

Blind Manipulation of Deformable Objects Based on Force Sensing and Finite Element Modeling

José Sanchez, Kamal Mohy El Dine, Juan Antonio Corrales Ramon, Belhassen-Chedli Bouzgarrou, Youcef Mezouar

▶ **To cite this version:**

José Sanchez, Kamal Mohy El Dine, Juan Antonio Corrales Ramon, Belhassen-Chedli Bouzgarrou, Youcef Mezouar. Blind Manipulation of Deformable Objects Based on Force Sensing and Finite Element Modeling. Frontiers in Robotics and AI, 2020, 1, 10.3389/frobt.2020.00073 . hal-02925459

HAL Id: hal-02925459
https://uca.hal.science/hal-02925459

Submitted on 29 Aug 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

frontiers in Robotics and AI

ORIGINAL RESEARCH
published: 09 June 2020
doi: 10.3389/frobt.2020.00073

Blind Manipulation of Deformable Objects Based on Force Sensing and Finite Element Modeling

Jose Sanchez, Kamal Mohy El Dine, Juan Antonio Corrales, Belhassen-Chedli Bouzgarrou and Youcef Mezouar*

Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal, Clermont-Ferrand, France

In this paper, we present a novel pipeline to simultaneously estimate and manipulate the deformation of an object using only force sensing and an FEM model. The pipeline is composed of a sensor model, a deformation model and a pose controller. The sensor model computes the contact forces that are used as input to the deformation model which updates the volumetric mesh of a manipulated object. The controller then deforms the object such that a given pose on the mesh reaches a desired pose. The proposed approach is thoroughly evaluated in real experiments using a robot manipulator and a force-torque sensor to show its accuracy in estimating and manipulating deformations without the use of vision sensors.

Keywords: deformation servoing, finite-element modeling, tactile sensing, robotic manipulation, force sensing

OPEN ACCESS

Edited by:
Ozgur Erkent, Inria Grenoble - Rhône-Alpes Research Centre, France

Reviewed by:
David Navarro-Alarcon, Hong Kong Polytechnic University, Hong Kong
Chenguang Yang, University of the West of England, United Kingdom

****Correspondence:***
Jose Sanchez
jose.sanchez.loza@gmail.com

Specialty section:
This article was submitted to Sensor Fusion and Machine Perception, a section of the journal Frontiers in Robotics and AI

Received: *20 December 2019*
Accepted: *04 May 2020*
Published: *09 June 2020*

Citation:
Sanchez J, Mohy El Dine K, Corrales JA, Bouzgarrou B-C and Mezouar Y (2020) Blind Manipulation of Deformable Objects Based on Force Sensing and Finite Element Modeling. Front. Robot. AI 7:73. doi: 10.3389/frobt.2020.00073

1. INTRODUCTION

As deformable objects are ubiquitous in many industries, automating their manipulation would have a great social impact. For instance, robots could perform tasks that are either dangerous or monotonous for workers. Examples of manipulation of deformable objects can be found in the automobile and aerospace industries, where cables and wires must be connected in order to assemble motors; in health care, where clothes are handled to dress disabled people; and in the food industry where meat and produce have to be processed with care. This has sparked interesting robotic solutions that attempt to address issues such as automating the manufacture of motors by manipulating cables as proposed by Rambow et al. (2012), Roussel and Ta (2015), and Shah and Shah (2016), using robots to perform clothing assistance as described by Yu et al. (2017) and Erickson et al. (2017). and even food handling as proposed by Bac et al. (2017) and Lehnert et al. (2017). The interest in the robotic community to address the sensing and manipulation of deformable objects has been increasing in recent years as shown by the number of works covered in different surveys as presented by Henrich and Wörn (2000), Khalil and Payeur (2010), and Sanchez et al. (2018a).

In this paper we propose an approach to both sense and manipulate the deformation of soft object[1]. Specifically, we extend our previous work where a pipeline to estimate the deformation of a soft object was proposed in Sanchez et al. (2018b). This deformation sensing pipeline requires an input force and an initial mesh to output the updated mesh that describes the deformation of the object. Here, in addition to enhancing the sensing capabilities of our previous work, we develop a controller to manipulate a deformable object. The sensing improvements are mainly due to changing the input sensor from a set of tactile sensors on the finger tips of a robotic hand to a single force-torque sensing attached to the end of a robotic arm. Regarding the manipulation of a

[1] An implementation of the code is available at https://github.com/jsanch2s/uca_deformation_sensing.

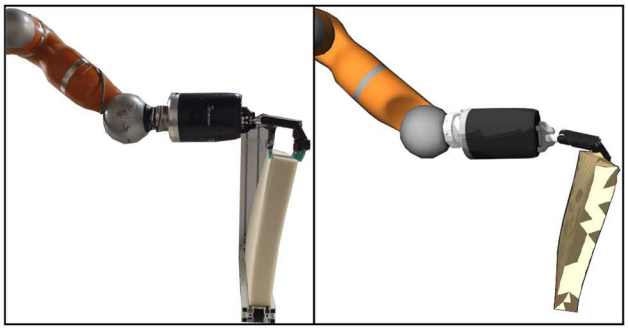


FIGURE 1 | Real deformation of a block-like object, shown on the left side, compared to the estimation of the proposed approach, visualized in RViz, shown on the right side.

soft object, replacing a robot hand by a robotic manipulator simplifies the manipulation problem since a robotic hand is limited to its finger motions (e.g., opening and closing the hand); while a seven degree of freedom manipulator is able to move in the 3D Cartesian space. In order to perform this manipulation we simplify the problem by controlling the position of a single pose on the mesh. An comparison between our estimation and a real world deformation can be seen in **Figure 1**.

Following this introduction we review the state of the art for robotic sensing and manipulation of deformable objects in section 2. In section 3, we detailed our approach to estimate the deformation of an object while being manipulated as well as controlling its deformation. An experimental validation of the proposed approach is presented in section 4 followed by a discussion of the experimental results in section 5. Finally, our concluding remarks and future work are summarized in section 6.

2. RELATED WORK

Two decades ago, one of the first works focusing on controlling the shape of a deformable object was presented by Wada et al. (1998), where the problem was formulated as an Indirect Simultaneous Positioning (ISP) problem. ISP consists in defining two sets of points, namely, controlled and manipulation points. Where the former are points on the surface of the object (usually away from the edges such that are not graspable), while the latter are defined by positions where a robotic manipulator is grasping the object.

Following the work of Wada, other researchers have tackled the ISP problem using vision to obtain the position of the controlled points. Navarro-Alarcón et al. in a series of works, proposed a way to control the configuration of a deformable object using visual servoing. Here, the configuration of the object is described using *deformation feature* vectors, based on a set of points tracked using markers. The proposed deformation vectors were defined by a set of points in an object to control a point, a distance or angle between two points, and a curvature described by three points. These approaches rely on a deformation Jacobian, which here refers to a matrix mapping the motion of the grippers to the deformation of the object. In Navarro-Alarcon et al. (2013), the deformation Jacobian is estimated using the Broyden method, which computes the Jacobian once at the beginning and then approximates it at each iteration using the previous Jacobian and the changes of the feature vectors and the end-effector's pose; and, in Navarro-Alarcon et al. (2014), they proposed a new estimation that used views from multiple cameras. However, both of these approaches were limited as they control the deformation features on a plane, namely in the image space. This was later addressed in Navarro-Alarcon et al. (2016), by using stereo-vision to track the points in 3D and subsequently define the deformation feature vectors also in 3D.

A similar approach proposed by Alambeigi et al. (2018) extended its application to heterogeneous objects while being robust to disturbances, e.g., the objects were filled with water beads (heterogenous) and then cut (disturbance) while the controller was running. To achieve this, instead of relying only on a deformation Jacobian, they combined it with an image Jacobian to consider both the deformation behavior of the object as well as the feedback points obtained by the vision system. Recently, Hu et al. (2018), using an RGB-D camera, were able to obtain a much faster convergence in the deformation control than the approaches proposed in Navarro-Alarcon et al. (2014, 2016) by using Gaussian Process Regression (GPR) to estimate the deformation Jacobian, since GPR is able to model non-linearities.

Instead of relying on vision, Berenson (2013) used simulated objects described by a set of points, where the positions were assumed to be known at any time. Unlike the approaches described above, Berenson estimated the deformation Jacobian by introducing the concept of *diminishing rigidity* which assumes that the controlled points closer to manipulated points (e.g., a point where the object is being held) behave rigidly and the farther the controlled points are from manipulated points the less rigid they become.

A novel approach for controlling the shape of an object without requiring an ISP formulation, was recently proposed in Navarro-Alarcon and Liu (2018), where the object's contour was described by Fourier coefficients. Thus, instead of controlling the deformation of an object based on a set of points, they deformed the object such that the contour of the object was similar to the desired one.

Other works have focused on sensing the deformation of an object rather than controlling its shape using vision, either by relying on a mesh model as proposed in Petit et al. (2015), Fugl et al. (2012), Schulman et al. (2013), and Frank et al. (2010) or being model-free (see Cretu et al., 2012; Güler et al., 2015). Due to space limitations we do not review the works focused on deformation sensing, but we refer the interested reader to a recent survey were these works are thoroughly reviewed (Sanchez et al., 2018a).

3. APPROACH

As described in the previous section, most work concerning deformation control has either relied on vision sensing or

assumed that the object's shape (e.g., a set of points describing the object) is known at any time. However, vision systems are sensitive to variations in illumination, have difficulties segmenting objects with similar colors to their backgrounds and, perhaps most importantly, they are limited by occlusions arising while manipulating objects. In order to provide a complementary sensing modality, we propose a method using force sensing, coupled with a deformation model, to estimate the shape of an object while being deformed.

We propose a manipulation controller, shown in **Figure 3**, which uses an improved version of the deformation sensing pipeline proposed in Sanchez et al. (2018b) and a novel force sensor model, described in Mohy el Dine et al. (2018), that estimates the pure contact forces generated while deforming an object grasped by a robot manipulator. This approach shows a potential application when vision is compromised, e.g., when a room is too dark or cluttered.

3.1. Force Sensor Model

Since the input to the deformation sensing pipeline is the contact force that deforms the object, we obtain this force using a force-torque sensor. However, as force-torque sensors not only measure contact forces but also non-contact forces generated by gravity, inertia, Coriolis and centrifugal forces (as depicted in **Figure 2**); it is necessary to first estimate the non-contact forces to then subtract them from the measured forces in order to obtain the pure contact forces. Thus, the output of a force-torque sensor can be expressed as:

$$\mathbf{F}_{meas} = \mathbf{F}_{nc} + \mathbf{F}_{c} \tag{1}$$

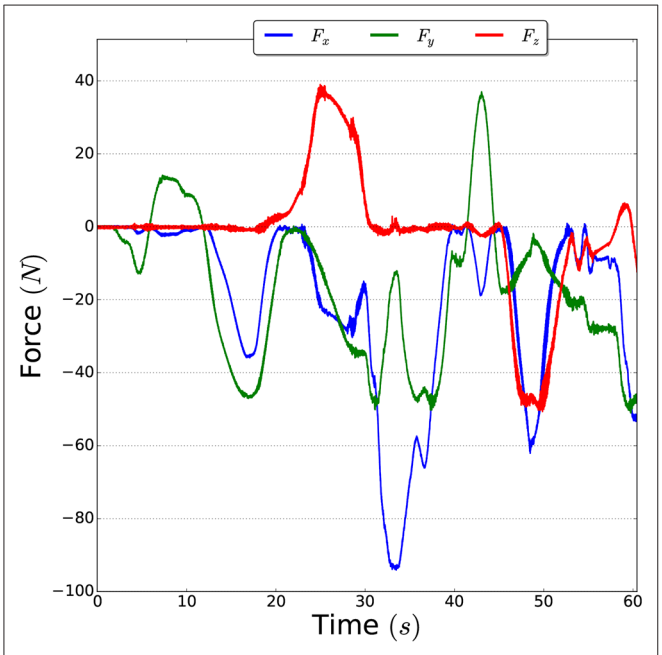


FIGURE 2 | Example of non-contact forces measured by the force-torque sensor when the robot arm is moved without any contacts.

where $\mathbf{F}_{meas}$ is the measured force, i.e., the output of a force-torque sensor, $\mathbf{F}_c$ represents the contact forces and the non-contact forces $\mathbf{F}_{nc}$ are defined as:

$$\mathbf{F}_{nc} = \mathbf{F}_{gravity} + \mathbf{F}_{inertia} + \mathbf{F}_{coriolis} + \mathbf{F}_{centrifugal} \tag{2}$$

In order to estimate these non-contact forces from a force-torque sensor, we use a recently proposed observer that was shown to outperform analytical based methods by using a Recurrent Neural Network. The Recurrent Neural Network observer (RNNOB) proposed in Mohy el Dine et al. (2018), takes as input the orientation and twist of the robot end-effector frame w.r.t. the robot base frame $[\mathbf{o}^{ee}, \boldsymbol{\upsilon}^{ee}, \boldsymbol{\omega}^{ee}]^T$ which are obtained by the first and second order forward kinematics; as well as the linear acceleration $\boldsymbol{\alpha}^{IMU}$ published by an IMU sensor. The output of the RNNOB is the estimated non-contact force w.r.t. the sensor frame $\hat{\mathbf{F}}^s_{nc}$. Then, the pure contact forces, expressed in the sensor frame, can be estimated as:

$$\mathbf{F}^s_c = \mathbf{F}^s_{meas} - \hat{\mathbf{F}}^s_{nc} \tag{3}$$

The network's architecture consisted of two hidden layers, with 15 and 10 LSTM units, and it was trained over 50 epochs using 80% of the data set (∼10,000 data points). The sequence length of the input layer was of 100 time steps (0.2 s). In the output layer, we applied Stochastic Gradient Descent with a learning rate of 0.01 to minimize the mean square error of the regression problem. A hyperbolic tangent sigmoid function was used as the activation function between the layers.

3.2. Deformation Sensing Pipeline

To estimate the deformation of the object we improve the sensing pipeline proposed in our previous work (Sanchez et al., 2018b). Namely, we replaced the tactile sensors by a single force-torque sensor and, instead of pushing the objects with the fingertips of a robot hand, the object is grasped by a robot hand. The advantages of grasping an object over pushing it with the fingertips, are twofold:

1. The object can be manipulated in multiple directions (i.e., it can be moved up/down and side to side and be pulled and pushed).
2. The contact locations can be assumed to remain fixed.

The first advantage allows us to better control the object, where as the second advantage improves the estimation since the location of the contact forces is known and constant (e.g., no slippage of contact). As shown in **Figure 3**, the deformation sensing pipeline takes as input the initial state of a mesh $\mathbf{q}_{init}$ and a force F to estimate the mesh deformation. Formally, a mesh $\mathbf{q} \in \mathbb{R}^{3n}$ is represented as set of n nodes connected by tethrahedral elements. The mesh's deformation is estimated by applying the external forces and propagating the internal forces throughout the mesh. By solving the following differential equation we can obtain the updated positions of the nodes:

$$\mathbf{F}_{ext} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{F}_{int}(\mathbf{q}) \tag{4}$$

where $\mathbf{F}_{ext} \in \mathbb{R}^{3n}$ is a vector of the three-dimensional forces applied to each node n. The position, velocity and acceleration

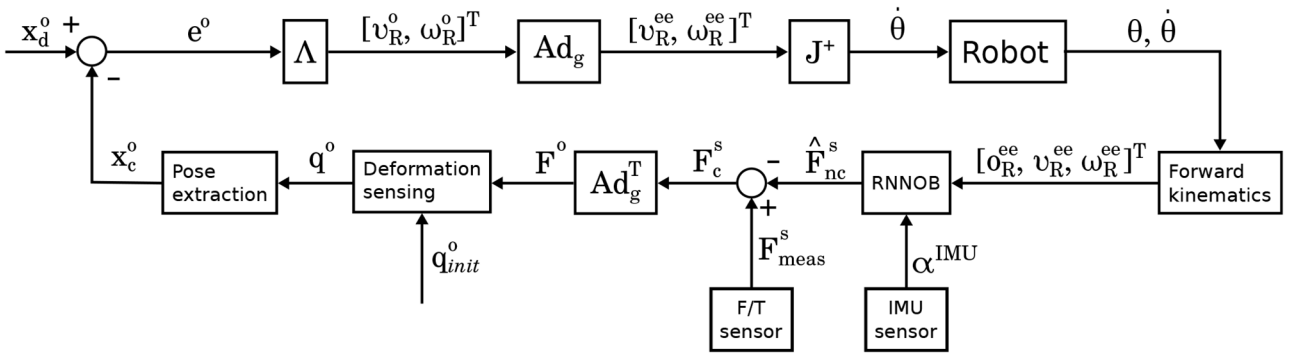


FIGURE 3 | Block diagram of the proposed deformation controller. The current and desired poses, $\mathbf{x}_c^o$ and $\mathbf{x}_d^o$, respectively, are described w.r.t. the object frame. As the robot expects the end-effector twist expressed in the robot base frame $\{R\}$, the twist expressed on the object frame, namely $[\boldsymbol{\upsilon}_R^o, \boldsymbol{\omega}_R^o]$, must be multiplied by an adjoint matrix $\mathbf{Ad}_g$ relating these two frames in order to obtain the desired twist ($[\boldsymbol{\upsilon}_R^{ee}, \boldsymbol{\omega}_R^{ee}]$).

of each node is represented by $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$, respectively, with $\mathbf{q} \in \mathbb{R}^{3n}$. The mass matrix is $\mathbf{M} \in \mathbb{R}^{3n \times 3n}$ and $\mathbf{D}$ represents the damping matrix.

Since the deformation sensing pipeline requires the input force to be described w.r.t. the object frame, we transform the force using the adjoint matrix Ad_g^T, as obtained by the force sensor model described in section 3.1, as follows:

$$\mathbf{F}^o = Ad_g^T \cdot \mathbf{F}_c^s \tag{5}$$

3.3. Manipulating Deformation

Since controlling the complete shape of an object (e.g., all the nodes of a mesh) is a heavily underactuated problem, except for trivial cases, we reduce the control problem by focusing on controlling a single pose on the mesh. To this end, we first must extract a suitable pose from the mesh and consequently control that pose such that it reaches a target pose. To extract a pose we exploit the fact that the nodes of the mesh are connected by tetrahedra so three suitable nodes would form a triangle as shown in **Figure 4**. By selecting three neighboring nodes (p_1, p_2, p_3) from the mesh, we can extract a pose as follows:

$$\mathbf{c} = \frac{1}{3}\sum_{i=1}^{3} \mathbf{p}_i \tag{6}$$

$$\mathbf{n} = (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1) \tag{7}$$

$$s = \cos\left(\frac{\pi}{2}\right), \quad \mathbf{v} = \mathbf{n} \cdot \frac{\sin\left(\frac{\pi}{2}\right)}{\|\mathbf{n}\|_2} \tag{8}$$

$$\mathbf{x} = [\mathbf{c}, (s, \mathbf{v})]^T \tag{9}$$

where $\mathbf{p}_i$ is the position of the node at the $i-$th index of the mesh $\mathbf{q}$ and $\mathbf{n}$ represents the normal of the plane formed by the points. The position and orientation of the pose are given by the centroid $\mathbf{c}$ and the quaternion, respectively, where s is the scalar part of the quaternion and $\mathbf{v}$ is the vector part. To manipulate the mesh to

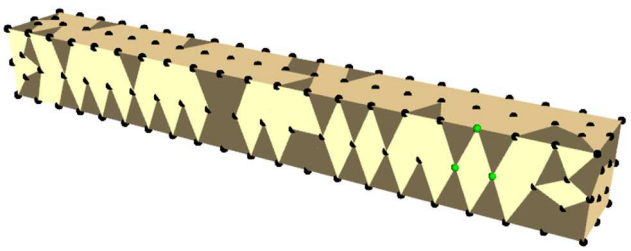


FIGURE 4 | Simulated mesh of a bar-like object. The mesh nodes are shown in black and the nodes used to extract a pose are in green.

a desired pose, let $\mathbf{x}_c$ denote the current pose, as extracted by the method described above, and $\mathbf{x}_d$ the desired pose for the mesh to reach. Then, we can define an error signal as follows:

$$\mathbf{e} = \mathbf{x}_d - \mathbf{x}_c \tag{10}$$

Subsequently, we can transform the error $\mathbf{e}$ into a twist command by multiplying it by a diagonal gain matrix Λ. As the twist is expressed in Cartesian space, we compute the joint velocities to command a robotic arm by joint velocity control as:

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^+(Ad_g \cdot \Lambda \cdot \mathbf{e}) \tag{11}$$

where $\mathbf{J}^+$ is the Moore–Penrose inverse Jacobian used for redundant manipulators.

4. EXPERIMENTAL EVALUATION

We evaluate the performance of our proposed approach by deforming four objects with different shapes and material properties. The two shapes of the objects[2] are described in **Table 1** and the materials properties are shown in **Table 2**. The

[2]The objects were bought from the following vendor: http://www.moussesurmesure.com/.

TABLE 1 | Geometric information of the test objects.

	Dimensions (cm)			Mesh	
	Length	Width	Height	Nodes	Elements
Block	6	40	40	360	1,079
Bar	6	6	50	207	536

TABLE 2 | Material properties of the test objects.

		Elasticity parameters		
	Material name	Mass density (kg/m^3)	Young modulus (Pa)	Poisson ratio
Hard	HR 45	45	18,500	0.15
Soft	Bultex 26	26	9,000	0.15

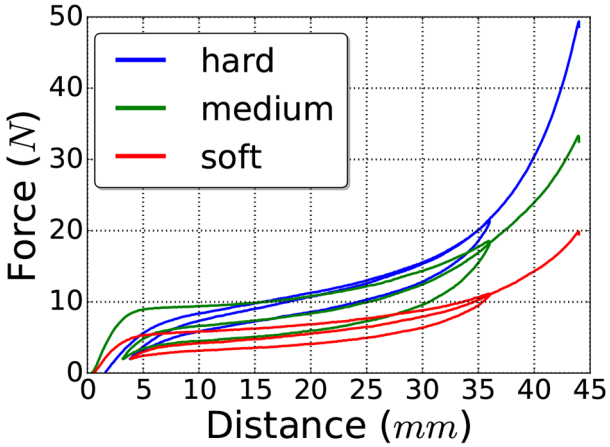


FIGURE 5 | Displacement-force curve showing the hysteresis of the test materials.

elastic behavior of isotropic materials can be represented by two parameters, such as the Young's modulus and the Poisson's ratio. One common way to obtain these parameters is to perform a compression test which consists on pushing down on an object while simultaneously recording the displacement[3] of the object and the applied force.

We attempted to estimate the material properties of a cube object with three different materials: *hard* (HR 45), *medium* (Bultex 30), and *soft* (Bultex 26) (note that only the *hard* and *soft* materials were used in the experiments described in section 5). For the test, the object was placed inside of a press that can be programmed to move on a vertical axis until a desired height and is equipped with a force sensor. The press was set to first move down to approximately compress the object 36 *mm*, then to move upwards until the compression was around 4 *mm* and finally the press was moved down until 44 *mm* of compression were reached. The results of this test can be seen in **Figure 5**, where the compression is plotted against the measured force.

It can be seen from these figures, that the behavior of all three materials is not only non-linear but also presents hysteresis. Hence, modeling these materials as linear will result in inaccuracies. Nevertheless, we can use the test findings to obtain a range estimation of the Young's modulus for each material and then tune them individually until the simulation behavior better matches reality.

Two sets of experiments were conducted: one to measure the accuracy of the deformation sensing and the second one to assess the manipulation capabilities.

The experimental setup used for both sets of experiments is shown in **Figure 6**, where the test rig ensures the contact points are consistent among the experiments. The seven degree of freedom robot manipulator KUKA LWR+4 arm (Bischoff et al., 2010) with the dexterous Shadow Dexterous Hand[4] attached at its end were used to manipulate the objects. The inputs to the sensor model were obtained by an ATI Gamma[5] force-torque sensor and an inertial measurement unit (IMU)[6] sensor, both attached to the robot arm as shown in **Figure 6**. The test objects were fixed on a test rig, where the bar objects were fixed such that their long axis were parallel to the XY plane of the robot frame and the block objects were attached by their bottom side, as seen in **Figures 6**, **8**.

The ANSYS[7] software was used to generate the volumetric meshes of the four test objects and the Co-rotational Linear method from the Vega FEM library (Sin et al., 2013) was used to implement the deformation model. As for the force sensor model, we used the same network architecture described in Mohy el Dine et al. (2018) but trained the network with motions of the robot arm that were adequate for our experiments.

4.1. Deformation Sensing

To evaluate the sensing accuracy of our proposed pipeline we commanded a set of poses in the XZ plane of the robot frame, as shown in **Figure 7**. The bar objects had to reach the six different poses seen in **Figure 7**, where as the block objects were commanded to move to three poses along the Z axis (see **Figure 8**), since they were fixed on their bottom side. The error signal was computed as the distance per axis between the pose where the end-effector grasped the object (reference) and a pose on the mesh (measured). The measured pose was obtained by using the pose extraction method outlined in section 3.3, where the indices were chosen such that the pose coincided with the reference pose before the object was deformed.

For each object, seven trials were performed consisting in three and six target positions for the block and the bar objects, respectively. The results are summarized in **Figure 9**, where, the mean of the absolute error between the reference and measured positions was computed for each trial.

[3]Here, displacement refers to the compression of the object, i.e., the object is fixed at one side while the opposite side is moved toward the fixed side.
[4]https://www.shadowrobot.com/products/dexterous-hand/.
[5]http://www.ati-ia.com/products/ft/ft_models.aspx?id=Gamma.
[6]We used the Adafruit (L3GD20H + LSM303) sensors: https://www.adafruit.com/product/1714.
[7]https://www.ansys.com/.

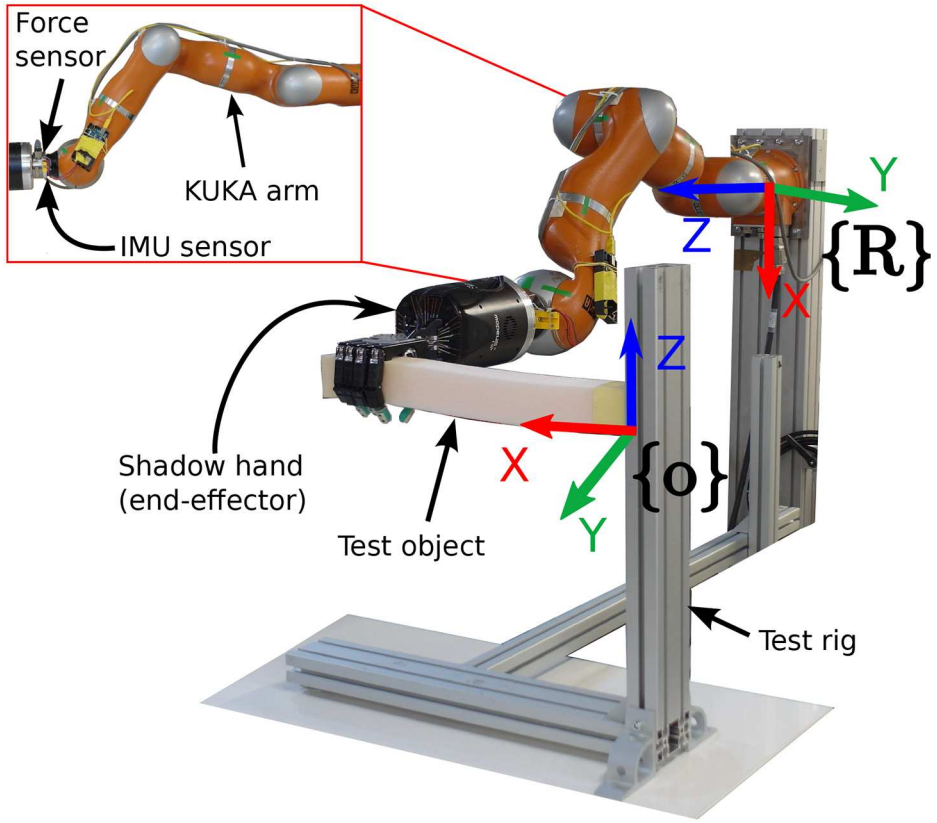


FIGURE 6 | Experimental setup for a bar-like object.

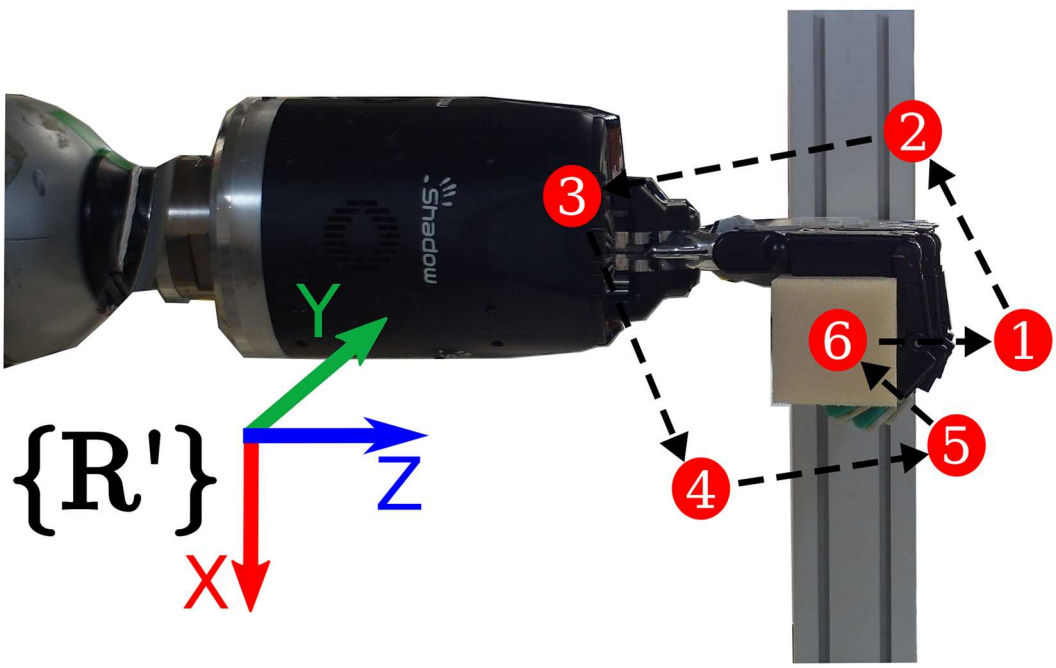


FIGURE 7 | Example of the path to follow the six test poses by the bar objects during the sensing evaluation. The *R*′ denotes a reference frame having the same orientation as the robot base frame (see **Figure 6**) but a different translation in order to make it visible.

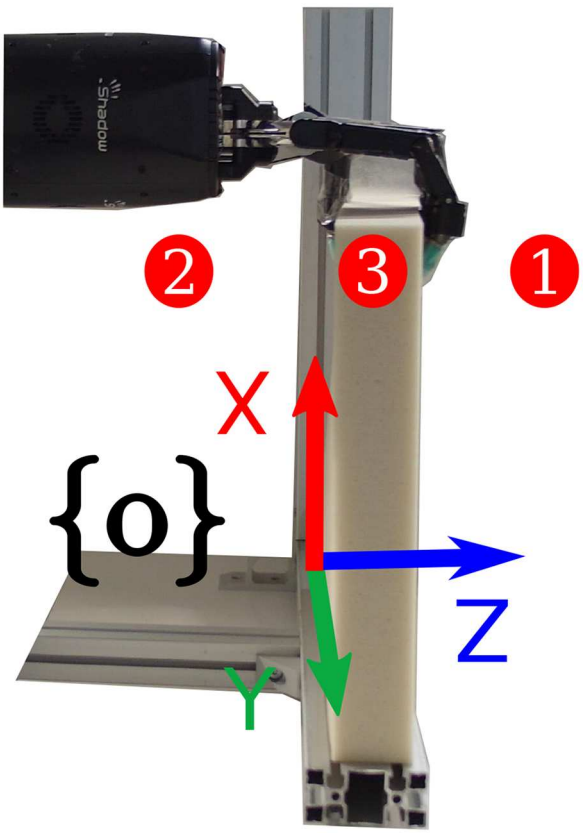


FIGURE 8 | Test poses used for the block-like objects for the sensing evaluation.

4.2. Manipulating Deformation

The performance of manipulating deformation was evaluated on the *bar soft* and *block hard* objects. For the bar object we set poses along the X and Z axis and the XZ plane as shown in **Figure 7**. As mentioned above, due to the block object being fixed at its bottom side, we only commanded poses along the Z axis as it can be seen in **Figure 8**. Similarly to the sensing evaluation, we defined the reference pose as the pose where the end-effector grasped the object and the measured pose was extracted using the pose extraction method as described above. Furthermore, we used a graphical user interface to set a target pose for the measured pose to reach.

Note that these control tasks could be solved by simply commanding a desired pose for the end-effector without regarding the mesh deformation. However, the purpose of this evaluation is to show that the shape of the object is estimated accurately enough to apply a controller.

5. RESULTS AND EXPERIMENTAL INSIGHTS

5.1. Sensing Evaluation

Before using the deformation sensing model in the control pipeline in **Figure 3**, it was evaluated quantitatively based on the experiments described in section 4.1 to draw ourselves and the reader an idea about the accuracy and precision of the model. The box plots for X and Z estimation of bar hard object are relatively small (< 2 cm) for 50% of the measurements, **Figure 9A** shows an accuracy of 1.6 cm in X and 2 cm in Z, however the precision is < 0.8 cm in X and < 0.3 cm in Z. Concerning the box plots of bar soft, for 50% of the measurements, they are < 2 cm as well in **Figure 9B**; the estimated position accuracy is 0.8 and 2 cm, respectively, in X and Z while the precision is < 0.4 cm in X and < 0.6 cm in Z. Concerning the blocks, the box plots are very small (< 2 mm) in **Figures 9C,D**. This reflects a 1 mm precision of the different measurements conducted for the both blocks. however, the accuracy is 3.8 cm along Z for the block hard and 2.25 cm for the block soft. The variances in the blocks position estimation measurements are smaller than the ones of the bars as the blocks are stiffer, hence they are more resistant to the small force reading errors. Regarding the different position accuracy measurements for the objects used, it is dependent on how good are the physical parameters in **Table 2** applied to the mesh.

5.2. Soft Bar Experiments

The current pose estimation using the mesh is evaluated in the velocity control loop shown in the block diagram in **Figure 3**. **Figure 10** shows the control position commands (green), along the X and Z axes, against the response positions extracted from the mesh (red) and the reference positions extracted using the robot kinematics (blue). The figures show that the controller is able to bring the mesh to the desired positions, along the X and Z axes, within a 2 cm error (the desired positions are reached at 19, 40, and 60 s) as shown in **Figure 11**. Moreover, in **Figure 10**, a clear delay can be seen between the command and the response of the mesh. This delay is mainly because of three reasons:

1. **Command:** Since the commanded signal is set by a GUI, the target poses for the controller result in a sharp slope.
2. **Elasticity parameters:** The deformation model is highly sensitive to the elasticity parameters, namely, the Young modulus and Poisson ration. Thus, an adequate identification is necessary.
3. **Sensor noise:** In order to obtain a smoother signal from the force sensors, a filter was applied but this introduced additional delay.

These reasons cause the controller to have a response similar to damped step response and takes around 20 s to converge to the commanded goal. Hence, the error increases the most when a command is received, then it converges to a value close zero (see **Figure 11**).

Concerning the small vibrations in the positions obtained from the mesh, they are mainly because of the small noise in the force readings. We note here that bar used in the tests is very soft (**Table 2**) that it can be deformed easily by small forces ($< 2N$), that's why the noises from residual non-contact forces and sensor noise have this much effect on the deformation model. For harder objects we expect the mesh to be less sensible to this noise.

5.3. Hard Block Experiments

The same controller shown in **Figure 3** was used to move the block hard object to a different poses along the Z axes. **Figure 12**

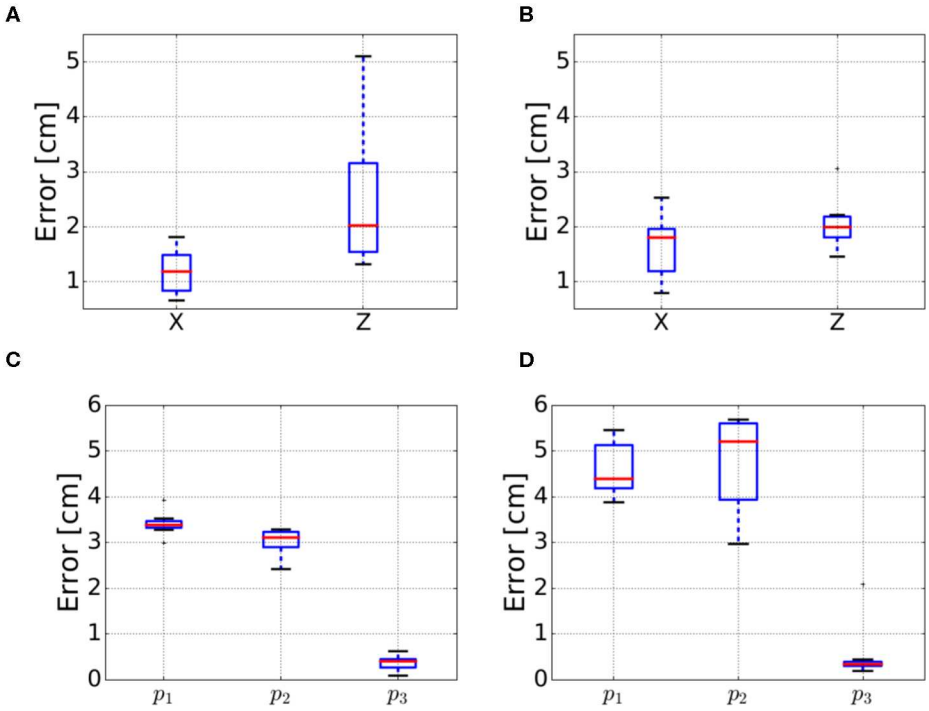


FIGURE 9 | Errors for the deformation sensing evaluation for the four test objects. The errors for the bar objects refer to the average errors on the *X* and *Z* axes. For the block objects p_1 refers to the front position, p_2 to the back position and p_3 is the original position after manipulating the object as shown in **Figure 8**. **(A)** Bar soft. **(B)** Bar hard. **(C)** Block soft. **(D)** Block hard.

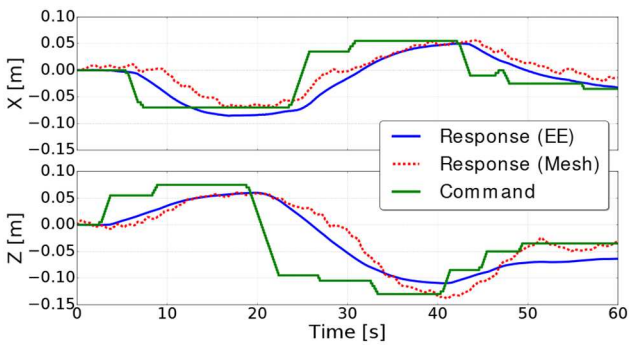


FIGURE 10 | Command and responses of the mesh and the robot end effector (EE) along the *X* and *Z* axes for the bar soft object.

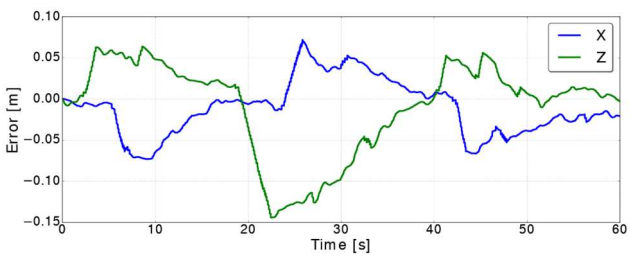


FIGURE 11 | Control errors along the *X* and *Z* axes for the bar soft object.

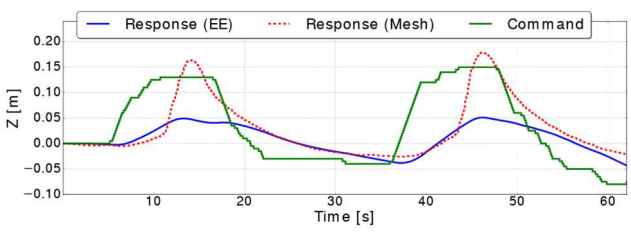


FIGURE 12 | Command and responses of the mesh and the robot end effector (EE) along the *Z* axis for the block hard object.

shows the commanded position against the position extracted from the mesh, where the response of the controller shows a similar delay to the one in the bar experiment. However, the mesh estimation of the position overshoots as the target goal is far as almost twice as the one from the bar experiments. This is caused by the controller commanding the arm to move faster which in turn generates more residual inertial forces resulting in the mesh overshooting. It is important to note, that in these experiments we focus on measuring how accurate the controller can bring the mesh to a desired position without using vision sensors, rather than evaluating the controller's performance. **Figure 13** shows the errors converge to less than

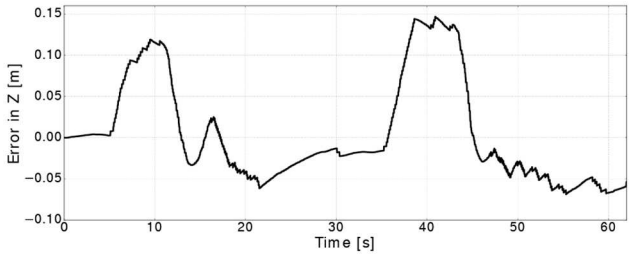


FIGURE 13 | Control error along the *Z* for the block hard object.

2 cm when the targets along the *Z* are reached (i.e., at 17, 35, and 45 s).

6. CONCLUSION AND FUTURE WORK

We have presented a pipeline that is able to simultaneously estimate and manipulate deformable objects without the need of a vision system, which is widely required by other approaches. The proposed deformation sensing pipeline takes as input a contact force and an initial mesh with the physical characteristics of the object. An FEM simulation estimates the deformation of the mesh based on the input force and in order to control the mesh, a pose, extracted from a given set of nodes of the mesh, is driven by a robot manipulator to reach a desired pose. The approach was experimentally validated using a robotic arm with a dexterous hand as an end-effector and an attached force-torque sensors. The experimental validation showed promising results and precision to a certain limit.

The accuracy of the approach is limited due to several factors such as the control loop being closed on the pose extracted from the mesh which is subject to drift due to noise of the force-torque sensor readings. Additionally, the physical parameters of each object are not obtained in a straight forward manner, thus affecting the estimation between the deformation of the real object and the estimation by the model. In future work, we will focus on incorporating vision to correct the errors in the estimation of the deformation, as well as investigating the performance of different controllers for deformation control found in the literature (e.g., Navarro-Alarcon et al., 2016; Hu et al., 2018). Furthermore, this approach can be extended to multi-robot systems, such that each robot controls a pose on the mesh and thus increasing the shape control capabilities.

DATA AVAILABILITY STATEMENT

All datasets generated for this study are included in the article/supplementary material.

AUTHOR CONTRIBUTIONS

All the authors participated in the development and validation of the research work presented in this paper.

FUNDING

This work has been sponsored by the French government research program Investissements d'Avenir through the RobotEx Equipment of Excellence (ANR-10-EQPX-44) and the IMobS3 Laboratory of Excellence (ANR-10-LABX-16-01), by the European Union through the program Regional competitiveness and employment 2007-2013 (ERDF - Auvergne region), by the Auvergne region and by the project CoMManDIA (SOE2/P1/F0638), which is cofinanced by the Interreg Sudoe Programme (European Regional Development Fund).

REFERENCES

Alambeigi, F.,Wang, Z., Hegeman, R., Liu, Y., and Armand, M. (2018). A robust data-driven approach for online learning and manipulation of unmodeled 3-D heterogeneous compliant objects. *IEEE Robot. Automat. Lett.* 3, 4140–4147. doi: 10.1109/LRA.2018.2863376

Bac, C., Hemming, J., Van Tuijl, B., Barth, R., Wais, E., and van Henten, E. (2017). Performance evaluation of a harvesting robot for sweet pepper. *J. Field Robot.* 34, 1123–1139 doi: 10.1002/rob.21709

Berenson, D. (2013). "Manipulation of deformable objects without modeling and simulating deformation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Tokyo), 4525–4532. doi: 10.1109/IROS.2013.6697007

Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schaeffer, A., Beyer, A., et al. (2010). "The KUKA-DLR Lightweight Robot arm - a new reference platform for robotics research and manufacturing," in *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)* (Munich), 1–8.

Cretu, A.-M., Payeur, P., and Petriu, E. M. (2012). Soft object deformation monitoring and learning for model-based robotic hand manipulation. *IEEE Trans. Syst. Man Cybernet. B* 42, 740–753. doi: 10.1109/TSMCB.2011.2176115

Erickson, Z., Clegg, A., Yu, W., Turk, G., Liu, K., and Kemp, C. (2017). "What does the person feel? Learning to infer applied forces during robot-assisted dressing," in *IEEE International Conference on Robotics and Automation* (Singapore), 6058–6065. doi: 10.1109/ICRA.2017.7989718

Frank, B., Schmedding, R., Stachniss, C., Teschner, M., and Burgard, W. (2010). "Learning the elasticity parameters of deformable objects with a manipulation robot," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Taipei), 1877–1883. doi: 10.1109/IROS.2010.5653949

Fugl, A. R., Jordt, A., Petersen, H. G., Willatzen, M., and Koch, R. (2012). "Simultaneous estimation of material properties and pose for deformable objects from depth and color images," in *Pattern Recognition*, Lecture Notes in Computer Science, eds A. Pinz, T. Pock, H. Bischof, and F. Leberl (Berlin; Heidelberg: Springer), 165–174. doi: 10.1007/978-3-642-32717-9_17

Güler, P., Pauwels, K., Pieropan, A., Kjellström, H., and Kragic, D. (2015). "Estimating the deformability of elastic materials using optical flow and position-based dynamics," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)* (Seoul), 965–971. doi: 10.1109/HUMANOIDS.2015.7363486

Henrich, D., and Wörn, H., editors (2000). *Robot Manipulation of Deformable Objects. Advanced Manufacturing*. London: Springer. doi: 10.1007/978-1-4471-0749-1

Hu, Z., Sun, P., and Pan, J. (2018). Three-dimensional deformable object manipulation using fast online Gaussian process regression. *IEEE Robot. Automat. Lett.* 3, 979–986. doi: 10.1109/LRA.2018.2793339

Khalil, F. F., and Payeur, P. (2010). "Dexterous robotic manipulation of deformable objects with multi-sensory feedback - a review," in *Robot Manipulators Trends and Development* (Intech), 587–620.

Lehnert, C., English, A., McCool, C., Tow, A., and Perez, A. (2017). Autonomous sweet pepper harvesting for protected cropping systems. *IEEE Robot. Automat. Lett.* 2, 872–879. doi: 10.1109/LRA.2017.2655622

Mohy el Dine, K., Sanchez, J., Corrales, J. A., Mezouar, Y., and Fauroux, J.-C. (2018). "Force-torque sensor disturbance observer using deep learning," in *Springer Proceedings in Advanced Robotics, ISER 2018*, Vol. 11 (Buenos Aires, Cham: Springer), 364–374. doi: 10.1007/978-3-030-33950-0_32

Navarro-Alarcon, D., Liu, Y., Romero, J., and Li, P. (2013). Model-free visually servoed deformation control of elastic objects by robot manipulators. *IEEE Trans. Robot.* 29, 1457–1468. doi: 10.1109/TRO.2013.2275651

Navarro-Alarcon, D., and Liu, Y.-H. (2018). Fourier-based shape servoing: a new feedback method to actively deform soft objects into desired 2-D image contours. *IEEE Trans. Robot.* 34, 272–279. doi: 10.1109/TRO.2017.2765333

Navarro-Alarcon, D., Liu, Y.-H., Romero, J. G., and Li, P. (2014). On the visual deformation servoing of compliant objects: uncalibrated control methods and experiments. *Int. J. Robot. Res.* 33, 1462–1480. doi: 10.1177/0278364914529355

Navarro-Alarcon, D., Yip, H. M., Wang, Z., Liu, Y.-H., Zhong, F., Zhang, T., et al. (2016). Automatic 3-D manipulation of soft objects by robotic arms with an adaptive deformation model. *IEEE Trans. Robot.* 32, 429–441. doi: 10.1109/TRO.2016.2533639

Petit, A., Lippiello, V., and Siciliano, B. (2015). "Real-time tracking of 3D elastic objects with an RGB-D sensor," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg), 3914–3921. doi: 10.1109/IROS.2015.7353928

Rambow, M., Schauss, T., Buss, M., and Hirche, S. (2012). "Autonomous manipulation of deformable objects based on teleoperated demonstrations," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Algarve), 2809–2814. doi: 10.1109/IROS.2012.6386002

Roussel, O., and Ta, M. (2015). "Deformable linear object manipulation planning with contacts," in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 1–5.

Sanchez, J., Corrales, J.-A., Bouzgarrou, B.-C., and Mezouar, Y. (2018a). Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey. *Int. J. Robot. Res.* 37, 688–716. doi: 10.1177/0278364918779698

Sanchez, J., Mateo, C. M., Corrales, J. A., Bouzgarrou, B.-C., and Mezouar, Y. (2018b). "Online shape estimation based on tactile sensing and deformation modeling for robot manipulation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid), 504–511. doi: 10.1109/IROS.2018.8594314

Schulman, J., Lee, A., Ho, J., and Abbeel, P. (2013). "Tracking deformable objects with point clouds," in *2013 IEEE International Conference on Robotics and Automation* (Karlsruhe), 1130–1137. doi: 10.1109/ICRA.2013.6630714

Shah, A., and Shah, J. (2016). "Towards manipulation planning for multiple interlinked deformable linear objects," in *IEEE International Conference on Robotics and Automation* (Stockholm), 3908–3915. doi: 10.1109/ICRA.2016.7487580

Sin, F. S., Schroeder, D., and Barbič, J. (2013). Vega: non-linear FEM deformable object simulator. *Comput. Graph. Forum* 32, 36–48. doi: 10.1111/j.1467-8659.2012.03230.x

Wada, T., Hirai, S., and Kawamura, S. (1998). "Indirect simultaneous positioning operations of extensionally deformable objects," in *1998 IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications (Cat. No.98CH36190)* (Victoria) Vol. 2, 1333–1338.

Yu, W., Kapusta, A., Tan, J., Kemp, C., Turk, G., and Liu, C. (2017). "Haptic simulation for robot-assisted dressing in *IEEE International Conference on Robotics and Automation* (Singapore). doi: 10.1109/ICRA.2017.7989716

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Sanchez, Mohy El Dine, Corrales, Bouzgarrou and Mezouar. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.