MaterialCloning: Acquiring Elasticity Parameters from Images for Medical Applications

Shan Yang and Ming C. Lin, *Fellow, IEEE*

Abstract—We present a practical approach for automatically estimating the material properties of soft bodies from two sets of images, taken before and after deformation. We reconstruct 3D geometry from the given sets of multiple-view images; we use a coupled simulation-optimization-identification framework to deform one soft body at its original, non-deformed state to match the deformed geometry of the same object in its deformed state. For shape correspondence, we use a distance-based error metric to compare the estimated deformation fields against the actual deformation field from the reconstructed geometry. The optimal set of material parameters is thereby determined by minimizing the error metric function. This method can simultaneously recover the elasticity parameters of multiple types of soft bodies using Finite Element Method-based simulation (of either linear or nonlinear materials undergoing large deformation) and particle-swarm optimization methods. We demonstrate this approach on real-time interaction with virtual organs in patient-specific surgical simulation, using parameters acquired from low-resolution medical images. We also highlight the results on physics-based animation of virtual objects using sketches from an artist's conception.

Index Terms—Physically based modeling, optimization, inverse problems

1 Introduction

Physically-based animation, unlike traditional keyframing, can automatically generate realistic motion and deformation without tedious and time-consuming low-level control [1], [2] and offers greater flexibility, allowing for quick prototyping of different designs of complex artifacts. But in physically-based animation, simulation parameters (such as material properties) often require many iterations of manual adjustment and re-assessment of the visual results. This iterative process is both unintuitive and costly. Automatic parameter identification from real-world data, such as simulation results, images, audio, video, and animators' sketches, is thus becoming a topic of increasing interest in computer animation [3], [4], [5], [6], [7], [8].

For example, the use of material-property estimation in the simulation of cloth has been suggested in [5], [6], [9]; it has also been used in procedures for designing and fabricating materials that produce a certain deformation behavior [4]. These parameter estimation methods focus on materials that can be put into a specialized video capturing system to measure displacements, and (in the case of elasticity parameters) often require a force measuring device [10], [11].

Beyond computer graphics, parameter estimation is also an area of interest in medical applications such as non-invasive cancer detection, since human tissues are generally difficult to measure and it is sometimes impossible to acquire the actual parameters of a live patient. Currently, 2D Elastography [12], [13], [14] is used to estimate the elasticity value of each pixel in medical images; most existing methods are based on a dense displacement field established by pixel-wise correspondence between pre- and post-compression images. However, in some imaging modalities, it is difficult to find a reliable dense displacement field, and some organs, such as prostate, are located so deep within the body that direct force measurements (such as 3D elastography) are practically impossible to use. Furthermore, the majority of the measurements taken in a laboratory are performed on organs *ex vivo*, but bodies' tissue properties can change from *in situ* to *ex vivo*. Therefore, obtaining patient-specific tissue elasticity parameters remains a practical challenge.

In computer animation, artists and animators often sketch out keyframes of conceptualized motion and body deformation. These drawings are mental images of animated *virtual models* that do not exist, so it is impossible to measure the elasticity parameters of these virtual objects to reproduce the artist's desired deformation. An efficient method for inferring these parameters directly from artist's sketches (which can be treated as images) can help automate the selection of elasticity parameters for physically-based animation of deformable bodies based on given hand drawings of keyframes.

In this paper, we introduce the "MaterialCloning" framework, which acquires elasticity parameters directly from images. We assume that (at least) two sets of multiple-view images for the same object are available to reconstruct the original geometry before and after deformation using state-of-art 3D geometry reconstruction [15], [16], [17], [18]. A 3D finite-element mesh is then constructed for each geometric representation of the deformable model using FEM mesh generation systems, such as TetGen [19]. We further assume, in order to set the boundary condition in the Finite Element Method (FEM) simulation, that the material parameters for the surrounding media are known (the standard default values); these parameters are used to compute the

- *The authors are with the Department of Computer Science, University of North Carolina, Chapel Hill, NC 27599-3175.*
  E-mail: {alexyang, lin}@cs.unc.edu.

Manuscript received 24 Nov. 2014; revised 14 Nov. 2015; accepted 23 Nov. 2015. Date of publication 3 Dec. 2015; date of current version 3 Aug. 2016.
Recommended for acceptance by J. Lee.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2015.2505285

contact force between the deformable body of interest and the surrounding media in solving the governing equation. Our algorithm then minimizes an (error) objective function based on the Hausdorff distance between the deformed mesh and the reference (target) surface mesh using a coupled simulation-optimization-identification framework, with a set of initial (default) elasticity values and boundary forces as the input parameters to the iterative optimizer. The optimization process is the critical part of our framework. We carefully design the error metric and choose the optimization method for different applications. When we have only two sets of images as input, we choose the Hausdorff distance between the deformed and the reference surfaces as our error metric. We choose the optimization method based on accuracy and convergence rate. In applications where the dimensionality of the boundary forces is very high, the optimizer can become slow to converge and is more likely get stuck in a local minimum. In these cases, we apply the Particle Swarm Optimization (PSO) method, which is agnostic and less prone to local-minima entrapment problems. Note that the elasticity parameter recovered using this framework is a *relative* value to the assumed material parameters of the surrounding media. This parameter value is very useful, especially for the target medical applications demonstrated in this paper.

The rest of this paper is structured as follows. We first review related work in Section 2. Section 3 gives an overview of the optimization framework and of the steps in our approach. We demonstrate the effectiveness of our algorithm by highlighting experimental results in shape deformation and 3D elastography in Section 4 , where we also analyze the accuracy of the recovered elasticity values. We conclude with a summary and discussion of future work.

2 Related Work

Physically-based deformable body simulation has been extensively studied in computer graphics for decades [1], [2], [20], [21], [22]. Here we briefly discuss prior work on the estimation of material parameters. Parameter estimation has received increasing attention in computer graphics for its uses in rendering, modeling, and simulating different materials. Pai et al. integrated a force measurement device with a trinocular stereo system to model the material properties of deformable objects by estimating the linear relationship between displacements and tractions using a least-squares formulation [10]. Other methods have combined a force sensor with a trinocular stereo vision system to measure the forces applied and the displacements of the vertices on the deformable surface; these measurements are then used to determine the nonlinear heterogeneous material properties [11]. Syllebranque and Boivin [23] used a force-capture device to measure boundary forces and estimate the mechanical properties of deformable solids. By using video-based metrics to optimize for Poisson's ratio and for the errors in computed boundary forces, their method optimized the value of Young's modulus.

Other models have focused on physically-based simulation of cloth, especially on determining the cloth model's stiffness and damping coefficients. [9] estimated cloth simulation parameters by comparing video of real fabric patches with simulated images; they used the orientation of each edge pixel to compute the error metric, and used the continuous simulated annealing method [24] to minimize estimation error. Becker and Teschner presented a novel framework based on quadratic programming to determine linear elastic parameters; they also assessed the effects of noisy measurements [25]. Most recently, improved data-driven methods have been proposed to estimate cloth parameters [5], [6], [7], which can photorealistically recreate the look of real fabric. While all these methods depend on rendering and/or vision algorithms that use known measurements and forces, our technique adopts a parallelizable Particle Swarm Optimization method that 'explores' and 'exploits' multiple solution candidates simultaneously. It is also more robust in its handling of sensor noise and can, as previously mentioned, minimize the local-minima entrapment problem.

Estimation of material parameters for human tissues is also well-studied in medical image analysis, where it is used in screening and detecting tumors, as cancerous tissues tend to be stiffer than healthy tissues. A non-invasive method, 2D elastography (also known as elasticity reconstruction), can acquire 'strain images' or 'elasticity images' of soft tissues [26]. Elastography is usually done by estimating the optimal deformation field that relates two ultrasound images, one taken at the rest state and the other taken when a known force is applied to the skin [12], [27]. Alternatively, the displacement field can be found with a modified MRI machine that is in tune with a mechanical vibration of tissues [28], [29]. Assuming that the physical model is linear, once the deformation field and external forces are known, the material properties can be computed by simply solving a least-squares problem [13] or by performing an iterative optimization to minimize the error in the deformation field [14], [30], [31]. Although these iterative methods are slower than directly solving the inverse problem, iterative methods are suitable for any physical model, since they do not require linearity of the underlying tissue model. [32] present an alternative 2D elastography for extracting tissue parameters. 3D elastography usually requires force measurements, but it is not always possible to obtain these measurements for deeply-seated organs, such as the prostate. Therefore, our use of medical images to automatically estimate the tissue elasticity parameters offers a practical approach for developing surgical simulators with patient-specific parameters.

3 Method

Given (at least) two sets of multi-view images of a deformable object, our framework can automatically estimate elasticity properties within multiple regions of this model.

Fig. 1 provides an overview of our system. We assume that (at least) two sets of multiple-view images are given, along with some initial guess at the elasticity parameters. First, we offer an overview of how our method uses the input images to produce the geometrical reconstructions that are then used in the elasticity-parameter reconstruction. Next, we describe each component of our system in more detail.

3.1 Geometry Reconstruction and Mesh Generation

There are many approaches to reconstructing 3D geometry from multiple images; the method chosen depends upon

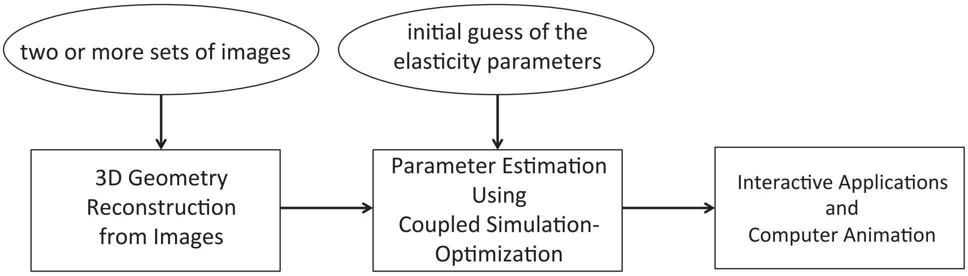
two or more sets of images
initial guess of the elasticity parameters
3D Geometry Reconstruction from Images
Parameter Estimation Using Coupled Simulation-Optimization
Interactive Applications and Computer Animation

Fig. 1. **The Flow Chart of Our Framework.** Our framework takes (at least) two sets of images (medical images or other multi-view images) as input; we use these images to reconstruct 3D meshes. The initial guess at the elasticity parameter is based on standard values and is given prior to the start of the optimization process. For each optimization iteration, the body deformation is recomputed using FEM simulation. The value of the distance objective function is also re-evaluated. At the end of each iteration, the elasticity parameter is updated; the new, refined value is used by the finite element model to continue the simulated-based optimization process.

the image sources. We briefly summarize below how the 3D geometry of input objects can be constructed using various image sources.

Medical Images are usually taken when the organs are in a static or quasi-static state. There are several widely-used imaging technologies, such as X-ray radiography, magnetic resonance imaging (MRI), medical ultrasonography or ultrasound, elastography, tactile imaging, thermography, nuclear medicine functional imaging, computed tomography (CT) scanning or computerized axial tomography (CAT), etc. Each set of CT or CAT scans provides image "slices", or the cross-sectional images of anatomy. Variants of MRI and ultrasound images can be used to reconstruct anatomical 3D geometry using public-domain libraries such as ITK-SNAP [16] or commercial systems such as AVS, 3D-Doctor, MxAnatomy, etc.

2D Drawings and Sketches can be converted to 3D models using widely available commercial CAD and 3D modeling systems, such as Rhino, Autodesk LABS, Dassault Systems SolidWorks, etc.

Multi-view Images from Cameras/Camcorder and other imaging technologies have been used for 3D geometry reconstruction. Excellent surveys of methods of extracting 3D models from images can be found in [15], [16], [17], [18]. These methods include algorithms using images for which camera parameters are unknown, uncalibrated structure-from-motion methods, metric reconstruction from images with additional knowledge about images, etc.

FEM Mesh Generation is accomplished by first building the input surface meshes as described above. If medical images (e.g., CT, MRI, etc.) are used as input they require an additional step before mesh generation: segmenting using ITK-SNAP [16] into multiple regions. After mesh simplification and smoothing, the entire region of interest can be tetrahedralized using TetGen [19].

3.2 Quasi-Static Process Elasticity Parameter Estimation

Our approach consists of two alternating phases: the **forward simulation** and the **inverse optimization process**.

3.2.1 Forward Simulation

This step uses the elasticity parameters generated from the inverse optimization process to compute the amount of deformation that the body would undergo. We use the FEM to solve the following governing equation of the deformable body.

$$\int_{\Omega} \delta \mathbf{u}^T \rho \ddot{\mathbf{u}} \, \mathrm{d}\Omega + \int_{\Omega} \delta(\varepsilon)^T \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{b} \, \mathrm{d}\Omega - \int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} \, \mathrm{d}\Gamma = 0, \tag{1}$$

with $\mathbf{u}$ as the displacement field, ε as the strain tensor, $\boldsymbol{\sigma}$ as the stress tensor, $\mathbf{b}$ as the body force and $\mathbf{t}$ as the tractions on the boundary Γ of the deformable body Ω. For the quasi-static deformation process the $\ddot{\mathbf{u}} = \mathbf{0}$. We can rewrite Eq. (1) as

$$\left[\int_{\Omega} \delta(\varepsilon)^T \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{b} \, \mathrm{d}\Omega \right] - \left[\int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} \, \mathrm{d}\Gamma \right] = 0, \tag{2}$$

with the first part of the equation as the internal body force and the second part as the external force. The computation of the stress force is determined by the material properties. Researchers have proposed many models for simulating different kinds of materials. These material models define the relation between the stress and the strain. To simulate the human organs in the abdomen and the soft tissue surrounding those organs, we use the isotropic hyperelastic material model, which is used commonly to approximate the deformation behavior of human tissue [33]. The stress-strain relation for the hyperelastic material model is defined through the strain energy density function $\boldsymbol{\Psi}$ (energy per unit undeformed volume). We will be using the Green-Lagrange strain tensor $\mathbf{E}$ with the second Piola-Kirchhoff stress tensor $\boldsymbol{\sigma}^{\mathrm{PK2}}$ [34]. The boundary conditions we apply are the tractions $\mathbf{t}$ applied on the boundary Γ. Our forward simulation framework uses an invertible FEM [35] to ensure that the deformed elements have positive volumes in the coupled simulation-optimization process.

3.2.2 Material Model

The elastic behavior of deformable bodies varies for different materials. For small deformations, most elastic materials (e.g., springs) exhibit linear elasticity, which can be described as a linear function between stress and strain.

Linear Elasticity Material Model: The linearly elastic model assumes a constant variation of stress and strain according to Hooke's law, with no permanent deformations after the applied stresses are removed. This holds true until the yield point, which is followed by an unrestricted plastic strain after yield. Assuming isotropic linear elasticity, we can write

$$\boldsymbol{\sigma} = \mathbf{D}\varepsilon, \tag{3}$$

where $\boldsymbol{\sigma}$ is the stress tensor induced by the *surface forces*, ε is the strain tensor defined by the spatial derivatives of the displacement $\mathbf{u}$, and $\mathbf{D}$ is a matrix defined by the material property parameters $\boldsymbol{\mu}$ ($\mathbf{D} = \mathbf{D}(\boldsymbol{\mu})$). Assuming an isotropic material, the commonly used material property parameters are Young's modulus E and Poisson's ratio ν.

Isotropic Nonlinear Hyperelastic Material Model: For many materials, linear elastic models cannot accurately capture the observed material behavior. Hyperelastic material models better describe the nonlinear material behavior exhibited when deformable bodies are subjected to large strains. For example, animal tissue and some common organic materials are known to be hyperelastic [33]. The nonlinearity is captured through the energy density function $\boldsymbol{\Psi}$ for hyperelastic material models. The energy function is a function of the strain tensor ε and the material property parameters $\boldsymbol{\mu}$, where $\boldsymbol{\Psi} = \boldsymbol{\Psi}(\varepsilon, \boldsymbol{\mu})$. With the energy function, the stress tensor is computed by taking the derivative of the energy function over the strain tensor.

$$\boldsymbol{\sigma} = \frac{\partial \boldsymbol{\Psi}(\varepsilon, \boldsymbol{\mu})}{\partial \varepsilon}. \tag{4}$$

The energy function takes different forms for different models of hyperelastic materials.

Energy Function: The energy density function determines the nonlinear behavior of the deformable object. Human organs are hyperelastic and nearly isotropic. Generally speaking, for an isotropic material model, the energy function is expressed as a function of the invariants $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_3$ of the deformation gradient $\mathbf{F}$,

$$\begin{aligned} \mathbf{I}_1 &= \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \\ \mathbf{I}_2 &= \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_1^2\lambda_3^2 \\ \mathbf{I}_3 &= \lambda_1^2\lambda_2^2\lambda_3^2, \end{aligned} \tag{5}$$

and the deformation gradient $\mathbf{F}$ is a function of the strain $\mathbf{F} = \mathbf{F}(\varepsilon)$. One general energy function for this class of incompressible materials, proposed by Rivlin [36], is

$$\boldsymbol{\Psi}_{\mathbf{R}} = \sum_{i,j=0}^{\infty} \mathbf{C}_{ij}(\mathbf{I}_1 - 3)^i(\mathbf{I}_2 - 3)^j, \tag{6}$$

where $\mathbf{C}_{ij}$ are the material parameters. To account for volume changes, compressible forms of this class of material are proposed by adding the third invariant to the above Rivlin expression.

$$\boldsymbol{\Psi} = \boldsymbol{\Psi}_{\mathbf{R}} + \boldsymbol{\Psi}(\mathbf{J}), \tag{7}$$

where $\mathbf{J}$ is the volume ratio $\mathbf{J} = \sqrt{\mathbf{I}_3}$. We refer interested readers to the brief tutorial, provided as a supplementary document, for more detail.

Mooney-Rivlin material model is widely known for its accuracy in modeling this property; we use this model in our implementation because of its popularity and wide adoption in both medical and engineering applications. In this paper, we use this form of the energy function of Mooney-Rivlin material model [36], [37]:

$$\boldsymbol{\Psi} = \frac{1}{2}\mu_1\left(\left(\mathbf{I}_1^2 - \mathbf{I}_2\right)/\mathbf{I}_3^{\frac{2}{3}} - 6\right) + \mu_2\left(\mathbf{I}_1/\mathbf{I}_3^{\frac{1}{3}} - 3\right) + v_1\left(\mathbf{I}_3^{\frac{1}{2}} - 1\right)^2, \tag{8}$$

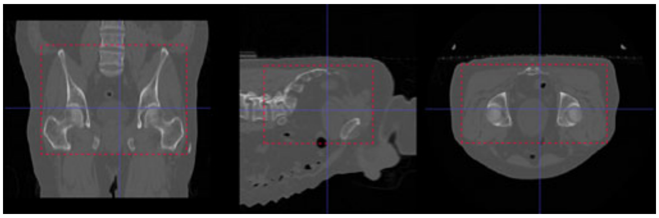

Fig. 2. **The CT image of male pelvic area.** The red dotted lines are the boundary of our model reconstruction.

where μ_1, μ_2 and v_1 are the material parameters. The first two elasticity parameters, μ_1 and μ_2, are related to the distortional response (i.e., together they describe the response of the material when subject to shear stress, uniaxial stress, and equibiaxial stress), while the last parameter, v_1, is related to volumetric response (i.e., it describes the material response to bulk stress). $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_3$ are the three invariants.

Incompressibility: In our simulation, we model abdominal organs as incompressible material [38]. There are several options for achieving incompressibility: One can add constraints to the governing equation to ensure that the determinant of the jacobian $\mathbf{J}$ of the deformation gradient $\mathbf{F}$ is equal to one. Alternatively, one can use the third material parameter (v_1 in Eq. (8)) to approximate incompressibility. To achieve incompressibility, we choose a fairly large v_1; this means v_1 will not be optimized. In order to accurately describe the material, we reconstruct both μ_1 and μ_2.

3.2.3 The Boundary Condition

The boundary condition is critical in solving Eq. (2). The boundary condition can be either the displacement field or the tractions on the boundary. Our target applications for this work include both medical applications and sketch-driven animation; for medical applications, we use the contact force between the organ and the surrounding tissue as the boundary condition. To compute the contact force we make two assumptions:

1) We simulate the surrounding tissue using a linear material model.
2) We know the (default) elasticity parameters for the surrounding tissue.

During the model reconstruction step, we include the surrounding soft tissue of the prostate, as well as the bones of the male pelvis area (as shown in Fig. 2). We simulate the surrounding tissue using a linear material model. This assumption is valid because the volume of the surrounding tissue is far larger than that of the target organ, so the amount of strain $\frac{\partial \mathbf{u}_s}{\partial \mathbf{X}_s}$ can be considered a small strain. The displacement of the surface of the surrounding tissue will populate the displacement field $\mathbf{u}_s$. For the second assumption, we set the elasticity parameters of the surrounding tissue to a default value. Then the elasticity parameters of the target organ become *relative* values with respect to the surrounding tissue. The second assumption is necessary for several reasons:

1) It is almost impossible to assess the elasticity properties of the tissue surrounding the target organ *in vivo*;
2) without the boundary condition we cannot accurately solve the governing equation, Eq. (2); and

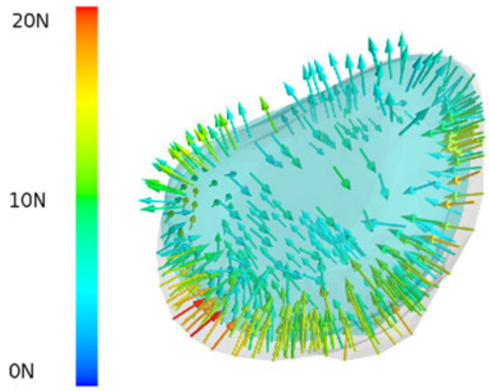

Fig. 3. **Reconstructed contact force of one patient's prostate.** The light colored transparent surface is the reference mesh; the nontransparent surface is the initial surface.

3) the relative material properties of the target organ have already proven to be useful for cancer detection [39].

Given the displacement field $\mathbf{u}_s$ of the surrounding tissue, we compute the contact force using the following equation:

$$\mathbf{K}\mathbf{u}_s = \mathbf{f}, \tag{9}$$

where $\mathbf{K}$ is the stiffness matrix of the surrounding tissue (whose elasticity parameters are known), and $\mathbf{f}$ is the resulting contact force. The FEM domain for the computation consists of elements belonging only to the surrounding tissue. An example of the reconstructed contact force is shown in Fig. 3.

3.2.4 Distance-Based Objective Function

Since our simulation framework requires only two three-dimensional surface models of organs, our error metric is constructed using the Hausdorff distance $\mathbf{d}_H$ between the deformed organ surface $\mathbf{S}_l$ and the target reference organ surface $\mathbf{S}_t$. The deformed organ surface is the output of the forward simulation. The one-sided Hausdorff distance of two sets of points $\mathbf{A}$ and $\mathbf{B}$ is defined as

$$\mathbf{d}_H(\mathbf{A}, \mathbf{B}) = \max_{a \in \mathbf{A}} \min_{b \in \mathbf{B}} d(a, b), \tag{10}$$

where set $\mathbf{A}$ represents points $\mathbf{v}_l$ of the deformed organ surface $\mathbf{S}_l$ and set $\mathbf{B}$ represents the points on the target reference organ surface $\mathbf{S}_t$.

Given this definition of the one-sided Hausdorff distance, our surface distance metric is given as

$$\Phi(\boldsymbol{\mu}) = \sum_{\mathbf{v}_l \in \mathbf{S}_l} \|\mathbf{d}_H(\mathbf{v}_l, \mathbf{S}_t)\|^2. \tag{11}$$

The above Eq. (11) will be the objective function for this optimization problem. The optimization problem is thus

$$\boldsymbol{\mu} = \operatorname*{argmin}_{\boldsymbol{\mu}} \sum_{\mathbf{v}_l \in \mathbf{S}_l} \|\mathbf{d}_H(\mathbf{v}_l, \mathbf{S}_t)\|^2, \tag{12}$$

where $\boldsymbol{\mu}$ is the material parameter vector. The $\boldsymbol{\mu}$ that minimizes the objective function is the optimized elasticity parameter vector.

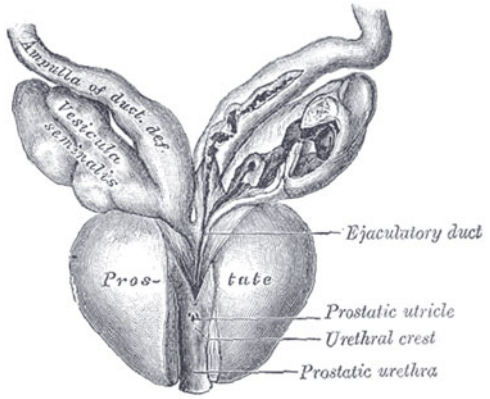

Fig. 4. **The prostatic urethra.** The prostatic urethra naturally divides the prostate into two parts. © Wikipedia [40]

3.2.5 Multi-Region Elasticity Parameter Estimation

Multi-region elasticity parameter reconstruction is much more complicated than single-body reconstruction. We consider two options for multi-region parameter reconstruction: 1) Alternate the parameter reconstruction process between the regions, or 2) simultaneously reconstruct the elasticity parameters for all regions. We adopt the second method in this paper, since our early experiments showed simultaneous reconstruction method to have a better convergence rate than the alternating method. Questions still remain regarding several elements of the process: 1) How to define the regions; 2) how to simulate the regions; and 3) how to define the objective function.

In the case of prostate, we define the regions under the guidance of physicians; in general, this step can be left to the users with knowledge in the target applications. In our examples, the prostate is naturally divided into two parts by the prostatic urethra, as shown in Fig. 4. Our multi-region elasticity parameter estimation is aimed at determining which part of the organ of interest is stiffer and therefore more likely to have cancers. This work will assist in diagnosing cancer and in performing simulation-guided biopsy and other surgical procedures.

We choose to simulate the regions of the deformable body as one deformable body. We do this because we need to maintain the continuity of the surface of the target organ. Multi-region elasticity parameter reconstruction requires some modifications to the objective function defined in Eq. (11). In this type of multi-region reconstruction, we use the following objective function:

$$\Phi(\boldsymbol{\mu}) = \sum_{m=1}^{M} \sum_{\mathbf{v}_l \in \mathbf{S}_{m,l}} \|\mathbf{d}_H(\mathbf{v}_l, \mathbf{S}_{m,t})\|^2, \tag{13}$$

with M as the total number of regions, $\mathbf{S}_{m,l}$ as the deformed surface of the m^{th} region, and $\mathbf{S}_{m,t}$ as the surface of the reference mth region. The definition of $\mathbf{S}_{m,l}$ is critical for the convergence of the reconstruction.

We exclude the nodes that are shared by other regions (as shown in Figs. 5 and 6). When the nodes shared by the regions are included in the objective function given in Eq. (13), the distance $\mathbf{d}_H(\mathbf{v}_l, \mathbf{S}_{m,t})$ may lead to optimization in the wrong direction; the decrease in the distance that is computed from these nodes' displacement fails to show

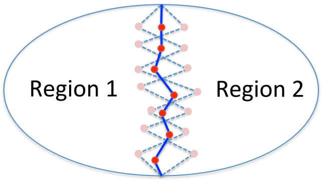

Fig. 5. **The 2D illustration of the nodes shared by two regions.** These nodes do not contribute to the convergence of the optimization.

that the optimizer is converging to the ground truth. In fact, it is possible for the distance between the nodes and the reference surface to decrease while the optimizer is *diverging* from the ground truth when nodes shared between regions are used in the computation. We address this issue by including only the original vertices on the surface of the object, not those on the shared boundary of two regions, as the $\mathbf{v}_l$ in Eq. (13). With this approach, our experiments indicate that the multi-region parameter estimation can converge to the right parameters for each region simultaneously.

3.2.6 The Inverse Step

This step estimates the recovered elasticity parameters of the target organ (or tissue). It determines the accuracy of the estimated parameter by measuring the Hausdorff distance between the (model) surface of the reference organ and that of the deformed organ, using the displacement computed by the forward simulation.

Particle Swarm Optimization: A critical part in the inverse step is the optimization process. After experimenting with several techniques, we have adopted a variant of the Particle Swarm Optimization method [41], [42], [43], a population-based stochastic optimization method. This variant of the PSO method has the following advantages:

1) It can cope with a noisy objective function that has many local minima;
2) it does not need to compute the gradient of the objective function; and
3) it is easy to parallelize the state updates of each particle.

Each particle in the PSO method corresponds to a state in the optimization process, and each particle possesses five attributes: the position, the velocity, the fitness value, the previous best position of itself, and the previous best position of the entire particle swarm. We use the subscript i to index the particle in the swarm, $\mathbf{p}_i$ to represent the previous best position of the ith particle, and $\mathbf{p}_{g,i}$ to represent the previous best position of its neighbors. Superscript t denotes the current iteration. Position will be a vector of N dimension represented as $\boldsymbol{\mu}_i^t$, and $\mathbf{v}_i^t$ is the velocity of the ith particle in the current iteration; y_i^t (scalar value) represents the fitness value of the ith particle in the current iteration. The swarm size is M, which usually ranges from 10 to 100. The dimension N of the particle position and velocity is the same as the dimensionality of the optimization problem space. The five attributes of the ith particle at the tth optimization iteration can then be defined as:

1) The position $\boldsymbol{\mu}_i^t = (\mu_{1,i}^t, \ldots, \mu_{n,i}^t, \ldots, \mu_{N,i}^t)$, with $\boldsymbol{\mu}_i^t \in \mathbf{H}$, $1 \le n \le N$, $1 \le i \le M$
2) The velocity $\mathbf{v}_i^t = (v_{1,i}^t, \ldots, v_{n,i}^t, \ldots, v_{N,i}^t)$, with $v_{\min} \le \|\mathbf{v}_i^t\| \le v_{\max}$, $1 \le n \le N$, $1 \le i \le M$
3) The fitness value $y_i^t = \Phi(\boldsymbol{\mu}_i^t)$, with $\Phi()$ as the fitness function or the objective function of the optimization problem
4) The previous best position of itself $\mathbf{p}_i^t = (p_{1,i}^t, \ldots, p_{n,i}^t, \ldots, p_{N,i}^t)$, with $\mathbf{p}_i^t \in \mathbf{H}$, $1 \le n \le N$, $1 \le i \le M$
5) The previous best position of its neighbors $\mathbf{p}_{g,i}^t = (p_{g,1,i}^t, \ldots, p_{g,n,i}^t, \ldots, p_{g,N,i}^t)$, with $\mathbf{p}_{g,i}^t \in \mathbf{H}$, $1 \le n \le N$, $1 \le i \le M$

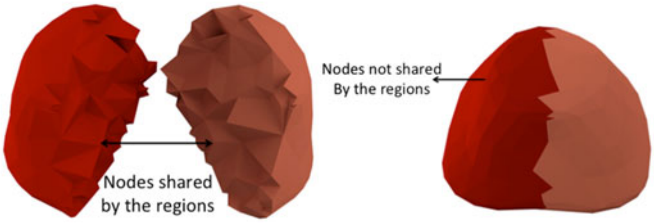

Fig. 6. **The 3D illustration of the nodes of regions.** The figure on the left shows nodes shared by the two regions; the figure on the right shows the nodes not shared by the two regions.

The position is a point in the Euclidean search space $\mathbf{H}$ of the optimization problem. In our problem, the search space is the range of all possible elasticity parameters. The number of parameters we are recovering is the dimension of the search space of the optimization. In other words, the positions of the particles are a set of the parameters we want to estimate. The fitness value is computed from the fitness function Φ, which is the objective function of the optimization problem (Eq. (11)). The velocity represents the search direction. The previous best position of the particle itself is the best set of parameters this particle has found so far. And the previous best position of its neighbors is the best set of parameters the neighbors of particles has found. The current position, the current velocity, the fitness value, the previous best position itself, and the previous best position of the swarm will be used to compute the velocity or the search direction.

Instead of optimizing one estimated solution at a time, a number of the particles are used together to collectively search for the best solution to the optimization problem (i.e., multiple coordinated searches going on simultaneously). Intuitively, the "particle swarm" will not only accelerate the search for the best solution, but will also increase the probability of finding the globally optimal solution.

The PSO method works by iteratively updating the particles' properties. The **canonical** PSO method uses the following two equations to update particle position and velocity, with *Rand*() as a random value generator.

$$\mathbf{v}_i^{(t+1)} = \mathbf{v}_i^t + Rand()(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t) + Rand()(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t), \tag{14}$$

$$\boldsymbol{\mu}_i^{(t+1)} = \boldsymbol{\mu}_i^t + \mathbf{v}_i^{t+1}\Delta t, \tag{15}$$

with $\Delta t = 1$. The particle's position update (Eq. (14), Eq. (15)) can be computed since it is determined by several factors: the current position $\boldsymbol{\mu}_i^t$; the persistence in the previous direction (first part of Eq. (14)); the influence of the previous best position of itself (second part of Eq. (14)); and the influence from its neighbors (third part of Eq. (14)). Two questions remain:

1) How to define the neighbors (how to compute $\mathbf{p}_{g,i}^t$), and
2) how to weigh the particle's persistence in its previous direction and the influence from its neighbors (i.e., how to refine Eq. (14)).

The design of appropriate PSO variants mainly focuses on these two issues.

Population Structure: The population structure of the swarm affects the convergence rate, as the structure determines how fast information propagates inside the swarm. If each particle in the swarm is informed by every other particle, the influence term in Eq. (14) will be same for every particle, meaning that all particles will move in similar directions. This makes it easy for the swarm to become entrapped in a local minimum. But if each particle in the swarm is only informed by one or two other swarm particles, the influence of other particles will be small. But if the particles are not informed enough, this slows down the process of finding the best solution. Therefore, the way that information is communicated from particle to particle can be crucial. Various kinds of swarm topology, or population structures [44], have been studied. The canonical particle swarm optimization (Eq. (14), Eq. (15)) uses the global best solution so far; it connects every particle with every other particle. After some experiments, we chose instead to use the adaptive random structure [45]. With this adaptive scheme, after every unsuccessful iteration, the neighbors of the particle i changes to K random neighbors. This adaptive random structure keeps the particles informed about different neighbors at every iteration. The value of K depends on the swarm size M and the properties of the objective function. For our problem, we chose $K = 3$ and $M = 40$ based on our experiments.

Velocity: The canonical way of updating velocity (dimension by dimension) is known to be biased [46]. Therefore, we adapted the method of computing the velocity or the search direction from [45]. For each iteration, we update the velocity of the ith particles by

$$\mathbf{v}_i^{(t+1)} = C(\mathbf{v}_i^t, \mathbf{p}_i^t - \boldsymbol{\mu}_i^t, \mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t). \tag{16}$$

The function C, denoting the velocity of the next iteration, is dependent on three terms: (a) the current velocity $\mathbf{v}_i^t$, (b) the difference between the current position $\boldsymbol{\mu}_i^t$ and the current *best position of itself* $\mathbf{p}_i^t$, and (c) the difference between the current position $\boldsymbol{\mu}_i^t$ and *the best position of its neighbors* $\mathbf{p}_{g,i}$. The best positions are the ones that result in best fitness value. For each iteration, the fitness value of each particle is evaluated to find the best position.

D being the dimension of the search space, which is denoted by a hypersphere $\mathbf{H}_i(\mathbf{G}_i^t, \|\mathbf{G}_i^t - \boldsymbol{\mu}_i^t\|)$, with the center $\mathbf{G}_i$ as the geometric center of the three points. These three points are: 1) the current position $\boldsymbol{\mu}_i^t$; 2) the point near the previous best position of it self $(\boldsymbol{\mu}_i^t + c(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t))$; and 3) the point near the previous best position of the neighbors $(\boldsymbol{\mu}_i^t + c(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t))$.

$$\mathbf{G}_i^t = \frac{\boldsymbol{\mu}_i^t + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t)) + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t))}{3}, \tag{17}$$

with c as a constant. We use $c = 1/2 + ln(2)$ [43] and we assume a radius of $r_{\max} = \|\mathbf{G}_i^t - \boldsymbol{\mu}_i^t\|$ We then randomly select a position $\boldsymbol{\mu}_i'^t$ within the space bounded by the D-sphere that has radius $r = U(0, r_{\max})$.

In addition to their known bias, the canonical velocity update methods (Eq. (14)) also result in velocity divergence. This problem was solved in paper [47] by the introduction of the inertia term ω; We follow [47] by incorporating ω into our velocity update function to solve the problem of velocity divergence. Now our velocity update function is defined as

$$\mathbf{v}_i^{(t+1)} = \omega\mathbf{v}_i^t + \boldsymbol{\mu}_i'^t - \boldsymbol{\mu}_i^t, \tag{18}$$

where the ω is a constant. We set $\omega = \frac{1}{2ln(2)}$ [43]. The update of the position of the ith particle is then computed as

$$\boldsymbol{\mu}_i^{(t+1)} = \omega\mathbf{v}_i^t + \boldsymbol{\mu}_i'^t. \tag{19}$$

If the position of the next iteration $\boldsymbol{\mu}_i^{t+1}$ is outside the current D-sphere, then the particle is "exploring" the area (searching in the unknown area); otherwise, it is "exploiting" (searching within the known area). Allowing the particles to "explore" and "exploit" can effectively avoid the problem of noise and local minima affecting the results. The pseudo code for our system work flow is provided in Algorithm. 1.

3.3 Sensitivity Analysis

In the sensitivity analysis, we aim to identify the uncertainties in the output of the system that come from variation in the input. The input to our optimization-simulation-identification framework is the amount of tissue deformation inferred from two sets of images, while the output is the estimated elasticity parameters. We wish to identify the relation between the changes in the input, i.e. the deformation, and the output, the elasticity parameters. Mathematically, we take the derivative of the elasticity parameters on both sides of the objective function (Eq. (11)):

$$\frac{\partial\Phi(\boldsymbol{\mu})}{\partial\boldsymbol{\mu}} = \frac{\partial\Phi}{\partial\mathbf{d}_H(\mathbf{u})}\frac{\partial\mathbf{d}_H(\mathbf{u})}{\partial\mathbf{u}}\frac{\partial\mathbf{u}}{\partial\boldsymbol{\mu}}. \tag{20}$$

Our study therefore focuses on testing the sensitivity of the deformation with respect to the elasticity parameters of the Mooney-Rivlin nonlinear material model, given fixed external forces. Specifically, we study the μ_1 and μ_2 in Eq. (8). Since μ_1 and μ_2 are related to distortional response, to simplify the test without losing generality we set μ_1 and μ_2 to the same value for the simulation of isotropic homogeneous deformable bodies. We will use μ to represent both μ_1 and μ_2 in the following text. To demonstrate the advantages of the nonlinear material model over the linear material model for large deformations, we also compare the results with those produced by linear material model.

We design the model of our test as a simple sphere embedded in a cube. The entire domain has been tetrahedralized using TetGen [19]. The sliced view of the model is shown in Fig. 7. Forces are applied on the surface of the sphere. The sphere will deform to an equilibrium state; the surface of the cube will be fixed. In this study we will use *relative* elasticity parameter instead of absolute values for

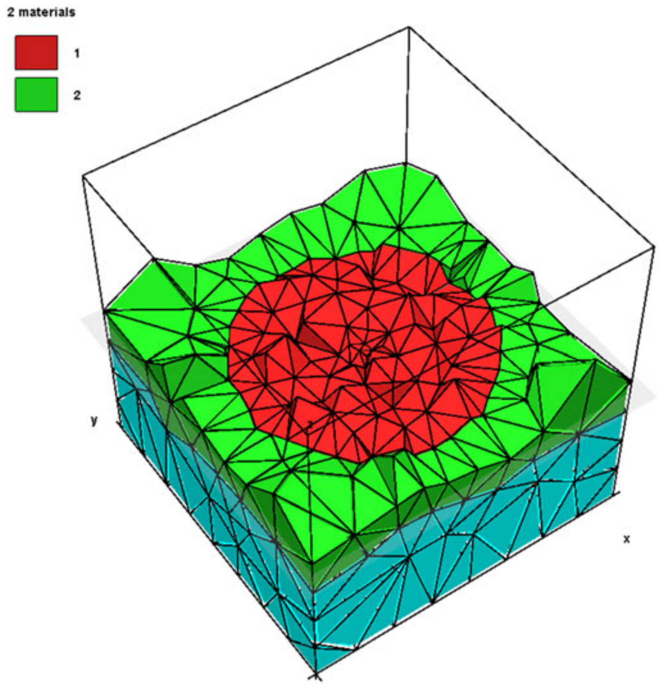

Fig. 7. **Sliced view of the model used for sensitivity analysis.**

two reasons: (a) comparison (in order to produce the same amount of deformation, the value of the elasticity parameters differ in linear and nonlinear material models), and (b) consistency (in application, we can usually assume default values for the elasticity parameters of the tissue surrounding the organ of interest).

Our analysis studies two types of material models: incompressible linear models and nonlinear models. To test the incompressibility of the material model, external forces are applied to only some of the nodes on the sphere surface, deforming it to an ellipsoid; the ratio of the length of the major axis and the minor axis will be used as the measurement of the amount of deformation. To ensure that the resulted system will be in equilibrium, the sum of the external forces is set to zero. Note that we also fixed the elasticity parameters of the area between the surface of the sphere and the cube.

We also study the difference in material behavior between the linear material model and the nonlinear material model. To make a fair comparison, we start with the set of elasticity parameters that will result in the same amount of deformation when the same amount of external force is applied to the model. We first set the elasticity parameters of the sphere and the surrounding area to this same set of values. Then we change the elasticity parameters of the sphere and record the deformation of the sphere.

To show the result, we plot the *relative* elasticity parameters against the amount of deformation, which is computed as the ratio of the ellipsoid major axis and the minor axis for the incompressible material model in Fig. 8. Fig. 8a shows that the value of the slope for the nonlinear material curve is high when the elasticity value is small. The slope of the curves in Fig. 8a is in fact $\frac{\partial \mathbf{u}}{\partial \mu}$. When the value of $\frac{\partial \mathbf{u}}{\partial \mu}$ is high, large variation or uncertainty in the input does not lead to large errors in the output. This implies that our optimization framework is less prone to uncertainties and numerical errors when the elasticity parameters are small; when the value of the elasticity parameter is large, the parameter value can still be recovered but its accuracy may be lower in comparison to the accuracy of lower parameter values. For the target medical applications, tumor tissues generally tend to be much stiffer and to have a much broader range of parameter values; therefore, comparably lower accuracy in this range poses little problem for the screening and diagnosis of cancer (see Fig. 13), for surgical procedures, or for haptic rendering.

Algorithm 1. Static Elasticity Parameter Estimation Method

```
procedure ForwardSimulation ($\mathbf{u}$, $\boldsymbol{\mu}_k$)
    $\mathbf{u} + \Delta\mathbf{u} \leftarrow \mathbf{f}(\mathbf{u}, \boldsymbol{\mu}_k)$ // fem
    return $\mathbf{u}_k$
procedure ObjectiveFunctionEvaluation ($\mathbf{u}$,$\boldsymbol{\mu}$)
    $\mathbf{u} + \Delta\mathbf{u} \leftarrow$ ForwardSimulation $\mathbf{u}, \boldsymbol{\mu}$
    $y \leftarrow \Phi(\mathbf{u}, \mathbf{u} + \Delta\mathbf{u}, \boldsymbol{\mu})$ // refer to Eq.(14)
    return $y$
procedure ParticleSwarmOptimization ($\mathbf{u}$)
    Step 1: Initialize
    for all the particles do
        $\boldsymbol{\mu}_i^0 \leftarrow$ randomly selected position from search space $\mathbf{H}$ with uniform distribution
        $\mathbf{v}_i^0 \leftarrow$ randomly selected vector with the length not larger than the radius of the search space $\mathbf{H}$
        $\mathbf{p}_i^0 \leftarrow \boldsymbol{\mu}_i^0$
        $\mathbf{p}_{g,i}^0 \leftarrow \boldsymbol{\mu}_i^0$
        $y_i^0 \leftarrow 0$
    close;

    Step 2: Iterate
    while not converged do
        for all the particles
            $y_i^t \leftarrow$ ObjectiveFunctionEvaluation $(\mathbf{u}, \boldsymbol{\mu}_i^t)$ // evaluate fitness function
            $\mathbf{G}_i^t \leftarrow \frac{\boldsymbol{\mu}_i^t+(\boldsymbol{\mu}_i^t+c(\mathbf{p}_i^t-\boldsymbol{\mu}_i^t))+(\boldsymbol{\mu}_i^t+c(\mathbf{p}_{g,i}^t-\boldsymbol{\mu}_i^t))}{3}$. // evaluate the center of the D-sphere
            $\boldsymbol{\mu}_i^{\prime t} \leftarrow$ randomly picked point from the $D$-sphere with uniform distribution
            $\mathbf{v}_i^{t+1} \leftarrow \omega\mathbf{v}_i^t + \boldsymbol{\mu}_i^{\prime t} - \boldsymbol{\mu}_i^t$ // update velocity
            $\boldsymbol{\mu}_i^{t+1} \leftarrow \boldsymbol{\mu}_i^t + \mathbf{v}_i^{t+1}$ // update position
            IF $y_i^t$ better than previous best fitness value then
                $\mathbf{p}_i^{t+1} = \boldsymbol{\mu}_i^t$
            else
                $\mathbf{p}_i^{t+1} = \mathbf{p}_i^t$
            close;
        close;
        for all the particles
            Iterate the neighbors update $\mathbf{p}_{g,i}^{t+1} \leftarrow$ the best $\mathbf{p}_i^{t+1}$
        close;
    close;
    return $\boldsymbol{\mu}$
procedure Main
    while not converged do
        ParticleSwarmOptimization ($\mathbf{u}$)
    close;
close;
```

Furthermore, in both Fig. 8a and Fig. 8b, we observe that the slope of the blue line is almost always larger than that of

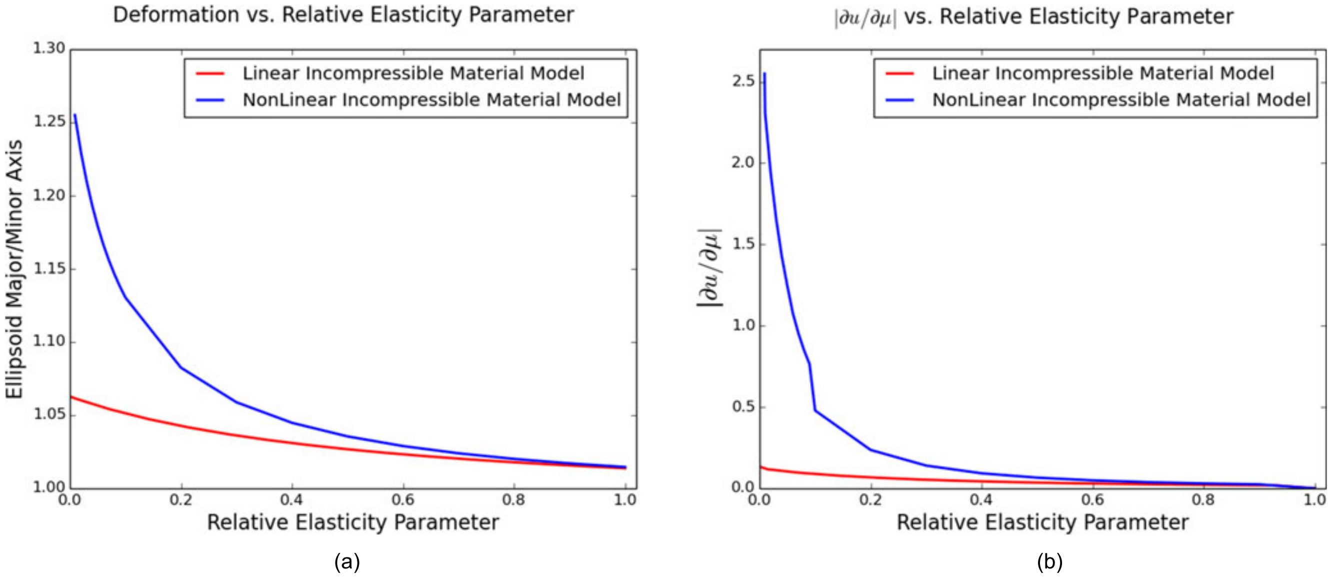

Fig. 8. **The sensitivity analysis results.** (a) shows the relation between amount of deformation represented by the inner sphere's radius changes versus the relative elasticity parameter. (b) shows the relation between the change in deformation per change in the relative elasticity parameter versus the relative elasticity parameter.

the red line (i.e., the blue line lies above the red line), as shown in Fig. 8a and in Fig. 8b. These results show that, compared to the linear material model, the nonlinear material model is more robust and less prone to error from variation/uncertainty in the input, and is thus better for medical applications.

4 Results

We have implemented our algorithms in C/C++ and have validated the results by 1) using a synthetic dataset with known elasticity values; 2) determining the correlation between the elasticity values extracted from the patients' medical (CT) images and their cancer stages, thereby indirectly validating the results of our approach on a real-world (live) patient dataset (we validated indirectly rather than by direct force measurements because 3D elastography cannot be performed *non-invasively* to obtain elasticity values *in vivo*); and 3) elasticity parameter estimation based on a sequence of 2D sketches. We demonstrate the application of our algorithm on two scenarios. The first is a 3D interaction with a virtual liver for surgical simulation; the second is a physics-based animation of letters spelling out 'AROMA' based on a user's conceptualized sketches as keyframes. From these drawings, we automatically estimate the elasticity parameters and recreate the desired deformation using the MaterialCloning algorithm. Please see the supplementary video for demonstration of these applications.

4.1 Multi-Region Elasticity-Parameter Reconstruction

The experiment is designed to validate our multi-region, elasticity-parameter reconstruction using synthetic data. In this experiment we divide the organ of interest into two regions; one of the regions contains an embedded tumor with high elasticity parameters (Fig. 9). We validate our method by recovering the elasticity parameters for the two different regions; the region with the embedded tumor should have higher elasticity parameters than the normal region. According to data from surgical experiments on human tissues, the two material parameters, μ_1 and μ_2, tend to be close [48]. Therefore, the average value of the two material parameters is used in the following studies. We test the accuracy and the robustness of our framework by varying the size of the tumor, as shown in Fig. 9.

In this experiment we assign the tumor elasticity parameters a value of $70\,kPa$; the rest of the elements are assigned $30\,kPa$. We expect the recovered elasticity parameter for the normal tissue region to be close to $30\,kPa$ and the recovered elasticity parameter for the tumor region to correlate with the tumor size. Results in Table 1 show that the recovered elasticity parameter of the region with the tumor almost linearly correlates with the size of the tumor: the linear correlation coefficient is 0.9659. The relative error for the normal tissue region is much less than 5 percent. An example optimization process for the normal tissue region is shown in Fig. 10.

We further validate our multi-region elasticity parameter reconstruction scheme by varying the tumor's elasticity parameters while keeping its size fixed. The tumor for this set of experiments occupies about 64 percent of the entire region on the left side of Fig. 9. The recovered value, shown in Table 2, has a high linear correlation coefficient of 0.9856.

One possible source of error in our multi-region elasticity parameter reconstruction scheme comes from variation in

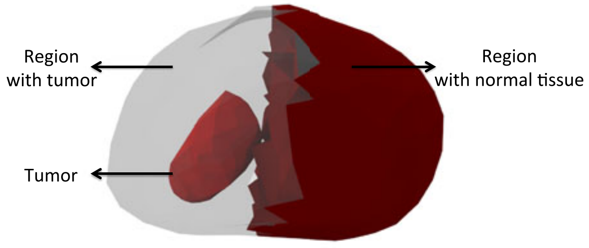

Fig. 9. **Different regions of the organ.** left with a tumor embedded; right with normal tissue.

TABLE 1
Multi-Region Elasticity Parameter Reconstruction

Tumor-to-Region ratio	0.022	0.14	0.30	0.49	0.65	0.76	0.85
Region with tumor $\bar{\mu}$ (kPa)	30.63	31.54	39.18	43.93	51.23	61.16	71.01
Region with normal tissue $\bar{\mu}$ (kPa)	29.15	28.89	31.22	30.17	31.49	29.31	30.46

As the volume ratio of tumor to the embedded region increases, so does the average stiffness value for the tumor-embedded region.

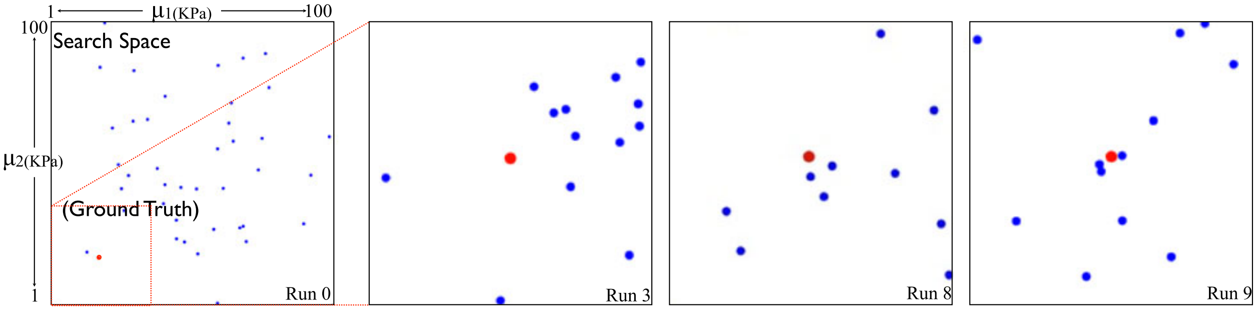

Fig. 10. **Particle-swarm optimization process:** The blue dots are the particles and the red dot signifies the ground truth.

TABLE 2
Multi-Region Elasticity Parameter Reconstruction

Tumor Elasticity Parameter (kPa)	70	140	210	280	350	420	490
Region 1 with tumor $\bar{\mu}$ (kPa)	51.23	112.92	157.44	186.78	202.22	254.20	272.58
Region 2 with normal tissue $\bar{\mu}$ (kPa)	31.49	28.28	30.04	28.56	27.62	29.61	25.18

As the the tumor becomes more stiff, the average elasticity value in the tumor region increases as well.

the mesh resolution. It is expected that finer mesh resolution will produce increased accuracy. To test this assumption, we re-run the first part of our multi-region experiment on models with different mesh resolutions. In this experiment, we vary the size of the tumor in the tumor region (shown in Fig. 9), then recover the elasticity parameter for the normal region and the tumor region using models of different mesh resolutions. In Fig. 11, we plot the relative error of elasticity parameters recovered from models with different mesh resolution against the tumor-to-region ratio. For the normal region, we set the ground truth to be $30\,kPa$. We found that varying the mesh resolutions from 1,500 nodes to 200 nodes was responsible for $1-5$ percent of the error in the recovered elasticity parameters (shown in Fig. 11). We also found that the recovered elasticity parameters tend to converge better (i.e., with less fluctuation) with higher mesh resolution. As Fig. 12 shows, the blue and the green lines, which

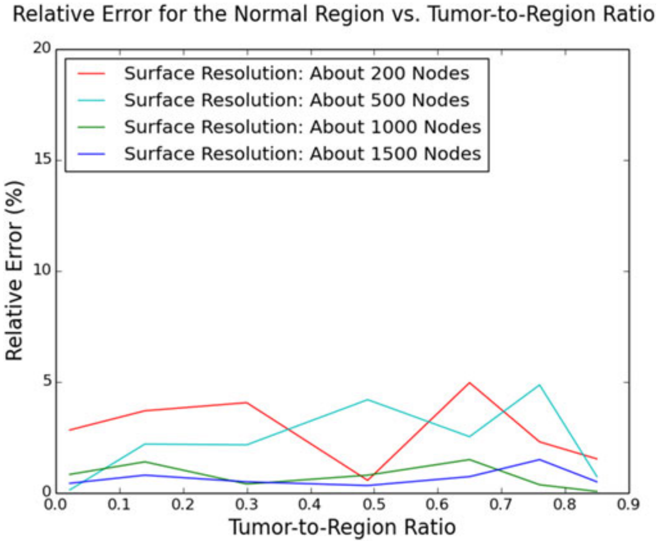

Fig. 11. **The relative error versus tumor-to-region ratio.** This figure shows the relative error of elasticity parameters for the normal region recovered using models with varying mesh resolutions plotted against the tumor-to-region ratio.

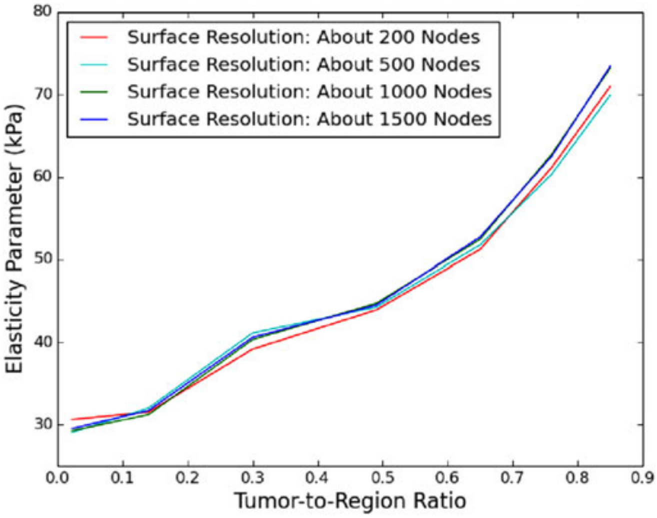

Fig. 12. **The recovered elasticity parameter versus tumor-to-region ratio.** This figure shows the recovered elasticity parameter for the tumor region using models with different mesh resolutions versus the tumor-to-region ratio.

TABLE 3
T-Stages for Prostate Cancer Definition

Stage	Definition
TX	Primary tumor cannot be assessed
T0	No evidence of primary tumor
T1	Clinically inapparent tumor by palpable or visible in imaging
T2	Tumor confined within prostate
T3	Tumor extends through the prostate capsule
T4	Tumor is fixed or invades structures other than seminal vesicles

indicate results with finer mesh resolutions, tend to be closer together; compared this with the red and the cyan lines, which plot lower mesh resolutions and diverge significantly. This study further indicates that the robustness and the convergence of our algorithm, as the resolution of the FEM meshes increases.

Other Sources of Error: We do not expect the recovered elasticity values to be completely free of errors, because error can come from multiple sources, including distance-field computation, accumulation of numerical errors from discretization, parameter dependency, etc. We focus on two aspects, namely the mesh resolution (see above) and the sensitivity analysis on parameter dependency (Section 3.3), since the other possible sources of error are similar because of discretization.

4.2 Correlating Estimated Tissue Parameters with Cancer T-Stages

We adopt the experimental protocol from [49], because it is impractical, if not impossible under existing regulations for human subject protection, to measure tissue elasticity of organs for a live patient *in vivo*. This experiment is designed to indirectly validate the effectiveness of the proposed tissue elasticity reconstruction approach in cancer staging. The T-stage, laid out in Table 3, follow the definitions in the TNM (Tumor, lymph Nodes, Metastasis) system [50], a common cancer staging system. We use ten sets of real patient data in our experimental study. The simulation scene includes the prostate, its surrounding tissue, and the bones within a male's pelvis area. The three-dimensional prostate models were reconstructed from the patient's CT images, which were taken when the patient was in two different states; the squished and the undeformed state. For the correlation experiment, we use the same elasticity parameter value for all the patients' prostate-surrounding tissue: $20\,kPa$ [51]. The result of our experiment is shown in Fig. 13. We further analyze the statistical significance of this correlation between the documented T-stages of each patient's cancer at the time of the imaging and the estimated elasticity of their prostates; we use Pearson linear correlation and Spearman correlation to compare the significance of the measured versus the estimated values. The Pearson linear correlation coefficient for the prostate's material parameters and T-stage is 0.8233, with a p-value of 0.0034. The Spearman rank correlation coefficient is 0.8304, with p-value of 0.0029. These statistics suggest a strong correlation between the prostates' elasticity parameters and their T-stages. The p-values computed from our experiments are one order smaller than that of previous work [49]; this indicates that our method is much more accurate.

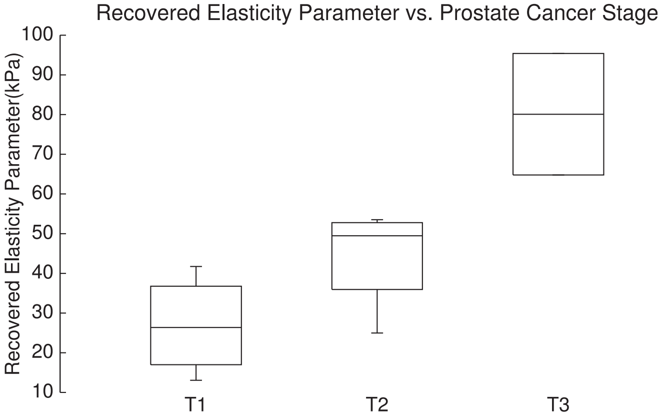

Fig. 13. **Box plot of estimated average elasticity parameters.** The estimated elasticity parameters $\bar{\mu}$ of the prostate of the ten patients versus their cancer stages shows positive correlation.

4.3 Performance Analysis for Quasi-Static Parameter Estimation

The model we use in our multi-region reconstruction validation experiment and cancer-stage correlation experiment consists of (on average) 4,000 tetrahedral elements. We run our experiment on a desktop with Intel(R) Core(TM) i7 CPU, 3.20GHz. With the Particle-Swarm Optimization method, the entire parameter reconstruction process takes no more than 24 hours on a single core. We gain nearly linear performance scaling by paralleling the PSO method using OpenMP, as shown in Fig. 14. The searching step of the PSO can be easily parallelized because each particle searches independently for the optimal solution. Parallelizing the particles' searches gives better performance. Compared to earlier elasticity parameter reconstruction methods [49], our reconstruction algorithm can be easily parallelized; it therefore achieves a performance much superior to other methods on modern parallel architectures (such as GPUs, many-core processors, etc).

4.4 Applications

We demonstrate the application of our MaterialCloning algorithm on several different scenarios:

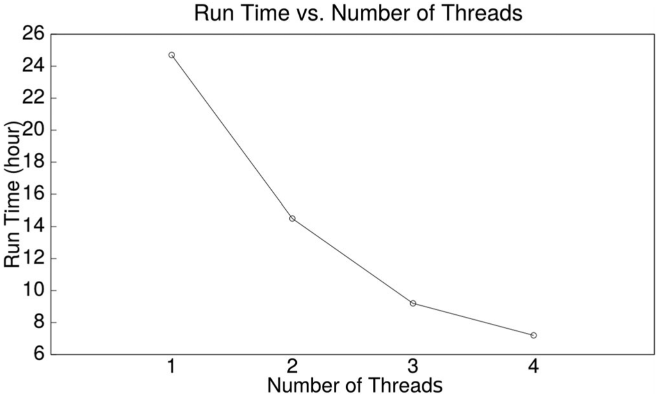

Fig. 14. **The running time of the reconstruction process versus the number of threads used.** The running time decreases almost linearly with the increase of the number of thread.

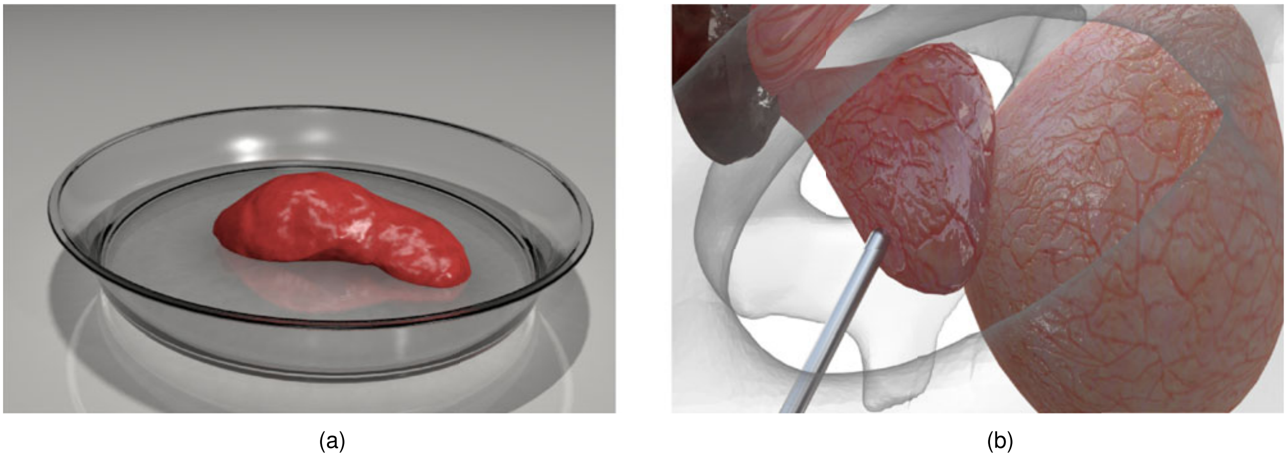

Fig. 15. **The virtual surgery application.** (a) shows the liver, with elasticity parameter reconstructed from patient data, resting on a plate. (b) is the screenshot of our virtual surgery system, using elasticity parameters for the prostate that were reconstructed from patient data.

- Dynamic simulation of a liver and a prostate, using extracted stiffness values from CT images of two different patients, dropped onto a medical dish (see Fig. 15a);
- A user poking and interacting with a 'virtual prostate' with patient-specific elasticity parameters that were automatically acquired from two sets of CT images of the patient on different days (see Fig. 15b);
- 'Animated letters' (see Fig. 16) and 'Animated figure' (see Fig. 17) are simulated from material parameters estimated directly from the artist's sketches, which serve as the keyframes to an animation sequence. Our system, combined with dynamic tracking mechanism presented in a technical report [52], can be extended to create animations from time-sequential sketches.

The visual illustrations of these examples are included in the supplementary video.

4.5 Comparison with Other Approaches

It is difficult to compare our work to existing methods on parameter estimation for soft tissues [4], [10], [11], [23], [25], as our approach does not require force-measuring devices such as force sensors or trinocular stereo vision systems. In contrast to earlier methods, our algorithm does not perform data fitting, but instead uses a coupled simulation-optimization framework to refine the estimated parameters until the optimizer converges. Instead of force measurements, we use (at least) two sets of images to reconstruct 3D geometry, then perform FEM simulation on them. The reconstructed 3D geometry and FEM meshes can introduce approximation and discretization errors, respectively. Nevertheless, our approach makes it possible to perform *non-invasive* parameter estimation directly on images in situations where 3D elastography is impossible or impractical, such as on live patients or sourced from hand drawings.

The work closest to ours is that of Lee et al. [49]. and in comparison to this work, our method shows much improved accuracy (see Section 4.2), as well as adds the ability to perform simultaneous parameter estimation for multiple materials. We use the Particle Swarm Optimization method, which is more easily parallelizable than iterative optimization methods [49] and is less prone to local minima entrapment. However, PSO is computationally more expensive and would require parallel implementation on commodity hardware (e.g., GPUs or many cores) to increase its performance. Furthermore, we demonstrate that our method can also be used to estimate elasticity parameters with the artist's sketches as keyframes to automate physics-based animation.

5 Conclusion and Future Work

In this paper, we presented the *MaterialCloning* algorithm, which can automatically acquire multi-region elasticity

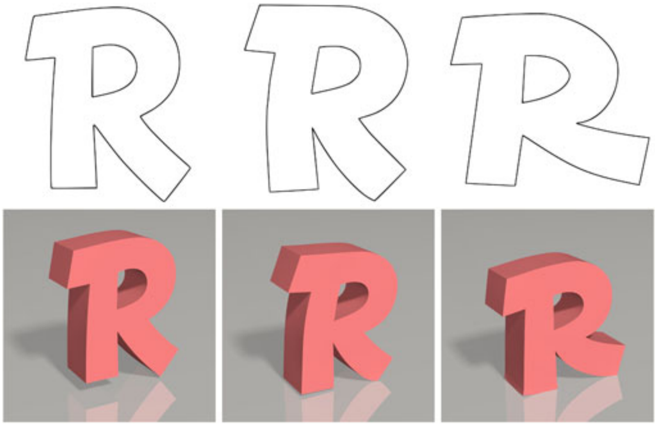

Fig. 16. **Animation from 2D sketches.** The three images in the first row are the 2D Sketches of three keyframes; the three images in the second row are the simulation result of the corresponding keyframes.

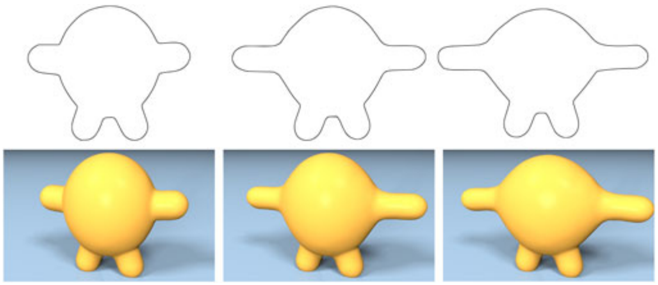

Fig. 17. **Animation from 2D sketches.** The three images in the first row are the 2D Sketches of three keyframes; the three images in the second row are the simulation result of the corresponding keyframes.

parameters directly from two sets of multiple-view images. We validate its effectiveness using both synthetic datasets and real-world medical images.

The results of our validation experiments using real-world data suggest some immediate applications: Cancer staging with only low-resolution CT images as input, computer animation of deformable models from keyframe sketches, and patient-specific surgical simulation. This method does not require any external forces to be measured; only the deformation of the body surface is needed. It can therefore be used on organs deeply seated in the human or animal body. Most importantly, this method can reconstruct elasticity parameters for multiple regions simultaneously. With this additional analysis, physicians can perform a more effective, image-guided biopsy, thereby leading to higher accuracy in cancer detection and diagnosis. We also demonstrate our algorithm on real-time interaction with virtual organs in surgical simulators, as well as on physics-based animation of virtual objects directly from the animator's conceptualized drawings.

One possible future direction is to significantly improve the algorithm's computational performance using reduced models, so that it is possible to adopt such a technique for real-time image-guided biopsy and surgery.

ACKNOWLEDGMENTS

The authors would like to thank Drs. Ronald Chen and Edward Chaney for CT images from their Lab, Dr. Huai-Ping Lee for data from his work for comparison, and Xin Zhao for assisting with the video. This work is supported in part by joint NIH/NSF SCH Program, #R01 EB020426-01.

REFERENCES

[1] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer, "Elastically deformable models," *ACM SIGGRAPH Comput. Graph.*, vol. 21, pp. 205–214, 1987.

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

[3] B. Bickel, M. Bächer, M. A. Otaduy, W. Matusik, H. Pfister, and M. Gross, "Capture and modeling of non-linear heterogeneous soft tissue," in *ACM Trans. Graph.*, vol. 28, no. 3, 2009, p. 89.

[4] B. Bickel, M. Bächer, M. A. Otaduy, H. R. Lee, H. Pfister, M. Gross, and W. Matusik, "Design and fabrication of materials with desired deformation behavior," in *ACM SIGGRAPH Papers*, 2010, pp. 1–10.

[5] H. Wang, R. Ramamoorthi, and J. O'Brien, "Data-driven elastic models for cloth: Modeling and measurement," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 71:1–71:12, 2011.

[6] E. Miguel, D. Bradley, B. Thomaszewski, B. Bickel, W. Matusik, M. Otaduy, and S. Marschner, "Data-driven estimation of cloth simulation models," *Comput. Graph. Forum*, vol. 31, no. 2, pp. 519–528, 2012.

[7] E. Miguel, R. Tamstorf, E. Bradley, S. Schvartzman, B. Thomaszewski, B. Bickel, W. Matusik, S. Marschner, and M. Otaduy, "Modeling and estimation of internal friction in cloth," *ACM Trans. Graph.*, vol. 32, no. 6, p. 212, 2013.

[8] Z. Ren, H. Yeh, and M. C. Lin, "Example-guided physically based modal sound synthesis," *ACM Trans. Graph.*, vol. 32, no. 1, p. 1, 2013.

[9] K. S. Bhat, C. D. Twigg, J. K. Hodgins, P. K. Khosla, Z. Popovic, and S. M. Seitz, "Estimating cloth simulation parameters from video," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation*, 2003, pp. 37–51.

[10] D. K. Pai, K. v. d. Doel, D. L. James, J. Lang, J. E. Lloyd, J. L. Richmond, and S. H. Yau, "Scanning physical interaction behavior of 3D objects," in *Proc. 28th Annu. Conf. Comput. Graph. Interactive Techn.*, 2001, p. 87–96.

[11] B. Bickel, M. Bächer, M. A. Otaduy, W. Matusik, H. Pfister, and M. Gross. (2009). Capture and modeling of non-linear heterogeneous soft tissue, in *Proc. ACM SIGGRAPH 2009 Papers* [Online]. pp. 1–9. Available: http://portal.acm.org/citation.cfm?id=1531395

[12] J. Ophir, S. Alam, B. Garra, F. Kallel, E. Konofagou, T. Krouskop, and T. Varghese, "Elastography: Ultrasonic estimation and imaging of the elastic properties of tissues," *Proc. Institution Mech. Eng., Part H: J. Eng. Med.*, vol. 213, no. 3, pp. 203–233, Jan. 1999.

[13] Y. Zhu, T. Hall, and J. Jiang, "A finite-element approach for young's modulus reconstruction," *IEEE Trans. Med. Imag.*, vol. 22, no. 7, pp. 890–901, Jul. 2003.

[14] F. Kallel and M. Bertrand, "Tissue elasticity reconstruction using linear perturbation method," *IEEE Trans. Med. Imag.*, vol. 15, no. 3, pp. 299–313, Jun. 1996.

[15] T. Moons, L. Van Gool, and M. Vergauwen, "3D reconstruction from multiple images," *Found. Trends Comput. Graph. Vis.*, vol. 4, no. 4, pp. 287–404, 2010.

[16] P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, and G. Gerig, "User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability," *NeuroImage*, vol. 31, no. 3, pp. 1116–1128, Jul. 2006.

[17] N. Snavely, M. S. Seitz, and R. Szeliski, Photo tourism: exploring photo collections in 3D, *ACM Tran. on Graph. (TOG)* "Bundler v0.9," vol. 25, no. 3, pp. 835–846. 2006.

[18] C. Wu, "VisualSFM: A visual structure from motion system," 2011.

[19] H. Si, "TetGen, a Delaunay-based quality tetrahedral mesh generator," *ACM Trans. on Math. Soft. (TOMS)*, vol. 41, no. 2, p. 11, 2015.

[20] M. Müller and M. Gross. (2004). Interactive virtual materials, in *Proc. Graph. Interface*, School of Computer Science, Univ. Waterloo, Waterloo, ON, Canada: Canadian Human-Computer Communications Society [Online]. pp. 239–246. Available: http://dl.acm.org/citation.cfm?id=1006058.1006087

[21] M. Teschner, B. Heidelberger, D. Manocha, N. Govindaraju, G. Zachmann, S. Kimmerle, J. Mezger, and A. Fuhrmann, "Collision handling in dynamic simulation environments: The evolution of graphics: Where to next?" in *Proc. Eurograph. Tut.*, 2005, pp. 79–185.

[22] O. C. Zienkiewicz and R. L. Taylor, *The Finite Element Method*, 6th ed. London, U.K.: Butterworth-Heinemann, Dec. 2005.

[23] C. Syllebranque and S. Boivin, "Estimation of mechanical parameters of deformable solids from videos," *Vis. Comput.*, vol. 24, no. 11, pp. 963–972, 2008.

[24] W. H. Press, *Numerical Recipes*. Cambridge, UK: Cambridge Univ. Press, 2007.

[25] M. Becker and M. Teschner, "Robust and efficient estimation of elasticity parameters using the linear finite element method," in *Proc. Conf. Simul. Vis.*, 2007, pp. 15–28.

[26] A. Skovoroda and S. Emelianov, "Tissue elasticity reconstruction based on ultrasonic displacement and strain images," *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 42, no. 4, p. 141, Jul. 1995.

[27] H. Rivaz, E. Boctor, P. Foroughi, R. Zellars, G. Fichtinger, and G. Hager, "Ultrasound elastography: A dynamic programming approach," *IEEE Trans. Med. Imag.*, vol. 27, no. 10, pp. 1373–1377, Oct. 2008.

[28] R. Muthupillai and R. L. Ehman, "Magnetic resonance elastography," *Nat. Med.*, vol. 2, no. 5, pp. 601–603, May 1996.

[29] D. Fu, S. Levinson, S. Gracewski, and K. Parker, "Non-invasive quantitative reconstruction of tissue elasticity using an iterative forward approach," *Phys. Med. Biol.*, vol. 45, no. 6, pp. 1495–1510, 2000.

[30] S. Balocco, O. Camara, and A. F. Frangi. (2008). Towards regional elastography of intracranial aneurysms, in *Proc. 11th Int. Conf. Med. Image Comput. Comput.-Assisted Intervention* [Online]. *11*, pp. 131–138. Available: http://www.ncbi.nlm.nih.gov/pubmed/18982598

[31] D. S. Schnur and N. Zabaras, "An inverse method for determining elastic material properties and a material interface," *Int. J. Numerical Methods Eng.*, vol. 33, no. 10, pp. 2039–2057, 1992.

[32] C. W. Washington and M. I. Miga, "Modality independent elastography (MIE): A new approach to elasticity imaging," *IEEE Trans. Med. Image*, vol. 23, no. 9, pp. 1117–1128, Sep. 2004.

[33] T. Hu and J. P. Desai, "Characterization of soft-tissue material properties: large deformation analysis," in *Proc. Int. Symp. Med. Simul.*, 2004, pp. 28–37.

[34] M. E. Gurtin, *An Introduction to Continuum Mechanics*. San Francisco, CA, US: Academic, 1982.

[35] J. Teran, E. Sifakis, G. Irving, and R. Fedkiw, "Robust quasistatic finite elements and flesh simulation," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation*, 2005, pp. 181–190.

[36] L. R. Treloar, H. Hopkins, R. Rivlin, and J. Ball, "The mechanics of rubber elasticity [and discussions]," *Proc. Roy. Soc. London. A. Math. Phys. Sci.*, vol. 351, no. 1666, pp. 301–330, 1976.

[37] R. S. Rivlin and D. Saunders, "Large elastic deformations of isotropic materials. vii. experiments on the deformation of rubber," *Philosophical Trans. Roy. Soc. London. Ser. A, Math. Phys. Sci.*, vol. 243, no. 865, pp. 251–288, 1951.

[38] A. Nava, E. Mazza, F. Kleinermann, N. J. Avis, and J. McClure, "Determination of the mechanical properties of soft human tissues through aspiration experiments," in *Proc. 6th Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2003, pp. 222–229.

[39] M. Tsutsumi, T. Miyagawa, T. Matsumura, N. Kawazoe, S. Ishikawa, T. Shimokama, T. Shiina, N. Miyanaga, and H. Akaza, "The impact of real-time tissue elasticity imaging (elastography) on the detection of prostate cancer: Clinicopathological analysis," *Int. J. Clinical Oncol.*, vol. 12, no. 4, pp. 250–255, 2007.

[40] H. Gray. (1918). [Online]. Available: http://en.wikipedia.org/wiki/Prostatic_urethra

[41] J. Kennedy, R. Eberhart, et al., "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.

[42] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, 2007.

[43] M. Clerc, *Particle Swarm Optimization*, vol. 93. New York, NY, US: Wiley, 2010.

[44] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," 2002.

[45] M. Clerc, "Standard particle swarm optimisation," 2012.

[46] C. K. Monson and K. D. Seppi, "Exposing origin-seeking bias in PSO," in *Proc. Conf. Genet. Evol. Comput.*, 2005, pp. 241–248.

[47] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. 7th Int. Conf. Evol. Program. VII*, 1998, pp. 591–600.

[48] D. Tang, C. Yang, S. Kobayashi, and D. N. Ku, "Steady flow and wall compression in stenotic arteries: A three-dimensional thick-wall model with fluid–wall interactions," *J. Biomech. Eng.*, vol. 123, no. 6, pp. 548–557, 2001.

[49] H.-P. Lee, M. Foskey, M. Niethammer, P. Krajcevski, and M. C. Lin, "Simulation-based joint estimation of body deformation and elasticity parameters for medical image analysis," *IEEE Trans. Med. Imag.*,vol. 31, no. 11, pp. 2156–2168, Nov. 2012.

[50] L. H. Sobin, M. K. Gospodarowicz, and C. Wittekind, *TNM Classification of Malignant Tumours*. New York, NY, US: Wiley, 2011.

[51] J. Rosen, J. D. Brown, M. Sinanan, B. Hannaford, and S. De, "Biomechanical properties of abdominal organs in vivo and postmortem under compression loads," *J. Biomech. Eng.*, vol. 130, no. 2, p. 021020, 2008.

[52] S. Yang and M. Lin, "Material property reconstruction from time sequential images," Dept. Comput. Sci., Univ. North Carolina, Chapel Hill, NC, US, Tech. Rep., 2015.

Shan Yang received the BE degree in computer science from the Shanghai Jiao Tong University. She is currently working toward the fourth-year PhD degree at the GAMMA Research Lab, University of North Carolina, Chapel Hill. Her research focuses on physically-based simulation.

Ming C. Lin received the BS, MS, and PhD degrees in electrical engineering and computer science from the University of California, Berkeley. She is currently the John R. & Louise S. Parker Distinguished Professor of computer science at the University of North Carolina (UNC), Chapel Hill and an honorary visiting chair professor (Yangtze Scholar) at the Tsinghua University, Beijing, China. She has received several honors and awards, including IEEE VGTC VR Technical Achievement Award 2010, and nine Best Paper Awards. Her research interests include physically-based modeling, real-time interactive 3D graphics, virtual environments, geometric modeling, and GPU-Computing. She has coauthored more than 250 refereed publications and coedited/authored four books. She was the editor-in-chief of *IEEE Transactions on Visualization and Computer Graphics*(2011-2014) and a member of six editorial boards of scientific journals. She is currently the chair of the IEEE Computer Society (CS) Transactions Operations Committee and a member of the IEEE CS Board of Governors. She also has served on the Steering and Executive Committees of conferences and Visualization & Graphics Technical Committee, as well as technical advisory boards for for the government, industry, and the international scientific research community. She is a fellow of the ACM and IEEE.