In-hand Recognition and Manipulation of Elastic Objects Using a Servo-Tactile Control Strategy

A. Delgado[1], C.A. Jara[2], F. Torres[2]

[1]University Institute of Computing Research, University of Alicante. Carretera San Vicente del Raspeig s/n – 03690. San Vicente del Raspeig, Alicante, Spain

[2]Physics, Systems Engineering and Signal Theory Department, University of Alicante. Carretera San Vicente del Raspeig s/n – 03690. San Vicente del Raspeig, Alicante, Spain.

{angel.delgado,carlos.jara,fernando.torres}@ua.es
Tel.: +34 965903400 2440 – Fax:+34 965903464

Abstract. Grasping and manipulating objects with robotic hands depend largely on the features of the object to be used. Especially, features such as softness and deformability are crucial to take into account during the manipulation tasks. Indeed, positions of the fingers and forces to be applied by the robot hand when manipulating an object must be adapted to the caused deformation. For unknown objects, a previous recognition stage is usually needed to get the features of the object, and the manipulation strategies must be adapted depending on that recognition stage. To obtain a precise control in the manipulation task, a complex object model is usually needed and performed, for example using the Finite Element Method. However, these models require a complete discretization of the object and they are time-consuming for the performance of the manipulation tasks. For that reason, in this paper a new control strategy, based on a minimal spring model of the objects, is presented and used for the control of the robot hand. This paper also presents an adaptable tactile-servo control scheme that can be used in in-hand manipulation tasks of deformable objects. Tactile control is based on achieving and maintaining a force value at the contact points which changes according to the object softness, a feature estimated in an initial recognition stage.

Keywords: Deformable object · Grasping · In-Hand Manipulation · Tactile Servoing

1. Introduction

In the field of robotic-hand manipulation, the study of how to grasp and manipulate deformable objects is still a challenging topic and has been studied during the last years in many contexts, ranging from industry to surgery [1]. Nowadays, grasping and manipulating any type of elastic or deformable object by a service, autonomous or assistive robot, are considered as extra and important capabilities.

Robot hand manipulation of rigid bodies is based on the principles of rigid body motion, where the use of the grasp matrix and fingers' Jacobian lead to mathematically model and track the behavior of the whole system [2]. In this case, control laws can be developed taking into account the well-known dynamics of rigid objects. In contrast, deformable objects are very difficult to model and simulate, and the robot hand-object system cannot be controlled using the same laws as rigid ones. This is mainly because deformable objects' dynamics depends on a large and complex set of parameters, such as deformation, softness and friction.

It is possible to obtain a perfect model of the deformable object using some of the current approaches, such as mass-spring model [3], mesh-less models [4] and the more general Finite Element Method (FEM) [5]. Although these are precise techniques, most of them require a fine discretization of the object and hence, they are time-consuming for on-line manipulation performance. Given these difficulties in this research field, several approaches have been developed to explore manipulation of deformable objects using a simple model or without explicitly modelling the objects [6][7][8]. In this last sense, the use of tactile data to obtain information about the deformability and softness, and also how to use this information as a reference to control the motion of the fingers is a key issue. This field of study is inspired on the scheme of human grasping and manipulation, which essentially depends on tactile sensation rather than vision or proprioception.

Force Closure Grasping (FCG) is considered a key point in order to perform a correct dexterous manipulation of rigid bodies. Regarding FCG, it is possible to determine the quality of a grasp for a rigid object by its shape and dimensions and the information of the robotic hand. In contrast, when the objects are deformable or elastic, it is necessary to incorporate a contact tactile-based readjustment algorithm on the grasp points to ensure the

correct operation of the FCG [9][10] and it is very difficult to ensure a FCG. This paper presents an approach based on tactile data in order to obtain important features of the deformable object and to manipulate it in a secure way.

Usually, grasping and manipulating a deformable object needs a multisensory system [11][12], mainly to control the state of the object on each time step. Thus, internal information from the hand (kinematics and dynamics), visual and tactile data are normally used together to control the whole system. In many works, only internal information from the manipulator and tactile data is used to control the manipulation task. In this cases, an accurate tactile sensor is needed to get reliable feedback of how the contacts interact with the objects [13][14]15][16]. Using tactile data, servo controllers can be used to maintain a target value of magnitude and position of the applied forces [17]. However, many of them are based on learning algorithms without taking into account objects' features or these approaches use complex models for the objects.

In this paper, a new tactile based grasp and manipulation control system for isotropic elastic objects is described. The main objective of this approach is to grasp and in-hand manipulate deformable objects using tactile data. A vision system is employed only to locate the object in the world, and may be required by the general system when contact is lost. To do that, two main strategies used by the system are described. First, a recognition of the object to be grasped is carried out in order to obtain an estimation of its softness, i.e the relation of stress and strain in the object (Young's modulus). Second, a tactile-servo framework is employed to maintain contacts with the object and pressure values. This tactile-servo is controlled by a task manager that sends reconfigurations to the finger positions and exerted forces. Once the object is detected and the hand is positioned in the initial configuration, the control and recognition system presented in this paper can work without depending on a vision system, so the processes are more agile than those that use visual processing. Besides, the goal of these strategies is to get a reliable manipulation control. Although a complete model of the object is not used, controlling the contact points as well as the exerted forces allows a reliable control of the movements and deformations in the object.

2. System Description

The system is basically composed by the Shadow robot hand [18] and a Kinect sensor. Besides, a tactile sensor, the Tekscan grip sensor [19], is attached to the robot hand in order to obtain force and pressure data from fingers. The Kinect sensor is employed to detect and recognize objects in the world. The Shadow hand is mounted on a structure that allows displacements in any direction, and angular movements (roll, pitch and yaw). Only movements of the hand are considered to grasp objects (see Figure 1).

The Shadow hand is a motor actuated robotic hand with five fingers and 20 degrees of freedom (2 on the wrist, 5 on the thumb, 3 on the first, middle and ring finger, and 4 on the little finger). To make it similar to a human hand, the nearest joint to the finger end effector on each finger except the thumb, is a coupled joint. This is, the hand has 24 degrees of freedom, but only 20 are controlled. Each of the joints is controlled both with position or force control. A kinematic model of the hand is used to compute both forward and inverse kinematic solutions. To model and simulate movements of the hand, the Gazebo simulator and MoveIt software in ROS are used. However, due to the deformability of the objects, these software platforms cannot give realistic simulations of how the system hand-object will behave. Thus, real experimental are necessary to show the veracity of the presented approach.

Fig. 1. System used for the experiments: the Shadow robot hand with the Tekscan Grip sensor, and the Kinect sensor.

Regarding the Tekscan grip sensor, is a hand shaped sensor that can be attached to human or robotic hands. It is used to extract a pressure and/or force map of the whole hand. The sensor is divided in 18 regions that cover the palm of the hand and its fingers. Each region is composed by an array of cells with different sizes depending on the position in the hand. For example, the region situated in the finger tips is an array of 4 by 4 element cells. For this paper, only the regions of the finger-tips are used, which are the common parts of the hand to use when a grasp has to be done. The region of each fingertip covers 1'7 by 1'7 cm. The response in pressure and force is calibrated in an initial stage before the sensor is used in real experiments. After this calibration, the response of each region is defined by a calibration curve. This hand was calibrated so that it has a linear response on each region between 0 and 7 newton. The sensor is connected to a computer with a local Ethernet connection, and the sampling rate is about 850 Hz. The sensor values are stored in array structures which store the whole pressure map, pressure map for a single finger, or pressure map for the fingertips.

1. Object Recognition Stage

In order to make this approach adaptable to different deformable objects with different degrees of softness, a recognition stage is performed during the first grasp of the object. The contact points are selected after the object is detected by the vision system, and they are chosen using different quality measures. Basically, the vision system obtains the position of the object and an approximated shape using 3D recognition techniques [20].

The idea of the recognition stage is to get a relative value of how the object is deformed relatively to the finger positions and applied forces. For elastic objects, Hooke's law may be used, because it relates stress and strain on the object. After getting values for pressure and strain on each finger, a relative stiffness degree is obtained, which is an approximation of the Young's modulus. For isotropic bodies, the Young's modulus should be the same independently of the points of contact. Figure 2 shows the algorithm which is executed during the recognition stage to obtain the values of stress and strain.

Fig. 2. Diagram of the tactile recognition algorithm.

As shown in the algorithm, the recognition stage starts at the configuration of the hand in which the fingertips are positioned in the selected contact points. The recognition stage is based on the human process to recognize softness of an object, in which fingers are moved towards the body's geometric center, and the softness estimation is relative to how much we can move the fingers, and how much pressure we feel in the fingers when the body tends physically to recover its initial configuration.

This stage is divided in two parts, a finger position adjustment, and a force readjustment. Small position steps are used to get not large deformations and get a linear response from stress to strain. After a minimum force value is obtained at one of the contact points, each finger force is readjusted until a maximum force value is obtained. This readjustment is executed only if the value of pressure is not zero (this is, the contact with the object still exists).

The minimum and maximum values are set after experimentation with different deformable objects. Minimum value can be understood as the minimum force needed to safely grasp an object, and maximum value is a force value in which the fingers are applying high pressures to the body, and could damage it.

The algorithm is executed for each of the fingers which grasp the object, and then an approximation of the Young's modulus for each finger is obtained as the coefficient between stress (final applied pressure) and strain. The strain is measured as the variation of the distance between each finger that has been moved, and the center of the object. To make most robust the estimation of the Young's modulus with this algorithm, for isotropic objects, the value of the finger with the most lineal relation between stress and strain is utilized (for the same body, each finger could give different values due to different response of the used sensors and possible errors). Fingers with pressure equal to zero are not considered, as this means a loss of contact in the sensor area.

After the recognition stage finishes, a minimal model of the object is obtained (see Figure 3). This model is based on a spring located between each contact point and the geometric center.

Fig. 3. Spring model of an object with five contact points, defined by 5 springs S_i. Each spring is limited by the initial positions of the contact points $P_0(C_i)$, and the object's center position $P_0(O)$. The change in length is determined by the final contact positions $P_n(C_i)$.

Each spring *Si* of the model is defined by an approximation of the factor of stiffness (Young's modulus) $\tilde{K}_{Si}$. This value is obtained as the relation between the applied stress on each finger σ_i, given by the tactile sensor, and the strain ϵ_i. This strain value is computed as the ratio between distances to object center $P_0(O)$ from the initial contact point $P_0(C_i)$ and final contact point $P_n(C_i)$. Assuming that the position of the center of the object keeps constant during the recognition stage, the variation of the distance is obtained directly from the kinematics of the hand, and hence, the position of the link of each fingertip. Only the fingers for which there are forces not equal to zero are considered (zero force values are considered as a loss of contact between finger and object).

$$\epsilon_i = \frac{distance_{object-contact_n} - distance_{object-contact_0}}{distance_{object-contact_0}} \tag{1}$$

$$K^{\sim}{}_{s_i} = \frac{\sigma_i}{\epsilon_i} \tag{2}$$

4. Tactile Servo Controller for In-Hand Manipulation

For in-hand manipulation tasks which involve only finger movements, a task planner module controls the position and forces applied by each finger. An in-hand manipulation task could require adding or removing fingers in contact, change position and contact, or change applied force on a current contact. To control each of the fingers, a unique tactile position based controller is used for each of the fingers. The features to be tracked are the position and magnitude of the applied force on the fingertip. The magnitude is directly given by the tactile sensor, and the position is obtained as the position of the cell of the sensor region with the maximum value. To obtain a desired deformation on a soft object, the spring model generated on the previous section is used to compute the needed force on each contact which causes the desired deformation.

The scheme of the controller is described in Figure 4.

Fig. 4. Control scheme of the tactile-servo controller for each finger.

The goal of the controller is to obtain the sensor motion $\mathbf{V}_s^{\mathbf{sensor}} \in \mathbb{R}^3$ that minimizes the error in the features $\Delta f \in \mathbb{R}^3$. In the case of the fingers of a robotic hand, the physical joints do not allow significant rotational movements of the fingertips, so only lineal velocity is controlled. The response values of the controller, given as velocity, are integrated in time to get the new position of the fingertip on each iteration time. Using the kinematics of the fingers, the relative joint position for each joint is obtained, and sent to the position controllers of the real hand.

The vector of features $\mathbf{f} = [f_x, f_y, f_m] \in \mathbb{R}^3$ includes the position and magnitude of the applied force in the sensor. Position features (f_x, f_y) are set as the position of the cell with the maximum values in the sensor region located at each fingertip, and the magnitude is set as the sum of the forces of all the cells in the sensor area A_{sensor}.

$$f_m = \sum_{ij \in R} f_{ij} \tag{3}$$

The features' error $\mathbf{\Delta f}(t) = [\Delta f_x, \Delta f_y, \Delta f_m] \in \mathbb{R}^3$ includes the error values, respect to the target values $\mathbf{f_{tgt}} = [f\_x_{tgt}, f\_y_{tgt}, f\_m_{tgt}] \in \mathbb{R}^3$. The target in position (f_x_{tgt}, f_y_{tgt}) is set as the central position of the sensor cell in order to maintain the contact point in the center of the sensor and fingertip. The magnitude value (f_m_{tgt}) is an adaptable value given by the task planner in each iteration time. This target force is set depending on the task to be developed and the desired deformation on the object. Thus, this value is established according to the estimation of Young's modulus obtained from the object in the recognition stage.

A PID controller is executed to minimize the error of the desired features $\mathbf{\Delta f}$. Each feature is controlled by a different configuration of the PID controller. The parameters of the PID that controls the error in force magnitude Δf_m are set in order to obtain smooth responses, because an oscillation may cause undesired deformations on the object. The obtained control variable $\mathbf{u}$ is used in the next step to compute the motion of the fingers.

Fig. 5. Visualization of the tactile sensor values in the sensor coordinates, and visualization of the Shadow hand model with examples of the position of the sensor frames (O_s) and palm frame (O_p).

To map the response of the controller to the sensor velocity that minimizes the error, a task inverted Jacobian $\mathbf{J_s^{-1}} \in \mathbb{R}^{3X3}$ must be defined. The axes of the plane of the sensor region (x, y) correspond respectively with the axes X_s and Z_s of the sensor frame O_s (see Figure 5), so errors in force position are mapped in the X_s and Z_s axes. Mapping the force magnitude value is not trivial, different configurations of $\mathbf{J_s^{-1}}$ were tested to obtain an appropriate and realistic movement of the fingers. If the error in force is related only with the Z_s axis of O_s, the finger movements are not the desired ones. Each error step related with the force is related with the Z_s and Y_s

axes, in order to achieve a trajectory that can be followed by the fingers. In the case of the thumb, the matrix elements are negative, because the thumb is moved in the opposite direction to the rest of the fingers, so that a closing movement is produced. The next equation shows how the error is mapped:

$$V_s^{sensor} = J_s^{-1} \cdot \Delta f = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} \Delta f_x \\ \Delta f_m \\ \Delta f_y \end{pmatrix} \tag{4}$$

Considering the PID controller, the obtained velocity $\mathbf{V_s}^{sensor} = [V_x, V_y, V_z] \in \mathbb{R}^3$ in the sensor frame is shown in the next equation:

$$V_s^{sensor} = J_s^{-1} \cdot \left(K_p \cdot \Delta f(t) + K_i \cdot \int \Delta f(t)dt + K_d \cdot (\Delta f(t) - \Delta f(t-1)) \right) \tag{5}$$

The obtained velocity in the sensor frame O_s is transformed to the velocity $\mathbf{V_s}^{palm} \in \mathbb{R}^3$ on the reference frame of each finger, placed on the palm of the hand O_p. To transform the velocity, the adjoint matrix $\mathbf{Ad_T} \in \mathbb{R}^{6 \times 6}$ is used, which is related with the current transformation matrix between O_p and O_s:

$$Ad_{T_{ps}} = \begin{pmatrix} R_{ps} & \hat{p}_{ps}R_{ps} \\ 0 & R_{ps} \end{pmatrix} \tag{6}$$

From the sensor velocity relative to the palm reference, the new position ($P_s^{palm} \in \mathbb{R}^3$) for the fingertip is obtained integrating the value for each axis between t and $t-1$, and the position for each joint (q) is obtained with the inverse kinematics of the hand.

Finally, the task planner controls both the target position and fingertips' force at each step of a task. Each step of a manipulation task is defined by a configuration of the fingers positions and forces to be applied by each of them. In this paper, it is only discussed how the task planner changes the target forces. The control of the deformations is based on the Hooke's Law, using the approximated value $K^\sim$ obtained in the recognition step to get the relative force to be applied
:

$$\sigma_d = K^\sim * \epsilon_d \tag{7}$$

$$F_d = \sigma_d * A_{sensor} \tag{8}$$

Given a target value of strain ϵ_d, the related values of stress σ_d are computed. Then, this stress value can be traduced to target forces F_d for the tactile servo control on each finger, using the sensor area A_{sensor}.

5. Experimental Results

Several experiments, using the system described in Section 2, have been developed to test the recognition algorithm and the servo tactile control described in the previous sections. Different deformable objects were grasped, for which the volumes are known after the initial visual detection, but their density and deformability degree is unknown. Once the size and position in the workspace are obtained, the kinematics of the hand is used to position the fingers in a correct grasp configuration. Once this is done, first the recognition stage is carried out. As explained in Section 3, the movements of the fingers follow a trajectory towards the center of the object. During this process, the pressure of tactile sensors read at each fingertip region should increase as long as the contact is kept. For rigid objects, the finger positions do not vary while the pressure increases; for deformable objects, finger positions and pressures should change maintaining a relationship. The analysis of this relationship is the basis of this stage. The results of the recognition stage are used later to set the target values for the servo tactile control system.

In the first subsection, several experiments for the recognition stage are shown, and in the second subsection the experiments for the tactile servo control are explained.

5.1. Experiments for the recognition of the softness

Different objects with different properties were used. These properties are completely unknown for the system before the recognition stage (only size and position are known). Depending on the relation between size of the object and size of the volume that the hand can cover, the number of contact points are obtained and distributed uniformly over the width of the object (the thumb, on the opposite side, is positioned in the center of the object's width). Also the positions are centered relative to the height of the object. Once the hand is moved to the initial configuration, the algorithm described in the Section 3 is executed to create a reduced spring model of the object, composed by springs with an estimated stiffness factor $K^\sim$. The algorithm is executed with a minimum value of force of 1 newton, and a maximum of 2.5 newton.

In this section results for three different objects with different shapes and densities are reported (see Figure 18). For all the cases, the measures of the finger thumb (*F5*) differ considerably from the measures of the rest of the fingers (*F1 (first finger), F2 (middle finger), F3 (ring finger), F4 (little finger)*). This is due to the different role of the thumb, which is used as an opposite link for the rest of the fingers (the same role as in the human hand). The propagation of the forces applied by the rest of the fingers affects the motion of the thumb, because it has to counteract a higher total force.

5.1.1. Sponge

The first object to test is a sponge with size: 0.13 x 0.075 x 0.045m. According to the relation between its size and the volume of the space covered by the hand, five fingers are used. As seen in Figure 7, the strain values decrease while the object is compressed. For elastic objects, negative values represent a compression in the object, and positive values represent an extension in the object. As each step takes the same (configurable) period of time, it can be seen that there is a change in the slope at *iteration*=30. This is interpreted as a change in the deformation behavior of the object as follows: the object is deformed quickly until a strain value between -0.15 and -0.11; for more negative values, the object is deformed more slowly. In Figure 6, the values tend to increase to a value near 1 newton, except for the finger 4, which has lower values. This is caused by the deformation which the rest of the fingers produce.

Fig. 6. Evolution of the applied forces by each finger in the first experiment.

Fig. 7. Evolution of the caused strain for each finger in the first experiment.

Figure 8 shows the stress-strain curves. It can be seen that the finger which offers the most lineal response is finger 3, so the most valid value for the approximation is obtained for this finger at between *strain*=0 and *strain*=-0.15 (point of end of the response with high deformation at 1 newton), and the value of pressure when this strain is reached, *stress*=0.25 N/cm². The approximated value for the young's modulus is $K^\sim$= 1.66 N/cm².

Fig. 8. Stress-Strain relation for each finger in the first experiment.

5.1.2. Soft Ball

The second object to be used is a soft spherical ball with *radius*=0.045m. For this object, considering the size of the ball, only 4 fingers are enough to envelop it. Figure 9 shows the evolution of the applied forces and the strain for each finger during the execution of the recognition algorithm is shown in Figure 10. It can be seen that the behavior of high deformation occurs between *strain*=0 and *strain*=-0.065. At this point, *iteration*=70, the strain values slightly vary, so for a force threshold of 1 newton, the volume with low deformation is set at the limit *strain*=-0.065. The graphic of the evolution of the forces shows that the finger that applies the major force is finger 2 (middle finger), which is aligned with thumb and the center of the ball, so the compression is higher at this point.

Fig. 9. Evolution of the applied forces by each finger in the second experiment.

Fig. 10. Evolution of the caused strain for each finger in the second experiment.

The stress-strain relationship for each finger is showed in Figure 11. The most lineal response is given by the interaction finger 1 (first finger). Between values of strain=0 to *strain*=-0.058, and *pressure*=0.12N/cm2. The approximated value for the Young's module is $K^\sim$= 2.06 N/cm2.

Fig. 11. Stress-Strain relation for each finger in the second experiment.

5.1.3. Cylinder shaped foam

The third object is a soft cylinder with a *height*=0.01m and a *radius*=0.03m. The evolution of the applied forces and strains using five fingers are shown in Figures 12 and 13. The stiffness of this object is much higher than the stiffness of the sponge and the ball. It can be seen that the number of iterations is lower than in the rest of the experiments, because the threshold forces are reached and the strain becomes stabilized at *iteration*=15, when the values are between *strain*=-0.035 and *strain*=-0.05. The graphic of the forces shows that most of the fingers apply a force near 1 newton at *iteration*=15. After the value of 1 newton is reached, it can be seen that after *iteration*=25, when the force readjustment begins, the forces increase, but the strain values don't decrease at the same rate.

Fig. 12. Evolution of the applied forces by each finger in the third experiment.

Fig. 13. Evolution of the caused strain for each finger in the third experiment.

Figure 14 shows the stress-strain curves obtained for this object. The value for which the response is more lineal, is for finger 4 (little finger). The approximated value for the Young's modulus is calculated between *strain*=0 to *strain*=-0.055, and the applied *stress*=0.6 N/cm^2. The estimated value is $K^\sim$= 10.90 N/cm^2.

Fig. 14. Stress-Strain relation for each finger in the third experiment.

From the previous experiments it can be seen that a reliable estimation of the stiffness of the objects is obtained with the proposed recognition stage. Although the obtained value of $K^\sim$ may differ from the real value of Young's modulus for each material, it is not necessary for this system to obtain the real value, because the estimated value is enough to calculate the deformations that are caused by the forces applied by the fingers.

5.2. Servo tactile control results

The second part of the experimentation consists in testing the tactile-servo controllers for each of the fingers of the hand, which control the applied forces and level of deformation of the grasped object during an in-hand manipulation task. These controllers use the values of the relationship between stress and strain caused in the object to send target values to each finger. The next example shows how the tactile servo is applied to control the deformation of one of the objects, the ball. For each finger is shown the evolution in the applied forces and the error in force magnitude. Figure 17 shows the output velocities obtained by the controller for each of the fingers, to minimize the error. The PID controllers are configured with values: *P*=0.5, *I*=0.3, *D*=0.3.

Fig. 15. Evolution of the applied forces during a cycle of the tactile-servo control.

Fig. 16. Evolution of the error in forces during a cycle of the tactile-servo control.

Fig. 17. Evolution of the output velocities of the tactile-servo control for: thumb (F5), first finger (F1), middle finger (F2), ring finger (F3).

The main problem is the uncontrolled variation in the values obtained from the sensor, inherent to the physical properties of the sensor, which affects the system stability. Therefore, a small scale in the error is used to minimize the effect of high variations. This scale is adapted to the resolution of the sensor. Despite of these variations in the sensor values, the force values converge to the target force and the error is minimized (see Figure 15 and Figure 16).

Fig. 18. Images of the robotic hand manipulating different elastic objects. a: sponge in the initial configuration. b: sponge after a high deformation is caused. c: ball in the initial configuration. d: ball after a medium deformation is caused. e: cylinder foam in the initial configuration. f: cylinder ball after a minimal deformation is caused.

6. Conclusions

In this paper, a new control strategy based in tactile information for deformable objects' in-hand recognition and manipulation is presented. This approach is divided in two stages: first, a recognition stage for the estimation of the object's softness; second, a tactile-servo controller is used to maintain pressure contacts with the object in order to obtain a secure grasp and to control the degree of deformation. As seen in the experiments, the approach presents acceptable results, where the stability of the object grasped by the hand is achieved.

As future works, this control strategy will be improved in order to get more reliable features of the object, since badly estimated features could cause undesired results when the object is manipulated. The results shown in this paper are successful, but authors will test the algorithms with more complex deformable objects. In regard to servo-tactile control, it has been shown a reliable control with a good velocity convergence. In addition, stable configurations of the robot hand are performed during grasping and in-hand manipulation tasks.

Finally, the approach has demonstrated that is not necessary to previously have a complex model of the deformable object to get reliable estimations of its features, and to control the deformations in a manipulation task using only the internal information of the robot hand and the tactile data.

Acknowledgements

References

[1] Khalil F and Payeur P. Dexterous robotic manipulation of deformable objects with multi-sensory feedback – a review. Robot Manipulators, Trends and Development, In-Teh, Ed.; 2010; ch. 28, pp. 587–621.

[2] Yoshikawa T. Multifingered robot hands: Control for grasping and manipulation. Annual Reviews in Control vol. 34 (2); 2010; p. 199-208.

[3] Desbrun M, Schroder P, and Barr A. Interactive animation of structured deformable objects. Proceedings of Graphics Interface; 1999; 1-8.

[4] Faure F, Gilles B, Bousquet G, and Pai D K. Sparse meshless models of complex deformable solids. SIGGRAPH, vol. 1, no. 212; p. 1; 2011.

[5] Kaufmann P, Martin S, Botsch M, and Gross M. Flexible simulation of deformable models using discontinuous Galerkin FEM. SIGGRAPH; 2008.

[6] Wada T, Hirai S, Kawarnura S, and Karniji N. Robust manipulation of deformable objects by a simple PID feedback. Proc. IEEE International Conference on Robotics and Automation (ICRA); 2001.

[7] Frank B, Stachniss C, Abdo N, and Burgard W. Efficient motion planning for manipulation robots in environments with deformable objects. Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); Sept. 2011; pp. 2180–2185.

[8] Romano J M, Hsiao K, Niemeyer G, Chitta S, and Kuchenbecker K. J. Human-inspired robotic grasp control with tactile sensing. IEEE Transactions on Robotics; 201. http://doi.org/10.1109/TRO.2011.2162271.

[9] Corrales J A, Torres F, Perdereau V. Finger Readjustment Algorithm for Object Manipulation Based on Tactile Information. Int J Adv Robot Syst; 2013; 10:9. DOI: 10 5772/53561.

[10] Kien-Cuong N, Perdereau V. Fingertip force control based on max torque adjustment for dexterous manipulation of an anthropomorphic hand. Intelligent Robots and Systems (IROS); 2013; 2013 IEEE/RSJ International Conference on vol., no., pp.3557-3563, 3-7.

[11] Li Q, Elbrechter C, Haschke R, Ritter H. Integrating vision, haptics and proprioception into a feedback controller for in-hand manipulation of unknown objects. Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Tokyo, 2013, pp. 2466-2471.

[12] Luo Y and Nelson B J. Fusing force and vision feedback for manipulating deformable objects. Journal of Robotic Systems; 2011; 18(3), 103–117.

[13] Yousef H, Boukallel M, and Althoefer K. Tactile sensing for dexterous in-hand manipulation in robotics—A review. Sensors and Actuators A: Physical; 2011; 167(2), 171–187. http://doi.org/10.1016/j.sna.2011.02.038.

[14] Wettels N and Loeb G E. Haptic feature extraction from a biomimetic tactile sensor: Force, contact location and curvature. In 2011 IEEE International Conference on Robotics and Biomimetics, ROBIO 2011. http://doi.org/10.1109/ROBIO.2011.6181676.

[15] Liu H, Song X, Nanayakkara T, Seneviratne L D and Althoefer K. A computationally fast algorithm for local contact shape and pose classification using a tactile array sensor. In Proceedings - IEEE International Conference on Robotics and Automation; 2012; pp. 1410–1415.

[16] Chitta S, Sturm J, Piccoli M and Burgard W. Tactile sensing for mobile manipulation. IEEE Transactions on Robotics; 2011; 27(3), 558–568.

[17] Li Q, Schürmann C, Haschke R and Ritter H. A control framework for tactile servoing. Proceedings of Robotics: Science and Systems; 2013;

[18] Shadow Robot, Dexterous Hand. [Online]. Available:http://www.shadowrobot.com/products/dexterous-hand/

[19] Tactile Sensor Tekscan. [Online]. Available: http://www.tekscan.com/grip-pressure-measurement.

[20] Mateo C M, Gil P, Torres F. Visual perception for the 3D recognition of geometric pieces in robotic manipulation. The International Journal of Advanced Manufacturing Technology; 2015; pp 1-15. .