

**NON-RIGID BODY MECHANICAL PROPERTY RECOVERY FROM IMAGES
AND VIDEOS**

Shan Yang

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

**Chapel Hill
2017**

To be approved by:

Ming C. Lin

Dinesh Manocha

Tamara Berg

Vladimir Jojic

Chris Bregler

©2017
Shan Yang
ALL RIGHTS RESERVED

ABSTRACT

Shan Yang: Non-rigid Body Mechanical Property Recovery from Images and Videos
(Under the direction of Ming C. Lin)

Material property has great importance in surgical simulation and virtual reality. The mechanical properties of the human soft tissue are critical to characterize the tissue deformation of each patient. Studies have shown that the tissue stiffness described by the tissue properties may indicate abnormal pathological process. The (recovered) elasticity parameters can assist surgeons to perform better pre-op surgical planning and enable medical robots to carry out personalized surgical procedures. Traditional elasticity parameters estimation methods rely largely on known external forces measured by special devices and strain field estimated by landmarks on the deformable bodies. Or they are limited to mechanical property estimation for quasi-static deformation. For virtual reality applications such as virtual try-on, garment material capturing is of equal significance as the geometry reconstruction.

In this thesis, I present novel approaches for automatically estimating the material properties of soft bodies from images or from a video capturing the motion of the deformable body. I 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. 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 regions of soft bodies using Finite Element Method-based simulation (of either linear or nonlinear materials undergoing large deformation) and particle-swarm optimization methods. I demonstrate the effectiveness of this approach on real-time interaction with virtual organs in patient-specific surgical simulation, using parameters acquired from low-resolution medical images. With the recovered elasticity parameters and the age of the prostate cancer patients as features, I build a cancer grading and staging classifier.

The classifier achieves up to 91% for predicting cancer T-Stage and 88% for predicting Gleason score. To recover the mechanical properties of soft bodies from a video, I propose a method which couples statistical graphical model with FEM simulation. Using this method, I can recover the material properties of a soft ball from a high-speed camera video that captures the motion of the ball.

Furthermore, I extend the material recovery framework to fabric material identification. I propose a novel method for garment material extraction from a single-view image and a learning-based cloth material recovery method from a video recording the motion of the cloth. Most recent garment capturing techniques rely on acquiring multiple views of clothing, which may not always be readily available, especially in the case of pre-existing photographs from the web. As an alternative, I propose a method that can compute a 3D model of a human body and its outfit from a single photograph with little human interaction. My proposed learning-based cloth material type recovery method exploits simulated data-set and deep neural network. I demonstrate the effectiveness of my algorithms by re-purposing the reconstructed garments for virtual try-on, garment transfer, and cloth animation on digital characters. With the recovered mechanical properties, one can construct a virtual world with soft objects exhibiting real-world behaviors.

ACKNOWLEDGEMENTS

The five years' Ph.D life has been an incredible experience for me. First and foremost, I would like to thank my advisor Prof. Ming C. Lin for taking me as her Ph.D student in the first place. She has been so kind to guide, help and support me through this journey. I would also love to thank my Ph.D committee members: Prof. Dinesh Manocha for all the advice on my papers; Prof. Vladimir Jojic for opening the gate of machine learning to me; Prof. Tamara Berg for all the suggestions on my garment recovery paper; Prof. Chris Bregler for all the great ideas and support through my internship.

It is not possible to finish this journey without all the kind help and support from my all of my friends and all the gamma members.

Last but not the least, I would like to thank my parents for encouraging me to pursue my dream and giving me support all the time.

TABLE OF CONTENTS

LIST OF TABLES	xi
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xix
LIST OF SYMBOLS	xx
1 Introduction	1
1.1 Human-tissue Material Property Identification	3
1.2 Cloth Material Property Recovery	4
1.3 Thesis Statement	5
1.4 Main Results	5
1.4.1 Human-tissue Material Recovery	6
1.4.2 Cloth Material Recovery	6
1.5 Organization	6
2 Previous Work	8
2.1 Deformable Body Simulation	8
2.2 Human-Tissue Mechanical Property Recovery	9
2.2.1 Measurement-based Methods	10
2.2.2 Elastography	10
2.2.3 Inverse Finite Element Methods	11
2.2.4 Probabilistic Graphical Models	11
2.3 Cloth Mechanical Property Recovery	12
3 Multi-region Image-based Elasticity Recovery	13

3.1	Introduction	13
3.2	Method	15
3.2.1	Geometry Reconstruction and Mesh Generation	16
3.2.2	Quasi-Static Process Elasticity Parameter Estimation	17
3.2.2.1	Forward Simulation	17
3.2.2.2	Material Model	18
3.2.2.3	The Boundary Condition	20
3.2.2.4	Distance-Based Objective Function	22
3.2.2.5	Multi-Region Elasticity Parameter Estimation	23
3.2.2.6	The Inverse Step	24
3.2.3	Sensitivity Analysis	29
3.3	Experiments	33
3.3.1	Multi-Region Elasticity-Parameter Reconstruction	33
3.3.2	Correlating Estimated Tissue Parameters with Cancer T-Stages	35
3.3.3	Performance Analysis for Quasi-Static Parameter Estimation	37
3.3.4	Applications	39
3.3.5	Comparison with Other Approaches	39
3.4	Conclusion and Future Work	40
4	Video-based Soft-body Mechanical Property Recovery	42
4.1	Introduction	42
4.2	Method	44
4.2.1	Generalized Dynamic Process and Decoupled State Parameter Estimation ..	45
4.2.2	Deformable Body Modeling	45
4.2.3	Bayesian Parameter Estimation	46
4.2.4	Unscented Kalman Filter for Parameter Estimation	47
4.2.5	Coupled State Estimation	50

4.3	Experiments	50
4.3.1	Synthetic Heart Experiment	50
4.3.2	Mechanical Parameters Recovered from Videos	53
4.4	Conclusion and Future Work	56
5	Classification of Prostate Cancer Grades and T-Stages based on Tissue Elasticity Using Medical Image Analysis	58
5.1	Introduction	58
5.2	Method	59
5.2.1	Forward Simulation: BioTissue Modeling	60
5.2.2	Inverse Process: Optimization for Parameter Identification	61
5.2.3	Classification Methods	62
5.3	Results	62
5.3.1	Preprocessing and Patient Dataset	62
5.3.2	Cancer Grading/Staging Classification based on Prostate Elasticity Parameters	63
5.4	Conclusion	65
6	Single-view Image-based Garment Recovery	67
6.1	Introduction	67
6.2	Related Work	69
6.3	Method	72
6.4	Data Preparation	74
6.4.1	Data Representations	74
6.4.2	Preprocessing	75
6.4.3	Initial Garment Registration	76
6.5	Image-Guided Parameter Identification	78
6.5.1	Overview	78
6.5.2	Garment Types, Patterns, and Parameters	79
6.5.3	From Wrinkle Density to Material Property and Sizing Parameters	80

6.5.4	Optimization-based Parameter Estimation	82
6.6	Joint Material-Pose Optimization.....	83
6.6.1	Optimal Parameter Selection	83
6.6.2	Application to Image-Based Virtual Try-On	85
6.7	Results and Discussion	86
6.7.1	Garment Recovery Results	86
6.7.2	Comparison with Other Related Work.....	89
6.7.3	Performance	92
6.7.4	Discussions and Limitations	92
6.8	Conclusion and Future Work	94
7	Learning-based Cloth Material Recovery from Video	95
7.1	Introduction.....	95
7.2	Related Work	97
7.3	Method	99
7.4	Visual, Material and Motion Representation	99
7.4.1	Appearance Representation.....	99
7.4.2	Material Representation	100
7.4.2.1	Material Model	100
7.4.2.2	Parameter Space Discretization	101
7.4.3	Motion Sub-space.....	102
7.5	Learning Method	103
7.5.1	Deep Neural Network Structure	103
7.6	Physics-based Synthetic Data-sets	106
7.6.1	Data Generation	106
7.7	Experiments	107
7.7.1	Data Preparation	108

7.7.2	Results	108
7.7.2.1	Baseline Results	108
7.7.2.2	Validation of My Method	109
7.7.2.3	Comparison.....	110
7.7.3	Application.....	111
7.7.4	Discussion and Limitations	111
7.8	Conclusion and Future Work	112
8	Discussion	113
8.1	Summary of Results	113
8.2	Future Work	113
	BIBLIOGRAPHY	114

LIST OF TABLES

3.1 Multi-Region Elasticity Parameter Reconstruction	34
3.2 Multi-Region Elasticity Parameter Reconstruction	35
3.3 T-stages for prostate cancer definition	37
4.1 The result for the synthetic heart experiment with noises in the initialization. The relative error of the recovered Young's modulus is within 7.5% of the ground-truth values. Our method reduces the average surface-tracking error down to less than 5%.	54
4.2 Impact of initial velocities on recovered Young's moduli.	56
4.3 Comparison with results of (Wang et al., 2015) and experimental measurements. The measured Young's modulus for the tennis ball is taken from (Sissler et al., 2010; Wójcicki et al., 2011) and for the foam ball is derived from (Moore et al., 2007). The recovered Young's modulus using our method is within the range of the Young's modulus measured, the Young's modulus recovered using (Wang et al., 2015) is not.	56
6.1 The accuracy of the recovered sizing and material parameters. The accuracy of the recovered sizing and material parameters of the t-shirt and the pants (in percentages).	89
6.2 The accuracy of recovered garment curvature and material parameters. The accuracy of the recovered garment local mean curvature and material parameters of the skirt (in percentages).	90
6.3 Difference between the recovered human body joint angles and the ground-truth values (in Degree) for different poses shown in Fig. 6.11.	90
7.1 Material parameter sub-space validation. The floating point numbers show the estimated stretching/bending parameter coefficients (\tilde{p}, \tilde{k}) , while the numbers in the parenthesis are the corresponding stretching/bending parameter (p, k) in my defined subspace \mathcal{P}	104
7.2 Testing results. The models are trained with the arm bending motion and wind blowing motion. Then they are tested on 432 simulated arm bending/wind blowing videos, where the ground truth is known. My method achieved up to 71.8% of accuracy for predicting 54 classes of materials for arm bending motion and upto 66.7% for wind blowing motion.	109

7.3 Stiffness/density correlation r values for (Bouman et al., 2013) vs. Ours

My method outperforms both (Bouman et al., 2013) and human perception, achieving the highest correlation value of 0.77 and 0.84 respectively for stiffness and density, undergoing larger motion due to stronger wind (W3-video). 111

LIST OF FIGURES

1.1 Prostate cancer statistics (Cook, 2017).	2
1.2 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.	3
1.3 Skirts with different material types (stylewe, 2014). Skirts with different material properties show case different appearance.	4
3.1 The Flow Chart of My Framework. My framework takes (at least) two sets of images (medical images or other multi-view images) as input; I 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.	15
3.2 The CT Image of Male Pelvic Area. The red dotted lines are the boundary of my model reconstruction.	21
3.3 Reconstructed Contact Force of One Patient's Prostate. The light colored transparent surface is the reference mesh; the nontransparent surface is the initial surface.	22
3.4 The Prostatic Urethra. The prostatic urethra naturally divides the prostate into two parts. © Wikipedia(Gray, 1918)	23
3.5 The 2D and 3D Illustration of the Nodes of Regions. (a) These nodes do not contribute to the convergence of the optimization. (b) 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.	24
3.6 Sliced View of the Model Used for Sensitivity Analysis.	32
3.7 The Sensitivity Analysis Results. (a) shows the relation between amount of deformation represented by the inner sphere's radius changes vs. the relative elasticity parameter. (b) shows the relation between the change in deformation per change in the relative elasticity parameter vs. the relative elasticity parameter.	32

3.8	Different Regions of the Organ. left with a tumor embedded; right with normal tissue.....	34
3.9	Particle-Swarm Optimization Process: The blue dots are the particles and the red dot signifies the ground truth.....	35
3.10	The Relative Error vs. 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.	36
3.11	The Recovered Elasticity Parameter vs. Tumor-to-Region Ratio. This figure shows the recovered elasticity parameter for the tumor region using models with different mesh resolutions vs. the tumor-to-region ratio.	37
3.12	Box Plot of Estimated Average Elasticity Parameters. The estimated elasticity parameters $\bar{\mu}$ of the prostate of the ten patients vs. their cancer stages shows positive correlation.	38
3.13	The Running Time of the Reconstruction Process vs. the Number of Threads Used. The running time decreases almost linearly with the increase of the number of thread.....	38
3.14	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.	39
4.1	The Flow Chart of Our Framework. Our framework takes a temporal sequence of deformation samples as the input. The UKF takes in the observations and drives the finite element simulation by optimizing both the hidden parameter and hidden states.	44
4.2	The human heart anatomy. In this project I model the left and right ventricle.© Texas Heart Institute (Institute, 1996).....	52
4.3	Reconstructed left ventricle and right ventricle of human heart. (a)-(b) the slices of a human heart ultrasound image © CETUS 2014 (Bernard et al., 2014), (c) the surface mesh of the reconstructed model, (d) the sliced view of the tetrahedra mesh from the surface mesh in (b).	53
4.4	The computed contraction force and synthetic heart simulation result. (a) the visualization of the contraction force on the surface of the 3D heart model using cool to hot color map, (b) the sliced view of the contraction force. (c) the simulated heart model by the end of systole phase, (d) the simulated heart model by the end of diastole phase.	53

4.5	The convergence graphs for synthetic heart experiment. (a) shows our framework reduces the distance between the surface with the optimized material parameter and the reference surface, as the framework iterates with the initial distance error at 3.2mm, (b) shows the convergence of the Young's modulus to the ground truth.....	54
4.6	Input tennis video clips and the tracked surface mesh. (a)-(e) the clips of the tennis video © Trevor Shannon (Shannon, 2009); (f)-(j) the tracked surface mesh at the corresponding time stamp	55
4.7	Input foam ball video clips and the tracked surface mesh. (a)-(e) the clips of the tennis video © Trevor Shannon (Shannon, 2009); (f)-(j) the tracked surface mesh at the corresponding time stamp.....	55
5.1	Real Patient CT Image and Reconstructed Organ Surfaces. (a) shows one slice of the patient CT images with the bladder, prostate and rectum segmented. (b) shows the reconstructed organ surfaces.....	63
5.2	Error Distribution of Cancer Grading/Staging Classification for Per-Image Study. (a) shows error distribution of our cancer staging classification using the recovered prostate elasticity parameter and the patient's age. For our patient dataset, the multinomial classifier (shown in royal blue and sky blue) outperforms the ordinal classifier (shown in crimson and coral). I achieve up to 91% accuracy using multinomial logistic regression and 89% using ordinal logistic regression for classifying cancer T-Stage based on recovered elasticity parameter and age. (b) shows the correlation between the recovered relative elasticity parameter and the Gleason score with/without the patient's age. I achieve up to 88% accuracy using multinomial logistic regression and 81% using ordinal logistic regression for classifying Gleason score based on recovered elasticity parameter and age.	64
5.3	Error Distribution of Cancer Aggression/Staging Classification for Per-Patient Study. (a) shows the accuracy and error distribution of our recovered prostate elasticity parameter and cancer T-Stage. For our patient dataset, the multinomial classifier (shown in royal blue and sky blue) outperforms the ordinal classifier (shown in crimson and coral). I achieve up to 84% accuracy using multinomial logistic regression and 82% using ordinal logistic regression for classifying cancer T-Stage based on our recovered elasticity parameter and patient age information. (b) shows the correlation between the recovered relative elasticity parameter and the Gleason score. I achieve up to 77% accuracy using multinomial logistic regression and 70% using ordinal logistic regression for classifying Gleason score based on our recovered elasticity parameter and patient age information.	66

6.1	Garment recovery and re-purposing results. From left to right, I show an example of (a) the original image (<i>Saaclothes, 2015</i>) ©, (b) the recovered dress and body shape from a single-view image, and (c)-(e) the recovered garment on another body of different poses and shapes/sizes (<i>Hillsweddingdress, 2015</i>) ©.....	67
6.2	The flowchart of my algorithm. I take a single-view image (<i>ModCloth, 2015</i>) ©, a human-body dataset, and a garment-template database as input. I preprocess the input data by performing garment parsing, sizing and features estimation, and human-body reconstruction. Next, I recover an estimated garment described by the set of garment parameters, including fabric material, design pattern parameters, sizing and wrinkle density, as well as the registered garment dressed on the reconstructed body. Finally, I perform joint material-pose optimization and show the recovered results using cloth simulation on the virtual mannequin.	68
6.3	Template sewing pattern and parameter space of a skirt, pants, and t-shirt. (a) The classic circle sewing pattern of a skirt. (b) My parametric skirt template showing dashed lines for seams and the four parameters $\langle l_1, r_1, r_2, \alpha \rangle$, in which parameter l_1 is related to the waist girth and parameter r_2 is related to the length of the skirt. (c) The classic pants sewing pattern. (d) My parametric pants template with seven parameters $\langle w_1, w_2, w_3, w_4, h_1, h_2, h_3 \rangle$. (e) The classic t-shirt sewing pattern. (f) My parametric t-shirt template with six parameters $\langle r, w_1, w_2, h_1, h_2, l_1 \rangle$	74
6.4	Initial garment registration process. (a) The human body mesh with the skeleton joints shown as the red sphere and the skeleton of the arm shown as the cyan cylinder. (b) The initial t-shirt with the skeleton joint shown as the red sphere and the skeleton of the sleeve part of it shown as the cyan cylinder. (c) The t-shirt and the human body mesh are aligned by matching the joints. (d) The result after aligning the skeleton and removing the interpenetrations.	78
6.5	Initial garment registration results. I fit garments to human bodies with different body shapes and poses.	78
6.6	Material parameter identification results. (a) The local curvature estimation before optimizing the bending stiffness coefficients (using the cool-to-hot color map). (b) The local curvature estimation after the optimization. (c) The original garment image (<i>ModCloth, 2015</i>) ©.	80
6.7	Joint Material-Pose Optimization results. (a) The pants recovered prior to the joint optimization. (b) The recovered pants after optimizing both the pose and the material properties. The wrinkles in the knee area better match with those in the original image. (c) The original pants image (<i>Anthropologie, 2015</i>) ©.	84

6.8	Skirt and pants recovery results. I recover the partially occluded, folded skirts from the single-view images in the first, fourth (<i>ModCloth, 2015; AliExpress, 2015</i>) © and seventh (<i>Anthropologie, 2015</i>) © columns. The recovered human body meshes are shown in the second, fifth and eighth columns overlaid on the original images. The recovered skirts are shown in the third, sixth and ninth columns.	86
6.9	Garment recovery results. For the first two rows, the input image (leftmost) (<i>ModCloth, 2015; AliExpress, 2015; RedBubble, 2015</i>) © and recovered garment on the extracted human body. In the last row, the input image (leftmost) and the recovered garment on a twisted body.....	87
6.10	Image-based garment transfer results. I dress the woman in (a) (<i>FashionableShoes, 2013; Boden, 2015</i>) © with the skirt I recovered from (b) (<i>AliExpress, 2015; ModCloth, 2015</i>) ©. (c) I simulate my recovered skirt with some wind motion to animate the retargeted skirt, as shown in (d). Another example of garment transfer is given in (e) - (h).	88
6.11	Synthetic evaluation scenes. (a)-(c) fixed body shape with different poses. (d)-(e) fixed body shape with a skirt of different material. (f)-(j) same scene setup as (a)-(e) but different lighting condition.	89
6.12	Comparison. (a) One frame of the video along with (b) the CMP-MVS results before and (c) after smoothing. (d) My results using only one frame of the video.	91
6.13	Comparison. (a) One frame of the multi-view video along with (b) the CMP-MVS results before and (c) after smoothing. (d) My results using only one frame of the video.	91
6.14	Comparison. (a) input image (© 2015 ACM) from paper Chen et al. (Chen et al., 2015) Figure 12. (b) my garment recover results from only a single-view RGB image (a). (c) recovery results (© 2015 ACM) from Chen et al. (Chen et al., 2015) using both RGB image and depth information.	91
6.15	Comparison. (a) input image (© 2015 Wiley) from Figure 3 in Jeong et al. (Jeong et al., 2015). (b) my garment recover results from (a). (c) recovery results (© 2015 Wiley) from Jeong et al. (Jeong et al., 2015).	92
7.1	Learning-based cloth material prediction and material cloning results. (a) learning samples generated using the state-of-art physically-based cloth simulator Arcsim(Narain et al., 2012) (b) example real-life cloth motion videos presented in(Bouman et al., 2013) (c) simulated skirt with the material type predicted from the real-life video in (b) using the learned model from samples presented in (a).....	95

7.2	An overview of my method. My cloth material recovery method learns an appearance-to-material mapping model from a set of synthetic training samples. With the learned mapping model, I perform material-type prediction given a recorded video of cloth motion.....	96
7.3	Stretching and bending parameters sensitivity analysis results. (best view in color) The x-axis is the reciprocal of parameter ratios to the basis material. The y-axis is the maximum amount of deformation of the cloth, i.e., maximum amount of stretching or maximum curvature, respectively. I use the vertical lines with different colors to represent the 10 types of materials presented in (Wang et al., 2011a).....	103
7.4	Appearance-to-material learning method. I apply CNN and LSTM (the original LRCN design presented in (Donahue et al., 2015a)) to learn the mapping between appearance and material.....	104
7.5	The five-layer CNN structure. The original design is presented in (Krizhevsky et al., 2012)	105
7.6	Learned CNN conv5-layer activation visualization. (best view in color) I overlay the conv5 layer activation using the “jet” color map with the original image. The model is trained with my simulated data set with the cloth wind-blowing motion.	105
7.7	Data generation pipeline. My simulated data learning samples generation pipeline consists of two steps: cloth simulation and image rendering.	106
7.8	Simulated data showcase. The first three rows are example frames from my Wind-blowing data set with the cloth in pose-1. The bottom two rows are example frames from my Wind-blowing simulated data set with the cloth in pose-2 consisting of two different types of material.	107
7.9	Material cloning results. The first column are the input cloth motion videos(Bouman et al., 2013). I predict the material type of the cloth in these input videos and clone those material on to the skirt. The simulated skirt are shown in the second-sixth columns.	112

LIST OF ABBREVIATIONS

FEM	Finite element method
LSTM	Long Short-Term Memory
CNN	Convolutional Neural Network

LIST OF SYMBOLS

\mathcal{C}	material property parameters of the garment
\mathcal{G}	garment (sizing) parameters
$\mathcal{G}_{\text{pants}}$	$\langle w_1, w_2, w_3, w_4, h_1, h_2, h_3 \rangle$
$\mathcal{G}_{\text{skirt}}$	$\langle l_1, r_1, r_2, \alpha \rangle$
\mathcal{G}_{top}	$\langle r, w_1, w_2, h_1, h_2, l_1 \rangle$
\mathbf{U}	2D triangle mesh representing garment pattern
\mathbf{u}	vertex of the 2D garment pattern mesh
\mathbf{V}	3D triangle surface mesh of the garment
\mathcal{V}	simulated 3D triangle surface mesh of the garment
\mathbf{B}_{body}	skeleton of 3D human body mesh
$\mathbf{B}_{\text{garment}}$	skeleton of 3D garment mesh
\mathbf{x}	vertex of the 3D triangle mesh
\mathcal{P}	average wrinkle density of the 2D segmented garment in the image
\mathbf{k}	bending stiffness parameters
\mathbf{w}	stretching stiffness parameters
\mathbf{F}	deformation gradient of the deforming garment
\mathcal{K}	regional average curvature, 1D scalar
\mathcal{S}	2D garment silhouette
Ψ	bending energy of the garment
Φ	stretching energy of the garment
v_{ij}	rigging weights of the 3D garment mesh
β	joint angles of the skeleton of the 3D garment mesh
\mathcal{D}_c	garment database
\mathcal{D}_h	human body database

CHAPTER 1: INTRODUCTION

Physically-based animation, unlike traditional keyframing, can automatically generate realistic motion and deformation without tedious and time-consuming low-level control (Terzopoulos et al., 1987) 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 (Bickel et al., 2010; Wang et al., 2011b; Miguel et al., 2012a, 2013; Ren et al., 2013).

For example, the use of material-property estimation in the simulation of cloth has been suggested in (Wang et al., 2011b; Miguel et al., 2012a); it has also been used in procedures for designing and fabricating materials that produce a certain deformation behavior (Bickel et al., 2010). 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 (Pai et al., 2001; Bickel et al., 2009).

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. Prostate cancer, for instance, is one of the most commonly diagnosed type of cancer for men worldwide (shown in Fig. 1.1). Early diagnosis can greatly increase the survival rate of cancer. Cancerous tissue tend to be stiffer than normal tissue. The recovery of the mechanical properties of those tissue can greatly help doctors for the diagnosis. Currently, 2D Elastography (Ophir et al., 1999; Zhu et al.,

2003a; Kallel and Bertrand, 1996) 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.

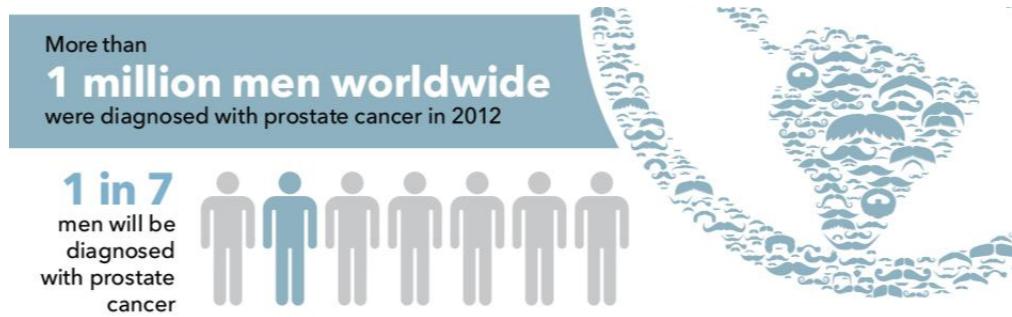


Figure 1.1: **Prostate cancer statistics (Cook, 2017).**

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.

To reconstruct a realistic virtual world, physics properties of the objects in the scene are also indispensable. Physics properties such as the mechanical properties of the object can greatly affect the virtual world human interaction. For example the material properties of the object determines the sound it makes. And for the application of virtual try-on, garment material properties have impact on both the appearance and the fit.

1.1 Human-tissue Material Property Identification

Currently screening of prostate cancers is usually performed through routine prostate-specific antigen (PSA) blood tests and/or a rectal examination. Based on positive PSA indication, a biopsy of randomly sampled areas of the prostate can then be considered to diagnose the cancer and assess its aggressiveness. Biopsy may miss sampling cancerous tissues, resulting in missed or delayed diagnosis, and miss areas with aggressive cancers, thus under-staging the cancer and leading to under-treatment.

Studies have shown that the tissue stiffness described by the tissue properties may indicate abnormal pathological process. Ex-vivo, measurement-based methods, such as (Ashab et al., 2015; Khojaste et al., 2015) using magnetic resonance imaging (MRI) and/or ultrasound, were proposed for study of prostate cancer tissue. However, previous works in material property reconstruction often have limitations with respect to their genericity, applicability, efficiency and accuracy (Yang and Lin, 2015b). More recent techniques, such as inverse finite-element methods (Goksel et al., 2013; Lee et al., 2012b; Shahim et al., 2013; Vavourakis et al., 2016; Yang and Lin, 2015b), stochastic finite-element methods (Shi and Liu, 2003), and image-based ultrasound (Uniyal et al., 2014) have been developed for in-vivo soft tissue analysis.

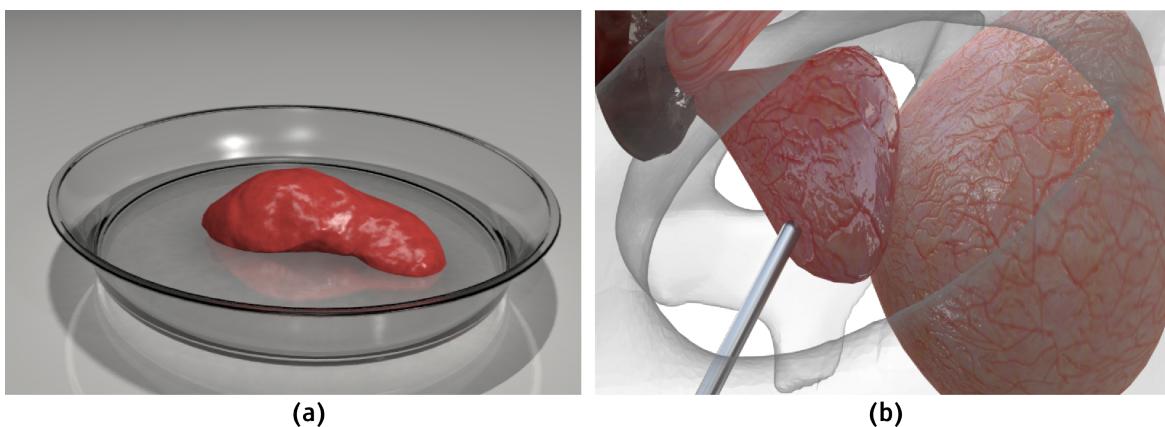


Figure 1.2: The Virtual Surgery Application. (a) shows the liver, with elasticity parameter reconstructed from patient data, resting on a plate. (b) is the screenshot of our virtual surgery system, using elasticity parameters for the prostate that were reconstructed from patient data.

1.2 Cloth Material Property Recovery

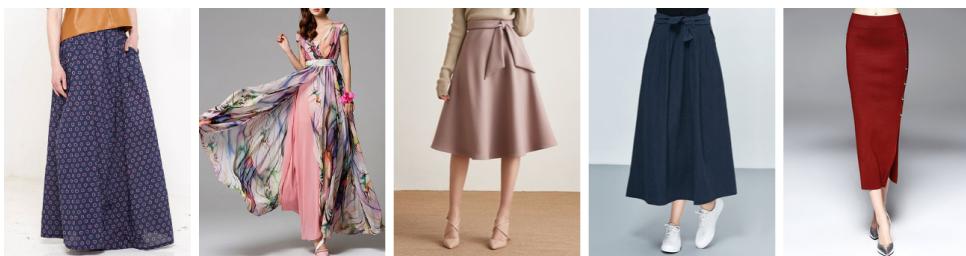


Figure 1.3: **Skirts with different material types (stylewe, 2014).** Skirts with different material properties show case different appearance.

Retail is a multi-trillion dollar business worldwide, with the global fashion industry valued at \$3 trillion (FashionUnited, 2016). Approximately \$1.6 trillion of retail purchasing in 2015 was done via online e-commerce sales, with growth rates in the double digits (Lindner, 2015). Thus, enabling better online apparel shopping experiences has the potential for enormous economic impact. Given the worldwide demand for fashion and the obvious impact of this demand on the apparel industry, technology-based solutions have recently been proposed, a few of which are already in use by commercial vendors. For example, there are several computer-aided design (CAD) software systems developed specifically for the apparel industry. The apparel CAD industry has focused predominantly on sized cloth-pattern development, pattern-design CAD software, 3D draping preview, and automatic generation of 2D clothing patterns from 3D body scanners or other measurement devices. Some of the leading apparel CAD companies include Gerber Technology, Lectra, Optitex, Assyst, StyleCAD, Marvelous Designer, Clo-3D and Avametric, etc. Unfortunately, developing these systems requires careful and lengthy design by an apparel expert.

More recently, there have been efforts to develop virtual try-on systems, such as triMirror, that allow users to visualize what a garment might look like on themselves before purchasing. These methods enable 3D visualization of various garments, fast animation of dynamic cloth, and a quick preview of how the cloth drapes on avatars as they move around. However, the capabilities of these new systems are limited, especially in terms of ease-of-use and applicability. Many of the virtual try-on systems use simple, fast image-based or texture-based techniques for a fixed number of avatar

poses. They do not typically account for the effects on fabric materials of different conditions (e.g. changing weather, varying poses, weight fluctuation, etc). Furthermore, almost all of these virtual try-on systems assume either that the user selects one of a pre-defined set of avatars or that accurate measurements of their own bodies have been captured via 3D scans.

Garments with different material types have different appearance as is shown in Fig. 1.3. Cotton skirts, for example are generally stiffer than silk skirts. And thus they have less wrinkles. Cotton tops and silk tops also fit differently on human body.

1.3 Thesis Statement

Material properties of the non-rigid objects presented in images or videos can be estimated using a coupled deterministic simulation optimization, a stochastic statistical inferring based method or a learning based method.

To support this thesis statement, I propose a simulation optimization coupled framework and a learning based method for soft-body material recovery. Given two sets of images, I apply the simulation optimization coupled framework to recover the material properties of the soft-body in the images. The two sets of images record the deformation of the soft-body. I evaluate the effectiveness of the framework using both cancer stage material property correlation test and synthetic tests.

In the following sections, I present the main results using the simulation optimization coupled framework results.

1.4 Main Results

In this section, I show the results for using the coupled simulation optimization framework to recover mechanical properties of human-tissue and the learning-based method for cloth material recovery results.

1.4.1 Human-tissue Material Recovery

We treat all the target images ($N = 84$) of all the patients as data points of equal importance. The elasticity feature for each target image is the recovered elasticity parameter $\hat{\mu}$. In this experiment, we train our classifier using the elasticity feature of the 83 images then cross validate with the one left out. Then, we add the patient's age as another feature to the classifier and perform the validation. The results for cancer staging (T-Stage) classification are shown in Fig. 5.2(a) and that for cancer grading (Gleason score) classification are shown in Fig. 5.2(b). The error metric is measured as the absolute difference between the classified cancer T-Stage and the actual cancer T-Stage. Zero error-distance means our classifier accurately classifies the cancer T-Stage.

The multinomial method outperforms the ordinal method for both cancer staging (T-Stage) and cancer aggression (Gleason score) classification. The main reason that we are observing this is due to the optimization weights or the unknown regression coefficients β (refer to supplementary document for the definition) dimension of the multinomial and ordinal logistic regression method. The dimension of the unknown regression coefficients of the multinomial logistic regression for cancer staging classification (with elasticity parameter and age as features) is 6 while that of ordinal logistic regression is 4. With the ‘age’ feature, we obtain up to 91% accuracy for predicting cancer T-Stage using multinomial logistic regression method and 89% using ordinal logistic regression method. For Gleason score classification we achieve up to 88% accuracy using multinomial logistic regression method and 81% using ordinal logistic regression method.

1.4.2 Cloth Material Recovery

1.5 Organization

The subsequent chapters of this dissertation are organized as follows.

Chapter 2 surveys previous work that are related to the thesis work including: physically-based deformable body simulation methods, human-tissue material recovery methods and cloth mechanical property identification methods.

Chapter 3 introduces my proposed method which recovers the material properties of multiple regions of a deformable body.

Chapter 4 presents the graphical model based dynamic deformable body mechanical property recovery from video.

Chapter 5 shows the real-life patient data analysis. I first identifies the material properties of the prostate of each patients using my proposed method introduced in Chapter 2. Then I build a classifier using two features: recovered prostate mechanical property and patients' age. I demonstrate in this chapter the effectiveness of the recovered material properties for cancer identification.

Chapter 6 presents a method which reconstruct garment from a single-view image.

Chapter 7 introduces a learning-based method to identify material properties of a piece of cloth from video.

CHAPTER 2: PREVIOUS WORK

In this chapter, I survey the previous work that are related to my thesis work. In the following sections, I first survey on physically-based soft-body simulation methods which are the basis of my research. Then, I introduce previous work on human-tissue mechanical property recovery methods and cloth mechanical property recovery methods. Finally, I survey on previous work on deep neural network which is related to my proposed learning-based cloth material recovery method.

2.1 Deformable Body Simulation

In computer graphics, extensive research have been done for deformable body simulation (Terzopoulos et al., 1987; Nealen et al., 2006; Müller and Gross, 2004; Meehan et al., 2003). The methods can be classified into couple of categories: non-physically based, physically-based and hybrid methods. Non-physically based are those methods that are pure geometric such as B-spline based surface editing method (Bartels and Beatty, 1989; Piegl, 1989a,b; Griessmair and Purgathofer, 1989) and free-form deformation (Barr, 1984; Chang and Rockwood, 1994). These methods are particularly useful for computer aided design (CAD). The most basic physically-based soft-body simulation is the mass-spring system (Chadwick et al., 1989). It discretizes the soft-body into vertices and springs. The springs connect the vertices to form a volumetric mesh of the soft body. The physics of the mass-spring system is based on the Hookd's law. The mass-spring system is widely used for real-time animation such as hair simulation (Hadap and Magnenat-Thalmann, 2001; Selle et al., 2008), muscle simulation (Nedel and Thalmann, 1998), human tissue simulation (Kühnapfel et al., 2000; Mollemans et al., 2003; Westwood et al., 2005; Hammer et al., 2011) but it has its limitations. Due to the inaccurate approximation to the underlying continuum physics, it is unstable primarily because of the time step size restriction when simulating stiff deformable body.

Finite element method (FEM) (Bathe, 2006) is one class of physically-based deformable body simulation methods. FEM was proposed in general as a numerical method to solve the underlying governing partial differential equation (PDE). It is frequently used for flesh/muscle simulation (Teran et al., 2003, 2005; Sifakis et al., 2006; Irving et al., 2007), surgery simulation (Bro-Nielsen, 1996; Bro-Nielsen and Cotin, 1996; Miller, 1999; Berkley et al., 2004).

Simulation of cloth and garments as one special kind of deformable body has also been extensively studied in computer graphics (Ng and Grimsdale, 1996; House and Breen, 2000; Bridson et al., 2002; Govindaraju et al., 2007). Methods for cloth simulation can also be divided classes based on the underlying numerical methods: mass-spring system based cloth simulation (Vassilev et al., 2001), FEM-based cloth simulation (Narain et al., 2012).

2.2 Human-Tissue Mechanical Property Recovery

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. There are mainly two kinds of soft tissue elasticity properties estimation method (Samur et al., 2007): invasive and non-invasive techniques. The invasive methods rely on a device to measure the displacement and force response (Carter et al., 2001; Kauer et al., 2002a; Rosen et al., 1999). These methods take organ samples either out of the human or animal bodies and perform the experiment in-vitro (out side the body) or do the procedure in-situ (inside the body). The collected data are then used to solve the inverse problem, which is to recover the elasticity properties, by constructing a polynomial interpolation (Bicchi et al., 1996) or by using a finite element model (Gao et al., 2009; Misra et al., 2010; Samur et al., 2007).

The non-invasive methods mostly base on image analysis techniques to measure the displacement. In the 1980s, several methods were proposed to measure the motion of the soft tissue, such as the one proposed by Wilson and Robinson (Wilson and Robinson, 1982) using radio frequency M-mode signals and the one proposed by Birnholz and Farrell (Birnholz and Farrell, 1985) using ultrasound B-scans. Researchers have also used medical image analysis on 2D ultrasound and/or

MRI images to estimate the elastic parameters of soft tissue (Gao et al., 1996; Manduca et al., 2001; Zheng and Mak, 1996).

(Washington and Miga, 2004) 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.

2.2.1 Measurement-based Methods

Other than distance field based methods, there are also other measurement algorithms. The modality-independent elastography (MIE) method (Miga, 2002) measures the elasticity parameters by maximizing the image similarity based on a number of landmarks. However, this technique does not apply to all the soft tissues, as landmarks cannot always be found in some of the organs such as prostate. Statistical and machine learning algorithms have also been used to classify soft tissues and estimate the parameters using multi-spectral MR images (Liang et al., 1994).

2.2.2 Elastography

2D elastography (also known as elasticity reconstruction), can acquire ‘strain images’ or ‘elasticity images’ of soft tissues (Skovoroda and Emelianov, 1995; Rogowska et al., 2014; Bilston and Tan, 2015). 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 (Ophir et al., 1999; Rivaz et al., 2008). They were proposed to avoid the explicit measurement of the displacement. Van Houten et al. (Van Houten et al., 1999, 2001) used elastography methods to estimate the Young’s modulus distribution of a deformable body. These methods need high-resolution displacement fields to recover the elasticity parameter (Manduca et al., 2001). The displacement field can be obtained through an external device using a vibration actuation mechanism. External device is still required for both the force exertion on the soft tissue and the external force measurement. Special vibrator were placed inside the organ (Chopra et al., 2009) to complete the procedure. Other than distance-field-based methods for high-resolution

magnetic resonance medical images, other video based method measure the displacement field from the video (Syllebranque et al., 2007).

Alternatively, the displacement field can be found with a modified MRI machine that is in tune with a mechanical vibration of tissues (Muthupillai and Ehman, 1996; Fu et al., 2000). 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 (Zhu et al., 2003a) or by performing an iterative optimization to minimize the error in the deformation field (Kallel and Bertrand, 1996; Balocco et al., 2008; Schnur and Zabaras, 1992). One limitation for this class of methods is that they require a device both to measure the external force and to exert external force on the deformable bodies.

2.2.3 Inverse Finite Element Methods

Inverse finite element methods use the implicitly known or computed external force to recover the mechanical property. Lee et al. proposed a model to estimate the Young's modulus based on low-resolution CT images and no external force is required to set the boundary condition (Lee et al., 2012a). The most recent work on identification of mechanical properties based on surface tracking (Wang et al., 2015) proposed a decoupled iterative tracking and parameter estimation framework. They applied a combined probabilistic physically-based method for surface tracking. Then, the tracked surfaces of the static state are the input to the parameter estimation framework. This approach does not require external force to be measured, but uses the gravity as the boundary condition. This class of methods avoid both the explicit measurement of external force and the displacement field. But the input references need to be in static state.

2.2.4 Probabilistic Graphical Models

Probabilistic graphical models, such as Kalman Filter and Hidden Markov Model (HMM), have been used for both state estimation and parameter identification of dynamical systems. E.J. Lefferts et al. applied Kalman Filter for estimation of spacecraft attitude (Lefferts et al., 1982).

Kalman filters have been widely used in tracking of dynamical systems and parameter estimation, including rigid-body tracking (Zhan and Wan, 2007; Moghari and Abolmaesumi, 2007), motion retargeting (Tak and Ko, 2005), geometric contour tracking (Dambreville et al., 2006), and more. We adopt a tightly coupled framework using FEM-based state estimation and unscented Kalman Filter (UKF) for parameter identification.

2.3 Cloth Mechanical Property Recovery

Other models have focused on physically-based simulation of cloth, especially on determining the cloth model's stiffness and damping coefficients. (Bhat et al., 2003) 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 (Press, 2007) to minimize estimation error. K.L. Bouman et. al. (Bouman et al., 2013) presented a learning based method to identify cloth material from a cloth moving video. They proposed to use visual features such as optical flow and texture features. Becker and Teschner presented a novel framework based on quadratic programming to determine linear elastic parameters; they also assessed the effects of noisy measurements (Becker and Teschner, 2007b). Most recently, improved data-driven methods have been proposed to estimate cloth parameters (Wang et al., 2011b; Miguel et al., 2012a, 2013), which can photorealistically recreate the look of real fabric. Lately, A. Davis (Davis et al., 2015) introduced a method to estimate the material properties of a deformable thin shell based on the video which records the vibration pattern of it.

CHAPTER 3: MULTI-REGION IMAGE-BASED ELASTICITY RECOVERY

3.1 Introduction

Material properties are important in depicting the characteristics of virtual objects for realistic computer animations of soft bodies. Medical robots have the potential to perform surgical procedures beyond current clinical capabilities. To enable medical robots to safely operate within highly unstructured, deformable human bodies and for designing responsive and dexterous surgical manipulators. In addition, virtual surgical simulation is also been increasingly used for rapid prototyping of clinical devices, pre-operation planning of medical procedures, virtual training exercises for surgeons and medical personnel, etc. And, tissue elasticity properties are important parameters for developing accurate and predictive surgical simulation. Futhermore, to compute desired and accurate force feedback for haptic display requires knowledge about the deformation of soft tissues and organs, which are characterized by patient-specific elastic parameters for different tissues and organs.

Studies(Fowlkes et al., 1995) (Garra et al., 1997) show that tissue elasticity parameters are important indicators of cancer and other diseases. Traditional cancer detection methods such as palpation have limitations. For tissues that locate inside the human body and far from the skin, procedures like palpation cannot reach. Besides, these methods are based on the empirical knowledge of the doctor. Further for each patient, the normal elastic properties may vary making it difficult for the doctor to establish a diagnoses standard. Thus, researchers proposed methods that are based on scientific computing instead of the sense of the doctor to estimate the elastic parameters of soft tissue. The emerging field of elastography tries to solve this problem. Methods that are based on medical image analysis and FEM based simulation provide a rigorous way of calculating the elasticity parameters of the potential cancer tissues.

Elastography(Ophir et al., 1991a) was first proposed to determine the elasticity properties by measuring the deformation of the tissue due to the application of the known external forces. The known external forces are the boundary condition needed to recover the exact elasticity parameter. Originally, the deformations were measured using ultrasound imagery(Krouskop et al., 1987), but such techniques produced coarse, two-dimensional representations of the moving tissue. More sophisticated imaging techniques, such as magnetic resonance imaging (MRI)(Chenevert et al., 1998; Van Houten et al., 1999)and computed tomography (CT), produce three-dimensional images of the deforming tissues, allowing a more accurate measurement of displacement.

In the early days, ultrasound images were used in elastography(Krouskop et al., 1987). However, ultrasound images can only provide low resolution information. Compared to ultrasound images, magnetic resonance imaging (MRI) (Chenevert et al., 1998; Van Houten et al., 1999)and computed tomography (CT) images offer high resolution three-dimensional data. Based on magnetic resonance images or computed tomography images, three dimensional model can be reconstructed. Finite element method is used to simulate the deformation of the three dimensional model. By applying optimization method or other kinds of method, elasticity parameters are recovered.

Being able to estimate the elasticity parameters of the patients tissue accurately can greatly help with the diagnoses. For prostate cancer diagnoses, doctors need to sample the tissue of the prostate in order to be certain about stage of the cancer and the kind of cancer. This procedure, however, is done without the prior knowledge of which region to sample. Thus, the tissue sample may offer misleading information. If I can learn before hand the elasticity of the two parts of the prostate, the sampling result may be able to provide valuable information. For problem like this, previous elastography methods have their limitations.

Toward realizing the concept of 3D physiological humans, I propose perhaps one of the first elasticity parameter estimation algorithm for *multiple, heterogeneous* deformable bodies simultaneously using medical images¹. My approach is based on a multi-dimensional optimization method that iteratively performs deformable body simulation using a finite element method on

¹ In this work, I use *computed tomography (CT)* images. But, the algorithm is also applicable to magnetic resonance imaging (MRI) images.

reconstructed organ models with the continuously refined, estimated elasticity parameters. The geometric models of organs are reconstructed based on low-resolution CT images. My objective function measures the sum of the distance between the nodes of the organ surface. In contrast to elastography methods (Becker and Teschner, 2007a; Eskandari et al., 2011; Zhu et al., 2003b), the only information I need is the displacement of the nodes of the organ surface. I do not need every pixel-wise displacement vector, thus no extra procedures need to be performed on the patient. Two sets of (medical) images are sufficient to recover the elasticity parameters using my method. Therefore, my method can be widely applicable to different imaging technology. It can be used for animatino of soft bodies (see supplementary video) and possibly for cancer staging using only low-resolution CT images.

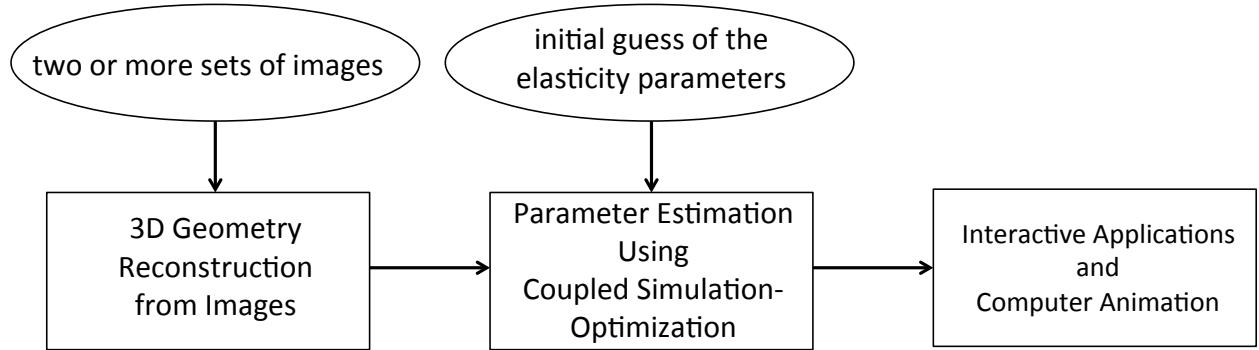


Figure 3.1: The Flow Chart of My Framework. My framework takes (at least) two sets of images (medical images or other multi-view images) as input; I use these images to reconstruct 3D meshes. The initial guess at the elasticity parameter is based on standard values and is given prior to the start of 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.

3.2 Method

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

Fig. 3.1 provides an overview of my system. I assume that (at least) two sets of multiple-view images are given, along with some initial guess at the elasticity parameters. First, I offer an overview of how my method uses the input images to produce the geometrical reconstructions that are then

used in the elasticity-parameter reconstruction. Next, I describe each component of my system in more detail.

3.2.1 Geometry Reconstruction and Mesh Generation

There are many approaches to reconstructing 3D geometry from multiple images; the method chosen depends upon the image sources. I 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 (Yushkevich et al., 2006b) 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 (Moons et al., 2010; Yushkevich et al., 2006b; Snavely, 2008; Wu, 2013b). 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 (Yushkevich et al., 2006b) into multiple

regions. After mesh simplification and smoothing, the entire region of interest can be tetrahedralized using TetGen (Si, 2007).

3.2.2 Quasi-Static Process Elasticity Parameter Estimation

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

3.2.2.1 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. I use the FEM to solve the following governing equation of the deformable body.

$$\int_{\Omega} \delta \mathbf{u}^T \rho \ddot{\mathbf{u}} \, d\Omega + \int_{\Omega} \delta(\boldsymbol{\varepsilon})^T \boldsymbol{\sigma} \, d\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{b} \, d\Omega - \int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} \, d\Gamma = 0, \quad (3.1)$$

with \mathbf{u} as the displacement field, $\boldsymbol{\varepsilon}$ 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 $\ddot{\mathbf{u}} = 0$. I can rewrite Eqn. 3.1 as

$$\left[\int_{\Omega} \delta(\boldsymbol{\varepsilon})^T \boldsymbol{\sigma} \, d\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{b} \, d\Omega \right] - \left[\int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} \, d\Gamma \right] = 0, \quad (3.2)$$

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, I use the isotropic hyperelastic material model, which is used commonly to approximate the deformation behavior of human tissue (Hu and Desai, 2004). The stress-strain relation for the hyperelastic material model is defined through the strain energy density function Ψ (energy per unit undeformed volume). I will be using the Green-Lagrange strain tensor

E with the second Piola-Kirchhoff stress tensor σ^{PK2} (Gurtin, 1982). The boundary conditions I apply are the tractions t applied on the boundary Γ . My forward simulation framework uses an invertible FEM (Teran et al., 2005) to ensure that the deformed elements have positive volumes in the coupled simulation-optimization process.

3.2.2.2 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, I can write

$$\sigma = D\epsilon, \quad (3.3)$$

where σ is the stress tensor induced by the *surface forces*, ϵ is the strain tensor defined by the spatial derivatives of the displacement u , and D is a matrix defined by the material property parameters μ ($D = D(\mu)$). 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 (Hu and Desai, 2004). The nonlinearity is captured through the energy density function Ψ for hyperelastic material models. The energy function is a function of the strain tensor ϵ and the material property parameters μ , where $\Psi = \Psi(\epsilon, \mu)$. With the energy function, the stress tensor is

computed by taking the derivative of the energy function over the strain tensor.

$$\boldsymbol{\sigma} = \frac{\partial \Psi(\boldsymbol{\varepsilon}, \boldsymbol{\mu})}{\partial \boldsymbol{\varepsilon}} \quad (3.4)$$

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} ,

$$\begin{aligned}\mathbf{I}_1 &= \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \\ \mathbf{I}_2 &= \lambda_1^2 \lambda_2^2 + \lambda_2^2 \lambda_3^2 + \lambda_1^2 \lambda_3^2 \\ \mathbf{I}_3 &= \lambda_1^2 \lambda_2^2 \lambda_3^2\end{aligned} \quad (3.5)$$

and the deformation gradient \mathbf{F} is a function of the strain $\mathbf{F} = \mathbf{F}(\boldsymbol{\varepsilon})$. One general energy function for this class of incompressible materials, proposed by Rivlin (Rivlin, 1948), is

$$\Psi_{\mathbf{R}} = \sum_{i,j=0}^{\infty} \mathbf{C}_{ij} (\mathbf{I}_1 - 3)^i (\mathbf{I}_2 - 3)^j, \quad (3.6)$$

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.

$$\Psi = \Psi_{\mathbf{R}} + \Psi(\mathbf{J}), \quad (3.7)$$

where \mathbf{J} is the volume ratio $\mathbf{J} = \sqrt{\mathbf{I}_3}$. I 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; I use this model in my implementation because of its popularity and wide adoption in both medical and engineering applications. In this work, I use this form of the energy function of Mooney-Rivlin

material model (Treloar et al., 1976; Rivlin and Saunders, 1951):

$$\Psi = \frac{1}{2}\mu_1((\mathbf{I}_1^2 - \mathbf{I}_2)/\mathbf{I}_3^{\frac{2}{3}} - 6) + \mu_2(\mathbf{I}_1/\mathbf{I}_3^{\frac{1}{3}} - 3) + v_1(\mathbf{I}_3^{\frac{1}{2}} - 1)^2, \quad (3.8)$$

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 my simulation, I model abdominal organs as incompressible material (Nava et al., 2003). 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 Eqn. 5.2) to approximate incompressibility. To achieve incompressibility, I choose a fairly large v_1 ; this means v_1 will not be optimized. In order to accurately describe the material, I reconstruct both μ_1 and μ_2 .

3.2.2.3 The Boundary Condition

The boundary condition is critical in solving Eqn. 3.2. The boundary condition can be either the displacement field or the tractions on the boundary. My target applications for this work include both medical applications and sketch-driven animation; for medical applications, I use the contact force between the organ and the surrounding tissue as the boundary condition. To compute the contact force I make two assumptions:

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

During the model reconstruction step, I include the surrounding soft tissue of the prostate, as well as the bones of the male pelvis area (as shown in Fig. 5.3). I 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, I 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 I cannot accurately solve the governing equation, Eqn. 3.2; and
3. the relative material properties of the target organ have already proven to be useful for cancer detection(Tsutsumi et al., 2007).

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

$$\mathbf{K}\mathbf{u}_s = \mathbf{f}, \quad (3.9)$$

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.

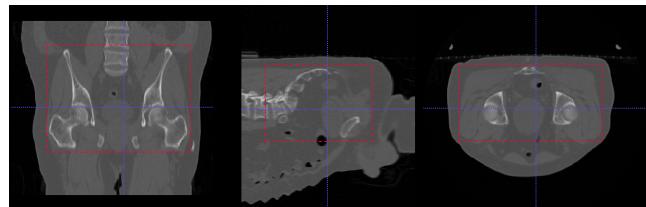


Figure 3.2: **The CT Image of Male Pelvic Area.** The red dotted lines are the boundary of my model reconstruction.

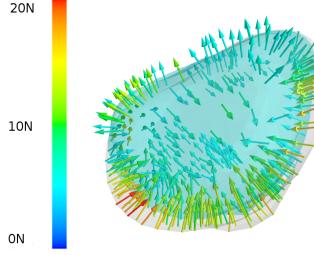


Figure 3.3: Reconstructed Contact Force of One Patient’s Prostate. The light colored transparent surface is the reference mesh; the nontransparent surface is the initial surface.

3.2.2.4 Distance-Based Objective Function

Since my simulation framework requires only two three-dimensional surface models of organs, my error metric is constructed using the Hausdorff distance d_H between the deformed organ surface S_l and the target reference organ surface 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

$$d_H(\mathbf{A}, \mathbf{B}) = \max_{a \in \mathbf{A}} \min_{b \in \mathbf{B}} d(a, b), \quad (3.10)$$

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

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

$$\Phi(\boldsymbol{\mu}) = \sum_{\mathbf{v}_l \in S_l} \|d_H(\mathbf{v}_l, S_t)\|^2. \quad (3.11)$$

The above Eqn. 6.12 will be the objective function for this optimization problem. The optimization problem is thus

$$\boldsymbol{\mu} = \operatorname{argmin}_{\boldsymbol{\mu}} \sum_{\mathbf{v}_l \in S_l} \|d_H(\mathbf{v}_l, S_t)\|^2, \quad (3.12)$$

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

3.2.2.5 Multi-Region Elasticity Parameter Estimation

Multi-region elasticity parameter reconstruction is much more complicated than single-body reconstruction. I 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. I adopt the second method in this work, since my 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, I 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 my examples, the prostate is naturally divided into two parts by the prostatic urethra, as shown in Fig. 3.4. My 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.

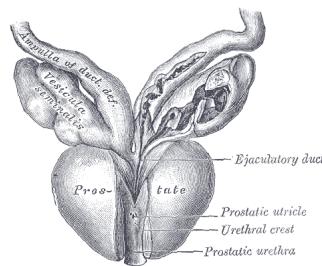


Figure 3.4: **The Prostatic Urethra.** The prostatic urethra naturally divides the prostate into two parts. © Wikipedia(Gray, 1918)

I choose to simulate the regions of the deformable body as one deformable body. I do this because I 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 Eqn. 6.12.

In this type of multi-region reconstruction, I use the following objective function:

$$\Phi(\boldsymbol{\mu}) = \sum_{m=1}^M \sum_{\mathbf{v}_l \in \mathbf{S}_{m,l}} \|\mathbf{d}_H(\mathbf{v}_l, \mathbf{S}_{m,t})\|^2, \quad (3.13)$$

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 m^{th} region. The definition of $\mathbf{S}_{m,l}$ is critical for the convergence of the reconstruction.

I exclude the nodes that are shared by other regions (as shown in Fig. 3.5(a) and Fig. 3.5). When the nodes shared by the regions are included in the objective function given in Eqn. 3.13, 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 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. I 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 Eqn. 3.13. With this approach, my experiments indicate that the multi-region parameter estimation can converge to the right parameters for each region simultaneously.

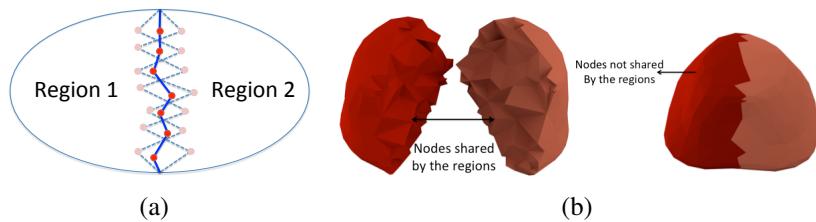


Figure 3.5: The 2D and 3D Illustration of the Nodes of Regions. (a) These nodes do not contribute to the convergence of the optimization. (b) The figure on the left shows nodes shared by the two regions; the figure on the right shows the nodes not shared by the two regions.

3.2.2.6 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, I have adopted a variant of the Particle Swarm Optimization (PSO) method (Kennedy et al., 1995; Poli et al., 2007; Clerc, 2010), 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. I use the subscript i to index the particle in the swarm, \mathbf{p}_i to represent the previous best position of the i^{th} 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 i^{th} particle in the current iteration; y_i^t (scalar value) represents the fitness value of the i^{th} 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 i^{th} particle at the t^{th} optimization iteration can then be defined as:

1. The position $\boldsymbol{\mu}_i^t = (\mu_{1,i}^t, \dots, \mu_{n,i}^t, \dots, \mu_{N,i}^t)$, with $\boldsymbol{\mu}_i^t \in \mathbf{H}$, $1 \leq n \leq N$, $1 \leq i \leq M$
2. The velocity $\mathbf{v}_i^t = (v_{1,i}^t, \dots, v_{n,i}^t, \dots, v_{N,i}^t)$, with $v_{\min} \leq \|\mathbf{v}_i^t\| \leq v_{\max}$, $1 \leq n \leq N$, $1 \leq i \leq M$
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 $\mathbf{p}_i^t = (p_{1,i}^t, \dots, p_{n,i}^t, \dots, p_{N,i}^t)$, with $\mathbf{p}_i^t \in \mathbf{H}$, $1 \leq n \leq N$,
 $1 \leq i \leq M$

5. The previous best position of its neighbors $\mathbf{p}_{g,i}^t = (p_{g,1,i}^t, \dots, p_{g,n,i}^t, \dots, p_{g,N,i}^t)$, with $\mathbf{p}_{g,i}^t \in \mathbf{H}$,
 $1 \leq n \leq N, 1 \leq i \leq M$

The position is a point in the Euclidean search space \mathbf{H} of the optimization problem. In my problem, the search space is the range of all possible elasticity parameters. The number of parameters I am 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 I want to estimate. The fitness value is computed from the fitness function Φ , which is the objective function of the optimization problem (Eqn. 6.12). 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 $\text{Rand}()$ as a random value generator.

$$\mathbf{v}_i^{(t+1)} = \mathbf{v}_i^t + \text{Rand}()(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t) + \text{Rand}()(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t) \quad (3.14)$$

$$\boldsymbol{\mu}_i^{(t+1)} = \boldsymbol{\mu}_i^t + \mathbf{v}_i^{t+1} \Delta t, \quad (3.15)$$

with $\Delta t = 1$. The particle's position update (Eqn. 3.14, Eqn. 3.15) can be computed since it is determined by several factors: the current position μ_i^t ; the persistence in the previous direction (first part of Eqn. 3.14); the influence of the previous best position of itself (second part of Eqn. 3.14); and the influence from its neighbors (third part of Eqn. 3.14). 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 Eqn. 3.14).

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 Eqn. 3.14 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 (Kennedy and Mendes, 2002), have been studied. The canonical particle swarm optimization (Eqn. 3.14, Eqn. 3.15) uses the global best solution so far; it connects every particle with every other particle. After some experiments, I chose instead to use the adaptive random structure (Clerc, 2012). 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 my problem, I chose $K = 3$ and $M = 40$ based on my experiments.

Velocity: The canonical way of updating velocity (dimension by dimension) is known to be biased (Monson and Seppi, 2005). Therefore, I adapted the method of computing the velocity or the

search direction from (Clerc, 2012). For each iteration, I update the velocity of the i^{th} particles by

$$\mathbf{v}_i^{(t+1)} = C(\mathbf{v}_i^t, \mathbf{p}_i^t - \boldsymbol{\mu}_i^t, \mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t). \quad (3.16)$$

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 ($\boldsymbol{\mu}_i^t + c(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t)$).

$$\mathbf{G}_i^t = \frac{\boldsymbol{\mu}_i^t + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t)) + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t))}{3}, \quad (3.17)$$

with c as a constant. I use $c = 1/2 + \ln(2)$ (Clerc, 2010) and I assume a radius of $r_{\max} = \|\mathbf{G}_i^t - \boldsymbol{\mu}_i^t\|$ I 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 (Eqn. 3.14) also result in velocity divergence. This problem was solved in paper (Shi and Eberhart, 1998) by the introduction of the inertia term ω ; I follow (Shi and Eberhart, 1998) by incorporating ω into my velocity update function to solve the problem of velocity divergence. Now my velocity update function is defined as

$$\mathbf{v}_i^{(t+1)} = \omega \mathbf{v}_i^t + \boldsymbol{\mu}_i'^t - \boldsymbol{\mu}_i^t, \quad (3.18)$$

where the ω is a constant. I set $\omega = \frac{1}{2\ln(2)}$ (Clerc, 2010). The update of the position of the i^{th} particle is then computed as

$$\boldsymbol{\mu}_i^{(t+1)} = \omega \mathbf{v}_i^t + \boldsymbol{\mu}_i'^t. \quad (3.19)$$

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 my system work flow is provided in Algorithm. 1.

3.2.3 Sensitivity Analysis

In the sensitivity analysis, I aim to identify the uncertainties in the output of the system that come from variation in the input. The input to my optimization-simulation-identification framework is the amount of tissue deformation inferred from two sets of images, while the output is the estimated elasticity parameters. I wish to identify the relation between the changes in the input, i.e. the deformation, and the output, the elasticity parameters. Mathematically, I take the derivative of the elasticity parameters on both sides of the objective function (Eqn. 6.12):

$$\frac{\partial \Phi(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}} = \frac{\partial \Phi}{\partial \mathbf{d}_H(\mathbf{u})} \frac{\partial \mathbf{d}_H(\mathbf{u})}{\partial \mathbf{u}} \frac{\partial \mathbf{u}}{\partial \boldsymbol{\mu}} \quad (3.20)$$

My 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, I study the μ_1 and μ_2 in Eqn. 5.2. Since μ_1 and μ_2 are related to distortional response, to simplify the test without losing generality I set μ_1 and μ_2 to the same value for the simulation of isotropic homogeneous deformable bodies. I 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, I also compare the results with those produced by linear material model.

Algorithm 1 Static Elasticity Parameter Estimation Method

```

1: procedure FORWARD SIMULATION( $\mathbf{u}$ ,  $\boldsymbol{\mu}_k$ )
2:    $\mathbf{u} + \Delta\mathbf{u} \leftarrow \mathbf{f}(\mathbf{u}, \boldsymbol{\mu}_k)$  // fem
3:   return  $\mathbf{u}_k$ 
4: procedure OBJECTIVE FUNCTION EVALUATION( $\mathbf{u}, \boldsymbol{\mu}$ )
5:    $\mathbf{u} + \Delta\mathbf{u} \leftarrow$  FORWARD SIMULATION( $\mathbf{u}, \boldsymbol{\mu}$ )
6:    $y \leftarrow \Phi(\mathbf{u}, \mathbf{u} + \Delta\mathbf{u}, \boldsymbol{\mu})$  // refer to Eqn.14
7:   return  $y$ 
8: procedure PARTICLE SWARM OPTIMIZATION( $\mathbf{u}$ )
9:   Step 1: Initialize
10:  for all the particles do
11:     $\boldsymbol{\mu}_i^0 \leftarrow$  randomly selected position from search space  $\mathbf{H}$  with uniform distribution
12:     $\mathbf{v}_i^0 \leftarrow$  randomly selected vector with the length not larger than the radius of the search
space  $\mathbf{H}$ 
13:     $\mathbf{p}_i^0 \leftarrow \boldsymbol{\mu}_i^0$ 
14:     $\mathbf{p}_{g,i}^0 \leftarrow \boldsymbol{\mu}_i^0$ 
15:     $y_i^0 \leftarrow 0$ 
16:  close;
17:
18:  Step 2: Iterate
19:  while not converged do
20:    for all the particles do
21:       $y_i^t \leftarrow$  OBJECTIVE FUNCTION EVALUATION( $\mathbf{u}, \boldsymbol{\mu}_i^t$ ) // evaluate fitness function
22:       $\mathbf{G}_i^t \leftarrow \frac{\boldsymbol{\mu}_i^t + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_i^t - \boldsymbol{\mu}_i^t)) + (\boldsymbol{\mu}_i^t + c(\mathbf{p}_{g,i}^t - \boldsymbol{\mu}_i^t))}{3}$ . // evaluate the center of the D-sphere
23:       $\boldsymbol{\mu}'_i^t \leftarrow$  randomly picked point from the  $D$ -sphere with uniform distribution
24:       $\mathbf{v}_i^{t+1} \leftarrow \omega \mathbf{v}_i^t + \boldsymbol{\mu}'_i^t - \boldsymbol{\mu}_i^t$  // update velocity
25:       $\boldsymbol{\mu}_i^{t+1} \leftarrow \boldsymbol{\mu}_i^t + \mathbf{v}_i^{t+1}$  // update position
26:      if  $y_i^t$  better than previous best fitness value then
27:         $\mathbf{p}_i^{t+1} = \boldsymbol{\mu}_i^t$ 
28:      else
29:         $\mathbf{p}_i^{t+1} = \mathbf{p}_i^t$ 
30:      close;
31:    close;
32:    for all the particles do
33:      Iterate the neighbors update  $\mathbf{p}_{g,i}^{t+1} \leftarrow$  the best  $\mathbf{p}_i^{t+1}$ 
34:    close;
35:  close;
36:  return  $\boldsymbol{\mu}$ 
37: procedure MAIN
38:   while not converged do
39:     PARTICLE SWARM OPTIMIZATION( $\mathbf{u}$ )
40:     close;
41:   close;

```

I design the model of my test as a simple sphere embedded in a cube. The entire domain has been tetrahedralized using TetGen (Si, 2007). The sliced view of the model is shown in Fig. 3.6. 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 I will use *relative* elasticity parameter instead of absolute values for 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, I can usually assume default values for the elasticity parameters of the tissue surrounding the organ of interest).

My 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 I also fixed the elasticity parameters of the area between the surface of the sphere and the cube.

I also study the difference in material behavior between the linear material model and the nonlinear material model. To make a fair comparison, I 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. I first set the elasticity parameters of the sphere and the surrounding area to this same set of values. Then I change the elasticity parameters of the sphere and record the deformation of the sphere.

To show the result, I 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. 3.7. Fig. 3.7(a) 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. 3.7(a) is in fact $\frac{\partial u}{\partial \mu}$. When the value of $\frac{\partial u}{\partial \mu}$ is high, large variation or uncertainty in the input does not lead to large errors in the output. This implies that my optimization framework is less prone to uncertainties and numerical errors when the elasticity parameters are small; when the value of the elasticity parameter

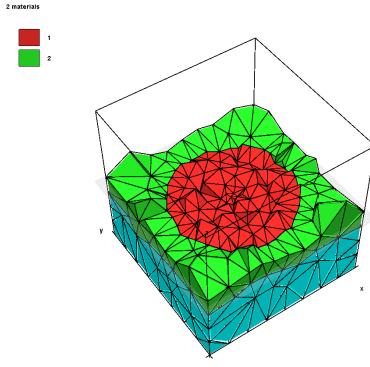


Figure 3.6: Sliced View of the Model Used for Sensitivity Analysis.

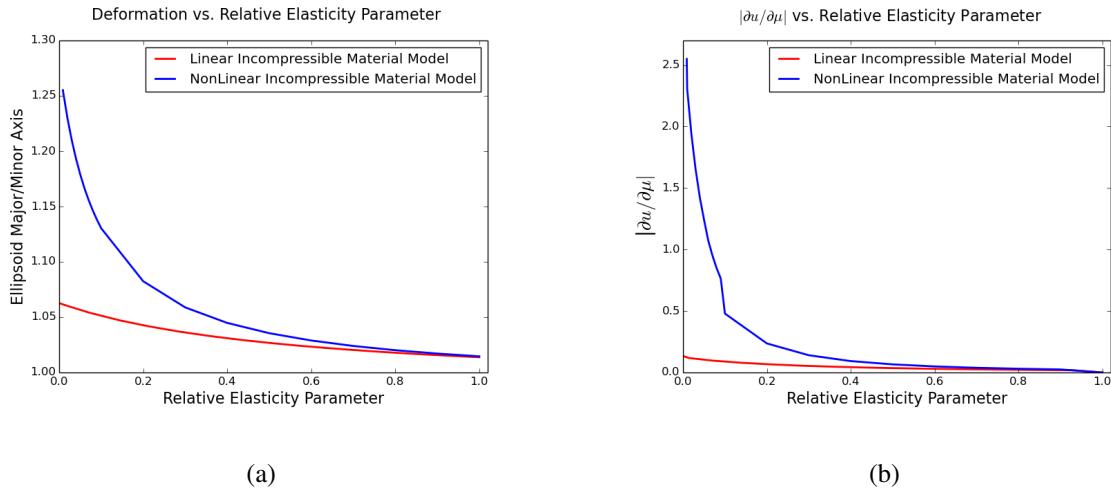


Figure 3.7: The Sensitivity Analysis Results. (a) shows the relation between amount of deformation represented by the inner sphere's radius changes vs. the relative elasticity parameter. (b) shows the relation between the change in deformation per change in the relative elasticity parameter vs. the relative 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. 3.12), for surgical procedures, or for haptic rendering.

Furthermore, in both Fig. 3.7(a) and Fig. 3.7(b), I observe that the slope of the blue line is almost always larger than that of the red line (i.e., the blue line lies above the red line), as shown in Fig. 3.7(a) and in Fig. 3.7(b). 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.

3.3 Experiments

I have implemented my 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 my approach on a real-world (live) patient dataset (I 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. I demonstrate the application of my 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, I automatically estimate the elasticity parameters and recreate the desired deformation using the MaterialCloning algorithm. Please see the supplementary video for demonstration of these applications.

3.3.1 Multi-Region Elasticity-Parameter Reconstruction

The experiment is designed to validate my multi-region, elasticity-parameter reconstruction using synthetic data. In this experiment I divide the organ of interest into two regions; one of the regions contains an embedded tumor with high elasticity parameters (Fig. 3.8). I validate my 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 (Tang et al., 2001). Therefore, the average value of the two material parameters is used in the following studies. I test the accuracy and the robustness of my framework by varying the size of the tumor, as shown in Fig. 3.8. In this experiment I assign the tumor elasticity parameters a value

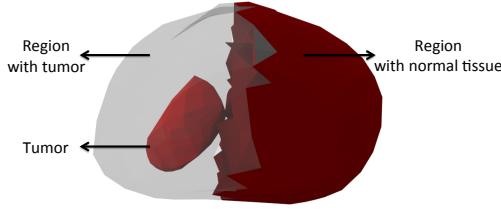


Figure 3.8: **Different Regions of the Organ.** left with a tumor embedded; right with normal tissue.

Table 3.1: Multi-Region Elasticity Parameter Reconstruction

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

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

of 70kPa ; the rest of the elements are assigned 30kPa . I expect the recovered elasticity parameter for the normal tissue region to be close to 30kPa and the recovered elasticity parameter for the tumor region to correlate with the tumor size. Results in Table 3.1 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%. An example optimization process for the normal tissue region is shown in Fig. 3.9.

I further validate my 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% of the entire region on the left side of Fig. 3.8. The recovered value, shown in Table. 3.2, has a high linear correlation coefficient of 0.9856.

One possible source of error in my multi-region elasticity parameter reconstruction scheme comes from variation in the mesh resolution. It is expected that finer mesh resolution will produce increased accuracy. To test this assumption, I re-run the first part of my multi-region experiment on models with different mesh resolutions. In this experiment, I vary the size of the tumor in the tumor region (shown in Fig. 3.8), then recover the elasticity parameter for the normal region and the tumor region using models of different mesh resolutions. In Fig. 3.10, I plot the relative error of elasticity parameters recovered from models with different mesh resolution against the tumor-to-region ratio.

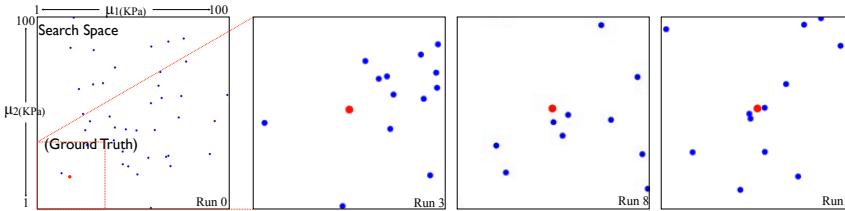


Figure 3.9: Particle-Swarm Optimization Process: The blue dots are the particles and the red dot signifies the ground truth.

Table 3.2: Multi-Region Elasticity Parameter Reconstruction

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

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

For the normal region, I set the ground truth to be 30 kPa . I 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. 3.10). I also found that the recovered elasticity parameters tend to converge better (i.e. with less fluctuation) with higher mesh resolution. As Fig. 3.11 shows, the blue and the green lines, which 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 my algorithm, as the resolution of the FEM meshes increases.

Other Sources of Error: I 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. I focus on two aspects, namely the mesh resolution (see above) and the sensitivity analysis on parameter dependency (Section 3.2.3), since the other possible sources of error are similar because of discretization.

3.3.2 Correlating Estimated Tissue Parameters with Cancer T-Stages

I adopt the experimental protocol from (Lee et al., 2012b), because it is impractical, if not impossible under existing regulations for human subject protection, to measure tissue elasticity of

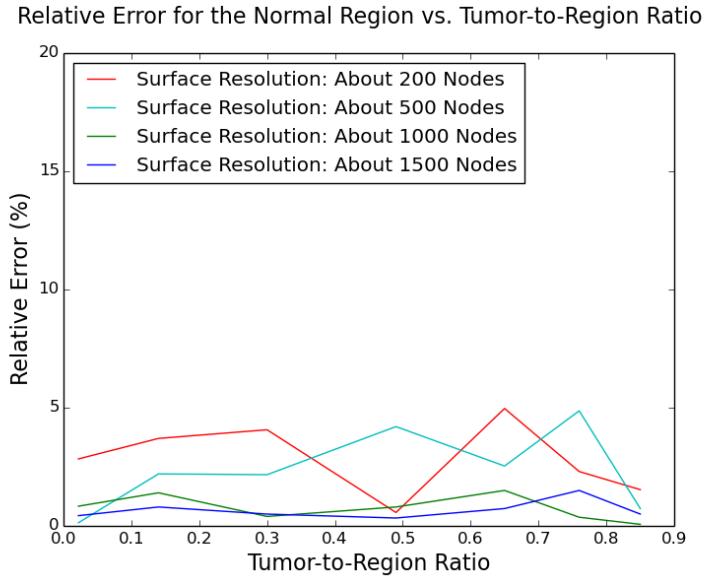


Figure 3.10: The Relative Error vs. Tumor-to-Region Ratio. This figure shows the relative error of elasticity parameters for the normal region recovered using models with varying mesh resolutions plotted against the tumor-to-region ratio.

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 3.3, follow the definitions in the TNM (Tumor, lymph Nodes, Metastasis) system (Sobin et al., 2011), a common cancer staging system. I use ten sets of real patient data in my 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, I use the same elasticity parameter value for all the patients' prostate-surrounding tissue: 20kPa (Rosen et al., 2008). The result of my experiment is shown in Fig. 3.12. I 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; I use Pearson linear correlation and Spearman correlation to compare the significance of the measured vs. 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

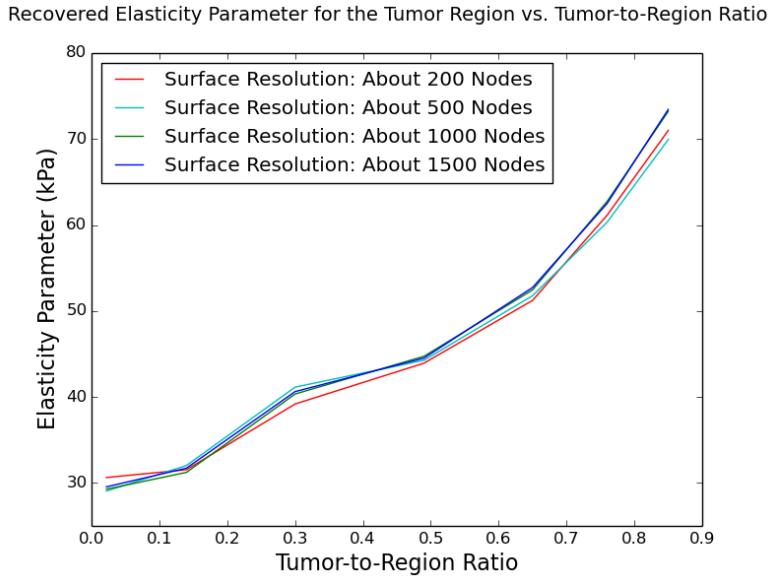


Figure 3.11: The Recovered Elasticity Parameter vs. Tumor-to-Region Ratio. This figure shows the recovered elasticity parameter for the tumor region using models with different mesh resolutions vs. the tumor-to-region ratio.

the prostates' elasticity parameters and their T-stages. The p-values computed from my experiments are one order smaller than that of previous work (Lee et al., 2012b); this indicates that my method is much more accurate.

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

Table 3.3: T-stages for prostate cancer definition

3.3.3 Performance Analysis for Quasi-Static Parameter Estimation

The model I use in my multi-region reconstruction validation experiment and cancer-stage correlation experiment consists of (on average) 4,000 tetrahedral elements. I run my experiment on

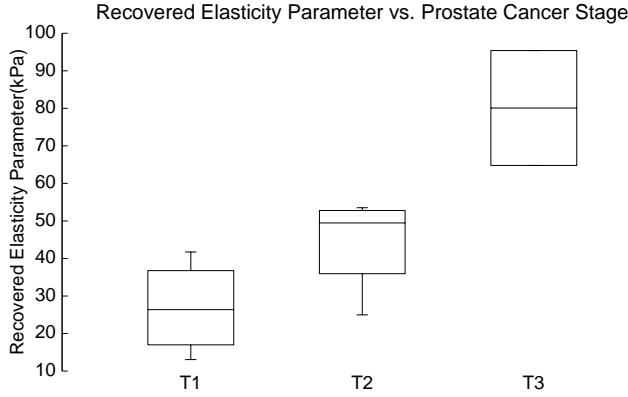


Figure 3.12: Box Plot of Estimated Average Elasticity Parameters. The estimated elasticity parameters $\bar{\mu}$ of the prostate of the ten patients vs. their cancer stages shows positive correlation.

a desktop with Intel(R) Core(TM) i7 CPU, 3.20GHz. With the Particle-Swarm Optimization (PSO) method, the entire parameter reconstruction process takes no more than 24 hours on a single core. I gain nearly linear performance scaling by paralleling the PSO method using OpenMP, as shown in Figure. 3.13. 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 (Lee et al., 2012b), my 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).

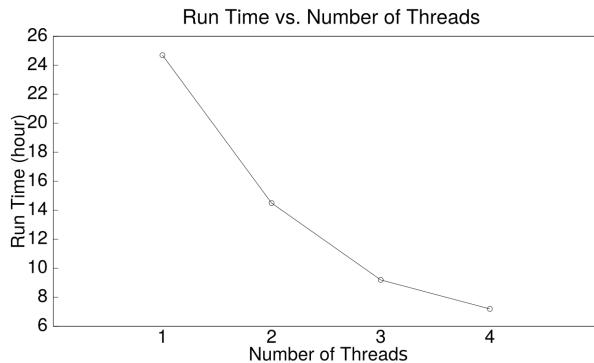


Figure 3.13: The Running Time of the Reconstruction Process vs. the Number of Threads Used. The running time decreases almost linearly with the increase of the number of thread.

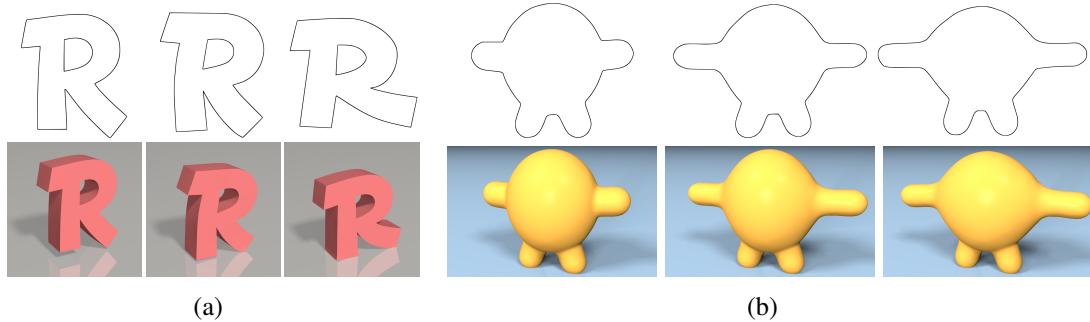


Figure 3.14: Animation from 2D Sketches. The three images in the first row are the 2D Sketches of three keyframes; the three images in the second row are the simulation result of the corresponding keyframes.

3.3.4 Applications

I demonstrate the application of my MaterialCloning algorithm on several different scenarios:

- 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. 1.2(a));
- A user poking and interacting with a ‘virtual prostate’ with patient-specific elasticity parameters that were automatically acquired from 2 sets of CT images of the patient on different days (see Fig. 1.2(b));
- ‘Animated letters’ (see Fig. 3.14(a)) and ’Animated figure’ (see Fig. 3.14(b)) are simulated from material parameters estimated directly from the artist’s sketches, which serve as the keyframes to an animation sequence. My system, combined with dynamic tracking mechanism presented in a technical report (Yang and Lin, 2015a), can be extended to create animations from time-sequential sketches.

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

3.3.5 Comparison with Other Approaches

It is difficult to compare my work to existing methods on parameter estimation for soft tissues (Pai et al., 2001; Becker and Teschner, 2007b; Syllebranque and Boivin, 2008a; Bickel et al., 2009,

2010), as my approach does not require force-measuring devices such as force sensors or trinocular stereo vision systems. In contrast to earlier methods, my 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, I 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, my 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.(Lee et al., 2012b). and in comparison to this work, my method shows much improved accuracy (see Sec. 3.3.2), as well as adds the ability to perform simultaneous parameter estimation for multiple materials. I use the Particle Swarm Optimization (PSO) method, which is more easily parallelizable than iterative optimization methods (Lee et al., 2012b) 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, I demonstrate that my method can also be used to estimate elasticity parameters with the artist’s sketches as keyframes to automate physics-based animation.

3.4 Conclusion and Future Work

In this chapter, I presented the *MaterialCloning* algorithm, which can automatically acquire multi-region elasticity parameters directly from two sets of multiple-view images. I validate its effectiveness using both synthetic datasets and real-world medical images.

The results of my 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. I also demonstrate my 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.

CHAPTER 4: VIDEO-BASED SOFT-BODY MECHANICAL PROPERTY RECOVERY

4.1 Introduction

Non-rigid materials are widely used in medical robotics, design and manufacturing, virtual surgery for soft robot planning, procedural rehearsal and training, etc. Identification of mechanical properties, such as tissue elasticity parameters, is critical to enable medical robots to safely operate within highly unstructured, deformable human bodies and to compute desired, accurate force feedback for individualized haptic display characterized by patient-specific parameters. In addition to medical robots, simulations are also increasingly used for rapid prototyping of clinical devices, pre-operation planning of medical procedures, virtual training exercises for surgeons and supporting personnel, etc. And, bio-tissue elasticity properties are central to developing realistic and predictive simulation and for designing responsive, dexterous surgical manipulators. Furthermore, with increasing interest in 3D printing for rapid creation of soft robots consisting of flexible materials, the ability to easily acquire material properties from existing sensor data, such as medical images and videos, can help to replicate similar material properties.

Traditional stress-strain matching methods have been practiced for decades to measure the mechanical properties of deformable bodies (Veronda and Westmann, 1970). Special external force measuring devices are often applied. Landmarks on the deformable bodies are usually used in these methods for measuring displacement fields. Following the wide adoption of ultrasound, elastography(Ophir et al., 1991b) emerged, which estimates relative elasticity properties by measuring both the deformation of the tissue via ultrasound images and explicitly measuring the external force using special devices. In the last decade or so, numerical methods, such as inverse Finite Element Methods (FEM) (Kauer et al., 2002b; Lee et al., 2012a), were proposed to estimate mechanical properties of deformable body without the measurement of the displacement field. But, these techniques are

generally limited to quasi-static deformation process. Human tissues, such as the heart, are often in a dynamic state.

In this chapter, I propose a novel Bayesian elasticity-parameter estimation algorithm for deformable bodies using temporal sequences of deformation samples. These deformation samples may be medical images, such as those from ultrasound and echocardiogram, videos, or hand sketches. The samples reflect the 4-dimensional (or 3D) behavior of the deformable bodies by providing a temporal sequence of 3D (or 2D) shapes in motion. Our method follows the iterative optimization from the Lavenberg-Marquardt Algorithm (Moré, 1978; Lourakis and Argyros, 2004). But, I address the problem using the Bayesian framework. Given the temporal sequences input, I adopt a probabilistic graphical model (Loague and Green, 1991; Madigan et al., 1995), the unscented Kalman Filter (Wan and Van Der Merwe, 2000), to continuously refine the estimated elasticity parameters by matching the simulated deformation (generated by FEM simulation) to the input temporal sequences of (captured) deformation samples. The geometric models of the deformable bodies are reconstructed from the given measurements, e.g. videos.

In contrast to the traditional strain-stress matching methods (Mehrabian and Samani, 2009; Ottensmeyer, 2001; Brouwer et al., 2001; Boonvisut and Cavusoglu, 2013), elastography methods (Rogowska et al., 2014; ?; Bilston and Tan, 2015), and the more recent inverse FEM (Kauer et al., 2002b; Lee et al., 2012a), I am able to recover the mechanical parameters of the deformable bodies by tracking and matching a temporal sequence of deformation samples with simulated deformation. Similar to the inverse FEM, our approach utilizes the implicitly known or computed external force as the boundary condition. But, I further maximize the utilization of the input video by constructing the posterior probability distribution of the estimated parameters using the given temporal sequence of deformation samples. I validate the effectiveness of our framework by recovering the mechanical parameters of a tennis ball and a foam ball from videos captured by high-speed cameras.

The key results of this work include

- An alternative approach to dynamically track the surface of a deformable body in motion;

- Reconstruction of non-rigid mechanical properties from temporal sequences of deformation samples using a probabilistic, graphical model.

The rest of this chapter is organized as follows. I describe our Bayesian framework for elasticity parameter estimation from temporal sequences of deformation samples in Section 4.2 and present the validation experiments and results in Section 4.3.

4.2 Method

Given a temporal sequence of deformation samples, such as ultrasound videos, high-speed camera videos, etc., the goal is to recover the mechanical properties of the non-rigid body in motion. Our framework uses the unscented Kalman Filter (Sec. 4.2.4) for the estimation of elasticity parameters and a tightly coupled module for hidden state estimation (Sec. 4.2.5) for deformable body undergoing motion. The input to our framework are the observed temporal sequence of deformation samples and the computed boundary condition. Our framework then computes the optimized the mechanical properties and the tracked surfaces of the deformable body in motion. The overview of our framework is given in Fig. 6.2.

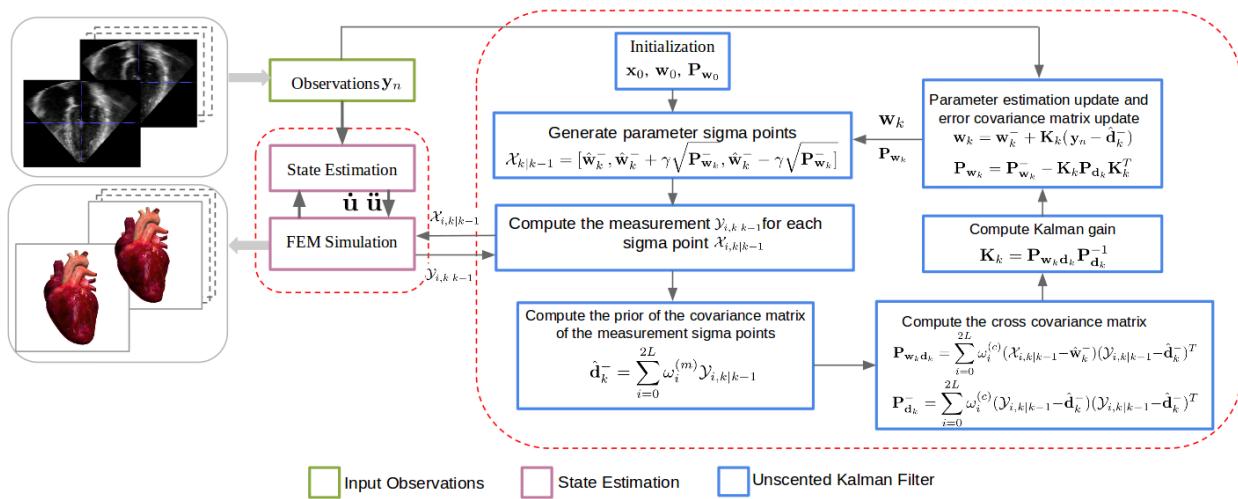


Figure 4.1: The Flow Chart of Our Framework. Our framework takes a temporal sequence of deformation samples as the input. The UKF takes in the observations and drives the finite element simulation by optimizing both the hidden parameter and hidden states.

4.2.1 Generalized Dynamic Process and Decoupled State Parameter Estimation

I first give an abstract model for the general dynamic process. The discrete-time dynamic deformation process can be described by a dynamic state-space model. I will use subscript k for the k^{th} iteration for the dynamic process and subscript n for the n^{th} deformation sample in the sequence. The model has the hidden system state \mathbf{x}_k with noise, the corresponding observation \mathbf{y}_k with noise, the hidden parameter \mathbf{w}_k , an update function $\mathbf{f}(\mathbf{x}; \mathbf{w})$ and an observation function $\mathbf{h}(\mathbf{x}; \mathbf{w})$. The model is given as:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}; \mathbf{w}) \quad (4.1)$$

$$\mathbf{y}'_k = \mathbf{h}(\mathbf{x}_k; \mathbf{w}) \quad (4.2)$$

\mathbf{y}'_k equals to \mathbf{y}_k when the hidden state and the hidden parameters are optimized.

When applied to our discrete-time dynamic deformation process, the hidden system state \mathbf{x} consists of the displacement \mathbf{u} , the velocity $\dot{\mathbf{u}}$ and the acceleration $\ddot{\mathbf{u}}$.

$$\mathbf{x} = [\mathbf{u} \ \dot{\mathbf{u}} \ \ddot{\mathbf{u}}] \quad (4.3)$$

The observation \mathbf{y} is the key frame configurations which is the displacement of the key frame deformations. The update function \mathbf{f} is the integrator. For different time discretization method, the update function \mathbf{f} is different. The observation function \mathbf{h} is the transformation matrix from \mathbf{x} to \mathbf{y}' with noise added.

Our dynamic elasticity parameters estimation problem can be stated as using the above dynamic state-space model for estimation of hidden system variables (including *states* and *parameters*) based on the keyframe observations.

4.2.2 Deformable Body Modeling

I use Finite Element Method (FEM) for the spacial integration in deformable body simulation. I use incompressible, linear stress-strain model in this work. For linear stress-strain model, it assumes

that the relation between stress and strain can be represented by a linear function. I apply the Lamé material with the Lamé's first and second parameter computed from the Young's modulus E and the Poisson's ratio ν . Lamé material model is a widely used simple linear material model. Though the linear material model is not accurate to approximate large deformations, for our application, it is accurate enough to identify different mechanical properties of various materials. Because I assume incompressibility, I use 0.45 as the Poisson's ratio. The Young's modulus E is the parameter I will optimize using our framework. FEM combined with the Newmark implicit time integration method forms the update function Eqn. 4.1

4.2.3 Bayesian Parameter Estimation

For Bayesian parameter estimation, we estimate the posterior probability distribution of the parameters \mathbf{w}_k ,

$$p(\mathbf{w}_k | \mathbf{y}_{1:n}) \quad (4.4)$$

based on the observations,

$$\mathbf{y}_{1:n} = \mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n \quad (4.5)$$

To motivate our reason for choosing unscented Kalman Filter as the estimation method, we will first give a brief introduction of the Recursive Bayesian Estimation algorithm. Recursive Bayesian estimation algorithm filters the posterior probability density function recursively for new observations. According to Bayes' theorem, the posterior probability density can be expressed as,

$$p(\mathbf{w}_k | \mathbf{y}_{1:n}) = \frac{p(\mathbf{y}_n | \mathbf{w}_k) p(\mathbf{w}_k | \mathbf{y}_{1:n-1})}{p(\mathbf{y}_n | \mathbf{y}_{1:n-1})} \quad (4.6)$$

We will not give details on every term in Eqn. 4.6. We focus on the hidden parameter transition prior $p(\mathbf{w}_k | \mathbf{w}_{k-1})$ and the observation likelihood densities $p(\mathbf{y}_n | \mathbf{w}_k)$. The computation of the hidden parameter transition prior $p(\mathbf{w}_k | \mathbf{w}_{k-1})$ depends on the update function $\mathbf{f}(\mathbf{x}; \mathbf{w})$. In our application, the state update function $\mathbf{f}(\mathbf{x}; \mathbf{w})$ is nonlinear. Thus we need an approximation method.

The basic Kalman Filter models the densities in Eqn. 4.6 by Gaussian distributions. And it assumes that the state posterior probabilistic density can consistently be minimized by updating only the first (mean) and second order moments (covariance) of the true probabilistic densities. The extended Kalman Filter applies the basic Kalman Filter to nonlinear dynamic state-space system by first linearize it using Taylor series. This linearization affects the accuracy of the estimation process. The unscented Kalman Filter approaches the nonlinear problem differently. Instead of linearizing the system using Taylor series, it uses a general deterministic sampling framework, or the sigma-point approach. Thus we choose to use unscented Kalman Filter as our basic estimator for our dynamic elasticity-parameter estimation problem.

4.2.4 Unscented Kalman Filter for Parameter Estimation

The unscented Kalman Filter (Wan and Van Der Merwe, 2000) handles the nonlinear problem with the idea that it is easier to approximate a random variable than a non-linear function. Like both basic Kalman filter and extended Kalman filter, the unscented Kalman filter consists of two steps: the prediction step and the correction step. In order to do parameter estimation, we first construct a mapping between the measured observation y' , state x and the parameters w . In the following, we will use d for y' . The observation y equals to the measured observation d when the state x and the parameters w are optimized. We define a function g as follows,

$$d_k = g(x_k, w_k) \quad (4.7)$$

We will use superscript minus for prior probabilistic densities, P for covariance matrix, hat symbol for the mean of the random variable, subscript k for the current estimation iteration and L as the size of the parameter vector w .

1. Initialize the mean of the parameters and the covariance matrix of the parameters

$$\hat{w}_0 = E[w_0] \quad (4.8)$$

with \mathbf{w}_0 as the initial guess of the parameters

$$\mathbf{P}_{\mathbf{w}_0} = E[(\mathbf{w}_0 - \hat{\mathbf{w}}_0)(\mathbf{w}_0 - \hat{\mathbf{w}}_0)^T] \quad (4.9)$$

2. For each estimation iteration k : Compute the prior of the mean of the parameters and the prior of the covariance of the parameters as,

$$\hat{\mathbf{w}}_k^- = \hat{\mathbf{w}}_{k-1} \quad (4.10)$$

$$\mathbf{P}_{\mathbf{w}_k}^- = \mathbf{P}_{\mathbf{w}_{k-1}} \quad (4.11)$$

Select a set of sigma points $\mathcal{X}_{k|k-1}$. The columns of the matrix $\mathcal{X}_{k|k-1}$ are the sampled parameters.

$$\mathcal{X}_{k|k-1} = [\hat{\mathbf{w}}_k^-, \hat{\mathbf{w}}_k^- + \gamma \sqrt{\mathbf{P}_{\mathbf{w}_k}^-}, \hat{\mathbf{w}}_k^- - \gamma \sqrt{\mathbf{P}_{\mathbf{w}_k}^-}] \quad (4.12)$$

where $\gamma = \sqrt{L + \lambda}$. The matrix expands as,

$$\begin{aligned} \mathcal{X}_{0,k|k-1} &= \hat{\mathbf{w}}_k & i = 0 \\ \mathcal{X}_{i,k|k-1} &= \hat{\mathbf{w}}_k + (\gamma \sqrt{\mathbf{P}_{\mathbf{w}_k}^-})_i & i = 1, \dots, L \\ \mathcal{X}_{i,k|k-1} &= \hat{\mathbf{w}}_k - (\gamma \sqrt{\mathbf{P}_{\mathbf{w}_k}^-})_i & i = L + 1, \dots, 2L \end{aligned} \quad (4.13)$$

The weights $\omega^{(m)}$ for computing the mean of the sigma-points and the weights $\omega^{(c)}$ for computing the covariance matrix of the sigma-points are,

$$\begin{aligned} \omega_0^{(m)} &= \frac{\lambda}{L+\lambda} & i = 0 \\ \omega_0^{(c)} &= \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta) & i = 0 \\ \omega_i^{(m)} &= \omega_i^{(c)} = \frac{1}{2(L+\lambda)} & i = 1, \dots, 2L \end{aligned} \quad (4.14)$$

$$\sum_{i=0}^{i=2L} \omega_i^{(c)} = 1, \quad \sum_{i=0}^{i=2L} \omega_i^{(m)} = 1 \quad (4.15)$$

where α and β are two tuned parameters for the filter. α affects the distribution of the sigma-points. The distribution of the sigma-points can affect the convergence rate to some extent. And β controls the tails of the posterior distribution. In our experiments we set $\beta = 2$ and α varies from 0.1 to 2. Then compute the measured observation sigma points of each element of the sigma points matrix $\mathcal{X}_{k|k-1}$,

$$\mathcal{Y}_{k|k-1} = \mathbf{g}(\mathbf{x}_k, \mathcal{X}_{k|k-1}) \quad (4.16)$$

the prior of the mean of the measurements \mathbf{d}_k^- ,

$$\hat{\mathbf{d}}_k^- = \sum_{i=0}^{2L} \omega_i^{(m)} \mathcal{Y}_{i,k|k-1} \quad (4.17)$$

the prior of the covariance matrix of the measurements $\mathbf{P}_{\mathbf{d}_k}^-$,

$$\mathbf{P}_{\mathbf{d}_k}^- = \sum_{i=0}^{2L} \omega_i^{(c)} (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{d}}_k^-) (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{d}}_k^-)^T + \mathbf{R}_{\mathbf{e}_k} \quad (4.18)$$

the cross covariance matrix for the parameter and the measurement,

$$\mathbf{P}_{\mathbf{w}_k \mathbf{d}_k} = \sum_{i=0}^{2L} \omega_i^{(c)} (\mathcal{X}_{i,k|k-1} - \hat{\mathbf{w}}_k^-) (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{d}}_k^-)^T \quad (4.19)$$

Compute the Kalman gain \mathbf{K}_k ,

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{w}_k \mathbf{d}_k} \mathbf{P}_{\mathbf{d}_k}^{-1} \quad (4.20)$$

Compute the posterior of the parameter and the posterior covariance of the parameter

$$\mathbf{w}_k = \mathbf{w}_k^- + \mathbf{K}_k (\mathbf{y}_n - \hat{\mathbf{d}}_k^-) \quad (4.21)$$

$$\mathbf{P}_{\mathbf{w}_k} = \mathbf{P}_{\mathbf{w}_k}^- - \mathbf{K}_k \mathbf{P}_{\mathbf{d}_k} \mathbf{K}_k^T \quad (4.22)$$

4.2.5 Coupled State Estimation

The hidden state variables for our dynamical system include the positions, the velocities, and the accelerations. Though the unscented Kalman Filter can be used for dual state-parameter estimation, the fact that the state and the parameters are coupled makes such a dual estimation fail to converge. Thus we use the finite difference method to estimate both the initial velocities and the accelerations. Given the observations of three frames \mathbf{y}_{n-1} , \mathbf{y}_n , \mathbf{y}_{n+1} and the time elapsed between the frames Δt , the velocities $\dot{\mathbf{u}}$,

$$\dot{\mathbf{u}} = \frac{\mathbf{y}_n - \mathbf{y}_{n-1}}{\Delta t} \quad (4.23)$$

the accelerations $\ddot{\mathbf{u}}$,

$$\ddot{\mathbf{u}} = \frac{\mathbf{y}_{n+1} - 2\mathbf{y}_n + \mathbf{y}_{n-1}}{(\Delta t)^2} \quad (4.24)$$

Our dynamic parameter estimation scheme is given in Algorithm 2.

4.3 Experiments

I have implemented our algorithms in C/C++ and have validated the results by first conduct a synthetic heart experiment. In this experiment, I use 3D meshes from simulation as the deformation samples. I also validate our framework by reconstructing the Young's moduli of a tennis ball and of a foam ball from videos captured by high-speed cameras.

4.3.1 Synthetic Heart Experiment

In this experiment, I recover the Young's modulus for a human heart from synthetic simulation results. I use 3D heart model reconstructed from a live patient ultrasound images, then simulate it with known Young's modulus and the computed contraction force using FEM. A human heart function by the contraction of the heart muscle of the right ventricle and right ventricle (shown in Fig. 4.2) to pump out blood. In this experiment, I will focus on the left and the right ventricles, because one of the main causes for diastolic dysfunction is due to the stiffening of muscles of the left or the right ventricle. I reconstructed our synthetic heart model from 3D ultrasound images (Bernard

Algorithm 2 Dynamic Elasticity Parameter Estimation

```
1: procedure MEASUREMENT COMPUTATION
2:   // this is the function g in Eqn. 8.
3:    $\dot{\mathbf{u}} \leftarrow (\mathbf{y}_n - \mathbf{y}_{n-1})/\Delta t$ 
4:    $\ddot{\mathbf{u}} \leftarrow (\mathbf{y}_{n+1} - 2\mathbf{y}_n + \mathbf{y}_{n-1})/(\Delta t)^2$ 
5:    $\mathbf{d} \leftarrow \mathbf{f}(\mathbf{x}; \mathbf{w})$  // evaluate measurement based on the parameters and the states
6: end Measurement Computation;
7: procedure UNSCENTED KALMAN FILTER ITERATE
8:    $\mathcal{X}_{k|k-1} \leftarrow [\hat{\mathbf{w}}_k^-, \hat{\mathbf{w}}_k^- + \gamma \sqrt{\mathbf{P}_{\mathbf{w}_k^-}}, \hat{\mathbf{w}}_k^- - \gamma \sqrt{\mathbf{P}_{\mathbf{w}_k^-}}]$ 
9:   // compute the sigma-points
10:   $\mathcal{Y}_{k|k-1} \leftarrow \mathbf{g}(\mathbf{x}_n, \mathbf{w}_k^-)$  // compute the prior of the mean of the measurement sigma-points
11:   $\hat{\mathbf{d}}_k^- \leftarrow \sum_{i=0}^{2L} \omega_i^{(m)} \mathcal{Y}_{i,k|k-1}$  // compute the prior of the covariance matrix of the measurement
    sigma-points
12:   $\mathbf{P}_{\mathbf{w}_k \mathbf{d}_k} \leftarrow \sum_{i=0}^{2L} \omega_i^{(c)} (\mathcal{X}_{i,k|k-1} - \hat{\mathbf{w}}_k^-) (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{d}}_k^-)^T$  // compute the cross covariance matrix
    of the measurements and the parameters
13:   $\mathbf{K}_k \leftarrow \mathbf{P}_{\mathbf{w}_k \mathbf{d}_k} \mathbf{P}_{\mathbf{d}_k}^{-1}$  // compute the Kalman gain
14:   $\hat{\mathbf{w}}_k \leftarrow \mathbf{w}_k^- + \mathbf{K}_k (\mathbf{y}_n - \hat{\mathbf{d}}_k^-)$ 
15:  // update the mean of the parameters
16:   $\mathbf{P}_{\mathbf{w}_k} \leftarrow \mathbf{P}_{\mathbf{w}_k}^- - \mathbf{K}_k \mathbf{P}_{\mathbf{d}_k} \mathbf{K}_k^T$ 
17:  // update the covariance matrix of the parameters
18: end Unscented Kalman Filter Iterate;
19: procedure MAIN
20:   while not converged do
21:     for Sampled Keyframes  $\mathbf{y}_{n_i}$  do
22:       Initialize  $\mathbf{x}_0, \mathbf{w}_0, \mathbf{P}_{\mathbf{w}_0}$ 
23:       Unscented Kalman Filter Iterate()
24:     end for;
25:   end while;
26: end Main;
```

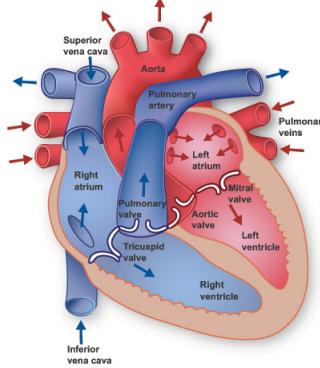


Figure 4.2: The human heart anatomy. In this project I model the left and right ventricle. © Texas Heart Institute (Institute, 1996)

et al., 2014) (shown in Fig. 4.3(a)- 4.3(b)) using ITK-SNAP (Yushkevich et al., 2006a) to manually segment both the left and right ventricles (shown in Fig. 4.3(c)). I use Tetgen (Si, 2015) to discretize the surface mesh for FEM simulation (shown in Fig. 4.3(d)). Then I use our statistical optimization framework to estimate both the Young's modulus and the state of the synthetic heart.

I compute the heart contraction force based on: a) the assumed Young's modulus; b) the statistics that by the end of the systole phase the volume of the heart reduced by around 40%. The computed heart contraction force with the model's Young's modulus being 40KPa is shown in Fig. 4.4. It ranges from 8.633mN to 37.40mN. This contraction force will serve as the boundary condition for solving the governing equation Eqn. 3.1. The observed deformation samples I feed into our framework are the 3D meshes from the synthetic simulation (shown in Fig. ??). I use deformation samples of two time stamp: a) the end of systole phase (shown in Fig. 4.4(c)) b) the end of diastole phase (shown in Fig. 4.4(d)).

The relative error for the recovered Young's modulus is within 7.5% of the ground-truth values, as shown in Table ???. The initial guess on the Young's modulus can contribute upto about 5% of relative error in the recovered value. I also measure the distance between the surface with optimized Young's modulus and the given reference surface using level set as another metric for validation. Our method can reduce the surface-tracking error down to less than 5%, as shown in Table ??, where the initial average separation distance between surfaces is 3.2mm. The average surface distance error between the tracked surface and the reference surface is shown in Table ???. These results

show that our method can also be used to track the surfaces of the non-rigid bodies in the temporal sequences of deformation samples. The convergence graphs of the optimization process are shown in Fig. 4.5.

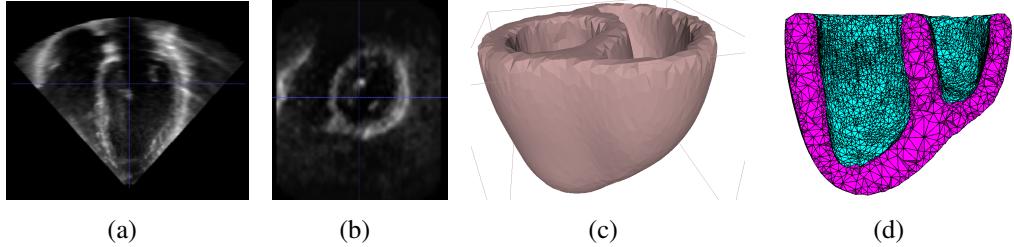


Figure 4.3: Reconstructed left ventricle and right ventricle of human heart. (a)-(b) the slices of a human heart ultrasound image © CETUS 2014 (Bernard et al., 2014), (c) the surface mesh of the reconstructed model, (d) the sliced view of the tetrahedra mesh from the surface mesh in (b).

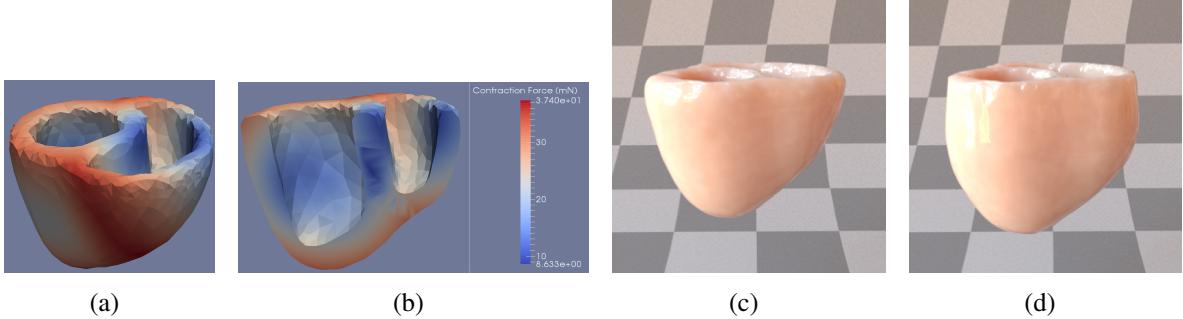


Figure 4.4: The computed contraction force and synthetic heart simulation result. (a) the visualization of the contraction force on the surface of the 3D heart model using cool to hot color map, (b) the sliced view of the contraction force. (c) the simulated heart model by the end of systole phase, (d) the simulated heart model by the end of diastole phase.

Possible Sources of Errors: 1) error introduced by the estimation of the velocities; 2) error introduced by incomplete information from the key frames; 3) error introduced by the sensitivity between the elasticity parameter and the amount of deformation.

4.3.2 Mechanical Parameters Recovered from Videos

I validate our framework further by recovering mechanical parameters of a tennis ball and a foam ball from videos captured by high-speed cameras. First, I reconstruct the 3D mesh from the video. I deduce 2D shape of the object from the video. Then I reconstruct the 3D shape by rotating the 2D

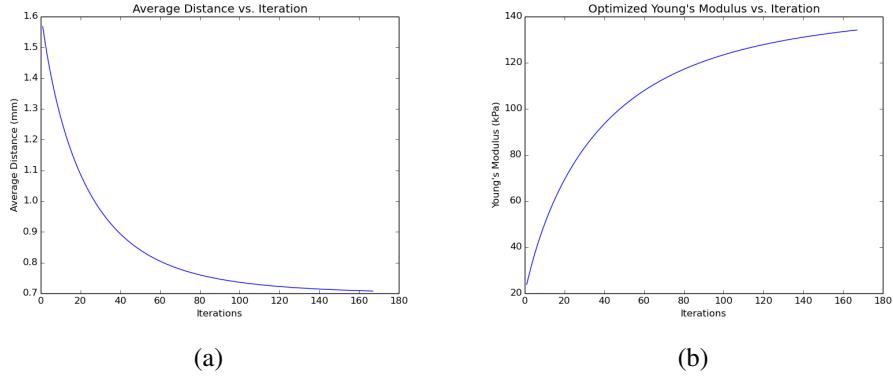


Figure 4.5: The convergence graphs for synthetic heart experiment.(a) shows our framework reduces the distance between the surface with the optimized material parameter and the reference surface, as the framework iterates with the initial distance error at 3.2mm, (b) shows the convergence of the Young's modulus to the ground truth.

Table 4.1: The result for the synthetic heart experiment with noises in the initialization. The relative error of the recovered Young's modulus is within 7.5% of the ground-truth values. Our method reduces the average surface-tracking error down to less than 5%.

Synthetic Young's Modulus (kPa)	40	80	100	140	180
Initialized Young's Modulus (kPa)	20 ± 10	40 ± 10	50 ± 10	70 ± 20	90 ± 20
Recovered Young's Modulus (kPa)	37 ± 2	76 ± 3	93 ± 4	134 ± 7	167 ± 10
Average Relative Error (%)	7.5	5	7	4.3	7.2
Standard Deviation of the Recovered Young's Modulus	1.41	2	3.2	7.5	10.2
Average Surface Distance Error (mm)	0.2 ± 0.1	0.3 ± 0.1	0.3 ± 0.1	0.5 ± 0.2	0.7 ± 0.2
Average Relative Tracking Error (%)	1.3	2.0	2.0	3.3	4.7
Contraction Force Range (mN)	8-37	17-80	17-85	25-110	30-170

shape (on the x,y plane) around the y-axis. Then I use our framework for estimation of non-rigid mechanical parameters. From the above synthetic experiment, I know the initialization can affect the accuracy of the method. I choose to initialize the Young's modulus after doing a line search. This process can greatly reduce the optimization iterations and increase the speed as well.

The boundary condition for both the videos are the gravity. The average density of a tennis ball is $0.4\text{g}/\text{cm}^3$ and that of a foam ball is around $30\text{g}/\text{cm}^3$.

From the tennis ball video (clips shown in 4.6(a)- 4.6(e)), I recovered the Young's modulus to be 2.2-2.5MPa. The surface meshes of the model with optimized mechanical parameters and states are shown in 4.6(f)- 4.6(j). The Young's modulus I recovered is within the range of the values

measured in the work (Sissler et al., 2010; Wójcicki et al., 2011). The estimated velocities of the tennis ball (before it hit the flat surface) from the video is about 20m/s. For the foam ball, the Young's modulus I recovered is about 15-21KPa. According to the video description, the foam ball in the video (clips shown in 4.7(a)- 4.7(e)) is a stress relief foam ball. This type of foam ball is made of polyurethane foam. Our estimated parameter is well within the range presented in the literature (Moore et al., 2007). One of the sources of error is the estimated initial value to the state

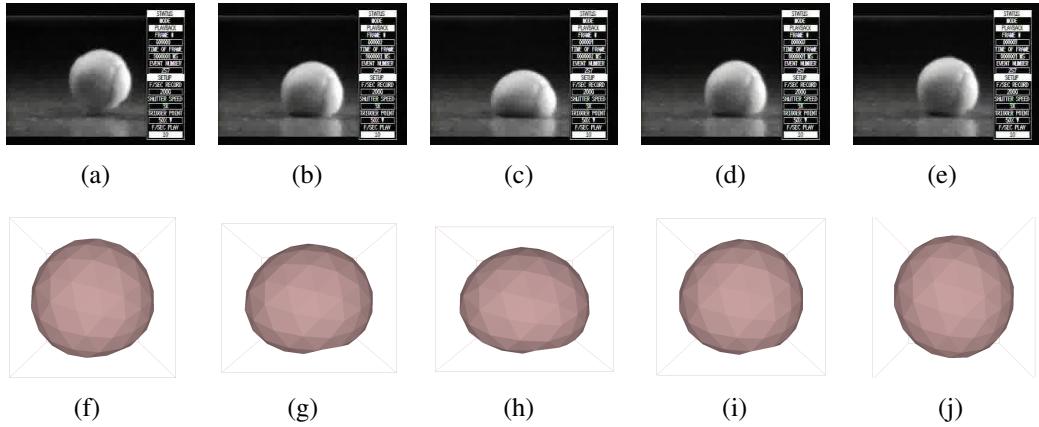


Figure 4.6: Input tennis video clips and the tracked surface mesh. (a)-(e) the clips of the tennis video © Trevor Shannon (Shannon, 2009); (f)-(j) the tracked surface mesh at the corresponding time stamp.

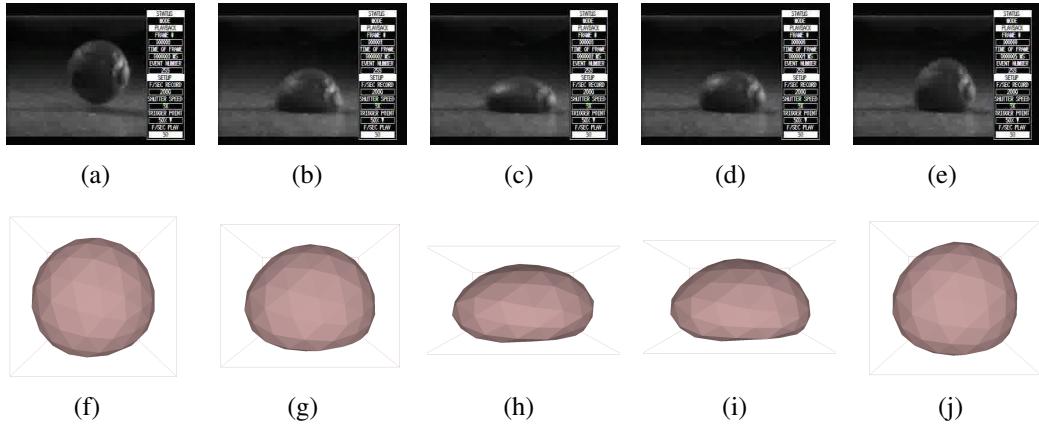


Figure 4.7: Input foam ball video clips and the tracked surface mesh. (a)-(e) the clips of the tennis video © Trevor Shannon (Shannon, 2009); (f)-(j) the tracked surface mesh at the corresponding time stamp.

estimation. Currently I estimated the initial velocity with a finite difference technique using the first

couple video frames. I conducted a study using the tennis ball data with different velocity magnitude before hitting the surface. The initial velocities could affect the recovered Young's modulus shown in Table ???. One reason for how much it affects the recovered Young's modulus is the size of the object. The impact of initial-value estimation is reduced as the size of the object increases.

I also compare our method with the most recent work by Wang et al. (Wang et al., 2015). They also use temporal sequences of deformation samples as input, but they assume the deformable objects are in static state. As is shown in Table ???, our results are much closer to the measured parameters from real-world experiments. Please see the supplementary video for demonstration of these experiments.

Table 4.2: **Impact of initial velocities on recovered Young's moduli.**

Initial Vel Mag (m/s)	0	0.01	20
Recovered E (kPa)	$5.1\text{-}5.9$	$6.2e^2\text{-}6.9e^2$	$2.2e^3\text{-}2.5e^3$

Table 4.3: **Comparison with results of (Wang et al., 2015) and experimental measurements.** The measured Young's modulus for the tennis ball is taken from (Sissler et al., 2010; Wójcicki et al., 2011) and for the foam ball is derived from (Moore et al., 2007). The recovered Young's modulus using our method is within the range of the Young's modulus measured, the Young's modulus recovered using (Wang et al., 2015) is not.

	Tennis Ball	Foam Ball
Wang et al. (Wang et al., 2015) E (KPa)	$5.1e^0\text{-}5.9e^0$	$1e^{-1}\text{-}2e^{-1}$
Our method E (KPa)	$2.2e^3\text{-}2.5e^3$	$1.2e^1\text{-}2.1e^1$
Measured E (KPa)	$1e^3\text{-}5e^3$	$1e^1\text{-}3e^1$

4.4 Conclusion and Future Work

In this chapter, I presented a Bayesian parameter estimation framework for dynamic deformable bodies. Our inputs are temporal sequences of deformation samples, such as simulation results or videos from ultrasound and high-speed cameras. This approach uses the Unscented Kalman Filter for hidden parameter estimation to recover the Young's modulus, tightly coupled with a simulation-based state estimation for surface tracking/matching. The results from synthetic experiments and

validation using real-life videos demonstrate the effectiveness of the proposed method in estimating the mechanical properties of the deformable bodies in motion.

For nonrigid bodies, state estimation still remains a challenging problem. The boundary conditions in our framework are known or can be computed. In many real-life situations, the boundary conditions can be complicated to derive. Detailed analysis and study on the impact of the boundary conditions would be possible avenues for future research, when applying this framework to different problem domains.

CHAPTER 5: CLASSIFICATION OF PROSTATE CANCER GRADES AND T-STAGES BASED ON TISSUE ELASTICITY USING MEDICAL IMAGE ANALYSIS

5.1 Introduction

Currently screening of prostate cancers is usually performed through routine prostate-specific antigen (PSA) blood tests and/or a rectal examination. Based on positive PSA indication, a biopsy of randomly sampled areas of the prostate can then be considered to diagnose the cancer and assess its aggressiveness. Biopsy may miss sampling cancerous tissues, resulting in missed or delayed diagnosis, and miss areas with aggressive cancers, thus under-staging the cancer and leading to under-treatment.

Studies have shown that the tissue stiffness described by the tissue properties may indicate abnormal pathological process. Ex-vivo, measurement-based methods, such as (Ashab et al., 2015; Khojaste et al., 2015) using magnetic resonance imaging (MRI) and/or ultrasound, were proposed for study of prostate cancer tissue. However, previous works in material property reconstruction often have limitations with respect to their genericity, applicability, efficiency and accuracy (Yang and Lin, 2015b). More recent techniques, such as inverse finite-element methods (Goksel et al., 2013; Lee et al., 2012b; Shahim et al., 2013; Vavourakis et al., 2016; Yang and Lin, 2015b), stochastic finite-element methods (Shi and Liu, 2003), and image-based ultrasound (Uniyal et al., 2014) have been developed for in-vivo soft tissue analysis.

In this chapter, I study the possible use of tissue (i.e. prostate) elasticity to help evaluate the prognosis of prostate cancer patients given at least two set of CT images. The clinical T-stage of a prostate cancer is a measure of how much the tumor has grown and spread; while a Gleason score based on the biopsy of cancer cells indicates aggressiveness of the cancer. They are commonly used for cancer staging and grading. I present an improved method that uses geometric and physical

constraints to deduce the relative tissue elasticity parameters. Although elasticity reconstruction, or elastography, can be used to estimate tissue elasticity, it is less suited for in-vivo measurements or deeply seated organs like prostate. I describe a non-invasive method to estimate tissue elasticity values based on pairs of CT images, using a finite-element based biomechanical model derived from an initial set of images, local displacements, and an optimization-based framework.

Given the recovered tissue properties reconstructed from analysis of medical images and patient's ages, I develop a multiclass classification system for classifying clinical T-stage and Gleason scores for prostate cancer patients. I demonstrate the feasibility of a statistically-based multiclass classifier that classifies a supplementary assessment on cancer T-stages and cancer grades using the computed elasticity values from medical images, as an additional clinical aids for the physicians and patients to make more informed decision (e.g. more strategic biopsy locations, less/more aggressive treatment, etc). Concurrently, extracted image features (Haq et al., 2015b, 2014, 2015a) using dynamic contrast enhanced (DCE) MRI have also been suggested for prostate cancer detection. These methods are complementary to ours and can be used in conjunction with ours as a multimodal classification method to further improve the overall classification accuracy.

5.2 Method

Our iterative simulation-optimization-identification framework consists of two alternating phases: the forward simulation to estimate the tissue deformation and inverse process that refines the tissue elasticity parameters to minimize the error in a given objective function. The input to our framework are two sets of 3D images. After iterations of the forward and inverse processes, I obtain the best set of elasticity parameters. Below I provide a brief overview of the key steps in this framework and I refer the interested readers to the supplementary document at <http://gamma.cs.unc.edu/CancerClass/> for the detailed mathematical formulations and algorithmic process to extract the tissue elasticity parameters from medical images.

5.2.1 Forward Simulation: BioTissue Modeling

In our system, I apply Finite Element Method (FEM) and adopt Mooney Rivlin material for bio-tissue modeling (?). After discretization using FEM, I arrive at a linear system,

$$\mathbf{K}\mathbf{u} = \mathbf{f} \quad (5.1)$$

with \mathbf{K} as the stiffness matrix, \mathbf{u} as the displacement field and \mathbf{f} as the external forces. The stiffness matrix \mathbf{K} is not always symmetric positive definite due to complicated boundary condition. The boundary condition I applied is the traction forces (shown in Fig. 7(a) of the supplementary document) computed based on the displacement of the surrounding tissue (overlapping surfaces shown in Fig. 7(b) of the supplementary document). I choose to use the Generalized Minimal Residual (GMRES) (Saad and Schultz, 1986) solver to solve the linear system instead of the Generalized Conjugate Gradient (GCG) (Liu and Storey, 1991), as GMRES can better cope with non-symmetric, positive-definite linear system.

The computation of the stiffness matrix \mathbf{K} in Eqn. 5.1 depends on the energy function Ψ of the Mooney Rivlin material model (Rivlin and Saunders, 1951; Treloar et al., 1976).

$$\Psi = \frac{1}{2}\mu_1((\mathbf{I}_1^2 - \mathbf{I}_2)/\mathbf{I}_3^{\frac{2}{3}} - 6) + \mu_2(\mathbf{I}_1/\mathbf{I}_3^{\frac{1}{3}} - 3) + v_1(\mathbf{I}_3^{\frac{1}{2}} - 1)^2, \quad (5.2)$$

where μ_1 , μ_2 and v_1 are the material parameters. In this work, I recover parameters μ_1 and μ_2 . Since prostate soft tissue (without tumors) tend to be homogenous, I use the average $\bar{\mu}$ of μ_1 and μ_2 as our recovered elasticity parameter. To model incompressibility, I set v_1 to be a very large value ($1 + e7$ was used in our implementation). v_1 is linearly related to the bulk modulus. The larger the bulk modulus, the more incompressible the object.

Relative Elasticity Value: In addition, I divide the recovered absolute elasticity parameter $\bar{\mu}$ by the that of the surrounding tissue to compute the relative elasticity parameter $\hat{\mu}$. This individualized relativity value helps to remove the variation in mechanical properties of tissues between patients,

normalizing the per-patient fluctuation in absolute elasticity values due to varying degrees of hydration and other temporary factors. I refer readers to our supplementary document for details regarding non-linear material models.

5.2.2 Inverse Process: Optimization for Parameter Identification

To estimate the patient-specific relative elasticity, our framework minimizes the error due to approximated parameters in an objective function. Our objective function as defined in Eqn. 5.3 consists of the two components. The first part is the difference between the two surfaces – one reconstructed from the reference (initial) set of images, deformed using FEM simulation with the estimated parameters toward the target surface, and one target surface reconstructed from the second set of images. This difference is measured by the Hausdorff distance (Dobuisson and Jain, 1994). In addition I add a Tikhonov regularization (Engl et al., 1989; Golub et al., 1999) term, which improves the conditioning of a possibly ill-posed problem.

With regularization, our objective function is given as:

$$\boldsymbol{\mu} = \underset{\boldsymbol{\mu}}{\operatorname{argmin}} \sum \| \mathbf{d}(\mathbf{S}_l, \mathbf{S}_t) \|^2 + \lambda \Gamma \mathbf{S}_l, \quad (5.3)$$

with $\mathbf{d}(\mathbf{S}_l, \mathbf{S}_t)$ as the distance between deformed surface and the reference surface, λ as the regularization weight, and Γ as the second-order differentiatial operator.

The second-order differential operator Γ on a continuous surface (2-manifolds) \mathbf{S} is the curvatures of a point on the surface. The curvature is defined through the tangent plane passing that point. I denote the normal vector of the tangent plane as \mathbf{n} and the unit direction in the tangent plane as \mathbf{e}_θ . The curvature related to the unit direction \mathbf{e}_θ is $\kappa(\theta)$. The mean curvature κ_{mean} for a continuous surface is defined as the average curvature of all the directions, $\kappa_{mean} = \frac{1}{2\pi} \int_0^{2\pi} \kappa(\theta) d\theta$. In our implementation, I use triangle mesh to approximate a continuous surface. I use the 1-ring neighbor as the region for computing the mean curvature normal on our discrete surface \mathbf{S}_l . I treat each

triangle of the mesh as a local surface with two conformal space parameters u and v . With these two parameters u and v the second-order differential operator Γ on vertex \mathbf{x} is, $\Delta_{u,v}\mathbf{x} = \mathbf{x}_{uu} + \mathbf{x}_{vv}$.

5.2.3 Classification Methods

For classification of cancer prognostic scores, I develop a learning method to classify patient cancer T-Stage and Gleason score based on the relative elasticity parameters recovered from CT images. Both the prostate cancer T-stage and the Gleason score are generally considered as ordinal responses. I study the effectiveness of ordinal logistic regression (Bender and Grouven, 1997) and multinomial logistic regression (Kleinbaum and Klein, 2010) in the context of prostate cancer staging and grading. For both cases I use RBF kernel to project our feature to higher dimensional space. I refer readers to supplementary document for method details and the comparison with the Random Forests method.

5.3 Results

5.3.1 Preprocessing and Patient Dataset

Given the CT images (shown in Fig. 5.1(a)) of the patient, the prostate, bladder and rectum are first segmented in the images. Then the 3D surfaces (shown in Fig. 5.1(b)) of these organs are reconstructed using VTK and these surfaces would be the input to our elasticity parameter reconstruction algorithm. Our patient dataset contains 113 (29 as the reference and 84 as target) sets of CT images from 29 patients, each patient having 2 to 15 sets of CT images. Every patient in the dataset has prostate cancer with cancer T-stage ranging from T1 to T3, Gleason score ranging from 6 to 10, and age from 50 to 85. Gleason scores are usually used to assess the aggressiveness of the cancer.

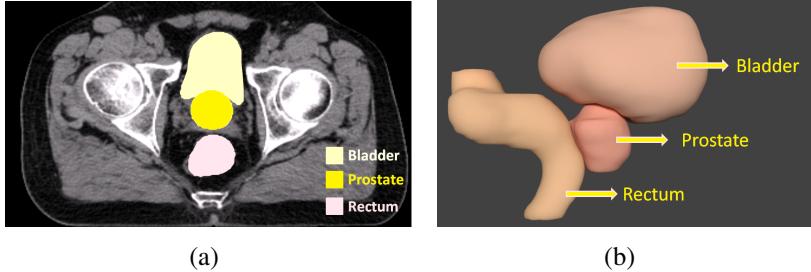


Figure 5.1: Real Patient CT Image and Reconstructed Organ Surfaces. (a) shows one slice of the patient CT images with the bladder, prostate and rectum segmented. (b) shows the reconstructed organ surfaces.

5.3.2 Cancer Grading/Staging Classification based on Prostate Elasticity Parameters

I further study the feasibility of using recovered elasticity parameters as a cancer prognostic indicator using our classifier based on relative tissue elasticity values and ages. Two classification methods, ordinal logistic regression and multinomial logistic regression, were tested in our study. I test each method with two sets of features. The first set of features contains only the relative tissue elasticity values $\hat{\mu}$. The resultant feature vector is one dimension. The second set of features contains both the relative tissue elasticity values and the age. The feature vector for this set of features is two dimensional. Our cancer staging has $C = 3$ classes, T1, T2 and T3. And the cancer grading has $G = 5$ classes, from 6 to 10. In our patient dataset, each patient has at least 2 sets of CT images. The elasticity parameter reconstruction algorithm needs 2 sets of CT images as input. I fix one set of CT images as the initial (reference) image and use the other M number of images \mathcal{T} , where $|\mathcal{T}| = M$ as the target (deformed) images. By registering the initial image to the target images, I obtain one elasticity parameter $\hat{\mu}_i, i = 1 \dots M$ for each image in \mathcal{T} . I perform both per-patient and per-image cross validation.

Per-Image Cross Validation: I treat all the target images ($N = 84$) of all the patients as data points of equal importance. The elasticity feature for each target image is the recovered elasticity parameter $\hat{\mu}$. In this experiment, I train our classifier using the elasticity feature of the 83 images then cross validate with the one left out. Then, I add the patient's age as another feature to the classifier and perform the validation. The results for cancer staging (T-Stage) classification are shown in

Fig. 5.2(a) and that for cancer grading (Gleason score) classification are shown in Fig. 5.2(b). The error metric is measured as the absolute difference between the classified cancer T-Stage and the actual cancer T-Stage. Zero error-distance means our classifier accurately classifies the cancer T-Stage.

The multinomial method outperforms the ordinal method for both cancer staging (T-Stage) and cancer aggression (Gleason score) classification. The main reason that I am observing this is due to the optimization weights or the unknown regression coefficients β (refer to supplementary document for the definition) dimension of the multinomial and ordinal logistic regression method. The dimension of the unknown regression coefficients of the multinomial logistic regression for cancer staging classification (with elasticity parameter and age as features) is 6 while that of ordinal logistic regression is 4. With the ‘age’ feature, I obtain up to 91% accuracy for predicting cancer T-Stage using multinomial logistic regression method and 89% using ordinal logistic regression method. For Gleason score classification I achieve up to 88% accuracy using multinomial logistic regression method and 81% using ordinal logistic regression method.

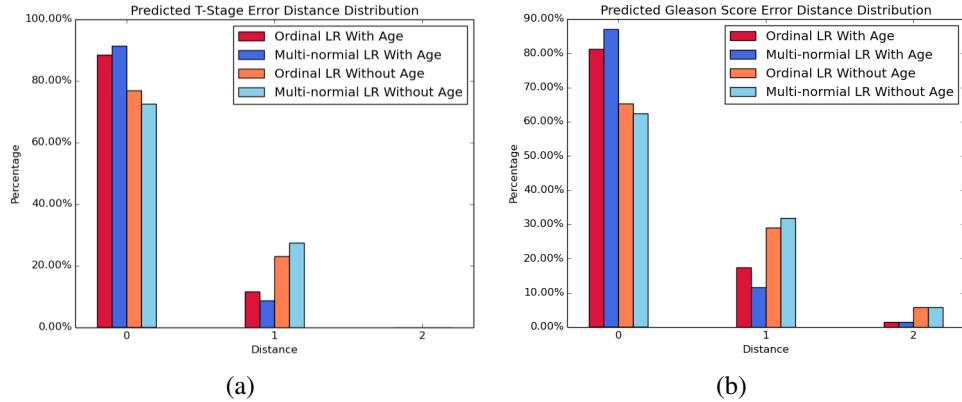


Figure 5.2: Error Distribution of Cancer Grading/Staging Classification for Per-Image Study. (a) shows error distribution of our cancer staging classification using the recovered prostate elasticity parameter and the patient’s age. For our patient dataset, the multinomial classifier (shown in royal blue and sky blue) outperforms the ordinal classifier (shown in crimson and coral). I achieve up to 91% accuracy using multinomial logistic regression and 89% using ordinal logistic regression for classifying cancer T-Stage based on recovered elasticity parameter and age. (b) shows the correlation between the recovered relative elasticity parameter and the Gleason score with/without the patient’s age. I achieve up to 88% accuracy using multinomial logistic regression and 81% using ordinal logistic regression for classifying Gleason score based on recovered elasticity parameter and age.

Per-Patient Cross Validation: For patients with more than 2 sets of images, I apply Gaussian sampling to $\hat{\mu}_i, i = 1 \dots M$ to compute the sampled elasticity parameter as the elasticity feature of the patient. I first train our classifier using the elasticity feature of the 28 patients then test the trained classifier on the remaining one patient not in the training set. I repeat this process for each of the 29 patients. Then I include the patient age as another feature in the classifier. The error distribution for cancer staging (T-Stage) classification results are shown in Fig. 5.3(a) and the error distribution of cancer grading (Gleason score) classification are shown in Fig. 5.3(b). I observe that the multinomial method in general outperforms the ordinal method. More interestingly, the age feature helps to increase the classification accuracy by 2% for staging classification and 7% for Gleason scoring classification. With the age feature, our multinomial classifier achieves up to 84% accuracy for classifying cancer T-Stage and up to 77% accuracy for classifying Gleason scores. And our ordinal classifier achieves up to 82% for cancer T-Stage classification and 70% for Gleason score classification. The drop in accuracy for per-patient experiments compared with per-image ones is primary due to the decrease in data samples.

Among the 16% failure cases for cancer staging classification, 15% of our multinomial classification results with age feature is only 1 stage away from the ground truth. And for the failure cases for scoring classification, only 10% of the classified Gleason scores is 1 away from the ground truth and 13% of them are 2 away from the ground truth.

5.4 Conclusion

In this chapter, I present an improved, non-invasive tissue elasticity parameter reconstruction framework using CT images. I further studied the correlation of the recovered *relative* elasticity parameters with prostate cancer T-Stage and Gleason score for multiclass classification of cancer T-stages and grades. The classification accuracy on our patient dataset using multinomial logistic regression method is up to 84% accurate for cancer T-stages and up to 77% accurate for Gleason

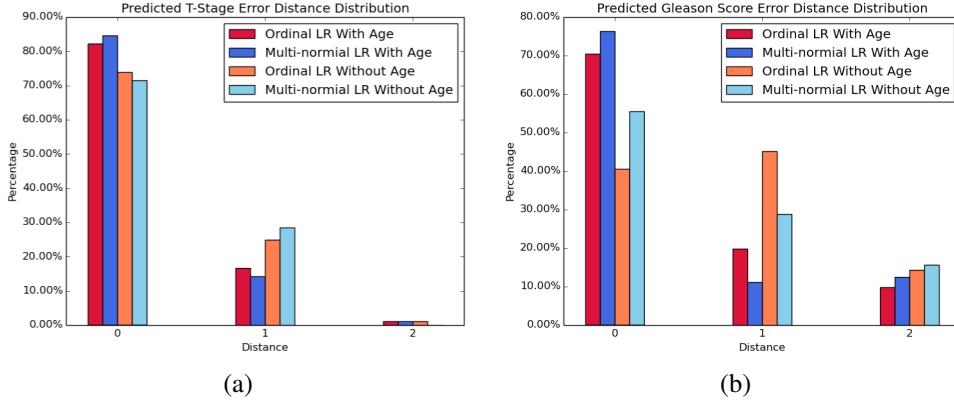


Figure 5.3: Error Distribution of Cancer Aggression/Staging Classification for Per-Patient Study. (a) shows the accuracy and error distribution of our recovered prostate elasticity parameter and cancer T-Stage. For our patient dataset, the multinomial classifier (shown in royal blue and sky blue) outperforms the ordinal classifier (shown in crimson and coral). I achieve up to 84% accuracy using multinomial logistic regression and 82% using ordinal logistic regression for classifying cancer T-Stage based on our recovered elasticity parameter and patient age information. (b) shows the correlation between the recovered relative elasticity parameter and the Gleason score. I achieve up to 77% accuracy using multinomial logistic regression and 70% using ordinal logistic regression for classifying Gleason score based on our recovered elasticity parameter and patient age information.

scores. This study further demonstrates the effectiveness of our algorithm for recovering (relative) tissue elasticity parameter *in-vivo* and its promising potential for correct classification in cancer screening and diagnosis.

Future Work: This study is performed on 113 sets of images from 29 prostate cancer patients all treated in the same hospital. More image data from more patients across multiple institutions can provide a much richer set of training data, thus further improving the classification results and testing/validating its classification power for cancer diagnosis. With more data, I could also apply our learned model for cancer stage/score prediction. And other features, such as the volume of the prostate can also be included in the larger study. Another possible direction is to perform the same study on normal subjects and increase the patient diversity from different locations. A large-scale study can enable more complete analysis and lead to more insights on the impact of variability due to demographics and hospital practice on the study results. Similar analysis and derivation could also be performed using other image modalities, such as MR and ultrasound, and shown to be applicable to other types of cancers.

CHAPTER 6: SINGLE-VIEW IMAGE-BASED GARMENT RECOVERY

6.1 Introduction

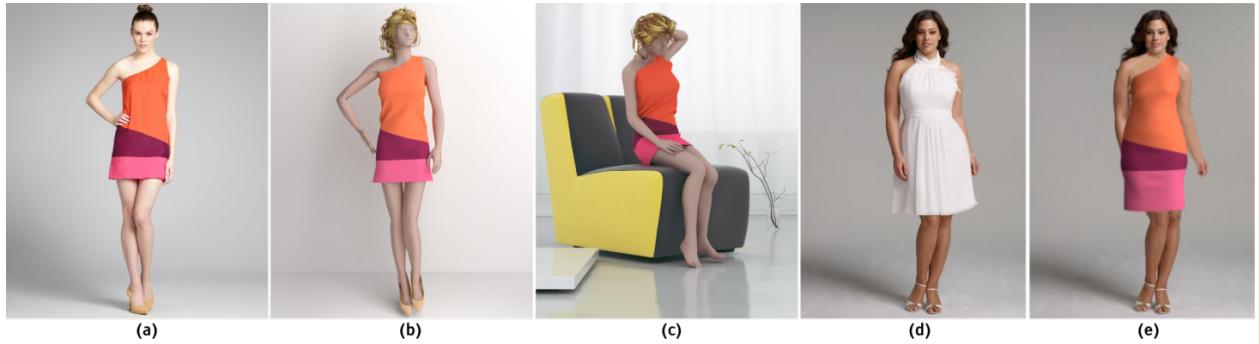


Figure 6.1: **Garment recovery and re-purposing results.** From left to right, I show an example of (a) the original image (*Saacclothes*, 2015) ©, (b) the recovered dress and body shape from a single-view image, and (c)-(e) the recovered garment on another body of different poses and shapes (*Hillsweddingdress*, 2015) ©.

In this work, I consider the problem of recovering garment models with both material and sizing parameters from a single-view image. Such a capability enables users to virtually try on garments given only a single photograph of themselves wearing clothing. Instead of representing the clothed human as a single mesh (Chen et al., 2013; Li et al., 2012), I define a separate mesh for a person’s clothing, allowing us to model the rich physical interactions between clothing and the human body. This approach also helps capture occluded wrinkles in clothing that are caused by various sources, including garment design that incorporates pleats, cloth material properties that influence the drape of the fabric, and the underlying human body pose and shape. Figure 6.1 illustrates some results generated by my system. In addition to virtual try-on applications, broader impacts in graphics include improving the accuracy of clothing models for animated characters, with the potential to further increase the visual realism of digital human models that already incorporate

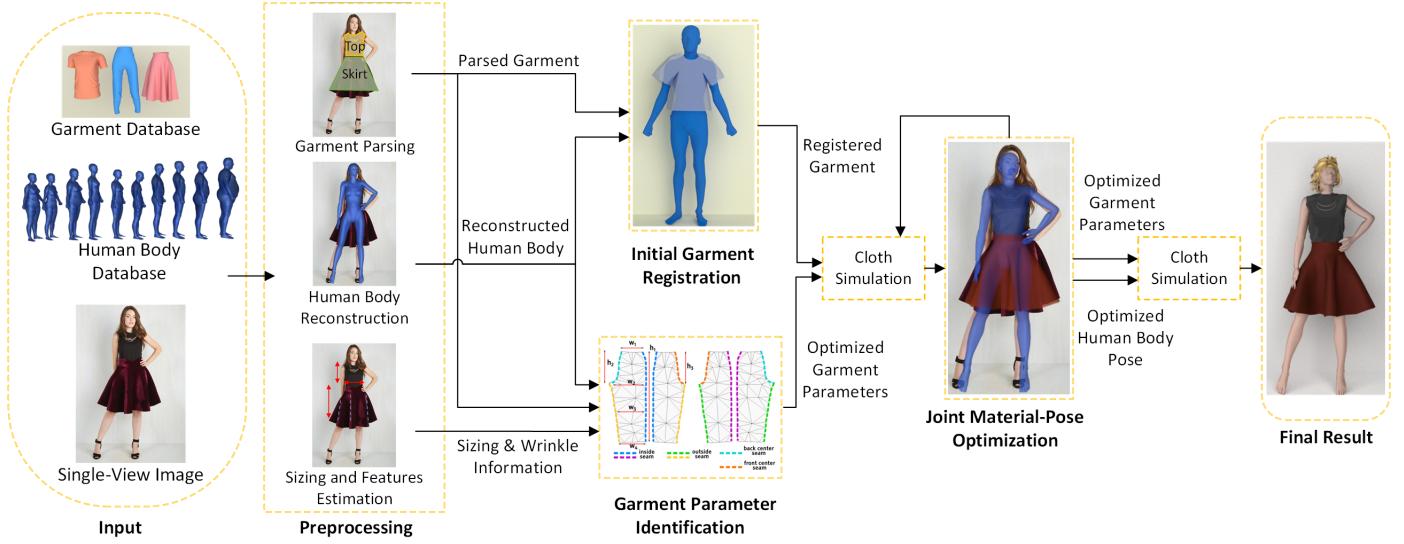


Figure 6.2: The flowchart of my algorithm. I take a single-view image (*ModCloth*, 2015) ©, a human-body dataset, and a garment-template database as input. I preprocess the input data by performing garment parsing, sizing and features estimation, and human-body reconstruction. Next, I recover an estimated garment described by the set of garment parameters, including fabric material, design pattern parameters, sizing and wrinkle density, as well as the registered garment dressed on the reconstructed body. Finally, I perform joint material-pose optimization and show the recovered results using cloth simulation on the virtual mannequin.

body-dependent priors for hair (Chai et al., 2012), face (Cao et al., 2013), skin (Nagano et al., 2015), and eyeballs (Bérard et al., 2014).

With limited input from a single-view image, I constrain the problem's solution space by exploiting three important priors. The first prior is a statistical human body distribution model constructed from a (naked) human body data set. This statistical model is used for extracting and matching the human body shape and pose in a given input image. The second prior is a collection of all sewing patterns of various common garment types, such as skirts, pants, shorts, t-shirts, tank tops, and dresses, from a database of all garment templates. Finally, the third prior is a set of all possible configurations and dynamical states of garments governed by their respective constitutive laws and simulated by a physically-based cloth model. Simulation helps provide additional 3D physical constraints lacking in a 2D image.

My method proceeds as follows. To construct an accurate body model, the user indicates 14 joint positions on the image and provides a rough sketch outlining the human body silhouette. (This

step can also be automated using image processing and body templates for standard unoccluded poses.) From this information, I use a statistical human model to automatically generate a human body mesh for the image. To estimate the clothing model, I first compute a semantic parse of the garments in the image to identify and localize depicted clothing items. This semantic segmentation is computed automatically using a data-driven method for clothing recognition (Yamaguchi et al., 2013). I then use the semantic parsing to extract garment sizing information, such as waist girth, skirt length and so on, which are then used to map the depicted garments onto the existing garment templates and adjust the sewing patterns based on the extracted parameters. I also analyze the segmented garments to identify the location and density of wrinkles and folds in the recovered garments, which are necessary for estimating material properties of the garments for virtual try-on.

Once I have obtained both the body and clothing models, I perform an image-guided parameter identification process, which optimizes the garment template parameters based on the reconstructed human body and image information. I fit my 3D garment template’s surface mesh onto the human body to obtain the initial 3D garment, then jointly optimize the material parameters, the body shape, and the pose to obtain the final result. The flow chart of the overall process is shown in Fig. 6.2. My main contributions include:

- An image-guided garment parameter selection method that makes the generation of virtual garments with diverse styles and sizes a simple and natural task (Section 6.5);
- A joint material-pose optimization framework that can reconstruct both body and cloth models with material properties from a single image (Section 6.6);
- Application to virtual try-on and character animation (Section 6.7).

6.2 Related Work

My work is built on previous efforts in cloth modeling, human pose/shape recovery, garment capture from single-view images, and semantic parsing.

Cloth Modeling: Cloth simulation is a traditional research problem in computer graphics. Early work on cloth simulation includes (Weil, 1986; Ng and Grimsdale, 1996; Baraff and Witkin, 1998; House and Breen, 2000). More recently, a number of methods were proposed to solve the complicated problems presented in cloth simulation, including collision detection (Govindaraju et al., 2007; Tang et al., 2009; Curtis et al., 2008), collision handling, friction handling (Bridson et al., 2002), strain limiting (Goldenthal et al., 2007; English and Bridson, 2008; Thomaszewski et al., 2009; Wang et al., 2010) and remeshing (Narain et al., 2012).

Realistic wrinkle simulation is an important problem in realistic cloth modeling. Volino and Magnenat-Thalmann (Volino and Magnenat-Thalmann, 1999) introduced a geometry-based wrinkle synthesis. Rohmer et al. (Rohmer et al., 2010) presented a method to augment a coarse cloth mesh with wrinkles. Physically based cloth wrinkle simulation depends on an accurate model of the underlying constitutive law; different bending and stretching energy models for wrinkle simulation have been proposed (Bridson et al., 2003). More recently, physically-based simulation has been proposed for direct 3D editing (?).

Garment modeling is built upon cloth simulation. It also needs to take into consideration the design and sewing pattern of the garment. Some methods start from the 2D design pattern (Protopsaltou et al., 2002; Decaudin et al., 2006; Berthouzoz et al., 2013) or 2D sketches (Turquin et al., 2007; Robson et al., 2011). Other methods explore garment resizing and transfer from 3D template garments (Wang et al., 2005; Meng et al., 2012; Sumner and Popović, 2004). In contrast, my work synthesizes different ideas and extends these methods to process 2D input image and fluidly transfer the results to the simulation of 3D garments. I can also edit the 2D sewing patterns with information extracted from a single-view image, which can be used to guide the generation of garments of various sizes and styles.

Human Pose and Shape Recovering: Human pose and shape recovery from a single-view image has been extensively studied in computer vision and computer graphics. Taylor (Taylor, 2000) presented an articulated-body skeleton recovery algorithm from a single-view image with limited user input. Agarwal et al. (Agarwal and Triggs, 2006) proposed a learning-based method to recover

human body poses from monocular images. Ye et al. (Ye et al., 2014) applied a template-based method for real-time single RGBD image human pose and shape estimation. I refer readers to this survey on human motion recovery and analysis (Moeslund et al., 2006).

Human pose and shape recovery in computer graphics focus primarily on reconstructing muscle accurately and on watertight 3D human body meshes. A realistic 3D human body mesh is the basis for character animation. A human body mesh is required for the recovery of clothing with rich details. For human body mesh generation, I follow the previous data-driven methods, most of which are PCA based. These techniques use a set of bases to generate a variety of human bodies of different poses and shapes. Seo and Thalmann (Seo and Magnenat-Thalmann, 2003) presented a method to construct human body meshes of different shapes. Following this work, Anguelov et al. (Anguelov et al., 2005) introduced the SCAPE model, which can produce human body meshes of different poses and shapes. Using the SCAPE model, Balan et al. (Balan et al., 2007) presented a method to recover detailed human shape and pose from images. Hasler et al. (Hasler et al., 2009) encode both human body shapes and poses using PCA and semantic parameters. Building upon these previous models, Zhou et al. (Zhou et al., 2010) proposed a method to recover the human body pose and shape from a single-view image.

Clothing Capturing: In the last decade, many methods have been proposed for capturing clothing from images or videos. Methods can be divided into two categories: marker-based and markerless. Most marker-based clothing capture methods require the markers to have been pre-printed on the surface of the cloth. Different kinds of markers have been used for capturing (Scholz and Magnor, 2006; Hasler et al., 2006; Tanie et al., 2005; Scholz et al., 2005; White et al., 2007). Markerless methods, which do not require pre-printed clothing markers, can be characterized into several categories of methods: single-view (Zhou et al., 2013; Jeong et al., 2015), depth camera based (Chen et al., 2015); and multi-view methods (Popa et al., 2009). Rest configuration recovery proposed by Casati et. al. (Casati et al., 2016) is used for cloth design. These methods have some limitations, however, including inability to capture fine garment details and material properties, the loss of the original garment design, and complexity of the capturing process. In contrast, my

method can retrieve the 2D design pattern with the individual measurements obtained directly from a single image. Using a joint human pose and clothing optimization method, my algorithm recovers realistic garment models with details (e.g. wrinkles and folds) and material properties.

Semantic Parsing: Semantic parsing is a well-studied problem in computer vision, where the goal is to assign a semantic label to every pixel in an image. Most prior work has focused on parsing general scene images (Long et al., 2015a; Farabet et al., 2013; Pinheiro and Collobert, 2014). I work on the somewhat more constrained problem of parsing clothing in an image. To obtain a semantic parse of the clothing depicted in an image, I make use of the data-driven approach by Yamaguchi et al. (Yamaguchi et al., 2013). This method automatically estimates the human body pose from a 2D image, extracts a visual representation of the clothing the person is wearing, and then visually matches the outfit to a large database of clothing items to compute a clothing parse of the query image.

6.3 Method

In this section, I give the formal definition of the problem. The input to my system is an RGB image Ω . I assume the image is comprised of three parts: the background region Ω_b , the foreground naked human body parts Ω_h and the foreground garment Ω_g , where $\Omega = \Omega_b \cup \Omega_h \cup \Omega_g$. In addition, I assume that both the human body and the garment are in a statically stable physical state. Although

this assumption precludes images capturing a fast moving human, it provides a crucial assumption for my joint optimization algorithm.

Problem: Given Ω_g, Ω_h , how to recover

- the garment described by a set of parameters $\langle \mathcal{C}, \mathcal{G}, \mathbf{U}, \mathbf{V} \rangle$,
- along with a set of parameters $\langle \boldsymbol{\theta}, \mathbf{z} \rangle$ that encode human body pose and shape obtained from the image.

Garment: For the clothing parameters, \mathcal{C} is the set of material parameters including stretching stiffness and bending stiffness coefficients; \mathbf{U} is the 2D triangle mesh representing the garment's pattern pieces; and \mathbf{V} is the 3D triangle mesh representation of the garment. For each triangle of the 3D garment mesh \mathbf{V} , there is a corresponding one in the 2D space \mathbf{U} . For each mesh vertex $\mathbf{x} \in \mathbf{V}$, such as those lying on a stitching seam in the garment, there might be multiple corresponding 2D vertices $\mathbf{u} \in \mathbf{U}$. The parameter \mathcal{G} is the set of parameters that defines the dimensions of the 2D pattern pieces. I adopt the garment sizing parameters based on classic sewing patterns (Barnfield, 2012) shown in Fig. 6.3(a), 6.3(c) and 6.3(e), with the corresponding parameters defined in Fig. 6.3(b), 6.3(d), and 6.3(f), respectively. For example, I define the parameter $\mathcal{G}_{\text{pants}} = \langle w_1, w_2, w_3, w_4, h_1, h_2, h_3 \rangle$ for pants, where the first four parameters define the waist, bottom, knee and ankle girth and the last three parameters indicate the total length, back upper, and front upper length. For each basic garment category, I can manually define this set of parameters \mathcal{G} . By manipulating the values of the parameters \mathcal{G} , garments of different styles and sizes can be modeled: capri pants vs. full-length pants, or tight-fitting vs. loose and flowy silhouettes.

Fabric Materials: I adopt the material model developed by Wang et al. (Wang et al., 2011a). The material parameters \mathcal{C} consist of 18 bending and 24 stretching parameters.

Human Body: For the human body parameters, $\boldsymbol{\theta}$ is the set of joint angles that together parameterize the body pose, and \mathbf{z} is the set of semantic parameters that describe the body shape. I follow the PCA encoding of the human body shape presented in (Hasler et al., 2009). The semantic parameters

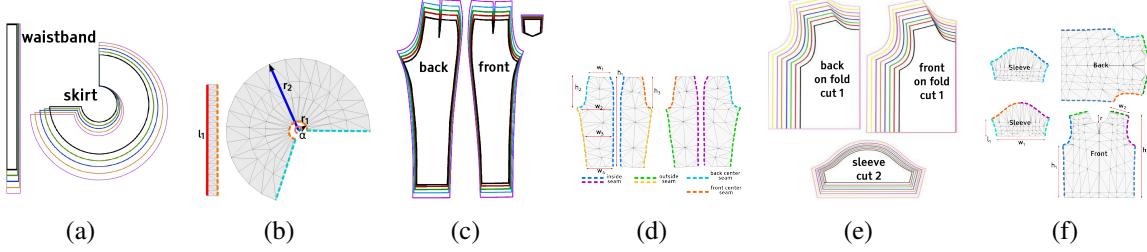


Figure 6.3: **Template sewing pattern and parameter space of a skirt, pants, and t-shirt.** (a) The classic circle sewing pattern of a skirt. (b) My parametric skirt template showing dashed lines for seams and the four parameters $\langle l_1, r_1, r_2, \alpha \rangle$, in which parameter l_1 is related to the waist girth and parameter r_2 is related to the length of the skirt. (c) The classic pants sewing pattern. (d) My parametric pants template with seven parameters $\langle w_1, w_2, w_3, w_4, h_1, h_2, h_3 \rangle$. (e) The classic t-shirt sewing pattern. (f) My parametric t-shirt template with six parameters $\langle r, w_1, w_2, h_1, h_2, l_1 \rangle$.

include gender, height, weight, muscle percentage, breast girth, waist girth, hip girth, thigh girth, calf girth, shoulder height, and leg length.

6.4 Data Preparation

This section describes the data preprocessing step. I begin with the data representations for the garment and the human body, followed by a brief description of each preprocessing module.

6.4.1 Data Representations

The garment template database can be represented as a set $\mathcal{D}_c = \{\langle \mathcal{C}_i, \mathcal{G}_i, \mathbf{U}_i, \mathbf{V}_i, \mathbf{B}_{\text{garment},i} \rangle | i \in 1, \dots, N\}$, where N is the number of garment templates. Each garment template consists of a 2D triangle mesh \mathbf{U} representing the sewing pattern, a 3D mesh \mathbf{V} , a set of dimension parameters \mathcal{G} for each pattern piece, the skeleton $\mathbf{B}_{\text{garment}}$, and a set of material parameters \mathcal{C} .

The human body template database $\mathcal{D}_h = \{\langle \boldsymbol{\theta}_j, \mathbf{z}_j, \mathbf{B}_{\text{body},j} \rangle | j \in 1, \dots, M\}$ consists of M naked human body meshes with point to point correspondence. The garment skeleton $\mathbf{B}_{\text{garment}}$ and the human body skeleton \mathbf{B}_{body} share the same structure, i.e. they have the same number of the joints and bones. Furthermore, the two skeletons are of same scale. I use several human body datasets, including the SCAPE dataset (Anguelov et al., 2005), the SPRING dataset (Yang et al., 2014), the TOSCA dataset (Bronstein et al., 2008; Young et al., 2007; Bronstein et al., 2006), and

the dataset from (Hasler et al., 2009). My garment template is defined in the same metric system as the human template to scale the garments during the initial registration. Each garment template and human template is rigged on a common skeleton with the same set of joints.

Parameterized Garment Model: Given the garment template database \mathcal{D}_c , each vertex \mathbf{u} of the 2D garment pattern mesh is computed as

$$\mathbf{u}(\mathcal{G}) = \sum_{i=0}^M \nu_i(\mathcal{G}) \mathbf{u}^0, \quad (6.1)$$

with M is the number of the neighboring vertices that have influence on vertex \mathbf{u} , ν_i is the weight associated with the vertex \mathbf{u} and \mathbf{u}^0 is the vertex position of the 2D garment pattern template.

Parameterized Human Model: Given the body database \mathcal{D}_h , I extract a statistical shape model for human bodies. Under this model, each world space vertex position \mathbf{p} on the human body is parameterized as

$$\mathbf{p}(\boldsymbol{\theta}, \mathbf{z}) = \sum_i^{|\mathbf{B}_{\text{body}}|} \omega_i \mathcal{T}_{\mathcal{B}_i}(\boldsymbol{\theta}) (\mathbf{p}^0 + \mathbf{Z}_i \mathbf{z}), \quad (6.2)$$

which is a composition of a linear blend skinning model (Kavan et al., 2010) and an active shape model (Zhao et al., 2003). Here ω_i and \mathcal{B}_i are the set of weights and bones associated with the vertex \mathbf{p} . $\mathcal{T}_{\mathcal{B}_i}$ is the transformation matrix of bone \mathcal{B}_i . \mathbf{p}^0 and \mathbf{Z}_i are the mean shape and active shape basis at the rest pose, respectively. The basis \mathbf{Z}_i is calculated by running PCA (Hasler et al., 2009) on \mathcal{D}_h .

6.4.2 Preprocessing

My preprocessing step consists of: a) human body reconstruction to recover the human body shape and pose from the input image, b) garment parsing to estimate the locations and types of garments depicted in the image, and c) parameter estimation to compute the sizing and fine features of the parsed garments.

Human Body Reconstruction: My human body recovery relies on limited user input. The user helps us identify the 14 human body joints and the human body silhouette. With the identified

joints, a human body skeleton is recovered using the method presented in (Taylor, 2000): the semantic parameters \mathbf{z} are optimized to match the silhouette. In this step, I ignore the camera scaling factor.

Garment Parsing: I provide two options for garment parsing. The first uses the automatic computer vision technique presented in (Yamaguchi et al., 2013). This approach combines global pretrained parse models with local models learned from nearest neighbors and transferred parse masks to estimate the types of garments and their locations on the person. The second option requires assistance from the user. Given the image Ω , I extract the clothing regions $\Omega_{b,h,g}$ by performing a two-stage image segmentation guided by user sketch. In the first stage, a coarse region boundary is extracted using a graph cut algorithm (Li et al., 2004). Then, the region is refined via re-clustering (Levin et al., 2008).

Image Information Extraction: Given the segmentation of the garment Ω_g , the next step is to convert it to pixel-level garment silhouette \mathcal{S} and compute the regional average wrinkle density \mathcal{P} . I define the region based on the contact between garment and human body joints. If there are no contact, the wrinkle density is that of the entire garment. I extract the average regional wrinkle density \mathcal{P} from the garment images using an improved implementation of (Popa et al., 2009). I first detect edges using Holistically-Nested edge detection (Xie and Tu, 2015) and then smooth the edges by fitting them to low-curvature quadric curves. I reconnect broken edges by merging those with nearby endpoints and similar orientations. Finally, I form 2D folds by matching parallel edges. Edges not part of a pair are unlikely to contribute to a fold and are discarded. The average number of wrinkles per region is the average wrinkle density \mathcal{P} .

6.4.3 Initial Garment Registration

My initial clothing registration step aims to dress my template garment onto a human body mesh of any pose or shape. I optimize the vertex positions of the 3D mesh, \mathbf{x} , of the template clothing based on the human body mesh parameters $\langle \theta, \mathbf{z} \rangle$. In this step, I ignore the fit of the clothing on the human body (this step is intended to fix the 2D triangle mesh \mathbf{U}). I follow the

method proposed in (Brouet et al., 2012) for registering a template garment to a human body mesh with a different shape. However, their method is unable to fit the clothing to human meshes with varying poses; I extend their approach by adding two additional steps.

The first step requires the alignment of the joints \mathcal{Q}_c of the template garment skeleton with the joints \mathcal{Q}_h of the human body mesh skeleton, as shown in Fig. 6.4. Each joint $\mathbf{q} \in \mathcal{Q}_c$ of the garment has one corresponding joint $\mathbf{t} \in \mathcal{Q}_h$ of the human body mesh. I denote the number of joints of the garment as K_c . This step is done by applying a rigid body transformation matrix \mathbf{T} on the joint of the garment, where \mathbf{T} minimizes the objective function

$$\sum_{i=0, \mathbf{q}_i \in \mathcal{Q}_c, \mathbf{t}_i \in \mathcal{Q}_h}^{K_c} \|\mathbf{T}\mathbf{q}_i - \mathbf{t}_i\|^2 \quad (6.3)$$

Next, I need to fit this transformed 3D template garment mesh onto the human body mesh with pose described by parameter $\boldsymbol{\theta}$, the vector of the angles of the joints. My template garment is then deformed according to $\boldsymbol{\theta}$. I denote the vector $\boldsymbol{\beta}$ as the joint angles of the template garment mesh. I set the value of the vector β_i to the value of the corresponding joint angle θ_i of the human body mesh. Then I compute the 3D garment template mesh such that it matches the pose of the underlying human body mesh according to this set of joint angles $\boldsymbol{\beta}$ by,

$$\mathbf{x}_i(\boldsymbol{\beta}) = \sum_j v_{ij} \mathcal{T}_{\mathcal{B}_j}(\boldsymbol{\beta}) \mathbf{x}^0, \quad (6.4)$$

where v_{ij} is the weight of bone \mathcal{B}_j on vertex \mathbf{x}_i and $\mathcal{T}_{\mathcal{B}_j}$ is the transformation matrix of the bone \mathcal{B}_j . An example result is shown in Fig. 6.4(c).

The final step is to remove collisions between the garment surface mesh and the human body mesh. I introduce two constraints: rigidity and non-interception. The deformation of the clothing should be as-rigid-as-possible (Igarashi et al., 2005). After this step, I have an initial registered garment with a 3D mesh $\hat{\mathbf{V}}(\mathbf{T}, \boldsymbol{\theta})$ that matches the underlying human pose and is free of interpenetrations with the human body. I show my initial garment registration results in Fig. 6.5.

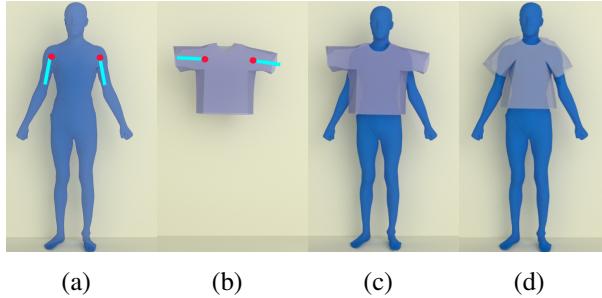


Figure 6.4: Initial garment registration process. (a) The human body mesh with the skeleton joints shown as the red sphere and the skeleton of the arm shown as the cyan cylinder. (b) The initial t-shirt with the skeleton joint shown as the red sphere and the skeleton of the sleeve part of it shown as the cyan cylinder. (c) The t-shirt and the human body mesh are aligned by matching the joints. (d) The result after aligning the skeleton and removing the interpenetrations.



Figure 6.5: Initial garment registration results. I fit garments to human bodies with different body shapes and poses.

6.5 Image-Guided Parameter Identification

In this section, I explain the step-by-step process of extracting garment material and sizing parameters $\langle \mathcal{C}, \mathcal{G} \rangle$ from an image.

6.5.1 Overview

Starting from my 2D triangle mesh \mathbf{U} of the pattern pieces, I select garment parameters based on the sizing and detailed information $\langle \mathcal{S}, \mathcal{P} \rangle$ estimated from the source image. In this step, I adjust the garment material and sizing parameters $\langle \mathcal{C}, \mathcal{G} \rangle$ but fix the 3D mesh $\hat{\mathbf{V}}(\mathbf{T}, \boldsymbol{\theta})$ (computed from Sec. 6.4.3) to obtain the garment that best matches the one shown in the image. I need two specific pieces of information from the image: the pixel-level garment silhouette \mathcal{S} and the regional average wrinkle density \mathcal{P} of the clothing. For example, for a skirt, I need to estimate the waist girth and the length of the skirt from the image. Using these two pieces of information, I initialize the garment sizing parameters \mathcal{G} . Based on the wrinkle information computed from the image, I then

optimize both the fabric material parameters \mathcal{C} and the sizing parameters of the garment pattern \mathcal{G} , to be described in the following two subsections.

6.5.2 Garment Types, Patterns, and Parameters

For basic garment types, such as skirts, pants, t-shirts, and tank tops, I use one template pattern for each. I modify the classic sewing pattern according to the parameters \mathcal{G} . By adjusting the garment parameters \mathcal{G} and fabric material parameters \mathcal{C} , I recover basic garments of different styles and sizes. The classic circle skirt sewing pattern is shown in Fig. 6.3(a). My parametric space, which is morphed from this circle sewing pattern, is shown in Fig. 6.3(b). For the skirt pattern, there are four parameters to optimize: $\mathcal{G}_{\text{skirt}} = \langle l_1, r_1, r_2, \alpha \rangle$. The ratio between the parameter l_1 and r_2 is constrained by the waist girth and skirt length information extracted from the image. The other two parameters, r_1 and α , are constrained by the wrinkle density. With different garment parameters, skirts can vary from long to short, stiff to soft, and can incorporate more or fewer pleats, enabling us to model a wide variety of skirts from a single design template.

Similarly for pants, the classic sewing pattern and my template pattern pieces are shown in Fig. 6.3(c) and Fig. 6.3(d). There are seven parameters for the dimensions of the pants template: $\mathcal{G}_{\text{pants}} = \langle w_1, w_2, w_3, w_4, h_1, h_2, h_3 \rangle$ with the first four parameters describing the waist, bottom, knee, and ankle girth, and the last three parameters representing the total, back-upper and front-upper lengths. The t-shirt sewing pattern is shown in Fig. 6.3(e), and my parametric t-shirt pattern is shown in Fig. 6.3(f) with the garment parameters $\mathcal{G}_{\text{tshirt}} = \langle r, w_1, w_2, h_1, h_2, l_1 \rangle$. Among the parameters $\mathcal{G}_{\text{tshirt}}$, parameter r describes the neckline radius, w_1 describes the sleeve width, w_2 describes the shoulder width, h_1 describes the bottom length, h_2 describes the total length, and l_1 describes the length of the sleeve.

Different sewing patterns result in very different garments. Traditional sewing techniques form skirt wrinkles by cutting the circular portion of the pattern. To simulate this process but make it generally applicable, I modify the parameter \mathcal{G} , which achieves the same effect. In addition to the differences created by sewing patterns, professional garment designers also take advantage of cloth

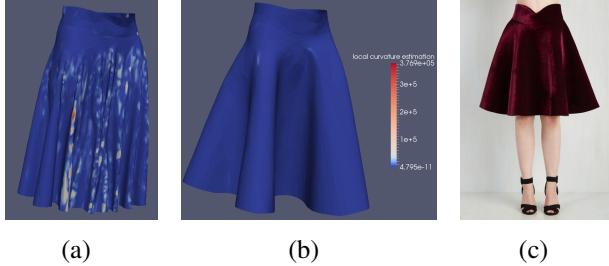


Figure 6.6: Material parameter identification results. (a) The local curvature estimation before optimizing the bending stiffness coefficients (using the cool-to-hot color map). (b) The local curvature estimation after the optimization. (c) The original garment image (*ModCloth, 2015*) ©.

material properties to produce different styles of clothing. I tune the bending stiffness coefficients and stretching stiffness coefficients in \mathcal{C} to simulate this parameter selection process.

6.5.3 From Wrinkle Density to Material Property and Sizing Parameters

One of my key contributions in this work is the identification of fabric materials based on wrinkles and folds, because different fabric stiffness produce varying wrinkle/folding patterns. I characterize the wrinkles and folds using their local curvatures. The first step is to map the regional wrinkle density \mathcal{P} (computed in Sec. 6.4.2) to the average regional curvature \mathcal{K} .

I recover the garment material parameter \mathcal{C}' by minimizing the regional average curvature differences between my recovered garment $\mathcal{K}(\mathcal{C}, \mathcal{G})$ and the reference garment $\mathcal{K}(\mathcal{P})_{\text{target}}$

$$\mathcal{C}' = \underset{\mathcal{C}}{\operatorname{argmin}} \|\mathcal{K}(\mathcal{C}, \mathcal{G}) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2. \quad (6.5)$$

The reference garment regional average curvature $\mathcal{K}(\mathcal{P})_{\text{target}}$ is computed by linear interpolation. I first approximate the regional average curvature threshold for the sharp wrinkles and smooth folds. The regional average curvature threshold for one sharp wrinkle is up to 10^5 m^{-1} and that for smooth folds is close to 10^{-4} m^{-1} . Sharp wrinkles or large folds are determined by the density of the extracted 2D wrinkles. Both sharp wrinkles and large folds are detected by edge detection. Sharp wrinkles are identified by the center of the wrinkles while the large folds are identified by the edges where the curvature are the largest. The density of the extracted 2D wrinkles ranges from 1

m^{-2} to 50 m^{-2} based on my observation. The interpolation process (with the linear interpolation function \mathcal{I}) is

$$\mathcal{K}(\mathcal{P})_{\text{target}} = \mathcal{I}(\mathcal{P}), \quad (6.6)$$

with the linear interpolation function $\mathcal{I}(1) = 10^{-4}$ and $\mathcal{I}(50) = 10^5$. Local curvature estimation of κ at each vertex is computed based on the bending between the two neighboring triangles sharing the same edge. For each vertex \mathbf{x} of the two triangles that share an edge \mathbf{e} , the local curvature κ is computed following the approach from Wang et al. (Wang et al., 2011a) and Bridson et al. (Bridson et al., 2003)

$$\kappa = \|\sin(\alpha/2)(h_1 + h_2)^{-1}|\mathbf{e}|\mathbf{x}\|, \quad (6.7)$$

where h_1 and h_2 are the heights of the two triangles that share the edge \mathbf{e} and α is the supplementary angle to the dihedral angle between the two triangles. The corresponding bending force \mathbf{f}_{bend} for each vertex \mathbf{x} is computed as

$$\mathbf{f}_{\text{bend}} = k \sin(\alpha/2)(h_1 + h_2)^{-1}|\mathbf{e}|\mathbf{x}, \quad (6.8)$$

where k is the bending stiffness coefficient.

Stretching also affects the formation of wrinkles. Each triangle $\langle \mathbf{u}_0, \mathbf{u}_1, \mathbf{u}_2 \rangle$ in the 2D template mesh is represented as $D_m = \begin{bmatrix} \mathbf{u}_1 - \mathbf{u}_0 \\ \mathbf{u}_2 - \mathbf{u}_0 \end{bmatrix}$, and each triangle in the 3D garment mesh $\langle \mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2 \rangle$ is represented as $d_m = \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}_0 \\ \mathbf{x}_2 - \mathbf{x}_0 \end{bmatrix}$. The stretching forces $\mathbf{f}_{\text{stretch}}$ are computed by differentiating the stretching energy Ψ , which depends on the stretching stiffness parameter w , the deformation gradient $\mathbf{F} = d_m D_m^{-1}$, and the Green strain tensor $\mathbf{G} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I})$ against the vertex 3D position \mathbf{x}

$$\mathbf{f}_{\text{stretch}} = \frac{\partial \Psi(w, \mathbf{F})}{\partial \mathbf{x}}. \quad (6.9)$$

The sizing and style of the garment described by the parameter \mathcal{G} obtained from the parsed garment are matched by minimizing the silhouette which is a 2D polygon differences between my recovered garment $\mathcal{S}(\mathcal{C}, \mathcal{G})$ and the reference garment silhouette $\mathcal{S}_{\text{target}}$

$$\mathcal{G}' = \underset{\mathcal{G}}{\operatorname{argmin}} \|\mathcal{S}(\mathcal{C}, \mathcal{G}) - \mathcal{S}_{\text{target}}\|^2. \quad (6.10)$$

The distance between two polygons is computed by summing up the distances between each point in polygon $\mathcal{S}(\mathcal{C}, \mathcal{G})$ to the other polygon $\mathcal{S}_{\text{target}}$. To compute the 2D silhouette $\mathcal{S}(\mathcal{C}, \mathcal{G})$, I first project the simulated garment 3D mesh $\mathcal{V}(\mathcal{C}, \mathcal{G}, \hat{\mathbf{V}}(\mathbf{T}, \boldsymbol{\theta}))$ onto the the 2D image with the orthogonal projection matrix \mathbf{H} , then compute the 2D polygon enclosing the projected points. The process is expressed as

$$\mathcal{S}(\mathcal{C}, \mathcal{G}) = f(\mathbf{H}\mathcal{V}(\mathcal{C}, \mathcal{G}, \hat{\mathbf{V}}(\mathbf{T}, \boldsymbol{\theta}))), \quad (6.11)$$

with f as the method that convert the projected points to a 2D polygon. I ignore the camera scaling factor in this step since the input is a single-view image. It is natural to scale the recovered clothing and human body shape as a postprocessing step.

Combining these two objectives, the objective (energy) function is expressed as

$$\begin{aligned} \mathbf{E} = & \|\mathcal{K}(\mathcal{C}, \mathcal{G}) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2 + \\ & \|\mathcal{S}(\mathcal{C}, \mathcal{G}) - \mathcal{S}_{\text{target}}\|^2. \end{aligned} \quad (6.12)$$

6.5.4 Optimization-based Parameter Estimation

The optimization is an iterative process (given in Algorithm 3), alternating between updates for the garment sizing and material parameters, \mathcal{G} and \mathcal{C} . I found that the value of the objective function is more sensitive to the cloth material properties \mathcal{C} than to the garment parameter \mathcal{G} , so I maximize the iterations when optimizing for \mathcal{C} , fixing \mathcal{G} . The optimization of parameter \mathcal{C} is coupled with the cloth dynamics. The underlying cloth simulator is based on the method proposed in (Narain et al., 2012). I drape the initial fitted garment onto the human body mesh. The garment is in the

dynamic state and subject to gravity. I couple my parameter estimation with this physically based simulation process. Before the simulation, I change the cloth material parameters \mathcal{C} so that when in static state the average of the local curvature κ matches the targeting threshold $\mathcal{K}_{\text{target}}$. That is to say, our optimizer minimizes $\|\mathcal{K} - \mathcal{K}_{\text{target}}\|^2$ by changing the bending stiffness parameters k and stretching stiffness parameters w .

I apply the Particle Swarm Optimization (PSO) (Kennedy, 2010) method for my material parameter optimization. The PSO method is known to be able to recover from local minima, making it the ideal method for some of the non-convex optimization problems. When the clothing reaches a static state, the optimizer switches to optimizing parameter \mathcal{G} . The optimizer for the parameter \mathcal{G} is not coupled with the garment simulation. The objective function is evaluated when the clothing reaches the static state. Similarly I adopt the PSO method for the parameter \mathcal{G} optimization. I use 40 particles for the parameter estimation process. The alternating process usually converges after four steps. One example result of the garment parameter process is shown in Fig. 6.6.

I constrain the cloth material parameter space. I use the “Gray Interlock” presented in (Wang et al., 2011a), which is composed of 60% cotton and 40% polyester, as the “softest” material, meaning it bends the easiest. I multiply the bending parameters of this material by 10^2 to give the “stiffest” material based on my experiments. My solution space is constrained by these two materials, and I initialize my optimization with the “softest” material parameters.

6.6 Joint Material-Pose Optimization

6.6.1 Optimal Parameter Selection

The parameter identification step provides us with the initial recovered garment described by the set of material and sizing parameters $\langle \mathcal{C}', \mathcal{G}' \rangle$. Many realistic garment wrinkles and folds, however, are formed due to the underlying pose of the human body, especially wrinkles that are located around human joints. Therefore, in this step, I further refine my results by optimizing both the pose parameters of the human body θ and the material properties of the cloth \mathcal{C}' . The

Algorithm 3 Garment Parameter Identification

```

1: procedure SIZINGPARAMIDENTIFICATION( $\mathcal{G}, \mathcal{S}_{\text{target}}$ )
2:   Compute silhouette  $\mathcal{S}(\mathcal{C}, \mathcal{G})$  using Eqn. 6.11
3:   Minimize  $\|\mathcal{S}(\mathcal{C}, \mathcal{G}) - \mathcal{S}_{\text{target}}\|^2$  using PSO
4:    $\mathcal{G}' = \operatorname{argmin}_{\mathcal{G}} \|\mathcal{S}(\mathcal{C}, \mathcal{G}) - \mathcal{S}_{\text{target}}\|^2$ 
5:   Update the 2D mesh  $\mathbf{U}'$  using Eqn. 6.1
6:   return  $\mathcal{G}', \mathbf{U}'$ 

7: procedure MATERIALPARAMIDENTIFICATION( $\mathcal{C}, \mathcal{K}_{\text{target}}$ )
8:   Compute regional curvature  $\mathcal{K}(\mathcal{C}, \mathcal{G})$ 
9:   Minimize  $\|\mathcal{K}(\mathcal{C}, \mathcal{G}) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2$  using PSO Method
10:   $\mathcal{C}' = \operatorname{argmin}_{\mathcal{C}} \|\mathcal{K}(\mathcal{C}, \mathcal{G}) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2$ 
11:  return  $\mathcal{C}'$ 

12: procedure MAIN( $\mathcal{C}, \mathcal{G}, \mathbf{U}, \epsilon$ )
13:   while  $E > \epsilon$  do //  $E$  as defined in Eqn. 6.12
14:     MATERIALPARAMIDENTIFICATION( $\mathcal{C}$ )
15:     SIZINGPARAMIDENTIFICATION( $\mathcal{G}$ )
16:   return  $\mathcal{G}', \mathcal{C}', \mathbf{U}'$ 

```

optimization objective for this step is

$$E_{\text{joint}} = \|\mathcal{K}(\mathcal{C}', \boldsymbol{\theta}) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2. \quad (6.13)$$

The optimization process (shown in Algorithm 4) is similar to the garment parameter identification step, alternating between updating the pose parameter $\boldsymbol{\theta}$ and the material parameters \mathcal{C}' . I use Particle Swarm Optimization method (Kennedy, 2010). The objective function (Eqn. 6.13) is more sensitive to the pose parameter $\boldsymbol{\theta}$ than to the material parameters \mathcal{C}' . I constrain the optimization space of parameter $\boldsymbol{\theta}$ by confining the rotation axis to only the three primal axes. An example of my joint material-pose optimization method is shown in Fig. 6.7.



Figure 6.7: Joint Material-Pose Optimization results. (a) The pants recovered prior to the joint optimization. (b) The recovered pants after optimizing both the pose and the material properties. The wrinkles in the knee area better match with those in the original image. (c) The original pants image (*Anthropologie, 2015*) ©.

Algorithm 4 Joint Pose-Material Parameter Identification

```

1: procedure MAIN(  $\mathcal{C}'$ ,  $\mathcal{G}'$ ,  $\theta$ ,  $\epsilon_1$ ,  $\epsilon_2$ )
2:   while  $E_{\text{joint}} > \epsilon_1$  and  $E_{\text{joint}}^{n+1}/E_{\text{joint}}^n > \epsilon_2$  do           //  $E_{\text{joint}}$  as defined in Eqn. 6.13
3:     Fix  $\mathcal{C}'$ , Optimize for  $\theta$  using Particle Swarm Method

$$\theta' = \operatorname{argmin}_{\theta} \|\mathcal{K}(\tilde{\mathcal{C}}, \theta) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2.$$
 (6.14)
4:     Fix  $\theta$ , Optimize for  $\mathcal{C}'$  using Particle Swarm Method

$$\tilde{\mathcal{C}} = \operatorname{argmin}_{\mathcal{C}'} \|\mathcal{K}(\mathcal{C}', \theta) - \mathcal{K}(\mathcal{P})_{\text{target}}\|^2.$$
 (6.15)
5:   return  $\tilde{\mathcal{C}}, \theta'$ 

```

6.6.2 Application to Image-Based Virtual Try-On

This joint material-pose optimization method can be applied directly to image-based virtual try-on. I first recover the pose and shape of the human body $\langle \theta, z \rangle$ from the single-view image. Then I dress the recovered human body with the reconstructed garments $\langle \tilde{\mathcal{C}}, \tilde{\mathcal{G}}, \tilde{\mathbf{U}}, \hat{\mathbf{V}} \rangle$ from other images. I perform the initial garment registration step (Sec. 6.4.3) to fit the 3D surface mesh $\hat{\mathbf{V}}$ onto the recovered human body $\langle \theta, z \rangle$.

Existing state-of-the-art virtual try-on rooms require a depth camera for tracking, and overlay the human body with the fitting garment (Ye et al., 2014). My algorithm, on the other hand, is able to fit the human body from a single 2D image with an optimized virtual outfit recovered from other images. I provide the optimized design pattern together with a 3D view of the garment fitted to the human body.

The fitting step requires iterative optimization in both the garment parameters and the human-body poses. As in a real fitting process, I vary the sizing of the outfits for human bodies of different sizes and shapes. When editing in parameter space using the methods introduced in the previous section, I ensure that the recovered garment can fit on the human body while minimizing the distortion of the original design. For each basic garment, I use one template pattern and the

corresponding set of parameters. To preserve the garment design, I do not change the material properties of the fabric when virtually fitting the recovered garment to a new mannequin.

6.7 Results and Discussion

I have implemented my algorithm in C++ and demonstrated the effectiveness of my approach throughout this chapter. In this section, I show example results, performance, and comparisons to other garment recovery methods.

6.7.1 Garment Recovery Results

I show several examples of garment recovery from a single-view image. In Fig. 6.8 and Fig. ??, I show that my method can recover garments of different styles and materials. Fig. ?? demonstrates the effectiveness of my method for the recovery of partially occluded garments. It also shows that my recovered garment can be applied to human bodies in different poses.



Figure 6.8: **Skirt and pants recovery results.** I recover the partially occluded, folded skirts from the single-view images in the first, fourth (*ModCloth, 2015; AliExpress, 2015*) © and seventh (*Anthropologie, 2015*) © columns. The recovered human body meshes are shown in the second, fifth and eighth columns overlaid on the original images. The recovered skirts are shown in the third, sixth and ninth columns.

Image-Based Garment Virtual Try-On: I show examples of my image-based garment virtual try-on method (Sec. 6.6.2) in Fig. 6.1 and Fig. 6.10. I can effectively render new outfits onto people from only a single input image.

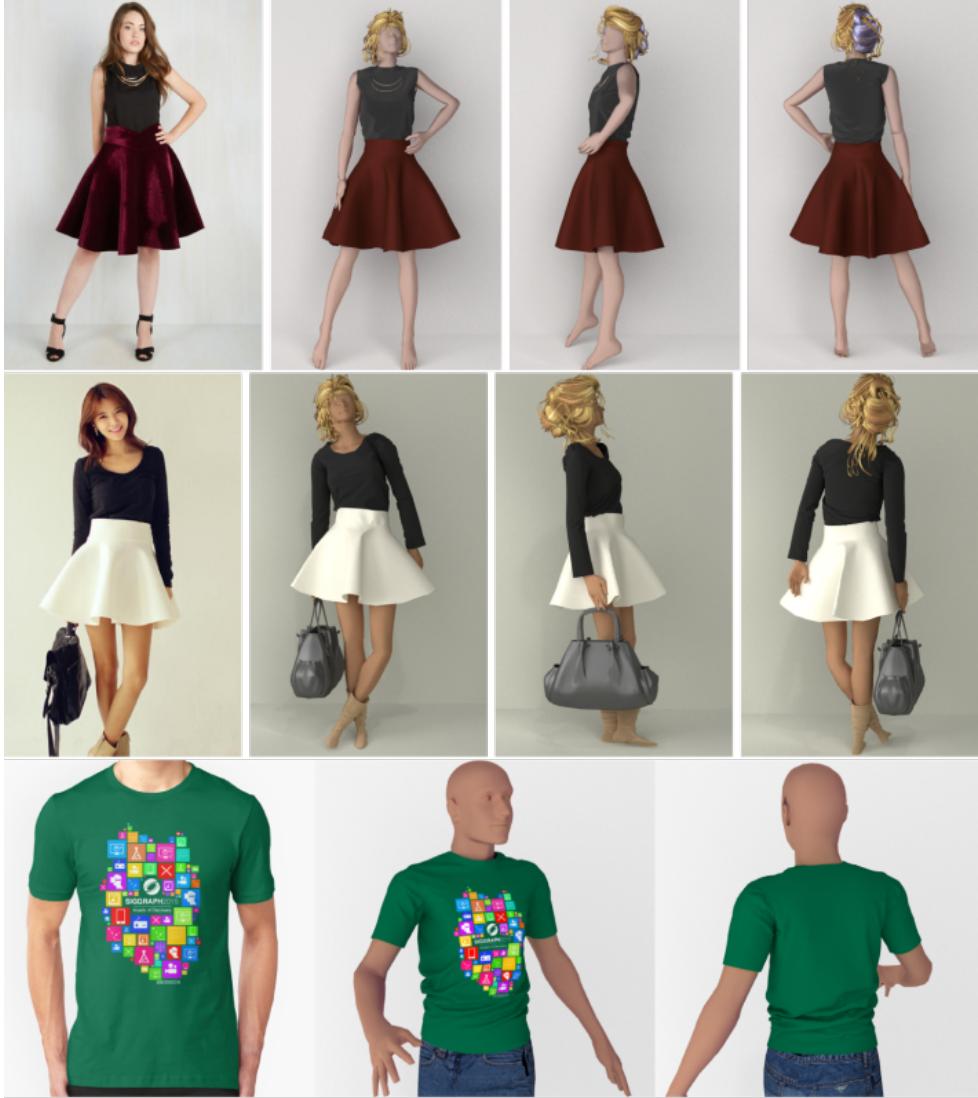


Figure 6.9: Garment recovery results. For the first two rows, the input image (leftmost) (*ModCloth, 2015; AliExpress, 2015; RedBubble, 2015*) © and recovered garment on the extracted human body. In the last row, the input image (leftmost) and the recovered garment on a twisted body.

Evaluation: I evaluate the accuracy of the recovered sizing parameters \mathcal{G} and regional average curvature \mathcal{K} using synthetic scenes. Each synthetic scene has two lighting conditions, mid-day, and sunset (shown in Fig. 6.11). I fix both the extrinsic and the intrinsic camera parameters for scene rendering, and the garments are in static equilibrium. Through these ten test cases, I can best validate the accuracy and reliability of my method against different body poses and lighting conditions on T-shirts and pants, as the sizing and material parameters are known exactly and do not require noisy measurements and/or data fitting to derive a set of estimated/measured parameters



Figure 6.10: Image-based garment transfer results. I dress the woman in (a) (*FashionableShoes, 2013; Boden, 2015*) © with the skirt I recovered from (b) (*AliExpress, 2015; ModCloth, 2015*) ©. (c) I simulate my recovered skirt with some wind motion to animate the retargeted skirt, as shown in (d). Another example of garment transfer is given in (e) - (h).

(which are not likely to be 100% accurate) to serve as the ground truth. The evaluation result, after eliminating the camera scaling factor, is shown in Table 6.1.

I found that the lighting conditions mainly affect the body silhouette approximation and the garment folding parsing, while the body skeleton approximation is affected by the pose. Overall, I achieve an accuracy of up to 90.2% for recovering the sizing parameters and 82% for recovering the material parameters for t-shirts and pants under different body poses and lighting conditions. The accuracy is computed as the average accuracy for each parameter from the ground truth.

I evaluate the accuracy of the recovered material properties by measuring the difference between the ground truth and that of the recovered garment for both the mean curvature and the material parameters, as the accuracy of mean-curvature recovery also correlates with the accuracy of the material-parameter estimation. As shown in Table 6.2, I am able to achieve an accuracy of up to 86.7% and 80.2%, respectively, for the recovery of mean curvatures and different material parameters for the skirt.

To test the sensitivity of my garment sizing and material recovery method with respect to the human body recovery, I further validate the accuracy of my recovered human-body skeleton joint

angles. I measure the accuracy of the human body pose by computing the difference between the recovered joint angles and the known ‘ground-truth’ joint angles, as shown in Table 6.3). I achieve 100% of accuracy when the joints are not occluded because some of those joints are in the initial states. For those joints that are occluded by the human body or covered by the garment, the relative error of the recovered joint angle ranges from 1.7% to 3.9%. The relative error is computed by dividing the angle difference by the angle range (upto 180 degree).

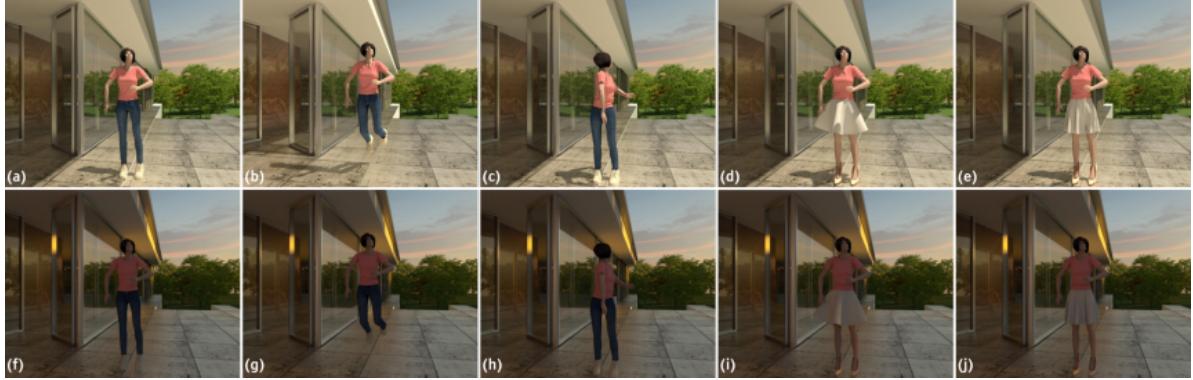


Figure 6.11: **Synthetic evaluation scenes.** (a)-(c) fixed body shape with different poses. (d)-(e) fixed body shape with a skirt of different material. (f)-(j) same scene setup as (a)-(e) but different lighting condition.

Table 6.1: **The accuracy of the recovered sizing and material parameters.** The accuracy of the recovered sizing and material parameters of the t-shirt and the pants (in percentages).

T-Shirt Pants Scene						
Pose	a	a	b	b	c	c
Lighting	Mid-Day	Sunset	Mid-Day	Sunset	Mid-Day	Sunset
\mathcal{G}_{tshirt} Accuracy	90.2	88.3	89.8	88.1	88.3	86.3
\mathcal{G}_{pants} Accuracy	89.3	87.6	85.8	83.3	88.2	87.5
\mathcal{C}_{tshirt} Accuracy	80.6	81.3	79.2	81.5	80.9	82.0
\mathcal{C}_{pants} Accuracy	80.3	78.6	80.0	80.7	80.3	80.5

6.7.2 Comparison with Other Related Work

I compare my results with the multi-view reconstruction method CMP-MVS (Jancosek and Pajdla, 2011) together with the structure-from-motion framework (Wu, 2011, 2013a). For a fair comparison, I apply smoothing (Taubin, 1995) to the results of their work. Fig. ?? and Fig. ??

Table 6.2: **The accuracy of recovered garment curvature and material parameters.** The accuracy of the recovered garment local mean curvature and material parameters of the skirt (in percentages).

T-Shirt Skirt Scene				
Material	Low Bending	Low Bending	High Bending	High Bending
Lighting	Mid-Day	Sunset	Mid-Day	Sunset
\mathcal{K}_{skirt} Accuracy	86.7	83.4	85.3	82.5
\mathcal{C}_{skirt} Accuracy	80.2	78.9	81.3	78.3

Table 6.3: **Difference between the recovered human body joint angles and the ground-truth values (in Degree) for different poses shown in Fig. 6.11.**

T-Shirt Pants Scene						
Pose	a	a	b	b	c	c
Lighting	Mid-Day	Sunset	Mid-Day	Sunset	Mid-Day	Sunset
Joint Name						
Angle Difference						
Head	3	3	3	3	3	3
Neck	0	0	0	0	0	0
Left Shoulder	0	0	0	0	4	4
Right Shoulder	0	0	0	0	4	4
Left Elbow	7	7	7	8	9	9
Right Elbow	5	6	6	6	8	7
Left Wrist	0	0	0	0	0	0
Right Wrist	0	0	0	0	0	0
Pelvis	0	0	0	0	0	0
Left Hip	0	0	4	4	0	0
Right Hip	0	0	5	6	0	0
Left Knee	0	0	5	6	0	0
Right Knee	0	0	4	5	0	0
Left Ankle	0	0	0	0	0	0
Right Ankle	0	0	0	0	0	0

show that the garment recovered using my method is clean and comparable in visual quality to the recovered garments using multi-view methods. In addition, I am able to estimate the material properties from one single-view image for virtual try-on applications.

I further compare the results of my work against two recent methods – one using 3D depth information and an expert-designed 3D database (Chen et al., 2015), and the other using a large database of manually labeled garment images (Jeong et al., 2015). My method, which does not require depth information, an expert-designed 3D database, or a large manually labeled garment



Figure 6.12: **Comparison.** (a) One frame of the video along with (b) the CMP-MVS results before and (c) after smoothing. (d) My results using only one frame of the video.

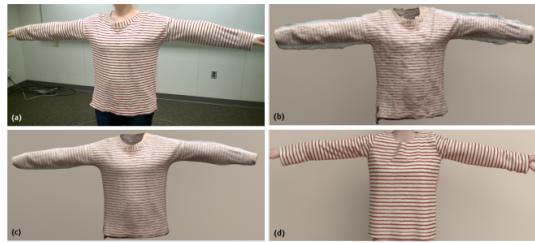


Figure 6.13: **Comparison.** (a) One frame of the multi-view video along with (b) the CMP-MVS results before and (c) after smoothing. (d) My results using only one frame of the video.

image database, achieves a comparable level of high accuracy to Chen et al. (Chen et al., 2015) (see Fig. 15) and higher visual quality when compared with Jeong et al. (Jeong et al., 2015) (see Fig. 16). In addition, my method is able to recover material and estimate sizing parameters directly from a given image.

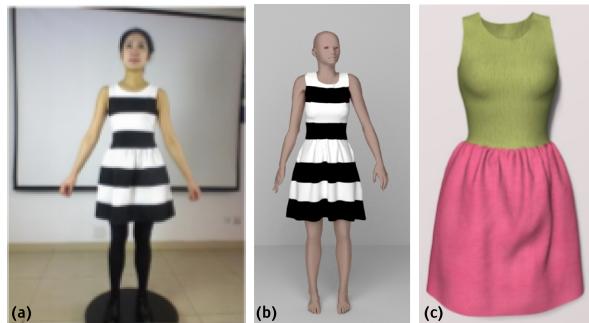


Figure 6.14: **Comparison.** (a) input image (© 2015 ACM) from paper Chen et al. (Chen et al., 2015) Figure 12. (b) my garment recover results from only a single-view RGB image (a). (c) recovery results (© 2015 ACM) from Chen et al. (Chen et al., 2015) using both RGB image and depth information.



Figure 6.15: **Comparison.** (a) input image (© 2015 Wiley) from Figure 3 in Jeong et al. (Jeong et al., 2015). (b) my garment recover results from (a). (c) recovery results (© 2015 Wiley) from Jeong et al. (Jeong et al., 2015).

6.7.3 Performance

I run my method on a desktop with an Intel(R) Core(TM) i7 CPU, 3.20GHz. For each garment, my pipeline takes on average 4 to 6 hours. The garment parameter identification (Sec. 6.5) and joint material-pose optimization (Sec. 6.6.1) takes around 60% – 80% of the entire process. The preprocessing step (Sec. 6.4.2) takes around 20% – 30%. The performance depends largely on the complexity of the garment, the image quality, and how much the garment is occluded.

6.7.4 Discussions and Limitations

Estimation of Material Parameters My material recovery method depends on the existence of wrinkles and folds of the garment. In cases where no or very few wrinkles or folds are present, other image features, such as textures and shading, would be required to identify the material properties. In most garments like shirts, skirts, or dresses, wrinkles and folds are common (especially around the joints or from the garment stylization), and can be highly informative with regards to garment material properties. Based on this observation, I am able to estimate *material parameters* as well as recover garment geometry from single-view images. This capability is one of my main objectives, and it is the key feature differentiating my work from existing techniques.

Accuracy of Geometry Reconstruction In general, it is expected that recovery from single-view images should yield less accurate results than from standard 3D reconstruction and/or the most

recent 3D multi-view methods. My method adopts accurate physics-based cloth simulation to assist in the recovery process and achieves comparable visual quality, with a focus on capturing plausible wrinkles and folds, as well as material parameters required for virtual try-on using only photographs.

However, it is important to note that high visual quality does not always guarantee geometric accuracy in 3D garment recovery. At the same time, for some applications such as virtual try-on, rapid design, and prototyping, it is unclear if a high degree of geometric accuracy is required; it is also unknown how much error tolerance is needed for the comfortable fitting of garments. These are important considerations for further investigation in application to fashion design and e-commerce.

Databases The current implementation of my approach depends on two databases: a database of commonly available garment templates and a database of human-body models. These databases introduce some limitations that can be mitigated.

Garment Templates: The range of garments I can recover is, to some extent, limited by the available garment templates. My parameter identification method can only generate garments that are “morphable” from the garment template, i.e. homeomorphic to the garment template. For example, since I use only one template for each garment type, I cannot yet model variations in some clothing details, e.g. multi-layered skirts, or collars on shirts. But for those garments that are not morphable from the template, my method can recover whichever version of the garment is closest to the actual garment. With a more extensive set of templates, I can begin to model more variations of styles and cuts, with richer garment details.

Human-body Shapes: Another limitation is the human body shape recovery. My reduced human body shape is described by a set of semantic parameters \mathbf{z} . The representation of this set of semantic parameters is not enormous, though it is sufficient to include most of the common human body shapes, as shown in my images. The known artifacts of linear human shape blending can also affect results. Aside from the human body shape recovery, my method is also limited by the state-of-art 3D human pose recovery methods. Manual intervention is needed when these methods fail to output a reasonably accurate 3D human pose.

Image Processing My current implementation cannot deal with texture, when the texture color is same as background. Although it can cope with small, partial occlusion, it cannot handle large occlusion.

6.8 Conclusion and Future Work

In this chapter I present an algorithm for physics-inspired garment recovery from a single-view image. My approach recovers a 3D mesh of the garment together with the 2D design pattern, fine wrinkles and folds, and material parameters. The recovered garment can be re-targeted to other human bodies of different shapes, sizes, and poses for virtual try-on and character animation.

In addition to addressing some limitations mentioned above, there are many possible future research directions. First of all, I plan to develop a parallelized implementation of my system on GPU or a many-core CPU for fast garment recovery. Both the underlying cloth simulator and the optimization process can be significantly accelerated. I also plan to extend my approach to enable fabric material transfer from videos for interactive virtual try-on. Furthermore, I hope to explore possible perception metrics, similar in spirit to (Sigal et al., 2015).

CHAPTER 7: LEARNING-BASED CLOTH MATERIAL RECOVERY FROM VIDEO

7.1 Introduction

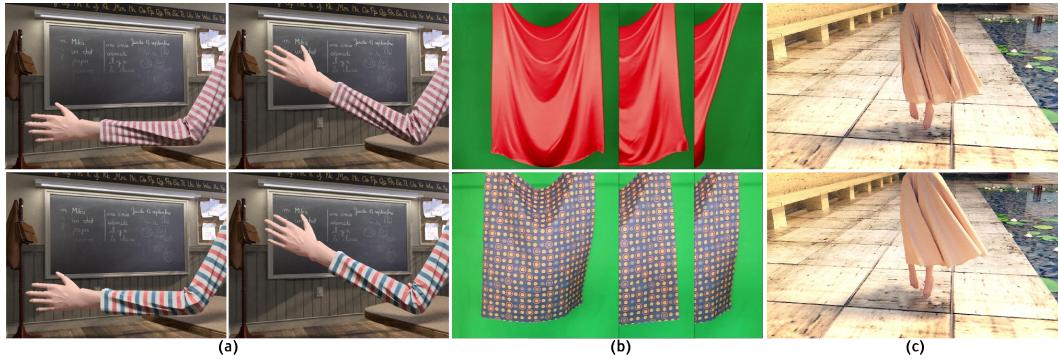


Figure 7.1: Learning-based cloth material prediction and material cloning results. (a) learning samples generated using the state-of-art physically-based cloth simulator Arcsim(Narain et al., 2012) (b) example real-life cloth motion videos presented in(Bouman et al., 2013) (c) simulated skirt with the material type predicted from the real-life video in (b) using the learned model from samples presented in (a).

Recent advances in virtual reality (VR) make it possible to recreate a vivid virtual world that can be captured as a collection of images or a video sequence. Better understanding of the physical scene can further assist in the virtual reconstruction of the real world by incorporating more realistic motion and physical interaction of virtual objects. With the introduction of the deep neural network and advances in image understanding, object detection and recognition have achieved an unprecedented level of accuracy. Capturing the physical properties of the objects in the environment can further provide a more realistic human-scene interaction. For example, in a virtual try-on system for clothing, it is critical to use material properties that correctly reflect the garment behavior; physical recreation of the fabric not only gives a compelling visual simulacrum of the cloth, but also affects how the garment feels and fits on the body. In this chapter, I propose a novel method of extracting physical information from videos in a way analogous to how humans perceive physical systems in an image or a video using “mental simulations” (Craik, 1967).

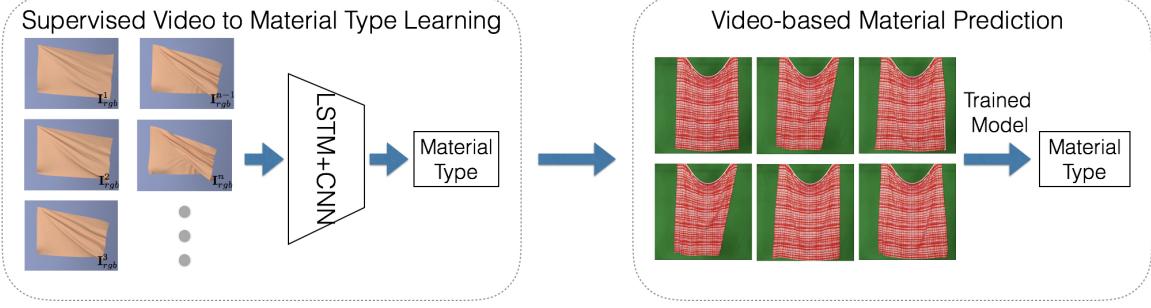


Figure 7.2: **An overview of my method.** My cloth material recovery method learns an appearance-to-material mapping model from a set of synthetic training samples. With the learned mapping model, I perform material-type prediction given a recorded video of cloth motion.

The key intuition behind my method is that the visual appearance of a piece of moving cloth encodes the intrinsic material characteristics. I use the parameters of the material model to represent the cloth’s material properties for the recorded fabrics. I adopt the cloth material model proposed by Wang et. al. (Wang et al., 2011a), which encodes the stretching and the bending of the cloth. To quantify the parameter space, I first find a parameter sub-space which discretizes the cloth material type into 54 classes. Each class defines a range of the stretching and bending parameters in the original continuous parameter space. To recover these stretching and bending parameters from the target video, I use machine learning to define the mapping between the “visual features” and the physics properties.

The visual features I use consist of the RGB information of each frame of the video. I assume that the videos are taken in controlled lighting conditions. Furthermore, I take advantages of simulated data from high-fidelity, physically-based cloth simulator to generate a very large set of videos that would be either difficult to obtain or too time-consuming and tedious to capture. With the recovered and tracked moving cloth, I can create a virtual world that uses fabrics with physical properties similar to those of the actual fabrics items in the captured video. The key contributions of this work are: a deep neural network based parameter-learning algorithm and the application of physically-based simulated data of cloth visual-to-material learning.

7.2 Related Work

Material Understanding: One of the fundamental problems in computer vision is image and video understanding. It includes the key processes, such as object segmentation (Hariharan et al., 2015; Noh et al., 2015; Dai et al., 2015; Girshick et al., 2016; Pinheiro and Collobert, 2015; Long et al., 2015b), object detection (Ren et al., 2015; Borji et al., 2015; Girshick et al., 2014; Dollár et al., 2014; Szegedy et al., 2015; Hariharan et al., 2014), object recognition (Donahue et al., 2014; Lin et al., 2014; Zhang et al., 2014; Aubry et al., 2014; Sharif Razavian et al., 2014; Russakovsky et al., 2015), scene understanding (Choi et al., 2013; Wang et al., 2013), human activities and behavior understanding (Shao et al., 2015, 2014; Solmaz et al., 2012), traffic pattern analysis (Geiger et al., 2014; Zhang et al., 2013), and surface material recognition (DelPozo and Savarese, 2007; Bell et al., 2015).

My proposed cloth material understanding is one sub-process of image/video understanding. More recently, “physical scene understanding,” which focuses on understanding the intrinsic properties of moving objects (Battaglia et al., 2013; Wu et al., 2015) has emerged as the next frontier of scene understanding. It is known that human brain can perceive dynamic systems in an image or a video. Inspired by human cognition, my method presents a computational framework that perceives the material properties of cloth in ways similar to how humans perceive dynamical systems in the physical world.

Deep Neural Network for Temporal Pattern Learning: With the advance in the artificial intelligence area, the deep neural network has been used for a vast number of tasks, especially the use of the recurrent neural network in the temporal sequence pattern learning tasks such as activity recognition (Donahue et al., 2015b; Ibrahim et al., 2016) and video captioning (Yu et al., 2016). My proposed neural network structure is inspired by the LRCN (Donahue et al., 2015b).

Use of Synthetic Data-set: The time and the energy needed to label captured data means that there is a limited amount of real-world data for training deep neural networks. Increasingly, researchers are starting to explore the use of synthetic databases to assist a variety of computer vision

tasks. For example, Chen et.al. (Chen et al., 2016) proposed a synthetic human-body data-set to help with 3D pose estimation; Keskin et.al. (Keskin et al., 2013) make use of synthetic hand images to train a hand-pose estimator; and many synthetic pedestrian data-sets (Hattori et al., 2015; Cheung et al., 2016) have been generated to study computer detection of humans in real-life images/videos.

Recovery of Physical Properties: Recovering the physical properties of a dynamical system has been a challenging problem across computer graphics, medical imaging, robotics, and computer vision for decades. And recovering physical properties of dynamical systems has become especially important with the rise of interest in VR research; the recovered physics properties from a real-life scene can be used in a virtual world or a synthetic environment to recreate a realistic animation of the given dynamical system. For example, in medical image analysis, accurately recreating the physical properties of patient tissues in virtual systems can increase diagnostic accuracy for certain kinds of diseases (Yang and Lin, 2015b; Yang et al., 2016).

Previous methods of recovering physical properties can be classified into to three key categories: measurement-based methods (Syllebranque and Boivin, 2008b; Miguel et al., 2012b; Wang et al., 2011a), which estimate the physical properties by sampling various physical quantities of the dynamical system; statistically based methods (Yang and Lin, 2016; Bouman et al., 2013; Davis et al., 2015), which learn the physical properties by observing the statistical parameters of the observed data; and iterative simulation-optimization techniques (Bhat et al., 2003; Yang and Lin, 2015b; Lee et al., 2012b; Mongus et al., 2012), which recover physical properties by simultaneously simulating the dynamical phenomena and identifying its physical properties. My method is a hybrid of these three methods. I take advantage of simulations of the dynamical phenomenon for more robust prior computations, and use the statistical method to better learn the intrinsic parameters characterizing the dynamical system, i.e. the moving cloth, in this work.

Cloth Simulation: Simulation of cloth and garments has been extensively studied in computer graphics (Bridson et al., 2002; Govindaraju et al., 2007; Narain et al., 2012). Methods for cloth simulation can be divided into two classes: one focuses on the accuracy of the simulation, and the other tackles the problem of real-time performance (Koh et al., 2014). This work takes advantage of

the state-of-art cloth simulator, ArcSim (Narain et al., 2012), which has a high degree of accuracy and visual fidelity.

7.3 Method

In this section I give a formal definition of the problem.

Problem Statement: Given a sequence of RGB images $\mathcal{V} = \{\Omega_1, \Omega_2, \dots, \Omega_N\}$, determine the type of material of the recorded cloth.

Figure 7.2 presents an overview of my approach. To constrain both my input and solution space, I first find the suitable material and the motion sub-space that can best represent the cloth material and motion in real life. Then, I exploit physically based cloth simulations to generate a much larger number of data samples within these sub-spaces that would otherwise be difficult or time-consuming to capture. The “appearance feature” of the cloth is represented by the pixel \mathbf{I}_{rgb} . With the data samples, I combine the image signal feature extraction method, Convolutional Neural Network (CNN), with the temporal sequence learning method, Long Short Term Memory (LSTM), to learn the mapping from visual “appearance” to “material”.

In the following sections, I present details of how my method learns the mapping between the visual appearance of cloth and its physical properties, and information on the generation of synthetic data-sets.

7.4 Visual, Material and Motion Representation

I first describe the visual appearance feature representation, material parameter space discretization and the motion sub-space of cloth.

7.4.1 Appearance Representation

I use the convoluted RGB color (\mathbf{I}_{rgb}) in the video as the appearance representation. I apply 5 layers of Convolutional Neural Network (CNN) to the RGB channels to extract both low and high-level visual features.

$$\mathbf{v}(\mathbf{I}_{rgb}) = W[\text{CNN}(\mathbf{I}_{rgb})] + b, \quad (7.1)$$

with W as the weights and b as the bias to be learned. The output of the final fully connected layer (fc6 layer) is the input to the LSTM as the appearance encoding.

7.4.2 Material Representation

Before I introduce my material representation, I first describe the material model I applied in my physically-based cloth simulator. Instead of using the types of manufacturing material of fabric from the physical world, I use the parameters of the material model of the physically-based simulator as the basis for representing the types of fabric material. Manufacturing fabric material, such as cotton, polyester, and linen, alone does not sufficiently define the material of the cloth. Other factors, such as the weaving patterns and thread count, also affect the material properties of a piece of cloth. Furthermore, since the driving application of this work is virtual try-on for e-commerce, my goal is to automatically determine the set of material parameters required for the physics-based cloth simulator that would reproduce the cloth dynamics observed in the video. The material model in the physically-based cloth simulator defines the cloth behavior under different external forces. The parameters of the material model thus appropriately defines the material type of the cloth under simulation. Therefore, I use the parameters of the material model of the physically-based cloth simulator to represent the types of fabric material in this work.

7.4.2.1 Material Model

The choice of material models defines the number of material types that can be approximated using a physically based simulator. In this work, I use a cloth material model proposed by Wang et al. (Wang et al., 2011a), which can be used to model most of the cloth materials in the real world.

A material model, in general, defines the relation between the stress σ and the strain ε . The cloth material consists of two sub-models, stretching and bending models. The stretching model describes how much the cloth would stretch, when subject to a certain amount of planar external forces. Similarly, the bending model defines how much the cloth would bend, when subject to

out-of-plane forces. A linear stress-strain relation can be expressed using a constant stiffness tensor matrix \mathbf{C} as: $\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon}$. To better approximate the stretching physics of a piece of cloth, Wang et al. (Wang et al., 2011a) proposed a stiffness tensor matrix that is not constant but depends on the in-plane-strain tensor $\mathbf{C}(\boldsymbol{\varepsilon})$. I refer readers to my supplementary file for detailed explanation on this material model.

7.4.2.2 Parameter Space Discretization

In the cloth material model (Wang et al., 2011a), the stretching parameter consists of 24 real numbers varying from 10 to 1000. This continuous space makes this problem intractable. To constrain my input/solution space, I discretize the original material parameter space and choose the “quantized” parameter sub-space as my material parameter sub-space. My output will be in this sub-space. To discretize the continuous material parameter space, I choose one of the material presented in the paper (Wang et al., 2011a), called “camel-ponte-roma”, as the basis. The material sub-space is constructed by multiplying this material basis with a positive coefficient. I further quantize the coefficients in continuous space to a discrete set of numbers. The size of this discrete set of numbers is the number of material types I used to represent the cloth material in real life. Using this mechanism, I discretized both the stretching and the bending parameter space.

To construct an optimal material parameter sub-space \mathcal{P} , optimal in the sense that the size of the coefficient set is minimized and the number of different real-life cloth materials that can be represented is maximized, I first conduct a material parameter sensitivity analysis. The material parameter sensitivity analysis examines the sensitivity of the material parameters κ with respect to the amount of deformation $D(\kappa)$. The sensitivity is measured as: $\frac{\partial D(\kappa)}{\partial \kappa}$, which is the slope of the curve shown in Fig. 7.3. For the stretching parameter p analysis, I hang a piece of cloth and measure the maximum amount of stretching $D(\mathcal{M})$ as in the length changes, when subjected to gravity. And, for bending parameter k sensitivity analysis, I fold a piece of cloth and keep track with the maximum curvature $C(\mathcal{M})$. The maximum amount of stretching $D(\mathcal{M})$ and the maximum curvature $C(\mathcal{M})$ are measured from the 3D mesh \mathcal{M} as follows:

$$D(\mathcal{M}) = \max_{\mathbf{u} \in V} \|\mathbf{u} - \mathbf{u}_0\|, \quad (7.2)$$

$$C(\mathcal{M}) = \max_{f_1, f_2 \in F, f_1 \cap f_2 = \mathbf{e}_0} \frac{\|\mathbf{e}_0\| \arccos(\mathbf{n}_1 \cdot \mathbf{n}_2)}{A_1 + A_2}, \quad (7.3)$$

where $\mathcal{M} = V, F, E$ is the 3D triangle mesh, which has a vertex set V , a face set F and an edge set E , of the cloth, \mathbf{u} is a vertex of the cloth's mesh \mathcal{M} and \mathbf{u}_0 is position of that vertex in rest configuration, f_1, f_2 are two adjacent faces with shared edge \mathbf{e}_0 , $\mathbf{n}_1, \mathbf{n}_2$ are their normals and A_1, A_2 are the area of those two faces.

The analysis results are shown in Fig. 7.3. The slope of the sensitivity curve (light blue) in Fig. 7.3 is positively related to how sensitive the cloth deformation/curvature is with respect to the stretching/bending coefficient. The jittering in the bending parameter sensitivity analysis is due to the re-meshing scheme. I further divide the x-axis in Fig. 7.3 into a set of segments based on the slope of the sensitivity curve. I divide the x-axis into more discrete sets when the slope of the sensitivity curve is large and vice versa. The discrete set segments of the x-axis are the stretching/bending coefficients set. Based on my analysis, the stretching parameter sub-space is $\mathcal{P}_s = \{0.5, 1, 2, 3, 10, 20\}$ and the bending parameter sub-space is $\mathcal{P}_b = \{0.5, 1, 2, 3, 4, 5, 10, 15, 20\}$. Combining the two sub-spaces $\mathcal{P} = \{(p, k) | p \in \mathcal{P}_s, k \in \mathcal{P}_b\}$, my discretized sub-space can represent 54 types of material.

To prove the validity of my material parameter sub-space, I illustrate that my material types have the ability to represent some of the commonly encountered real-life fabric material classes. I use the ten material types presented in the paper (Wang et al., 2011a) for the validation experiment. Firstly, I estimate the parameters (the floating point numbers (\tilde{p}, \tilde{k}) in Table 7.1). And then I discretize them into my subspace (the numbers in the parenthesis (p, k) in Table 7.1). As shown in Table 7.1, my discretized material types can represent these 10 types of cloth with a limited amount of error.

7.4.3 Motion Sub-space

To further make my problem tractable, I constrain the motion space of the cloth by controlling the external forces of the cloth. Under controlled external forces, the cloth moves in a motion sub-space. In addition, I need to make sure that the motion subspace is spanned in a way to capture

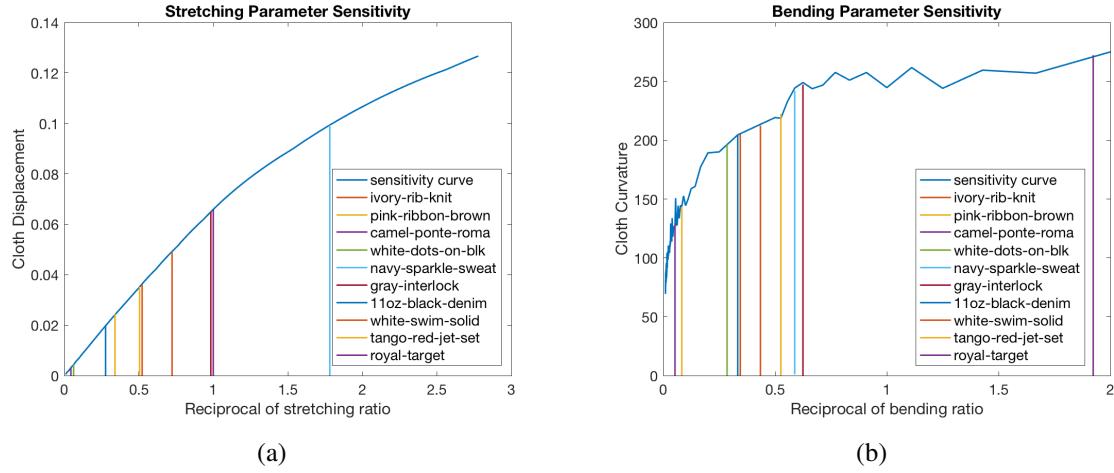


Figure 7.3: **Stretching and bending parameters sensitivity analysis results. (best view in color)** The x-axis is the reciprocal of parameter ratios to the basis material. The y-axis is the maximum amount of deformation of the cloth, i.e., maximum amount of stretching or maximum curvature, respectively. I use the vertical lines with different colors to represent the 10 types of materials presented in (Wang et al., 2011a).

the relation between the motion and the material properties of the cloth. I choose two types of external forces: constant-velocity wind blowing and fixed-size arm bending. The constant-velocity wind blowing can stretch the cloth to its maximum amount of stretching deformation, while the fixed-size arm bending can bend the cloth to its highest curvature.

7.5 Learning Method

In this section, I explain how to establish the mapping between the visual appearance of a moving cloth and its physical properties using deep neural network.

7.5.1 Deep Neural Network Structure

Design Rational: I propose to combine CNN with LSTM (similar to the LRCN (Donahue et al., 2015a) structure) for my appearance-to-material learning (network structure shown in Fig. 7.4). CNN is used to extract both low- and high-level visual features. LSTM part of the network focuses on temporal motion pattern learning. In the following sections, I will briefly introduce my network structure.

Table 7.1: **Material parameter sub-space validation.** The floating point numbers show the estimated stretching/bending parameter coefficients (\tilde{p}, \tilde{k}) , while the numbers in the parenthesis are the corresponding stretching/bending parameter (p, k) in my defined subspace \mathcal{P} .

Material (Wang et al., 2011a)	Stretching Ratio $\tilde{p}(p)$	Bending Ratio $\tilde{k}(k)$
ivory-rib-knit	1.3817(1)	2.3(2)
pink-ribbon-brown	2.9343(3)	12(10)
camel-ponte-roma	1(1)	0.52(0.5)
white-dots-on-blk	15.8108(20)	3.5(4)
navy-sparkle-sweat	0.5613(0.5)	1.7(2)
gray-interlock	1.0164(1)	1.6(2)
11oz-black-denim	3.6079(3)	3(3)
white-swim-solid	1.9126(2)	2.9(3)
tango-red-jet-set	1.9784(2)	1.9(2)
royal-target	22.2857(20)	19(20)

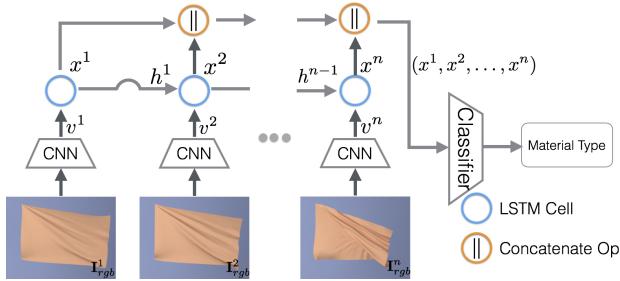


Figure 7.4: **Appearance-to-material learning method.** I apply CNN and LSTM (the original LRCN design presented in (Donahue et al., 2015a)) to learn the mapping between appearance and material.

Convolutional Neural Network for Hierarchical Visual Feature Learning: Convolutional neural network was first proposed by LeCun et. al. (LeCun et al., 1998) for digit recognition. The basis of the convolutional neural network is the convolution operation. The convolution operation serves as a filtering operation on an image. Layers of convolutional neural network (CNN) with convolution kernels of different dimensions extract features at various levels of details.

I applied a five-layer CNN (shown in Fig. 7.5) for its ability in hierarchical visual feature selection. This part of the network structure is similar to the AlexNet (Krizhevsky et al., 2012). The fifth convolution layer is followed by one fully connected layer. The output of the fully connected layer (fc6) is the input to each LSTM cell.

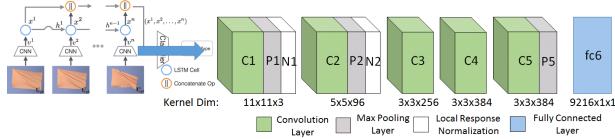


Figure 7.5: **The five-layer CNN structure.** The original design is presented in (Krizhevsky et al., 2012)

To demonstrate the effectiveness of my CNN design, I visualize the activation of the fifth convolution layer (the “conv5” layer). In Fig. 7.6, I overlay the real-life cloth moving images with the “conv5” layer activation which is visualized using the “jet” color map. The model is trained with my simulated wind-blowing data set. It is shown that I successfully trained the neural network in paying attention to the cloth area (high-lighted in yellow-red) and the cloth moving edges (high-lighted in red) of real-life images.

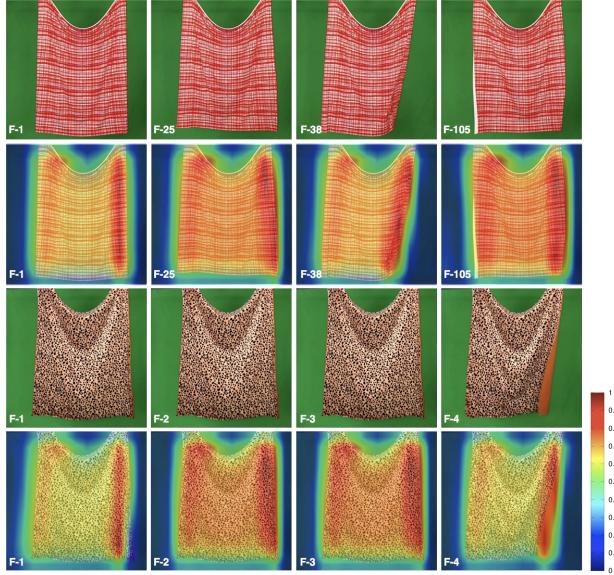


Figure 7.6: **Learned CNN conv5-layer activation visualization. (best view in color)** I overlay the conv5 layer activation using the “jet” color map with the original image. The model is trained with my simulated data set with the cloth wind-blowing motion.

Recurrent Neural Network for Sequential Pattern Learning: A single image contains a limited amount of information concerning the physics properties of a piece of cloth. But a video can be more powerful to demonstrate how the physics properties, such as the material properties of a piece of cloth, can affect its motions. To approximate this mapping between the material properties of the cloth and its sequential movement, I apply the recurrent neural network. Unlike the feed-forward neural network, the recurrent neural network has a feedback loop. The loop connects

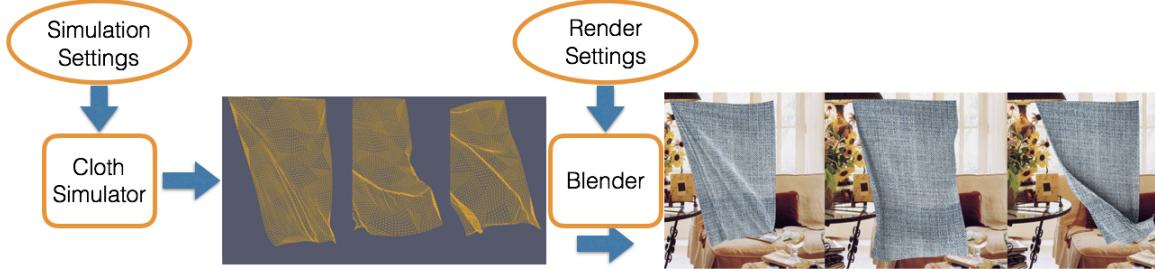


Figure 7.7: **Data generation pipeline.** My simulated data learning samples generation pipeline consists of two steps: cloth simulation and image rendering.

the output of the current cell to the input of the cell at the next step. The feedback loop act as the “memory” of the recurrent neural network. With the “memory”, the recurrent neural network has the ability to gradually extract the pattern of the input sequence.

Following the intuition behind the recurrent neural network, I choose the LSTM (Hochreiter and Schmidhuber, 1997) instead of the traditional recurrent neural network architecture for its ability to deal with vanishing/exploding gradient and fast convergence to learn the pattern in temporal sequence of data.

7.6 Physics-based Synthetic Data-sets

To learn the mapping between the visual appearance of a moving cloth and its material characteristics using a statistical method, I require a large number of data samples. Instead of using limited number of real-life recorded videos of cloth moving, I use simulation data as training samples. My synthetic data generation exploits physically based cloth simulation. This approach enables us to automatically generate a large number of data samples in a short amount of time without any manual recording or labeling.

In the following section, I will introduce my learning data samples generation pipeline.

7.6.1 Data Generation

My data generation pipeline (shown in Fig. 7.7) consists of two steps: cloth simulation and image rendering. The cloth meshes are generated through physically based simulation (ArcSim (Narain et al., 2012)). The cloth is subject to external forces such as gravity, wind and arm bending.



Figure 7.8: **Simulated data showcase.** The first three rows are example frames from my Wind-blowing data set with the cloth in pose-1. The bottom two rows are example frames from my Wind-blowing simulated data set with the cloth in pose-2 consisting of two different types of material.

Those external forces will drive the movement of the cloth. I vary the stretching, bending parameters to simulate a number of sequences of cloth motion. For each set of stretching, bending parameters I generate a sequence of 3D cloth meshes. The sequence is divided into sub-sequences as temporal training samples. Then I render each frame of the 3D cloth meshes to 2D images using Blender. The images are rendered under controlled lighting conditions and camera settings. Instead of rendering the cloth as uniform colored, for each sequence of 3D mesh, I randomly assign them with a texture image. I further composite the foreground cloth with a random background image (as shown in Fig. 7.8) to make the scene more complicated and to train the network to pay attention to the cloth area (as shown in Fig. 7.6). My background images are chosen from the indoor scene image dataset (Quattoni and Torralba, 2009).

7.7 Experiments

I implemented my method using the Caffe (Jia et al., 2014) deep neural network framework. The training process takes around 12 hours with a NVIDIA-TitanTM GPU. It takes up to 40,000 iterations to converge.

7.7.1 Data Preparation

My data set is generated using physically-based cloth simulation. By changing the simulation parameters, I obtain cloth with different material properties. I also observed two key factors that can affect the learning process: the remeshing scheme (adaptive vs. uniform resolution) and the texture of the cloth. The remeshing scheme affects the wrinkle formulation of the simulated cloth, while the texture affects the visual feature that the CNN can extract. For each motion, remeshing scheme, and texture type, I generated 2,592 sequences of cloth motion, using the method I introduced in Sec. 7.6.1. Among the 2,592 sequences, 2,106 of them were used for training and the rest 432 were used for testing. Each sequence consists of 10 frames. I tested my learned model on both the simulated data set and the real-life videos.

7.7.2 Results

My training data consists of two different types of motion: arm bending and wind blowing, with 54 material types (by varying a combination of 6 bending and 9 stretching parameters).

7.7.2.1 Baseline Results

To validate my network structure, I constructed two baseline tests. My first baseline test excludes the sequential pattern learning part (LSTM). I fine-tune the pre-trained AlexNet (Krizhevsky et al., 2012) with all the frames (210,600 images) of my training videos. Then I test my fine-tuned model on simulated data (43,200 images). Test results are shown in Table 7.2. My first baseline framework achieves 53.6% of accuracy for predicting 54 classes of materials for arm bending motion and that of 56.9% for wind blowing motion when testing on simulated data. For the second baseline, I fix my CNN part of the network but train the LSTM part. The second test aims to validate the effectiveness of my CNN sub-network. This framework obtains 56.9% of accuracy for predicting 54 classes of materials for arm bending motion and that of 57.0% for wind blowing motion when testing on simulated data. As is shown in Table 7.2 and Table 7.3, the accuracy for cloth material

type prediction from both simulated images/videos and real-life images/videos for both baseline frameworks is lower than my (CNN+LSTM) model.

Table 7.2: **Testing results.** The models are trained with the arm bending motion and wind blowing motion. Then they are tested on 432 simulated arm bending/wind blowing videos, where the ground truth is known. My method achieved up to 71.8% of accuracy for predicting 54 classes of materials for arm bending motion and upto 66.7% for wind blowing motion.

Data Setting		Method	Base-1	Base-2	CNN+LSTM
Type	Re-mesh	Texture	RGB-I	RGB-V	RGB-V
Arm	Adapt	Grid	56.0	54.0	63.3
		Color	52.9	50.2	66.0
		Rand	53.0	54.3	71.1
	Unif-1	Rand1	54.0	56.8	62.9
	Unif-2	Rand2	51.9	57	62.7
	Unif-3	Rand3	53.6	56.9	71.8
Wind	Adapt	Grid	50.4	48.0	63.4
		Color	54.0	51.2	68.0
		Rand	53.7	53.2	67.7
	Unif-1	Rand1	53.6	53.3	64.7
	Unif-2	Rand2	58.9	57.0	64.5
	Unif-3	Rand3	56.9	53.0	66.7

7.7.2.2 Validation of My Method

To validate my method, I first test the accuracy of the model trained with only the simulated arm bending motion for predicting material type of the arm bending videos. I achieve up to 71.8% of accuracy for predicting from the 54 classes of material types when using only the three-channel RGB video. The model that has the best accuracy is the one trained with the texture randomly assigned and the mesh uniformly remeshed three times. And the second best model is the one trained with the adaptive remeshing scheme and randomly assigned texture. The main reason behind this is that the meshes that are uniformly remeshed three times contain more details than the adaptive remeshed ones.

Next, I train the deep neural network with the wind blowing motion data set and test the learned model on the simulated wind blowing videos. The results are shown in Table 7.2. Similar to the

arm bending results, the best performing model is the one that is trained with the texture randomly assigned and the mesh uniformly remeshed three times. I achieve up to 66.7% of accuracy for predicting among 54 material types when using only the three-channel RGB Video.

Finally, I test my learned model on 90 real-life videos (Bouman et al., 2013). The 90 videos record the wind blowing motion of 30 kinds of cloth with three different wind strength. I correlate my predicted material type with both the ground truth stiffness value and the ground truth density value. My material types are 54 discrete numbers range from 0 to 53. The higher the number generally means the cloth is stiffer. Among the models I trained, the one which is trained with the wind blowing motion, uniformly remeshed three times, texture randomly assigned performs the best on both simulated data according to Table 7.3.

The prediction from this model also correlates the best with both the ground truth stiffness value and the ground truth density value. I achieve up to 0.50 and 0.64, respectively, as of the R value which is close to the one when human predicting material from a single image presented in (Bouman et al., 2013) for the correlation test. My experiment results also show that my prediction results is sensitive to the cloth motion as the predicted material type correlate better with the ground truth values as the strength of the wind increases (W3-Video). Further comparison analysis is given in the following section.

7.7.2.3 Comparison

In Table 7.3, I compare my method with the other cloth material recovery methods (Bouman et al., 2013) that addresses the same problem as ours. Inspired by the feature selection proposed in (Bouman et al., 2013), I propose a more general feature extraction method based on deep neural network. To make fair comparison with K. Bouman et.al. (Bouman et al., 2013), I also removed 7 videos which lack of texture or of high specularity. After excluding those 7 videos, my correlation coefficient R value for predicting cloth stiffness is 0.77 which is higher than those presented in (Bouman et al., 2013). I demonstrated in my experiments that my learned model can

Table 7.3: **Stiffness/density correlation r values for (Bouman et al., 2013) vs. Ours** My method outperforms both (Bouman et al., 2013) and human perception, achieving the highest correlation value of 0.77 and 0.84 respectively for stiffness and density, undergoing larger motion due to stronger wind (W3-video).

Method	Input	Stiffness	Density
Human (Bouman et al., 2013)	Image	0.48	0.45
Human (Bouman et al., 2013)	Video	0.73	0.83
AlexNet (baseline1)	Image	0.04	0.06
preCNN+LSTM (baseline2)	30 W3-Videos	0.12	0.13
CNN+LSTM (ours)	30 W1-Videos	0.47	0.55
CNN+LSTM (ours)	30 W2-Videos	0.43	0.62
CNN+LSTM (ours)	30 W3-Videos	0.50	0.64
K. Bouman et. al. (Bouman et al., 2013)	23 W1-Videos	0.74	0.77
K. Bouman et. al. (Bouman et al., 2013)	23 W2-Videos	0.67	0.85
K. Bouman et. al. (Bouman et al., 2013)	23 W3-Videos	0.70	0.77
CNN+LSTM (ours)	23 W1-Videos	0.71	0.75
CNN+LSTM (ours)	23 W2-Videos	0.69	0.80
CNN+LSTM (ours)	23 W3-Videos	0.77	0.84

predict material type from videos more accurately than using the feature proposed in (Bouman et al., 2013) and human perception.

7.7.3 Application

I further demonstrate my proposed framework with the application of “material cloning”. With my trained deep neural network model, I can predict the material type from a video recording the motion of the cloth in a fairly small amount of time. The recovered material type can be “cloned” on another piece of cloth or a piece of garment as shown in Fig. 7.9. I refer readers to my supplementary file for video demos.

7.7.4 Discussion and Limitations

My current learning samples are generated using physically based simulator. However, there are differences between the simulated data and real-life recorded videos. The difference comes from the numerical errors in the cloth simulation method, but also quality of the rendered images. My experiments show great promise of my learned model using data from cloth simulator in predicting material types of cloth in the real-life videos. But there is still room for improvement. With a more



Figure 7.9: **Material cloning results.** The first column are the input cloth motion videos(Bouman et al., 2013). I predict the material type of the cloth in these input videos and clone those material on to the skirt. The simulated skirt are shown in the second-sixth columns.

accurate physics simulator and more photo-realistic rendering, the proposed deep neural network framework can learn a better model from these sampled simulation data. Moreover, the neural network structure can also be improved to deal with the cross domain learning.

7.8 Conclusion and Future Work

In this chapter, I have presented a learning based algorithm to recover material properties from videos, using training datasets generated by simulators. I proposed a physically-based method to generate synthetic data-sets. My learned model can recover the physical properties (e.g. fabric material) of the cloth from a video. In this work, the videos contain only a single piece of cloth and the recorded cloth is not interacting with any other object. While this is not always the case in real-world scenarios, this method provides new insights on addressing this challenging problem. A natural extension would be to learn from videos of cloth directly interacting with the human body, under varying lighting conditions and partial occlusion.

CHAPTER 8: DISCUSSION

In this dissertation, I presented three frameworks to recover material properties of a soft body from images or videos. The first optimization simulation coupled framework is applied to recover the material properties of multi-regions of a soft body from two sets of images. The two sets of images record the deformation of the soft body. I validate the framework using both synthetic data and real-life patients data. The results for real-life patients are presented in Chapter 5. I further build a cancer staging/grading classifier using the recovered mechanical properties of the organ as one of the features. The classifier achieve upto 90% of accuracy for prostate cancer staging prediction. To recover the material properties of a dynamic soft body, I further propose a statistical graphical model based framework. I coupled a statistical graphical model with FEM simulation to optimize for the material properties of a dynamic deformable body from a video. Finally, I presented learning-based framework to identify material properties of a piece of cloth from a video.

8.1 Summary of Results

I applied my optimization simulation coupled framework to prostate cancer prediction.

8.2 Future Work

BIBLIOGRAPHY

- Agarwal, A. and Triggs, B. (2006). Recovering 3d human pose from monocular images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):44–58.
- AliExpress (2015). <http://www.aliexpress.com>.
- Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., and Davis, J. (2005). Scape: shape completion and animation of people. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 408–416. ACM.
- Anthropologie (2015). <http://www.anthropologie.com>.
- Ashab, H. A.-D., Haq, N. F., Nir, G., Kozlowski, P., Black, P., Jones, E. C., Goldenberg, S. L., Salcudean, S. E., and Moradi, M. (2015). Multimodal classification of prostate tissue: a feasibility study on combining multiparametric mri and ultrasound. In *SPIE Medical Imaging*, pages 94141B–94141B. International Society for Optics and Photonics.
- Aubry, M., Maturana, D., Efros, A. A., Russell, B. C., and Sivic, J. (2014). Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3762–3769.
- Balan, A. O., Sigal, L., Black, M. J., Davis, J. E., and Haussecker, H. W. (2007). Detailed human shape and pose from images. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.
- Balocco, S., Camara, O., and Frangi, A. F. (2008). Towards regional elastography of intracranial aneurysms. In *Medical Image Computing and Computer-Assisted Intervention*, volume 11, pages 131–8. PMID: 18982598.
- Baraff, D. and Witkin, A. (1998). Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 43–54. ACM.
- Barnfield, J. (2012). *The pattern making primer : all you need to know about designing, adapting and customizing sewing patterns*. Barron's Educational Series, Inc, Hauppauge, N.Y.
- Barr, A. H. (1984). Global and local deformations of solid primitives. In *ACM Siggraph Computer Graphics*, volume 18, pages 21–30. ACM.
- Bartels, R. H. and Beatty, J. C. (1989). A technique for the direct manipulation of spline curves. In *Graphics Interface*, volume 89, pages 33–39.
- Bathe, K.-J. (2006). *Finite element procedures*. Klaus-Jurgen Bathe.
- Battaglia, P. W., Hamrick, J. B., and Tenenbaum, J. B. (2013). Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332.
- Becker, M. and Teschner, M. (2007a). Robust and efficient estimation of elasticity parameters using the linear finite element method. *Proc. of Simulation and Visualization*, pages 15–28.

- Becker, M. and Teschner, M. (2007b). Robust and efficient estimation of elasticity parameters using the linear finite element method. *Proc. of Simulation and Visualization*, pages 15–28.
- Bell, S., Upchurch, P., Snavely, N., and Bala, K. (2015). Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487.
- Bender, R. and Grouven, U. (1997). Ordinal logistic regression in medical research. *Journal of the Royal College of Physicians of London*, 31(5):546–551.
- Bérard, P., Bradley, D., Nitti, M., Beeler, T., and Gross, M. (2014). High-quality capture of eyes. *ACM Transactions on Graphics (TOG)*, 33(6):223.
- Berkley, J., Turkiyyah, G., Berg, D., Ganter, M., and Weghorst, S. (2004). Real-time finite element modeling for surgery simulation: An application to virtual suturing. *IEEE Transactions on visualization and computer graphics*, 10(3):314–325.
- Bernard, O., Heyde, B., Alessandrini, M., Barbosa, D., Camarasu-Pop, S., Cervenansky, F., Valette, S., Mirea, O., Galli, E., Geleijnse, M., et al. (2014). Challenge on endo-cardial three-dimensional ultrasound segmentation (cetus). *Proceedings of the MICCAI Challenge on Endocardial Three-dimensional Ultrasound Segmentation-CETUS*, pages 1–8.
- Berthouzoz, F., Garg, A., Kaufman, D. M., Grinspun, E., and Agrawala, M. (2013). Parsing sewing patterns into 3d garments. *ACM Transactions on Graphics (TOG)*, 32(4):85.
- Bhat, K. S., Twigg, C. D., Hodgins, J. K., Khosla, P. K., Popović, Z., and Seitz, S. M. (2003). Estimating cloth simulation parameters from video. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 37–51. Eurographics Association.
- Bicchi, A., Canepa, G., De Rossi, D., Iacconi, P., and Scillingo, E. P. (1996). A sensor-based minimally invasive surgery tool for detecting tissue elastic properties.
- Bickel, B., Bächer, M., Otaduy, M. A., Lee, H. R., Pfister, H., Gross, M., and Matusik, W. (2010). Design and fabrication of materials with desired deformation behavior. In *ACM SIGGRAPH 2010 papers*, pages 1–10, Los Angeles, California. ACM.
- Bickel, B., Bächer, M., Otaduy, M. A., Matusik, W., Pfister, H., and Gross, M. (2009). Capture and modeling of non-linear heterogeneous soft tissue. In *ACM SIGGRAPH 2009 papers*, pages 1–9, New Orleans, Louisiana. ACM.
- Bilston, L. E. and Tan, K. (2015). Measurement of passive skeletal muscle mechanical properties in vivo: recent progress, clinical applications, and remaining challenges. *Annals of biomedical engineering*, 43(2):261–273.
- Birnholz, J. and Farrell, E. (1985). Fetal lung development: compressibility as a measure of maturity. *Radiology*, 157(2):495–498.
- Boden (2015). <http://www.bodenusa.com>.

- Boonvisut, P. and Cavusoglu, M. C. (2013). Estimation of soft tissue mechanical parameters from robotic manipulation data. *Mechatronics, IEEE/ASME Transactions on*, 18(5):1602–1611.
- Borji, A., Cheng, M.-M., Jiang, H., and Li, J. (2015). Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722.
- Bouman, K. L., Xiao, B., Battaglia, P., and Freeman, W. T. (2013). Estimating the material properties of fabric from video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1984–1991.
- Bridson, R., Fedkiw, R., and Anderson, J. (2002). Robust treatment of collisions, contact and friction for cloth animation. In *ACM Transactions on Graphics (ToG)*, volume 21, pages 594–603. ACM.
- Bridson, R., Marino, S., and Fedkiw, R. (2003). Simulation of clothing with folds and wrinkles. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 28–36. Eurographics Association.
- Bro-Nielsen, M. (1996). Surgery simulation using fast finite elements. In *Visualization in Biomedical Computing*, pages 529–534. Springer.
- Bro-Nielsen, M. and Cotin, S. (1996). Real-time volumetric deformable models for surgery simulation using finite elements and condensation. In *Computer graphics forum*, volume 15, pages 57–66. Wiley Online Library.
- Bronstein, A. M., Bronstein, M. M., and Kimmel, R. (2006). Efficient computation of isometry-invariant distances between surfaces. *SIAM Journal on Scientific Computing*, 28(5):1812–1836.
- Bronstein, A. M., Bronstein, M. M., and Kimmel, R. (2008). *Numerical geometry of non-rigid shapes*. Springer Science & Business Media.
- Brouet, R., Sheffer, A., Boissieux, L., and Cani, M.-P. (2012). Design preserving garment transfer. *ACM Trans. Graph.*, 31(4):36.
- Brouwer, I., Ustin, J., Bentley, L., Dhruv, A., and Tendick, F. (2001). Measuring in vivo animal soft tissue properties for haptic modeling in surgical. In *Medicine meets virtual reality*, volume 81, page 69.
- Cao, C., Weng, Y., Lin, S., and Zhou, K. (2013). 3d shape regression for real-time facial animation. *ACM Transactions on Graphics (TOG)*, 32(4):41.
- Carter, F. J., Frank, T. G., Davies, P. J., McLean, D., and Cuschieri, A. (2001). Measurements and modelling of the compliance of human and porcine organs. *Medical Image Analysis*, 5(4):231–236.
- Casati, R., Daviet, G., and Bertails-Descoubes, F. (2016). *Inverse Elastic Cloth Design with Contact and Friction*. PhD thesis, Inria Grenoble Rhône-Alpes, Université de Grenoble.
- Chadwick, J. E., Haumann, D. R., and Parent, R. E. (1989). Layered construction for deformable animated characters. In *ACM Siggraph Computer Graphics*, volume 23, pages 243–252. ACM.

- Chai, M., Wang, L., Weng, Y., Yu, Y., Guo, B., and Zhou, K. (2012). Single-view hair modeling for portrait manipulation. *ACM Trans. Graph.*, 31(4):116:1–116:8.
- Chang, Y.-K. and Rockwood, A. P. (1994). A generalized de casteljau approach to 3d free-form deformation. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 257–260. ACM.
- Chen, W., Wang, H., Li, Y., Su, H., Lischinsk, D., Cohen-Or, D., Chen, B., et al. (2016). Synthesizing training images for boosting human 3d pose estimation. *arXiv preprint arXiv:1604.02703*.
- Chen, X., Guo, Y., Zhou, B., and Zhao, Q. (2013). Deformable model for estimating clothed and naked human shapes from a single image. *The Visual Computer*, 29(11):1187–1196.
- Chen, X., Zhou, B., Lu, F., Wang, L., Bi, L., and Tan, P. (2015). Garment modeling with a depth camera. *ACM Transactions on Graphics (TOG)*, 34(6):203.
- Chenevert, T. L., Skovoroda, A. R., Odonnell, M., and Emelianov, S. Y. (1998). Elasticity reconstructive imaging by means of stimulated echo MRI. *Magnetic Resonance in Medicine*, 39(3):482–490.
- Cheung, E., Wong, T. K., Bera, A., Wang, X., and Manocha, D. (2016). Lcrowdv: Generating labeled videos for simulation-based crowd behavior learning. *arXiv preprint arXiv:1606.08998*.
- Choi, W., Chao, Y.-W., Pantofaru, C., and Savarese, S. (2013). Understanding indoor scenes using 3d geometric phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 33–40.
- Chopra, R., Arani, A., Huang, Y., Musquera, M., Wachsmuth, J., Bronskill, M., and Plewes, D. (2009). In vivo MR elastography of the prostate gland using a transurethral actuator. *Magnetic Resonance in Medicine*, 62(3):665–671.
- Clerc, M. (2010). *Particle swarm optimization*, volume 93. John Wiley & Sons.
- Clerc, M. (2012). Standard particle swarm optimisation.
- Cook (2017). <https://www.cookmedical.com/urology/by-the-numbers-prostate-cancer/>.
- Craik, K. J. W. (1967). *The nature of explanation*, volume 445. CUP Archive.
- Curtis, S., Tamstorf, R., and Manocha, D. (2008). Fast collision detection for deformable models using representative-triangles. In *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, pages 61–69. ACM.
- Dai, J., He, K., and Sun, J. (2015). Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3992–4000.

- Dambreville, S., Rathi, Y., and Tannenbaum, A. (2006). Tracking deformable objects with unscented kalman filtering and geometric active contours. In *American Control Conference, 2006*, pages 6–pp. IEEE.
- Davis, A., Bouman, K. L., Chen, J. G., Rubinstein, M., Durand, F., and Freeman, W. T. (2015). Visual vibrometry: Estimating material properties from small motion in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5335–5343.
- Decaudin, P., Julius, D., Wither, J., Boissieux, L., Sheffer, A., and Cani, M.-P. (2006). Virtual garments: A fully geometric approach for clothing design. In *Computer Graphics Forum*, volume 25, pages 625–634. Wiley Online Library.
- DelPozo, A. and Savarese, S. (2007). Detecting specular surfaces on natural images. In *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*, pages 1–8. IEEE.
- Dollár, P., Appel, R., Belongie, S., and Perona, P. (2014). Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1532–1545.
- Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015a). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634.
- Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015b). Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*.
- Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2014). Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655.
- Dubuisson, M.-P. and Jain, A. K. (1994). A modified hausdorff distance for object matching. In *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 566–568. IEEE.
- Engl, H. W., Kunisch, K., and Neubauer, A. (1989). Convergence rates for tikhonov regularisation of non-linear ill-posed problems. *Inverse problems*, 5(4):523.
- English, E. and Bridson, R. (2008). Animating developable surfaces using nonconforming elements. In *ACM Transactions on Graphics (TOG)*, volume 27, page 66. ACM.
- Eskandari, H., Salcudean, S., Rohling, R., and Bell, I. (2011). Real-time solution of the finite element inverse problem of viscoelasticity. *Inverse Problems*, 27(8):085002.
- Farabet, C., Couprie, C., Najman, L., and LeCun, Y. (2013). Learning hierarchical features for scene labeling. In *Pattern Analysis and Machine Intelligence*.
- FashionableShoes (2013). <http://bestfashionableshoess.blogspot.com>.
- FashionUnited (2016). Global fashion industry statistics - international apparel.

- Fowlkes, J. B., Emelianov, S. Y., Pipe, J. G., Skovoroda, A. R., Carson, P. L., Adler, R. S., and Sarvazyan, A. P. (1995). Magnetic-resonance imaging techniques for detection of elasticity variation. *Medical physics*, 22(11 Pt 1):1771–1778.
- Fu, D., Levinson, S., Gracewski, S., and Parker, K. (2000). Non-invasive quantitative reconstruction of tissue elasticity using an iterative forward approach. *PHYSICS IN MEDICINE AND BIOLOGY*, 45(6):1495–1510.
- Gao, L., Parker, K. J., Lerner, R. M., and Levinson, S. F. (1996). Imaging of the elastic properties of tissueA review. *Ultrasound in Medicine & Biology*, 22(8):959–977.
- Gao, Z., Kim, T., James, D. L., and Desai, J. P. (2009). Semi-automated soft-tissue acquisition and modeling for surgical simulation.
- Garra, B. S., Cespedes, E. I., Ophir, J., Spratt, S. R., Zuurbier, R. A., Magnant, C. M., and Pennanen, M. F. (1997). Elastography of breast lesions: initial clinical results. *Radiology*, 202(1):79–86.
- Geiger, A., Lauer, M., Wojek, C., Stiller, C., and Urtasun, R. (2014). 3d traffic scene understanding from movable platforms. *IEEE transactions on pattern analysis and machine intelligence*, 36(5):1012–1025.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2016). Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):142–158.
- Goksel, O., Eskandari, H., and Salcudean, S. E. (2013). Mesh adaptation for improving elasticity reconstruction using the fem inverse problem. *Medical Imaging, IEEE Transactions on*, 32(2):408–418.
- Goldenthal, R., Harmon, D., Fattal, R., Bercovier, M., and Grinspun, E. (2007). Efficient simulation of inextensible cloth. *ACM Transactions on Graphics (TOG)*, 26(3):49.
- Golub, G. H., Hansen, P. C., and O’Leary, D. P. (1999). Tikhonov regularization and total least squares. *SIAM Journal on Matrix Analysis and Applications*, 21(1):185–194.
- Govindaraju, N. K., Kabul, I., Lin, M. C., and Manocha, D. (2007). Fast continuous collision detection among deformable models using graphics processors. *Computers & Graphics*, 31(1):5–14.
- Gray, H. (1918). http://en.wikipedia.org/wiki/Prostatic_urethra.
- Griessmair, J. and Purgathofer, W. (1989). Deformation of solids with trivariate b-splines. In *Proceedings of eurographics*, volume 89, pages 137–148.
- Gurtin, M. E. (1982). *An introduction to continuum mechanics*. Academic press.

- Hadap, S. and Magnenat-Thalmann, N. (2001). Modeling dynamic hair as a continuum. In *Computer Graphics Forum*, volume 20, pages 329–338. Wiley Online Library.
- Hammer, P. E., Sacks, M. S., Pedro, J., and Howe, R. D. (2011). Mass-spring model for simulation of heart valve tissue mechanical behavior. *Annals of biomedical engineering*, 39(6):1668–1679.
- Haq, N. F., Kozlowski, P., Jones, E. C., Chang, S. D., Goldenberg, S. L., and Moradi, M. (2014). Improved parameter extraction and classification for dynamic contrast enhanced mri of prostate. In *SPIE Medical Imaging*, pages 903511–903511. International Society for Optics and Photonics.
- Haq, N. F., Kozlowski, P., Jones, E. C., Chang, S. D., Goldenberg, S. L., and Moradi, M. (2015a). A data-driven approach to prostate cancer detection from dynamic contrast enhanced mri. *Computerized Medical Imaging and Graphics*, 41:37–45.
- Haq, N. F., Kozlowski, P., Jones, E. C., Chang, S. D., Goldenberg, S. L., and Moradi, M. (2015b). Prostate cancer detection from model-free t1-weighted time series and diffusion imaging. In *SPIE Medical Imaging*, pages 94142X–94142X. International Society for Optics and Photonics.
- Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. (2014). Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer.
- Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. (2015). Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 447–456.
- Hasler, N., Asbach, M., Rosenhahn, B., Ohm, J.-R., and Seidel, H.-P. (2006). Physically based tracking of cloth. In *Proc. of the International Workshop on Vision, Modeling, and Visualization, VMV*, pages 49–56.
- Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B., and Seidel, H.-P. (2009). A statistical model of human pose and body shape. In *Computer Graphics Forum*, volume 28, pages 337–346. Wiley Online Library.
- Hattori, H., Naresh Boddeti, V., Kitani, K. M., and Kanade, T. (2015). Learning scene-specific pedestrian detectors without real data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3819–3827.
- Hillsweddingdress (2015). <http://hillsweddingdress.xyz>.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.
- House, D. H. and Breen, D. E. (2000). *Cloth modeling and animation*. AK Peters, Ltd.
- Hu, T. and Desai, J. P. (2004). Characterization of soft-tissue material properties: large deformation analysis. In *Medical Simulation*, pages 28–37. Springer.
- Ibrahim, M. S., Muralidharan, S., Deng, Z., Vahdat, A., and Mori, G. (2016). A hierarchical deep temporal model for group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1971–1980.

- Igarashi, T., Moscovich, T., and Hughes, J. F. (2005). As-rigid-as-possible shape manipulation. In *ACM transactions on Graphics (TOG)*, volume 24, pages 1134–1141. ACM.
- Institute, T. H. (1996). <http://www.texasheart.org/HIC/Anatomy/anatomy2.cfm>.
- Irving, G., Schroeder, C., and Fedkiw, R. (2007). Volume conserving finite element simulations of deformable models. In *ACM Transactions on Graphics (TOG)*, volume 26, page 13. ACM.
- Jancosek, M. and Pajdla, T. (2011). Multi-view reconstruction preserving weakly-supported surfaces. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3121–3128.
- Jeong, M.-H., Han, D.-H., and Ko, H.-S. (2015). Garment capture from a photograph. *Computer Animation and Virtual Worlds*, 26(3-4):291–300.
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.
- Kallel, F. and Bertrand, M. (1996). Tissue elasticity reconstruction using linear perturbation method. *Medical Imaging, IEEE Transactions on*, 15(3):299–313.
- Kauer, M., Vuskovic, V., Dual, J., Szekely, G., and Bajka, M. (2002a). Inverse finite element characterization of soft tissues.
- Kauer, M., Vuskovic, V., Dual, J., Székely, G., and Bajka, M. (2002b). Inverse finite element characterization of soft tissues. *Medical Image Analysis*, 6(3):275–287.
- Kavan, L., Sloan, P.-P., and O’Sullivan, C. (2010). Fast and efficient skinning of animated meshes. In *Computer Graphics Forum*, volume 29, pages 327–336. Wiley Online Library.
- Kennedy, J. (2010). Particle swarm optimization. In *Encyclopedia of Machine Learning*, pages 760–766. Springer.
- Kennedy, J., Eberhart, R., et al. (1995). Particle swarm optimization. In *Proceedings of IEEE international conference on neural networks*, pages 1942–1948. Perth, Australia.
- Kennedy, J. and Mendes, R. (2002). Population structure and particle swarm performance.
- Keskin, C., Kıracı, F., Kara, Y. E., and Akarun, L. (2013). Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*, pages 119–137. Springer.
- Khojaste, A., Imani, F., Moradi, M., Berman, D., Siemens, D. R., Sauerber, E. E., Boag, A. H., Abolmaesumi, P., and Mousavi, P. (2015). Characterization of aggressive prostate cancer using ultrasound rf time series. In *SPIE Medical Imaging*, pages 94141A–94141A. International Society for Optics and Photonics.
- Kleinbaum, D. G. and Klein, M. (2010). Ordinal logistic regression. In *Logistic regression*, pages 463–488. Springer.

- Koh, W., Narain, R., and O'Brien, J. F. (2014). View-dependent adaptive cloth simulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 159–166. Eurographics Association.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.
- Krouskop, T., Dougherty, D., Vinson, F., et al. (1987). A pulsed doppler ultrasonic system for making noninvasive measurements of the mechanical properties of soft tissue. *J Rehabil Res Dev*, 24(2):1–8.
- Kühnapfel, U., Cakmak, H. K., and Maaß, H. (2000). Endoscopic surgery training using virtual reality and deformable tissue simulation. *Computers & graphics*, 24(5):671–682.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- Lee, H.-P., Foskey, M., Niethammer, M., Krajcevski, P., and Lin, M. C. (2012a). Simulation-Based Joint Estimation of Body Deformation and Elasticity Parameters for Medical Image Analysis. *Medical Imaging, IEEE Transactions on*, 31(11):2156 –2168.
- Lee, H.-P., Foskey, M., Niethammer, M., Krajcevski, P., and Lin, M. C. (2012b). Simulation-based joint estimation of body deformation and elasticity parameters for medical image analysis. *IEEE transactions on medical imaging*, 31(11):2156–2168.
- Lefferts, E. J., Markley, F. L., and Shuster, M. D. (1982). Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics*, 5(5):417–429.
- Levin, A., Lischinski, D., and Weiss, Y. (2008). A closed-form solution to natural image matting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2):228–242.
- Li, H., Luo, L., Vlasic, D., Peers, P., Popović, J., Pauly, M., and Rusinkiewicz, S. (2012). Temporally coherent completion of dynamic shapes. *ACM Transactions on Graphics*, 31(1).
- Li, Y., Sun, J., Tang, C.-K., and Shum, H.-Y. (2004). Lazy snapping. In *ACM Transactions on Graphics (ToG)*, volume 23, pages 303–308. ACM.
- Liang, Z., MacFall, J. R., and Harrington, D. P. (1994). Parameter estimation and tissue segmentation from multispectral MR images.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer.
- Lindner, M. (2015). Global e-commerce sales set to grow 25% in 2015. <https://www.internetretailer.com/2015/07/29/global-e-commerce-set-grow-25-2015>.
- Liu, Y. and Storey, C. (1991). Efficient generalized conjugate gradient algorithms, part 1: theory. *Journal of Optimization Theory and Applications*, 69(1):129–137.

- Loague, K. and Green, R. E. (1991). Statistical and graphical methods for evaluating solute transport models: overview and application. *Journal of contaminant hydrology*, 7(1):51–73.
- Long, J., Shelhamer, E., and Darrell, T. (2015a). Fully convolutional networks for semantic segmentation. *CVPR (to appear)*.
- Long, J., Shelhamer, E., and Darrell, T. (2015b). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440.
- Lourakis, M. and Argyros, A. (2004). The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical report, Technical Report 340, Institute of Computer Science-FORTH, Heraklion, Crete, Greece.
- Madigan, D., York, J., and Allard, D. (1995). Bayesian graphical models for discrete data. *International Statistical Review/Revue Internationale de Statistique*, pages 215–232.
- Manduca, A., Oliphant, T. E., Dresner, M. A., Mahowald, J. L., Kruse, S. A., Amromin, E., Felmlee, J. P., Greenleaf, J. F., and Ehman, R. L. (2001). Magnetic resonance elastography: non-invasive mapping of tissue elasticity. *Medical image analysis*, 5(4):237–254.
- Meehan, M., Teschner, M., and Girod, S. (2003). Three-dimensional simulation and prediction of craniofacial surgery. *Orthodontics & Craniofacial Research*, 6(s1):102–107.
- Mehrabian, H. and Samani, A. (2009). Constrained hyperelastic parameters reconstruction of pva (polyvinyl alcohol) phantom undergoing large deformation. In *SPIE Medical Imaging*, pages 72612G–72612G. International Society for Optics and Photonics.
- Meng, Y., Wang, C. C., and Jin, X. (2012). Flexible shape control for automatic resizing of apparel products. *Computer-Aided Design*, 44(1):68–76.
- Miga, M. I. (2002). New approach to elastograph imaging: modality-independent elastography. pages 604–611.
- Miguel, E., Bradley, D., Thomaszewski, B., Bickel, B., Matusik, W., Otaduy, M., and Marschner, S. (2012a). Data-driven estimation of cloth simulation models. *Computer Graphics Forum (Eurographics 2012)*, 31(2).
- Miguel, E., Bradley, D., Thomaszewski, B., Bickel, B., Matusik, W., Otaduy, M. A., and Marschner, S. (2012b). Data-driven estimation of cloth simulation models. In *Computer Graphics Forum*, volume 31, pages 519–528. Wiley Online Library.
- Miguel, E., Tamstorf, R., Bradley, E., Schvartzman, S., Thomaszewski, B., Bickel, B., Matusik, W., Marschner, S., and Otaduy, M. (2013). Modeling and estimation of internal friction in cloth. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 32(6).
- Miller, K. (1999). Constitutive model of brain tissue suitable for finite element analysis of surgical procedures. *Journal of biomechanics*, 32(5):531–537.

- Misra, S., Ramesh, K. T., and Okamura, A. M. (2010). Modelling of non-linear elastic tissues for surgical simulation. *Computer Methods in Biomechanics and Biomedical Engineering*, 13(6):811–818.
- ModCloth (2015). <http://www.modcloth.com>.
- Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126.
- Moghari, M. H. and Abolmaesumi, P. (2007). Point-based rigid-body registration using an unscented kalman filter. *Medical Imaging, IEEE Transactions on*, 26(12):1708–1728.
- Mollemans, W., Schutyser, F., Van Cleynenbreugel, J., and Suetens, P. (2003). Tetrahedral mass spring model for fast soft tissue deformation. In *Surgery Simulation and Soft Tissue Modeling*, pages 145–154. Springer.
- Mongus, D., Repnik, B., Mernik, M., and Žalik, B. (2012). A hybrid evolutionary algorithm for tuning a cloth-simulation model. *Applied Soft Computing*, 12(1):266–273.
- Monson, C. K. and Seppi, K. D. (2005). Exposing origin-seeking bias in pso. In *Proceedings of the 2005 conference on Genetic and evolutionary computation*, pages 241–248. ACM.
- Moons, T., Van Gool, L., and Vergauwen, M. (2010). 3d reconstruction from multiple images. *Foundations and Trends in Computer Graphics and Vision*, 4(4):287–404.
- Moore, B., Jaglinski, T., Stone, D., and Lakes, R. (2007). On the bulk modulus of open cell foams. *Cellular Polymers*, 26(1):1–10.
- Moré, J. J. (1978). The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer.
- Müller, M. and Gross, M. (2004). Interactive virtual materials. In *Proceedings of Graphics Interface 2004*, GI '04, page 239–246, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada. Canadian Human-Computer Communications Society.
- Muthupillai, R. and Ehman, R. L. (1996). Magnetic resonance elastography. *Nat Med*, 2(5):601–603.
- Nagano, K., Fyffe, G., Alexander, O., Barbiç, J., Li, H., Ghosh, A., and Debevec, P. (2015). Skin microstructure deformation with displacement map convolution. *ACM Trans. Graph.*, 34(4):109:1–109:10.
- Narain, R., Samii, A., and O’Brien, J. F. (2012). Adaptive anisotropic remeshing for cloth simulation. *ACM transactions on graphics (TOG)*, 31(6):152.
- Nava, A., Mazza, E., Kleinermann, F., Avis, N. J., and McClure, J. (2003). Determination of the mechanical properties of soft human tissues through aspiration experiments. In *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2003*, pages 222–229. Springer.
- Nealen, A., Muller, M., Keiser, R., Boxerman, E., and Carlson, M. (2006). Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25:809–836.

- Nedel, L. P. and Thalmann, D. (1998). Real time muscle deformations using mass-spring systems. In *Computer Graphics International, 1998. Proceedings*, pages 156–165. IEEE.
- Ng, H. N. and Grimsdale, R. L. (1996). Computer graphics techniques for modeling cloth. *Computer Graphics and Applications, IEEE*, 16(5):28–41.
- Noh, H., Hong, S., and Han, B. (2015). Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528.
- Ophir, J., Alam, S., Garra, B., Kallel, F., Konofagou, E., Krouskop, T., and Varghese, T. (1999). Elastography: ultrasonic estimation and imaging of the elastic properties of tissues. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 213(3):203–233.
- Ophir, J., Céspedes, I., Ponnekanti, H., Yazdi, Y., and Li, X. (1991a). Elastography: A quantitative method for imaging the elasticity of biological tissues. *Ultrasonic Imaging*, 13(2):111–134.
- Ophir, J., Cespedes, I., Ponnekanti, H., Yazdi, Y., and Li, X. (1991b). Elastography: a quantitative method for imaging the elasticity of biological tissues. *Ultrasonic imaging*, 13(2):111–134.
- Ottensmeyer, M. P. (2001). *Minimally invasive instrument for in vivo measurement of solid organ mechanical impedance*. PhD thesis, Massachusetts Institute of Technology.
- Pai, D. K., Doel, K. v. d., James, D. L., Lang, J., Lloyd, J. E., Richmond, J. L., and Yau, S. H. (2001). Scanning physical interaction behavior of 3D objects. In *Proceedings of SIGGRAPH 2001*, SIGGRAPH '01, page 87–96, New York, NY, USA. ACM.
- Piegl, L. (1989a). Modifying the shape of rational b-splines. part 1: curves. *Computer-Aided Design*, 21(8):509–518.
- Piegl, L. (1989b). Modifying the shape of rational b-splines. part 2: surfaces. *Computer-Aided Design*, 21(9):538–546.
- Pinheiro, P. H. and Collobert, R. (2014). Recurrent convolutional neural networks for scene labeling. In *ICML*.
- Pinheiro, P. O. and Collobert, R. (2015). From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1713–1721.
- Poli, R., Kennedy, J., and Blackwell, T. (2007). Particle swarm optimization. *Swarm intelligence*, 1(1):33–57.
- Popa, T., Zhou, Q., Bradley, D., Kraevoy, V., Fu, H., Sheffer, A., and Heidrich, W. (2009). Wrinkling captured garments using space-time data-driven deformation. In *Computer Graphics Forum*, volume 28, pages 427–435. Wiley Online Library.
- Press, W. H. (2007). *Numerical recipes*. Cambridge University Press.

- Protopsaltou, D., Luible, C., Arevalo, M., and Magnenat-Thalmann, N. (2002). *A body and garment creation method for an Internet based virtual fitting room*. Springer.
- Quattoni, A. and Torralba, A. (2009). Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 413–420. IEEE.
- RedBubble (2015). <http://www.redbubble.com>.
- Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.
- Ren, Z., Yeh, H., and Lin, M. C. (2013). Example-guided physically based modal sound synthesis. *ACM Transactions on Graphics (TOG)*, 32(1):1.
- Rivaz, H., Boctor, E., Foroughi, P., Zellars, R., Fichtinger, G., and Hager, G. (2008). Ultrasound elastography: A dynamic programming approach. *Medical Imaging, IEEE Transactions on*, 27(10):1373–1377.
- Rivlin, R. (1948). Large elastic deformations of isotropic materials. iv. further developments of the general theory. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 241(835):379–397.
- Rivlin, R. S. and Saunders, D. (1951). Large elastic deformations of isotropic materials. vii. experiments on the deformation of rubber. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 243(865):251–288.
- Robson, C., Maharik, R., Sheffer, A., and Carr, N. (2011). Context-aware garment modeling from sketches. *Computers & Graphics*, 35(3):604–613.
- Rogowska, J., Patel, N., Plummer, S., and Brezinski, M. (2014). Quantitative optical coherence tomographic elastography: method for assessing arterial mechanical properties. *The British journal of radiology*.
- Rohmer, D., Popa, T., Cani, M.-P., Hahmann, S., and Sheffer, A. (2010). Animation wrinkling: augmenting coarse cloth simulations with realistic-looking wrinkles. In *ACM Transactions on Graphics (TOG)*, volume 29, page 157. ACM.
- Rosen, J., Brown, J. D., Sinanan, M., Hannaford, B., and De, S. (2008). Biomechanical properties of abdominal organs in vivo and postmortem under compression loads. *Journal of Biomechanical Engineering*, 130(2):021020.
- Rosen, J., Hannaford, B., MacFarlane, M. P., and Sinanan, M. N. (1999). Force controlled and teleoperated endoscopic grasper for minimally invasive surgery-experimental performance evaluation.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252.

- Saacclothes (2015). <http://www.saacclothes.com>.
- Saad, Y. and Schultz, M. H. (1986). Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869.
- Samur, E., Sedef, M., Basdogan, C., Avtan, L., and Duzgun, O. (2007). A robotic indenter for minimally invasive measurement and characterization of soft tissue response. *Medical Image Analysis*, 11(4):361–373.
- Schnur, D. S. and Zabaras, N. (1992). An inverse method for determining elastic material properties and a material interface. *International Journal for Numerical Methods in Engineering*, 33(10):2039–2057.
- Scholz, V. and Magnor, M. (2006). Texture replacement of garments in monocular video sequences. In *Proceedings of the 17th Eurographics conference on Rendering Techniques*, pages 305–312. Eurographics Association.
- Scholz, V., Stich, T., Keckeisen, M., Wacker, M., and Magnor, M. (2005). Garment motion capture using color-coded patterns. In *Computer Graphics Forum*, volume 24, pages 439–447. Wiley Online Library.
- Selle, A., Lentine, M., and Fedkiw, R. (2008). A mass spring model for hair simulation. *ACM Transactions on Graphics (TOG)*, 27(3):64.
- Seo, H. and Magnenat-Thalmann, N. (2003). An automatic modeling of human bodies from sizing parameters. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 19–26. ACM.
- Shahim, K., Jürgens, P., Cattin, P. C., Nolte, L.-P., and Reyes, M. (2013). Prediction of cranio-maxillofacial surgical planning using an inverse soft tissue modelling approach. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013*, pages 18–25. Springer.
- Shannon, T. (2009). <http://www.trevorshp.com/photography/hsivideos/index.htm>.
- Shao, J., Change Loy, C., and Wang, X. (2014). Scene-independent group profiling in crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2219–2226.
- Shao, J., Kang, K., Loy, C. C., and Wang, X. (2015). Deeply learned attributes for crowded scene understanding. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4657–4666. IEEE.
- Sharif Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 806–813.

- Shi, P. and Liu, H. (2003). Stochastic finite element framework for simultaneous estimation of cardiac kinematic functions and material parameters. *Medical Image Analysis*, 7(4):445–464.
- Shi, Y. and Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. In *Evolutionary Programming VII*, pages 591–600. Springer.
- Si, H. (2007). Tetgen. a quality tetrahedral mesh generator and three-dimensional delaunay triangulator.
- Si, H. (2015). Tetgen, a delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Softw.*, 41(2):11:1–11:36.
- Sifakis, E., Selle, A., Robinson-Mosher, A., and Fedkiw, R. (2006). Simulating speech with a physics-based facial muscle model. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 261–270. Eurographics Association.
- Sigal, L., Mahler, M., Diaz, S., McIntosh, K., Carter, E., Richards, T., and Hodgins, J. (2015). A perceptual control space for garment simulation. *ACM Transactions on Graphics (TOG)*, 34(4):117.
- Sissler, L., Jones, R., Leaney, P., and Harland, A. (2010). Viscoelastic modelling of tennis ball properties. In *IOP Conference Series: Materials Science and Engineering*, volume 10, page 012114. IOP Publishing.
- Skovoroda, A. and Emelianov, S. (1995). Tissue elasticity reconstruction based on ultrasonic displacement and strain images. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 42(4):141.
- Snavely, N. (2008). Bundler v0.9.
- Sobin, L. H., Gospodarowicz, M. K., and Wittekind, C. (2011). *TNM classification of malignant tumours*. John Wiley & Sons.
- Solmaz, B., Moore, B. E., and Shah, M. (2012). Identifying behaviors in crowd scenes using stability analysis for dynamical systems. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):2064–2070.
- stylewe (2014). <https://www.stylewe.com>.
- Sumner, R. W. and Popović, J. (2004). Deformation transfer for triangle meshes. *ACM Transactions on Graphics (TOG)*, 23(3):399–405.
- Syllebranque, C. and Boivin, S. (2008a). Estimation of mechanical parameters of deformable solids from videos. *The Visual Computer*, 24(11):963–972.
- Syllebranque, C. and Boivin, S. (2008b). Estimation of mechanical parameters of deformable solids from videos. *The Visual Computer*, 24(11):963–972.

- Syllebranque, C., Boivin, S., Duriez, C., and Chaillou, C. (2007). Estimation of hookean parameters of deformable bodies from real videos. In *Cyberworlds, 2007. CW'07. International Conference on*, pages 330–337. IEEE.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9.
- Tak, S. and Ko, H.-S. (2005). A physically-based motion retargeting filter. *ACM Transactions on Graphics (TOG)*, 24(1):98–117.
- Tang, D., Yang, C., Kobayashi, S., and Ku, D. N. (2001). Steady flow and wall compression in stenotic arteries: a three-dimensional thick-wall model with fluid–wall interactions. *Journal of biomechanical engineering*, 123(6):548–557.
- Tang, M., Curtis, S., Yoon, S.-E., and Manocha, D. (2009). Iccd: Interactive continuous collision detection between deformable models using connectivity-based culling. *Visualization and Computer Graphics, IEEE Transactions on*, 15(4):544–557.
- Tanie, H., Yamane, K., and Nakamura, Y. (2005). High marker density motion capture by retroreflective mesh suit. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2884–2889. IEEE.
- Taubin, G. (1995). A signal processing approach to fair surface design. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '95*, pages 351–358, New York, NY, USA. ACM.
- Taylor, C. J. (2000). Reconstruction of articulated objects from point correspondences in a single uncalibrated image. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 677–684. IEEE.
- Teran, J., Blemker, S., Hing, V., and Fedkiw, R. (2003). Finite volume methods for the simulation of skeletal muscle. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 68–74. Eurographics Association.
- Teran, J., Sifakis, E., Irving, G., and Fedkiw, R. (2005). Robust quasistatic finite elements and flesh simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 181–190. ACM.
- Terzopoulos, D., Platt, J., Barr, A., and Fleischer, K. (1987). Elastically deformable models. 21:205–214.
- Thomaszewski, B., Pabst, S., and Strasser, W. (2009). Continuum-based strain limiting. In *Computer Graphics Forum*, volume 28, pages 569–576. Wiley Online Library.
- Treloar, L. R., Hopkins, H., Rivlin, R., and Ball, J. (1976). The mechanics of rubber elasticity [and discussions]. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 351(1666):301–330.

- Tsutsumi, M., Miyagawa, T., Matsumura, T., Kawazoe, N., Ishikawa, S., Shimokama, T., Shiina, T., Miyanaga, N., and Akaza, H. (2007). The impact of real-time tissue elasticity imaging (elastography) on the detection of prostate cancer: clinicopathological analysis. *International journal of clinical oncology*, 12(4):250–255.
- Turquin, E., Wither, J., Boissieux, L., Cani, M.-P., and Hughes, J. F. (2007). A sketch-based interface for clothing virtual characters. *IEEE Computer Graphics and Applications*, (1):72–81.
- Uniyal, N., Imani, F., Tahmasebi, A., Agarwal, H., Bharat, S., Yan, P., Kruecker, J., Kwak, J. T., Xu, S., Wood, B., et al. (2014). Ultrasound-based predication of prostate cancer in mri-guided biopsy. In *Clinical Image-Based Procedures. Translational Research in Medical Imaging*, pages 142–150. Springer.
- Van Houten, E., Paulsen, K., Miga, M., Kennedy, F., and Weaver, J. (1999). An overlapping subzone technique for MR-based elastic property reconstruction. *Magnetic Resonance in Medicine*, 42(4):779–786.
- Van Houten, E. E., Miga, M. I., Weaver, J. B., Kennedy, F. E., and Paulsen, K. D. (2001). Three-dimensional subzone-based reconstruction algorithm for MR elastography. *Magnetic Resonance in Medicine*, 45(5).
- Vassilev, T., Spanlang, B., and Chrysanthou, Y. (2001). Fast cloth animation on walking avatars. In *Computer Graphics Forum*, volume 20, pages 260–267. Wiley Online Library.
- Vavourakis, V., Hipwell, J. H., and Hawkes, D. J. (2016). An inverse finite element u/p-formulation to predict the unloaded state of in vivo biological soft tissues. *Annals of biomedical engineering*, 44(1):187–201.
- Veronda, D. and Westmann, R. (1970). Mechanical characterization of skinfinite deformations. *Journal of biomechanics*, 3(1):111–124.
- Volino, P. and Magnenat-Thalmann, N. (1999). Fast geometrical wrinkles on animated surfaces. In *Seventh International Conference in Central Europe on Computer Graphics and Visualization (Winter School on Computer Graphics)*.
- Wan, E. A. and Van Der Merwe, R. (2000). The unscented kalman filter for nonlinear estimation. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pages 153–158. IEEE.
- Wang, B., Wu, L., Yin, K., Ascher, U., Liu, L., and Huang, H. (2015). Deformation capture and modeling of soft objects. *ACM Transactions on Graphics (TOG)*, 34(4):94.
- Wang, C. C., Wang, Y., and Yuen, M. M. (2005). Design automation for customized apparel products. *Computer-Aided Design*, 37(7):675–691.
- Wang, H., Gould, S., and Roller, D. (2013). Discriminative learning with latent variables for cluttered indoor scene understanding. *Communications of the ACM*, 56(4):92–99.

- Wang, H., O'Brien, J., and Ramamoorthi, R. (2010). Multi-resolution isotropic strain limiting. In *ACM Transactions on Graphics (TOG)*, volume 29, page 156. ACM.
- Wang, H., O'Brien, J. F., and Ramamoorthi, R. (2011a). Data-driven elastic models for cloth: modeling and measurement. *ACM Transactions on Graphics (TOG)*, 30(4):71.
- Wang, H., Ramamoorthi, R., and O'Brien, J. (2011b). Data-driven elastic models for cloth: Modeling and measurement. *ACM Transactions on Graphics (SIGGRAPH)*, 30(4):71:1–71:12.
- Washington, C. W. and Miga, M. I. (2004). Modality independent elastography (MIE): a new approach to elasticity imaging. *IEEE Trans. on Medical Image*, 23(9):1117–1128.
- Weil, J. (1986). The synthesis of cloth objects. *ACM Siggraph Computer Graphics*, 20(4):49–54.
- Westwood, J. D. et al. (2005). A gpu accelerated spring mass system for surgical simulation. *Medicine Meets Virtual Reality 13: The Magical Next Becomes the Medical Now*, 111:342.
- White, R., Crane, K., and Forsyth, D. A. (2007). Capturing and animating occluded cloth. In *ACM Transactions on Graphics (TOG)*, volume 26, page 34. ACM.
- Wilson, L. S. and Robinson, D. E. (1982). Ultrasonic measurement of small displacements and deformations of tissue. *Ultrasonic Imaging*, 4(1):71–82.
- Wójcicki, K., Puciłowski, K., and Kulesza, Z. (2011). Mathematical analysis for a new tennis ball launcher. *acta mechanica et automatica*, 5:110–119.
- Wu, C. (2011). Visualsfm: A visual structure from motion system. URL: <http://homes.cs.washington.edu/~ccwu/vsfm>, 9.
- Wu, C. (2013a). Towards linear-time incremental structure from motion. In *3D Vision - 3DV 2013, 2013 International Conference on*, pages 127–134.
- Wu, C. (2013b). Visualsfm : A visual structure from motion system.
- Wu, J., Yildirim, I., Lim, J. J., Freeman, B., and Tenenbaum, J. (2015). Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In *Advances in Neural Information Processing Systems*, pages 127–135.
- Xie, S. and Tu, Z. (2015). Holistically-nested edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1395–1403.
- Yamaguchi, K., Kiapour, M. H., and Berg, T. (2013). Paper doll parsing: Retrieving similar styles to parse clothing items. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3519–3526. IEEE.
- Yang, S., Jojic, V., Lian, J., Chen, R., Zhu, H., and Lin, M. C. (2016). Classification of prostate cancer grades and t-stages based on tissue elasticity using medical image analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 627–635. Springer.

- Yang, S. and Lin, M. (2015a). Material property reconstruction from time sequential images. Technical report, Department of Computer Science, University of North Carolina at Chapel Hill.
- Yang, S. and Lin, M. (2015b). Materialcloning: Acquiring elasticity parameters from images for medical applications. *IEEE transactions on visualization and computer graphics*.
- Yang, S. and Lin, M. C. (2016). Bayesian estimation of non-rigid mechanical parameters using temporal sequences of deformation samples. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 4036–4043. IEEE.
- Yang, Y., Yu, Y., Zhou, Y., Du, S., Davis, J., and Yang, R. (2014). Semantic parametric reshaping of human body models. In *3D Vision (3DV), 2014 2nd International Conference on*, volume 2, pages 41–48. IEEE.
- Ye, M., Wang, H., Deng, N., Yang, X., and Yang, R. (2014). Real-time human pose and shape estimation for virtual try-on using a single commodity depth camera. *IEEE transactions on visualization and computer graphics*, 20(4):550–559.
- Young, S., Adelstein, B., and Ellis, S. (2007). Calculus of nonrigid surfaces for geometry and texture manipulation. *Visualization and Computer Graphics, IEEE Transactions on*, 13(5):902–913.
- Yu, H., Wang, J., Huang, Z., Yang, Y., and Xu, W. (2016). Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4584–4593.
- Yushkevich, P. A., Piven, J., Cody Hazlett, H., Gimpel Smith, R., Ho, S., Gee, J. C., and Gerig, G. (2006a). User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128.
- Yushkevich, P. A., Piven, J., Hazlett, H. C., Smith, R. G., Ho, S., Gee, J. C., and Gerig, G. (2006b). User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage*, 31(3):1116–1128.
- Zhan, R. and Wan, J. (2007). Iterated unscented kalman filter for passive target tracking. *Aerospace and Electronic Systems, IEEE Transactions on*, 43(3):1155–1163.
- Zhang, H., Geiger, A., and Urtasun, R. (2013). Understanding high-level semantics by modeling traffic patterns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3056–3063.
- Zhang, N., Donahue, J., Girshick, R., and Darrell, T. (2014). Part-based r-cnns for fine-grained category detection. In *European Conference on Computer Vision*, pages 834–849. Springer.
- Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM computing surveys (CSUR)*, 35(4):399–458.
- Zheng, Y.-p. and Mak, A. F. T. (1996). An ultrasound indentation system for biomechanical properties assessment of soft tissues in-vivo.

- Zhou, B., Chen, X., Fu, Q., Guo, K., and Tan, P. (2013). Garment modeling from a single image. In *Computer Graphics Forum*, volume 32, pages 85–91. Wiley Online Library.
- Zhou, S., Fu, H., Liu, L., Cohen-Or, D., and Han, X. (2010). Parametric reshaping of human bodies in images. In *ACM Transactions on Graphics (TOG)*, volume 29, page 126. ACM.
- Zhu, Y., Hall, T., and Jiang, J. (2003a). A finite-element approach for young’s modulus reconstruction. *Medical Imaging, IEEE Transactions on*, 22(7):890–901.
- Zhu, Y., Hall, T., and Jiang, J. (2003b). A finite-element approach for Young’s modulus reconstruction. *Medical Imaging, IEEE Transactions on*, 22(7):890 –901.