

Robust dexterous telemanipulation following object-orientation commands

Andrés Montaña and Raúl Suárez

*Institute of Industrial and Control Engineering (IOC)
Universitat Politècnica de Catalunya (UPC), Barcelona, Spain.*

Abstract

Purpose This paper presents a procedure to change the orientation of a grasped object using dexterous manipulation. The manipulation is controlled by teleoperation in a very simple way with the commands introduced by an operator using a keyboard.

Design/methodology/approach The paper shows a teleoperation scheme, hand kinematics and a manipulation strategy to manipulate different objects using the Schunk Dexterous Hand (SDH2). A state machine is used to model the teleoperation actions and the system states. A virtual link is used to include the contact point on the the hand kinematics of the SDH2.

Findings Experiments were conducted to evaluate the proposed approach with different objects, varying the initial grasp configuration and the sequence of actions commanded by the operator.

Originality/value The proposed approach uses a shared telemanipulation schema to perform dexterous manipulation, in this schema the operator sends high level commands and a local system uses this information, jointly with tactile measurements and the current status of the system, to generate proper setpoints for the low-level control of the fingers, which may be a commercial close one. The main contribution of this work is the mentioned local system, simple enough for practical applications and robust enough to avoid object falls.

Keywords: Manipulation, Teleoperation, Tactile Sensors, Grasping

Article Type: Research paper

1 Introduction

Teleoperation of robots is a challenging subject in applications in which an operator takes decisions and the robots perform actions following the operator commands. Some application fields where the teleoperation is relevant are: handling hazardous material, telesurgery, underwater vehicles, space robots, mobile robots, among others (Hokayem and Spong, 2006; Hvilshoj et al., 2012). Object dexterous manipulation is a problem involved in these fields. A detailed discussion of the general problems related to teleoperation as well

as a description of typical applications was presented in (Basañez and Suárez, 2009).

Autonomy of the robotic system in teleoperation has been addressed following different approaches. On the one side, the operator has the full control of the movements and actions of the robot (a fully teleoperated system), and, on the other side, the control can also be shared between the remote operator and the local robot (Sheridan, 1992).

Talking about the teleoperation of robot arms and hands, different input interfaces has been used to teleoperate a robotic arm, like, for the case of the arms, trackers (Rosell et al., 2014) or wiimotes (Ciobanu et al., 2013), and for the case of the hands, gloves (Rosell et al., 2014; Kuklinski et al., 2014), multi touch interfaces (Toh et al., 2012), or video based systems (Ciobanu et al., 2013). One of the main problems in these approaches is the determination of an adequate mapping of the human pose and movements to those of the robot.

The manipulation of unknown objects has been addressed using different strategies. A control law to manipulate the object was introduced in (Tahara et al., 2010), that uses virtual object frame to change the pose of the object varying the triangular fingertip configuration of a three-fingered hand; however the lack of sensorial feedback limits the accuracy of the control approach. A composite position-force control scheme was presented in (Li et al., 2012). The relative position of the object with respect to the hand is changed following an input trajectory; the control scheme is evaluated in simulations introducing noise on the sensor measurements to simulate a real environment, however other grasp aspects, as the initial grasp configuration or the stability of the grasp, are not addressed. Following another approach, the shape of an unknown object was recognized using tactile information obtained during the object manipulation (Montaño and Suárez, 2013).

In the case of dexterous telemanipulation, the grasping element (gripper or robotic hand) can also be fully controlled by the operator or the control can be shared with the robot. In the first case, a mapping between the grasping element and the hand of the operator is required. Three mapping methods can be distinguished in the literature (Colasanto et al., 2013): joint-to-joint mapping, which is applied to anthropomorphic hands (Kyriakopoulos et al., 1997); pose mapping, which tries to find robot hand poses correlated with human hand poses (Kjellstrom et al., 2008); and point-to-point mapping, which maps the fingertip positions of the human hand to the fingertip positions of the robot hand (Peer et al., 2008). In the second case, the human operator may not have direct access to remote tactile and force information at the slave fingertips, but he/she can provide high level commands that are complemented with a low level automatic control that uses such information (Griffin et al., 2005). Tactile sensing has become a common feedback source in dexterous manipulation tasks (Tegin and Wikander, 2005).

In this work, which is an extended version of (Montaño and Suárez, 2015), we use a shared control scheme, i.e. the operator provides high level commands to the robotic hand to change the orientation of a grasped object, and the robotic system uses the tactile and kinematics information to control locally the forces and movements in order to avoid object falls. It must be remarked that the geometric model of the object is unknown and that, during the manipulation, the rotation limits are given by the friction constraints (supervised during the manipulation to avoid object falls) and by the kinematic constraints of the

fingers (joint limits). The proposed approach is intended to perform rotations of objects in a plane, like, for instance, to match the orientation of two pieces to do an assembly or to inspect an object. This type of rotation, jointly with planar sliding, is a quite frequent manipulation action in every-day and industrial tasks (Chang and Pollard, 2009; Toh et al., 2012). Changing the rotational axis using only the fingers is out of the scope of this work.

The remainder of the paper is arranged as follows. Section 2 introduces the approach overview. Section 3 presents the dexterous manipulation details and the motion strategy to avoid object falls. Experimental results are described in Section 4. Finally, Section 5 presents the summary and future work.

2 Approach description

The problem addressed in this work is the remote dexterous manipulation of unknown objects. The robotic system manipulates the object with dexterity following high level commands given by the operator and, with independence of these commands, the system autonomously cares about the robustness of the grasp in order to avoid objects falls.

The proposed approach uses a shared telemanipulation schema. The operator sends high level commands indicating the action to be done, and the local system uses this information, jointly with tactile measurements and the current status of the system, to generate proper setpoints for the low-level control of the finger joints (which may be a commercial close controller, as it is our case).

In the experimental part of this work, the operator introduces the commands in a very simple and intuitive way using a keyboard and the hand is fixed on a base over a table, but it can be assembled in a robotic arm as well, this is not of relevance for the proposed approach. The teleoperation of the robotic arm is out of the scope of this work. The manipulation task is focused on grasping an unknown object and change its orientation by rotating the object with the fingers, thus the operator can command the hand to close and open the fingers, and to turn the object clockwise or counter clockwise once it has been grasped. The commands are sent using four keys of a keyboard. In this work we consider that the absolute position of the object in the space can be controlled by the arm, and therefore only the orientation will be controlled by the fingers of the hand.

For the experimentation, we use the Schunk Dexterous Hand (SDH2) shown in Figure 1a. This is a three-finger hand (gripper) with seven active degrees of freedom (*dof*). The SDH2 has tactile sensors on the surface of the proximal and distal phalanges, i.e. there are six sensor pads; Figure 1b shows the sensor pad of a distal phalange (the fingertip). Two fingers of the hand can be rotated on their bases to work opposite to each other in the same plane, as shown in Figure 1c. Using these fingers it is possible to perform a prismatic precision grasp (MacKenzie and Iberall, 1994), which is comparable with a human grasp using the thumb and index fingers. These two fingers, working opposite to each other, are used in the experimental part of this work.

The manipulation algorithm is described by the state machine shown in Figure 2, which includes the following states:

S_{init} : The hand is in the initial configuration, ready to perform a grasp.

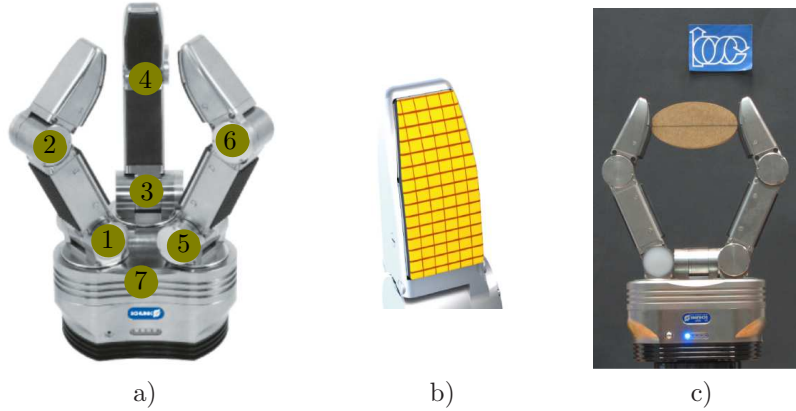


Figure 1: a) Schunk Dexterous Hand (SDH2) with the joints labels; b) detail of the sensor pad showing the texels on the fingertip; c) grasp configuration with two fingers working in an opposite way.

S_{close} : The fingers are closed until reaching a desired grasp force.

S_{open} : The fingers are opened to release the grasped object.

S_{grasp} : The object is grasped and the hand is waiting for a command.

S_{turnC} : The next configuration for a clockwise rotation of the grasped object is computed.

S_{turnCC} : The next configuration for a counterclockwise rotation of the grasped object is computed.

S_{move} : The hand executes the next configuration rotating the object.

The state transitions are determined by:

Keyboard signals K_c , K_o , K_{tc} and K_{tcc} : These are four signals generated by the operator using a standard keyboard, each signal is simply generated by pressing a predetermined key. The signals command the four possible actions during the teleoperation: close (K_c), open (K_o), rotate clockwise (K_{tc}) and rotate counterclockwise (K_{tcc}).

Force signal S_F : It is a binary signal that is activated when $F_k > F_d$, where F_k is the grasp force in the k -th step and F_d is a desired grasping force. This condition is reached when the hand has been closed and the object is in contact with the sensor pads.

Friction signal S_G : It is a binary signal that is activated when the friction constraints allow the grasp to firmly hold the object (these constraints are detailed in Section 3). The binary complement of S_G is represented as $\overline{S_G}$.

The state machine for the teleoperation starts in the state S_{init} , where the hand is waiting for the command K_c , to be introduced by the operator in order to close the fingers. When the command K_c is introduced, the system evolves to the state S_{close} . The system remains in the state S_{close} until the

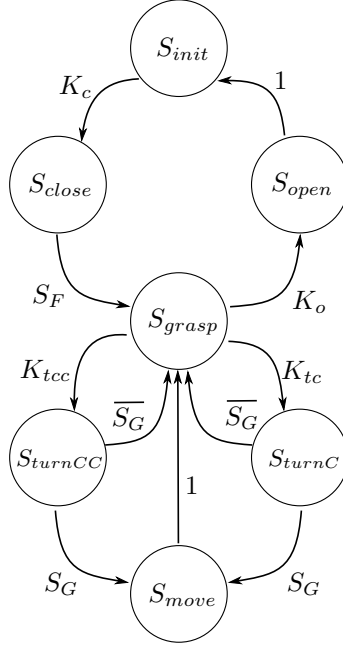


Figure 2: State machine for the teleoperation approach.

measured force on the sensor pads is greater than a desired grasp force, and the object has been actually grasped. Once the grasp is done and the proper force is detected, the system evolves to state S_{grasp} . In the state S_{grasp} the system is waiting for the commands K_{tc} , K_{tcc} or K_o , in order to do a clockwise rotation, a counterclockwise rotation or to open the fingers to release the object, respectively. The finger movements to rotate the object are computed in the states S_{turnC} and S_{turnCC} depending on the direction of rotation indicated by the operator.

In these states an autonomous dexterous manipulation algorithm (introduced later in Section 3) is used to compute the next finger positions (but no movement is executed yet). If reachable finger positions are found and the friction constraints are satisfied, then the system evolves to the state S_{move} , where the fingers are moved towards their new positions, and, once they are reached, the system comes back to S_{grasp} . If reachable finger positions are not found or they do not satisfy the friction constraints, then the system comes back to S_{grasp} without passing through S_{move} , i.e. without executing any movement. Note that the time-delay that may exist in the communication channel between the operator and the hand controller does not affect the robustness of the manipulation, which is locally assured by the system.

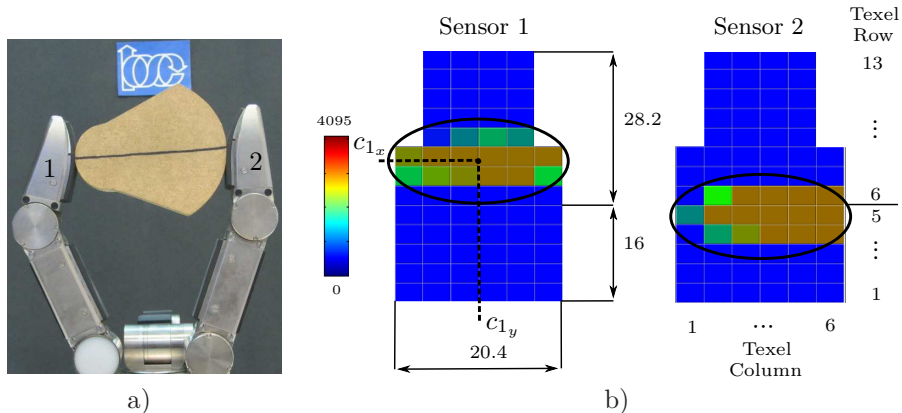


Figure 3: a) View of the two opposite fingers grasping an object; b) graphical representation of the tactile measurements showing the contact region on each sensor pad (distances are in millimeters), the bar in the left indicates the value of the measurement corresponding to each color.

3 Dexterous manipulation

3.1 Hand kinematics

The contact with the object produces a contact region on the tactile sensor, we consider the barycenter of this region as the contact point between the object and the finger and the average force over all the region as the contact force, as proposed in (Wörn and Haase, 2012) (see Figure 3).

Consider an absolute reference frame Σ_O located at the base of the finger 1 when the fingers have been rotated to work opposite to each other, and a reference frame Σ_{t_i} at the base of each fingertip i . The lateral view of the sensor surface on the fingertips is composed of two parts, a segment of a straight line and an arc of a circle with radius 60 mm centered at point $K = (33.5, -45)$ with respect to Σ_{t_i} (see Figure 4). The planar part of the sensor includes the rows of texels 1 to 5 with a length of 16 mm, and the curved part the rows of texels 6 to 13 with a length of 28.2 mm. The wide of the sensor is 6 texels with a length of 20.4 mm in the lower part and 4 texels with a length of 13.6 mm in the upper part, as illustrated in Figure 3b. The curved shape of the sensor is taken into account to compute the absolute positions of the contact point in Σ_O . When the tactile sensor in the fingertip i touches an object, it returns a value in the range $[0, 4095]$ indicating forces between zero and 3.6 N (the maximum measurable force before the sensor saturation) and the barycenter (c_{i_x}, c_{i_y}) of the contacted texels on the sensor pad. This barycenter is considered as the equivalent contact point. Applying grasping forces large enough, the possible errors in the contact point and force measurement are negligible in the proposed approach.

In this work, only the coordinate c_{i_x} of the contact point is relevant since the manipulation plane is orthogonal to the c_{i_y} -axis and therefore c_{i_y} does not add relevant information. Given c_{i_x} , the coordinates of the contact points P'_i

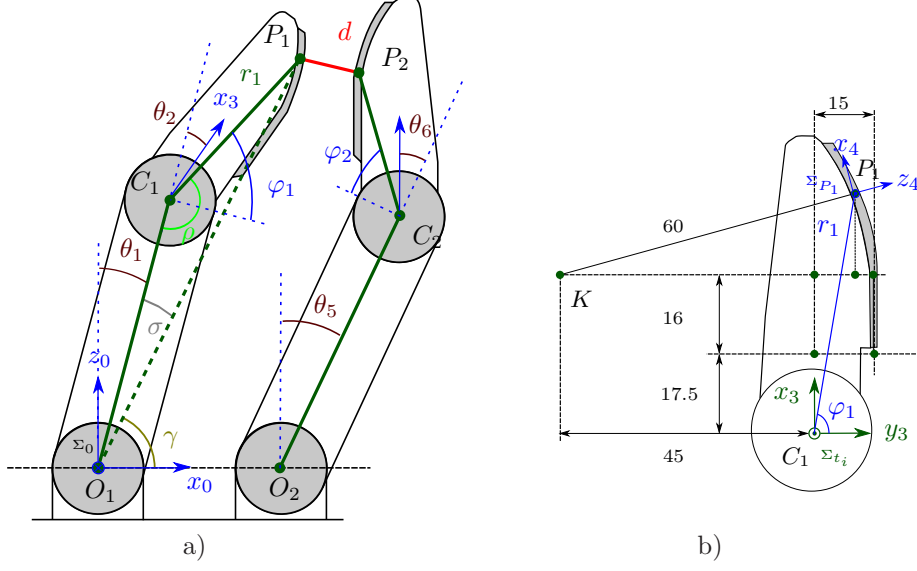


Figure 4: Dimensions and coordinate systems on SDH2 used for the kinematics analysis: a) lateral view of the two opposite fingers; b) details of the fingertip (distances are in millimeters), the shaded region represents the tactile pad.

on the fingertip i with respect to the reference system Σ_{t_i} are given by,

$$P'_{i_x} = \begin{cases} 17.5 + c_{i_x} & \text{if } c_{i_x} < 16 \\ 33.5 + 60 \sin\left(\frac{c_{i_x} - 16}{60}\right) & \text{if } c_{i_x} \geq 16 \end{cases} \quad (1)$$

$$P'_{i_y} = \begin{cases} 15 & \text{if } 17.5 < P'_{i_x} < 33.5 \\ -45 + \sqrt{60^2 - (P'_{i_x} - 33.5)^2} & \text{if } 33.5 \leq P'_{i_x} < 66.4 \end{cases} \quad (2)$$

A virtual link is used in order to include the contact point information in the hand kinematics. This virtual link adds an extra *dof* to each finger, and it is defined by the length of the segment r_i between the origin of Σ_{t_i} and P'_i , and by the angle φ_i between the y -axis of Σ_{t_i} and the segment r_i (see 3b). r_i and φ_i are expressed in frame Σ_{t_i} as,

$$r_i = \sqrt{P'^2_{i_x} + P'^2_{i_y}} \quad (3)$$

$$\varphi_i = \arctan\left(\frac{P'_{i_x}}{P'_{i_y}}\right) \quad (4)$$

The points P'_1 and P'_2 can be expressed with respect to Σ_O as P_1 and P_2 given by (see Figure 4a):

$$P_{1_x} = 86.5 \cos(\theta_1) - r_1 \cos(\theta_1 + \theta_2 + \varphi_1) \quad (5)$$

$$P_{1_z} = 86.5 \sin(\theta_1) - r_1 \sin(\theta_1 + \theta_2 + \varphi_1) \quad (6)$$

$$P_{2_x} = 66 - (86.5 \cos(\theta_5) - r_2 \cos(\theta_5 + \theta_6 + \varphi_2)) \quad (7)$$

$$P_{2z} = 86.5 \sin(\theta_5) - r_2 \sin(\theta_5 + \theta_6 + \varphi_2) \quad (8)$$

In order to be able to make the fingertips reaching desired points P_1 and P_2 , it is necessary to solve the inverse kinematics of the fingers. There are different ways of computing the inverse kinematics of the fingers, in this work we use a simple geometric analysis. We will develop here the equations to find the angles θ_1 and θ_2 for a given absolute position of P_1 on finger 1, the same reasoning is applied for P_2 on finger 2. Using the cosine law in the triangle defined by O_1 , C_1 and P_1 (see Figure 4), the angles ρ , σ and γ are given by,

$$\rho = \arccos\left(\frac{-|\overline{O_1P_1}|^2 + |\overline{O_1C_1}|^2 + |\overline{C_1P_1}|^2}{2|\overline{O_1C_1}||\overline{C_1P_1}|}\right) \quad (9)$$

$$\sigma = \arccos\left(\frac{-|\overline{C_1P_1}|^2 + |\overline{O_1C_1}|^2 + |\overline{O_1P_1}|^2}{2|\overline{O_1C_1}||\overline{O_1P_1}|}\right) \quad (10)$$

$$\gamma = \arctan\left(\frac{P_{1z}}{P_{1x}}\right) \quad (11)$$

There are two possible solutions for the inverse kinematics for P_1 , however only one of them is valid considering the geometric constraints imposed by the manipulation problem that must allow that fingers work opposed to each other. Thus, valid configurations are obtained only for values of θ_2 satisfying $\theta_2 > \varphi_1 - \pi/2$. Then, the values for θ_1 and θ_2 are given by,

$$\theta_1 = -\sigma - \gamma + \text{sign}(P_{1x})\pi/2 \quad (12)$$

$$\theta_2 = \rho - \pi/2 - \varphi_1 \quad (13)$$

where

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \quad (14)$$

3.2 Manipulation strategy

The manipulation strategy must be able to rotate an unknown object using as input information the tactile and the kinematic information of the hand. The strategy have to determine a sequence of finger movements to perform the desired rotation of the object (i.e. determine a sequence of proper positions of the contact points), while trying to keep the grasping force F_k within a desired range (F_{\min}, F_{\max}). F_k is computed as the average of the contact forces F_{1k} and F_{2k} measured, respectively, at each fingertip, minimizing in this way potential measurement errors, i.e.

$$F_k = \frac{F_{1k} + F_{2k}}{2} \quad (15)$$

The contact force is used to adjust, in the step $k+1$, the distance d_k between the contact points P_1 and P_2 existing at step k as,

$$d_{k+1} = d_k + \Delta d \quad (16)$$

with Δd being a function of F_k according to the follow relationship,

$$\Delta d = \begin{cases} 0 & \text{if } F_{\min} < F_k < F_{\max} \\ +\lambda & \text{if } F_k \leq F_{\min} \\ -\lambda & \text{if } F_k \geq F_{\max} \end{cases} \quad (17)$$

where the constant values F_{\min} , F_{\max} and λ are empirically determined based on the sensors response, and the distance d_k between contact points P_1 and P_2 is directly the Euclidean distance given by,

$$d_k = \sqrt{(P_{1x_k} - P_{2x_k})^2 + (P_{1y_k} - P_{2y_k})^2} \quad (18)$$

The grasping force F_k , the position of the contact points P_1 and P_2 , the distance between contact points d_k and d_{k+1} are computed in state S_{grasp} .

In order to compute the expected positions of the contact points in the next step, we consider as hypothesis that the fingers are moved over a circular path whose diameter is given by the distance d_{k+1} , as illustrated in Figure 5. This is actually a general initial approximation, since the manipulated object is unknown. The fingers are moved to $P_{1_{k+1}}$ and $P_{2_{k+1}}$ and the distance d_k is actualized to d_{k+1} at same time. Let ϕ be the object orientation. The orientation of the object resulting from the initial grasp is considered as the reference orientation, i.e. $\phi = 0$. It must be remarked that there are not external measurements of the object orientation as, for instance, using vision systems. The variation of $\phi = 0$ are computed using uniquely proprioceptive information.

The expected contact points $P_{1_{k+1}}$ and $P_{2_{k+1}}$, are computed as,

$$P_{1x_{k+1}} = C_{x_k} - (d_{k+1}/2) \cos(\phi + \Delta\phi) \quad (19)$$

$$P_{1z_{k+1}} = C_{z_k} - (d_{k+1}/2) \sin(\phi + \Delta\phi) \quad (20)$$

$$P_{2x_{k+1}} = C_{x_k} + (d_{k+1}/2) \cos(\phi + \Delta\phi) \quad (21)$$

$$P_{2z_{k+1}} = C_{z_k} + (d_{k+1}/2) \sin(\phi + \Delta\phi) \quad (22)$$

where $\Delta\phi$ is chosen positive to turn the object clockwise or negative to turn the object counterclockwise. $\Delta\phi$ is chosen small enough to assure small movements of the object on each manipulation step. The point C_k is the center of the circular path followed by the fingers, and it is given by,

$$C_{x_k} = \frac{P_{2x_k} - P_{1x_k}}{2} + P_{1x_k} \quad (23)$$

$$C_{z_k} = \frac{P_{2z_k} - P_{1z_k}}{2} + P_{1z_k} \quad (24)$$

The expected contact points $P_{1_{k+1}}$ and $P_{2_{k+1}}$ are computed in states S_{turnC} and S_{turnCC} depending on the command introduced by the operator to rotate the object clockwise or counterclockwise, respectively.

In order to avoid sliding, each force applied on the object must lie inside the friction cone centered at the direction normal to the sensor surface at the contact point. When this condition is satisfied, the binary signal S_G is activated allowing a transition in the manipulation state machine to the state S_{move} , where the finger motions are executed. When S_G is not activated (i.e. $\overline{S_G}$ is activated) the state changes to S_{grasp} and the system waits for new commands.

A planar grasp with two frictional contact points is force-closure when the segment connecting the contact points lies inside the friction cone at both contact points, as shown in Figure 6. Using the Coulomb friction model, the friction cone is given by $\alpha = \arctan \mu$, with μ being the friction coefficient.

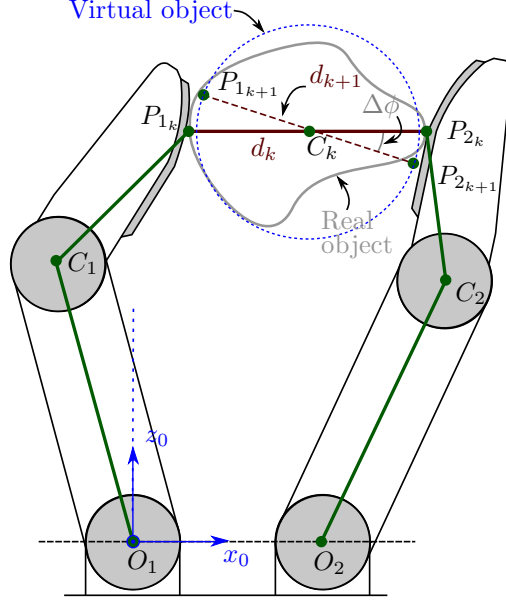


Figure 5: Two-finger model used to control the object orientation.

Any applied force that belongs to the friction cone will not produce slippage, therefore the angle β_i , $i = 1, 2$, between the normal direction at each contact point and the segment between the two contact points must satisfy $\beta_i < \alpha$. This condition can be expressed as,

$$\pi/2 - \alpha < \omega_i < \pi/2 + \alpha \quad (25)$$

where ω_i , $i = 1, 2$, is computed for contact point i as,

$$\omega_1 = \arccos\left(\frac{-|C_1P_2|^2 + r_1^2 + |P_1P_2|^2}{2r_1|P_1P_2|}\right) - \theta_2 - \pi/2 + \varphi_1 \quad (26)$$

$$\omega_2 = \arccos\left(\frac{-|C_2P_1|^2 + r_2^2 + |P_1P_2|^2}{2r_2|P_1P_2|}\right) - \theta_6 - \pi/2 + \varphi_2 \quad (27)$$

The points C_1 and C_2 , the distances r_1 and r_2 , and the angles φ_1 and φ_2 are computed using the kinematics of the hand and the information of the contact points. A complete description of the kinematics of the hand SDH2 can be found in (Montaño and Suárez, 2014).

The reachable limits of the object orientation depend on the friction constraints and the kinematic constraints of the fingers, which in turn depend on the shape of the object being manipulated.

4 Experimental results

The described approach has been fully implemented using C++ for unknown object telemanipulation with the SDH2 hand. Figure 7 shows the set of objects used in the examples below. When an object is located between the two opposed fingers of the SDH2, the operator generates, clicking the proper key in the

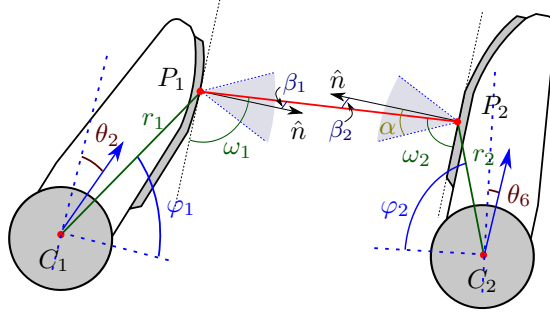


Figure 6: Detail of the fingertips and angles used to compute the friction constraints.

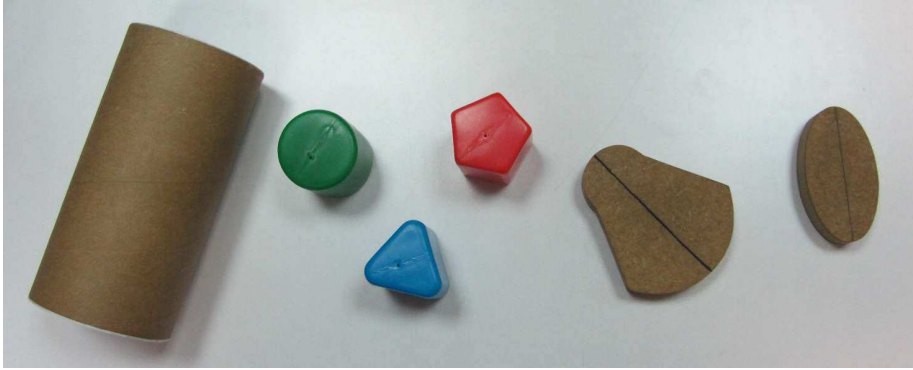


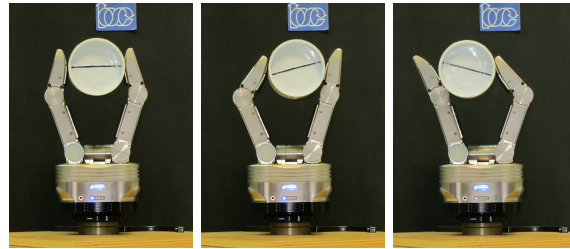
Figure 7: Set of manipulated objects.

keyboard, the command to close the fingers until the measured force reaches the desired value $F_d = 2$ N. Note that the initial contact points are unknown, i.e. the initial grasp configuration changes at each execution of the experiment. After this, the operator can manipulate the object, rotating it clockwise or counterclockwise by means of a simple teleoperation.

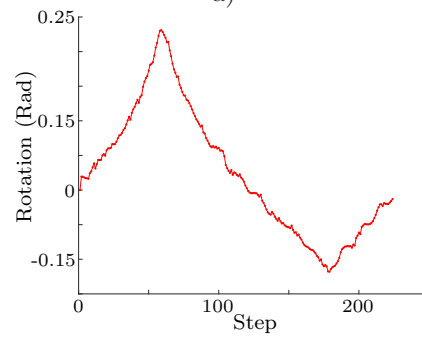
The material of the sensor pads is rubber and the material of the objects is wood, cardboard or plastic, thus we consider a worst case friction coefficient $\mu = 0.4$, which is lower than the friction coefficient between rubber and wood $\mu = 0.7$, rubber and cardboard $\mu = 0.5$, and rubber and plastic $\mu = 0.6$ (Kutz, 2015). The constant λ to adjust the distance d_k is set to 1 mm. The orientation variation in each manipulation step was set to $\Delta\phi = 0.25$ degrees.

As application examples of the proposed approach, each object was grasped and rotated as much as possible in both senses. Figures 8-13 show, for each example, the initial configuration of the grasped object, the configuration when the limits of rotation are reached in counterclockwise and clockwise directions, and a graphical representation of the variation of the object orientation for each step during the telemanipulation process. It is worth reminding here that the variation of the orientation is computed using only the tactile information and the hand kinematics, without using any external measurement.

The previous examples have shown the evolution of the object rotation per step, the following example details the evolution as a function of time.

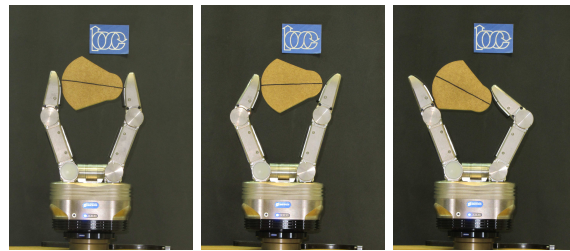


a)

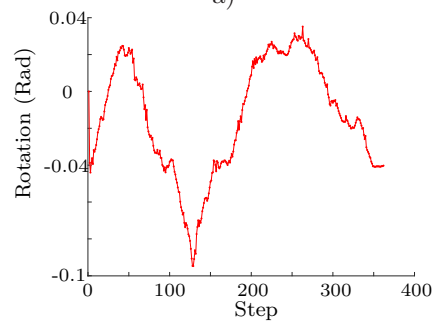


b)

Figure 8: Example 1: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.



a)



b)

Figure 9: Example 2: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.

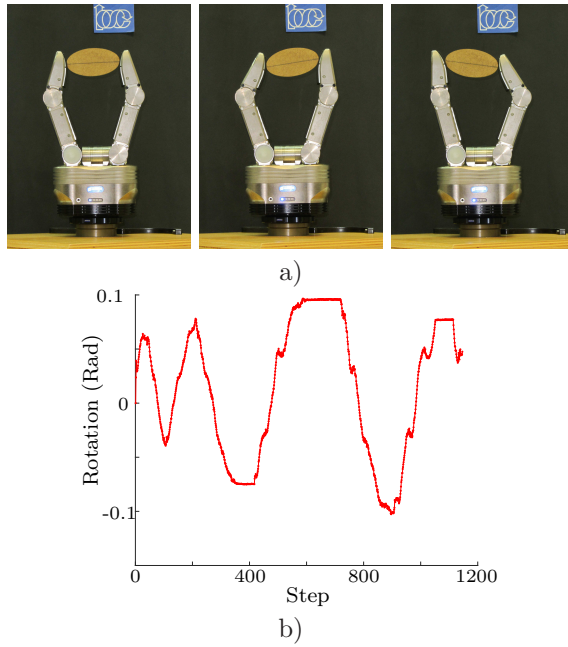


Figure 10: Example 3: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.

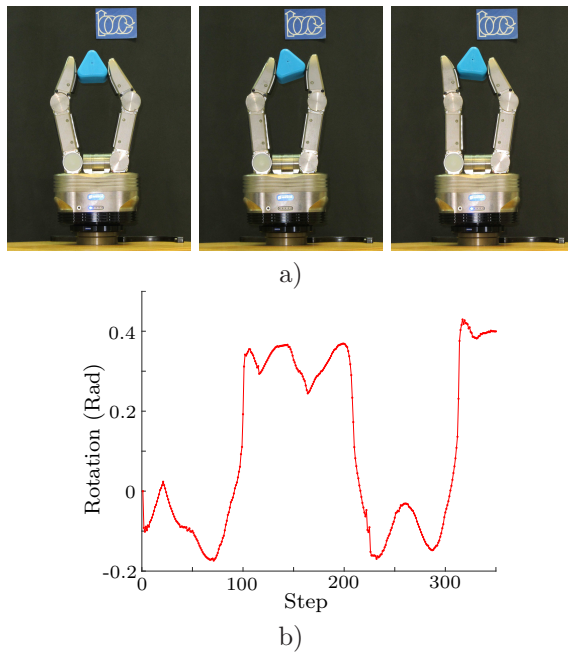
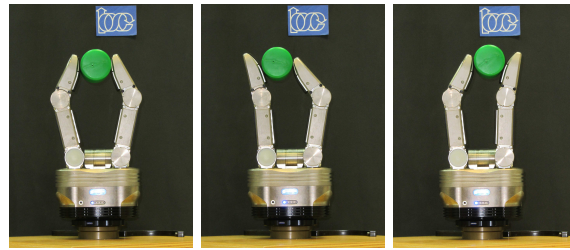
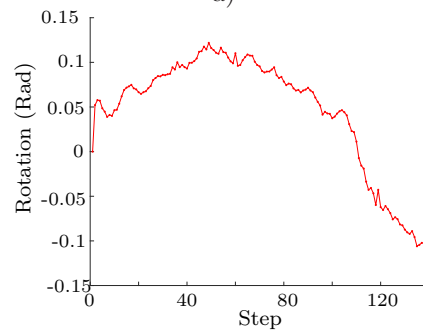


Figure 11: Example 4: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.

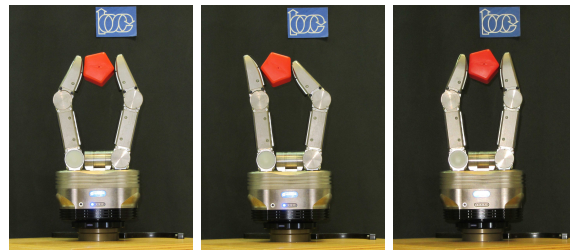


a)

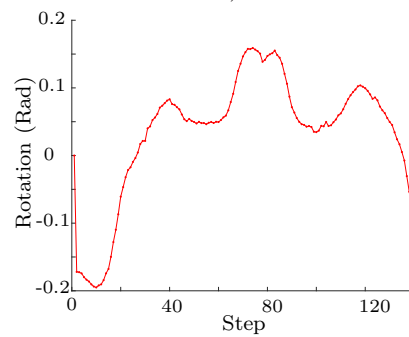


b)

Figure 12: Example 5: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.



a)



b)

Figure 13: Example 6: a) initial configuration (left), limits of rotation reached in counterclockwise direction (center) and in clockwise direction (right); b) variation of the object orientation for each step during the telemanipulation.

The commands introduced by the operator to rotate the object are shown in Figure 14a and Figure 14b. Figure 14c and Figure 14d show the variation of the joints for each step and over time during the manipulation, respectively, and Figure 14e and Figure 14f do so for the object orientation as well. The time elapse between the reception of two commands in the remote system is shown in Figure 14g, it includes the system delays and the operator response. A snapshot of the moment when the limit of rotation was reached in clockwise direction is shown in Figure 14h. In this case the experiment lasted for 270 steps with a duration of 306.1 seconds. Note that the clockwise object rotation reached a maximum value -14.88° in the step 175 and remains there until step 186, even when the operator is still sending rotation commands in that sense. The reason for this is that the grasps is on the limit of the friction constraints and therefore the system did not go further in order to assure the grasp robustness and avoid the object fall. Between steps 49 and 71 and between steps 234 and 253 the operator introduced commands to rotate the object alternately in both senses, and the system responds correctly.

In order to show that the delays in the network do not affect the robustness of the manipulation we include another example in which high and random time delays were included in the communication channel between the local machine with the input interface and the remote machine with the robotic hand. The delays were randomly generated between 0 and 1.5 seconds, in both senses of the communication. In this example the aim of the telemanipulation was to rotate the object to -5° with respect to the initial orientation at the grasping time, and the feedback received by the operator in the local station is only the object orientation, no visual feedback was allowed in this case. We asked an untrained person without knowledge of the system response to execute the task. The commands introduced by the operator are shown in Figure 14a, per step, and in Figure 14b, over time. The task was accomplished after 57 steps with a duration of 114.5 seconds. The random delays introduced per step in both senses of the communication channel are shown in Figure 14c and Figure 14d. The joint values in each step and over time are shown in Figure 14e and Figure 14f, respectively, and the orientation of the object, per step and over time, are shown in Figure 14g and in Figure 14h. Note that despite the oscillation of the object orientation around -5° , the object manipulation is always robust and stable. A video of these examples is available at <https://goo.gl/3QECq1>

5 Summary and future work

This work has proposed a telemanipulation method to rotate unknown objects using two fingers and tactile information. The objects are initially grasped in any orientation and the telemanipulation allows to change their orientations by rotating them in a plane according to the commands given by an operator, which in this implementation is done using a keyboard.

The approach is based on a shared control scheme, where the operator provides high level commands without receiving direct feedback about the forces applied on the object and the system autonomously controls the hand movements avoiding object falls. The experimental results show that the approach is effective to manipulate different types of objects.

A natural extension of this work is the use of three fingers in the manipulation

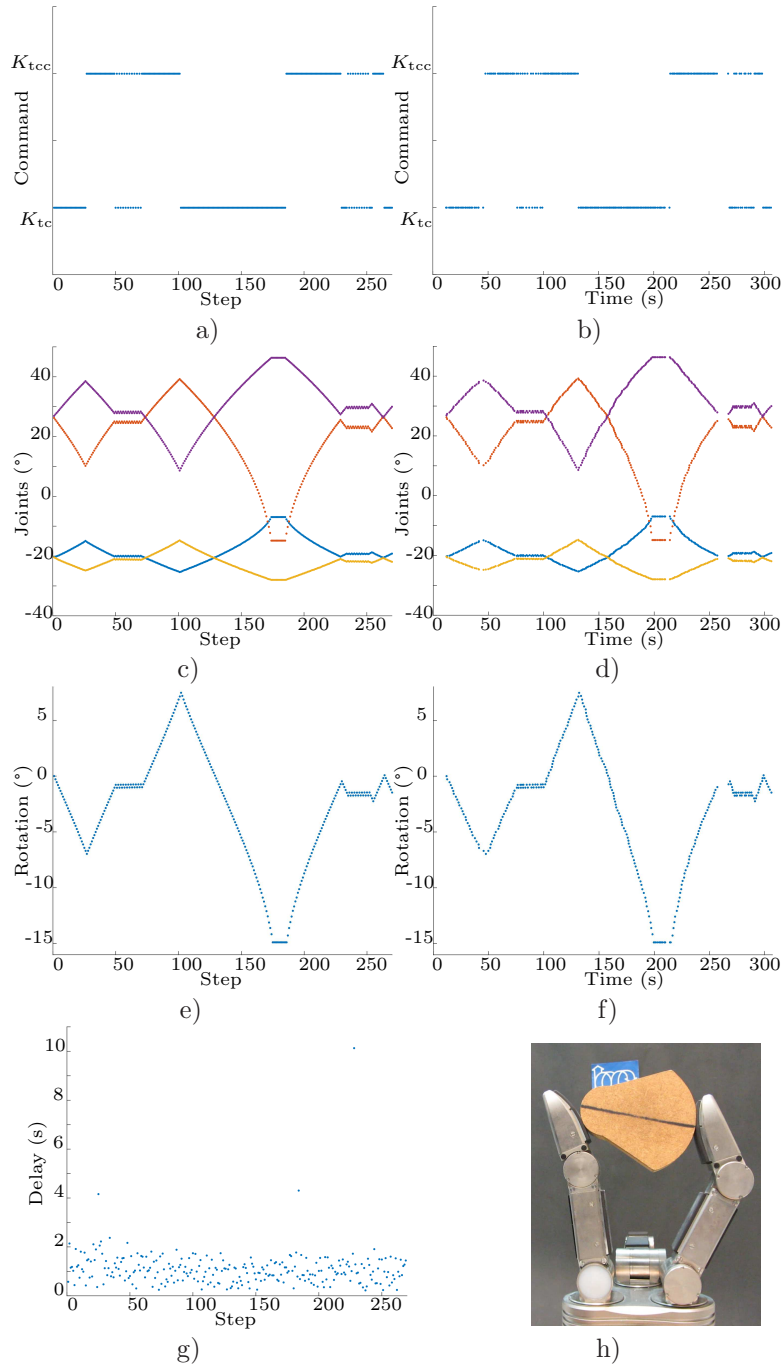


Figure 14: Example 7: a) introduced commands in each step; b) introduced commands over time; c) variation of the joints in each step; d) variation of the joints over time; e) variation of the object orientation in each step; f) variation of the object orientation over time; g) time elapse between the reception of two commands in the remote station; h) limit of rotation reached in clockwise direction (steps 175 to 186).

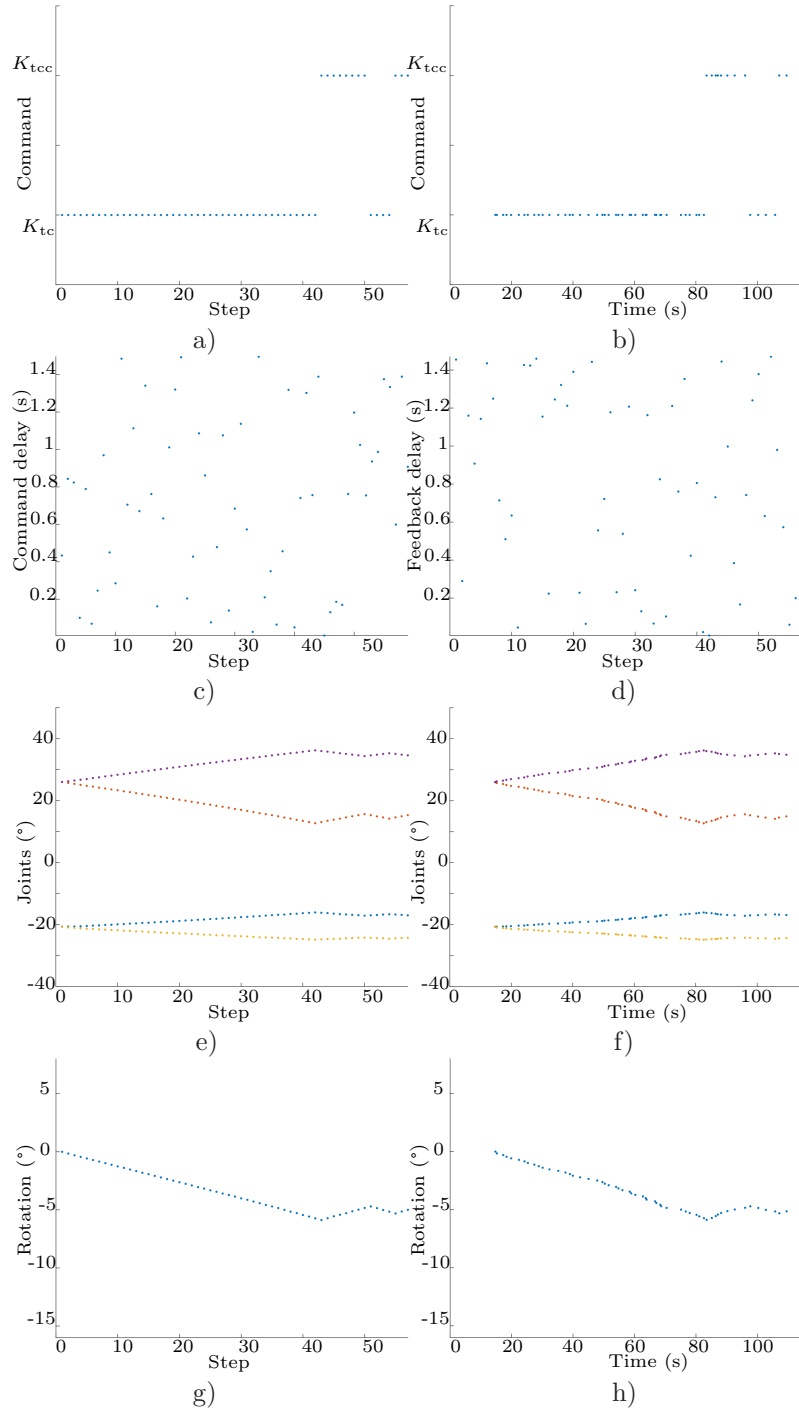


Figure 15: Example 8: a) introduced commands in each step; b) introduced commands over time; c) delays introduced per step; d) delays introduced over time; e) variation of the joints in each step; f) variation of the joints over time; g) variation of the object orientation in each step; h) variation of the object orientation over time.

process, which would allow to consider other motion strategies addressing the 3-D rotational problem and the use of a system to recognize human-hand movements that could be used to generate the input commands to the telemanipulation system.

Acknowledgment

This work was partially supported by the Spanish Government through projects DPI2013-40882-P, DPI2014-57757-R and DPI2016-80077-R.

References

- Basañez, L. and Suárez, R. (2009), Teleoperation, *in* S. Y. Nof, ed., ‘Springer Handbook of Automation’, Springer-Verlag, pp. 449–468.
- Chang, L. Y. and Pollard, N. S. (2009), Video survey of pre-grasp interactions in natural hand activities, *in* ‘RSS Workshop: Understanding the Human Hand for Advancing Robotic Manipulation’, Carnegie Mellon University, Pittsburgh, pp. 1–2.
- Ciobanu, V., Popescu, N., Petrescu, A. and Noeske, M. (2013), Robot telemanipulation system, *in* ‘Int. Conf. on System Theory, Control and Computing (ICSTCC)’, pp. 681–686.
- Colasanto, L., Suárez, R. and Rosell, J. (2013), ‘Hybrid Mapping for the Assistance of Teleoperated Grasping Tasks’, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **43**(2), 390–401.
- Griffin, W. B., Provancher, W. R. and Cutkosky, M. R. (2005), ‘Feedback Strategies for Telemanipulation with Shared Control of Object Handling Forces’, *Presence: Teleoperators and Virtual Environments* **14**(6), 720–731.
- Hokayem, P. F. and Spong, M. W. (2006), ‘Bilateral teleoperation: An historical survey’, *Automatica* **42**(12), 2035–2057.
- Hvilshoj, M., Bogh, S., Nielsen, O. S. and Madsen, O. (2012), ‘Autonomous industrial mobile manipulation (AIMM): past, present and future’, *Industrial Robot: An International Journal* **39**(2), 120–135.
- Kjellstrom, H., Romero, J. and Kragic, D. (2008), Visual recognition of grasps for human-to-robot mapping, *in* ‘Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems’, pp. 3192–3199.
- Kuklinski, K., Fischer, K., Marhenke, I., Kirstein, F., aus der Wieschen, M. V., Solvason, D., Kruger, N. and Savarimuthu, T. R. (2014), Teleoperation for learning by demonstration: Data glove versus object manipulation for intuitive robot control, *in* ‘Int. Cong. on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)’, pp. 346–351.
- Kutz, M. (2015), *Mechanical engineers’ handbook*, Wiley.

- Kyriakopoulos, K. J., Van Riper, J., Zink, A. and Stephanou, H. E. (1997), ‘Kinematic analysis and position/force control of the Anthrobot dextrous hand’, *IEEE Transactions on Systems, Man, and Cybernetics: Cybernetics* **27**(1), 95–104.
- Li, Q., Haschke, R., Ritter, H. and Bolder, B. (2012), Towards Manipulation of Unknown Objects, in ‘IFAC Symposium on Robot Control’, IFAC.
- MacKenzie, C. L. and Iberall, T. (1994), *The Grasping Hand*, Vol. 104, Elsevier.
- Montaño, A. and Suárez, R. (2013), Object shape reconstruction based on the object manipulation, in ‘Proceedings of IEEE International Conference on Advanced Robotics’, pp. 1–6.
- Montaño, A. and Suárez, R. (2014), Getting comfortable hand configurations while manipulating an object, in ‘Proceedings of IEEE International Conference on Emerging Technologies and Factory Automation’.
- Montaño, A. and Suárez, R. (2015), Unknown object manipulation based on tactile information, in ‘Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems’, pp. 5642–5647.
- Peer, A., Einenkel, S. and Buss, M. (2008), Multi-fingered telemanipulation - mapping of a human hand to a three finger gripper, in ‘IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)’, pp. 465–470.
- Rosell, J., Suárez, R. and Pérez, A. (2014), Safe Teleoperation of a Dual Hand-Arm Robotic System, in M. A. Armada, A. Sanfeliu and M. Ferre, eds, ‘ROBOT2013: First Iberian Robotics Conference’, Vol. 253 of *Advances in Intelligent Systems and Computing*, Springer International Publishing, pp. 615–630.
- Sheridan, T. B. (1992), *Telerobotics, Automation, and Human Supervisory Control*, MIT Press, Cambridge, MA, USA.
- Tahara, K., Arimoto, S. and Yoshida, M. (2010), Dynamic object manipulation using a virtual frame by a triple soft-fingered robotic hand, in ‘Proceedings of IEEE International Conference on Robotics and Automation’, pp. 4322–4327.
- Tegin, J. and Wikander, J. (2005), ‘Tactile sensing in intelligent robotic manipulation a review’, *Industrial Robot: An International Journal* **32**(1), 64–70.
- Toh, Y. P., Huang, S., Lin, J., Bajzek, M., Zeglin, G. and Pollard, N. S. (2012), Dexterous telemanipulation with a multi-touch interface, in ‘Proceedings of IEEE-RAS International Conference on Humanoid Robots’, pp. 270–277.
- Wörn, H. and Haase, T. (2012), Force approximation and tactile sensor prediction for reactive grasping, in ‘Proceedings of World Automation Congress’, pp. 1–6.