Simultaneous Tracking and Elasticity Parameter Estimation of Deformable Objects

Agniva Sengupta, Romain Lagneau, Alexandre Krupa, Eric Marchand, Maud Marchal

Simultaneous Tracking and Elasticity Parameter Estimation of Deformable Objects

Agniva Sengupta*[1], Romain Lagneau*[2], Alexandre Krupa[1], Eric Marchand[1], Maud Marchal[2]

***Abstract*— In this paper, we propose a novel method to simultaneously track the deformation of soft objects and estimate their elasticity parameters. The tracking of the deformable object is performed by combining the visual information captured by a RGB-D sensor with interactive Finite Element Method simulations of the object. The visual information is more particularly used to distort the simulated object. In parallel, the elasticity parameter estimation minimizes the error between the tracked object and a simulated object deformed by the forces that are measured using a force sensor. Once the elasticity parameters are estimated, our tracking algorithm can be used to estimate the deformation forces applied to an object without the use of a force sensor. We validated our method on several soft objects with different shape complexities. Our evaluations show the ability of our method to estimate the elasticity parameters as well as its use to estimate the forces applied to a deformable object without any force sensor. These results open novel perspectives to better track and control deformable objects during robotic manipulations.**

I. INTRODUCTION

Nowadays, robots are efficient in manipulating rigid objects. However, simultaneous manipulation and tracking of deformable objects remains a challenging problem and could improve different applications, such as humanoid robots interacting with soft objects, compliance testing in product manufacturing or detection of abnormal skin stiffness in preventive healthcare. The aim of our approach is to automatically estimate the elasticity parameters of a soft object that is being deformed by the end-effector of a robot. For that purpose, we propose a closed-loop method consisting of: **a)** a deformation tracking method and **b)** an elasticity estimation algorithm. This closed-loop method allows to achieve two objectives: *STEPE* (Simultaneous Tracking and Estimation of Parameters of Elasticity) and *remote force estimation*. The required inputs of *STEPE* are a coarse 3D geometric model (mesh) of the object to track and the external measurements of the forces deforming the object. The deformation tracking method uses a depth sensor to capture the deformation and tracks it using a physics-based simulation of the object deformations. The elasticity estimation algorithm uses the result of the deformation tracking method and measurements of the deformation forces, obtained from an external sensor, to estimate the elasticity parameters of the object. Once the elasticity estimation is achieved, the parameters can be used by the deformation tracking method, thereby closing the loop. The method will iteratively converge towards a correct estimation of the elasticity parameters of the object. The elasticity parameters obtained at the convergence of the method are thereafter used for *remote force estimation*. From this stage, the deformation tracking method can be used to estimate the deformation forces acting on the object without any external force sensor.

*Co-first authors
[1]*Univ. Rennes, INRIA, IRISA, CNRS, Rennes, France*
[2]*Univ. Rennes, INSA, IRISA, Inria*
e-mail: {agniva.sengupta, romain.lagneau, alexandre.krupa, eric.marchand, maud.marchal}@irisa.fr.

A. State of the Art

1) Deformation tracking: There are two possible approaches for tracking the deformation of non-rigid objects, using either a physics-based model, or a geometric-based model for computing the deformations. Both physics-based model such as [1], [2] or geometric-based models such as [3], [4] offer efficient mechanisms to track complex 3D shapes without relying on any kind of training data. However, it is not possible to use them to iteratively update the physical parameters of the object being tracked. Zhang *et al.* used a RGB-D camera to track external forces applied to an object, but with specialized soft robots for the purpose, instead of arbitrary deformable objects [5]. The approach of Petit *et al.* [6] is more suitable for the purpose of deformation tracking using a physics-based model. However, since they rely solely on depth data to determine the direction and magnitude of deformation, their approach is more susceptible to error induced from incorrect correspondences due to heavy occlusions. Among some learning-based approaches for tracking deformation, Varol *et al.* used a constrained latent variable model for learning the deformation behavior of an object from a latent state to an output state [7]. The results from this approach have however only been validated on planar objects. Among the available literature, the approach of Frank *et al.* [8] is the closest to the method proposed in this paper for simultaneously tracking deformation and estimating parameters. The approach utilizes a tightly coupled tracking and elasticity estimation module with a point-to-point ICP [9] for pointcloud registration along with a linear, tetrahedral FEM as underlying deformation model. The use of linear FEM makes this approach not suitable when large rotational deformations occur. Additionally, point-to-point ICP cannot handle occlusion adequately, which compels them to utilize a narrow wooden manipulator to avoid occlusion. To the best of our knowledge this approach has not been used as a force estimator.

2) Physical parameter estimation: Manipulation of deformable objects is facilitated when the elasticity parameters of the objects are known. Sedef *et al.* proposed a method using a haptic device and a force sensor to estimate the

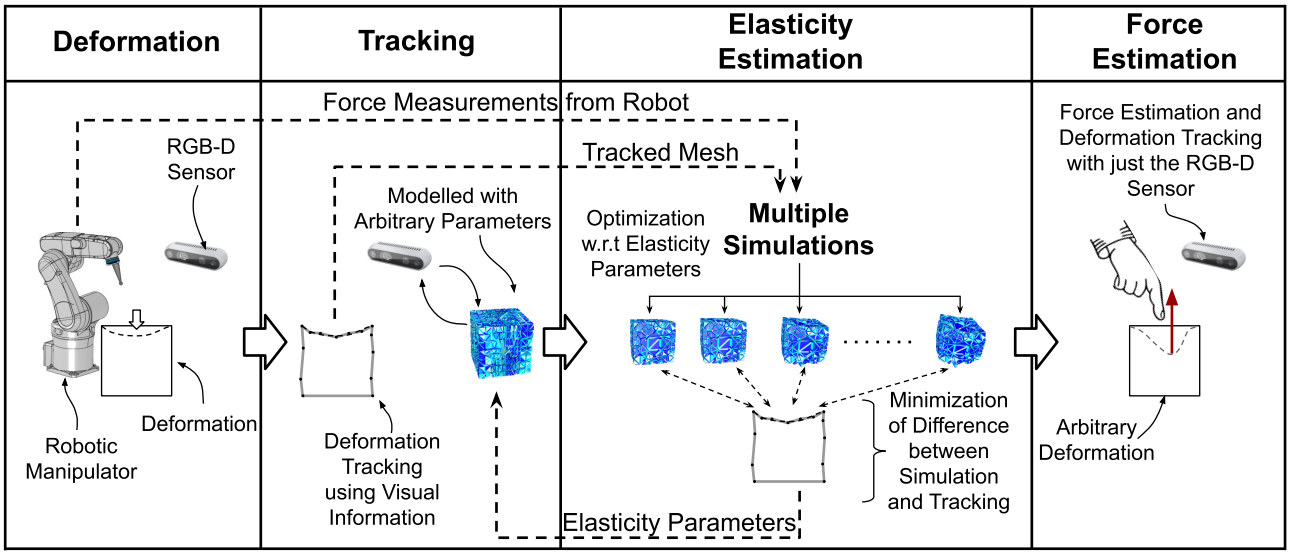

Fig. 1. Overview of the STEPE and Remote Force Estimation architecture.

linear viscoelastic parameters of a homogeneous soft tissue [10]. Bickel *et al.* proposed a method handling heterogeneous elastic objects [11]. A deformable object, observed by a marker-based stereovision system, is stimulated by a probe equipped with a force sensor to estimate the Lamé's coefficients of heterogeneous deformable objects. Data-driven methods have also been developed to determine elasticity parameters of heterogeneous elastic objects. For instance, Wang *et al.* proposed an approach that iteratively performs physics-based deformation tracking and physical parameter estimation using a two-step optimization method [12]. A method, based on the static Euler-Bernouilli deformation model along with a tracking method based on a multi-section parameterization, was proposed for simultaneous tracking and estimation of deformable object [13]. The method is based on two assumptions: on the one hand the object is homogeneous and linearly elastic, on the other hand the deformation is the bending of the object subject to gravity. However, this method has only been tested on planar object.

B. Contributions

We perform simultaneous tracking and elasticity estimation of arbitrary soft objects using the same physical model, which can be updated iteratively, as opposed to [8]. The deformation tracking does not require any external fiducial markers. A coarse model of the object is enough for both deformation tracking and elasticity parameter estimation. Contrary to [5], the *remote force estimator* does not require a prior knowledge of the elasticity parameter to estimate the deformation forces. The *STEPE* method can run on inexpensive hardware requiring only a RGB-D sensor and force measurements.

The remainder of the article is divided into three sections. The next section details the methodology for the proposed method. The experimental results are presented next, followed by discussions and perspectives about our method.

II. APPROACH FOR ELASTICITY ESTIMATION PROCESS

Our goal is to simultaneously track a deformable object and estimate its elasticity parameters. To achieve this goal, we propose a method composed of three different modules: one performing the deformation tracking, one estimating the elasticity parameters and one measuring the external deformation forces applied to the object. In *STEPE*, all the modules are involved. When doing *remote force estimation*, only the module performing the deformation tracking method is considered. The overall method is depicted in Fig. 1.

A. Modeling

We chose to model the objects using the Finite Element Method (FEM) [14]. Objects are represented using tetrahedral meshes, such as the cube depicted in Fig. 1. To avoid large rotational deformation artefacts, we use the **corotational** FEM variant of this method [15]. In this paper, we use the *Hooke's law* that relates the stress and the strain applied to an object using a linear equation that depends on the elasticity parameters of the object, that are the Young's modulus and the Poisson ratio. This law is valid only for small deformations of linear elastic objects.

Let n be the number of vertices of the mesh, $\mathbf{M} \in \mathbb{R}^{3n\times 3n}$ the mass matrix and $\mathbf{D} \in \mathbb{R}^{3n\times 3n}$ the damping matrix of the whole object, $\mathbf{K} \in \mathbb{R}^{3n\times 3n}$ the stiffness matrix of the whole object that depends on the Young's modulus and Poisson ratio of the object, $\mathbf{x}(t) \in \mathbb{R}^{3n}$ (resp $\mathbf{x}^u \in \mathbb{R}^{3n}$) the position of each vertex of the mesh at time t (respectively in the undeformed state), $\mathbf{f}^{ext} \in \mathbb{R}^{3n}$ the external forces applied to the object and $\mathbf{f}^{int} \in \mathbb{R}^{3n}$ the internal forces resulting from the deformation. The linear algebraic equation of motion of an object is given by:

$$\mathbf{M}\frac{\mathrm{d}^2}{\mathrm{d}t^2}(\mathbf{x}(t)-\mathbf{x}^u)+\mathbf{D}\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{x}(t)-\mathbf{x}^u)+\mathbf{K}(\mathbf{x}(t)-\mathbf{x}^u) = \mathbf{f}^{ext}+\mathbf{f}^{int} \quad (1)$$

This equation is used both to estimate the elasticity parameters from deformation observations and to estimate the external deformation forces from deformation observations when the elasticity parameters converged.

B. STEPE

1) External measurements module: The external measurements module is in charge of measuring the deformation forces that are applied to the object. To ensure repeatability of experiments, we decided to use a robot equipped with a force sensor both to deform the object and to measure the resulting forces. The end-effector deforms the object while recording the forces that are exerted. The end-effector motion is stopped at one point in order to maintain a static deformed state. The module stores the external force measurements and timestamps at which they were acquired to transmit them to the estimation algorithm when a steady-state has been reached in the deformation tracking.

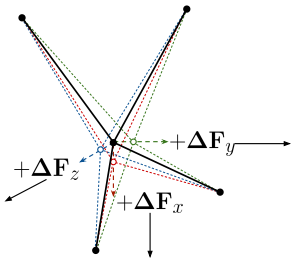

Fig. 2. Every node that needs to be tracked is perturbed by a small force along 3-axis. Computation of (4) for each deformed configuration is used to compute the gradient of the error at that particular node.

2) Deformation tracking method: In this paper, we implemented a deformation tracking method similar to the one proposed in [16]. The initial pose of the object(O) with regards to (w.r.t.) the camera (C) is obtained in the first image frame using pre-trained markers [17]. The pose of the camera w.r.t. the object, denoted by the homogeneous matrix ${}^{C}\mathbf{T}_{O}$, is updated at the beginning of each frame, using an approximate rigid object tracking method with the assumption that the object is rigid. The cost function jointly minimized for approximate rigid tracking of the object is given by:

$$\mathbf{e}^{D}({}^{k-1}\mathbf{q}_k) = ({}^{k-1}\mathbf{R}_k\,{}^{k-1}\mathbf{P} + {}^{k-1}\mathbf{t}_k)\cdot\mathbf{n}_j - d_j \tag{2}$$

$$\mathbf{e}^{K}({}^{k-1}\mathbf{q}_k) = \begin{pmatrix} x({}^{k-1}\mathbf{q}_k) - x^* \\ y({}^{k-1}\mathbf{q}_k) - y^* \end{pmatrix} \tag{3}$$

where ${}^{k-1}\mathbf{q}_k = ({}^{k-1}\mathbf{t}_k, \theta\mathbf{u})$, such that θ and $\mathbf{u}$ are the angle and axis of the rotation ${}^{k-1}\mathbf{R}_k$ from the current image frame 'k' to the previous image frame. ${}^{k-1}\mathbf{P} = (X, Y, Z)$ represents an arbitrary 3D point from the last pointcloud, which has been matched to the j-th plane of the object's 3D model, denoted by the normal vector $\mathbf{n}_j = (n_j^X, n_j^Y, n_j^Z)$ and distance to origin d_j. $(x({}^{k-1}\mathbf{q}_k), y({}^{k-1}\mathbf{q}_k))$ are the projection of the same arbitrary point in the (k-1)-th image while (x^*, y^*) are the same image points matched in the k-th image using Harris corner features.

Once an approximate estimate of ${}^{k-1}\mathbf{q}_k$ has been obtained, the non-rigid tracking is done by minimizing the point-to-plane error, given by:

$$\mathbf{e}^{N}(\mathbf{P}) = \mathbf{n}_j \cdot {}^{k-1}\mathbf{P} - d_j \tag{4}$$

The Jacobian ($\mathbf{J}$) is computed using the method shown in Fig. 2. Every node is subjected to a small force given by $\Delta\mathbf{F}_x$, $\Delta\mathbf{F}_y$ and $\Delta\mathbf{F}_z$, acting along the three axes. The displacement of all vertices of the mesh are computed by solving (1) using an Euler-Implicit mechanism of a conjugate gradient based linear solver [18]. The Jacobian $\mathbf{J}$ is computed using finite differences in the value of $\mathbf{e}^N$ obtained from the three deformed configurations. The force update is given by:

$$\Delta\mathbf{F} = -\boldsymbol{\lambda}(\mathbf{W}\mathbf{J})^{+}\mathbf{W}\mathbf{e}^{N}({}^{k-1}\mathbf{P}) \tag{5}$$

where $\mathbf{W}$ is a weighting matrix from Tukey based M-estimator [19], and $\boldsymbol{\lambda} = \begin{pmatrix}\lambda_X & \lambda_Y & \lambda_Z\end{pmatrix}$ is a scaling factor. The deformation caused in the mesh due to the Jacobian computation is discarded and the update ${}^{k-1}\mathbf{F} + \Delta\mathbf{F}$ is applied to the node, where ${}^{k-1}\mathbf{F}$ is the external force already existing on the node at the previous image frame. This Iteratively Re-weighted Least Squares scheme is allowed to repeat for a fixed number of iteration at every image frame.

3) Estimation algorithm: In this paper, we chose to use the Hooke's law evoked in II-A to represent the deformable objects. Consequently the elasticity parameters that can be estimated are the Young's modulus and the Poisson ratio. In order to estimate the elasticity parameters $\boldsymbol{\xi}$ at a contact point of a deformable object, the algorithm relies on the tracking results $\mathbf{x}^t$ and the external measurements of the deformation forces $\mathbf{f}^{meas}$ taken at the same point. Let V be the list of vertices that the camera can see denoted by v and used to compute a cost-function S, minimized using the Levenberg-Marquardt algorithm [20]. S is given by $S = \sum_{v\in V}\mathbf{e}_v(\boldsymbol{\xi})^2$. In this expression, $\mathbf{e}$ is the error function defined by $\mathbf{e}_v(\boldsymbol{\xi}) = \|\mathbf{x}_v^t - \mathbf{x}_v^s(\mathbf{f}^{meas}, \boldsymbol{\xi})\|$ and is equal to the Euclidean distance between $\mathbf{x}_v^t$ and $\mathbf{x}_v^s$, which are respectively the tracked and simulated 3D positions of v. $\mathbf{x}_v^t$ is taken at the final deformed state and $\mathbf{x}_v^s$ at the end of the simulation. The estimated parameters are the ones for which the cost-function S is minimized.

The simulation applies a force field, which replicates the forces that were recorded during the experiment, on the region surrounding the contact point location on the simulated object. The Jacobian matrix of the cost-function is needed to compute the update of the elasticity parameters between two steps k and $k+1$ of the minimization algorithm. This Jacobian matrix is numerically estimated by running several corotational FEM simulations using different values of elasticity parameters and computing the error between the experimental data and the simulation result. Let $\Delta\boldsymbol{\xi}(k)$ be the update of the elasticity parameters at the k^{th} step, $\nabla\mathbf{e}$ be the Jacobian matrix at this step of the function $\mathbf{x}^s(\mathbf{f}^{meas}, p)$ w.r.t. the elasticity parameters $\boldsymbol{\xi}$ and μ be a damping factor that is adjusted at each Levenberg-Marquardt step. The update of the elasticity parameters can be computed by solving for

$\Delta\boldsymbol{\xi}(k)$ the linear equation:

$$\Delta\boldsymbol{\xi}(k) = \left(\nabla\mathbf{e}^T\nabla\mathbf{e} + \mu\ \mathrm{diag}(\nabla\mathbf{e}^T\nabla\mathbf{e})\right)^{-1}\left(\nabla\mathbf{e}^T\mathbf{e}\right) \quad (6)$$

$$\boldsymbol{\xi}(k+1) = \boldsymbol{\xi}(k) + \Delta\boldsymbol{\xi}(k) \quad (7)$$

In this paper, we focus on estimating the Young's modulus E. The Poisson ratio is assumed to be known a priori. Thus, the elasticity parameters vector is given by $\boldsymbol{\xi} = (E) \in \mathbb{R}$. At this stage, the method permits to estimate the Young's modulus $\hat{E}$ of a deformable object from deformation observations and external force measurements.

C. Remote force estimation

When doing *remote force estimation*, the deformation tracking module can be used without external force sensor. Let A be the set of vertices representing the active region of the object. The active region is an approximation of the surface of the model where the deformation is happening. This region is determined using some practical heuristics e.g., neither the vertices lying on the ground plane nor the vertices not visible from the camera are taken into account. Active vertices, denoted by a, are the vertices belonging to the active region. The method is able to estimate the external deformation forces, given by $\widehat{\mathbf{f}^{ext}} = \sum_a \widehat{\mathbf{f}_a^{ext}}$, which are applied to the object in the active region, as shown in Fig. 3.

The estimated forces are projected onto the normal of the active surface $\mathbf{n}^A = \frac{\sum_a \mathbf{n}_a}{\|\sum_a \mathbf{n}_a\|}$ because the tracking method cannot have information about tangential forces, since (4) is a point-to-plane distance. This produces the following estimate of the magnitude of the deformation forces: $\|\widehat{\mathbf{f}^{ext}}\| \approx \widehat{\mathbf{f}^{ext}} \cdot \mathbf{n}^A$.

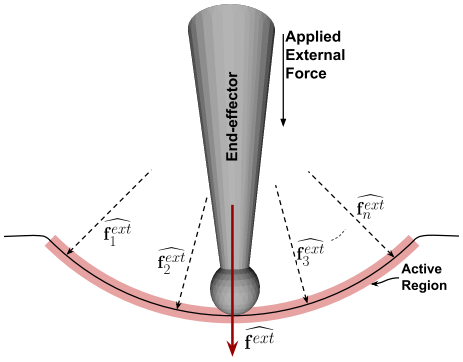

Fig. 3. Force estimation in the active region.

III. EXPERIMENTS

A. Experimental setup

Our experimental setup shown in Fig. II-C consists of a 6-DOF anthropomorphic robot arm (a Viper 850 from ADEPT) equipped with a ATI's Gamma IP65 force/torque sensor and a 3D-printed stylus used as an end-effector distal tool. A RGB-D Intel Realsense D435 camera is used for the tracking. Our method has been implemented on a computer that has an Intel Xeon CPU working at 3.70GHz with 16 logical cores. Our method has been tested with different soft objects:

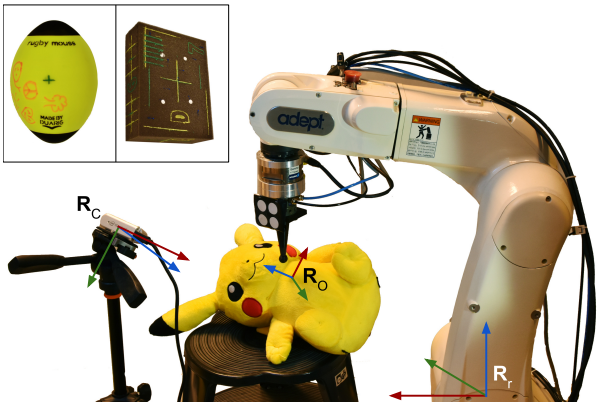

Fig. 4. Experimental setup showing the robot (R_r) deforming a plush toy (R_o) while the data is captured from an Intel RealSense D435 RGB-D Camera (R_c.) The other objects that have been experimented upon are shown in the inset above.

a foam block, a soft ball and a complex-shaped plush toy. The transformation between the robot frame and the object frame is computed once at the initialization of our method. Thereafter, it is used to express measured forces into the object coordinate frame. The mesh of the plush toy has been generated by photogrammetry using the Meshroom software [21], [22]. The corotational FEM simulations are performed using SOFA framework [18].

B. Results

The following section describes a series of experimental conditions and their corresponding results[1]. Each *condition* represents a particular position of a specific object, e.g., the first *condition* describes the rectangular foam being deformed at the center of its largest face, while being placed horizontally on a table. To evaluate the accuracy of the estimation process, the ground truth of Young's modulus E_{GT} has been determined through indentation tests for each condition of experiment [23]. The indentation tests are repeated, slow and incremental vertical displacements applied to the objects using the robot while measuring the forces and the displacements. Forces are measured from the force sensor while displacements are obtained using the odometry of the robot. For each condition of experiment, the average of several indentation tests is taken as ground truth $\overline{E_{GT}}$.

Let $\hat{E}(i)$ be the estimated Young's modulus obtained from the i^{th} deformation tracking results. The Young's modulus that is thereafter used for the external force estimation, denoted by E_{est}, is the one for which the following convergence criterion is respected:

$$\frac{\hat{E}(i) - \hat{E}(i-1)}{\hat{E}(i-1)} < 0.05 \quad (8)$$

Some examples of the output of the deformation tracking method are shown in Fig. 5. The mean of the norm of the error ($\|\overline{\mathbf{e}^N}\|$) was found to be 1.53 mm, 0.51 mm and 0.21 mm for the foam block, soft ball and plush toy, while the standard

[1]Additional material can be found at https://youtu.be/k1MPnmqmovQ

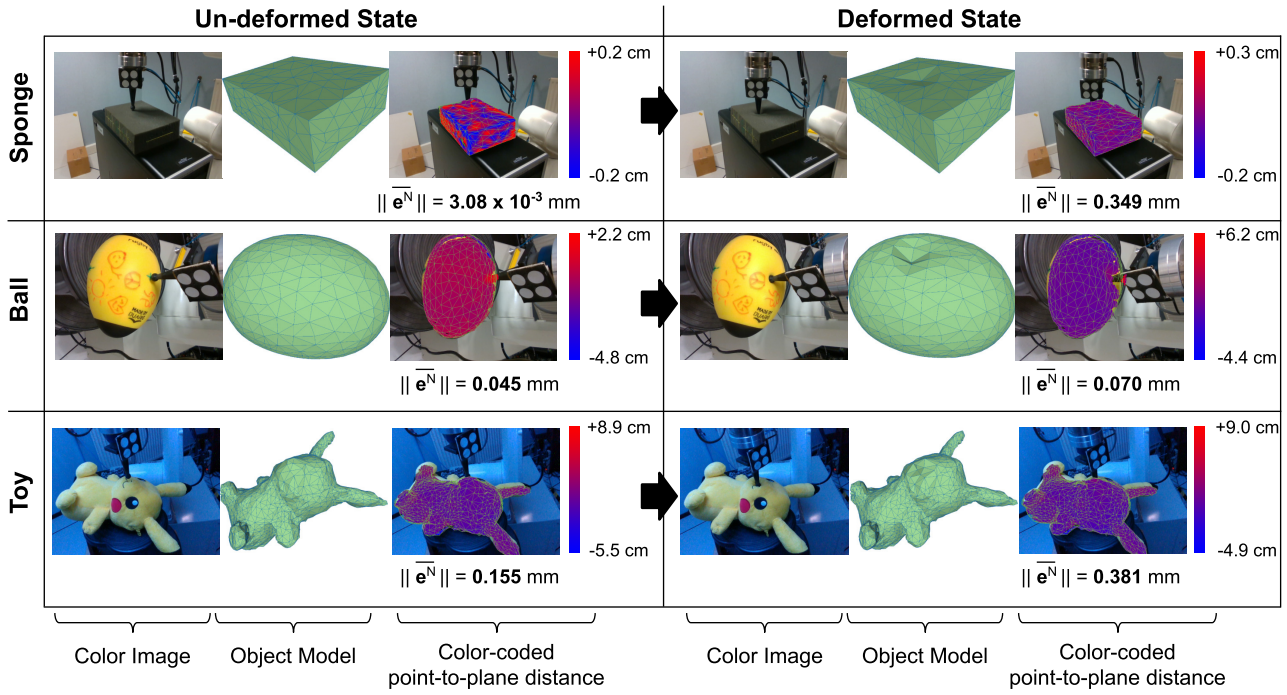

Fig. 5. Tracking of the objects for estimating their elasticity parameters. The first and the fourth columns show the color image, the second and the fifth columns show the tracked object model and the third and the sixth columns show the value of $\mathbf{e}^N$, color-coded and augmented on the object

deviations were 7.2 mm, 0.7 mm and 1.4 mm respectively, across all the experiments reported in this article. In the three example sequences shown in Fig. 5, the mean of the norm of error varied between a maximum of 0.381 mm (for the plush toy, deformed by its nose) to as low as 3 μm (for the undeformed sponge block), despite having outlying correspondence or noise in the range of -5.5 cm to +9.0 cm (which gets rejected by the M-estimator). In the experiments shown in Table I, the time required for tracking varies between 1.21 to 3.1 sec/frame. However, it was possible to run the same algorithm on the foam block at $\sim$ 800 ms/frame without using any GPU based computation, with negligible loss of accuracy ($<$ 10%).

Our method can be evaluated on different criteria. The first set of experiments was conducted to evaluate the time-performance of the tracking and the quality of the Young's modulus estimation w.r.t. the number of vertices of both the visual mesh and the mechanical mesh. The results of this set of experiments are grouped in Table I. In this table, #Visual designates the number of vertices of the visual mesh while #Mechanical designates the number of vertices of the mechanical mesh. $\overline{t_{tr}}$ is the average deformation tracking time, std(t_{tr}) is its standard deviation, t_{st} is the time that was required to deform the object and reach a steady-state and t_{est} is the time to estimate the Young's modulus. All the times are expressed in seconds. $\overline{E_{GT}}$ and E_{est} correspond to the average ground truth of Young's modulus and estimated Young's modulus respectively, both expressed in kilo Pascals. Finally, Error designates the percentage of error of the estimation and is given by $Error = 100 * abs(\overline{E_{GT}} - E_{est})/\overline{E_{GT}}$ where abs designates the absolute value.

The second set of experiments, whose results are grouped in Table II, was conducted to evaluate the consistency of the Young's modulus estimation w.r.t. the initial estimate E_0. Experiments have been conducted on two locations of the foam and the plush toy, as indicated in the first column

TABLE I

EVALUATION OF TRACKING AND ESTIMATION TIMES AND YOUNG'S MODULUS (IN KPA) ESTIMATION ACCURACY W.R.T. DIFFERENT QUALITY OF VISUAL AND MECHANICAL MESHES.

Objects	#Visual	#Mechanical	$\overline{t_{tr}}$ (s)	std(t_{tr}) (s)	t_{st} (s)	t_{est} (s)	$\overline{E_{GT}}$	E_{est}	Error (%)
foam	35	1049	1.94	0.14	62	310	454	431	5.08
foam	415	1049	2.37	0.15	122	180	454	438	3.62
foam	178	554	1.21	0.02	60	180	454	497	9.47
ball	404	627	1.38	0.03	126	70	156	136	12.8
ball	404	1060	1.61	0.02	118	96	156	149	5.0
ball	404	1954	3.09	0.07	130	330	156	148	5.1

TABLE II

EVALUATION OF THE ESTIMATION OF THE YOUNG'S MODULUS (IN KPA).

Objects	E_0	$\overline{E_{GT}}$	E_{est}	**Error**
foam middle	200	454	480	5.7%
	1500	454	451	0.7%
	15000	454	488	7.5%
foam corner	200	247	224	9.3%
ball	1000	156	154	1.3%
toy nose	200	51	53	3.7%
toy leg	50	42	43.1	2.6%

of Table II, to evaluate if the method can handle non-homogeneous objects. Low initial estimate are of the order of $0.1\times\overline{E_{GT}}$ and high initial estimate of the order of $10\times\overline{E_{GT}}$.

TABLE III

COMPARISON OF THE AVERAGE NORM OF THE MEASURED AND ESTIMATED FORCES.

Objects	**Orientation**	**Depth (cm)**	$\overline{f_{GT}}$ **(N)**	f_{est} **(N)**	Δf **(N)**
foam	horizontal	2	18.8	20.6	1.8
foam	tilted	2	21.36	23.11	1.75
foam	horizontal	4	45.46	42.67	-2.79
foam	tilted	4	38.94	35.59	-3.35
ball	horizontal	2.5	9.49	8.57	-0.92
toy	horizontal	2	5.32	5.81	0.49

The last set of experiments was conducted to evaluate the estimation of the deformation forces in different conditions of experiments. For these experiments, the estimated Young's modulus values summarized in Table II were used by the deformation tracking method. The results of this set of experiments are grouped in Table III. $\overline{f_{GT}}$ and f_{est} designate the average norm of the ground truth forces measured by the force sensor and the norm of the forces estimated by the deformation tracking method respectively, expressed in Newtons. $\Delta f = f_{est} - \overline{f_{GT}}$ represents the error between the estimated and ground truth forces. The estimated forces are projected onto the measured forces when compared with the ground truth. The force experiments have been conducted with different orientations of the objects (horizontal and tilted by approximately 25 degrees w.r.t. the z-axis of the object frame). Different depths of stimulation have also been tested and are indicated in the depth column of Table III.

Finally, Fig. 6 depicts a possible use case of our method to estimate deformation forces exerted on an object in an environment where no force measurements are available.

IV. DISCUSSION

The first set of experiments whose results are summarized in Table I shows that the accuracy of the Young's modulus estimation is improved as the resolution of the mechanical mesh increases. This results from the fact that the deformation can be represented with finer details and is thus closer to the reality. Table II shows that our method is able to estimate the Young's modulus of an arbitrary object even when the initial estimate is much different from the ground truth. A future work would consist in using our method to estimate both Poisson ratio and Young's modulus within an iterative process. Table III shows that the estimated Young's modulus could be used by our method to accurately estimate the deformation forces from visual information only. The use case scenario depicted in Fig.6 shows one possible application of our method in environments where no force sensor could be used, for instance in human-performed manipulation tasks. This application opens novel perspectives for better estimating the forces applied to a deformable object by different operators, as well as on a cooperative task on such objects.

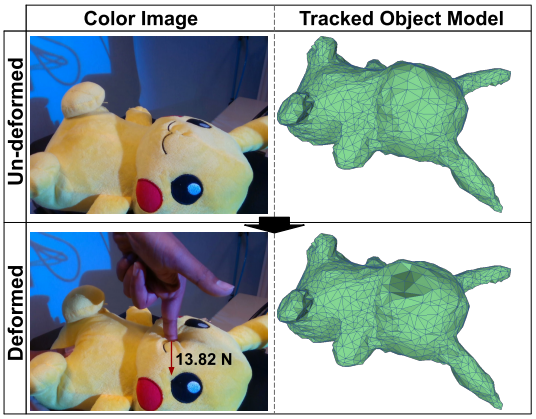

Fig. 6. Force estimation while the plush toy gets freely deformed by hand. The deformation forces (13.8 N) are in the expected range.

V. CONCLUSION

In this paper, we proposed a method for simultaneously tracking arbitrary linear elastic objects and estimating their elasticity parameter. For estimating the elasticity parameters, our method requires only a coarse 3D mesh of the object and external force measurements. Deformation tracking and elasticity parameter estimation are run alternatively. We showed experimentally that the method is able to correctly estimate the Young's modulus of an arbitrary soft object, even when starting from a highly coarse initial estimate. Finally, we demonstrated that our method could be used for remotely estimating the deformation forces applied to an arbitrary object using visual information only. These results pave the way for better control of deformable objects manipulated by a robot.

ACKNOWLEDGMENTS

We gratefully acknowledge the support from *The Research Council of Norway* through participation in iProcess - 255596 project. This research has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 731761 for project "Imagine".

References

[1] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 343–352.

[2] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "Volumedeform: Real-time volumetric non-rigid reconstruction," in *Proceedings of European Conference on Computer Vision*. Springer, 2016, pp. 362–379.

[3] S. Bronte, L. Bergasa, D. Pizarro, and R. Barea, "Model-based real-time non-rigid tracking," *Sensors*, vol. 17, no. 10, p. 2342, 2017.

[4] S. Bronte, M. Paladini, L. M. Bergasa, L. Agapito, and R. Arroyo, "Real-time sequential model-based non-rigid sfm," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, 2014, pp. 1026–1031.

[5] Z. Zhang, A. Petit, J. Dequidt, and C. Duriez, "Calibration and external force sensing for soft robots using an rgb-d camera," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2356–2363, 2019.

[6] A. Petit, S. Cotin, V. Lippiello, and B. Siciliano, "Capturing deformations of interacting non-rigid objects using rgb-d data," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, 2018, pp. 491–497.

[7] A. Varol, M. Salzmann, P. Fua, and R. Urtasun, "A constrained latent variable model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2012, pp. 2248–2255.

[8] B. Frank, R. Schmedding, C. Stachniss, M. Teschner, and W. Burgard, "Learning the elasticity parameters of deformable objects with a manipulation robot," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems*, 2010, pp. 1877–1883.

[9] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–606.

[10] M. Sedef, E. Samur, and C. Basdogan, "Real-time finite-element simulation of linear viscoelastic tissue behavior based on experimental data," *IEEE Computer Graphics and Applications*, vol. 26, no. 6, 2006.

[11] B. Bickel, M. Bächer, M. A. Otaduy, W. Matusik, H. Pfister, and M. Gross, "Capture and modeling of non-linear heterogeneous soft tissue," in *Proceedings of the ACM Conference of Transactions on Graphics*, vol. 28. ACM, 2009, p. 89.

[12] B. Wang, L. Wu, K. Yin, U. Ascher, L. Liu, and H. Huang, "Deformation Capture and Modeling of Soft Objects," *ACM Transaction on Graphique*, vol. 34, no. 4, pp. 94:1–94:12, 2015.

[13] A. R. Fugl, A. Jordt, H. G. Petersen, M. Willatzen, and R. Koch, "Simultaneous estimation of material properties and pose for deformable objects from depth and color images," in *Proceedings of Joint German Association for Pattern Recognition and OAGM Symposium*. Springer, 2012, pp. 165–174.

[14] A. Nealen, M. Muller, R. Keiser, E. Boxerman, and M. Carlson, "Physically based deformable models in computer graphics," *Computer Graphics Forum*, vol. Vol 25, pp. 809–836, 2006.

[15] M. Muller and M. Gross, "Interactive virtual materials," in *Proceedings of Graphics interface*. Canadian Human-Computer Communications Society, 2004, pp. 239–246.

[16] A. Sengupta, A. Krupa, and E. Marchand, "Tracking of non-rigid objects using rgb-d camera," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 2019.

[17] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 12, pp. 2633–2651, 2015.

[18] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin, "SOFA: A Multi-Model Framework for Interactive Physical Simulation," in *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, ser. Studies in Mechanobiology, Tissue Engineering and Biomaterials. Springer, 2012, vol. 11, pp. 283–321.

[19] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim, "Robust regression methods for computer vision: A review," *International journal of computer vision*, vol. 6, no. 1, pp. 59–70, 1991.

[20] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis*. Springer, 1978, pp. 105–116.

[21] P. Moulon, P. Monasse, and R. Marlet, "Adaptive structure from motion with a contrario model estimation," in *Proceedings of the Asian Computer Vision Conference*. Springer Berlin Heidelberg, 2012, pp. 257–270.

[22] M. Jancosek and T. Pajdla, "Multi-view reconstruction preserving weakly-supported surfaces," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[23] C. T. McKee, J. A. Last, P. Russell, and C. J. Murphy, "Indentation versus tensile measurements of young's modulus for soft biological tissues," *Tissue Engineering Part B: Reviews*, vol. 17, no. 3, pp. 155–164, 2011.