Chapter 9
Planning framework for robotic pizza dough stretching with a rolling pin

Jung-Tae Kim, Fabio Ruggiero, Vincenzo Lippiello, Bruno Siciliano

Abstract Stretching a pizza dough with a rolling pin is a nonprehensile manipulation. Since the object is deformable, force closure cannot be established, and the manipulation is carried out in a nonprehensile way. The framework of this pizza dough stretching application that is explained in this chapter consists of four sub-procedures: (*i*) recognition of the pizza dough on a plate, (*ii*) planning the necessary steps to shape the pizza dough to the desired form, (*iii*) path generation for a rolling pin to execute the output of the pizza dough planner, and (*iv*) inverse kinematics for the bi-manual robot to grasp and control the rolling pin properly. Using the deformable object model described in Chapter 3, each sub-procedure of the proposed framework is explained sequentially.

Jung-Tae Kim
MIKNMEK Inc., CheongAm-ro 87, Nam-gu, Pohang-si, Gyeongsangbuk-do, 37673, Republic of Korea, e-mail: tickleti1221@gmail.com

Fabio Ruggiero
CREATE Consortium & University of Naples Federico II, Department of Electrical Engineering and Information Technology, PRISMA Lab, Via Claudio 21, 80125, Naples, Italy, e-mail: fabio.ruggiero@unina.it

Vincenzo Lippiello
CREATE Consortium & University of Naples Federico II, Department of Electrical Engineering and Information Technology, PRISMA Lab, Via Claudio 21, 80125, Naples, Italy, e-mail: vincenzo.lippiello@unina.it

Bruno Siciliano
CREATE Consortium & University of Naples Federico II, Department of Electrical Engineering and Information Technology, PRISMA Lab, Via Claudio 21, 80125, Naples, Italy, e-mail: bruno.siciliano@unina.it

Table 9.1: Main symbols used in this chapter.

Definition	**Symbol**
Center of the pizza dough	$\mathbf{x}_0 = \begin{bmatrix} c_x & c_y \end{bmatrix}^T \in \mathbb{R}^2$
Generic point of the dough	$\mathbf{x} \in \mathbb{R}^2$
Boundary of the pizza dough	∂S
Thickness of the dough	$h > 0$
Frame associated to the plate	$\mathcal{D}$
Angle between the longest-axis of the dough shape and the x-axis of $\mathcal{D}$	$\theta \in \mathbb{R}$

9.1 Brief introduction

Making a pizza is a wonderful art, and it requires delicate skills like preparing a pizza dough mixed with wheat powder, water, salt, and other ingredients. An accurate ratio is put in the preparation, stretching it dynamically and quickly into a disk-shaped, saucing or dressing it with proper ingredients and quantity, and finally burning it evenly and sufficiently in a wood-burning oven.

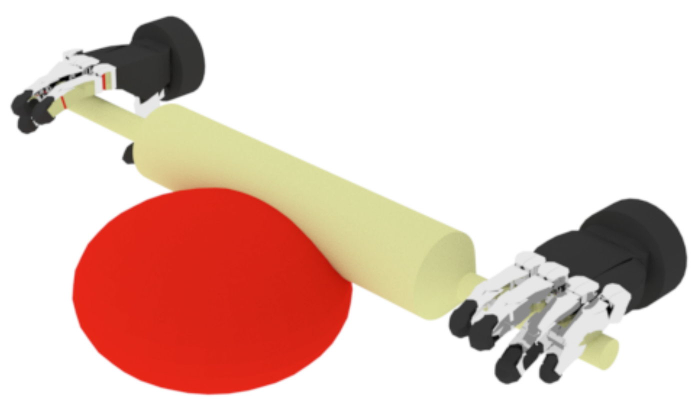


Fig. 9.1: Application: stretching a pizza dough with a rolling pin.

Among these technical processes, this chapter focuses on stretching a pizza dough with a rolling pin (Fig. 9.1). This process includes two critical techniques: (i) the manipulation of a deformable object, and (ii) the control of the rolling pin to execute the proper actions. Regarding the former, difficulties arise during the manipulation planning of a deformable object because of the absence of a precise model. Indeed, typical deformable objects can not be described with just one property but have multiply properties like viscosity, elasticity, plasticity, or others, and mixed of them. Even though several proposed models represent these properties of a deformable object, it is not easy to find a suitable one for the pizza dough. In this chapter, the SPH formulation explained in Chapter 3 is used to model a highly viscous deformable object like the pizza dough, which is eligible to describe highly deformable objects, even for liquids. Regarding the latter, researches about controlling a tool with a robotic system are well established. However, non-prehensile manipulation is still a relevant and challenging topic. A general inverse-kinematics manipulation planning for the RoDyMan robot equipped with a rolling pin has been used in this pizza dough stretching application.

Both techniques are explained in this chapter. The outline of the chapter consists of this introduction and a survey about the related state of the art (section 9.2). The sketch of the proposed framework for the pizza dough stretching is depicted in section 9.3, while the explanation of each sub-procedure is given from section 9.4 to section 9.7. Simulation results are explained in section 9.8. Finally, section 9.9 concludes the chapter.

9.2 Related research

Manipulation of a deformable or a rheological object (*i.e.*, the pizza dough) requires an understanding of the object's properties like viscosity, elasticity, and plasticity. For example, a bread dough consists of gluten proteins and various minor ingredients, including minerals. The gluten proteins play a crucial role in determining the unique baking quality of wheat by conferring water absorption capacity, cohesivity, viscosity, and elasticity. Gluten proteins can be divided into two main fractions according to their solubility in aqueous alcohols: the soluble gliadins and the insoluble glutenins [339]. It is widely accepted that gliadin accounts for the viscous properties and glutenin imparts the strength and elasticity that are necessary to hold the gases that are produced during fermentation and baking [322].

Many researchers have tried to characterize the bread dough's fundamental properties and analyze the influences of the substances. The densities of various doughs were measured in [46], while the viscoelasticity of bread dough was examined in [51, 91, 186]. The main substances for the bread dough that are H_2O (water), D_2O (heavy water), esterifying agents for glutamine residues, urea, salts, agents affecting disulfide bonding, and the protein subunits, were listed up in [3], also characterizing the influence of the substances for the rheology of the bread dough. Seventeen commercially available European wheat cultivars were sampled in [322]. Through these samples, the authors had the creep-recovery experiments and analyzed for a set of chemical and rheological parameters and baking quality using the PCA method. The dynamic rheological properties of glutens fractions with two English-grown wheat cultivars, Hereward and Riband, were studied in [154]. The authors confirmed that the viscoelasticity of the glutenin sub-fraction of gluten and differences in the ratio of gliadin to glutenin are the main factors governing inter-cultivar differences in the viscoelasticity of wheat gluten. Similarly, in [321], the authors experimented with the uniaxial elongational and shear rheology properties of doughs affected by the protein contents or glutenin-to-gliadin ratio. The conclusion was that increasing protein content lowered the maximum shear viscosity while increasing the glutenin-to-gliadin ratio increased the maximum shear viscosity. Stress and strain of the dough related to the ratio of the feed sheet thickness of the roller gap and the roller's speed in the sheeting system were studied in [87]. The authors applied the lubrication approximation for the equation of motion and used an inelastic power-law model for the dough rheology. The relationship between the rheological properties for static dough and dynamic rheological properties for dough crumbs was instead investigated in [36]. The former was evaluated by texture profile analysis, like uni-axis (stretching) and bi-axial extension (inflation), and gluten index in static compression. The latter was evaluated through dynamic mechanical analysis and thermal mechanical analysis in dynamic compression. In [200], the authors described the influence of various substances in a dough. They introduced widely used modeling methods for dough rheology like power-law, Maxwell model, Lethersich model, Peleg model, and listing the various measurement methods for dough properties like farinograph, mixograph, rheomixer, extensigraph, alveograph, amylograph, maturograph, and so on. Base on these known properties, several visco-elastoplastic models were proposed, as the Herschel-Bulkley model and the K-BKZ model [291]. Bingham model is also one of the well-known models for representing plastic properties [27].

The baking industry uses rolling (or sheeting) between counter-rotating rolls as a dough forming process for various products, such as cookies, crackers, pizza, bread, and pastry. The rolling process is akin to calendering, which is used in many industries, such as the paper, plastics, rubber, and steel industries [202, 290]. An overall process for stretching of bread doughs was designed and implemented in [316].

There are more general researches for acquiring the properties of deformable objects. A four-element model for characterizing the viscoelasticity was proposed in [132]. In [68], a neural network model was employed to estimate an object's elastic properties. A FEM model was instead employed in [333] to estimate such properties. Interactive approaches to get object's elastic properties, even without the use of any specific model, were used in [25, 99, 161, 248, 289].

9.3 Framework for a pizza dough stretching behaviour

The application handled by this chapter roughly consists of three main components: (*i*) a robotic system grasping a tool, (*ii*) a deformable object, and (*iii*) the tool itself. In this application, the employed robotic system is the RoDyMan robot, the deformable object is a pizza dough, and the tool is a rolling pin (see Fig. 9.1).

The proposed framework is described in this section. Input data from a sensor device are acquired. This generates a proper action sequence for the robot. The employed sensor device is a RGB-D Kinect camera mounted on the head of the RoDyMan robot.

The devised framework can be split into four components: (*i*) (deformable) object recognition, (*ii*) planning actions on the deformable object; (*iii*) planning actions for the tool, and (*iv*) robot manipulaiton planning. As evident from Fig. 9.2, these components are concatenated each other.

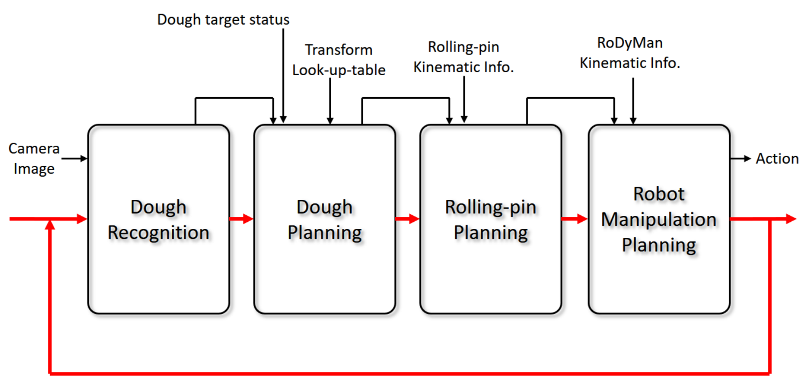


Fig. 9.2: Sketch of the overall process that consists of four concatenated components.

The first component, which is the object recognition module, gets sensor data as input. Its output is the status of the recognized object. In this application, an RGB-D Kinect camera takes pictures of the pizza dough on a plate. Afterwards, this component separates the area of the pizza dough and the background. It reconstructs the 3D shape of the pizza dough based on the 2D image data and some additional information about the pizza dough. Finally, this object recognition module describes the shape of the pizza dough through numerical data. This description indicates the current status of the pizza dough.

The second component, which is the one in charge of planning the actions on the deformable object, gest the output of the previous component, the desired final shape of the dough, and some additional information (*i.e.*, the transform look-up-table, which will be illustrated in the following sections). Its output is the planned sequence of actions on the deformable object. In this application, the current shape of the pizza dough, the desired shape of the pizza dough, and information about the dough deformation (*i.e.*, how a particular action of the rolling pin deforms the object into another shape) are given. Then, this component finds out the best sequence of actions of the rolling pin to get the desired shape.

Similarly, the third component gets the output of the previous one and generates a continuous motion of the tool to realize the desired actions on the deformable object. Each action from the previous component is disconnected from the other. Therefore, it is necessary to generate a smooth continuous motion, which is the output of this module.

Finally, the last component is in charge of the robot planning to realize the output of the previous component. Considering the kinematic information of the employed robot and all the constraints, this component generates a smooth motion sequence for the robot to suitably moving the tool as specified by the previous component.

Each component is explained in detail wtihin the sequel sections.

9.4 Pizza dough recognition

The pizza dough recognition component consists of two procedures: (*i*) digital image processing for raw data from the sensor device, and (*ii*) how to describe the status of the flatted deformable object.

9.4.1 Image processing for sensor data

The perception procedure for a deformable object roughly consists of two parts in this application; one is the image processing for the camera sensor data, and the other is the representation of the status of the pizza dough. The perception method depends highly on the application; therefore, in this case, we analyse the stretching action of a pizza dough using the RoDyMan robot (Fig. 9.3a), and the recognition procedure naturally depends on a specific detector for the pizza dough. To reduce the difficulties of the perception based on image sensor data, some artificial restrictions are applied: restriction to the workspace by defining the boundary area (*i.e.*, a plain rectangular plate) and the usage of a particular coloured deformable object (*i.e.*, the blue colour) contrasted to the background colour (*i.e.*, white colour). However, these restrictions are not critical but can be addressed with additional image processing.

With the help of the absolute position information of at least three corner markers in a rectangular plate and the kinematic information provided by the RoDyMan robot, it is not difficult to induce the frame transformation between a point in a given 2D camera image and the corresponding absolute position in the world frame [125]. In the carried out experiments, QR codes are placed at each corner of the rectangular plate, and then using an image matching with SIFT [179] the corner points were detected. Other feature detectors like the SURF [20] or the FAST [262] are also available. In order to easier detection of the corners, AR codes and the corresponding code detectors (*i.e.*, ARToolkit [149]) can be used. The primary purpose of corner detection is to remove the dough plate's outlier and induce a transformation $\mathcal{T}$ from a 2D view image to a top-down viewed 2D space.

For the initial pizza dough, assuming that the ball- or bell-shaped pizza dough is symmetric to the vertical rotation axis, the reconstruction of the 3D shape is obtainable with partial camera views or depth sensors [28, 26]. Suppose the pizza dough is shallow after being pushed by a rolling pin so that the height difference is ignorable. In that case, the transformation $\mathcal{T}$ can be directly applied to the detected pizza dough as well as the plate. However, the initial pizza dough is more like a ball- or bell-shaped than a shallow disk. We do not commit much error in seeing the pizza dough as a 2D figure on a plate with uniform thickness: indeed, the more shallow the thickness of the dough, the minor error it occurs regarding a cylinder shape.

An example of the image processing steps to identify the pizza dough on the plate is given in Fig. 9.3b-g. With reference to the labels in the figure, the steps now briefly described. *b*) The raw image data from the camera mounted on the head of the RoDyMan robot is shown. With default information, we assume that the camera's view covers the whole area of the dough plate. *c*) There are four corner marks on the white dough plate. After detecting the marks, the area of the dough plate is outlined with blue lines. *d*) The outer area of the dough plate is removed from the image. *e*) Within the dough plate, the dough clay (the green object) is detected, and the outline is extracted through an edge detector [112], like the Canny edge detector [47]. *f*) The raw 2D camera image is deformed into the orthogonal top-down view with predefined width and height. *g*) The deformed outline of the detected pizza dough is finally obtained.

9.4.2 Description for a status of a pizza dough

Pizza dough has various shapes. However, according to general experiences and our experiments in making a pizza, there is no problem assuming that the pizza dough's shape is convex. Hence, we can define the state of the pizza dough as the set of distances between the center of the dough and its boundary and the related tickness

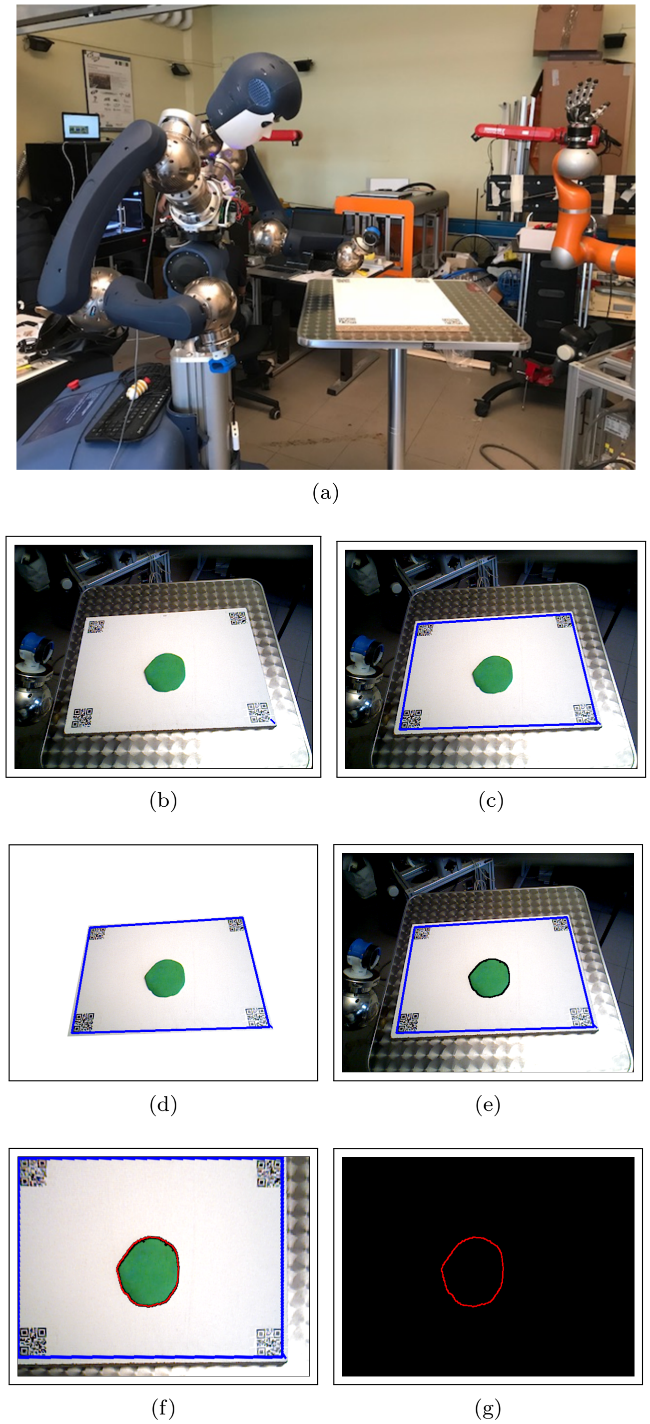


Fig. 9.3: Detection of the pizza dough on a white rectangular plate.

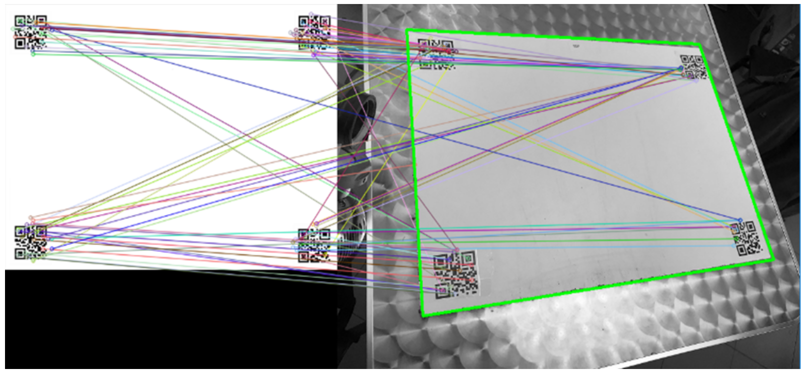


Fig. 9.4: Feature matching on the corners of the plate through the SURF method. On the left, the template image. On the right, the camera image. The green lines indicate the boundary of the detected plate.

$$\mathcal{C}_X : \left\{ \begin{bmatrix} \|\mathbf{x} - \mathbf{x}_0\| \\ h \end{bmatrix} \in \mathbb{R}^2 : \mathbf{x} \in \partial S \right\}. \tag{9.1}$$

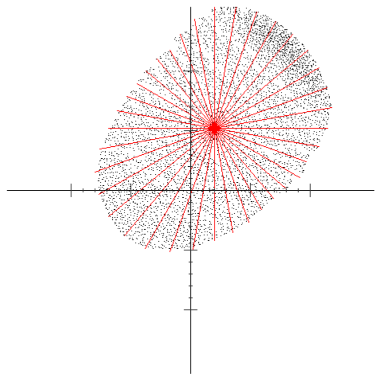
The dough center is intended as the geometric center of its shape. During the implementation, we discretized the configuration as a set of angle-equally sampled distances (see Fig. 9.5a) and vectorized them clock-wisely. The histogram of the vectorized configuration is shown in Fig. 9.5b.

When the dough shape is deformed through the action of the rolling pin, the centre position and the rotated angle of the dough are less important. Indeed, the relative angle between the rolling pin and the pizza dough would affect the deformation. To get a rotation-invariant configuration, the order of the histogram elements is rearranged so that the longest distance vector is the first one in the set (see Fig. 9.5c). This method has been frequently used in object recognition algorithms of image processing, *e.g.*, SIFT [180], SURF [20], MSER [193], FAST [262], and BRISK [167].

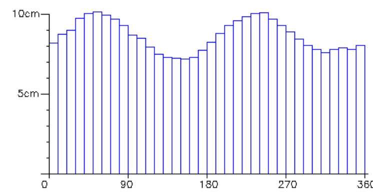
An extended configuration space $\mathcal{C}_Y$ is introduced to include the position of the centre and the angle between the longest distance vector (the first one in the rearranged histogram) and the x-axis of the frame $\mathcal{D}$

$$\mathcal{C}_Y : \left\{ \begin{bmatrix} \boldsymbol{\chi} \\ c_x \\ c_y \\ \theta \end{bmatrix} \in \mathbb{R}^5 : \boldsymbol{\chi} \in \mathcal{C}_x \right\}. \tag{9.2}$$

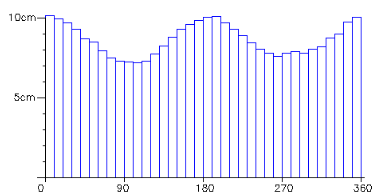
It can be seen in Fig. 9.5d how the reconstructed shape by the extended configuration $\mathcal{C}_Y$ is similar to the original shape. The original configuration space $\mathcal{C}_X$ of the pizza dough is a subspace of the extended configuration space $\mathcal{C}_Y$: their relation is drawn in Fig. 9.6. Through space projection, the extended



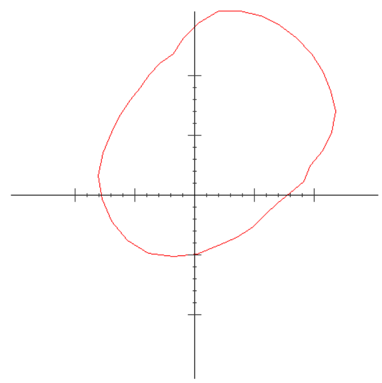
(a) Deformable object model projected onto the 2D space.



(b) Histogram of the distance from the deformable object's centre.



(c) Rearranged histogram for the longest one to be the first bin following by sequel bins.



(d) Reconstructed shape from the configuration $\mathbf{y}$.

Fig. 9.5: Configuration for a dough state. The red radial lines in (a) indicate equal-angle sampled distance from the centre of the pizza dough shape to the boundary. Blue line is the longest distance from the centre. The sampled distances are ordered clock-wisely.

configuration space $\mathcal{C}_Y$ can be submerged into the original configuration space $\mathcal{C}_X$, that is, $\mathcal{C}_Y/\mathcal{I} = \mathcal{C}_X$ where $\mathcal{I} = \{c_x = 0, c_y = 0, \theta = 0\}$.

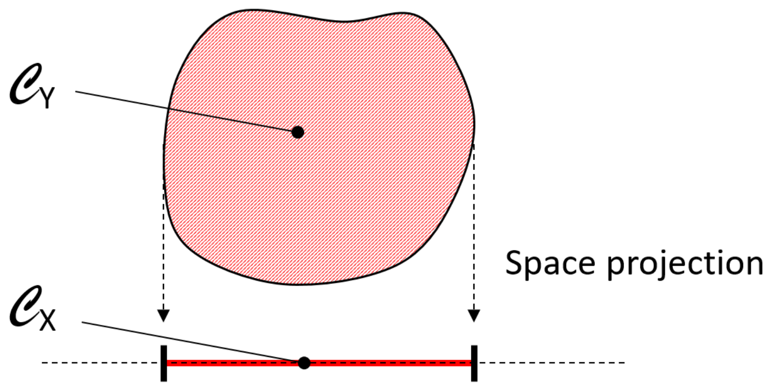


Fig. 9.6: The configuration space $\mathcal{C}_X$ and its extended configuration space $\mathcal{C}_Y$.

9.5 Construction of a planer for pizza dough stretching

Given a generic configuration space $\mathcal{C}$, a initial status $q_I \in \mathcal{C}$, and a final status $q_G \in \mathcal{C}$, the planner finds the sequence of intermediate status

$$q_I \to q_1 \to \cdots \to q_{n-1} \to q_G, \tag{9.3}$$

to reach q_G from q_I, and where $q_1, \cdots, q_{n-1} \in \mathcal{C}$. Each status is referred to as a configuration q in a given configuration space $\mathcal{C}$.

A way to find such a sequence of intermediate configurations is to associate to each of them a cost value relative to q_G. Let $V(q_i) > 0$ be a cost value related to the distance from $q_i \in \mathcal{C}$ to q_G, then a gradient descent method might be used to find the sequence of interemdiate configurations such that

$$V(q_I) \geq V(q_1) \geq \cdots \geq V(q_{n-1}) \geq V(q_G) = 0. \tag{9.4}$$

A movement from one configuration to another is called a transition $\mathcal{T}$. The transition occurs directly or indirectly through an action $\alpha \in \mathcal{A}$, where $\mathcal{A}$ is the set of admissible actions. Therefore, we can move from a configuration $q \in \mathcal{C}$ to a configuration $q' \in \mathcal{C}$ as

$$q \to q' = \mathcal{T}(q, \alpha). \tag{9.5}$$

The robotic system must execute the action α through a proper control design, taking into account possible errors during the execution.

Concerning the pizza stretching application, the configuration of the pizza dough is given by the sets $\mathcal{C}_X$ and $\mathcal{C}_Y$. The cost function $V(\cdot)$, the actions

$\alpha \in \mathcal{A}$ and the transitions $\mathcal{T}$ to achieve the sough task are explained in the following.

9.5.1 Cost value function

In this application, the cost value function $V(\cdot)$ is defined as how the shape of the pizza in the configuration q is different from the shape of the pizza in the desired configuration q_G. Hence, the more similar the shape of q is to q_G, the smaller the cost value V is.

A comparison between two 3D objects is typically made by checking the respective volumes. However, the comparison can be simplified in this application by projecting the two 3D volumes into a 2D plane. In fact, the target object has a shallow disk, and the error between the approximated cylinder shape and the original dough shape decreases as deformations of rolling pin actions are made.

The comparison is made by calculating the ratio of the occupied 2D area of the current dough shape over the 2D area of the target one while ignoring the space out of the target 2D area (see Fig. 9.7). The areas can be measured by counting the occupied grid cells after discretizing the 2D plate.

The designed cost value function for this application is

$$V(q) = 1 - \frac{\text{area}(q)}{\text{area}(q_G)}, \tag{9.6}$$

where $q, q_G \in \mathcal{C}_Y$ are the current and the target configurations of the pizza dough shape, respectively. The operator to compute the area is defined as

$$\text{area}(q) = \sum_{ij} \text{occ}(q, i, j),$$

where i and j are the indexes of discretized 2D space, and the occupancy function $\text{occ}(q, i, j)$ is defined as

$$\text{occ}(q, i, j) = \begin{cases} 1 & \text{if } (i, j) \in A(q) \cap A(q_G) \\ -\kappa & \text{if } (i, j) \in A(q) \cap \neg A(q_G) \\ 0 & \text{otherwise} \end{cases}, \tag{9.7}$$

where $A(q)$ is the part of the plate occupied by the dough in the configuration q, $\neg A(q)$ is the complement of $A(q)$, and $\kappa > 0$ is a penalty weight. In particular, $\kappa = 0$ means that we do not care if the current shape of the pizza dough is outside the target one, while $\kappa \gg 0$ indicates that we strongly penalize the case in which the current dough is outside the target one even partially. As it is built, the cost function assumes values in the range $[0, 1]$. When the cost function is zero, the current dough shape covers all the target

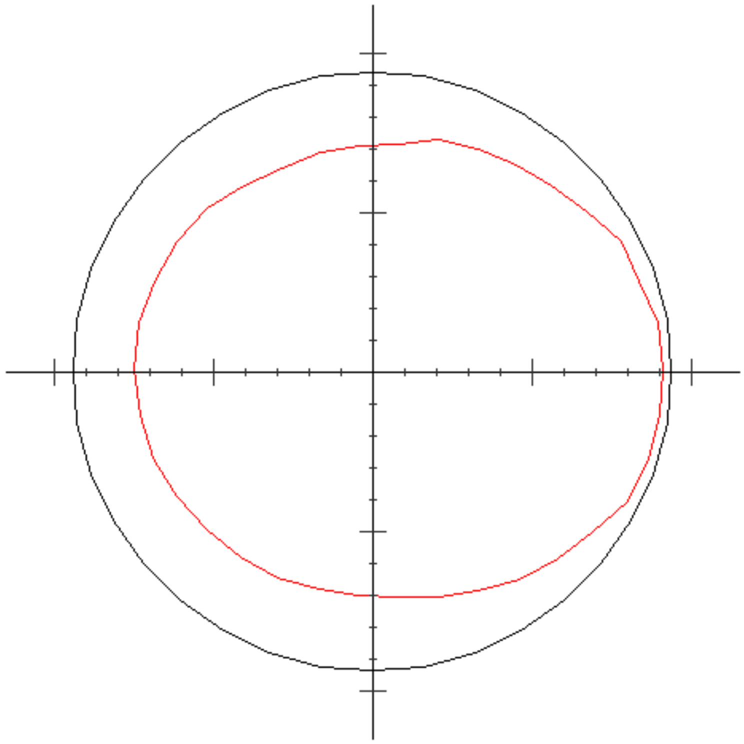


Fig. 9.7: An comparison between the current shape (red line) and the target shape (black line).

one. In practice, we verified that it is difficult and inefficient to cover the target shape area completely: we thus recommend saturating to zero the cost function when its value is under 0.1, corresponding to the fact that about the 90% of the target shape area is covered.

9.5.2 Actions for a deformable object

There are a lot of possible actions to modify a pizza dough employing a rolling pin. For simplicity, the available actions in this application are limited. First, slanted or downing movements are prohibited. Besides, an action cannot change its angle during the movement. Then, the distance between the rolling pin and the plate is constant during all the movement. Finally, the rolling pin's movement is always done in contact with the dough.

Following the above constraints, the action set is defined as follows

$$\mathcal{A} : \left\{ \begin{bmatrix} \delta \\ \phi \end{bmatrix} \in \mathbb{R}^2 \right\}, \tag{9.8}$$

where $\delta > 0$ is the height between the rolling pin and the table and $\phi \in [0°, \dots, 180°]$ is the angle between the longest axis of the pizza dough and the x-axis in $\mathcal{D}$. Because the rolling pin's forward and backward movements are not discriminated in this application, the provided interval for ϕ is enough. During simulations, the action set $\mathcal{A}$ consisting of a height δ and an angle ϕ was discretized. The number of heights and angles used in our simulations are provided in Section 9.8.

Notably, the deformations caused by the actions defined above are relative to the local configuration space $\mathcal{C}_X$ and not the global configuration space $\mathcal{C}_Y$. This means that the deformation of the pizza after an action $\boldsymbol{\alpha} \in \mathcal{A}$ is not affected by the global status $\mathbf{y} \in \mathcal{C}_Y$. However, it depends on $\boldsymbol{\chi} \in \mathcal{C}_X$ and the relative angle between the pizza dough and the rolling pin.

9.5.3 Transition originated from an action

A transition changes the status of the dough to another thanks to an action. The transition function $\mathbf{T} : \mathcal{C}_Y \times \mathcal{A} \to \mathcal{C}_Y$ can be defined as

$$\mathbf{y}' = T(\mathbf{y}, \boldsymbol{\alpha}), \tag{9.9}$$

where $\mathbf{y} \in \mathcal{C}_Y$ is the current state of the pizza dough and $\mathbf{y}' \in \mathcal{C}_Y$ is the one obtained after the execution of the action $\boldsymbol{\alpha} \in \mathcal{A}$.

As previously mentioned, the deformation of the pizza dough through an action is more related to the relative angle between the rolling pin and the longest axis rather than the absolute pose of the pizza dough. Therefore, the equation (9.9) can be rewritten as

$$\begin{aligned} \mathbf{y}' &= T(\mathbf{y}, \boldsymbol{\alpha}) \\ &= T(\begin{bmatrix} \mathbf{x}^T & \mathbf{0}_3^T \end{bmatrix}^T, \begin{bmatrix} \delta & \phi - \theta \end{bmatrix}^T) + \begin{bmatrix} \mathbf{0}_2^T & \mathbf{x}_0^T & \theta \end{bmatrix}^T . \end{aligned} \tag{9.10}$$

The new action $\boldsymbol{\alpha}' = \begin{bmatrix} \delta & \phi - \theta \end{bmatrix}^T$ is already included within $\mathcal{A}$; hence, there is no change in the size of the action set $\mathcal{A}$. On the other hand, the state $\begin{bmatrix} \mathbf{x}^T & \mathbf{0}_3^T \end{bmatrix}^T$ might be interpreted as adimension reduction by projection (Fig. 9.6). In this way, the domain of the transition function T can be drastically reduced.

In the next section, how to generate such transitions will be explained.

9.5.4 LUT method

Even though the deformation of an object due to an action is very well simulated through the SPH models (see Chapter 3.1), the problem is the computation time for each transition from one state to another state. The time depends on the resolution of the deformable object model and other conditions. In this chapter, the carried-out simulations took several minutes to hours for one transition. Considering that a planner requires at least several dozens, thousands, or much more transitions, this object model is not suitable for real-time or short-time planning.

LUT method is well-known for overcoming this computing-time problem. Possible transitions are calculated previously off-line, and then only the results are recorded into a transition database. Working online, the planner looks for a suitable transition in the database and uses it.

As explained in the previous subsection 9.5.3, a transition T is independent from the position and rotation angle of the current state $\mathbf{y} \in \mathcal{C}_Y$. Therefore, LUT contains only transitions from a state $\boldsymbol{\chi} \in \mathcal{C}_X$ at centre $\mathbf{x}_0 = \mathbf{0}_2$ and zero rotation angle, $\theta = 0$, with actions $\boldsymbol{\alpha} \in \mathcal{A}$.

A transition space is continuous, but it is infeasible to generate and store all possible transitions. Therefore, discretizing the transition space is necessary, which requires storing selected transitions into a database and a method to find a proper transition within it. The conventional method to find a proper transition is to look for the most similar one and use it [97], or to use an interpolating method to estimate unknown transitions from a given state $\boldsymbol{\chi} \in \mathcal{C}_X$ with the neighbour transitions. In the following subsections, we investigate how to find similar transitions into the database and interpolating them.

9.5.4.1 Similarity

In order to find similar transitions within the database, similarity measure functions between two states $\boldsymbol{\chi}_1, \boldsymbol{\chi}_2 \in \mathcal{C}_X$ and between two actions $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \in \mathcal{A}$ are needed. We use the diagonal Mahalanobis distance $\mathrm{sim}_X : \mathcal{C}_X \times \mathcal{C}_X \to \mathbb{R}^{\geq 0}$ as a function for the pizza dough configurations, whose definition is given below

$$\mathrm{sim}_X(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2) = \sqrt{\|\boldsymbol{\chi}_1 - \boldsymbol{\chi}_2\|_{\mathbf{S}_X^{-1}}}, \tag{9.11}$$

where $\mathbf{S}_X = \mathrm{diag}\left(\begin{bmatrix} 1 & \frac{1}{\beta} \end{bmatrix}\right) \in \mathbb{R}^{2\times 2}$, with $\beta > 1$. We use the function $\mathrm{sim}_A : \mathcal{A} \times \mathcal{A} \to \mathbb{R}^{\geq 0}$ for the actions as

$$\mathrm{sim}_A(\alpha_1, \alpha_2) = \sqrt{\|\boldsymbol{\alpha}_1 - \boldsymbol{\alpha}_2\|_{\mathbf{S}_A^{-1}}}, \tag{9.12}$$

where $\mathbf{S}_A = \text{diag}\left(\begin{bmatrix}\frac{1}{\gamma} & 1\end{bmatrix}\right) \in \mathbb{R}^{2\times 2}$, with $\gamma > 1$.

9.5.4.2 Interpolation

Like the basic idea of SPH formation in subsection 3.2.2, we find the neighbour transitions within a specific distance, and then we interpolate them by substituting the transition function T in (9.9) with a new transition function $\overline{T} : \mathcal{C}_Y \times \mathcal{A} \to \mathcal{C}_Y$ as

$$\mathbf{y}' = \overline{T}(\mathbf{y}, \boldsymbol{\alpha}) = \frac{\sum_i \overline{T}_a(\mathbf{y}_i, \boldsymbol{\alpha})\, W(\text{sim}_X(\bar{\boldsymbol{\chi}}, \bar{\boldsymbol{\chi}}_i), h_y)}{\sum_i W(\text{sim}_X(\bar{\boldsymbol{\chi}}, \bar{\boldsymbol{\chi}}_i), h_y)} \tag{9.13}$$

and

$$\overline{T}_a(\mathbf{y}, \boldsymbol{\alpha}) = \frac{\sum_j T(\mathbf{y}, \boldsymbol{\alpha}_j) W(\text{sim}_A(\boldsymbol{\alpha}, \boldsymbol{\alpha}_j), h_a)}{\sum_j W(\text{sim}_A(\boldsymbol{\alpha}, \boldsymbol{\alpha}_j), h_a)}, \tag{9.14}$$

where the kernel function $W : \mathbb{R}^{\geq 0} \times \mathbb{R}^{+} \to \mathbb{R}^{\geq 0}$ and the kernel ranges $h_y, h_a \in \mathbb{R}$ have been used. The denominator of the previous expressions is used for normalization purposes, similarly to the SPH formation in subsection 3.2.2. Notice that $\mathbf{y} = \begin{bmatrix}\bar{\boldsymbol{\chi}}^T & \mathbf{0}_3^T\end{bmatrix}^T$.

After execution of the action by the robot, if the deformed status is the same or similar to the expected one, then the following action will be executed sequentially. Otherwise, a new step of the planner is required.

9.6 Path generation for the rolling pin

One of the particular features in the proposed planning framework is the independent planning for the tool itself. In contrast, most of the other planning frameworks integrate the planning of the tool and the planning of the robot manipulator. There are benefits and drawbacks to this planning separation. The separation makes the robot manipulation planning manageable. At the same time, there is a need to treat the infeasible action sequences that are a problem when the robot manipulator generates its action sequences to follow the trajectory of the tool. In this application, the preference of simplicity of the manipulation planning makes us separate the planning for a tool from the planning for the robot manipulator.

From the previous sections, the actions to stretch the dough towards the desired shape are generated. A possible sequence of actions is something like: stretch forward first, then stretch 30°, and so on, all with specified heights from the plate. There are two issues for generating a proper action sequence for the rolling pin: the generation of the action itself (called primitive action) and the connection to the following action (called connecting action).

The primitive action is direct and intuitive. We must keep the desired height of the rolling pin from the plate and try to act at the specified angle by passing through the dough's center. The connecting action is instead artificial and specific to this application. First, two proper poses for the rolling pin are defined, namely the ready pose and the final pose. Then, a routine is repeated, starting with a connecting action from the initial pose to the ready pose until the generated action is completed. The routine steps are as follows: (a) doing a connecting action from the ready pose to the initial pose of a primitive action; (b) doing the primitive action; (c) doing a connecting action from the final pose of the primitive action to the defined final pose; (d) doing the connecting action from the defined final pose to the ready pose. This routine is capable of covering all generated actions from the previous dough planning.

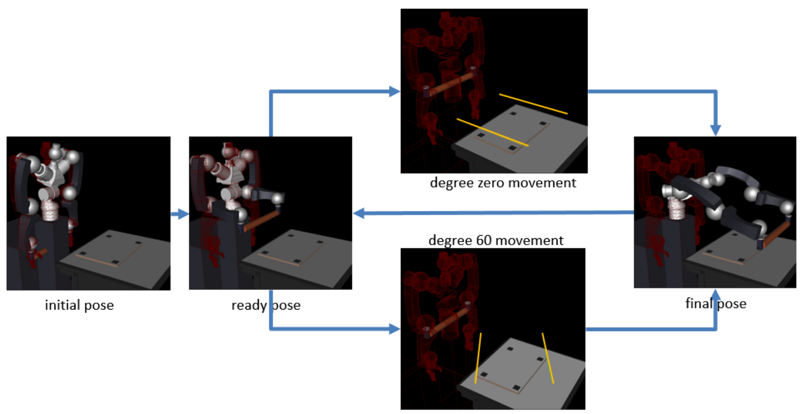


Fig. 9.8: Example of rolling-pin planning for two given actions on the dough.

There is an example of this process in Fig. 9.8 with two given actions from the previous dough planning, that is, a zero degree movement and a 60° movement. The generated action sequence consists of 1) a connecting action from the initial pose to the ready pose; 2) a connecting action from the ready pose to the initial pose of the zero-degree movement; 3) the primitive action related to the zero-degree movement; 4) a connecting action from the final pose of the zero-degree action to the final pose; 5) a connecting action from the final pose to the ready-pose; 6) a connecting action from the ready pose to the initial pose of the 60° movement; 7) the primitive action related to the 60° movement; 8) a connecting action from the final pose of the 60° movement to the final pose; 9) a connecting action from the final pose to the ready pose. The yellow lines indicate the trajectories of two holding points of the rolling pin while doing a primitive action.

The left figure in Fig 9.9 shows the trajectory of the steps above from 1 to 5. The right figure in Fig. 9.9 shows the trajectory of the steps above from

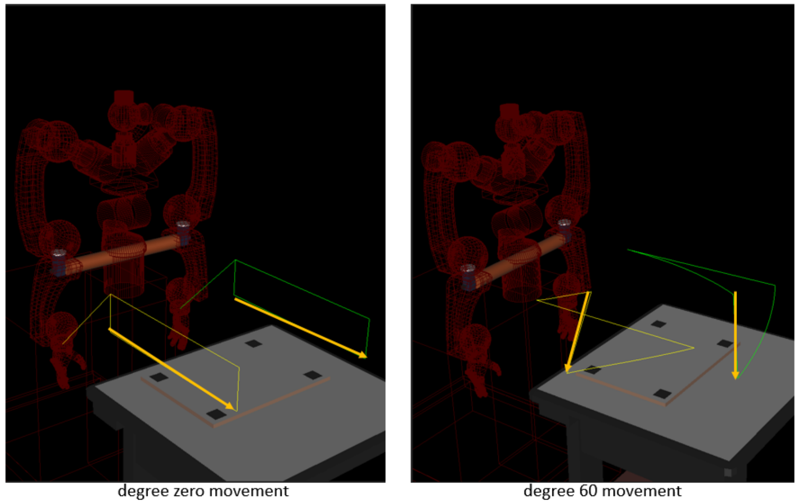


Fig. 9.9: Example trajectories of two holding points of a rolling pin for a given action on the dough. On the left, a zero-degree action. On the right, a 60° action.The yellow and red lines indicate the trajectories of the primitive and the connecting actions, respectively.

6 to 9, in which the yellow lines and the red lines are those of the primitive and connecting actions, respectively.

9.7 Inverse kinematics for the RoDyMan robot

RoDyMan is the employed robot platform that consists of a mobile base, a 2-DoFs torso, a 2-DoFs neck, and two 7-DoFs arms. This process step aims to make the arms grasp the rolling pin properly and solve the inverse kinematics problem to plan the joint movements.

Many inverse kinematics algorithms are well established in the literature, as those using the LM algorithm or the damping least-squares method [284]. The employed method follows the closed-loop inverse kinematics algorithm with redundancy management explained in [284]. Redundancy is exploited to avoid unnatural postures of the robot.

Fig. 9.10 shows the ready pose of RoDyMan before and after an action. Fig. 9.10b shows the unnatural behaviour that can be avoided through a proper redundancy management, as shows in Fig. 9.11. Details are omitted here for brevity since this part is well established in the literature and does not bring any new insight into the problem faced by this chapter.

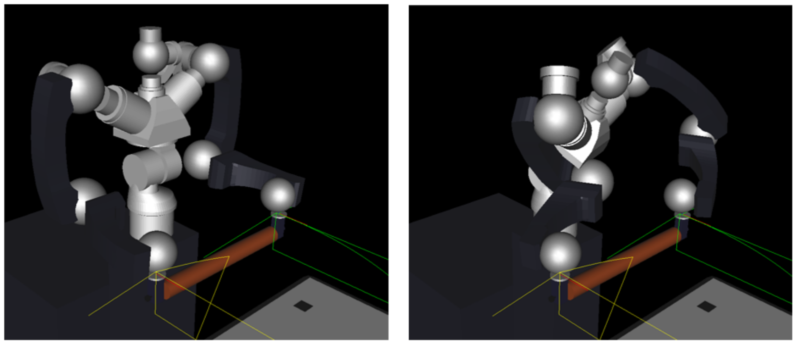


Fig. 9.10: Ready pose of RoDyMan before and after an action.

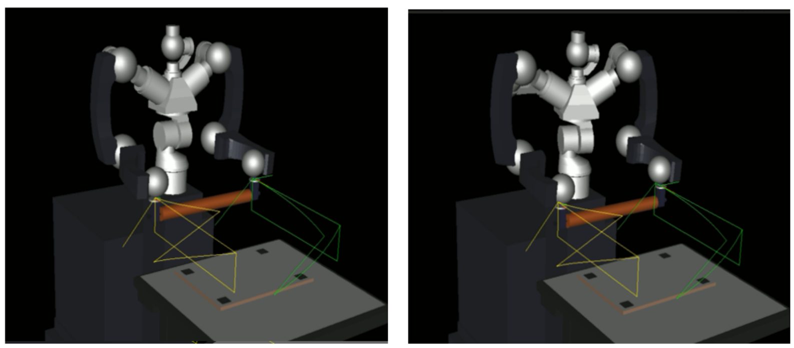


Fig. 9.11: Improved ready pose of RoDyMan before and after an action thanks to the redundancy management.

9.8 Simulations

For simulation purposes, the employed system consists of an Intel®Core™ i7-6500U CPU@2.50 GHz, Memory 8.0 GB with Windows®10 x64 operating system. We used the Microsoft Foundation Class (MFC) library of Microsoft®, the Qt library[1], Boost[2], Eigen[3], and OpenCV[4] for 2D graphic or OpenSceneGraph[5] for 3D graphic libraries based on C++11 programming language. Additionally, Houdini™ of SideFX®and Blender™ were used to re-

[1] https://www.qt.io
[2] https://www.boost.org
[3] http://eigen.tuxfamily.org
[4] https://opencv.org
[5] http://www.openscenegraph.org

construct the mesh from particles and for graphical rendering, respectively. 3D reconstruction is done with VisualSFM[6] and MeshLab[7].

9.8.1 Modelling of a deformable object

Even though a model for a deformable object like a pizza dough is designed in Chapter 3.1, the deformable object's properties like viscosity and elasticity vary case by case. Therefore, there is the need to tuning the model's parameters to fit the target object. Figure 9.12 shows how to measure a real deformable object. During preliminary experiments, a toy clay was used instead of pizza dough for convenience.

The process during the preliminary experiments has been carried out as follows. At first, the measure of the deformable object's surface and the reconstruction of its shape in 3D virtual space, at the initial and deformed statuses, have been carried out. Afterwards, an SPH-based object model is made following the reconstructed shape of the initial and deformed object. Finally, by applying the various actions on the SPH-based object model for the initial object, the best matching parameters for the model are found, which generates the SPH-based object model that is close to the reconstructed shape of the deformed object.

The reconstruction of a deformable object in the 3D virtual space at the initial status and the deformed status, respectively, are shown in Fig. 9.12 and Fig. 9.13. The pictures in Fig. 9.12 deals with a ball shape, representing the pizza dough before any deformation. The pictures in Fig. 9.12 show a disk shape after the deformation actuated by the rolling pin. To measure the shape of the object, a structure from motion method is used, which gathers some pictures (Fig. 9.12a and Fig. 9.13a) with various views and generates a 3D model (Fig. 9.12b and Fig. 9.13b). Usually, the generated 3D model is very rough and too complex. Hence, it needs post-processing to smooth the surface and remove the outliers. A simpler ball mesh-model (Fig. 9.12c) and cylinder mesh-model (Fig. 9.13c) are used to match the generated 3D model as close as possible, and the final reconstructed 3D model (Fig. 9.12d and Fig. 9.13d) are fixed, respectively.

The process of finding the best matching parameters for the deformable object model is shown in Fig. 9.14. Various deformed shapes are generated with various parameters: among them, we must find the best matching shape and its parameter. The matching process is done offline.

---

[6] http://ccwu.me/vsfm/

[7] http://www.meshlab.net/

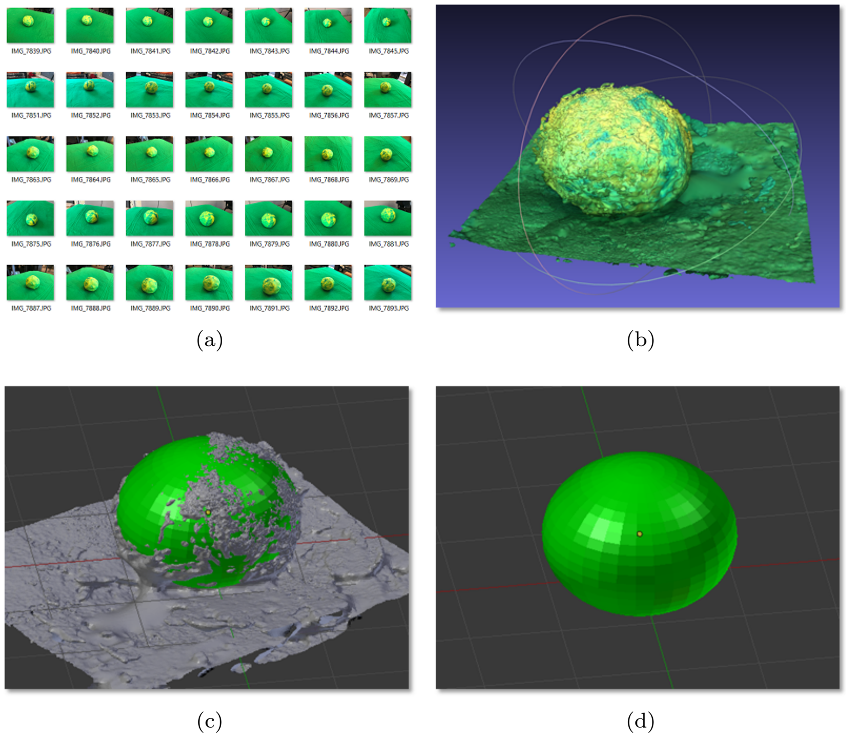
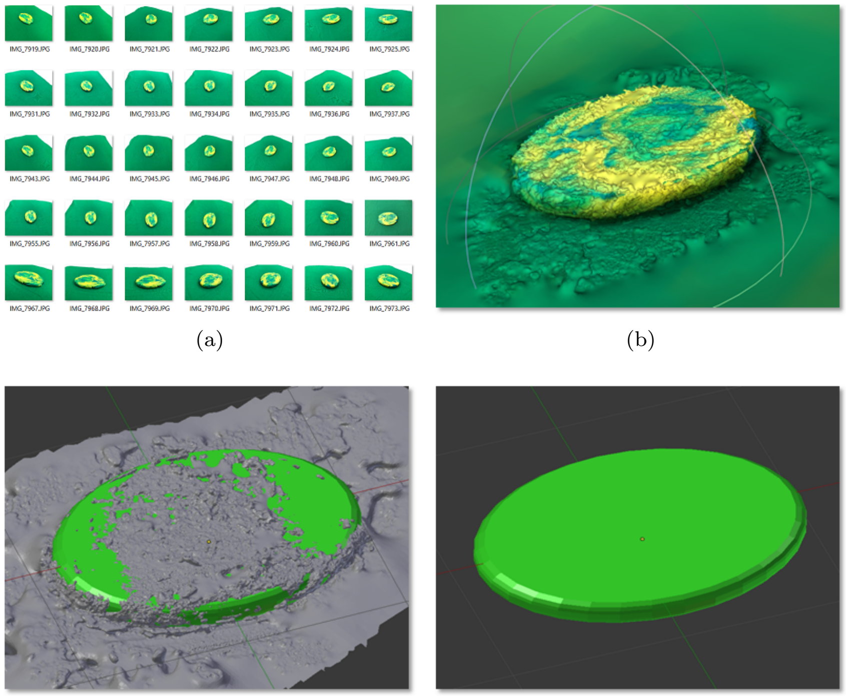


Fig. 9.12: Measuring of a real dough shape at initial status. (a) pictures with various views for real dough; (b) 3D reconstruction; (c) matching of the reconstructed shapes with a sphere; (d,h) the final matched shapes.

9.8.2 Pizza dough transition look-up-table

The transition LUT is used to speed up the planning algorithm, which is generated using various statuses of the pizza dough and various actions. However, it is not easy, and it needs much time to have experiments with real pizza dough. Therefore, using the deformable object model obtained before is more efficient than a real deformable object.

The previous section defines the model for a deformable object and its parameters obtained from real experiments. We had experiments with SPH particle radius of $1.25 \cdot 10^{-3}$ m, a pizza dough density of 1.276 kg/m^3, a solid density of $200 \cdot 10^3$ kg/m^3, an IISPH [136] for an incompressible fluid, and a viscosity coefficient of 250 kg/ms. Base on this model, the transition LUT is made by simulating the deformation from several configurations of the pizza dough and with various actions on the object. In particular, the action set $\mathcal{A}$ consists of seven heights, from $8 \cdot 10^{-3}$ m to $20 \cdot 10^{-3}$ m spanned each



(a) (b)

(c) (d)

Fig. 9.13: Measuring of a real dough shape at initial status. (a) pictures with various views for real dough; (b) 3D reconstruction; (c) matching of the reconstructed shapes with a disk; (d,h) the final matched shapes.

$2 \cdot 10^{-3}$ m, and eight directions, from zero degrees to 180°, spanned each 45° and considering both forward and backward motions.

A simulation is shown in Fig. 9.15 where the pizza dough is represented by the red particles and the rolling pin by the green ones. An example where the pizza dough is stretched is depicted in Fig. 9.16 and Fig. 9.17. The former shows the transition LUT for sequential statuses of the pizza dough with height and angular table, while the latter shows the occupancy of the pizza dough for each status.

9.9 Discussion and conclusion

This chapter explained a planning method to stretch a pizza dough with a rolling pin actuated by a robotic system. Base on the perception chapters,

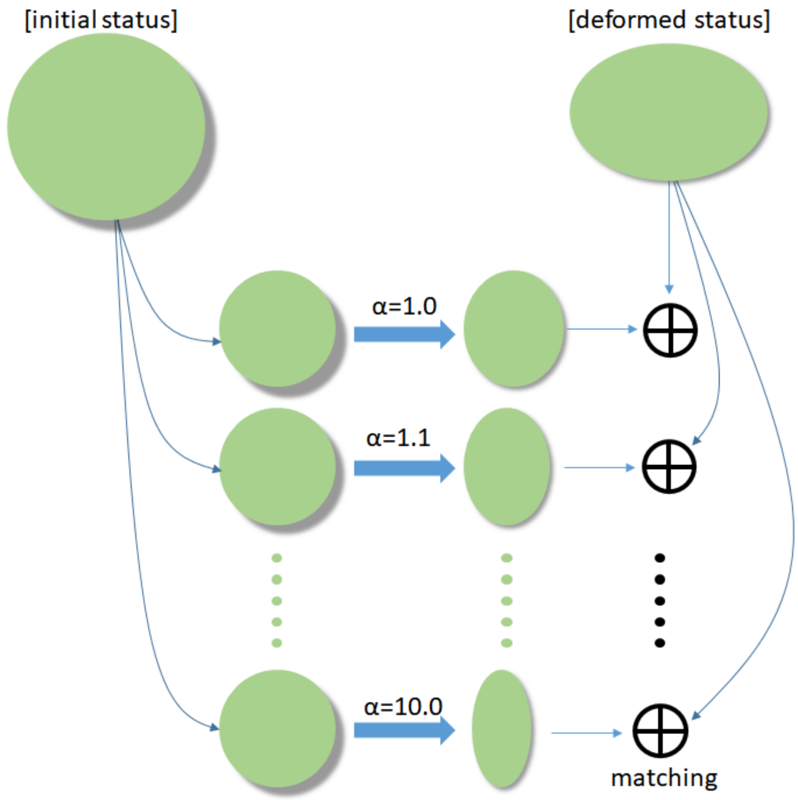


Fig. 9.14: Schematic diagram for finding the parameters of the deformable object model. α is just an example of a parameter for a SPH-based model: name and values are different depending on the model.

the deformable object is modelled, and the deformation information is used to plan the stretching actions on the pizza dough to reach the desired shape. An object recognition algorithm, a method to model deformable objects with high viscosity, the definition of the status of the pizza dough, the definition of the actions through the rolling pin, an inverse kinematics algorithm for the robot have been integrated to achieve the sought goal. Experiments were carried out to identify the parameters of the pizza dough. The stretching actions were planned offline thanks to a LUT database. Future experiments will definitely validate the proposed approach.

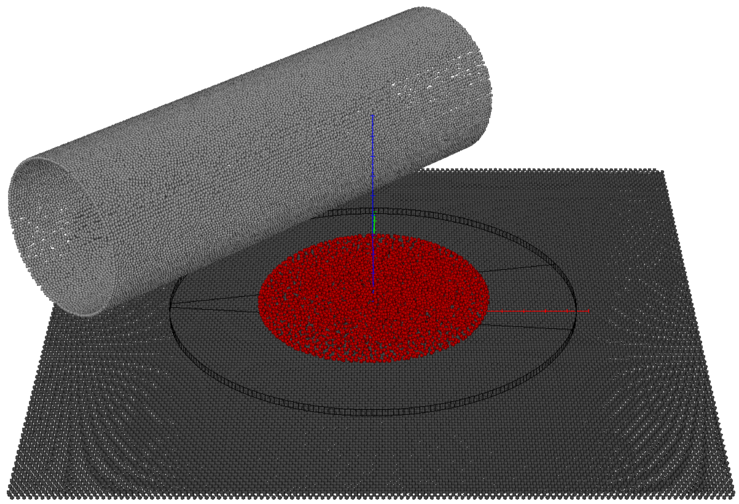
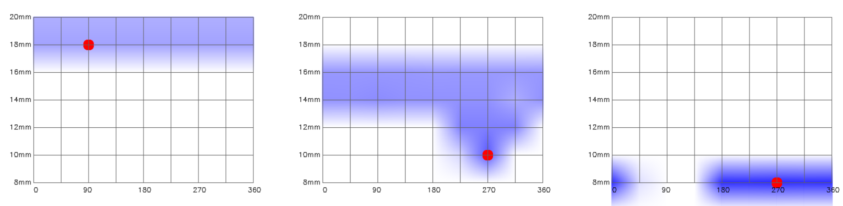
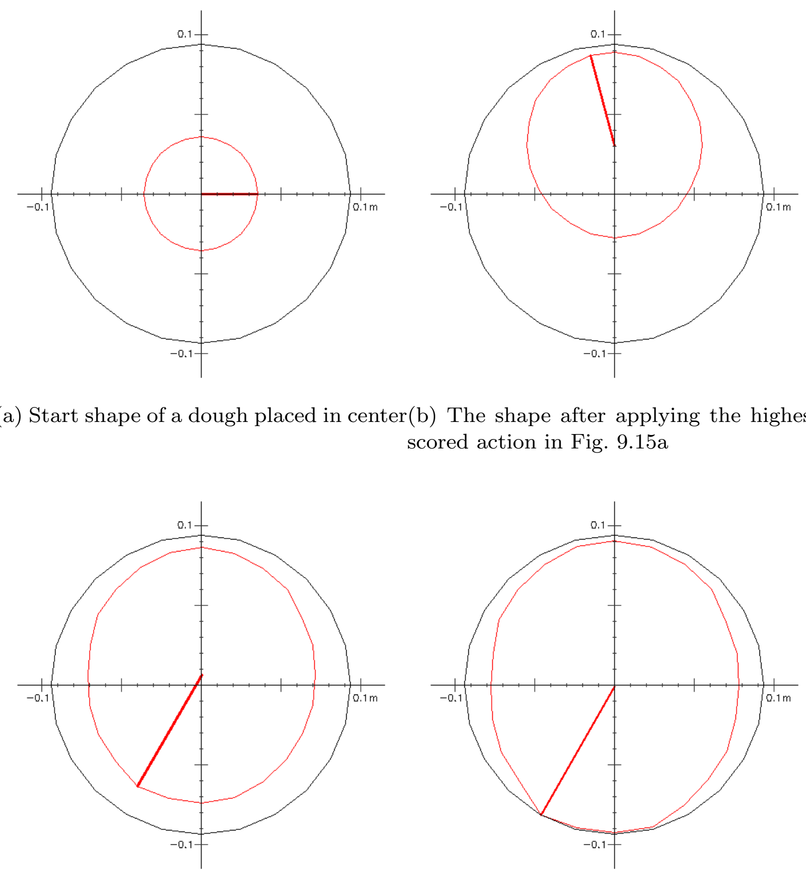


Fig. 9.15: SPH-based transition simulation for the pizza dough (red object) with a rolling pin (grey object).



(a) (b) (c)

Fig. 9.16: LUT method. The x-axis and the y-axis indicate the angle and the height of a rolling pin motion, respectively. The white colour means lower score, while the darker blue colour means higher score. Red ball is the highest score action.



(a) Start shape of a dough placed in center

(b) The shape after applying the highest scored action in Fig. 9.15a

(c) The shape after applying the highest scored action in Fig. 9.15b

(d) The shape after applying the highest scored action in Fig. 9.15c

Fig. 9.17: Shapes of the current dough (red) and its target (black). The red line indicates the angle of the first bin in the dough's state.

References

1. B. Allain, J.-S. Franco, and E. Boyer. An efficient volumetric framework for shape tracking. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, pages 268–276, 2015.
2. J. Allard, F. Faure, H. Courtecuisse, F. Falipou, C. Duriez, and P.G. Kry. Volume contact constraints at arbitrary resolution. In *ACM SIGGRAPH 2010 Papers*, pages 1–10, 2010.
3. M.R. Amjid, A. Shehzad, S. Hussain, M.A. Shabbir, M.R. Khan, and M. Shoaib. A comprehensive review on wheat flour dough rheology. *Pakistan Journal of Food Sciences*, 23:105–123, 2013.
4. T.L. Anderson. *Fracture Mechanics: Fundamentals and Applications*. CRC Press, 2005.
5. L.F.d.S. Andrade, M. Sandim, F. Petronetto, P. Pagliosa, and A. Paiva. SPH fluids for viscous jet buckling. In *27th SIBGRAPI Conference on Graphics, Patterns and Images*, pages 65–72, 2014.
6. L.F.d.S. Andrade, M. Sandim, F. Petronetto, P. Pagliosa, and A. Paiva. Particle-based fluids for viscous jet buckling. *Computers & Graphics*, 52:106–115, 2015.
7. W.W. Armstrong and M.W. Green. The dynamics of articulated rigid bodies for purposes of animation. *The visual computer*, 1(4):231–240, 1985.
8. P. Arpenti, F. Ruggiero, and V. Lippiello. A constructive methodology for the IDA-PBC of underactuated 2-DoF mechanical systems with explicit solution of PDEs. *International Journal of Control, Automation and Systems*, In press, 2021.
9. P. Asgari, P. Zarafshan, and S.A.A. Moosavian. Manipulation control of an armed ballbot with stabilizer. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 229(5):429–439, 2015.
10. I. Badami, J. Stückler, and S. Behnke. Depth-enhanced Hough Forests for object-class detection and continuous pose estimation. In *Workshop on Semantic Perception, Mapping and Exploration*, 2013.
11. J. Baillieul, A.M. Bloch, P. Crouch, and J. Marsden. *Nonholonomic Mechanics and Control*. Interdisciplinary Applied Mathematics. Springer New York, 2008.
12. L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys. Motion capture of hands in action using discriminative salient points. In *European Conference on Computer Vision*, pages 640–653, 2012.
13. A.H. Barr. Superquadrics and angle-preserving transformations. *IEEE Computer graphics and Applications*, 1(1):11–23, 1981.
14. A.H. Barr. Global and local deformations of solid primitives. *Readings in Computer Vision*, pages 661–670, 1987.

15. H. Barreiro, I. García-Fernández, I. Alduán, and M.A. Otaduy. Conformation constraints for efficient viscoelastic fluid simulation. *ACM Transactions on Graphics*, 36(6):221:1–221:11, 2017.
16. A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.
17. A. Bartoli and A. Zisserman. Direct estimation of non-rigid registrations. In *British Machine Vision Conference*, pages 899–908, 2004.
18. R. Barzel and A.H. Barr. A modeling system based on dynamic constraints. In *5th Annual Conference on Computer Graphics and Interactive Techniques*, pages 179–188, 1988.
19. C. Batty and R. Bridson. Accurate viscous free surfaces for buckling, coiling, and rotating liquids. In *2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 219–228, 2008.
20. H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In *Computer Vision and Image Understanding*, pages 346–359, 2008.
21. M. Becker and M. Teschner. Weakly compressible SPH for free surface flows. In *2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 209–217, 2007.
22. J. Bender and D. Koschier. Divergence-free smoothed particle hydrodynamics. In *14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 147–155, 2015.
23. J. Bender and D. Koschier. Divergence-free SPH for incompressible and viscous fluids. *IEEE Transactions on Visualization and Computer Graphics*, 23:1193–1206, 2017.
24. J. Bender, M. Müller, M.A. Otaduy, M. Teschner, and M. Macklin. A survey on position-based simulation methods in computer graphics. *Computer Graphics Forum*, 33(6):228–251, 2014.
25. D. Berenson. Manipulation of deformable objects without modeling and simulating deformation. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4525–4532, 2013.
26. M. Berger, A. Tagliasacchi, L.M. Seversky, P. Alliez, J.A. Levine, A. Sharf, and C.T. Silva. State of the art in surface reconstruction from point clouds. In S. Lefebvre and M. Spagnuolo, editors, *Eurographics 2014 - State of the Art Reports*. The Eurographics Association, 2014.
27. C.R. Beverly and R.I. Tanner. Numerical analysis of three-dimensional Bingham plastic flow. *Journal of Non-Newtonian Fluid Mechanics*, 42(1):85–115, 1992.
28. A. Bey, R. Chaine, R. Marc, G. Thibault, and S. Akkouche. Reconstruction of consistent 3D CAD models from point cloud data using a priori CAD models. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3812:289–294, 2011.
29. G. Birkhoff, H. Burchard, and D. Thomas. *Nonlinear Interpolation by Splines, Pseudosplines and Elastica*. Research Laboratories, General Motors Corporation, 1965.
30. S. Bittanti, A.J. Laub, and J.C. Willems. *The Riccati Equation*. Springer-Verlag, first edition, 1991.
31. A. Bloch, J. Baillieul, P. Crouch, J.E. Marsden, D. Zenkov, P.S. Krishnaprasad, and R.M. Murray. *Nonholonomic Mechanics and Control*, volume 24. Springer, 2003.
32. A.M. Bloch and P.E. Crouch. Nonholonomic control systems on Riemannian manifolds. *SIAM Journal on Control and Optimization*, 33(1):126–148, 1995.
33. K.-F. Bohringer, B.R. Donald, and N.C. MacDonald. Programmable force fields for distributed manipulation, with applications to MEMS actuator arrays and vibratory parts feeders. *The International Journal of Robotics Research*, 18(2):168–200, 1999.
34. K.F. Bohringer, B.R. Donald, L.E. Kavraki, and F. Lamiraux. Part orientation with one or two stable equilibria using programmable force fields. *IEEE Transactions on Robotics and Automation*, 16(2):157–170, 2000.

35. I. Boiko. *Discontinuous Control Systems: Frequency-domain Analysis and Design*. Springer Science & Business Media, 2008.
36. C. Bollain, A. Angioloni, and C. Collar. Relationships between dough and bread viscoelastic properties in enzyme supplemented wheat samples. *Journal of Food Engineering*, 77:665–671, 2006.
37. M. Bollini, S. Tellex, T. Thompson, N. Roy, and D. Rus. Interpreting and executing recipes with a cooking robot. In *Experimental Robotics*, pages 481–495. Springer, 2013.
38. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.
39. J.U. Brackbill and H.M. Ruppel. FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *Journal of Computational physics*, 65(2):314–343, 1986.
40. R.W. Brockett. Asymptotic stability and feedback stabilization. In *Differential Geometric Control Theory*, pages 181–191. Birkhauser, 1983.
41. J. Brown, J.-C. Latombe, and K. Montgomery. Real-time knot-tying simulation. *The Visual Computer*, 20(2-3):165–179, 2004.
42. M.A. Brubaker, L. Sigal, and D.J. Fleet. Estimating contact dynamics. In *2009 IEEE International Conference on Computer Vision*, pages 2389–2396, 2009.
43. G. Brunnett. A new characterization of plane elastica. In *Mathematical methods in computer aided geometric design II*, pages 43–56. Elsevier, 1992.
44. G. Buck and J. Simon. Knots as dynamical systems. *Topology and its Applications*, 51(3):229–246, 1993.
45. S. Burion, F. Conti, A. Petrovskaya, C. Baur, and O. Khatib. Identifying physical properties of deformable objects by using particle filters. In *2008 IEEE International Conference on Robotics and Automation*, pages 1112–1117, 2008.
46. G.M. Campbell, C.D. Rielly, P.J. Fryer, and P.A. Sadd. Measurement and interpretation of dough densities. *Cereal Chemistry*, 70(5):517–521, 1993.
47. J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.
48. M. Carlson, P.J. Mucha, R.B. Van Horn III, and G. Turk. Melting and flowing. In *2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 167–174, 2002.
49. M. Cefalo, L. Lanari, and G. Oriolo. Energy-based control of the butterfly robot. In *International IFAC Symposium on Robot Control*, pages 1–6, 2006.
50. Y. Chang, K. Bao, J. Zhu, and E. Wu. High viscosity fluid simulation using particle-based method. In *2011 IEEE International Symposium on VR Innovation*, pages 199–205, 2011.
51. M. Charalambides, L. Wanigasooriya, S.M. Goh, and J.G. Williams. Viscoelasticity of bread dough. In *2005 SEM Annual Conference and Exposition on Experimental and Applied Mechanics*, pages 879–888, 2005.
52. C.-T. Chen. *Linear System Theory and Design*. Oxford University Press, third edition, 1999.
53. Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image and vision computing*, 10(3):145–155, 1992.
54. Y. Chen, Q.-H. Zhu, A. Kaufman, and S. Muraki. Physically-based animation of volumetric objects. In *Computer Animation '98*, pages 154–160, 1998.
55. G.S. Chirikjian and J.W. Burdick. Kinematically optimal hyper-redundant manipulator configurations. *IEEE transactions on Robotics and Automation*, 11(6):794–806, 1995.
56. A.E. Chorin. *A Mathematical Introduction to Fluid Mechanics*. Springer-Verlag, 1992.
57. T.J. Chung. *Computational Fluid Dynamics*. Cambridge University Press, 2010.

58. P. Cigliano, V. Lippiello, F. Ruggiero, and B. Siciliano. Robotic ball catching with an eye-in-hand single-camera system. *IEEE Transactions on Control Systems Technology*, 23(5):1657–1671, 2015.
59. S. Clavet, P. Beaudoin, and P. Poulin. Particle-based viscoelastic fluid simulation. In *2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 219–228, 2005.
60. L.D. Cohen and I. Cohen. Deformable models for 3-D medical images using finite elements and balloons. In *1992 IEEE Conference on Computer Vision and Pattern Recognition*, pages 592–598, 1992.
61. F. Colin, R. Egli, and F.Y. Lin. Computing a null divergence velocity field using smoothed particle hydrodynamics. *Journal of Computational Physics*, 217(2):680–692, 2006.
62. R.D. Cook. *Finite Element Modeling for Stress Analysis*. Wiley, 1994.
63. J. Cornelis, M. Ihmsen, A. Peer, and M. Teschner. IISPH-FLIP for incompressible fluids. *Computer Graphics Forum*, 33(2):255–262, 2014.
64. J.-M. Coron. On the stabilization in finite time of locally controllable systems by means of continuous time-varying feedback law. *SIAM Journal on Control and Optimization*, 33(3):804–833, 1995.
65. S. Cotin, H. Delingette, and N. Ayache. Efficient linear elastic models of soft tissues for real-time surgery simulation. Technical Report RR-3510, INRIA, 1998.
66. S. Cotin, H. Delingette, and N. Ayache. Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):62–73, 1999.
67. M.G. Coutinho and P.M. Will. A general theory for positioning and orienting 2D polygonal or curved parts using intelligent motion surfaces. In *1998 IEEE International Conference on Robotics and Automation*, pages 856–862, 1998.
68. A.-M. Cretu, P. Payeur, and E.M. Petriu. Neural network mapping and clustering of elastic behavior from tactile and range imaging for virtualized reality applications. *IEEE Transactions on Instrumentation and Measurement*, 57(9):1918–1928, 2008.
69. M.M. Cross. Rheology of non-Newtonian fluids: A new flow equation for pseudoplastic systems. *Journal of Colloid Science*, 20:417–437, 1965.
70. P. Crouch and F.S. Leite. The dynamic interpolation problem: On Riemannian manifolds, Lie groups, and symmetric spaces. *Journal of Dynamical and control systems*, 1(2):177–202, 1995.
71. L. Cui and J. S. Dai. A coordinate-free approach to instantaneous kinematics of two rigid objects with rolling contact and its implications for trajectory planning. In *2009 IEEE International Conference on Robotics and Automation*, pages 612–617, Kobe, J, 2009.
72. S.J. Cummins and M. Rudman. An SPH projection method. *Journal of Computational Physics*, 152(2):584–607, 1999.
73. R.S. Dahiya, G. Metta, M. Valle, and G. Sandini. Tactile sensing—from humans to humanoids. *IEEE Transactions on Robotics*, 26(1):1–20, 2010.
74. H. Date, M. Sampei, M. Ishikawa, and M. Koga. Simultaneous control of position and orientation for ball plate manipulation problem based on time state control form. *IEEE Transactions on Robotics and Automation*, 20(3):465–480, 2004.
75. C. de Boor. *A Practical Guide to Splines*. Springer-Verlag, 1978.
76. A. De Luca and G. Oriolo. Modelling and control of nonholonomic mechanical systems. In *Kinematics and Dynamics of Multi-body Systems*, pages 277–342. Springer, 1995.
77. G. Debunne, M. Desbrun, M.-P. Cani, and A.H. Barr. Dynamic real-time deformations using space & time adaptive sampling. In *28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 31–36, 2001.
78. H. Delingette, G. Subsol, S. Cotin, and J. Pignon. Craniofacial surgery simulation testbed. In *Visualization in Biomedical Computing 1994*, pages 607–618, 1994.

79. M. Desbrun and M.-P. Gascuel. Smoothed particles: A new paradigm for animating highly deformable bodies. In *Eurographics Workshop on Computer Animation and Simulation '96*, pages 61–76, 1996.
80. M.P. do Carmo. *Riemannian Geometry*. Mathematics (Boston, Mass.). Birkhäuser, 1992.
81. A. Donaire, R. Mehra, R. Ortega, S. Satpute, J. G. Romero, F. Kazi, and N.M. Sing. Shaping the energy of mechanical systems without solving partial differential equations. *IEEE Transactions on Automatic Control*, 61(4):1051–1056, 2016.
82. A. Donaire, J. G. Romero, R. Ortega, B. Siciliano, and M. Crespo. Robust IDA-PBC for underactuated mechanical systems subject to matched disturbances. *International Journal of Robust and Nonlinear Control*, 27(6):1000–1016, 2017.
83. A. Donaire, F. Ruggiero, L. R. Buonocore, V. Lippiello, and B. Siciliano. Passivity-based control for a rolling-balancing system: The nonprehensile disk-on-disk. *IEEE Transactions on Control System Technology*, 25(6):2135–2142, 2017.
84. H.-S. Dou and B. Khoo. Investigation of turbulent transition in plane Couette flows using energy gradient method. *Advances in Applied Mathematics and Mechanics*, 3:165–180, 2005.
85. C. Elbrechter, R. Haschke, and H. Ritter. Bi-manual robotic paper manipulation based on real-time marker tracking and physical modelling. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1427–1432, 2011.
86. L.D. Elsgolc. *Calculus of Variations*. Courier Corporation, 2012.
87. J. Engmann, M.C. Peck, and D.I. Wilson. An experimental and theoretical investigation of bread dough sheeting. *Food and Bioproducts Processing*, 83(3):175–184, 2005.
88. D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell. A hybrid particle level set method for improved interface capturing. *Journal of Computational Physics*, 183(1):83–116, 2002.
89. O. Etzmuß, M. Keckeisen, and W. Straßer. A fast finite element solution for cloth modelling. In *11th Pacific Conference on Computer Graphics and Applications*, pages 244–251, 2003.
90. R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of Numerical Analysis*, 7:713–1018, 2000.
91. J.M. Faubion and R.C. Hoseney. The viscoelastic properties of wheat flour doughs. In *Dough Rheology and Baked Product Texture*, pages 29–66. Springer, 1990.
92. F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin. Sofa: A multi-model framework for interactive physical simulation. *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, pages 283–321, 2012.
93. C.A. Felippa and B. Haugen. A unified formulation of small-strain corotational finite elements: I. Theory. *Computer Methods in Applied Mechanics and Engineering*, 194(21):2285–2335, 2005.
94. C.R. Feynman. *Modeling the Appearance of cloth*. PhD thesis, Massachusetts Institute of Technology, 1986.
95. F. Ficuciello, L. Villani, and B. Siciliano. Variable impedance control of redundant manipulators for intuitive human–robot physical interaction. *IEEE Transactions on Robotics*, 31(4):850–863, 2015.
96. S.D. Fisher and J.W. Jerome. Stable and unstable elastica equilibrium and the problem of minimum curvature. In *Minimum Norm Extremals in Function Spaces*, pages 90–106. Springer, 1975.
97. B. Frank, M. Becker, C. Stachniss, W. Burgard, and M. Teschner. Efficient path planning for mobile robots in environments with deformable objects. In *2008 IEEE International Conference on Robotics and Automation*, pages 3737–3742, 2008.
98. B. Frank, R. Schmedding, C. Stachniss, M. Teschner, and W. Burgard. Learning the elasticity parameters of deformable objects with a manipulation robot. In *2010*

IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 1877–1883, 2010.
99. B. Frank, C. Stachniss, N. Abdo, and W. Burgard. Efficient motion planning for manipulation robots in environments with deformable objects. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2180–2185, 2011.
100. P.U. Frei. An intelligent vibratory conveyor for the individual object transportation in two dimensions. In *2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1832–1837, 2002.
101. S. Fukuhara. Energy of a knot. In *A fête of topology*, pages 443–451. Elsevier, 1988.
102. Y.-C. Fung. *Foundations of Solid Mechanics (Book on deformation and motion of elastic and plastic solids including variational calculus and tensor analysis)*. Prentice-Hall, 1965.
103. R. Gahleitner. Ball on ball: Modeling and control of a novel experiment set-up. *IFAC-PapersOnLine*, 48(1):796–801, 2015.
104. M. Gardner. *Mathematical Carnival: A New Round-up Tantalizers and Puzzles from Scientific American*. Vintage Books, 1977.
105. A.V. Gelder. Approximate simulation of elastic membranes by triangulated spring meshes. *Journal of Graphics Tools*, 3(2):21–41, 1998.
106. M.C. Gemici and A. Saxena. Learning haptic representation for manipulating deformable food objects. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 638–645, 2014.
107. S.F.F. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical report, Citeseer, 1997.
108. R.A. Gingold and J.J. Monaghan. Smoothed particle hydrodynamics-theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389, 1977.
109. L. Glondu, M. Marchal, and G. Dumont. Real-time simulation of brittle fracture using modal analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(2):201–209, 2013.
110. T.G. Goktekin, A.W. Bargteil, and J.F. O'Brien. A method for animating viscoelastic fluids. *ACM Transactions on Graphics*, 23(3):463–468, 2004.
111. M. Gomez-Gesteira, B.D. Rogers, R.A. Dalrymple, and A. Crespo. State-of-the-art of classical SPH for free-surface flows. *Journal of Hydraulic Research*, 48:6–27, 2010.
112. R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Prentice-Hall, 2006.
113. M. Grégoire and E. Schömer. Interactive simulation of one-dimensional flexible parts. *Computer-Aided Design*, 39(8):694–707, 2007.
114. T.R. Grieve, J.M. Hollerbach, and S.A. Mascaro. Force prediction by fingernail imaging using active appearance models. In *2013 World Haptics Conference*, pages 181–186, 2013.
115. Y.-L. Gu and Y. Xu. A normal form augmentation approach to adaptive control of space robot systems. *Dynamics and Control*, 5(3):275–294, 1995.
116. K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu. Real-time geometry, albedo, and motion reconstruction using a single RGB-D camera. *ACM Transactions on Graphics*, 36(4):32:1–32:13, 2017.
117. A. Gutierrez-Giles, F. Ruggiero, V. Lippiello, and B. Siciliano. Modelling and control of a robotic hula-hoop system without velocity measurements. In *20th World IFAC Congress*, pages 9808–9814, 2017.
118. A. Gutierrez-Giles, F. Ruggiero, V. Lippiello, and B. Siciliano. Nonprehensile manipulation of an underactuated mechanical system with second-order nonholonomic constraints: The robotic hula-hoop. *IEEE Robotics and Automation Letters*, 3(2):1136–1143, 2018.
119. A. Gutierrez-Giles, F. Ruggiero, V. Lippiello, and B. Siciliano. Closed-loop control of a nonprehensile manipulation system inspired by a pizza-peel mechanism. In *European Control Conference*, pages 1580–1585, Naples, I, 2019.

120. V. Hagenmeyer, S. Streif, and M. Zeitz. Flatness-based feedforward and feedback linearisation of the ball & plate lab experiment. In *IFAC Symposium on Nonlinear Control Systems*, 2004.
121. H. Hamer, K. Schindler, E. Koller-Meier, and L. Van Gool. Tracking a hand manipulating an object. In *2009 IEEE International Conference OnComputer Vision*, pages 1475–1482, 2009.
122. N. Haouchine, J. Dequidt, M.-O. Berger, and S. Cotin. Monocular 3D reconstruction and augmentation of elastic surfaces with self-occlusion handling. *IEEE Transactions on Visualization and Computer Graphics*, 21(12):1363–1376, 2015.
123. N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, and S. Cotin. Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery. In *2013 IEEE International Symposium on Mixed and Augmented Reality*, pages 199–208, 2013.
124. F.H. Harlow. The particle-in-cell method for numerical solution of problems in fluid dynamics. Technical report, Los Alamos Scientific Lab., N. Mex., 1962.
125. R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.
126. J. Hauser, S. Sastry, and P. Kokotovic. Nonlinear control via approximate input-output linearization: The ball and beam example. *IEEE Transactions on Automatic Control*, 37(3):392–398, 1992.
127. J. Hauser, S. Sastry, and P. Kokotovic. Nonlinear control via approximate input-output linearization: The ball and beam example. *IEEE Transactions on Automatic Control*, 37(3):392–398, 1992.
128. L. Hertig, D. Schindler, M. Bloesch, C.D. Remy, and R. Siegwart. Unified state estimation for a ballbot. In *2013 IEEE International Conference on Robotics and Automation*, pages 2471–2476, 2013.
129. M. Higashimori, Y. Omoto, and M. Kaneko. Non-grasp manipulation of deformable object by using pizza handling mechanism. In *2009 IEEE International Conference on Robotics and Automation*, pages 120–125, 2009.
130. M. Higashimori, K. Utsumi, and M. Kaneko. Dexterous hyper plate inspired by pizza manipulation. In *2008 IEEE International Conference on Robotics and Automation*, pages 399–406, 2008.
131. M. Higashimori, K. Utsumi, Y. Omoto, and M. Kaneko. Dynamic manipulation inspired by the handling of a pizza peel. *IEEE Transactions on Robotics*, 25(4):829–838, 2009.
132. M. Higashimori, K. Yoshimoto, and M. Kaneko. Active shaping of an unknown rheological object based on deformation decomposition into elasticity and plasticity. In *2010 IEEE International Conference on Robotics and Automation*, pages 5120–5126, 2010.
133. B.K.P. Horn. The curve of least energy. *ACM Transactions on Mathematical Software (TOMS)*, 9(4):441–460, 1983.
134. D. Hristu, N. Ferrier, and R.W. Brockett. The performance of a deformable-membrane tactile sensor: Basic results on geometrically-defined tasks. In *2000 IEEE International Conference on Robotics and Automation*, pages 508–513, 2000.
135. D. Hutchinson, M. Preston, and T. Hewitt. Adaptive refinement for mass/spring simulations. In *Computer Animation and Simulation'96*, pages 31–45. Springer, 1996.
136. M. Ihmsen, J. Cornelis, B. Solenthaler, C. Horvath, and M. Teschner. Implicit incompressible SPH. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):426–435, 2014.
137. M. Ihmsen, J. Orthmann, B. Solenthaler, A. Kolb, and M. Teschner. SPH fluids in computer graphics. *EUROGRAPHICS 2014 — State of The Art Report*, pages 21–42, 2014.
138. A.N. Inal, O. Morgul, and U. Saranli. A 3D dynamic model of a spherical wheeled self-balancing robot. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5381–5386, 2012.

139. H.M. Irvine. *Cable Structures*, volume 17. MIT Press, 1981.
140. G. Irving, J. Teran, and R. Fedkiw. Invertible finite elements for robust simulation of large deformation. In *2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 131–140, 2004.
141. P. Jiménez, F. Thomas, and C. Torras. 3D collision detection: A survey. *Computers & Graphics*, 25(2):269–285, 2001.
142. A. Jordt and R. Koch. Direct model-based tracking of 3D object deformations in depth and color video. *International Journal of Computer Vision*, 102:1–17, 2013.
143. E. Jou and . Han. Minimal-energy splines with various end constraints. In *Curve and surface design*, pages 23–40. SIAM, 1992.
144. E. Jou and W. Han. Elastica and minimal–energy splines. In *Curves and surfaces*, pages 247–250. Elsevier, 1991.
145. O. Kähler, V.A. Prisacariu, and D.W. Murray. Real-time large-scale dense 3d reconstruction with loop closure. In *European Conference on Computer Vision*, pages 500–516, 2016.
146. M. Kallay. Plane curves of minimal energy. *ACM Transactions on Mathematical Software (TOMS)*, 12(3):219–222, 1986.
147. M. Kallay. Method to approximate the space curve of least energy and prescribed length. *Computer-Aided Design*, 19(2):73–76, 1987.
148. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.
149. H. Kato and M. Billinghurst. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *2nd IEEE and ACM International Workshop on Augmented Reality*, pages 85–94, 1999.
150. E. Keeve, S. Girod, and B. Girod. Craniofacial surgery simulation. In *Visualization in Biomedical Computing 1996*, pages 541–546. Springer, 1996.
151. M. Kelager. Lagrangian fluid dynamics using smoothed particle hydrodynamics. Master's thesis, University of Copenhagen, 2006.
152. H.K. Khalil. *Nonlinear Systems*. Prentice Hall, 2002.
153. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Autonomous robot vehicles*, pages 396–404. Springer, 1986.
154. B.S. Khatkar, A.E. Bell, and J.D. Schofield. The dynamic rheological properties of glutens and gluten sub-fractions from wheats of good and poor bread making quality. *Journal of Cereal Science*, 22(1):29–44, 1995.
155. K. Kim, V. Lepetit, and W. Woo. Keyframe-based modeling and tracking of multiple 3d objects. In *2010 9th IEEE International Symposium on Mixed and Augmented Reality*, pages 193–198, 2010.
156. P. Kormushev, S. Calinon, and D.G. Caldwell. Robot motor skill coordination with EM-based reinforcement learning. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3232–3237, 2010.
157. M. Kumaga and T. Ochiai. Development of a robot balanced on a ball: Application of passive motion to transport. In *2009 IEEE International Conference on Robotics and Automation*, pages 4106–4111, 2009.
158. N. Kyriazis and A. Argyros. Physically plausible 3D scene tracking: The single actor hypothesis. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 9–16, 2013.
159. N. Kyriazis and A. Argyros. Scalable 3D tracking of multiple interacting objects. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3430–3437, 2014.
160. C. Lanczos. *The Variational Principles of Mechanics*. Univ. of Toronto Press, 1960.
161. J. Lang. An acquisition method for interactive deformable models. In *2nd International Conference on Creating, Connecting and Collaborating through Computing*, pages 165–170, Oct 2004.

162. F. Largilliere, V. Verona, E. Coevoet, M. Sanz-Lopez, J. Dequidt, and C. Duriez. Real-time control of soft-robots using asynchronous finite element modeling. In *2015 IEEE International Conference on Robotics and Automation*, pages 2550–2555, 2015.
163. S.R. Larimi, P. Zarafshan, and S.A.A. Moosavian. A new stabilization algorithm for a two-wheeled mobile robot aided by reaction wheel. *Journal of Dynamic Systems, Measurement, and Control*, 137(1):011009–1–011009–8, 2015.
164. T. Lauwers, G. Kantor, and R. Hollis. One is enough! In *12th International Symposium for Robotics Research*, pages 12–15, 2005.
165. T.B. Lauwers, G.A. Kantor, and R.L. Hollis. A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive. In *2006 IEEE International Conference on Robotics and Automation*, pages 2884–2889, 2006.
166. K.-K. Lee, G. Bätz, and D. Wollherr. Basketball robot: Ball on plate with pure haptic information. In *2008 IEEE International Conference on Robotics and Automation*, pages 2410–2415, Pasadena, CA, USA, 2008.
167. S. Leutenegger, M. Chli, and R.Y. Siegwart. BRISK: Binary Robust Invariant Scalable Keypoints. In *2011 International Conference on Computer Vision*, pages 2548–2555, 2011.
168. S. Leutenegger and P. Fankhauser. Modeling and control of a ballbot. Bachelor thesis, Swiss Federal Institute of Technology, 2010.
169. A.D. Lewis and R.M. Murray. Variational principles for constrained systems: theory and experiment. *International Journal of Non-Linear Mechanics*, 30(6):793–815, 1995.
170. C.-W. Liao, C.-C. Tsai, Y.Y. Li, and C.-K. Chan. Dynamic modeling and sliding-mode control of a ball robot with inverse mouse-ball drive. In *SICE Annual Conference, 2008*, pages 2951–2955, 2008.
171. V. Lippiello, F. Ruggiero, and B. Siciliano. Floating visual grasp of unknown objects using an elastic reconstruction surface. In C. Pradalier, R. Siegwart, and G. Hirzinger, editors, *Robotics Research: The Fourteenth International Symposium, in Springer Tracts in Advanced Robotics 70*, pages 329–344. Springer, 2011.
172. V. Lippiello, F. Ruggiero, and B. Siciliano. 3D monocular robotic ball catching. *Robotics and Autonomous Systems*, 61(12):1615–1625, 2013.
173. V. Lippiello, F. Ruggiero, and B. Siciliano. The effects of shapes in input-state linearization for stabilization of nonprehensile planar rolling dynamic manipulation. *IEEE Robotics and Automation Letters*, 1(1):492–499, 2016.
174. G.R. Liu, J. Zhang, K.Y. Lam, H. Li, G. Xu, Z.H. Zhong, G.Y. Li, and X. Han. A gradient smoothing method (GSM) with directional correction for solid mechanics problems. *Computational Mechanics*, 41(3):457–472, 2008.
175. K.-C. Liu, J. Friend, and L. Yeo. Rotating bouncing disks, tossing pizza dough, and the behavior of ultrasonic motors. *Physical Review E*, 80(4):046201–1–046201–11, 2009.
176. M.B. Liu and G.R. Liu. Smoothed Particle Hydrodynamics (SPH): An overview and recent developments. *Archives of Computational Methods in Engineering*, 17(1):25–76, 2010.
177. S.-Y. Liu, Y. Rizal, and M.-T. Ho. Stabilization of a ball and sphere system using feedback linearization and sliding mode control. In *8th Asian Control Conference*, pages 1334–1339, Kaohsiung, Taiwan, 2011.
178. B.A. Lloyd, G. Székely, and M. Harders. Identification of spring parameters for deformable object simulation. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):1081–1094, 2007.
179. D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.
180. D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
181. L.B. Lucy. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82:1013–1024, 1977.

182. K.M. Lynch and M.T. Mason. Dynamic nonprehensile manipulation: Controllability, planning, and experiments. *International Journal of Robotics Research*, 18(1):64–92, 1999.
183. K.M. Lynch and F.C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, first edition, 2017.
184. K.M. Lynch, N. Shiroma, H. Arai, and K. Tanie. The roles of shape and motion in dynamic manipulation: The butterfly example. In *1998 IEEE International Conference on Robotics and Automation*, pages 1958–1963, 1998.
185. M. Macklin and M. Müller. Position based fluids. *ACM Transaction on Graphics*, 32(4):104:1–104:12, 2013.
186. E. Magaña Barajas, B. Ramírez-Wong, P.I. Torres-Chávez, and I. Morales-Rosas. Use of the stress-relaxation and dynamic tests to evaluate the viscoelastic properties of dough from soft wheat cultivars. In Juan de Vicente, editor, *Viscoelasticity*, pages 259–272. IntechOpen, 2012.
187. M.A. Malcolm. On the computation of nonlinear spline functions. *SIAM Journal on Numerical Analysis*, 14(2):254–282, 1977.
188. A. Malti, R. Hartley, A. Bartoli, and J.-H. Kim. Monocular template-based 3D reconstruction of extensible surfaces with local linear elasticity. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1522–1529, 2013.
189. S. Marchesseau, T. Heimann, S. Chatelin, R. Willinger, and H. Delingette. Fast porous visco-hyperelastic soft tissue model for surgery simulation: Application to liver surgery. *Progress in biophysics and molecular biology*, 103(2):185–196, 2010.
190. A. Marigo and A. Bicchi. Rolling bodies with regular surface: Controllability theory and applications. *IEEE Transactions on Automatic Control*, 45(9):1586–1599, 2000.
191. J.E. Marsden and T.S. Ratiu. *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. Texts in Applied Mathematics. Springer, 1999.
192. S.A. Mascaro and H.H. Asada. Photoplethysmograph fingernail sensors for measuring finger forces without haptic obstruction. *IEEE Transactions on Robotics and Automation*, 17(5):698–708, 2001.
193. J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.
194. T. McInerney and D. Terzopoulos. A finite element model for 3D shape reconstruction and nonrigid motion tracking. In *1993 IEEE International Conference on Computer Vision*, pages 518–523, 1993.
195. U. Meier, O. López, C. Monserrat, M.C. Juan, and M. Alcaniz. Real-time deformable models for surgery simulation: A survey. *Computer Methods and Programs in Biomedicine*, 77(3):183–197, 2005.
196. M.S. Menon, G.K. Ananthasuresh, and A. Ghosal. Natural motion of one-dimensional flexible objects using minimization approaches. *Mechanism and Machine Theory*, 67:64–76, 2013.
197. M.S. Menon, V.C. Ravi, and A. Ghosal. Trajectory planning and obstacle avoidance for hyper-redundant serial robots. *Journal of Mechanisms and Robotics*, 9(4):041010–1–041010–9, 2017.
198. D.N. Metaxas. *Physics-based Deformable Models: Applications to Computer Vision, Graphics and Medical Imaging*, volume 389. Springer Science & Business Media, 2012.
199. G.S.P. Miller. The motion dynamics of snakes and worms. In *15th Annual Conference on Computer Graphics and Interactive Techniques*, pages 169–173, 1988.
200. H. Mirsaeedghazi, Z. Eman-Djomeh, and S.M.A. Mousavi. Rheometric measurement of dough rheological characteristics and factors affecting it. *International Journal of Agriculture & Biology*, 10(1):112–119, 2008.
201. E. Mitsoulis. Numerical simulation of calendering viscoplastic fluids. *Journal of Non-Newtonian Fluid Mechanics*, 154(2-3):77–88, 2008.

202. E. Mitsoulis and S.G. Hatzikiriakos. Rolling of bread dough: Experiments and simulations. *Food and Bioproducts Processing*, 87(2):124–138, 2009.
203. J.J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30(1):543–574, 1992.
204. J.J. Monaghan. Simulating free surface flows with SPH. *Journal of Computational Physics*, 110(2):399–406, 1994.
205. J.J. Monaghan. Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(1):1703–1759, 2005.
206. J.J. Monaghan and R.A. Gingold. Shock simulation by the particle method SPH. *Journal of Computational Physics*, 52(2):374–389, 1983.
207. D.J. Montana. The kinematics of contact and grasp. *The International Journal of Robotics Research*, 7(3):17–32, 1988.
208. J.P. Morris, P.J. Fox, and Y. Zhu. Modeling low Reynolds number incompressible flows using SPH. *Journal of Computational Physics*, 136(1):214–226, 1997.
209. M. Müller, D. Charypar, and M. Gross. Particle-based fluid simulation for interactive applications. In *ACM Eurographics/SIGGRAPH Symposium on Computer Animation*, pages 154–159, 2003.
210. M. Müller and M. Gross. Interactive virtual materials. In *Graphics Interface 2004*, pages 239–246, 2004.
211. M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff. Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109–118, 2007.
212. M. Müller, R. Keiser, A. Nealen, M. Pauly, M. Gross, and M. Alexa. Point based animation of elastic, plastic and melting objects. In *2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 141–151, 2004.
213. M. Müller, T.-Y. Kim, and N. Chentanez. Fast simulation of inextensible hair and fur. In *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, pages 39–44, 2012.
214. M. Müller, L. McMillan, J. Dorsey, and R. Jagnow. *Real-time Simulation of Deformation and Fracture of Stiff Materials*. Springer, 2001.
215. T.D. Murphey and J.W. Burdick. Feedback control methods for distributed manipulation systems that involve mechanical contacts. *The International Journal of Robotics Research*, 23(7-8):763–781, 2004.
216. R.M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC press, 1994.
217. U. Nagarajan, G. Kantor, and R. Hollis. Integrated planning and control for graceful navigation of shape-accelerated underactuated balancing mobile robots. In *2012 IEEE International Conference on Robotics and Automation*, pages 136–141, 2012.
218. U. Nagarajan, G. Kantor, and R. Hollis. The ballbot: An omnidirectional balancing mobile robot. *The International Journal of Robotics Research*, 33(6):917–930, 2013.
219. U. Nagarajan, Byungjun Kim, and R. Hollis. Planning in high-dimensional shape space for a single-wheeled balancing mobile robot with arms. In *2012 IEEE International Conference on Robotics and Automation*, pages 130–135, 2012.
220. U. Nagarajan, A. Mampetta, G.A. Kantor, and R.L. Hollis. State transition, balancing, station keeping, and yaw control for a dynamically stable single spherical wheel mobile robot. In *2009 IEEE International Conference on Robotics and Automation*, pages 998–1003, 2009.
221. A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson. Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25(4):809–836, 2006.
222. D.N. Nenchev. Redundancy resolution through local optimization: A review. *Journal of Robotic Systems*, 6(6):769–798, 1989.
223. M. Nesme, Y. Payan, and F. Faure. Efficient, physically plausible finite elements. In *Eurographics*, 2005.

224. R.A. Newcombe, A.J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon. KinectFusion: real-time dense surface mapping and tracking. In *10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011.
225. R.A. Newcombe, D. Fox, and S.M. Seitz. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, pages 343–352, 2015.
226. H.G. Nguyen, J. Morrell, K.D. Mullens, A.B. Burmeister, S. Miles, N. Farrington, K.M. Thomas, and D.W. Gage. Segway robotic mobility platform. In *Mobile Robots XVII*, volume 5609, pages 207–220. SPIE, 2004.
227. J. Nishizaki, S. Nakamura, and M. Sampei. Modeling and control of hula-hoop system. In *48th IEEE Conference on Decision and Control*, pages 4125–4130, Shanghai, C, 2009.
228. A. Norton, G. Turk, B. Bacon, J. Gerth, and P. Sweeney. Animation of fracture by physical modeling. *The Visual Computer*, 7(4):210–219, 1991.
229. J.F. O'Brien and J.K. Hodgins. Graphical modeling and animation of brittle fracture. In *26th Annual Conference on Computer Graphics and Interactive Techniques*, pages 137–146, 1999.
230. I. Oikonomidis, N. Kyriazis, and A.A. Argyros. Full DoF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *2011 IEEE International Conference on Computer Vision*, pages 2088–2095, 2011.
231. I. Oikonomidis, N. Kyriazis, and A.A. Argyros. Tracking the articulated motion of two strongly interacting hands. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1862–1869, 2012.
232. R. Olfati-Saber. Cascade normal forms for underactuated mechanical systems. In *39th IEEE Conference on Decision and Control*, pages 2162–2167, 2000.
233. G. Oriolo and Y. Nakamura. Control of mechanical systems with second-order nonholonomic constraints: Underactuated manipulators. In *30th IEEE Conference on Decision and Control*, pages 2398–2403, 1991.
234. G. Oriolo and M. Vendittelli. A framework for the stabilization of general nonholonomic systems with an application to the plate-ball mechanism. *IEEE Transactions on Robotics*, 21(2):162–175, 2005.
235. R. Ortega, A. Donaire, and J.G. Romero. Passivity-based control of mechanical systems. In *Feedback Stabilization of Controlled Dynamical Systems*, pages 167–199. Springer, 2017.
236. R. Ortega, M.W. Spong, F. Gómez-Estern, and G. Blankenstein. Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment. *IEEE Transactions on Automatic Control*, 47(8):1218–1233, 2002.
237. J. O'hara. Energy of a knot. *Topology*, 30(2):241–247, 1991.
238. D.K. Pai. Strands: Interactive simulation of thin solids using cosserat models. *Computer Graphics Forum*, 21(3):347–352, 2002.
239. C.J. Paulus, N. Haouchine, S.-H. Kong, R.V. Soares, D. Cazier, and S. Cotin. Handling topological changes during elastic registration. *International Journal of Computer Assisted Radiology and Surgery*, 12(3):461–470, 2017.
240. A. Peer and M. Ihmsen. Prescribed velocity gradients for highly viscous SPH fluids with vorticity diffusion. *IEEE Transactions on Visualization and Computer Graphics*, 23(12):2656–2662, 2016.
241. A. Peer, M. Ihmsen, J. Cornelis, and M. Teschner. An implicit viscosity formulation for SPH fluids. *ACM Transactions on Graphics*, 34(4):114:1–114:10, 2015.
242. A. Petit, S. Cotin, V. Lippiello, and B. Siciliano. Capturing deformations of interacting non-rigid objects using RGB-D data. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 491–497, Madrid, E,, 2018.
243. 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*, 88:187–201, 2017.

244. A. Petit, V. Lippiello, and B. Siciliano. Real-time tracking of 3D elastic objects with an RGB-D sensor. In *2015 IEEE International Conference on Intelligent Robots and Systems*, pages 3914–3921, 2015.
245. A. Petit, V. Lippiello, and B. Siciliano. Tracking fractures of deformable objects in real-time with an RGB-D sensor. In *2015 International Conference on 3D Vision*, pages 632–639, Lyon, F, 2015.
246. T.-H. Pham, A. Kheddar, A. Qammaz, and A.A. Argyros. Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2810–2819, 2015.
247. J. Phillips, A. Ladd, and L.E. Kavraki. Simulated knot tying. In *2002 IEEE International Conference on Robotics and Automation*, pages 841–846, 2002.
248. C. Phillips-Grafflin and D. Berenson. A representation of deformable objects for motion planning with no physical simulation. In *2014 IEEE International Conference on Robotics and Automation*, pages 98–105, 2014.
249. G. Picinbono, H. Delingette, and N. Ayache. Real-time large displacement elasticity for surgery simulation: Non-linear tensor-mass model. In *Medical Image Computing and Computer-Assisted Intervention*, pages 643–652, 2000.
250. J. Pilet, V. Lepetit, and P. Fua. Fast non-rigid surface detection, registration and realistic augmentation. *International Journal of Computer Vision*, 76(2):109–122, 2007.
251. A. Powell. MIT materials science and engineering. 3.21 Lectures on Fluid Flow and Kinectics, 2003.
252. D. Prattichizzo and J.C. Trinkle. Grasping. In Bruno Siciliano and Oussama Khatib, editors, *Springer Handbook of Robotics*, pages 955–988. Springer Science & Business Media, second edition, 2016.
253. S. Premžoe, T. Tasdizen, J. Bigler, A. Lefohn, and R.T. Whitaker. Particle-based simulation of fluids. *Computer Graphics Forum*, 22(3):401–410, 2003.
254. I.G. Ramirez-Alpizar, M. Higashimori, M. Kaneko, C.-H. Tsai, and I. Kao. Nonprehensile dynamic manipulation of a sheet-like viscoelastic object. In *2011 IEEE International Conference on Robotics and Automation*, pages 5103–5108, 2011.
255. I.G. Ramirez-Alpizar, M. Higashimori, M. Kaneko, C.-H.D. Tsai, and I. Kao. Dynamic nonprehensile manipulation for rotating a thin deformable object: An analogy to bipedal gaits. *IEEE Transactions on Robotics*, 28(3):607–618, 2012.
256. V.C. Ravi, S. Rakshit, and A. Ghosal. Redundancy resolution using tractrix—simulations and experiments. *Journal of Mechanisms and Robotics*, 2(3):031013–1–031013–7, 2010.
257. W.T. Reeves. Particle systems—a technique for modeling a class of fuzzy objects. *ACM SIGGRAPH Computer Graphics*, 17(3):359–375, 1983.
258. M. Reyhanoglu, A. van der Schaft, N.H. McClamroch, and I. Kolmanovsky. Dynamics and control of a class of underactuated mechanical systems. *IEEE Transactions on Automatic Control*, 44(9):1663–1671, 1999.
259. D. Reznik, E. Moshkovich, and J. Canny. Building a universal planar manipulator. In *Distributed Manipulation*, pages 147–171. Springer, 2000.
260. D.S. Reznik and J.F. Canny. C'mon part, do the local motion! In *2001 IEEE International Conference on Robotics and Automation*, pages 2235–2242, 2001.
261. E. Rossi, A. Colagrossi, D. Durante, and G. Graziani. Simulating 2D viscous flow around geometries with vertices through the diffused vortex hydrodynamics method. *Computer Methods in Applied Mechanics and Engineering*, 302:147–169, 2016.
262. E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, 1, pages 430–443, 2006.
263. C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23:309–314, 2004.
264. M. Ryalat and D.S. Laila. A simplified IDA–PBC design for underactuated mechanical systems with applications. *European Journal of Control*, 27:1–16, 2016.

265. J.-C. Ryu, F. Ruggiero, and K.M. Lynch. Control of nonprehensile rolling manipulation: Balancing a disk on a disk. *IEEE Transactions on Robotics*, 29(5):1152–1161, 2013.
266. M. Salzmann, J. Pilet, S. Ilic, and P. Fua. Surface deformation models for nonrigid 3D shape recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1481–1487, 2007.
267. M. Salzmann and R. Urtasun. Physically-based motion models for 3D tracking: A convex formulation. In *2011 IEEE International Conference on Computer Vision*, pages 2064–2071, 2011.
268. A.C. Satici, A. Donaire, and B. Siciliano. Intrinsic dynamics and total energy-shaping control of the ballbot systems. *International Journal of Control*, 90:2734–2747, 2017.
269. A.C. Satici, F. Ruggiero, V. Lippiello, and B. Siciliano. Coordinate-free framework for robotic pizza tossing and catching. In *2016 IEEE International Conference on Robotics and Automation*, pages 3932–3939, Stockholm, S, 2016.
270. A.C. Satici, F. Ruggiero, V. Lippiello, and B. Siciliano. Intrinsic euler-lagrange dynamics and control analysis of the ballbot. In *2016 American Control Conference*, pages 5685–5690, 2016.
271. K. Sato, K. Kamiyama, N. Kawakami, and S. Tachi. Finger-shaped gelforce: Sensor for measuring surface traction fields for robotic hand. *IEEE Transactions on Haptics*, 3(1):37–47, 2010.
272. M.M. Schill, F. Gruber, and M. Buss. Quasi-direct nonprehensile catching with uncertain object states. In *2015 IEEE International Conference on Robotics and Automation*, pages 2468–2474, 2015.
273. D. Schneider. Non-holonomic Euler-Poincaré equations and stability in Chaplygin's sphere. *Dynamical Systems: An International Journal*, 17(2):87–130, 2002.
274. J. Schulman, A. Lee, J. Ho, and P. Abbeel. Tracking deformable objects with point clouds. In *2013 IEEE International Conference on Robotics and Automation*, pages 1130–1137, 2013.
275. T.W. Sederberg and S.R. Parry. Free-form deformation of solid geometric models. In *13th Annual Conference on Computer Graphics and Interactive Techniques*, pages 151–160, 1986.
276. D. Serra, J. Ferguson, F. Ruggiero, A. Siniscalco, A. Petit, V. Lippiello, and B. Siciliano. On the experiments about the nonprehensile reconfiguration of a rolling sphere on a plate. In *26th Mediterranean Conference on Control and Automation*, pages 13–20, Zadar, HR, 2018.
277. D. Serra, F. Ruggiero, A. Donaire, L.R. Buonocore, V. Lippiello, and B. Siciliano. Control of nonprehensile planar rolling manipulation: A passivity-based approach. *IEEE Transactions on Robotics*, 35(2):317–329, 2019.
278. A.P. Seyranian and A.O. Belyakov. How to twirl a hula hoop. *American Journal of Physics*, 79(7):712–715, 2011.
279. S. Shao and E.Y.M. Lo. Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface. *Advances in Water Resources*, 26(7):787–800, 2003.
280. A. Shiriaev, J.W. Perram, and C. Canudas-de Wit. Constructive tool for orbital stabilization of underactuated nonlinear systems: Virtual constraints approach. *IEEE Transactions on Automatic Control*, 50(8):1164–1176, 2005.
281. A.S. Shiriaev, L.B. Freidovich, and S.V. Gusev. Transverse linearization for controlled mechanical systems with several passive degrees of freedom. *IEEE Transactions on Automatic Control*, 55(4):893–906, 2010.
282. A.S. Shiriaev, L.B. Freidovich, and M.W. Spong. Controlled invariants and trajectory planning for underactuated mechanical systems. *IEEE Transactions on Automatic Control*, 59(9):2555–2561, 2014.
283. B. Siciliano and O. Khatib. *Springer Handbook of Robotics*. Springer Science & Business Media, second edition, 2016.

284. B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer, London, UK, 2009.
285. J.K. Simon. Energy functions for polygonal knots. *Journal of Knot Theory and its Ramifications*, 3(3):299–320, 1994.
286. J.-J. Slotine and W. Li. *Applied Nonlinear Control*, volume 199. Prentice-Hall, 1991.
287. C. Smith, Y. Karayiannidis, L. Nalpantidis, X. Gratal, P. Qi, D.V. Dimarogonas, and D. Kragic. Dual arm manipulation: A survey. *Robotics and Autonomous Systems*, 60(10):1340–1353, 2012.
288. J. Smith, A. Witkin, and D. Baraff. Fast and controllable simulation of the shattering of brittle objects. *Computer Graphics Forum*, 20(2):81–91, 2001.
289. J. Smolen and A. Patriciu. Deformation planning for robotic soft tissue manipulation. In *2009 Second International Conferences on Advances in Computer-Human Interactions*, pages 199–204, 2009.
290. S. Sofou and E. Mitsoulis. Calendering of pseudoplastic and viscoplastic sheets of finite thickness. *Journal of Plastic Film & Sheeting*, 20:185–222, 2004.
291. S. Sofou, E.B. Muliawan, S.G. Hatzikiriakos, and E. Mitsoulis. Rheological characterization and constitutive modeling of bread dough. *Rheologica Acta*, 47(4):369–381, 2008.
292. B. Solenthaler and R. Pajarola. Predictive-corrective incompressible SPH. *ACM Transactions on Graphics*, 28(3):40:1–40:6, 2009.
293. B. Solenthaler and R. Pajarola. Predictive-corrective incompressible SPH. In *ACM SIGGRAPH 2009 Papers*, pages 40:1–40:6, 2009.
294. J. Spillmann and M. Teschner. CoRdE: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 63–72, 2007.
295. J. Spillmann and M. Teschner. An adaptive contact model for the robust simulation of knots. *Computer Graphics Forum*, 27(2):497–506, 2008.
296. M.W. Spong. Partial feedback linearization of underactuated mechanical systems. In *1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, pages 314–321, 1994.
297. M.W. Spong. Partial feedback linearization of underactuated mechanical systems. In *1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, pages 314–321, 1994.
298. S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt. Real-time joint tracking of a hand manipulating an object from RGB-D input. In *European Conference on Computer Vision*, pages 294–310, 2016.
299. A. Stomakhin, C. Schroeder, L. Chai, J. Teran, and A. Selle. A material point method for snow simulation. *ACM Transactions on Graphics*, 32(4):102:1–102:10, 2013.
300. D. Sulsky, S.-J. Zhou, and H.L. Schreyer. Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications*, 87(1):236–252, 1995.
301. Y. Sun, J.M. Hollerbach, and S.A. Mascaro. Predicting fingertip forces by imaging coloration changes in the fingernail and surrounding skin. *IEEE Transactions on Biomedical Engineering*, 55(10):2363–2371, 2008.
302. Y. Sun, J.M. Hollerbach, and S.A. Mascaro. Estimation of fingertip force direction with computer vision. *IEEE Transactions on Robotics*, 25(6):1356–1369, 2009.
303. M. Surov, A. Shiriaev, L. Freidovich, S. Gusev, and L. Paramonov. Case study in non-prehensile manipulation: Planning perpetual rotations for "Butterfly" robot. In *2015 IEEE International Conference on Robotics and Automation*, pages 1484–1489, 2015.
304. H.J. Sussmann. Subanalytic sets and feedback control. *Journal of Differential Equations*, 31(1):31–52, 1979.
305. H.J. Sussmann and V. Jurdjevic. Controllability of nonlinear systems. *Journal of Differential Equations*, 12(1):95–116, 1972.

306. T. Takahashi, Y. Dobashi, I. Fujishiro, and T. Nishita. Volume preserving viscoelastic fluids with large deformations using position-based velocity corrections. *The Visual Computer*, 32(1):57–66, 2016.
307. T. Takahashi, Y. Dobashi, I. Fujishiro, T. Nishita, and M.C. Lin. Implicit formulation for SPH-based viscous fluids. *Computer Graphics Forum*, 34(2):493–502, 2015.
308. A. Tejani, D. Tang, R. Kouskouridas, and T.-K. Kim. Latent-Class Hough Forests for 3D object detection and pose estimation. In *European Conference on Computer Vision*, pages 462–477, 2014.
309. R. Temam. *Navier-Stokes Equations: Theory and Numerical Analysis*. AMS Chelsea Publishing, 1977.
310. M. Tenorth and M. Beetz. A unified representation for reasoning about robot actions, processes, and their effects on objects. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1351–1358, 2012.
311. J. Teran, S. Blemker, V. Hing, and R. Fedkiw. Finite volume methods for the simulation of skeletal muscle. In *2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 68–74, 2003.
312. D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. *ACM Siggraph Computer Graphics*, 21(4):205–214, 1987.
313. D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36(1):91–123, 1988.
314. M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-.P Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, and P. Volino. Collision detection for deformable objects. *Computer Graphics Forum*, 24:61–81, 2005.
315. E. Todorov, T. Erez, and Y. Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.
316. S. Tokumoto and S. Hirai. Deformation control of rheological food dough using a forming process model. In *2002 IEEE International Conference on Robotics and Automation*, pages 1457–1464, 2002.
317. L. Trefethen, A.E. Trefethen, S.C. Reddy, and T. Driscoll. Hydrodynamic stability without eigenvalues. *Science*, 261(5121):578–584, 1993.
318. A. Tsoli and A.A. Argyros. Tracking deformable surfaces that undergo topological changes using an RGB-D camera. In *2016 Fourth International Conference on 3D Vision*, pages 333–341, 2016.
319. P. Umbanhowar and K.M. Lynch. Optimal vibratory stick-slip transport. *IEEE Transactions on Automation Science and Engineering*, 5(3):537–544, 2008.
320. S. Urban, J. Bayer, C. Osendorfer, G. Westling, B.B. Edin, and P. Van Der Smagt. Computing grip force and torque from finger nail images using gaussian processes. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4034–4039, 2013.
321. S. Uthayakumaran, M. Newberry, M. Keentok, F.L. Stoddard, and F. Bekes. Basic rheology of bread dough with modified protein content and glutenin-to-gliadin ratios. *Cereal Chemistry Journal*, 77(6):744–749, 2000.
322. F. Van Bockstaele, I. De Leyn, M. Eeckhout, and K. Dewettinck. Rheological properties of wheat flour dough and the relationship with bread volume. I. Creep-recovery measurements. *Cereal Chemistry Journal*, 85(6):753–761, 2008.
323. A.J. van der Schaft. *L2-gain and Passivity Techniques in Nonlinear Control*. Springer-Verlag, 2000.
324. K. Varanasi, A. Zaharescu, E. Boyer, and R. Horaud. Temporal surface tracking using mesh evolution. In *European Conference on Computer Vision*, pages 30–43, 2008.
325. M. Vidyasagar. *Nonlinear Systems Analysis*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2002.

326. T.H. Vose, P. Umbanhowar, and K.M. Lynch. Vibration-induced frictional force fields on a rigid plate. In *2007 IEEE International Conference on Robotics and Automation*, pages 660–667, 2007.
327. T.H. Vose, P. Umbanhowar, and K.M. Lynch. Friction-induced lines of attraction and repulsion for parts sliding on an oscillated plate. *IEEE Transactions on Automation Science and Engineering*, 6(4):685–699, 2009.
328. T.H. Vose, P. Umbanhowar, and K.M. Lynch. Sliding manipulation of rigid bodies on a controlled 6-DoF plate. *The International Journal of Robotics Research*, 31(7):819–838, 2012.
329. H. Wakamatsu, S. Hirai, and K. Iwata. Modeling of linear objects considering bend, twist, and extensional deformations. In *1995 IEEE International Conference on Robotics and Automation*, pages 433–438, 1995.
330. H. Wakamatsu, K. Takahashi, and S. Hirai. Dynamic modeling of linear object deformation based on differential geometry coordinates. In *2005 IEEE International Conference on Robotics and Automation*, pages 1028–1033, 2005.
331. B. Wang, L. Wu, K. Yin, U. Ascher, L. Liu, and H. Huang. Deformation capture and modeling of soft objects. *ACM Transactions on Graphics*, 34(4):94:1–94:12, 2015.
332. Y. Wang, J. Min, J. Zhang, Y. Liu, F. Xu, Q. Dai, and J. Chai. Video-based hand manipulation capture through composite motion control. *ACM Transactions on Graphics*, 32(4):43, 2013.
333. Z. Wang and S. Hirai. Modeling and estimation of rheological properties of food products for manufacturing simulations. *Journal of Food Engineering*, 102(2):136–144, 2011.
334. K. Ward, F. Bertails, T.-Y. Kim, S.R. Marschner, M.-P. Cani, and M.C. Lin. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):213–234, 2007.
335. J. Weil. The synthesis of cloth objects. *ACM Siggraph Computer Graphics*, 20(4):49–54, 1986.
336. M. Weiler, D. Koschier, M. Brand, and J. Bender. A physically consistent implicit viscosity solver for SPH fluids. *Computer Graphics Forum*, 37(2):145–155, 2018.
337. A. Weiss, D. Hirshberg, and M.J. Black. Home 3D body scans from noisy image and range data. In *2011 IEEE International Conference on Computer Vision*, pages 1951–1958, 2011.
338. N. Wettels, J.A. Fishel, Z. Su, C.H. Lin, and G.E. Loeb. Multi-modal synergistic tactile sensing. In *Tactile Sensing in Humanoids—Tactile Sensors and Beyond Workshop, 9th IEEE-RAS International Conference on Humanoid Robots*, 2009.
339. H. Wieser. Chemistry of gluten proteins. *Food Microbiology*, 24(2):115–119, 2007.
340. J.P. Wilhelms and B.A. Barsky. Using dynamic analysis to animate articulated bodies such as humans and robots. In *Computer-Generated Images*, pages 209–229. Springer, 1985.
341. A. Witkin, K. Fleischer, and A. Barr. Energy constraints on parameterized models. *ACM SIGGRAPH Computer Graphics*, 21(4):225–232, 1987.
342. X. X, J. Ouyang, W. Li, and Q. Liu. SPH simulations of 2D transient viscoelastic flows using brownian configuration fields. *Journal of Non-Newtonian Fluid Mechanics*, 208:59–71, 2014.
343. T. Yoshikawa. Analysis and control of robot manipulators with redundancy. In *Robotics Research: The First International Symposium*, pages 735–747, 1984.
344. H. Yousef, M. Boukallel, and K. Althoefer. Tactile sensing for dexterous in-hand manipulation in robotics–A review. *Sensors and Actuators A: Physical*, 167(2):171–187, 2011.
345. W. Yuan, R: Li, M.A. Srinivasan, and E.H. Adelson. Measurement of shear and slip with a GelSight tactile sensor. In *2015 IEEE International Conference on Robotics and Automation*, pages 304–311, 2015.

346. A. Zaharescu, E. Boyer, and R. Horaud. Topology-adaptive mesh deformation for surface evolution, morphing, and multiview reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4):823–837, 2011.
347. K.E. Zanganeh and J. Angeles. The inverse kinematics of hyper-redundant manipulators using splines. In *1995 IEEE International Conference on Robotics and Automation*, pages 2797–2802, 1995.
348. M. Zefran, V. Kumar, and C.B. Croke. On the generation of smooth three-dimensional rigid body motions. *IEEE Transactions on Robotics and Automation*, 14(4):576–589, 1998.
349. W. Zhao, J. Zhang, J. Min, and J. Chai. Robust realtime physics-based motion control for human grasping. *ACM Transactions on Graphics*, 32(6):1–12, 2013.
350. Y. Zhu and R. Bridson. Animating sand as a fluid. *ACM Transactions on Graphics*, 24(3):965–972, 2005.
351. O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method Set*. Butterworth-Heinemann, 2005.
352. M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger. Real-time non-rigid reconstruction using an RGB-D camera. *ACM Transactions on Graphics*, 33(4):1–12, 2014.