object frames that will be the algorithm's inputs to estimate the object pose. An algorithm flow is represented in Fig. 4 to make our method clearer to the reader.

VI. Results

Our method addresses occlusion issues in vision-based localization and is designed for online execution. To better evaluate our approach, we compare its performance with an online vision method for estimating object posture, using a ground truth for validation. Additionally, we compare the linear localization results with a touch-vision method tested on the same platform [8], aiming to assess our algorithm's suitability for online applications. The following sections detail the platform and present localization results.

A. System overview

To test our algorithm, we used a robotic hand equipped with fingertip sensors and an RGB-D camera to detect the object's position.

The proposed algorithm was implemented in C++ using the ROS platform and tested on a real-world system, such as the one displayed in Fig. 4b. The platform consists of a Shadow hand-arm robot with a Microsoft Kinect RGB-D camera positioned on the shoulder and custom-designed fingertips with 6-axis ATI nano17 force and torque sensors. An RGB-D camera is configured, so that the object is always detectable in the camera frame.
Liu et al. 2012 [27] introduced a tactile sensing technique to calculate contact location and local torque from force and torque measurements on a convex surface (Fig. 2d). This method computes the normal and tangential interaction forces by finding the contact point $P_c = (x, y, z)$ that satisfies moment and force equations, assuming torque is normal to the surface:

$$\begin{cases} P_c \times \vec{F} + \vec{m} = \vec{M} \\ S(x, y, z) = 0 \end{cases} \tag{18}$$

Tested in [28], it achieved $266\mu m$ accuracy and ran at frequencies above $800Hz$.

To demonstrate the advantages of our method, we implemented a vision algorithm capable of locating objects under occlusion, using the fiducial marker-based method by Garrido-Jurado et al. 2014 [5].

B. Results

The approach was tested on two different objects: a regular cylinder and a square parallelepiped. We chose these two objects due to their unique characteristics. The faces and normals of the square parallelepiped are clearly defined, whereas rotational symmetry exists in cylindrical objects.

For our experimental setup, we only needed to focus on in-hand manipulations. Therefore, we fixed the shadow hand to immovable support, and all the object manipulations were done by actuating the fingers' joints. This setup allowed us to move the object around its axis by 30° or -30° from its neutral orientation. We chose to limit ourselves to a +25° movement to limit the mechanical constraints on the fingers since this range of motion was enough to test our method and compare it to the camera-based approach. Due to the

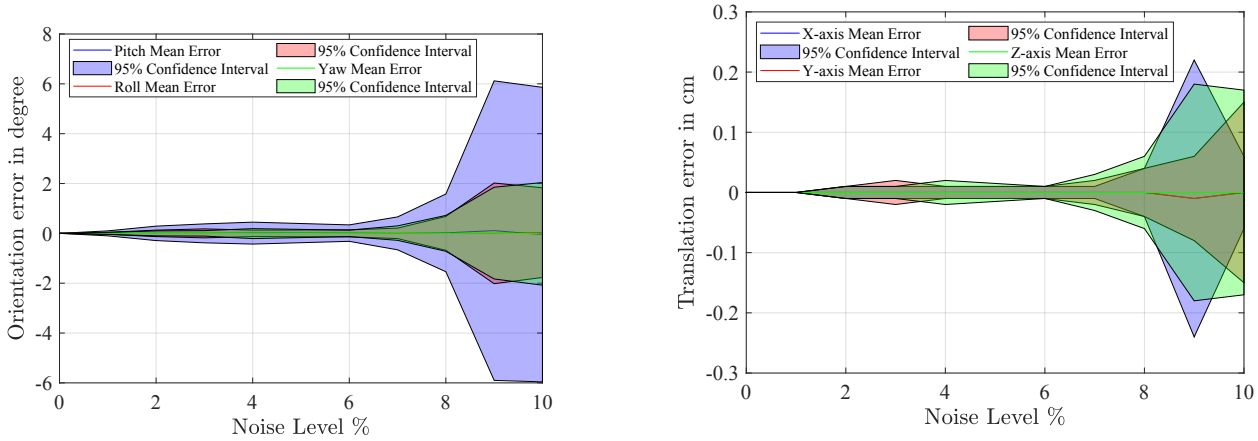
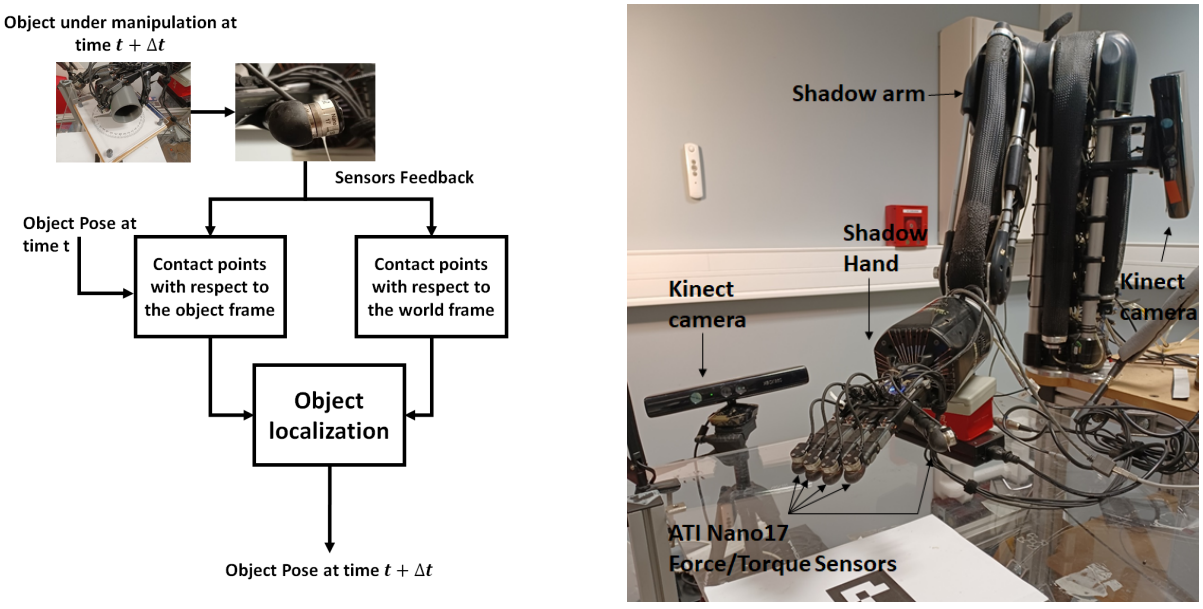


Fig. 3: Effect of Noise on Measurement Error for Orientation and Translation. The subfigures show the mean error and confidence range as noise levels increase. The subfigure on the left demonstrates that orientation prediction errors reach up to 6 degrees at high noise levels, while the confidence range widens significantly. The subfigure on the right illustrates that translation errors can increase up to 0.2 cm under high noise conditions, with a similar expansion in the confidence range. These results indicate that the system effectively corrects for object orientation and translation errors when noise is minimal. However, as noise levels rise, the system's ability to accurately predict object pose during in-hand manipulation deteriorates, leading to increased localization errors.



(a) (b)

Fig. 4: (a) Our object localization method inputs are the object pose at time t and the sensor data from its pose at time $t+\Delta t$. We can directly measure the point of contact with respect to the world frame based on the hand and fingertip sensors. The points of contact with respect to the object frame are estimated by projecting the fingertip sensor feedback onto the object at time t. Object localization uses the representation of the contact points with respect to the world frame and the object frame to estimate the object pose at time $t+\Delta t$. (b) Platform description.

difficulty of having a continuous ground truth, the validation method consisted of fixing the object to a measuring angle system, as shown and explained in Fig 6. The robotic hand rotated the object from pose i to pose $i+1$, paused on pose $i+1$ for a minimum of two seconds, and then moved to the next desired pose $i+2$. We adopted this procedure to compare the ground truth value read with the vision method and our object localization method. The robotic hand rotated the object until the ArUco marker was occluded and no longer visible to the RGB-D camera. The mean error of the vision estimates (measured when the ground truth was known) was 5° for the cylindrical object and 4° for the square parallelepiped when the object was not occluded while using our object localization method it was reduced to 3° and 2.5°,

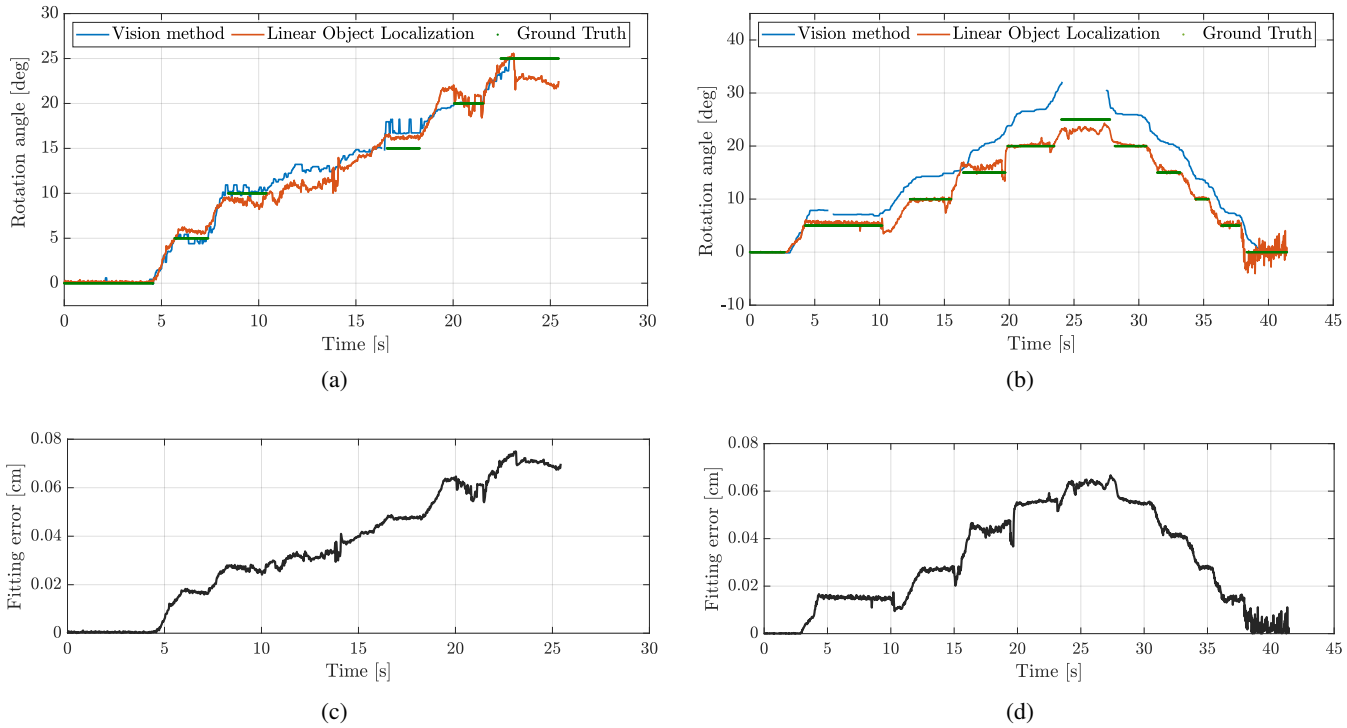


Fig. 5: In (a) and (b), the experimental results – the blue stars and the red line represent the rotation angle based on vision and our localization method, respectively. The green dots represent the recorded ground truth. The vision-based method lost the ability to detect the object pose for both objects when its rotation was 25 °. In the cylinder results, we can see how the vision-based method was able to re-detect the object when the Shadow hand was rotating the object back to 0°. In (c) and (d), the correction factor – the blue line represents Frobenius norm ε of the difference between R and $\tilde{R}$. This estimation error represents how much we had to correct the matrix R in order to be orthonormal.

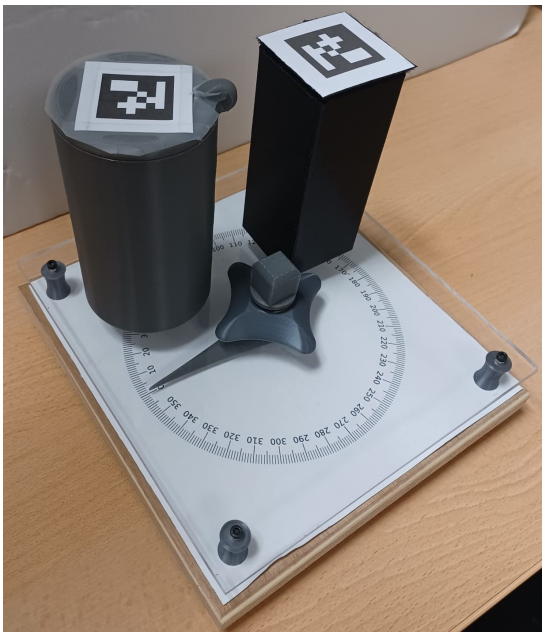


Fig. 6: Tested objects in our experimental validation are placed on the ground truth measurement tool to determine the angles at which the objects are rotated by the robotic hand.

respectively. On the other hand, when the ArUco marker was occluded, the error with the vision method started to increase until not being able to detect the object pose as we see in Fig. 5a and 5b. The fitting error reached a maximum of 0.0745 cm across all recorded manipulation actions, as shown in Fig. 5c and 5d for two manipulation actions. This indicates that the quasi-static updates on the rotation matrix and translation vector were minimal during our experimental trials.

The EKF concept is the basis of the method suggested in [18]. The object pose was estimated in power grasp, finger grasp, and in-hand manipulation actions in experimental validation. Since we aim to estimate the object pose during in-hand manipulation, we will compare our results with their experiment results. The object localization based on the filtering concept in [18] was able to estimate the object pose under an uncertainty range of 3σ. The error around the main rotation component during the in-hand manipulation action lies between -1 ° and 7.5 ° with respect to the vision system ground truth. At the end of the experiment, the rotation angle error is 2°. On the other hand, our method's error during the in-hand manipulation task lies between -3.5° and 1.4°, and at the end of the experiment, the rotation angle error is -3.5°.

The algorithm's execution time was 0.0132 seconds on average, with an average number of iterations of 80. On the other hand, methods that combine vision and tactile sensors,

such as the method presented in [8], took 0.171 seconds on average as running time and 91.2 iterations on average. Finally, the experiment showed the ability of our algorithm to overcome the occlusion scenarios and keep on detecting the object pose. At the same time, vision failed to localize the object when the ArUco marker was occluded or not fully visible in the camera frame. The comparison between the method of Pfanne et al. [18] and ours showed the ability of our method to detect the object pose better than other online methods that take into account the filtering concept. The comparison between the method of Bimbo et al. 2013 [8] and ours shows the speed of our algorithm compared to other online methods that take into account tactile sensors to localize the object.

VII. Discussion and Conclusion

This work presented a method for object localization during in-hand manipulation. While vision systems perform well in unoccluded environments, occlusions from hand movements reduce tracking accuracy. To address this, we developed an algorithm that uses fingertip contact detection to estimate object pose by computing a transformation matrix from contact points in the object and wrist frames. The algorithm successfully localized the object under both occluded and non-occluded conditions, with a mean computation time of 0.0132 seconds, enabling real-time tracking. The estimation error was between -3.5 and 1.4 degrees. Future work aims to enhance robustness by incorporating local torque data to estimate the object's center of mass and integrating a slip detection mechanism for corrective adjustments.

ACKNOWLEDGMENT

This work has been supported by the CHIST-ERA (2014-2020) project InDex and received funding from Agence Nationale de la Recherche (ANR) under grant agreement No. ANR-18-CHR3-0004.

References

[1] P. K. Allen, A. Timcenko, B. Yoshimi, and P. Michelman, "Automated tracking and grasping of a moving object with a robotic hand-eye system," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 2, pp. 152–165, 1993.

[2] M. Ulrich, C. Wiedemann, and C. Steger, "Cad-based recognition of 3d objects in monocular images.," in *ICRA*, vol. 9, pp. 1191–1198, 2009.

[3] T. Patten, K. Park, M. Leitner, K. Wolfram, and M. Vincze, "Object learning for 6d pose estimation and grasping from rgb-d videos of in-hand manipulation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4831–4838, 2021.

[4] S. Cui, R. Wang, J. Hu, J. Wei, S. Wang, and Z. Lou, "In-hand object localization using a novel high-resolution visuotactile sensor," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 6, pp. 6015–6025, 2021.

[5] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[6] R. S. Johansson and J. R. Flanagan, "Coding and use of tactile signals from the fingertips in object manipulation tasks," *Nature Reviews Neuroscience*, vol. 10, no. 5, pp. 345–359, 2009.

[7] R. S. Johansson and J. R. Flanagan, "Tactile sensory control of object manipulation in human, volume handbook of the senses: Vol. 5-somatosensation," 2007.

[8] J. Bimbo, L. D. Seneviratne, K. Althoefer, and H. Liu, "Combining touch and vision for the estimation of an object's pose during manipulation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4021–4026, IEEE, 2013.

[9] A. N. Chaudhury, T. Man, W. Yuan, and C. G. Atkeson, "Using collocated vision and tactile sensors for visual servoing and localization," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 3427–3434, 2022.

[10] M. Prats, P. J. Sanz, and A. P. Del Pobil, "Vision-tactile-force integration and robot physical interaction," in *2009 IEEE international conference on robotics and automation*, pp. 3975–3980, IEEE, 2009.

[11] K. Gadeyne and H. Bruyninckx, "Markov techniques for object localization with force-controlled robots," in *10th Int'l Conf. on Advanced Robotics*, pp. 91–96, 2001.

[12] A. Petrovskaya, O. Khatib, S. Thrun, and A. Y. Ng, "Bayesian estimation for autonomous object manipulation based on tactile sensors," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 707–714, IEEE, 2006.

[13] S. Haidacher, *Contact point and object position from force/torque and position sensors for grasps with a dextrous robotic hand*. PhD thesis, Technische Universität München, 2004.

[14] S. Haidacher and G. Hirzinger, "Estimating finger contact location and object pose from contact measurements in 3d grasping," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 2, pp. 1805–1810, IEEE, 2003.

[15] C. Corcoran and R. Platt, "A measurement model for tracking hand-object state during dexterous manipulation," in *2010 IEEE International Conference on Robotics and Automation*, pp. 4302–4308, IEEE, 2010.

[16] R. Platt, F. Permenter, and J. Pfeiffer, "Inferring hand-object configuration directly from tactile data," in *Electronically published proceeding of the Mobile Manipulation Workshop, ICRA*, Citeseer, 2010.

[17] L. Zhang, S. Lyu, and J. Trinkle, "A dynamic bayesian approach to real-time estimation and filtering in grasp acquisition," in *2013 IEEE International Conference on Robotics and Automation*, pp. 85–92, IEEE, 2013.

[18] M. Pfanne and M. Chalon, "Ekf-based in-hand object localization from joint position and torque measurements," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2464–2470, 2017.

[19] M. Chalon, J. Reinecke, and M. Pfanne, "Online in-hand object localization," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2977–2984, IEEE, 2013.

[20] G. KISSOUM and V. PERDEREAU, "Simultaneous tactile localization and reconstruction of an object during robotic manipulation," in *2021 20th International Conference on Advanced Robotics (ICAR)*, pp. 948–954, 2021.

[21] A. Aggarwal and F. Kirchner, "Object recognition and localization: The role of tactile sensors," *Sensors*, vol. 14, no. 2, pp. 3227–3266, 2014.

[22] O. Azulay, I. Ben-David, and A. Sintov, "Learning haptic-based object pose estimation for in-hand manipulation control with underactuated robotic hands," *IEEE Transactions on Haptics*, vol. 16, no. 1, pp. 73–85, 2023.

[23] D. Goudie and A. Galata, "3d hand-object pose estimation from depth with convolutional neural networks," in *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pp. 406–413, 2017.

[24] B. Doosti, S. Naha, M. Mirbagheri, and D. J. Crandall, "Hope-net: A graph-based model for hand-object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[25] F. Thomas, "Approaching dual quaternions from matrix algebra," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1037–1048, 2014.

[26] X. Wang and H. Zhu, "On the comparisons of unit dual quaternion and homogeneous transformation matrix," *Advances in Applied Clifford Algebras*, vol. 24, no. 1, pp. 213–229, 2014.

[27] H. Liu, X. Song, J. Bimbo, L. Seneviratne, and K. Althoefer, "Surface material recognition through haptic exploration using an intelligent contact sensing finger," in *2012 IEEE/RSJ international conference on intelligent robots and systems*, pp. 52–57, IEEE, 2012.

[28] H. Liu, X. Song, J. Bimbo, K. Althoefer, and L. Senerivatne, "Intelligent fingertip sensing for contact information identification," in *Advances in Reconfigurable Mechanisms and Robots I*, pp. 599–608, Springer, 2012.