Robotics and Autonomous Systems 194 (2025) 105134

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Multi-scale elastic contour mapping for deformable object shape control

Ignacio Cuiral-Zueco *, Gonzalo López-Nicolás

Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, C/ María de Luna n° 1, Zaragoza, 50018, Spain

ARTICLE INFO

Keywords:
Shape control
Deformable object manipulation
Multi-robot manipulation
Elastic mapping
Shape servoing

ABSTRACT

Shape control involves manipulating a deformable object towards a desired shape. In this paper we present a multi-scale elastic contour mapping method for shape control of texture-less deformable objects. Our method considers similar geometric features between the current and the target shape by means of multi-scale Laplacian descriptors. We compute elastic maps and thus define a contour point error that considers the stretching and compressing processes involved in shape control tasks. We validate the applicability of our novel mapping method in real shape control frameworks by means of a proposed shape control law. We perform simulations and experiments on different objects and materials with two robotic arms to validate our method.

1. Introduction

Visual-based object manipulation encompasses a variety of areas such as the manipulation of deformable objects. Within this area, shape control is a major challenge for a number of reasons, amongst them: there is a wide variety of deformable objects that can be classified according to different characteristics, such as shape, material or texture [1]. Another issue concerns the fact that, in a realistic scenario with a limited number of actuators, a deformable object is an underactuated system, as it holds an infinite number of degrees of freedom. Defining the shape control problem to be solved is considerably challenging, given the large range of diverse tasks that can be performed on deformable objects. This variety of different objects and tasks suggests the need to formalise the problem, something that has been addressed in surveys such as [2–5].

A deformable object classification within the robot manipulation context is proposed in [2]. Using a combination of physical-based and shape-based criteria, objects are classified as cloth-like, linear, planar, or solid. The tasks that can be performed on each type of object are also categorised, including tying knots (for linear objects), folding (for cloth-like and planar objects), hanging, splitting, cutting, etc. Our proposed method focuses on the shape control problem, which is relevant to the large-strain object group, i.e. objects that present a low Young's modulus (according to [2]). Shape control methods differ depending on the object type. However, they can be generally grouped according to a number of characteristics such as the model used by the control strategy (e.g., physical models, interaction matrix estimation or learning-based), the considered type of deformation (isometric, strain deformation, etc.) or the geometric features that change during the deformation (small/local or large/global deformations).

1.1. Related work

Based on a discrete network of mass–spring-damper systems and using curve parametrisation, the method in [6] controls several manipulators on planar objects in order to achieve local deformations (in simulation) by means of a dynamic energy-based control law. Addressing the problem of manipulating deformable objects locally, the approach in [7] approximates the Jacobian of the deformable object while also considering the stretching limits of the object and gripper collision with obstacles. Using Shape-from-Template (SfT) with monocular perception, [8] performs real experiments involving large isometric deformations on planar objects.

The visual-feature based estimation of the Jacobian matrix in [9] allows to control multiple visual feedback points of compliant elastic objects. In [10] an adaptive deformation model is proposed and validated with experiments carried out with various materials such as meat, foam and plastic. In [11], truncated Fourier series represent the 2D object shape and the deformation parameters are estimated to approximate the deformation Jacobian matrix. The method is validated through experiments carried out with two grippers (one active and one passive). Zhu et al. estimate the interaction matrix by applying Principal Component Analysis (PCA) to the 2D contour points of the shape's image projection [12]. Simulations and experiments involving one gripper and interaction with a passive element of the scene validate the method. Making use of a Fourier series parametrisation, a dual-arm flexible cable manipulation method is presented and validated with experiments in [13].

In the approach presented in [14], image contour moments are used to define a sliding control strategy that allows to perform shape control

* Corresponding author.
E-mail address: ignaciocuiral@unizar.es (I. Cuiral-Zueco).

https://doi.org/10.1016/j.robot.2025.105134
Received 15 September 2023; Received in revised form 15 January 2025; Accepted 18 July 2025
Available online 6 August 2025

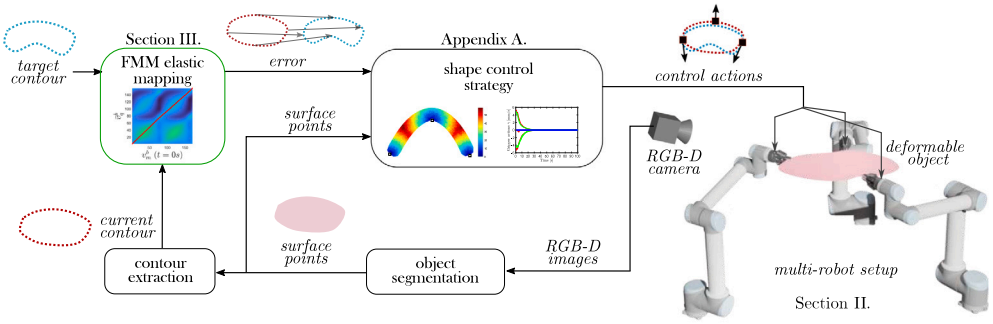
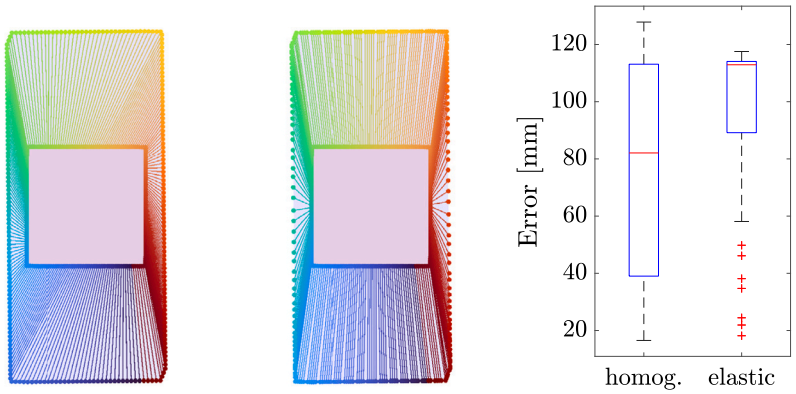


Fig. 1. Method's general overview. Current contour points are retrieved from the object's surface RGB-D information. Our proposed FMM Laplacian-based elastic mapping, which constitutes the main contribution of this paper, is performed between current and target contour points. The mapping allows to define the shape error for the control strategy, which generates actions for each of the robots' end effectors.



(a) Homogeneous. (b) Elastic. (c) Distance error.

Fig. 2. Illustrative example of a homogeneous map (a) and an elastic map obtained using our proposed multi-scale elastic mapping method (b). A square (current shape) is mapped to a rectangle (target shape). The lines between points represent mapped points (for clarity and ease of comparison, the lines have been coloured). In contrast to the homogeneous mapping, the elastic mapping favours the preservation of similar geometrical features (such as corners, in this example). The differences are also evident in the point-to-point distance distributions (c) each map generates. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

with objects that range from soft to rigid (articulated). Stability proof and experiments with a dual-arm robot setup validate their approach. The As-Rigid-As-Possible (ARAP [15]) geometric model allows the authors to compute a deformation Jacobian for shape servoing in [16]. They achieve experimental results with planar objects such as plastic mats, shoe soles and tire treads.

1.2. Proposed method overview

We propose an FMM (Fast Marching Method) Laplacian-based elastic contour mapping method to be used in robotic applications that involve the manipulation of deformable objects (e.g., see Fig. 1 where an overview of a deformable object shape control framework is illustrated). Our method tackles the definition of a holistic shape error that does not rely on visual texture and thus can be suitable for real applications and tasks such as food manipulation, where positioning fiducial markers is not a suitable option. The formulation of the problem is further elaborated in Section 2.

One of the major problems within shape control is to define a target (control reference) for the control system. The concept of *shape* is vague and lacks mathematical formalisation. This has led authors to develop different methods of measuring error in their shape control approaches. Some methods define the error through descriptors that partially encapsulate the geometry of the object, as it is the case of [10]. Other approaches compare shapes by using less compressed geometric information as it is the case of contour maps [12,17]. When a shape control reference is established through a contour map, the definition of the mapping process implicitly carries an estimation of how the object is expected to be deformed throughout the control process. This can be crucial in those cases where a map is not representative of realistic deformations: see Fig. 2, where the homogeneous map infers severe deformations such as the need to completely flatten the upper right corner of the square (for example), while the elastic one aims to stretch the object vertically, thus implying a more realistic deformation. Homogeneous maps between contours is a common approach in the literature [12,17]. However, homogeneous maps can only capture shape resemblance when no elastic deformation processes are involved in the shape control task. In this paper we develop on the computation of elastic maps between the current and the target shape contours that seeks shape resemblance on multi-scale level. The proposed elastic mapping method constitutes both a formalisation and an extension of our previous work in [18].

The novel contributions presented in this paper and main improvements with respect to [18] are:

- We improved the mapping method so that we no longer need to assume that the contour mapping is homogeneous in the locality of a contour point. This increased the method's performance when analysing features at larger scales resulting in a smoother multi-scale contour mapping.
- Our method now disregards the cost of non-injective maps. That is, we provide a more accurate estimation of mapping costs as we do not consider the cost of unfeasible maps in the locality of points.
- In the literature, since closed contours have no ends, methods such as [12] need to previously define the first matching points between

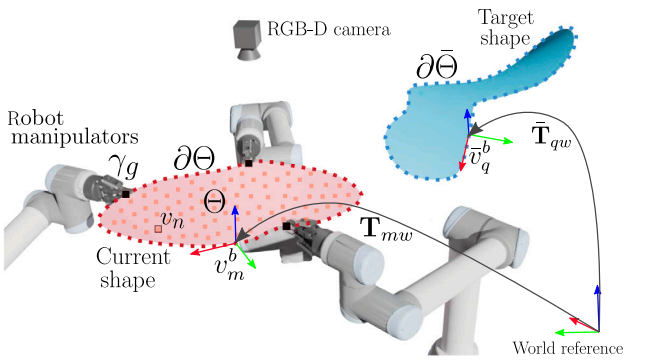


Fig. 3. Problem setup representation. Current object's visible surface Θ and its boundary $\partial\Theta$ are represented along with the target shape's boundary $\partial\bar{\Theta}$. Current v_m^b and target $\bar{v}_q^b$ contour points are extracted from both shapes. The object is grasped by G grippers at γ_g points. Rigid transforms $\mathbf{T}_{mw}$ and $\bar{\mathbf{T}}_{qw}$ define each of the contour points' v_m^b and $\bar{v}_q^b$ local references.

the current and the target contour. In this paper, we propose an improved closed contour mapping strategy that, within a single optimisation process, automatically defines proper first matching points as well as the rest of the elastic contour map between the contours.

- We validate the applicability of the novel elastic mapping method in real shape control setups whereas only simulations were presented in [18].

2. Problem formulation

Our proposed method focuses on elastic planar objects, which can be geometrically represented by surfaces with 2D intrinsic coordinates embedded in 3D Euclidean space (extrinsic coordinates). We define the object's shape as its visible 1D closed contour (Jordan curve) embedded in 3D Euclidean space. As we are perceiving the object with visual sensors, we assume the object's surface remains visible during the deformation process (something that can be achieved with optimal perception methods [19]). The number of grippers, placed along the object's contour, is defined by G $\geq$ 2 and we assume the grasping of the deformable object to be solved (something that can be achieved through methods like [20]). In this paper we do not tackle the automatic gripper positioning problem as we define gripper locations manually. Note that, in order to cause any deformation to a deformable object at steady-state, at least two grippers are required when inertia and gravity are negligible (this is generally assumed in the shape control literature, for instance [10,21]). In addition, the modification of the gripper-to-object contact points during the shape control process is not considered.

A representation of the problem setup is shown in Fig. 3. The planar object 2D surface Θ (embedded in 3D Euclidean space) can be segmented from the RGB-D sensor data as a set of 3D points $v_n \in V = \{v_n, n = 1, \dots, N\}$ with position vectors $\mathbf{v}_n \in \mathbb{R}^3$. The boundary of Θ, denoted by $\partial\Theta$, is retrieved from V by means of an α-shape [22] contour extraction that generates ordered contour points $v_m^b \in V^b = \{v_m^b, m = 1, \dots, M\} \ni V^b \subseteq V$. For now on, letter b (both as sub-index or super-index) denotes that an element belongs to a boundary. Contour points v_m^b are contour-wise ordered, meaning v_m^b's contour neighbours are v_{m-1}^b and v_{m+1}^b. As contour points are homogeneously sampled (uniform spacing between them) and contour-wise ordered, sub-index m acts as the discrete parameter of curve $\partial\Theta$. Contour points v_m^b have associated position vectors $\mathbf{v}_m^b \in \mathbb{R}^3$ stacked in matrix $\mathbb{V}^b \in \mathbb{R}^{M\times 3}$, $\mathbb{V}^b = [(\mathbf{v}_m^b)^\intercal]$.

Along contour $\partial\Theta$, grippers $\gamma_g \in \Gamma = \{\gamma_g, g = 1, \dots, \mathrm{G}\}$ are positioned. We approximate the object grasping by a single contact point per gripper. The grippers' position vectors $\boldsymbol{\gamma}_g \in \mathbb{R}^3$ allow us to define the single integrator dynamics as $\dot{\boldsymbol{\gamma}}_g = \mathbf{u}_g$, where $\mathbf{u}_g \in \mathbb{R}^3$ is the control action. We want to define control actions $\mathbf{u}_g$ so that the current contour $\partial\Theta$ acquires the target contour's shape $\partial\bar{\Theta}$. In this paper, elements of the target shape are denoted with a bar above. Therefore, we refer to the points belonging to the target contour $\partial\bar{\Theta}$ as $\bar{v}_q^b \in \bar{V}^b = \{\bar{v}_q^b, q = 1, \dots, Q\}$, with position vectors $\bar{\mathbf{v}}_q^b \in \mathbb{R}^3$ stacked in matrix $\bar{\mathbb{V}}^b \in \mathbb{R}^{Q\times 3}$, being $\bar{\mathbb{V}}^b = [(\bar{\mathbf{v}}_q^b)^\intercal]$. Note that M and Q do not need to be the same and M is not necessarily time constant as points are homogeneously sampled through iterations, e.g., if the object stretches M will increase. We define a local reference $\mathbf{T}_{mw}, \bar{\mathbf{T}}_{qw} \in \mathbb{R}^{4\times 4}$ with respect to the global reference frame for each of the current and target contour points (respectively). The local reference axes of each contour point v_m^b ($\bar{v}_q^b$ respectively) are defined as $(\mathbf{x}_m, \mathbf{y}_m, \mathbf{z}_m)$, with $\mathbf{x}_m$ the contour's tangent vector at point v_m^b; $\mathbf{y}_m$ the locally normal surface vector at v_m^b; and $\mathbf{z}_m$ orthonormal to $\mathbf{x}_m$ and $\mathbf{y}_m$. Note that these local frames of reference need to be coherent for a proper performance of the mapping method (2D surfaces present a surface normal on each side), we propose defining $\mathbf{z}_m$ towards the sensor.

3. Elastic contour mapping

In this section we begin with an overview of our multi-scale Laplacian descriptors, introduced in [18]. We then present the elastic mapping method with the assumption that contour parametrisation origins are favourable, i.e. that points v_1^b and $\bar{v}_1^b$ constitute a good match. Then, we generalise our method to tackle unfavourable parametrisation origin points.

3.1. Multi-scale Laplacian descriptors

These descriptors are computed using contour point coordinates $\mathbf{v}_m^b, \bar{\mathbf{v}}_q^b$ and their associated local references $\mathbf{T}_{mw}, \bar{\mathbf{T}}_{qw}$ (see Fig. 3). Two contour points are adjacent at scale λ when they lie within a contour's topological distance $s^\lambda \in S = \{s^\lambda, \lambda = 1, \dots, \Lambda\}$. Distances s^λ increase uniformly with λ being $s^\lambda = \lambda\, r_{voxel}$ and r_{voxel} the 3D sensor's resolution. The maximum topological distance between two contour points defines s^Λ. We use transforms $\mathbf{T}_{mw}$ to define local-coordinate multi-scale Laplacian surfaces $\Omega_x, \Omega_y, \Omega_z \in \mathbb{R}^{M\times\Lambda}$ (see Section 3. A in [18]). Analogously, using $\bar{\mathbf{T}}_{qw}$ we can compute the target Laplacian surfaces $\bar{\Omega}_x, \bar{\Omega}_y, \bar{\Omega}_z \in \mathbb{R}^{Q\times\Lambda}$. In Fig. 4, Laplacian surfaces $\Omega_x, \Omega_y, \Omega_z$ and $\bar{\Omega}_x, \bar{\Omega}_y, \bar{\Omega}_z$ for a current and a target shape can be visualised. These surfaces constitute what we refer to as Multi-scale Laplacian descriptors (see [18] for detailed explanation).

3.2. Laplacian descriptor based FMM contour mapping

The Fast Marching Method (FMM) [24] solves the Eikonal equation

$$\left|\nabla T(\theta,\bar{\theta})\right| F(\theta,\bar{\theta}) = 1, \quad \theta \in \partial\Theta,\ \bar{\theta} \in \partial\bar{\Theta}, \tag{1}$$

where $\theta, \bar{\theta} \in \mathbb{R}$ are the coordinates for the continuous parametrisation of contours $\partial\Theta, \partial\bar{\Theta}$. We define this parametrisation so that values of $\theta = m, \bar{\theta} = q$ correspond to the location of the uniformly sampled points $v_m^b, \bar{v}_q^b$ on contours $\partial\bar{\Theta}, \partial\Theta$ respectively (see Fig. 4). Eq. (1) is typically used to model the propagation of a surface front moving with a normal speed $F(\theta,\bar{\theta})$ and crossing a point $(\theta,\bar{\theta})$ with a time cost of $T(\theta,\bar{\theta})$. $T(\theta,\bar{\theta})$ can be seen as a cost function in which the cost of passing through point $(\theta,\bar{\theta})$ translates in more travel time for higher $T(\theta,\bar{\theta})$ values.

FMM has been used to perform curvature-based contour mapping between discrete curves with sub-resolution accuracy [25]. We propose an FMM contour mapping approach that considers curvature of 1D curves embedded in 3D and performs a multi-scale analysis by means of Laplacian surfaces Ω as our descriptors. We will first define a discrete contour similarity surface $\mathbf{F}(m,q) \in \mathbb{R}^{M\times Q}$, i.e. the discrete equivalent of $F(\theta,\bar{\theta})$ in (1), that, by means of the FMM, provides us with our discrete assignation cost surface $\mathbf{T}(m,q) \in \mathbb{R}^{M\times Q}$, i.e. discrete equivalent of $T(\theta,\bar{\theta})$. Using central differences, we then propose computing a continuous gradient descent path $\mathcal{P}(\theta)$ along surface $\mathbf{T}$ that allows us to generate an elastic map that favours multi-scale geometrical resemblance between the current and the target contour.

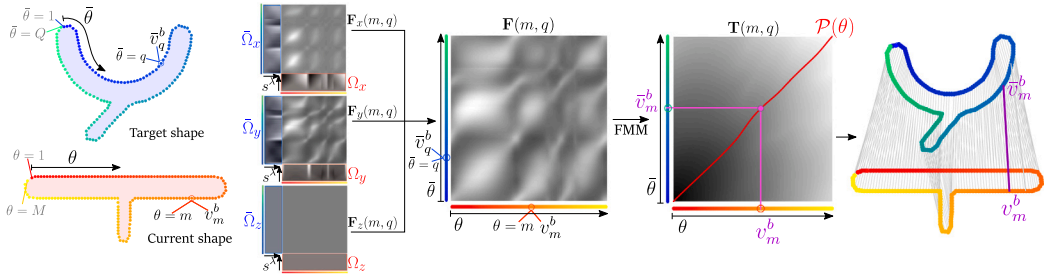


Fig. 4. Illustrative example of our FMM based elastic mapping process. Multi-scale Laplacian descriptors are computed for the current and target contour points $v_m^b, \bar{v}_q^b$ along scale values s^λ. The Laplacian descriptor' components $\Omega_x, \Omega_y, \Omega_z$ (current contour descriptors) and $\bar{\Omega}_x, \bar{\Omega}_y, \bar{\Omega}_z$ (target contour descriptors) allow us to compute similarity surfaces $\mathbf{F}_x(m,q), \mathbf{F}_y(m,q), \mathbf{F}_z(m,q)$. The similarity surface $\mathbf{F}(m,q)$ (obtained using $\mathbf{F}_x(m,q), \mathbf{F}_y(m,q), \mathbf{F}_z(m,q)$) provides the input for the MSFMM [23] which allows to compute the assignation-cost surface $\mathbf{T}(m,q)$. In this figure, regarding Laplacian descriptors, a lighter tone denotes larger positive values and darker tones lower negative values. As for the similarity surface $\mathbf{F}(m,q)$, lighter tones imply more similarity between contour points. The continuous gradient descent path $\mathcal{P}(\theta)$ on $\mathbf{T}(m,q)$ defines the elastic map.

3.3. Discrete speed function definition: the similarity surface

The discrete surface $\mathbf{F} \in \mathbb{R}^{M\times Q}$ (Fig. 4) represents the geometrical similarity between contour points v_m^b and $\bar{v}_q^b$ at a multi-scale level. We define each element $\mathbf{F}(m,q)$:

$$\mathbf{F}(m,q) = \left\| \left(\mathbf{F}_x(m,q), \mathbf{F}_y(m,q), \mathbf{F}_z(m,q)\right) \right\|_2 . \tag{2}$$

Surface $\mathbf{F}_x \in \mathbb{R}^{M\times Q}$ acts as similarity matrix for component x (in the local reference). Element (m,q) of $\mathbf{F}_x$ yields:

$$\mathbf{F}_x(m,q) = \left(\sum_{\lambda=1}^{\Lambda} \left[\mathbf{W}^\lambda * \mathbf{E}_x^\lambda(m,q) \right] + \beta \right)^{-1} , \tag{3}$$

where operator $*$ denotes the convolution product with wrap-around matrix edge handling (as current and target points constitute closed contours). We define parameter $\beta > 0$ to ensure $\mathbf{F}_x(m,q) > 0$. More insight on β will be provided in upcoming paragraphs. Discrete surface $\mathbf{E}_x^\lambda \in \mathbb{R}^{M\times Q}$ contains the Laplacian error of mapping point v_m^b to point $\bar{v}_q^b$ at scale λ:

$$\mathbf{E}_x^\lambda(m,q) = \left| \bar{\Omega}_x(q,\lambda) - \Omega_x(m,\lambda) \right| . \tag{4}$$

Matrix $\mathbf{W}^\lambda \in \mathbb{R}^{(2\lambda+1)\times(2\lambda+1)}$ in (3) is obtained as follows:

$$\mathbf{W}^\lambda = \mathbf{H}^\lambda \circ \mathbf{G}^\lambda, \tag{5}$$

where operator $\circ$ is the Hadamard product and matrix $\mathbf{H}^\lambda = \text{blkDiag}\left(\mathbf{1}_{\lambda\times\lambda}, 1, \mathbf{1}_{\lambda\times\lambda}\right)$, $\mathbf{H}^\lambda \in \mathbb{R}^{(2\lambda+1)\times(2\lambda+1)}$. We denote $\mathbf{1}_{\lambda\times\lambda}$ as a $\lambda\times\lambda$ all ones matrix. Matrix $\mathbf{G}^\lambda(\mathrm{u},\mathrm{v}) \in \mathbb{R}^{(2\lambda+1)\times(2\lambda+1)}$ represents a discrete Gaussian function centred at $(\lambda+1, \lambda+1)$ and it is defined as:

$$\mathbf{G}^\lambda = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(\mathrm{u}-\lambda-1)^2 + (\mathrm{v}-\lambda-1)^2}{2\sigma^2} \right), \tag{6}$$

where $\mathrm{u}, \mathrm{v} \in \mathbb{N}$ are the matrix indices and $\sigma = \lambda/\sqrt{2}$. Similarly to $\mathbf{E}_x^\lambda$, $\mathbf{E}_y^\lambda$ and $\mathbf{E}_z^\lambda$ are computed, i.e. using the remaining components (y and z) of the Laplacian Descriptors as in (4), and thus $\mathbf{F}_y(m,q), \mathbf{F}_z(m,q)$ can be obtained (see Fig. 4). Note that $\mathbf{W}^\lambda$ only varies with scale λ and not with spatial dimensions x, y, z. A representation of two matrices $\mathbf{E}_y^\lambda$ and $\mathbf{G}^\lambda$ is shown in Fig. 5.

In combination with the convolution product in (3), $\mathbf{W}^\lambda$ serves a two-fold purpose:

(1) It defines a discrete weight surface that applies a probability distribution on surface $\mathbf{E}_x^\lambda$ in the neighbourhood of coordinates m and q. The neighbourhood in $\mathbf{E}_x^\lambda$ is defined by indexes within index boundaries $(m \pm \lambda, q \pm \lambda)$ as to be coherent with the scale analysis. The distribution $\mathbf{G}^\lambda$ models the likelihood of how, given a match of points v_m^b and $\bar{v}_q^b$, the contour neighbours of v_m^b will be matched to those of $\bar{v}_q^b$. Note how matrix $\mathbf{H}^\lambda$ in (5) cancels both the second and the fourth quadrant of the Gaussian surface $\mathbf{G}^\lambda$ (see Fig. 5). This allows to disregard errors of non-injective matches in the neighbourhood of v_m^b and $\bar{v}_q^b$, i.e. we do not consider the cost of point matches that would surely imply mapping the same contour point twice.

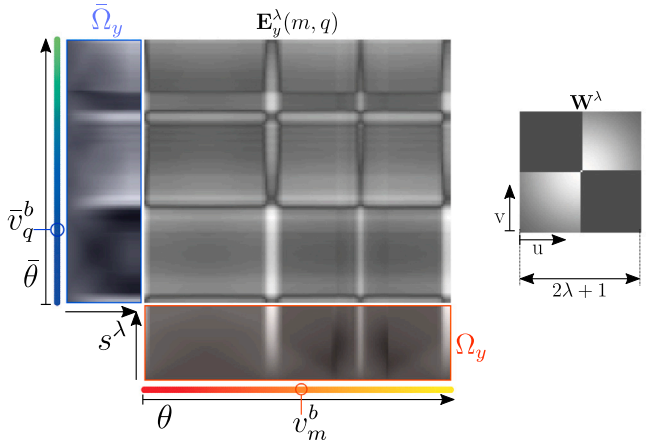


Fig. 5. Visualisation of the error surface $\mathbf{E}_y^\lambda(m,q) \geq 0$ and the matrix $\mathbf{W}^\lambda \geq 0$ for a given scale s^λ. The colour tone intensity of the Laplacian descriptors $\Omega_y, \bar{\Omega}_y$ represents larger (positive) values for lighter tones and lower (negative) values for darker tones. Note that surface $\mathbf{W}^\lambda(\mathrm{u},\mathrm{v}) = 0$ on the second and fourth quadrants (dark tone).

(2) Through the convolution product, $\mathbf{W}^\lambda$ integrates the Laplacian error of mapping point v_m^b to point $\bar{v}_q^b$ along with the probability-weighted Laplacian error of their neighbouring points' potential matches.

Note how the summation term within (3) is always equal or greater than zero. Therefore, $\beta > 0$ implies that the parenthesis term in (3) is always greater than zero, and thus $\mathbf{F}(m,q) > 0$. This is conceptually important as $\mathbf{F}(m,q)$ in (2) represents a speed function and $\mathbf{F}(m,q) \leq 0$ would imply an infinite or a negative time-cost for the front to propagate through point (m,q), both undesirable scenarios as that would mean that the wave front could stall or recede (i.e. it could generate non-injective matches).

We perform the Multi Stencil Fast Marching Method (MSFMM) [23] with $\mathbf{F}$ as our input speed surface and obtain the discrete cost function $\mathbf{T} \in \mathbb{R}^{M\times Q}$ (the MSFMM considers 8 neighbours per point in $\mathbf{F}$).

We use central differences to compute a continuous gradient descent path $\mathcal{P}(\theta)$ on surface $\mathbf{T}$ (from $\mathbf{T}(M,Q)$ to $\mathbf{T}(1,1)$). Path $\mathcal{P}(\theta)$ can be regarded as a continuous function $\mathcal{P} : \mathbb{R} \to \mathbb{R}^2$ with current contour parameter θ as input and matched target contour parameter $\bar{\theta}$ and mapping cost $T(\theta,\bar{\theta})$ as outputs. Each uniformly sampled current contour point v_m^b has an associated parameter value of $\theta = m$ and it is matched

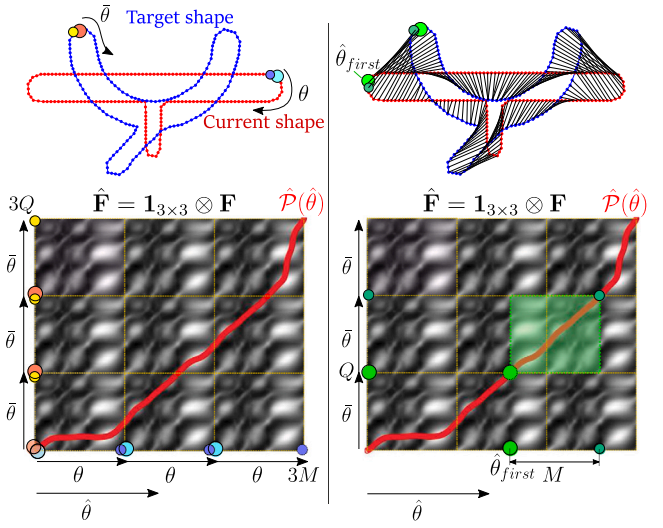


Fig. 6. On the left: elastic contour mapping process for inconveniently parametrised contours with first and last matching points ($\theta = 1$, $\bar{\theta} = 1$) and ($\theta = M$, $\bar{\theta} = Q$) (respectively). The first and last contour points are represented with larger circles and different colours. Large blue and orange circles for the first contour points $(1, 1)$ and small dark blue and yellow circles for the last contour points (M, Q). Below, the extended surface $\hat{\mathbf{F}}$ is represented along with a gradient descent path $\hat{\mathcal{P}}(\hat{\theta})$ (red path) obtained from the $\hat{\mathbf{T}}$ surface. On the top right, the result for the contour mapping is displayed with lines linking current and target contour points. The well conditioned first match is represented with a large green circle and the new last contour point is displayed as a smaller dark green circle. Below, the $\hat{\mathbf{F}}$ and the gradient descent is represented again along with the obtention of $\hat{\mathcal{P}}(\hat{\theta}_{first})$. The semi-transparent green patch contains $\mathcal{P}$, the well conditioned mapping path. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

to a target contour point with a parameter value $\bar{\theta}$ obtained from:

$$\mathcal{P}(\theta = m) = (\bar{\theta}, T(m, \bar{\theta})). \tag{7}$$

New discrete contour points $\bar{v}_m^b$ are retrieved from $\partial\bar{\Theta}$ according to the values of $\bar{\theta}$ in (7) and thus the elastic contour mapping is defined. For each sampled (perceived) current contour point v_m^b we obtained an elastically matched target contour point $\bar{v}_m^b$ (see Fig. 4).

3.4. Elastic contour mapping for closed contours

Performing a gradient descent from $\mathbf{T}(M, Q)$ to $\mathbf{T}(1, 1)$ is equivalent to assuming that point matches $(v_M^b, \bar{v}_Q^b)$ and $(v_1^b, \bar{v}_1^b)$ are convenient, which, as seen in the contour parametrisation shown in Fig. 6, is not necessarily the case. A first matching point estimation method for closed contours is suggested in [25]. They propose performing a gradient descent on an extended similarity matrix (different from the one presented here). They state that such path contains the optimal mapping but they do not develop on how to retrieve it. In this paper, we propose generating matrix $\hat{\mathbf{F}} = \mathbf{1}_{3\times3} \otimes \mathbf{F}$, $\hat{\mathbf{F}} \in \mathbb{R}^{3M\times3Q}$ (instead of $3M \times 2Q$, as it would be if we followed their approach in our problem) in order to facilitate the convergence of the gradient descent path $\hat{\mathcal{P}}(\hat{\theta})$ down $\hat{\mathbf{F}} \in \mathbb{R}^{3M\times3Q}$ in unfavourable cases (see Fig. 6). We use $\hat{\theta}$ to refer to the parametrisation of $\hat{\mathcal{P}}$ on extended matrices $\hat{\mathbf{F}}$. Furthermore, we propose retrieving the optimal mapping path $\mathcal{P}$ directly from $\hat{\mathcal{P}}(\hat{\theta})$. We can find $\hat{\theta}_{first}$ such that $\hat{\mathcal{P}}(\hat{\theta}_{first}) = (Q, T(\hat{\theta}_{first}, Q))$. We obtain $\mathcal{P}(\theta)$ as the segment of $\hat{\mathcal{P}}(\hat{\theta})$ defined by $\hat{\theta} \in [\hat{\theta}_{first}, \hat{\theta}_{first} + M]$. We also propose using $\hat{\theta}_{first}$ from one iteration to re-parametrise the contour points of the next iteration and thus enhance convergence to the optimal path through iterations.

Regarding parameter $\beta > 0$ in (3), β has an impact on how much the mapping cost $\mathbf{T}(m, q)$ increases when coordinates (m, q) lie far away from the values along the diagonal that goes from $(1, 1)$ to (M, Q). The larger β, the higher the cost of deviating from an homogeneous mapping path (i.e., with a very large β, a path $\mathcal{P}$ would run along the diagonal). However, our method needs to effectively deviate from the diagonal when contour parametrisation is not favourable. This requires our β to take low values ($\beta \approx 0.001$) as to allow $\hat{\mathcal{P}}$ to deviate from the diagonal (when necessary) and to converge to point matches that result in lower overall cost.

The elastic mapping method proposed here is performed at each iteration during the shape control process and constitutes a key element for the definition of a suitable shape control reference. Note that the proposed Laplacian-based descriptors are intrinsic and thus remove the dependence on the shape's embedding. This is something of great interest for shape analysis as, through the FMM, the proposed descriptors allow to define a contour mapping that is completely independent (decoupled) from other embedding-dependent metrics (such as pure global–local translation, rotation, or Procrustes distance).

4. Simulations and experiments

This section illustrates the applicability of the proposed Multi-scale elastic contour mapping in a shape control application. We conducted several simulations and experiments using our contour mapping method as input for the basic shape control strategy outlined in Appendix. In this framework, the discrete control law in (A.8) defines robot actions with magnitudes proportional to the shape error module so that they effectively prevent overshooting, ensuring safe convergence for a given gripper configuration (see Lemma A.1), as proven in Theorem A.8.

4.1. Simulation results

We have performed several simulations using the As Rigid As Possible (ARAP) model [15] to simulate the object's response to our control law (A.8), as it fits our purpose of manipulating large-strain objects. We present 4 simulations that constitute several scenarios for the application of the shape control strategy (find more simulations on the accompanying video). Figs. 7 and 8 contain two simulations each, with 6 column-wise graphics per simulation.

In the first two rows, the initial and final state of the object is shown with a red triangular mesh. In both cases the target contour is also shown in blue and the mapping between the current and the target contour is displayed with thin grey lines. Just below these two graphics the similarity surfaces $\mathbf{F}$ corresponding to the initial and final time instants are shown. In the similarity surface warmer colours represent higher similarity while colder colours represent lower similarity between contour points. The gradient descent path $\mathcal{P}$ is displayed in red. Note that the contours have been re-parametrised at every iteration (using the method in Section 3.4) so that path $\mathcal{P}$ and surface $\mathbf{F}$ are easier to interpret in the figures (i.e. $\mathcal{P}$ goes from the upper right corner to the lower left corner of $\mathbf{F}$). We refer to $\mathbf{F}$ after shifting the current contour points indexes (m) as $\mathbf{F}_{shifted}$. The first current and target contour points in the $\mathbf{F}_{shifted}$ surface approximately correspond to the leftmost points of the figures above. Plots on the fifth row display the contour point error module $\|\mathbf{E}\|$, as defined in (A.5), in [mm] (red colour) along with the relative stretching of the simulation mesh edges (blue colour). Both display the mean value (dashed line) and the standard deviation (shaded area). The remaining plot shows the action of the grippers $\dot{\gamma}_{g,x}, \dot{\gamma}_{g,y}, \dot{\gamma}_{g,z}$ (red, green and blue) in [mm/s]. All target shapes in the simulations were arbitrarily defined thus feasibility was not ensured, i.e. there was no certainty of achieving all-zero components on the error distribution.

In the first simulation in Fig. 7, a shape control task involving geometric features at large and intermediate scales is presented. The values along the diagonal of the $\mathbf{F}_{shifted}$ surface presents larger similarity values after the shape control task has taken place. In this simulation grippers are conveniently positioned and thus the error can be highly

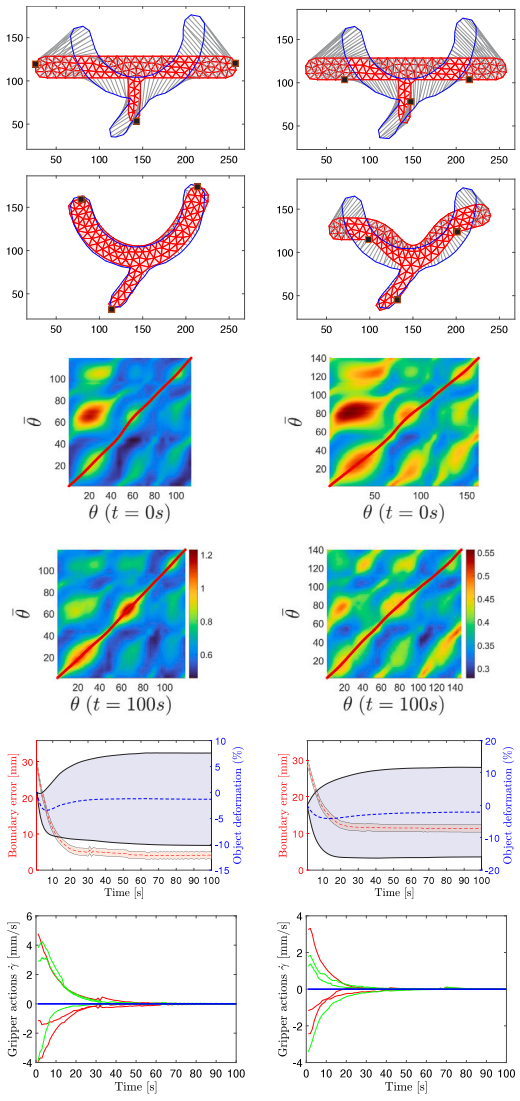


Fig. 7. Two simulations for the same shape control problem with different gripper configuration: a favourable one (left column) and an inconvenient one (right column). See Section 4.1 for a detailed explanation of the figure. The first two rows display the object's mesh (red), target contour (blue), and the mapping (grey). On the object's mesh, the 3 actuation points are represented with black squares. Below these, the similarity surfaces $\mathbf{F}_{shifted}$ are shown, where warmer colours indicate higher similarity, along with the gradient path $\mathcal{P}$ (red). The fifth row presents the contour error (mean and standard deviation in solid and shaded red) and mesh-edge stretching (mean and standard deviation in dashed and shaded blue). Finally, the last plot illustrates the gripper actions, with red, green, and blue lines representing the x, y, and z components, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

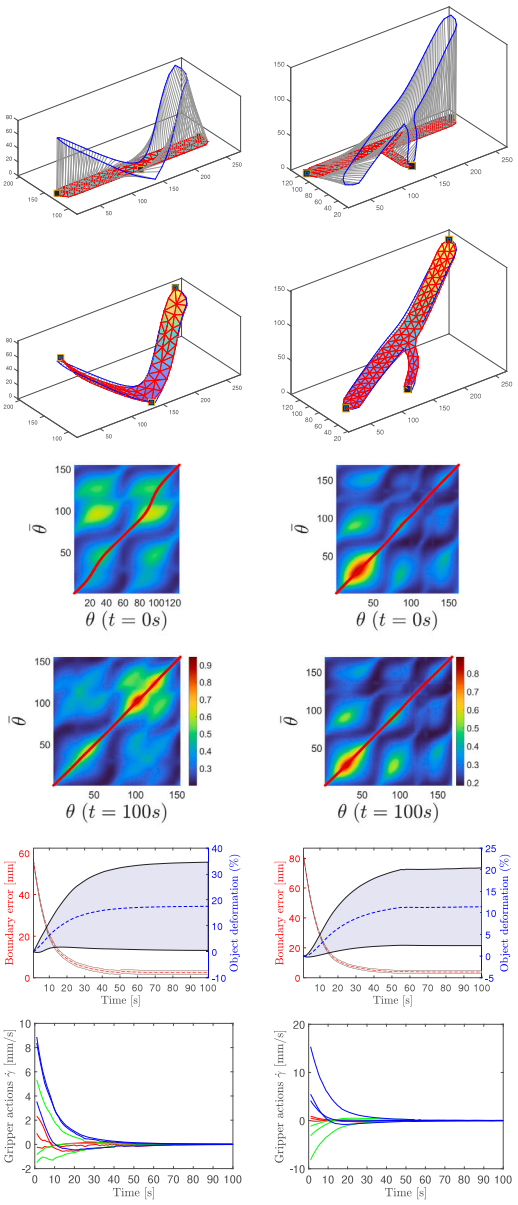


Fig. 8. Two simulations are presented, each addressing a 3D shape control problem involving three grippers. The first simulation (left) illustrates a 3D U-shaped bending process. The second simulation (right) focuses on manipulating an object with geometric features at varying scales, including two symmetric ends and a shorter, thinner appendix. For a detailed explanation of the elements depicted in the graphs and plots, refer to Section 4.1. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

reduced. To test the system in a less favourable scenario, the second column in Fig. 7 presents a less convenient gripper configuration. Although the error cannot be as highly reduced as in the first simulation, the control strategy manages to provide a suitable solution. The higher stress generated by the unfavourable gripper positioning is reflected in the object deformation plot, where the standard deviation of mesh strain almost doubles with respect to the one in the first column.

The two simulations in Fig. 8 demonstrate our mapping and shape control framework applied to 3D shapes using three grippers. The first simulation involves a U-shaped bar, while the second features a T-shaped object with geometric features at diverse scales. These *diverse scales* refer to features of varying relative sizes within the overall shape: a large main bar, a medium-sized appendix, and small-scale curvatures at the ends and junctions of the T-shape. The 3D multi-scale analysis of our method effectively analyses and considers this feature richness when computing the elastic map.

4.2. Experimental setup

The experimental setup (Fig. 9) consists of 5 elements: two robotic arms, the deformable object that is being controlled, an RGB-D camera and a spotlight. The robotic arms are the PhantomX model with 12 A Dynamixel servos that allow to perform position control. These robotic arms have 4 spatial degrees of freedom and one extra degree of freedom for opening or closing the grippers. The camera is the IntelRealsense D-435. The object segmentation has been performed on the RGB video

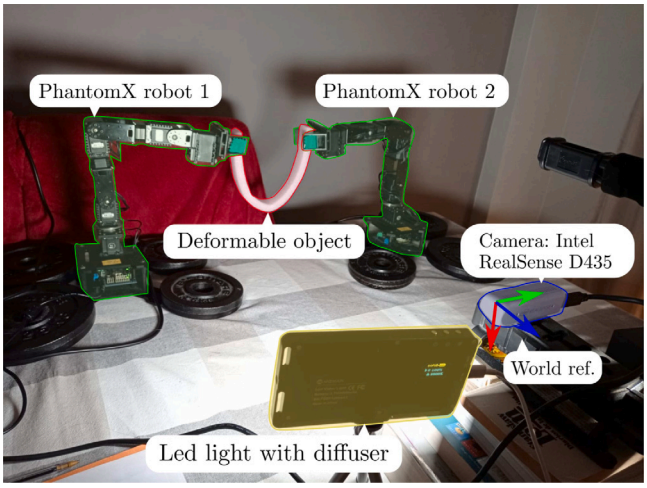


Fig. 9. Experimental setup. The elements that constitute the setup are two PhantomX robotic arms holding a deformable object, an Intel Realsense D435 RGB-D camera and a light with a diffuser. The world reference axes, positioned at the camera's optical centre, are displayed as well. The world reference Z axis is aligned with the camera's optical axis and points backwards with respect to the orientation of the camera. The X and Y axes are contained in the image plane (approximately parallel to the 2D object plane).

images by segmenting the object's colour in the CIELab space as it constitutes a perceptually uniform colour space. The object's depth with respect to the camera has been retrieved from the RGB-aligned depth map. The global reference axes are positioned at the optical centre of the RGB camera and oriented as shown in Fig. 9, where the RGB (red, green and blue) colours correspond to the X, Y, Z global axes. Our elastic mapping method, implemented in MATLAB with C++ (MEX), processes ~100 contour points across 15 scales in about 10 ms (experiments were conducted on an Intel(R) Core(TM) i7-8565U CPU with 1.99 GHz and 16 GB of RAM). This processing time provides adequate online capacity for other processes, allowing tasks such as segmentation, contour extraction, and control action computation to be performed at loop camera rates of 20–30 Hz.

Our control strategy defines actions in $\mathbb{R}^3$ and thus only position commands are transmitted to the grasped points. This means that the grasped points must be free to rotate in any direction around the gripper's contact point. The PhantomX grippers perform a full-contact grasp that does not provide the three rotational degrees of freedom. By traversing the deformable objects with rods and making the grippers grasp the rods perpendicularly (see Fig. 10), we were able to release the rotation of the object in one plane. Therefore, the setup meets the design requirements of our control system for those deformations that take place in the plane defined by the robot grippers when they face each other (as they are set in Fig. 9).

4.3. Experimental results

In this section, we address challenges associated with real data, including noise and non-ideal segmentation. While explicit occlusion handling is beyond the scope of this work, our multi-scale approach effectively mitigates issues such as noise and gripper-induced occlusions, ensuring proper mapping.

We present 6 experiments that represent several scenarios for the application of the shape control strategy. Each experiment is presented through 8 column-wise graphics. See, for example, Fig. 12), where the first 4 rows show a sequence of different illustrative time instants during the deformation process (in chronological order). On these frames the current contour segmentation and the target contour are

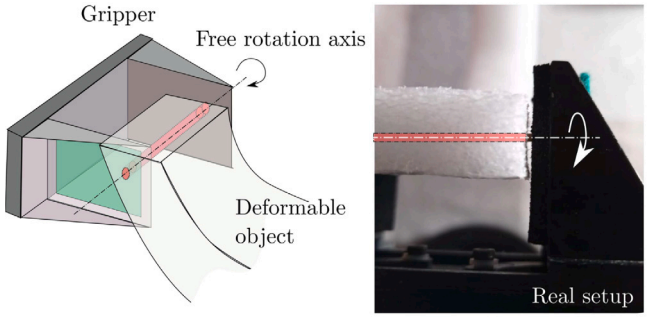


Fig. 10. Grasping of the objects. The deformable object is pierced by a thin rod (represented in red) so that it can freely rotate in the plane defined by the two grippers.

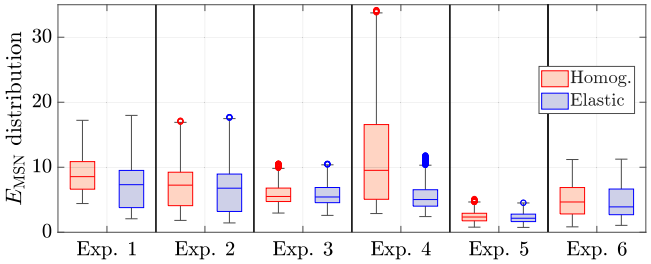


Fig. 11. Comparison of inferred deformation cost distributions between homogeneous (conventional) and elastic (proposed) mapping methods across the six experiments, in order of appearance, of Figs. 12 and 13. The box-charts show the deformation cost distribution along the contour, with the elastic mapping generally performing better (lower deformation costs), especially in cases with significant elastic deformations (e.g., Exp. 4).

Table 1
Comparison of the mean ($\bar{E}_{MSN}$) and standard deviation (σ_{MSN}) of inferred deformation costs for homogeneous and elastic mapping methods across the six experiments. Bold (larger) values in the mean cost indicate the best performance. Narrower cost distributions indicate uniform deformation across the object, while wider distributions show deformation concentrated in specific object regions.

		Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	Exp.6
$\bar{E}_{MSN}$	Homog.	8.71	7.03	5.82	11.18	2.40	5.00
	Elastic	**6.89**	**6.52**	**5.72**	**5.48**	**2.23**	**4.86**
σ_{MSN}	Homog.	**2.52**	**3.04**	**1.45**	6.82	0.86	**2.47**
	Elastic	3.04	3.26	1.54	**2.02**	**0.78**	2.65

represented in red and blue respectively. Our proposed multi-scale contour mapping is displayed with thin grey lines, the gripper positions are displayed with green squares and a white number identifying them (grippers 1 and 2). In the fifth and sixth rows both the initial and final $\mathbf{F}_{shifted}$ surfaces and their corresponding assignation paths $\mathcal{P}$ are shown. In the seventh row a plot displays the boundary error $\|\mathbf{E}_k\|$. The boundary error $\|\mathbf{E}_k\|$, expressed in millimetres, is represented by its mean value (red line) and the range covered by its standard deviation (shaded regions). The last row shows, in millimetres, the components of the actions of both grippers: $\Delta\gamma_{g,x}, \Delta\gamma_{g,y}, \Delta\gamma_{g,z}$ (red, green and blue respectively) with respect to the world reference that is displayed in Fig. 9.

The first experiment (first column in Fig. 12) is a case of large-scale deformation in which a long polyethylene bar is bent. In this first experiment, the grippers are conveniently positioned. However, in the second column of Fig. 12, the same shape control problem is solved but with unfavourable gripper positions. The control system still manages

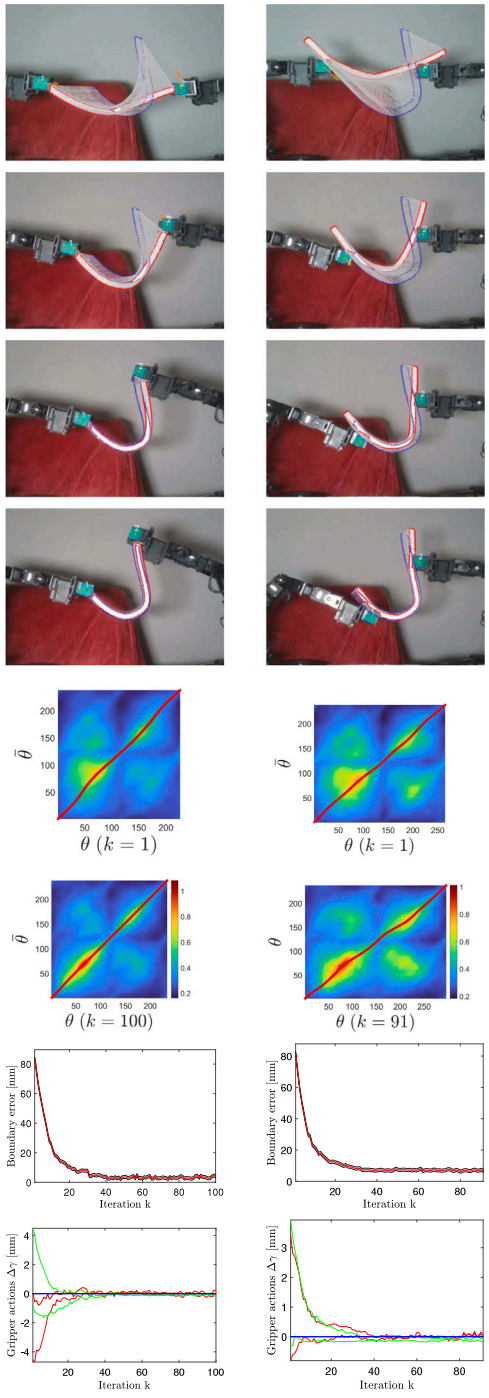


Fig. 12. Two experiments involving the same shape control problem with favourable (left) and inconvenient (right) gripper configurations. See Section 4.3 for a detailed explanation of the elements that constitute graphs and plots. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

to obtain the target shape (with larger final error). In Fig. 13 the first column constitutes a deformation case in which the polyethylene bar is deformed with a very inconvenient gripper configuration (half of the bar is cantilevered). The next experiment (second column in Fig. 13) involves the stretching process of a wrinkled paper napkin in which the horizontal axis data in $\mathbf{F}_{shifted}$ expands as the object is stretched and more contour points are retrieved in each frame. In the third column of Fig. 13, as a task related to food manipulation, a piece of banana skin is placed back to its original position on the banana. Finally, the last column in Fig. 13 constitutes a case of a local deformation process carried out on a sponge with a hole. It is particularly interesting to see how the assignation path $\mathcal{P}$ loses curvature until it practically becomes a straight line, indicating greater similarity between the final shape and the target shape.

Fig. 11 and Table 1 present a comparison of our proposed elastic mapping method with the standard homogeneous mapping method (e.g., as in [12,17]) for each shape control tasks. They are evaluated using a deformation cost metric derived from the multi-scale Laplacian (curvature) error E_{MSN} as defined in [26]. That is, for each shape control task, the deformation cost that each mapping method generates is quantified. The results, which include the deformation cost distributions (E_{MSN}), mean ($\bar{E}_{\text{MSN}}$), and standard deviation (σ_{MSN}), provide insight into the deformation inferred by each mapping method. Larger mean values ($\bar{E}_{\text{MSN}}$) indicate greater overall deformation, while the dispersion of the distribution (σ_{MSN}) reveals how this deformation cost is applied: narrower cost distributions imply more homogeneously widespread deformation all over the object, whereas wider cost distributions indicate that deformation is concentrated on the specific parts of the object. The elastic mapping demonstrates superior performance compared to the homogeneous mapping by inferring lower overall deformation costs. It achieves this through estimating concentrated yet effective deformations, especially in scenarios involving significant elastic deformations (e.g., Experiment 4). Nonetheless, even in near-isometric cases (e.g., Experiment 6), although by a smaller margin, the elastic method still demonstrates superior performance. To ensure a fair comparison, the initial parametrisation points for both methods were optimally determined using our elastic mapping framework, offering a favourable starting point for the homogeneous method.

The qualitative improvements with respect to [18] are also reflected in the experiment's results. For example, the smoothness of our elastic map and its ability to avoid infeasible matches are illustrated in the $\mathbf{F}_{shifted}$ surfaces, where patterns that would lead to abrupt path changes or to non-injective paths do not emerge. In contrast, [18] leads to similarity surfaces with patterns exhibiting sudden changes and infinite or negative slopes relative to the main diagonal, which result in locally non-injective paths $\mathcal{P}$. To ensure injectivity under these conditions, users would need to employ very large values of β (e.g., $\beta > 10$), potentially overlooking non-isometric deformations and diminishing the advantages of the elastic mapping method. Furthermore, the automatic definition of initial matching points is consistently validated in experiments by gradient descent trajectories (i.e., paths $\mathcal{P}$) reliably aligning and following the main diagonal of the comparison surface $\mathbf{F}$. For additional insights into these experiments and simulations, we refer the reader to the **accompanying video**.

5. Conclusions

In this paper we have proposed an FMM-based multi-scale contour mapping for shape control. Through simulations and experiments with a proposed basic shape control system, we validate the use and applicability of our novel multi-scale elastic mapping method. Our elastic contour mapping method successfully allowed to handle diverse shape control tasks using a basic deformation model and control approach. This suggests its applicability in more advanced shape control strategies described in existing literature (e.g. strategies involving a finer identification of the deformation model). It would also be interesting to explore a generalisation to higher dimensions of the elastic mapping method presented here. Another potential future direction is to benchmark existing shape control methods with and without our elastic mapping to assess performance improvements, though this would require addressing challenges in implementation. Additionally, exploring dynamic gripper repositioning during the manipulation process could potentially reduce shape error.

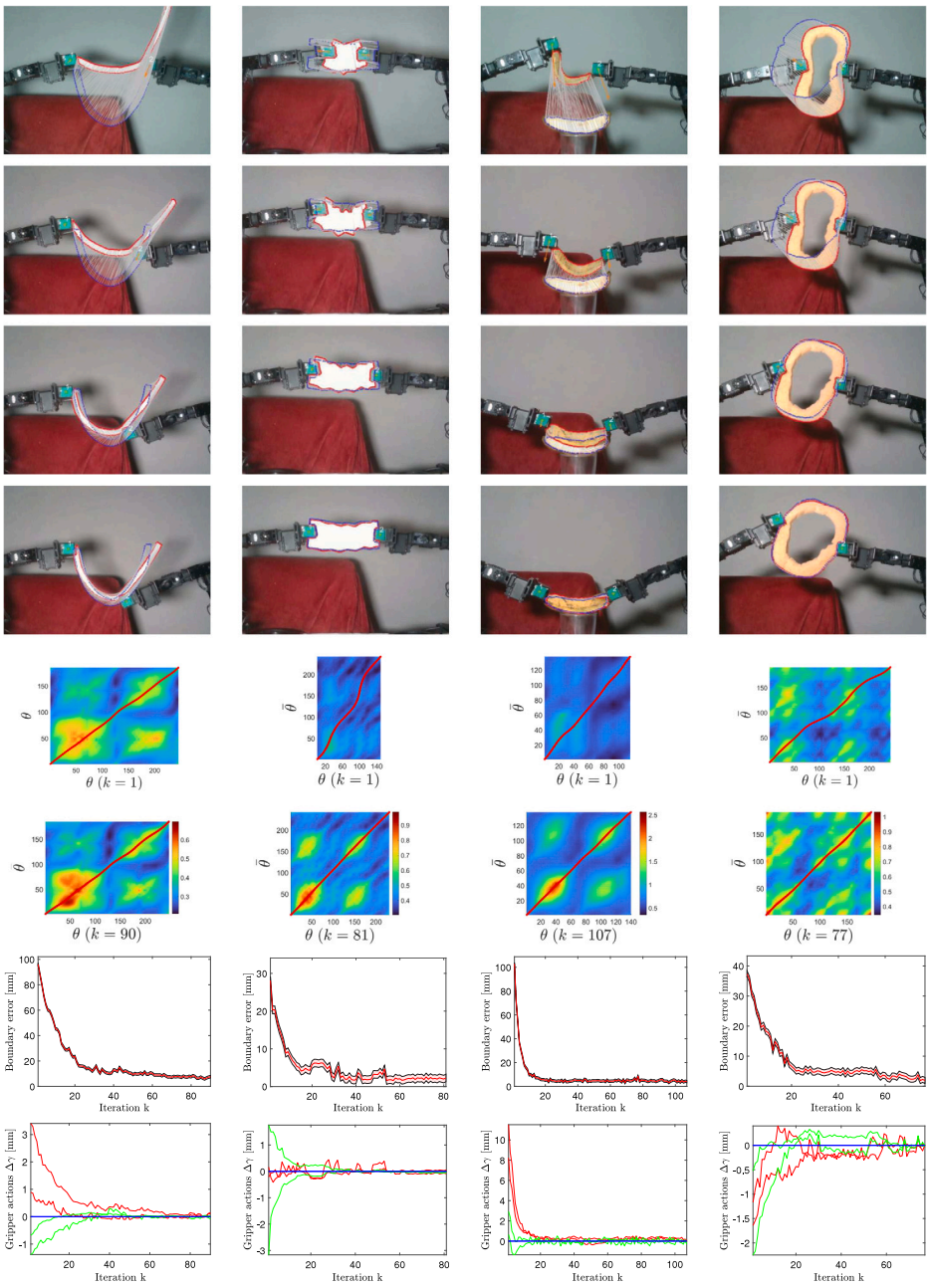


Fig. 13. Four experiments involving different materials and deformation cases, from left to right: polyethylene bar being bent with unfavourable gripper configuration, wrinkled napkin being stretched, banana skin being placed back to its original position and sponge undergoing local deformation. See Section 4.3 for a detailed explanation of the elements that constitute graphs and plots. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

CRediT authorship contribution statement

Ignacio Cuiral-Zueco: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Gonzalo López-Nicolás:** Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing – review & editing, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported via project REMAIN S1/1.1/E0111 (Interreg Sudoe Programme, ERDF) and projects PID2021-124137OB-I00 and TED2021-130224B-I00 funded by MCIN/AEI/10.13039/501100011033, by ERDF A way of making Europe and by the European Union NextGenerationEU/PRTR.

Appendix. Shape control strategy

In this appendix we introduce a basic deformation model and a control law to validate the applicability to shape control of our proposed FMM-based elastic mapping. The model is motivated by the concept of diminishing rigidity presented in [7] and also includes an object segmentation into regions that constitute an approximation of

the gripper's influence. This makes the control law presented here more robust to unfavourable gripper configurations in comparison to our previously proposed control law [18], which constitutes a more local approach.

A.1. Object deformation model

The model proposed here is acquired by analysing the geometry and the gripper configuration of the at-rest state of the elastic object (neither stretched nor compressed). The model approximates the extent to which object points respond to a gripper action on a rigid manner when lying at a specific topological distance from the gripper. Points' rigid response decreases as they lie further away from the grippers. Recall that the gripper dynamics is modelled by a single integrator $\dot{\boldsymbol{\gamma}}_g = \mathbf{u}_g = (u_{g,x}, u_{g,y}, u_{g,z})$, being $\dot{\boldsymbol{\gamma}}_g$ the gripper's velocity. Given node v_n and an action $\mathbf{u}_g$ on gripper γ_g, we model v_n's velocity $\dot{\mathbf{v}}_n \in \mathbb{R}^3$ caused by $\mathbf{u}_g$ as:

$$\dot{\mathbf{v}}_n^\intercal = \mathbf{J}(n,g)\mathbf{u}_g^\intercal, \tag{A.1}$$

where $\mathbf{J}(n,g) \in \mathbb{R}^{3\times 3}$ is $\mathbf{J}(n,g) = J(n,g)\,\mathbf{I}_3$. Being $\mathbf{I}_3 \in \mathbb{R}^{3\times 3}$ the identity matrix and $J(n,g)$:

$$J(n,g) := \begin{cases} \exp\left(-\frac{d_{min}(n)}{d_{max}}\right); & \text{if } d(n,g) = d_{min}(n) \\ 0; & \text{otherwise.} \end{cases} \tag{A.2}$$

To compute the velocities of all the object points we define matrix $\mathcal{J} \in \mathbb{R}^{3N\times 3G}$ as $\mathcal{J} = [\mathbf{J}(n,g)]$. Distance $d(n,g)$ in (A.2) is the geodesic distance (along the object's surface Θ) between gripper's γ_g at-rest positions $\boldsymbol{\gamma}_g(t_0) \in \mathbb{R}^3$ and v_n's at-rest position $\mathbf{v}_n(t_0) \in \mathbb{R}^3$. Geodesic distance $d_{min}(n) = \min(d(n,1), \dots, d(n,G))$ represents the minimum distance from v_n to any gripper γ_g in the at-rest configuration of the object. The furthest an object point is from its nearest gripper is $d_{max} = \max(d_{min}(1), \dots, d_{min}(N))$. Term $J(n,g)$ defines an object surface segmentation that keeps track of which points' dynamics are being dominated by each gripper.

The velocities $\dot{\mathbf{v}}_n$ of each of the object points are stacked in the column vector $\dot{\mathbf{V}} \in \mathbb{R}^{3N}, \dot{\mathbf{V}} = \left[(\dot{\mathbf{v}}_1)^\intercal, \dots, (\dot{\mathbf{v}}_N)^\intercal\right]^\intercal$ and are defined as:

$$\dot{\mathbf{V}}(t) = \mathcal{J}\mathbf{U}(t) = \mathcal{J}\dot{\boldsymbol{\Gamma}}(t), \tag{A.3}$$

where matrix $\mathbf{U} \in \mathbb{R}^{3G}, \mathbf{U} = \left[(\mathbf{u}_1)^\intercal, \dots, (\mathbf{u}_G)^\intercal\right]^\intercal$ stacks all the action vectors $\mathbf{u}_g$ in a column matrix. Matrix $\boldsymbol{\Gamma} \in \mathbb{R}^{3G}, \boldsymbol{\Gamma} = \left[(\boldsymbol{\gamma}_1)^\intercal, \dots, (\boldsymbol{\gamma}_G)^\intercal\right]^\intercal$ stacks gripper positions and, given our single integrator dynamics, $\dot{\boldsymbol{\Gamma}} = \mathbf{U}$. Regarding (A.3), the object state over time (given at-rest configuration $\mathbf{V}(t_0)$) yields:

$$\mathbf{V}(t) = \mathbf{V}(t_0) + \mathcal{J}(\boldsymbol{\Gamma}(t) - \boldsymbol{\Gamma}(t_0)), \tag{A.4}$$

where initial gripper state $\boldsymbol{\Gamma}(t_0)$ is determined from the gripper positions in the at-rest object state. Note that $\mathcal{J}$ does not change with time: we are modelling elastic deformations and time changing values of $\mathcal{J}$ would be representing elasto-plastic deformations.

A.2. Control law

We define matrix $\mathcal{J}_b \in \mathbb{R}^{3M\times 3G}, \mathcal{J}_b = \mathbf{S}\mathcal{J}$ as the matrix defining the contour points kinematics. Matrix $\mathbf{S} \in \mathbb{R}^{3M\times 3N}$ acts as a selector matrix that retrieves the rows from $\mathcal{J}$ that define contour points velocity components. Note that, if contour points $V^b(t)$'s indexes $m = 1, \dots, M$ are equal to the first M indexes in V, matrix $\mathbf{S} = [\mathbf{I}_{3M}, \mathbf{0}_{3M\times 3(N-M)}]$. From now on time dependence notation is omitted when it can be easily inferred.

The error $\mathbf{e}_m \in \mathbb{R}^3$ for a contour point v_m^b with its matched target contour point $\bar{v}_m^b$ is $\mathbf{e}_m = \mathbf{v}_m^b - \bar{\mathbf{v}}_m^b$, being $\mathbf{v}_m^b, \bar{\mathbf{v}}_m^b \in \mathbb{R}^3$ the position vectors of $v_m^b, \bar{v}_m^b$ respectively. All of the error vectors $\mathbf{e}_m$ are stacked column-wise in $\mathbf{E} \in \mathbb{R}^{3M}, \mathbf{E} = \left[(\mathbf{e}_1)^\intercal, \dots, (\mathbf{e}_M)^\intercal\right]^\intercal$. The equation that defines error $\mathbf{E}$ with respect to time is:

$$\mathbf{E}(\boldsymbol{\Gamma}(t)) = \mathbf{V}^b(\boldsymbol{\Gamma}(t)) - \bar{\mathbf{V}}^b, \tag{A.5}$$

where $\mathbf{V}^b \in \mathbb{R}^{3M}$ stacks vectors $\mathbf{v}_m^b$ as $\mathbf{V}^b = \left[(\mathbf{v}_1^b)^\intercal, \dots, (\mathbf{v}_M^b)^\intercal\right]^\intercal$ and, similarly, $\bar{\mathbf{v}}_m^b$ are stacked in $\bar{\mathbf{V}}^b \in \mathbb{R}^{3M}, \bar{\mathbf{V}}^b = \left[(\bar{\mathbf{v}}_1^b)^\intercal, \dots, (\bar{\mathbf{v}}_M^b)^\intercal\right]^\intercal$. Assuming constant reference $\bar{\mathbf{V}}^b$, the error (A.5) derivative is

$$\dot{\mathbf{E}}(t) = \mathcal{J}_b\dot{\boldsymbol{\Gamma}}(t), \tag{A.6}$$

regarding (A.6), our control law yields:

$$\mathbf{U} = \dot{\boldsymbol{\Gamma}} = -\mathcal{J}_b^+\mathbf{E}(\boldsymbol{\Gamma}), \tag{A.7}$$

where $\mathcal{J}_b^+$ is the Moore–Penrose left pseudo-inverse. We propose the discrete equivalent for (A.7):

$$\mathbf{U}_k = \Delta\boldsymbol{\Gamma}_k = -\xi\mathcal{J}_b^+\mathbf{E}(\boldsymbol{\Gamma}_k), \tag{A.8}$$

where $k \in \mathbb{N}$ denotes the iteration number and $\xi \in \mathbb{R}, \xi > 0$ the step-length. A stability analysis of the discrete system (A.8) (involving the choice of ξ) is presented below.

A.3. Stability analysis of the proposed control system

In this appendix we begin by providing stability analysis of the continuous system constituted by (A.6) and control law (A.7). Then, we analyse the stability conditions for the error system (A.6) and control law (A.8) (discrete analogous of (A.7)).

Lemma A.1. *Error function* $\mathbf{E}(\boldsymbol{\Gamma}) : D \subset \mathbb{R}^{3G} \to \mathbb{R}^{3M}$ *is continuous differentiable as* $\mathcal{J}_b$ *constitutes its Fréchet derivative* $\forall\, \boldsymbol{\Gamma} \in D$.

Proof. By definition of Eq. (A.6).

Proposition A.2. *Function*

$$\mathcal{V}(\boldsymbol{\Gamma}) = \frac{1}{2}\left\|\mathbf{E}(\boldsymbol{\Gamma})\right\|^2 - \frac{1}{2}\left\|\mathbf{E}(\boldsymbol{\Gamma}^*)\right\|^2. \tag{A.9}$$

is a Lyapunov function for the continuous system constituted by (A.6) and (A.7), being $\boldsymbol{\Gamma}^*$ *an equilibrium point.*

Proof. $\mathcal{V}(\boldsymbol{\Gamma}^*) = 0$ and $\mathcal{V}(\boldsymbol{\Gamma}) > 0\ \forall \boldsymbol{\Gamma} \neq \boldsymbol{\Gamma}^*$. Given Lemma A.1, $\mathbf{E}(\boldsymbol{\Gamma})$ is continuous differentiable and for $\boldsymbol{\Gamma}(t)$ any solution to (A.7), $d\,\mathcal{V}(\boldsymbol{\Gamma}(t))/dt$ is:

$$\frac{d}{dt}\mathcal{V}(\boldsymbol{\Gamma}(t)) = -\mathbf{E}^\intercal\mathcal{J}_b\mathcal{J}_b^+\mathbf{E}, \tag{A.10}$$

where, for clarity, the dependence on $\boldsymbol{\Gamma}(t)$ of the right-hand side terms has been omitted. In (A.10), $\mathcal{J}_b\mathcal{J}_b^+$ is the orthogonal projector on the column space of $\mathcal{J}_b$ and thus is positive semi-definite, therefore $d\,\mathcal{V}(\boldsymbol{\Gamma}(t))/dt \leq 0$.

Lemma A.3. *The smallest nonzero eigenvalue of* $\mathcal{J}_b^\intercal\mathcal{J}_b$, $\alpha(\boldsymbol{\Gamma})$, *is bounded away from zero by* $1\ \forall\, \boldsymbol{\Gamma} \in D$. *Where* D *is an open convex set such that* $\mathbf{E}(\boldsymbol{\Gamma}) : D \subset \mathbb{R}^{3G} \to \mathbb{R}^{3M}$.

Proof. Each gripper g has three associated identical eigenvalues α_g for $\mathcal{J}_b^\intercal\mathcal{J}_b$. As a contour point v_m^b can only be grabbed by one gripper γ_g, these eigenvalues can be computed as:

$$\alpha_g = \sum_{m=1}^{M} J^2(m,g). \tag{A.11}$$

Regarding the smallest possible value of α_g, the worst-case scenario would imply every $J(m,g) = 0$ for every contour point except for the gripper's contour point, which always presents $J(m,g) = 1$ (recall definition of $J(m,g)$ in (A.2)). In the worst-case scenario, (A.11) would result in $\alpha_g = 1$ implying $\alpha(\boldsymbol{\Gamma}) \geq 1\ \forall\, \boldsymbol{\Gamma} \in D$. This not only holds for the proposed function (A.2) but also for any function that may be considered as object deformation model as long as it presents non-increasing values (with respect to increasing gripper-to-point topological distances) and as long as such values are not time-varying (i.e., the object behaves purely elastically and no elasto-plastic deformations take place).

Proposition A.4. *Let $\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma})$ be Lipschitz continuous (Lemma A.1) with constant K on compact set $\mathcal{D}$ and let constant $c > 0$, $c \in \mathbb{R}$, be independent of $\mathbf{\Gamma}(t)$ such that*

$$-d\mathcal{V}/dt \geq c \left\| \mathcal{J}_b^+ \mathbf{E} \right\|. \tag{A.12}$$

Then, $\mathcal{V}(\mathbf{\Gamma})$ in (A.9), which is Lyapunov for the continuous system constituted by (A.6) and (A.7) (Proposition A.2), is also Lyapunov for the system constituted by (A.6) and the discrete control law (A.8) given step-lengths $\xi \in (0, 2c/K)$.

Proof. Proposition A.4 constitutes a direct application of Theorem 2.6 in [27]. We need to prove that, for $\mathbf{\Gamma} \in D$, if $\mathbf{\Gamma} \neq \mathbf{\Gamma}^*$ and $\xi \in (0, 2c/K)$,

$$\mathcal{V}(\mathbf{\Gamma} - \xi \mathcal{J}_b^+ \mathbf{E}(\mathbf{\Gamma})) \leq \mathcal{V}(\mathbf{\Gamma}) \tag{A.13}$$

is satisfied. This is equivalent to proving (A.12) holds. Substituting (A.10) and expanding the right-side term, (A.12) yields:

$$\mathbf{E}^\intercal \mathcal{J}_b \mathcal{J}_b^+ \mathbf{E} \geq c\, \mathbf{E}^\intercal \mathcal{J}_b^{+\intercal} \mathcal{J}_b^+ \mathbf{E}, \tag{A.14}$$

with $c > 0$. Since $\mathcal{J}_b = \mathcal{J}_b^{+\intercal} \mathcal{J}_b^\intercal \mathcal{J}_b$, (A.14) is equivalent to:

$$(\mathbf{E}^\intercal \mathcal{J}_b^{+\intercal}) \mathcal{J}_b^\intercal \mathcal{J}_b (\mathcal{J}_b^+ \mathbf{E}) \geq c\, \mathbf{E}^\intercal \mathcal{J}_b^{+\intercal} \mathcal{J}_b^+ \mathbf{E}. \tag{A.15}$$

Therefore, $(\mathbf{E}^\intercal \mathcal{J}_b^{+\intercal}) \mathcal{J}_b^\intercal \mathcal{J}_b (\mathcal{J}_b^+ \mathbf{E}) > c\ \mathbf{E}^\intercal \mathcal{J}_b^{+\intercal} \mathcal{J}_b^+ \mathbf{E}$ holds when $c > \alpha(\mathbf{\Gamma})$, being $\alpha(\mathbf{\Gamma})$ the smallest nonzero eigenvalue of $\mathcal{J}_b^\intercal \mathcal{J}_b$. $\alpha(\mathbf{\Gamma})$ must be uniformly bounded away from zero $\forall\ \mathbf{\Gamma} \in \mathcal{D}$. In particular, given Lemma A.3, $\alpha(\mathbf{\Gamma}) \geq 1\, \forall\ \mathbf{\Gamma} \in \mathcal{D}$. Given the Lipschitz condition and by Ortega and Rheinboldt ([28], Theorem 3, 2.12), condition in (A.13) holds when $\xi c > K\xi^2/2$, therefore (A.12) holds for step-lengths $\xi \in (0, 2c/K)$.

Lemma A.5. *There is a unique error state $\mathbf{E}(\mathbf{\Gamma})$ for each gripper state $\mathbf{\Gamma}$. That is, function $\mathbf{E}(\mathbf{\Gamma})$ is injective ($\mathbf{E}(\mathbf{\Gamma}_1) = \mathbf{E}(\mathbf{\Gamma}_2)$ implies $\mathbf{\Gamma}_1 = \mathbf{\Gamma}_2$).*

Proof. Note that deformation model (A.3)–(A.4), with error dynamics (A.6), determines the feasible error states. That is, the deformation model only allows to reach $\mathbf{E}(t)$ within the image domain $\mathbf{E}[\mathbf{\Gamma}] = \{\mathbf{E}(\mathbf{\Gamma}) \text{ in (A.5)} : \forall\, \mathbf{\Gamma} \in \mathbb{R}^{3G}\}$, therefore $\mathbf{E}[\mathbf{\Gamma}] \subset \mathbb{R}^{3M}$. Considering (A.4) and (A.5), we can analyse $\mathbf{E}(t)$:

$$\begin{aligned} \mathbf{E}(t) &= \mathbf{E}(t_0) + \mathcal{J}_b \left(\mathbf{\Gamma}(t) - \mathbf{\Gamma}(t_0) \right), \\ &\text{with } \mathcal{J}_b = \left(\mathbf{I}_{3M} \circ \left(\mathbf{1}_{3M}^\intercal \otimes (\mathcal{J}_b \mathbf{1}_{3G}) \right) \right) \left(S \otimes \mathbf{I}_3 \right). \end{aligned} \tag{A.16}$$

Where operators $\circ$ and $\otimes$ denote the Hadamard and the Kronecker product, respectively. In (A.16), $\mathbf{1}_{3M}$ and $\mathbf{1}_{3G}$ are column vectors of ones and matrix $S \in \mathbb{R}^{M \times G}$ is defined by elements $s_{m,g} = 1$ when $J(m,g) > 0$ and $s_{m,g} = 0$ otherwise. Note that only one element per row of S is non-zero and thus $\left(S \otimes \mathbf{I}_3 \right) \left(\mathbf{\Gamma}(t) - \mathbf{\Gamma}(t_0) \right)$ replicates and vertically stacks elements of $\left(\mathbf{\Gamma}(t) - \mathbf{\Gamma}(t_0) \right)$. This means that term $\left(S \otimes \mathbf{I}_3 \right) \left(\mathbf{\Gamma}(t) - \mathbf{\Gamma}(t_0) \right)$ constitutes an injective function for $\mathbf{\Gamma}(t)$. On the other hand, $\left(\mathbf{I}_{3M} \circ \left(\mathbf{1}_{3M}^\intercal \otimes (\mathcal{J}_b \mathbf{1}_{3G}) \right) \right)$ is a definite positive diagonal matrix (invertible) and thus constitutes a bijection. The composition of an injection and a bijection renders (A.16) an injective map between $\mathbf{\Gamma}$ and $\mathbf{E}$, that is, there is a unique error state $\mathbf{E}(\mathbf{\Gamma})$ for each $\mathbf{\Gamma}$.

Lemma A.6. *Gripper state $\mathbf{\Gamma}^*$ such that $\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma}^*) = 0$ is unique.*

Proof. Regarding (A.4) and (A.5), the evolution of $\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma})$ with respect to $\mathbf{\Gamma}(t)$ can be expressed as

$$\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma}(t)) = \mathcal{J}_b^\intercal \mathbf{E}(t_0) + \mathcal{J}_b^\intercal \mathcal{J}_b (\mathbf{\Gamma}(t) - \mathbf{\Gamma}(t_0)), \tag{A.17}$$

with $\mathbf{E}(t_0)$ the error state when the object is in its at-rest configuration (i.e., $\mathbf{E}(t_0) = \mathbf{V}^b(t_0) - \bar{\mathbf{V}}^b$). Since $\mathcal{J}_b^\intercal \mathcal{J}_b$ in (A.17) is a diagonal matrix with all positive values (i.e., positive definite and thus invertible) system (A.17) presents uniqueness of solution $\mathbf{\Gamma}^*$ for which $\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma}^*) = \mathbf{0}$. Therefore, given an object with at-rest error state $\mathbf{E}(t_0)$ and model $\mathcal{J}_b$ determined by both the at-rest shape $\mathbf{V}(t_0)$ and the initial gripper configuration $\mathbf{\Gamma}(t_0)$, $\mathbf{\Gamma}^*$ constitutes the unique solution to $\mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma}^*) = 0$. Recall $\mathbf{E}(\mathbf{\Gamma})$ cannot be arbitrary, the feasible states of $\mathbf{E}(\mathbf{\Gamma})$ are determined by model (A.3)–(A.4) that leads to error dynamics (A.6) (Lemma A.5).

Lemma A.7. *Matrix $\mathcal{J}_b$ has constant rank on $\mathcal{D}$.*

Proof. There is always at least G fully actuated contour points, i.e. the contour points grabbed by the grippers (being G constant). These points always present $d_{min}(n) = 0$ and thus, given (A.2), $\mathbf{J}(n,g)$ for gripper points is always $\mathbf{J}(n,g) = \mathbf{I}_3$, which ensures $\mathcal{J}_b$ has always full rank $3G$ (i.e. constant rank).

Note that the proposed analysis allows for a broader scope of application than the linear model considered in Appendix A.1, as it can encompass non-linear models (provided that they meet the requirements set out herein, in particular Lemmas A.1, A.3, A.5, A.6 and A.7).

The following Theorem for global asymptotic stability of the system (A.6) and (A.8) is an adaptation of Theorem 3.3, in [27].

Theorem A.8. *Let $\mathbf{E}(\mathbf{\Gamma}) : \mathcal{D} \subset \mathbb{R}^{3G} \to \mathbb{R}^{3M}$ be continuous differentiable (Lemma A.1) and $\mathcal{J}_b^\intercal \mathbf{E}$ be Lipschitz continuous on the open convex set $\mathcal{D}$. Let $\mathcal{J}_b$ have constant rank on $\mathcal{D}$ and $\mathcal{Z} \subset \mathcal{D}, \mathcal{Z} = \{\mathbf{\Gamma} : \mathcal{J}_b^\intercal \mathbf{E}(\mathbf{\Gamma}) = \mathbf{0}\}$ be bounded. Then there is constant ξ such that any limit point from the discrete sequence defined by the discrete control law (A.8) from any $\mathbf{\Gamma}_k \in \mathcal{D}$, using step-lengths $\xi \in (0, 2c/K)$, is a member of $\mathcal{Z}$. Furthermore, regarding model (A.3)(A.4) with error dynamics (A.6), set $\mathcal{Z}$ is only constituted by one equilibrium point $\mathbf{\Gamma}^*$ that leads to a unique error equilibrium $\mathbf{E}(\mathbf{\Gamma}^*)$.*

Proof. Given Lemma A.7 $\mathcal{J}_b$ has constant rank on $\mathcal{D}$, since $\mathcal{V}$ in (A.9) is a Lyapunov function on $\mathcal{Z}$ for (A.8) (Proposition A.4) and $\mathcal{Z}$ is bounded, uniformity conditions on $\mathcal{V}$ and $\mathcal{J}_b^+ \mathbf{E}$ are ensured. These conditions allow for the application of Theorem 2.9 from [27] which proves that, for any $\mathbf{\Gamma} \in D$, the sequence defined in (A.8), with $\xi \in (0, 2c/K)$, converges to $\mathcal{Z}$. Set $\mathcal{Z}$ is constituted by a unique gripper state $\mathbf{\Gamma}^*$ (Lemma A.6) and, since $\mathbf{E}(\mathbf{\Gamma})$ is an injective function (Lemma A.5), the error equilibrium $\mathbf{E}(\mathbf{\Gamma}^*)$ associated to $\mathbf{\Gamma}^*$ is also unique. Given uniqueness of $\mathbf{E}(\mathbf{\Gamma}^*)$, we conclude global asymptotic stability of the system constituted by the error system (A.6) and the discrete control law (A.8).

Remark A.9. The global minimum residual $\|\mathbf{E}(\mathbf{\Gamma}^*)\|$ is determined by the at-rest object geometry and the target shape ($\mathbf{E}(t_0) = \mathbf{V}^b(t_0) - \bar{\mathbf{V}}^b$ in (A.16)) as well as by the gripper configuration (through the definition of the model (A.2)–(A.3)). Global minimum $\|\mathbf{E}(\mathbf{\Gamma}^*)\|$ will be lower in those systems that present favourable gripper configurations and larger in those in which the gripper configuration is inconvenient (see [29] for a method on favourable gripper positioning for deformable object manipulation).

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.robot.2025.105134.

Data availability

No data was used for the research described in the article.

References

[1] I. Cuiral-Zueco, G. López-Nicolás, Taxonomy of deformable object shape control, IEEE Robot. Autom. Lett. 9 (10) (2024) 9015–9022.

[2] J. Sanchez, J.A. Corrales, B.C. Bouzgarrou, Y. Mezouar, Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey, Int. J. Robot. Res. 37 (7) (2018) 688–716.

[3] R. Herguedas, G. López-Nicolás, R. Aragüés, C. Sagüés, Survey on multi-robot manipulation of deformable objects, in: 24th IEEE International Conference on Emerging Technologies and Factory Automation, 2019, pp. 977–984.

[4] H. Yin, A. Varava, D. Kragic, Modeling, learning, perception, and control methods for deformable object manipulation, Sci. Robot. 6 (54) (2021) eabd8803.

[5] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li, et al., Challenges and outlook in robotic manipulation of deformable objects, IEEE Robot. Autom. Mag. 29 (3) (2022) 67–77.

[6] J. Das, N. Sarkar, Autonomous shape control of a deformable object by multiple manipulators, J. Intell. Robot. Syst. 62 (1) (2011) 3–27.

[7] D. Berenson, Manipulation of deformable objects without modeling and simulating deformation, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 4525–4532.

[8] M. Aranda, J.A. Corrales Ramon, Y. Mezouar, A. Bartoli, E. Özgür, Monocular visual shape tracking and servoing for isometrically deforming objects, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2020, pp. 7542–7549.

[9] D. Navarro-Alarcón, Y. Liu, J.G. Romero, P. Li, Model-free visually servoed deformation control of elastic objects by robot manipulators, IEEE Trans. Robot. 29 (6) (2013) 1457–1468.

[10] D. Navarro-Alarcon, H.M. Yip, Z. Wang, Y.H. Liu, F. Zhong, T. Zhang, P. Li, Automatic 3-D manipulation of soft objects by robotic arms with an adaptive deformation model, IEEE Trans. Robot. 32 (2) (2016) 429–441.

[11] D. Navarro-Alarcon, Y.H. Liu, Fourier-based shape servoing: a new feedback method to actively deform soft objects into desired 2-D image contours, IEEE Trans. Robot. 34 (1) (2017) 272–279.

[12] J. Zhu, D. Navarro-Alarcon, R. Passama, A. Cherubini, Vision-based manipulation of deformable and rigid objects using subspace projections of 2-D contours, Robot. Auton. Syst. 142 (2021) 103798.

[13] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, A. Cherubini, Dual-arm robotic manipulation of flexible cables, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2018, pp. 479–484.

[14] J. Qi, G. Ma, J. Zhu, P. Zhou, Y. Lyu, H. Zhang, D. Navarro-Alarcon, Contour moments based manipulation of composite rigid-deformable objects with finite time model estimation and shape/position control, IEEE/ASME Trans. Mechatronics 27 (5) (2022) 2985–2996.

[15] O. Sorkine, M. Alexa, As-rigid-as-possible surface modeling, in: Symposium on Geometry Processing, Vol. 4, 2007, pp. 109–116.

[16] M. Shetab-Bushehri, M. Aranda, Y. Mezouar, E. Ozgur, As-rigid-as-possible shape servoing, IEEE Robot. Autom. Lett. 7 (2) (2022) 3898–3905.

[17] G. López-Nicolás, R. Herguedas, M. Aranda, Y. Mezouar, Simultaneous shape control and transport with multiple robots, in: IEEE International Conference on Robotic Computing, 2020, pp. 218–225.

[18] I. Cuiral-Zueco, G. López-Nicolás, Multi-scale Laplacian-based FMM for shape control, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2021, pp. 3792–3797.

[19] I. Cuiral-Zueco, G. López-Nicolás, RGB-D tracking and optimal perception of deformable objects, IEEE Access 8 (2020) 136884–136897.

[20] L. Zaidi, J.A. Corrales, B. Chedli Bouzgarrou, Y. Mezouar, L. Sabourin, Model-based strategy for grasping 3D deformable objects using a multi-fingered robotic hand, Robot. Auton. Syst. 95 (2017) 196–206.

[21] L. Han, H. Wang, Z. Liu, W. Chen, X. Zhang, Visual tracking control of deformable objects with a FAT-based controller, IEEE Trans. Ind. Electron. 69 (2) (2022) 1673–1681.

[22] H. Edelsbrunner, E.P. Mücke, Three-dimensional alpha shapes, ACM Trans. Graph. 13 (1) (1994) 43–72.

[23] M. Sabry Hassouna, A.A. Farag, Multistencils fast marching methods: A highly accurate solution to the Eikonal equation on Cartesian domains, IEEE Trans. Pattern Anal. Mach. Intell. 29 (9) (2007) 1563–1574.

[24] J.A. Sethian, A.M. Popovici, 3-d traveltime computation using the fast marching method, Geophysics 64 (2) (1999) 516–523.

[25] M. Frenkel, R. Basri, Curve matching using the fast marching method, in: International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, 2003, pp. 35–51.

[26] I. Cuiral-Zueco, Y. Karayiannidis, G. López-Nicolás, Contour based object-compliant shape control, IEEE Robot. Autom. Lett. 8 (8) (2023) 5164–5171.

[27] P.T. Boggs, The convergence of the Ben-Israel iteration for nonlinear least squares problems, Math. Comp. 30 (135) (1976) 512–522.

[28] J.M. Ortega, W.C. Rheinboldt, Iterative solution of nonlinear equations in several variables, Academic Press, 1970.

[29] I. Cuiral-Zueco, G. López-Nicolás, H. Araujo, Gripper positioning for object deformation tasks, in: IEEE International Conference on Robotics and Automation, 2022, pp. 963–969.

Ignacio Cuiral-Zueco received the Industrial Engineering Master degree majoring in robotics and computer vision in the Universidad de Zaragoza, Spain, in 2019. He received his Ph.D. in Systems Engineering and Computer Science from the Universidad de Zaragoza in 2024. He is a member of the Robotics, Computer Vision and Artificial Intelligence (RoPeRT) research group. His current research interests include computer vision, control engineering and robotics.

Gonzalo López-Nicolás received the Ph.D. degree in Systems Engineering and Computer Science from Universidad de Zaragoza, Spain, in 2008. He is currently Full Professor of the Departamento de Informática e Ingeniería de Sistemas, Universidad de Zaragoza. He is a member of the Instituto de Investigación en Ingeniería de Aragón (I3A). His current research interests are focused on deformable object shape control, visual control, multi-robot systems, and application of computer vision to robotics.