



HAL
open science

Vision-Based Sensing of External Forces Acting on Soft Robots Using Finite Element Method

Zhongkai Zhang, Jérémie Dequidt, Christian Duriez

► **To cite this version:**

Zhongkai Zhang, Jérémie Dequidt, Christian Duriez. Vision-Based Sensing of External Forces Acting on Soft Robots Using Finite Element Method. *IEEE Robotics and Automation Letters*, 2018, 3 (3), pp.1529 - 1536. 10.1109/LRA.2018.2800781 . hal-01720645

HAL Id: hal-01720645

<https://inria.hal.science/hal-01720645v1>

Submitted on 1 Mar 2018

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.

Vision-Based Sensing of External Forces Acting on Soft Robots Using Finite Element Method

Zhongkai Zhang, Jeremie Dequidt, Christian Duriez

Abstract—In this paper, we propose a new framework of external force sensing for soft robots based on the fusion of vision-based measurements and Finite Element Model (FEM) techniques. A precise mechanical model of the robot is built using real-time FEM to describe the relationship between the external forces acting on the robot and the displacement of predefined feature points. The position of these feature points on the real robot is measured using a vision system and is compared with the equivalent feature points in the finite element model. Using the compared displacement, the intensities of the external forces are computed by solving an inverse problem. Based on the developed FEM equations, we show that not only the intensities but also the locations of the external forces can be estimated. A strategy is proposed to find the correct locations of external forces among several possible ones. The method is verified and validated using both simulation and experiments on a soft sheet and a parallel soft robot (both of them have non-trivial shapes). The good results obtained from the experimental study demonstrate the capability of our approach.

I. INTRODUCTION

Soft robots can interact with the environment in a safe and compliant way because of their deformable structures. The property of force sensing increases their autonomy and safe level for object manipulation and medical applications. A general strategy to make the robot feel its environment is realized by integrating external force sensors into the soft body. Many kinds of sensors have been designed to adapt to the deformable structures, such as sensors in [1], [2], [3].

However, the integration of external force sensors could increase the complexity and the volume of mechanical structures. To deal with this drawback, force sensing by the soft robot itself provides a promising strategy for applications with possible image monitoring (for instance: MRI, scan or ultra-sound for medical applications) and limited workspace. It is challenging to compute external force applied to soft body without using external force sensors, due to complicated geometrical shape and lack of sensing feedback. Several methods have been developed for external force estimation. However, they highly depend on the equations of soft robots and their applications are limited. There is no general strategy which can be employed for soft robots with all the features like complex structure, 3D workspace, force sensing on the whole body, and multi-force actuation.

The idea of this paper is to show a general method to sense the external force in terms of intensity and location using the

deformation of soft structures. Computation of the compliance matrix between external forces and feature points (effectors) on the soft structure is a bottleneck to realize the force sensing. Besides, the coupling between the control inputs and the external forces should be taken into consideration when the robot is actuated. Traditionally, constant curvature model (CCM) [4] and non-constant curvature model (NCCM) [5] are two popular methods to model soft robot. However, they can only model a reduced class of deformable robot, in particular the robot with curve-like shape. In [6], the object's structure and flexibility are modelled by a chain of rigid bodies connected by torsional springs. Finite Element Model (FEM) provides a general method to model soft robots. Both the deformation analysis [7] and real-time simulation [8] are available using FEM. Thanks to the real-time FEM¹, the compliance matrix can be obtained on real-time by numerical computation.

For the implementation of our strategy, the 3D position of several feature points on the soft robot should be measured. In this paper, we also propose a strategy to track multi-points on soft robot. The feature points are tracked by an optical tracking system and their indexes are allocated based on the FEM of soft robot. This strategy increases the robustness of the system with the ability to detect the hidden points.

To our best knowledge, this is the first paper which addresses the real-time external force sensing of soft robot based on FEM. Three main contributions are shown in this paper: (1) we propose a novel strategy for feature point allocation for soft robots using FEM. (2) We propose a real-time method to compute the external forces on soft robot without using force sensors. The strategy is implemented based on the inverse model. (3) We prove that both the intensity and the locations of external forces can be estimated only using the feature points on soft robot. Then we propose a strategy to estimate the unknown locations of external forces.

This paper is organized as follows. In Section II, the related research about force sensing of soft robot is reviewed. The discrete-time equation of soft robot is introduced in Section III based on real-time Finite Element Method. Section IV presents the method to compute the intensity of external forces with known locations. Then in Section V, the implementation of location estimation of external forces is investigated. The validations of the external force sensing algorithm using simulation and real experiments are presented in Section VI and Section VII, respectively. Finally, discussion and conclusion are shown in Section VIII and Section IX.

This project has been financed by the Conseil Regional Haut-de-France, by Inria and by the European Union through the European Regional Development Fund (ERDF). The authors are with INRIA, University of Lille, France. Contact: {zhongkai.zhang, jeremie.dequidt, christian.duriez}@inria.fr

¹SOFA framework: <http://www.sofa-framework.org/>

II. RELATED RESEARCH

The proposed algorithm for external force sensing is based on FEM. Therefore, in this section, the related researches about both force sensing and control of soft robotics using real-time FEM are reviewed.

In the field of rigid robotics research, the torque of joint motors can be used to compute the external forces applied on the tip of the robot [9], [10]. Being compliant, soft robots are deformed when external forces are applied. One can then try to capture the location, the intensity of external forces based on the deformation. In [11], certain components of an applied end-effector wrench can be determined by sensing the actuation loads. A probabilistic approach is described in [12] to estimate forces based on pose measurements. In [13], the force is estimated using the measured tension force and the pose of end effector. To increase the computation efficiency, [14] employs pose measurements to estimate forces based on a low-dimensional model. Using the information of tip pose and shape measurements, a 3D force estimation platform is designed in [15] for tendon-driven catheters based on the Cosserat rod model and Kalman filter. Using the pose of the robot tip and the tensions of the driving cables, [16] proposes a force sensing method based on the shape reconstruction algorithm and a robot kinematic-static model. These approaches are often limited: application to 2D configuration, loads just on the tip, very simple geometrical shapes, etc. These limitations are due to the simplifications of the models. In this paper, we explore a general method to estimate the external point force based on FEM. In our approach, only a set of feature point positions on the soft robot needs to be measured at each sampling time.

Most model-based control strategies are proposed based on the CCM [17] and NCCM [18]. In order to provide a general method to control soft robot, an open-loop controller is designed based on the FEM [19], [8]. To increase the control performances, a closed-loop control strategy is proposed in [20] where a predictor is designed to get the value of Jacobian matrix. Then in [21], the robust stability is analyzed and a switched control strategy is proposed by combining the open-loop controller and closed-loop controller. The strategies are available to control soft robots with complicated geometrical shapes. One advantage of these strategies based on FEM is the computation of a precise compliance matrix, between the control inputs and outputs. Therefore, in this paper, the strategy to get the compliance matrix is employed for external force sensing.

III. DISCRETE-TIME KINEMATIC EQUATION OF SOFT ROBOT

In this section, we introduce the discrete-time kinematic equation of soft robot based on real-time Finite Element Method [20], [21]. With the assumption of ignoring inertia, the kinematic equation is obtained by computing a succession of quasi-static problems. The main feature of this method is the mapping of the equilibrium equation into the constraint space so that its dimension can be reduced and the real-time computation is possible.

Based on FEM, the computation of soft robotic configurations is implemented step by step in the workspace of soft robots. Assuming that the position vector of all FEM nodes is \mathbf{x} which varies according to the configuration of the soft robots at each sampling time, the quasi-static equilibrium function of the entire robot at the $(k+1)$ th sampling time is given by:

$$\mathbf{H}_f^T(\mathbf{x}_{k+1})\lambda_{f,k+1} - \mathbf{f}(\mathbf{x}_{k+1}) + \mathbf{H}_a^T(\mathbf{x}_{k+1})\lambda_{a,k+1} = 0 \quad (1)$$

where $\mathbf{f}(\mathbf{x})$ represents the internal stiffness force. $\mathbf{H}_a^T\lambda_a$ represents the contribution of the external controlled inputs (where \mathbf{H}_a is the mapping matrix between nodes and directions of controlled inputs and λ_a is the contribution vector of controlled inputs). $\mathbf{H}_f^T\lambda_f$ gathers the contribution of the external forces (where \mathbf{H}_f is the mapping matrix between nodes and external force directions and λ_f is the contribution vector of external forces).

In (1), the internal stiffness force $\mathbf{f}(\mathbf{x})$ is non-linear. At the $(k+1)$ th step, they are expressed as:

$$\mathbf{f}(\mathbf{x}_{k+1}) \approx \mathbf{f}(\mathbf{x}_k) + \mathbf{K}(\mathbf{x}_k)d\mathbf{x}_{k+1} \quad (2)$$

where the tangent stiffness matrix $\mathbf{K}(\mathbf{x}_k)$ is highly sparse and depends on the actual position of the nodes \mathbf{x} . $d\mathbf{x}_{k+1}$ is the displacement between consecutive configurations ($d\mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k$).

Position vectors of actuators (the nodes where the external controlled inputs act on) and effectors (feature points) are defined as δ_a and δ_e , respectively. Combining (1) and (2), the equilibrium equation of soft robots at each step can be established. To reduce its size, the equation can be projected into the constraint space.

$$\delta_{e,k+1} = \mathbf{W}_{ea}(\mathbf{x}_k)\lambda_{a,k} + \mathbf{W}_{ef}(\mathbf{x}_k)\lambda_{f,k} + \delta_{e,k}^{free} \quad (3)$$

$$\delta_{a,k+1} = \mathbf{W}_{aa}(\mathbf{x}_k)\lambda_{a,k} + \mathbf{W}_{af}(\mathbf{x}_k)\lambda_{f,k} + \delta_{a,k}^{free} \quad (4)$$

where $\mathbf{W}_{ij}(\mathbf{x}_k) = \mathbf{H}_i(\mathbf{x}_{k+1})\mathbf{K}^{-1}(\mathbf{x}_k)\mathbf{H}_j^T(\mathbf{x}_{k+1})$ ($i = e, a$, $j = a, f$, and \mathbf{H}_e is the mapping matrix between nodes and effector directions). δ_e^{free} and δ_a^{free} are respectively position vectors of effectors and actuators when $\lambda_a = \lambda_f = 0$. Using $\mathbf{W}_{ea}(\mathbf{x})$, we can get a measure of the mechanical coupling between effectors and actuators. $\mathbf{W}_{ef}(\mathbf{x})$ measures the coupling between effectors and external forces. The mechanical coupling between actuators and between actuators and external forces are represented by matrices $\mathbf{W}_{aa}(\mathbf{x})$ and $\mathbf{W}_{af}(\mathbf{x})$, respectively.

With the continuity assumption of $\mathbf{W}_{ea}(\mathbf{x}_{k-1}) \approx \mathbf{W}_{ea}(\mathbf{x}_k)$ and $\mathbf{W}_{ef}(\mathbf{x}_{k-1}) \approx \mathbf{W}_{ef}(\mathbf{x}_k)$, (3) can be simplified as [21]:

$$\delta_{e,k+1} = \delta_{e,k} + \mathbf{W}_{ea}(\mathbf{x}_k)\Delta\lambda_{a,k} + \mathbf{W}_{ef}(\mathbf{x}_k)\Delta\lambda_{f,k} \quad (5)$$

where $\Delta\lambda$ is the incremental vector of λ .

At the next sampling time, the configuration of the robot is updated by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{K}_k^{-1}\mathbf{H}_a^T\Delta\lambda_{a,k} + \mathbf{K}_k^{-1}\mathbf{H}_f^T\Delta\lambda_{f,k} \quad (6)$$

so that the compliance matrices can be computed at each sampling time.

Remark 1. The tangent stiffness matrix $\mathbf{K}(\mathbf{x})$ is positive definite when the robot is constrained to have no rigid body motion. By placing the actuators on different nodes of the FEM or with different directions, there will be no linear dependency between lines of \mathbf{H}_a , \mathbf{H}_e and \mathbf{H}_f so that the matrices $\mathbf{W}_{aa}(\mathbf{x})$ is positive-definite, $\mathbf{W}_{ea}(\mathbf{x})$ and $\mathbf{W}_{ef}(\mathbf{x})$ have full row rank when the number of lines in \mathbf{H}_e is smaller than the number of lines in $\mathbf{H}_a(\mathbf{H}_f)$. In the opposite case, $\mathbf{W}_{ea}(\mathbf{x})$ and $\mathbf{W}_{ef}(\mathbf{x})$ are matrices of full column rank.

IV. EXTERNAL FORCE COMPUTATION

In this section, the strategy to compute external forces for soft robot is shown in detail. We introduce an overview of the implementation (see Fig. 1) in the first subsection. Then, in the second subsection, the sufficient condition to realize the external force computation is investigated and the forces are computed by solving an inverse problem using an optimization-based strategy.

A. Overview of Implementation

The implementation of this method requires a FEM of soft robot and a tracking system. Based on the constitutive law of the material (in this study, we have considered a hookean material), the FEM provides the compliance matrix between the external forces and the effectors. The tracking system is used to get the 3D position of effectors which are defined in both the soft robot and its FEM. Based on the position error of effectors on soft robot and its FEM, the external force can be computed at each sampling time.

The computed external force u_f , together with the control input u_c , contributes to update the configuration of the FEM. It is noted that control inputs for the soft robot and its FEM are same so that their influences on both systems can be compensated. However, to compute the incremental external forces within each step, the coupling between the external forces and control inputs should be considered.

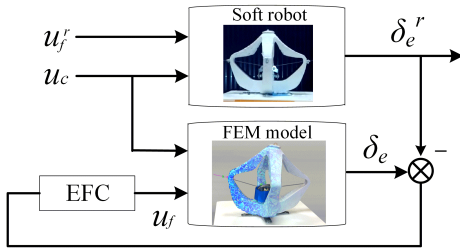


Fig. 1. The strategy of external force computation for soft robot. EFC (external force computation) module is the algorithm to compute external forces. u_f^r and u_c are the external forces and the control input of the soft robot. u_f is the computed external force using EFC. δ_e^r and δ_e are the position vector of feature points for soft robot and its FEM, respectively.

The advantages of our strategy are: (1) it can be employed to compute external forces of soft robots with complicated structures. (2) The implementation does not need the whole shape of the robot or the actuator force sensor. (3) The input of actuators is considered during the computation of external forces. (4) What we need is to build a FEM of the soft robot instead of mathematical analysis.

B. Algorithm of External Force Computation

In Fig. 1, the sufficient condition is needed for the external force computation and is shown as the following theorem:

Theorem 2. *Given the locations of external forces, the intensity can be computed if $m \geq n$ where m and n are respectively the number of rows and columns in the compliance matrix \mathbf{W}_{ef} .*

Proof: For the implementation of strategy shown in Fig.1, we have $\Delta\lambda_{a,k} = \Delta\lambda_{a,k}^r$ where the superscript r represents the corresponding variables for real robot. We assume that the FEM is accurate so that we have $\mathbf{W}_{ea}^r(\mathbf{x}_k) = \mathbf{W}_{ea}(\mathbf{x}_k)$ and $\mathbf{W}_{ef}^r(\mathbf{x}_k) = \mathbf{W}_{ef}(\mathbf{x}_k)$.

Without considering the computation delay of the implementation, from (5), we have

$$\Delta\delta_{e,k}^r - \Delta\delta_{e,k} = \mathbf{W}_{ef}(\mathbf{x}_k) (\Delta\lambda_{f,k}^r - \Delta\lambda_{f,k}) \quad (7)$$

where $\mathbf{W}_{ef} \in \mathbb{R}^{m \times n}$ has full row rank ($m \geq n$) or full column rank ($m \leq n$).

In order to make sure that $\Delta\lambda_{f,k}^r - \Delta\lambda_{f,k}$ have unique zero solution when $\Delta\delta_{e,k}^r = \Delta\delta_{e,k}$, we need $m \geq n$. ■

It is easy to make the condition hold by choosing more feature points as effectors. Therefore, the task for external force computation can be converted to compute $\Delta\lambda_f$ so that δ_e can track its desired value δ_e^r . In the following, we propose an optimization-based method to compute the external forces.

At each sampling time, the coupling between the external forces and actuators is considered for the computation. We rewrite the equation of the FEM as

$$\delta_{e,k+1} = \delta_{e,k} + \mathbf{W}_{el}(\mathbf{x}_k) \Delta\lambda_{I,k} \quad (8)$$

where $\mathbf{W}_{el} = [\mathbf{W}_{ea} \quad \mathbf{W}_{ef}]$ and $\Delta\lambda_I = [\Delta\lambda_a \quad \Delta\lambda_f]^T$.

To get the contribution of external forces, the optimization problem can be defined as $\Delta\lambda_{f,k} = \arg \min \Gamma(\Delta\lambda_{I,k})$ where $\Gamma(\Delta\lambda_{I,k})$ is the objective function:

$$\Gamma(\Delta\lambda_{I,k}) = \frac{1}{2} \| \mathbf{W}_{el}(\mathbf{x}_k) \Delta\lambda_{I,k} - (\delta_{e,k} - \delta_{e,k}^r) \|^2 \quad (9)$$

In the case of constrained optimization, the numerical optimization algorithm (like active set method) is necessary to be employed. In order to match the standard quadratic problem (QP) formulation for the usage of software packages [22], (9) can be converted to be

$$\min_{\Delta\lambda_{a,k}, \Delta\lambda_{f,k}} \frac{1}{2} (\Delta\lambda_{I,k})^T \mathbf{Q}_k \Delta\lambda_{I,k} + \mathbf{c}_k^T \Delta\lambda_{I,k} \quad (10)$$

$$\text{s.t. } \Delta\lambda_{a,k} = \Delta\lambda_{a,k}^r \quad (11)$$

$$\text{or } \Delta\delta_{a,k} = \Delta\delta_{a,k}^r \quad (12)$$

where $\mathbf{Q}_k = \mathbf{W}_{el}^T(\mathbf{x}_k) \mathbf{W}_{el}(\mathbf{x}_k)$ and $\mathbf{c}_k^T = -(\delta_{e,k} - \delta_{e,k}^r)^T \mathbf{W}_{el}$ with the condition $m \geq n$. $\Delta\lambda_{a,k}^r$ or $\Delta\delta_{a,k}^r$ are the control inputs generated by the motion controller. For robots actuated by air pressure, $\Delta\lambda_{a,k}$ is usually chosen as the control input. For robots actuated by cables, $\Delta\delta_{a,k}$ can also be chosen as control input and it can be converted into the force constraint of $\Delta\lambda_{a,k}$ using (4).

For soft objects without actuators and other constraints,

(9) has an analytical solution which can be obtained by defining the Hamiltonian H as $H = \Gamma(\Delta\lambda_{f,k})$. $\Delta\lambda_{f,k}$ can be computed from $dH/d(\Delta\lambda_{f,k}) = 0$ as

$$\Delta\lambda_{f,k} = (\mathbf{W}_{ef}^T \mathbf{W}_{ef})^{-1} \mathbf{W}_{ef}^T (\delta_{e,k} - \delta_e^r) \quad (13)$$

The condition $m \geq n$ in Theorem 2 is sufficient to compute the external forces. It is noted that m and n correspond to the number of feature points and that of the external forces, respectively. Usually, more feature points are employed so that the accurate forces can be computed even if some feature points are hidden at some sampling time. In this case, the compliance matrix \mathbf{W}_{ef} and the vector $(\delta_{e,k} - \delta_e^r)$ should be recomputed by removing the corresponding rows.

Remark 3. It is noted that the constraints in QP formulation (10) can be deleted to compute both the external force and control input. This strategy can reduce the delay generated by the measure of feature points or control inputs. However, we needs more feature points to make sure that the optimization has unique solution. In addition, if the FEM is not accurate, the increased number of optimization variables could also reduce the accuracy of force computation. With this in mind, we employ the QP formulation with the constraints.

V. ESTIMATION OF EXTERNAL FORCES LOCATION

In this section, we first prove a property: the locations of external forces can be estimated just based on the displacement of feature points. Then we propose a simple strategy to estimate the locations.

In order to choose the number of feature points for the location estimation of external forces, the following theorem is needed.

Theorem 4. *Given the number of possible locations n_p , the locations of external forces can be estimated if the minimal number of feature points is n_p .*

Proof: All the nodes (the number is n_{all}) of the FEM are numbered so that we can get the locations if we get the corresponding indexes of actuated nodes. S_p is the number set of possible locations. We assume that the actual external forces λ_1 are applied on the node set S_1 ($S_1 \subseteq S_p$) and the displacement of defined effectors is $\Delta\delta_{1a}$. Then λ_1 can be written in the argued form λ_{1a} with n_{all} elements and the zero elements means that there is no force applied on the corresponding nodes. From (8), we have

$$\Delta\delta_{1a} = \mathbf{W}_{efa} \Delta\lambda_{1a} \quad (14)$$

where \mathbf{W}_{efa} is the compliance matrix between the defined effectors and the argued force vector.

If there is other group of external forces (the argued vector λ_{2a}) applied on the node set S_2 ($S_1 \cup S_2 \subseteq S_p$) to generate the same displacement of defined effectors $\Delta\delta_{1a}$, (14) can be written as

$$\Delta\delta_{1a} = \mathbf{W}_{efa} \Delta\lambda_{2a} \quad (15)$$

Combining both (14) and (15), we have

$$\mathbf{W}_{efa} (\Delta\lambda_{1a} - \Delta\lambda_{2a}) = 0 \quad (16)$$

We assume that the number of possible locations is n_p . Considering that $S_1 \cup S_2 \subseteq S_p$, the largest number of non-zero elements in vector $(\Delta\lambda_{1a} - \Delta\lambda_{2a})$ is n_p . If (16) has unique zero solution, i.e. $\Delta\lambda_{1a} = \Delta\lambda_{2a}$, the location and intensity of external forces can be estimated. We assume that the number of elements in $\Delta\delta_{1a}$ is n_e which is equal to that of the rows in \mathbf{W}_{efa} . If $n_e \geq n_p$, the unique solution of (16) is $\Delta\lambda_{1a} = \Delta\lambda_{2a}$.

Therefore, with unknown external forces in terms of locations and intensity, they can be uniquely determined if the number of defined effectors n_e is not less than that of the candidate external forces n_p . ■

Based on this theorem, we show our strategy into two steps to estimate the locations.

At the first step, the possible locations of external forces (the number is n_p) are predefined. Based on the Theorem 4, we define more effectors (the number is n_e and $n_e \geq n_p$) to realize the estimation.

Then, the intensity of all candidate external forces is computed at the same time using the optimization (10). The non-zero elements in $\Delta\lambda_{I,k}$ correspond to the indexes of external forces.

The feasibility is explained as following:

Considering all the possible locations, the solution $\Delta\lambda_{I,k}$ computed by (10) is unique to make the objective function (9) reach zero, i.e. $\mathbf{W}_{eI} \Delta\lambda_{I,k} = (\delta_{e,k} - \delta_{e,k}^r)$.

Using the manipulation in the upper proof, the actual force vector $\Delta\lambda_{f,k}$ can be argued to $\Delta\lambda_{I,k}^a$ where the non-zero elements correspond to elements in $\Delta\lambda_{f,k}$. $\Delta\lambda_{I,k}^a$ can also make sure that $\mathbf{W}_{eI} \Delta\lambda_{I,k}^a = (\delta_{e,k} - \delta_{e,k}^r)$. Considering the feature of unique solution, we have $\Delta\lambda_{I,k} = \Delta\lambda_{I,k}^a$ so that the non-zero elements in $\Delta\lambda_{I,k}$ correspond to the location of external forces.

Remark 5. For real applications, the FEM is not precise so that the zero element in the idea case maybe become non-zero. However, if the error of FEM is limited in a small bound, their absolute values are still much smaller than other elements. The absolute vector of all external forces are obtained by accumulating the computation result of each step. By setting a threshold for the elements in the vector, the indexes of the elements which absolute values are larger than the threshold are the correct locations.

VI. VALIDATION BY SIMULATION

A first verification of our algorithm has been performed using a simulation of the robot. It allows to verify the precision we can expect from our algorithm by setting aside, temporarily, the problems of camera sensors inaccuracies. Therefore, in the framework of implementation, two FEMs are needed: one for direct simulation and the other for inverse simulation. Using this setup, we can easily define the external forces, test the sensitivity to the parameters of the soft robot model, and benchmark the algorithm for different control inputs.

Two types of structures are tested in this section: a soft sheet and a parallel soft robot. For the soft sheet, there is no control input. However, we consider the control inputs for the soft parallel robot (for each actuator, the force is defined as

2*N*). In the validation, we also consider the case when the inverse model is not accurate. The setup of both cases consists of a direct FEM and an inverse FEM (see Fig.2). For external force computation, the locations are given and the intensities are computed based on the inverse FEM. Several candidate locations and forces should be defined for location estimation. Using the strategy proposed in Section V, the correct locations can be determined. The Young's Modules for both direct FEMs of soft sheet and soft robot are defined as 300 kPa. The same external forces $(0,0,0.4)N$ and $(0,0,-0.4)N$ are applied on both systems. The elements number and sampling frequency are respectively 4923 and 16Hz for soft sheet, and 4147 and 20Hz for soft robot.

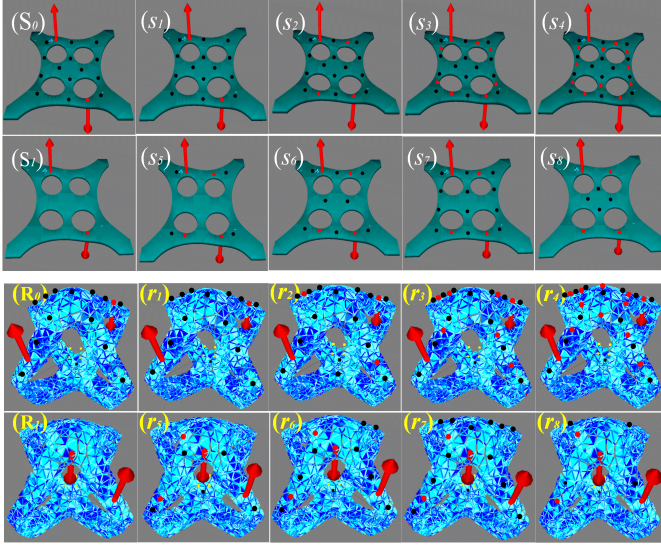


Fig. 2. Validation setup of soft sheet (S) and soft parallel robot (R). The red arrows stand for the applied forces and the computed forces respectively for direct FEMs and inverse FEMs. The black points and the red ones show the distribution of markers and candidate locations respectively. S_0, S_1 and R_0, R_1 are the direct FEMs for soft sheet and soft robot respectively. $s_1 \sim s_8$ and $r_1 \sim r_8$ are inverse FEMs and the differences between them are the distribution of markers and candidate locations. The same distribution of markers is defined for the groups $S_0, s_1 \sim s_4$ and $R_0, r_1 \sim r_4$. For the groups $S_1, s_5 \sim s_8$ and $R_1, r_5 \sim r_8$, the same distribution of candidate locations are employed to test the algorithm.

A. Evaluation index

In order to quantify the performance of the proposed strategies, we define two evaluation indexes I_e and I_w . I_e is used to evaluate the computation error and I_w is employed to evaluate the distribution of markers and candidate locations.

Due to the model error, the error of computed forces can not be zero. Based on the computation error, the index I_e is defined as $I_e = \|\mathbf{f}^c - \mathbf{f}^a\|$ where \mathbf{f}^c and \mathbf{f}^a are the vector of all computed and applied force respectively.² Given the same applied forces, a larger I_e means a larger estimation error.

If the deformation of structures is small, the compliance matrix \mathbf{W}_{ef} is close to a constant matrix so that the kinematic

²In the computation, each external force consists of three components, i.e. $\mathbf{f}_i = (f_x, f_y, f_z)^T$ defined in the same coordinate system. The symbol $\|\cdot\|$ means the norm of \cdot . In this paper, we compute the 1-norm for all vectors and matrices.

equation can be simplified as $\Delta\delta_e = \mathbf{W}_{ef}\Delta\lambda_f$. Then we have $\|\Delta\lambda_f\| \leq \|\mathbf{W}_{ef}^+\| \|\Delta\delta_e\|$. For the matrix $\mathbf{W}_{ef} \in \mathbb{C}^{m \times n}$ ($m \geq n$) of full rank, the evaluation index I_w is defined as $I_w = \|\mathbf{W}_{ef}^+\|$. If I_w is small, \mathbf{W}_{ef} is well-conditioned; if I_w is large, \mathbf{W}_{ef} is ill-conditioned. A smaller I_w corresponds to a better distribution of markers and candidate locations. Given the displacement error of markers, I_w can be employed to estimated the maximum error of computed forces.

B. Validation Results based on Accurate Inverse Model

In this section, eight cases (two groups) are tested for each system to evaluate our algorithm. As shown in Table. I, the evaluation index I_e is limited into a very small range so that the computed forces can track the applied forces in high accuracy. I_w is computed based on the initial compliance matrix for each case. For different distribution of markers and locations, I_w is very different.

TABLE I
EVALUATION INDEXES I_e AND I_w FOR BOTH VALIDATIONS. THE SHORTHAND NOTATIONS ARE S(SOFT SHEET), R(SOFT ROBOT), DM(DIRECT MODEL), AND IM(INVERSE MODEL). THE UNITES OF I_e AND I_w ARE mN AND mm/N , RESPECTIVELY.

S	DM	S_0				S_1			
	IM	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8
	I_e	0.3	1.1	4.6	33.0	2.8	1.3	1.4	130.3
	I_w	0.5	1.4	3.3	20.4	1.3	1.6	1.5	1991
R	DM	R_0				R_1			
	IM	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
	I_e	3.2	3.2	8.7	60.2	9.0	25.2	8.8	79.9
	I_w	0.2	0.8	8.1	168.9	28.5	12.1	4.9	52

C. Error Analysis

To evaluate the algorithm with different model error, we employ the setup shown in Fig. 2. The errors are generated by setting different parameters or position vectors for inverse FEM. We consider the imprecise Young's Modulus and the imprecise markers position as two kinds of errors for both systems. Besides, the error of actuator input for soft robot is also considered. For both systems, three imprecise data of Young's Modulus are tested: 280 kPa, 260 kPa and 240 kPa. Three errors in the position of the marker points are also used: errors of 0.05, 0.1 and 0.15mm along the z axis. Three input errors of each actuator 0.1*N*, 0.15*N*, and 0.2*N* are tested for the soft robot.

TABLE II
EVALUATION INDEX I_e (ERROR OF COMPUTED FORCE IN mN) FOR SOFT SHEET. TWO CASES ARE CONSIDERED: A (IMPRECISE YOUNG'S MODULUS) AND B (IMPRECISE MARKERS POSITION).

direct FEM		S_0				S_1		
inverse FEM		s_2	s_3	s_4	s_5	s_6	s_7	
A	280	109.8	197.9	278.3	137.7	118.4	114.0	
	260	216.3	395.4	551.1	273.5	234.7	224.9	
	240	319.4	590.2	807.9	404.4	347.8	332.4	
B	0.05	132.6	145.4	307.7	57.7	131.2	131.1	
	0.10	265.1	290.1	612.0	113.4	262.5	262.2	
	0.15	397.5	434.8	918.2	169.1	393.9	393.2	

TABLE III
EVALUATION INDEX I_e (ERROR OF COMPUTED FORCE IN mN) FOR SOFT ROBOT. THREE CASES ARE CONSIDERED: IMPRECISE YOUNG'S MODULUS (A), IMPRECISE MARKERS POSITION (B) AND IMPRECISE ACTUATOR INPUT (C).

direct FEM		\mathbf{R}_0			\mathbf{R}_1		
inverse FEM		r_2	r_3	r_4	r_5	r_6	r_7
A	280	62.1	337.7	592.1	118.0	239.8	100.5
	260	125.2	704.1	1230.0	221.1	475.8	202.9
	240	187.3	1094.3	1951.2	320.7	795.2	315.4
B	0.05	47.7	300.2	950.6	125.9	349.1	158.3
	0.10	94.6	601.0	2088.0	245.7	712.0	314.5
	0.15	142.1	900.5	3379.2	368.8	1081.1	474.0
C	0.10	23.8	452.9	663.6	196.7	330.2	74.8
	0.15	34.5	656.9	938.6	298.2	448.8	104.6
	0.20	46.1	877.1	1267.0	396.2	613.6	138.1

For location estimation, several candidate locations are pre-defined for both soft sheet and soft parallel robot. The forces on all candidate locations are computed at the same time by solving the inverse FEM problem. The goal of introducing these errors is to evaluate the sensitivity of our approach. The performances of our strategy are shown in Table II and III. If the model error is small, the predefined forces applied on the wrong locations are computed around zero. Therefore, only the locations where the predefined forces are computed to be a larger value are considered as the locations of external forces. If the inverse kinematic model is not accurate, the forces on the wrong locations becomes larger so that the location estimation and the force computation become to be imprecise.

The 1-norm error bound of computed forces can be estimated by I_w . This can be validated by the case B in Tab. II and III. In addition, I_w can be employed to choose a better distribution of markers. We can validate it from case B using direct FEM \mathbf{S}_0 and \mathbf{R}_0 .

Based on the comparison of data in Tab. II and III, the displacement vector of markers plays a more important role for our strategy. The error generated by imprecise Young's Modulus and actuator forces can be reduced by calibration before the experiment.

VII. EXPERIMENTS IMPLEMENTATION

For the implementation of our strategies, the 3D positions of several feature points should be obtained at each sampling time. We proposed an allocation algorithm based on FEM to get the positions in the first subsection. Then the experimental setup is introduced in the second subsection. In the last two subsections, we show the performances of external force sensing algorithms.

A. Feature Point Allocation

In this paper, the strategy of external force sensing is implemented using the position of feature points. At each sampling time, the feature points on the real robot should be allocated according to their corresponding ones on the FEM. Usually, this task can be realized by image identification [23] or multi-object tracking algorithm [24].

We propose a robust strategy for the allocation of feature points even in the situation of feature points loss. During the process of external force sensing, the hidden feature points have to be detected. The idea is inspired by the registration strategies for soft object in [25] where deformable registration is realized using closest point correspondences and FEM.

We define a threshold displacement for the detection. For each effector on the FEM, its closest correspondence on the real robot can be found. If the displacement between them is larger than the threshold, the corresponding point on the real robot is hidden. Then the compliance matrix and the effector vector are regenerated for the external force sensing.

More feature points should be defined than needed to sense the external forces. This allow to continue the force sensing when some points are not available. Besides, we can employ the effectors which have larger displacements for the external force sensing to increase the accuracy (see the experiment results).

B. Experimental Setup

The experimental setup for the external force sensing using the soft sheet and the soft parallel robot is shown in Fig. 3. Three components are needed for the implementation of the experiments: a soft object with several predefined feature points, a real-time simulated model, and a position tracking system. Both the soft sheet and the soft parallel robot are made of silicone and the Young's modulus are 290 kPa and 220kPa respectively. Four corners of the soft sheet is fixed and the soft parallel robot is actuated by cables (In the experiment, two cable actuators are considered). Several marker points are fixed on the surface of the soft objects and detected by the commercial optical tracking system (*OptiTrack* by NatrualPoint company) with sampling frequency 100Hz and a precision of 0.1mm. According to the position of marker points, the position of the effectors on the FEM are defined. The applied forces on the real soft objects are generated by rigid weights hung on the actuated points.

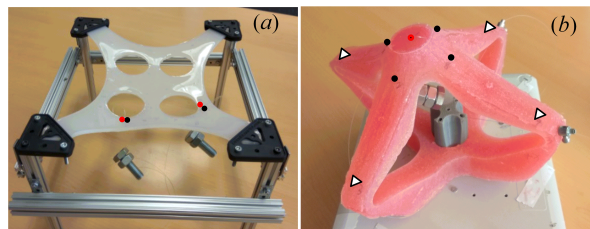


Fig. 3. Experimental setup of soft sheet (a) and soft parallel robot (b). The black and the red points stand for the position of marker points and location of external forces respectively. Two group of markers are defined for soft robot. The triangle signs represent the group of marker points which are close to the location of external force. Several weights are employed to generate applied external forces.

C. Experiment Results of External Force Computation

The algorithm can find both locations and magnitudes of external forces at the same time. However, in order to increase the accuracy of magnitudes computation, a better

distribution of feature points should be employed according to the estimated locations. The distribution of external forces and marker points is shown in Fig. 3. In this subsection, we show the performances of external force computation for soft sheet and soft parallel robot respectively.

TABLE IV
RESULTS OF TWO FORCES COMPUTATION FOR SOFT SHEET. THE UNITE OF THE FORCES IS mN .

applied forces	F_1	559	559	559	459	459	359
	F_2	559	459	359	459	359	359
computed forces	F_1^c	577	572	570	470	471	375
	F_2^c	532	443	344	440	354	356
average error		0.04	0.03	0.03	0.03	0.02	0.03

1) *force computation of soft sheet*: The performances of external force computation is shown in Table IV. Two external forces are applied on the soft sheet with known locations and directions. We employ three weights (i.e. $559mN$, $459mN$, and $359mN$) to generate different intensities of external forces. Six groups of applied forces are tested and the average error for each test is computed by $\frac{1}{2} \sum_{i=1}^2 |F_i^c - F_i| / F_i$. The accuracy of our method to compute external forces can reach about 97%.

TABLE V
RESULTS OF ONE FORCE COMPUTATION FOR SOFT PARALLEL ROBOT. THE UNITE OF THE FORCES IS mN . THE UNITE OF DISPLACEMENT IS mm .

cases	fixed cable displacement				fixed applied force			
	applied forces				cable displacement			
	859	659	459	259	6	8	10	12
C_1	882	670	463	260	661	670	643	642
error	0.03	0.02	0.01	0	0	0.02	0.02	0.03
C_2	829	621	421	230	627	621	618	613
error	0.03	0.06	0.08	0.11	0.05	0.06	0.06	0.07

2) *force computation of soft parallel robot*: As shown in Fig. 3, we divide the markers into two groups (four markers in each group). The external force is applied on the tip of the robot. In the experiments, we test two cases (using different group of markers) for external force computation. In Table V, the computed forces C_1 and C_2 use the first (close to the external force location) and the second (far from the external force location) group of marker points, respectively. Considering the influence of cable displacement, we test our algorithm with fixed cable displacement ($8mm$ for each cable) and with fixed applied force ($659mN$). The accuracy of computed forces C_1 and C_2 can reach to be 98% and 93%, respectively.

D. Experiment Results of Location Estimation

The screenshot of the location estimation is shown in Fig. 4. Before the experiments, candidate locations of external forces should be predefined. In our experiments, 14 markers and 12 candidate location are defined for soft sheet. For the soft robot, we employ 9 markers and 8 candidate locations. After deleting the smaller element in the computed external force vector, the location and the intensity of external forces are estimated and shown by red arrows in Fig. 4. Due to the modeling error of FEM, the number of candidate locations

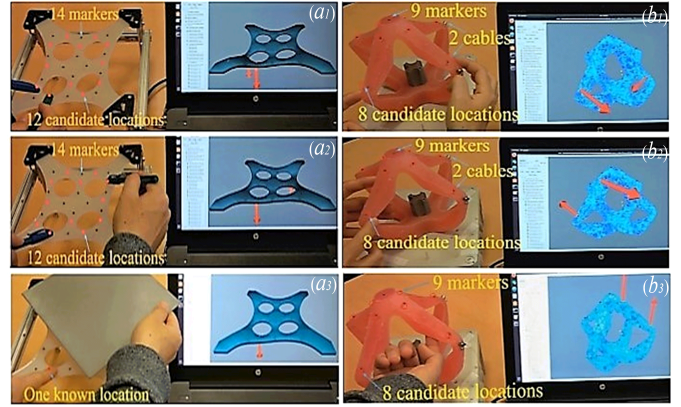


Fig. 4. Location estimation of external force for soft sheet (left one: a_1, a_2 estimation of forces; a_3 force computation when some markers are hidden) and soft parallel robot (right one: b_1, b_2 estimation of forces; b_3 rough estimation of location).

should not be very large and the candidate forces should not generate slight displacement of markers. Using the algorithm proposed in this paper, the locations of external forces can be obtained even if the number of actual external forces is unknown. However we need to have a set of candidate locations and a sufficient number of feature points.

We also test the performance of external force computation when some markers are not available. If some markers are not available, the feature point allocation algorithm will detect them and generates a new displacement vector and compliance matrix. Several candidate locations should be predefined to employ our algorithm. The method works if the external forces are applied on the predefined locations. However, for real applications, the forces may be applied everywhere which deteriorates the performances of the algorithm. We test actual cases where the external forces are not applied on the candidate locations precisely. As shown in Fig. 4, if the real locations are close to some candidate ones, our algorithm will find these candidate locations as the approximation of the real ones. However, if the real locations are not very close to candidate ones, several candidate locations around the real ones will be founded so that we can get a rough estimation using the geometrical center of founded candidate locations.

VIII. DISCUSSION

The results show that the algorithm proposed in this paper can be used not only to compute external forces but also to estimate their locations on soft object and/or soft robots. Thanks to the broad applicability of FEM strategy, the algorithm is general and can be employed for complex structure, 3D workspace, force sensing on whole soft body, and multi-force actuation. The algorithm can also be employed for external force sensing when the robot is actuated. However, some important limitations need to be emphasized.

First, the performance of our algorithm highly depends on the FEM. The performances deteriorate if the material parameters or the position vector are not accurate. The size and the complexity of the mesh plays an important role:

while we would need a very fine mesh for obtaining the convergence of the FEM method, we are constrained by the computation time if we want a real-time measurement. Due to the computation time, our method is probably not a good solution for embedded systems. In future work, we plan to investigate the use of model order reduction techniques to have a better trade-off between precision and computation performances.

Second, the computer vision system could also play an important role. Indeed, the force sensing is based on the observation of the global deformation of the soft object. This observation is done using a set of feature points. The placement of these feature points should be carefully distributed over the surface and a sufficient number of feature points should be used to capture multiple forces, or multiple locations. Moreover, the external forces applied on the soft robot can not be too small, otherwise no deformation could be observed.

Third, the model employed in our paper is quasi-static and the assumption is large displacements but small strain. The method has not been tested with large stress using a non-linear material model. The method can not capture dynamic/visco-elastic behaviors. In addition, our model accounts for the non-linear relationship between the displacements of the nodes and the strain and our approach is limited to the cases where the stress on the material remains small.

IX. CONCLUSION

Despite some identified limitations, this paper proposes a generic method to measure external forces applied on soft body and soft robots, based on computer vision and FEM. One of the current limitation is the distribution of the feature points that is manually performed. In the future work, other algorithms of computer vision could be tested to increase the number of feature points so that we can increase the number of locations of external forces and maybe the overall precision of the algorithm. Another improvement could be to use more precise FEM. One direction is to use more complete constitutive laws, like hyper-elastic models to test our algorithm on situations with more strain and an other direction would be the use of elements with more nodes (like cubic and quadratic tetrahedrons) We believe this algorithm could be useful in applications where it is complex to place force sensors on the robot, like for medical applications. Besides, the hybrid position and force control of soft robots will also be explored in our further work.

REFERENCES

- [1] N. Lu and D.-H. Kim, "Flexible and stretchable electronics paving the way for soft robotics," *Soft Robotics*, vol. 1, no. 1, pp. 53–62, 2014.
- [2] R. Xu, A. Yurkewich, and R. V. Patel, "Curvature, torsion, and force sensing in continuum robots using helically wrapped FBG sensors," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1052–1059, 2016.
- [3] H. Zhao, K. O'Brien, S. Li, and R. F. Shepherd, "Optoelectronically innervated soft prosthetic hand via stretchable optical waveguides," *Science Robotics*, vol. 1, no. 1, p. eaai7529, 2016.
- [4] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

- [5] D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact models for soft robotic manipulators," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 773–780, 2008.
- [6] T. M. Caldwell, D. Coleman, and N. Correll, "Optimal parameter identification for discrete mechanical systems with application to flexible object manipulation," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pp. 898–905, IEEE, 2014.
- [7] P. Polygerinos, Z. Wang, J. T. Overvelde, K. C. Galloway, R. J. Wood, K. Bertoldi, and C. J. Walsh, "Modeling of soft fiber-reinforced bending actuators," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 778–789, 2015.
- [8] E. Coevoet, T. Morales-Bieze, F. Largilliere, Z. Zhang, M. Thieffry, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, and J. Dequidt, "Software toolkit for modeling, simulation, and control of soft robots," *Advanced Robotics*, vol. 31, no. 22, pp. 1208–1224, 2017.
- [9] A. Colome, D. Pardo, G. Alenya, and C. Torras, "External force estimation during compliant robot manipulation," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 3535–3540, IEEE, 2013.
- [10] A. Wahrburg, S. Zeiss, B. Matthias, and H. Ding, "Contact force estimation for robotic assembly using motor torques," in *Automation Science and Engineering (CASE), 2014 IEEE International Conference on*, pp. 1252–1257, IEEE, 2014.
- [11] K. Xu and N. Simaan, "An investigation of the intrinsic force sensing capabilities of continuum robots," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 576–587, 2008.
- [12] D. C. Rucker and R. J. Webster, "Deflection-based force sensing for continuum robots: A probabilistic approach," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 3764–3769, IEEE, 2011.
- [13] M. Khoshnam, I. Khalaji, and R. V. Patel, "A robotics-assisted catheter manipulation system for cardiac ablation with real-time force estimation," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 3202–3207, IEEE, 2015.
- [14] S. Hasanzadeh and F. Janabi-Sharifi, "Model-Based Force Estimation for Intracardiac Catheters," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 1, pp. 154–162, 2016.
- [15] M. Kouh Soltani, S. Khanmohammadi, and F. Ghalichi, "A Three-Dimensional Shape-Based Force and Stiffness-Sensing Platform for Tendon-Driven Catheters," *Sensors*, vol. 16, no. 7: 990, 2016.
- [16] H. Yuan, P. W. Y. Chiu, and Z. Li, "Shape-Reconstruction-Based Force Sensing Method for Continuum Surgical Robots With Large Deformation," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 1972–1979, 2017.
- [17] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, vol. 35, no. 7, pp. 840–869, 2016.
- [18] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler, "Design and control of concentric-tube robots," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 209–225, 2010.
- [19] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 3982–3987, IEEE, 2013.
- [20] Z. Zhang, J. Dequidt, A. Kruszewski, F. Largilliere, and C. Duriez, "Kinematic modeling and observer based control of soft robot using real-time Finite Element Method," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pp. 5509–5514, IEEE, 2016.
- [21] Z. Zhang, T. Bieze, J. Dequidt, A. Kruszewski, and C. Duriez, "Visual servoing control of soft robots based on finite element model," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pp. 2895–2901, IEEE, 2017.
- [22] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
- [23] J. Wang and Y. Yagi, "Integrating color and shape-texture features for adaptive real-time object tracking," *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 235–240, 2008.
- [24] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1, pp. 212–219, IEEE, 2005.
- [25] A. Petit, V. Lippiello, G. A. Fontanelli, and B. Siciliano, "Tracking elastic deformable objects with an RGB-D sensor for a pizza chef robot," *Robotics and Autonomous Systems*, vol. 88, pp. 187–201, 2017.