Virtual Environments for Medical Training: Graphic and Haptic Simulation of Tool-Tissue Interactions

By

Jung Kim
M.S. Precision Engineering and Mechatronics
Korea Advanced Institute of Science and Technology, 1993

SUBMITTED TO THE
DEPARTMENT OF MECHANICAL ENGINEERING
IN PARTICAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY IN MECHANICAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
FEBRUARY 2004

Author _____

Department of Mechanical Engineering

December 12, 2003

Certified by _____

Dr. Mandayam A. Srinivasan

Thesis Supervisor, Department of Mechanical Engineering

Accepted by

_____

Professor Ain A. Sonin

Chairman, Department Committee on Graduate Students

Virtual Environments for Medical Training: Graphic and Haptic Simulation of Tool-Tissue Interactions

By

Jung Kim

Submitted to the Department of Mechanical Engineering

on December 12, 2003, in Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy in Mechanical Engineering

Abstract

For more than 2,500 years, surgical teaching has been based on the so called "see one, do one, teach one" paradigm, in which the surgical trainee learns by operating on patients under close supervision of peers and superiors. However, higher demands on the quality of patient care and rising malpractice costs have made it increasingly risky to train on patients. Minimally invasive surgery, in particular, has made it more difficult for an instructor to demonstrate the required manual skills. It has been recognized that, similar to flight simulators for pilots, virtual reality (VR) based surgical simulators promise a safer and more comprehensive way to train manual skills of medical personnel in general and surgeons in particular. One of the major challenges in the development of VR-based surgical trainers is the real-time and realistic simulation of interactions between surgical instruments and biological tissues. It involves multi-disciplinary research areas including soft tissue mechanical behavior, tool-tissue contact mechanics, computer haptics, computer graphics and robotics integrated into VR-based training systems.

The research described in this thesis addresses many of the problems of simulating tool-tissue interactions in medical virtual environments. First, two kinds of physically based real time soft tissue models - the local deformation and the hybrid deformation model - were developed to compute interaction forces and visual deformation fields that provide real-time feed back to the user. Second, a system to measure *in vivo* mechanical properties of soft tissues was designed, and eleven sets of animal experiments were performed to measure *in vivo* and *in vitro* biomechanical properties of porcine intra-abdominal organs. Viscoelastic tissue parameters were then extracted by matching finite element model predictions with the empirical data. Finally, the tissue parameters were combined with geometric organ models segmented from the Visible Human Dataset and integrated into a minimally invasive surgical simulation system consisting of haptic interface devices inside a mannequin and a graphic display. This system was used to demonstrate deformation and cutting of the esophagus, where

the user can haptically interact with the virtual soft tissues and see the corresponding organ deformation on the visual display at the same time.

Thesis Supervisor:

Dr. Mandayam A. Srinivasan, Dept. of Mechanical Engineering

Thesis Committee:

Prof. Sanjay Sarma, Dept. of Mechanical Engineering

Dr. David Brock, Dept. of Mechanical Engineering

Dr. Mark P. Ottensmeyer, CIMIT/MGH

Dedication

To the memory of my father

Seung Jik Kim

Acknowledgement

Since I learned Korean characters from my grandmother at five years old, my education in school has finally reaches to the final chapter. First of all, I would like to express my love and respect to my father, Sengjik Kim. He always gave me encouragements throughout his entire life. He has been a role model and hero in every moment of my life. Father, I always miss you. I also want to thank my mother for her love and support to us.

Special thanks go to my advisor Dr. Srinivasan for giving me the opportunity to work in such a great research and for his encouragement, understanding, patience, and guidance throughout the study. I have also appreciated my committee members, Professor Sanjay Sarma, Dr. David Brock and Dr. Mark P. Ottensmeyer for their helpful suggestions and precious comments.

I have appreciated my TouchLab fellow members and alumni for their help and for making the lab a pleasurable place to work. They have contributed many parts in this thesis.

Love to my wife Hyeyoung for her great love and my lovely daughter, Justine the gift from the God. Because of them, the entire period of my study is full of happiness and the God's blessings. I also want to thank to my parents-in-law for their concern and love.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

The training paradigm for surgeons has not changed substantially in many centuries. Surgical teaching has been based traditionally on the "preceptor" or "apprenticeship" model, in which the novice surgeon learns with small groups of peers and supervisors, over time, in the course of patient care. The novice surgeon acquires skills by first observing experienced surgeons in action and then by progressively performing, under varying degrees of supervision, more of the surgical procedures, as his/her training advances and his/her skill levels increases.

This so called "see one, do one, teach one" paradigm, has proven to be reasonably effective for more than 2,500 years. However, there are many reasons to develop non-patient methods for teaching and evaluating surgical procedures. Significant reasons include: 1) the increasing sensitivity of our society about the ethical issues in using patients for teaching purposes 2) the high cost of training facilities 3) the increasing intolerance of error 4) the difficulty in standardizing testing conditions and 5) the difficulty in ensuring appropriate coverage of new procedures. These issues have led to a variety of efforts around the world to develop non-patient platforms for teaching and testing of procedural skills for surgeons.

Virtual reality (VR) based training systems have the ability to revolutionize medical education and augment training through: (1) quantification of performance and progress, (2) standardization of training conditions independent of patient population, and (3) exposure to rare but important procedures. In addition to allowing novice and experienced surgeons to practice new techniques, surgical simulators may allow the trainee to encounter "patients" with rare medical conditions and practice techniques to handle these situations. Thus, this computer based simulation is a promising educational tool.

For the past several years, aerospace, maritime, military, nuclear energy and other high-risk professions have been using simulators for training difficult and demanding tasks. The success of flight simulators provides a convincing example of the importance of simulation technology when applied to the teaching of skills. A real time multi-modal simulation system that provides the surgeon with visual and haptic cues promises to be a powerful aid for training medical personnel and monitoring their performance and has several major advantages over traditional training:

- The simulation system can be used to certify medical personnel based on their practical skills in executing a surgical procedure. Furthermore, training course can be customized to each individual being trained, based on that trainee's skills, deficiencies, and progress.
- The simulation system can reduce the number of animals annually sacrificed in order to train medical personnel. In addition to the sensitivity of the public to animal-based training, this type of training is expensive and only one animal can be used per training session.
- Functions can be built into the software of a surgical system to record performance of master surgeons and to use this performance to automatically tutor surgeons in ideal manual performance of new surgical tasks.

Although medical imaging techniques and multi-modal interfaces like haptic interfaces are available, one unsolved but essential problem is tissue modeling [1]. To determine the correct forces and visual deformation fields to feed back in response to user actions, the simulator must compute the deformation of the target tissues in real time. In addition, the lack of data in current literature on *in vivo* mechanical properties of organs has been a significant impediment to the development of the simulator because the properties of organs change significantly posthumously.

This thesis presents the issues of tissue modeling in medical virtual environments. First, we will develop a soft tissue model simulating tool tissue interactions in real time. Second, *in vivo* measurements of intra-abdominal tissues and characterization techniques will be presented. Finally, we will present the VR-based simulator and discuss implementation issues in comprehensive medical training environments.

The following chapters will begin with an introduction to background issues of VR-based surgical simulation. Chapter 2 will include a brief review of minimally invasive surgical procedures and related technologies required for surgical simulation. This section describes the scope of the research covered by this thesis. The real time tissue modeling will be described in Chapters 3 and 4. Chapter 3 begins with a review of previous work in deformation modeling, and then gives a detailed description of real time simulation of tissue deformations induced by tools. Chapter 4 describes development of hybrid models of the organs providing physically based organ deformation and haptic feedback. Chapters 5 and 6 are dedicated to methods for characterizing soft tissue properties of living subjects, which are useful for the simulator itself or further off-line analysis. Chapter 5 presents methods and results of *in vivo* experiments performed on intra-abdominal organs of animals. Chapter 6 explains the techniques used to calibrate the parameters of a specific model of tissue biomechanics, and the parameter values calibrated by this method. In Chapter 7, we present a surgical simulator equipped with an abdominal mannequin and instrumented force feedback devices for the simulation of esophageal procedures including pulling and cutting of the esophagus. The technical components, which are not covered by the thesis but are necessary to build the simulator, are briefly explained. Summarizing the conclusions and future suggestions are discussed in Chapter 8. It revisits the important elements of the proceeding chapters and discusses further directions in the research.

Chapter 2 Background

This chapter begins with a brief introduction to minimally invasive surgical (MIS) procedures, which are currently the major targets for VR-based simulation. The second part of the chapter will introduce the Visible Human Datasets by the National Library of Medicine (NLM). These datasets have provided the input images necessary to build three dimensional human anatomical models.

The last part of the chapter will review the haptic modality in virtual environments. Haptic feedback has been considered an important aspect of surgical simulation because it provides higher fidelity in a training environment. There are three major components of haptics in virtual environments [2], namely, human haptics (user), machine haptics (hardware) and computer haptics (software). We will focus on the second and third of these issues. Additionally, effectiveness of haptics for surgical training will be discussed in the following section.

This chapter concludes by describing the scope of the research covered by this thesis, and briefly reviews areas outside of this scope.

2.1 Minimally Invasive Surgical Procedures

Minimally invasive surgery (MIS) has had a profound impact on surgical practice due to its significant advantages over traditional open surgeries. In traditional open surgical procedures, surgeons can directly touch or see the structures on which they operate through large openings in the abdomen. Such large openings leave patients with unnecessarily large scars, and long and expensive hospital stays. Advances in video imaging and instrumentation have made it possible to perform many surgical procedures without making

large openings in the abdomen. This kind of procedure is generally called minimally invasive surgery (MIS). The media has referred to MIS by such evocative monikers as "keyhole" and "Nintendo" surgery. Minimally invasive surgery is becoming widespread due to its significant advantages over traditional open surgeries. A major advantage of MIS is that patients recover more quickly, dramatically shortening the duration of stay in the hospital. Reduced discomfort and improved cosmetics are also benefits from MIS procedures. According to the American Medical Association, 85% of gallbladder removal procedures and 95% of Nissen fundoplication procedures were performed by a minimally invasive approach in the United States [3].

In spite of these advantages, surgeons are still handicapped by the limitations of the current technology. They face four main types of problems in the operating room [4].

- The visualization of the internal organs is achieved by a wide-angle camera, but the vision is still monoscopic and is limited by the field of view (FOV) of camera.
- Hand-eye coordination is unnatural since the fulcrum action of the surgical port reverses motion of hands relative to the instrument tips.
- The haptic (tactile sensing and force feedback) cues to the surgeon are substantially reduced since he/she has to interact with internal organs by means of surgical instruments attached to a long slender tube.
- The instruments rotate about a fixed point (i.e. the port fixed in the abdomen wall) and it is not possible to make direct translational movements while interacting with organs.

These problems mean MIS surgeons must climb a relatively long learning curve with considerable repetitive practice required to reach a necessary skill level (Figure 2-1). Traditionally, surgical training has been performed on rubber models, cadavers, animals, or human patients. The animal model is generally preferred for its realism, but is expensive since training requires time in an operating room, trains only one surgeon, and incurs the cost of animal care and housing. In addition to these drawbacks, objective assessment of the trainee's performance cannot be performed.

These inherent difficulties of current surgical training methods and improvements in virtual reality technology have motivated the development of VR-based surgical

simulation. Simulated applications in MIS are increasing rapidly including arthroscopy [5], laparoscopy [6] [7] [8], Gynecology [9] and Bronchoscopy [10].

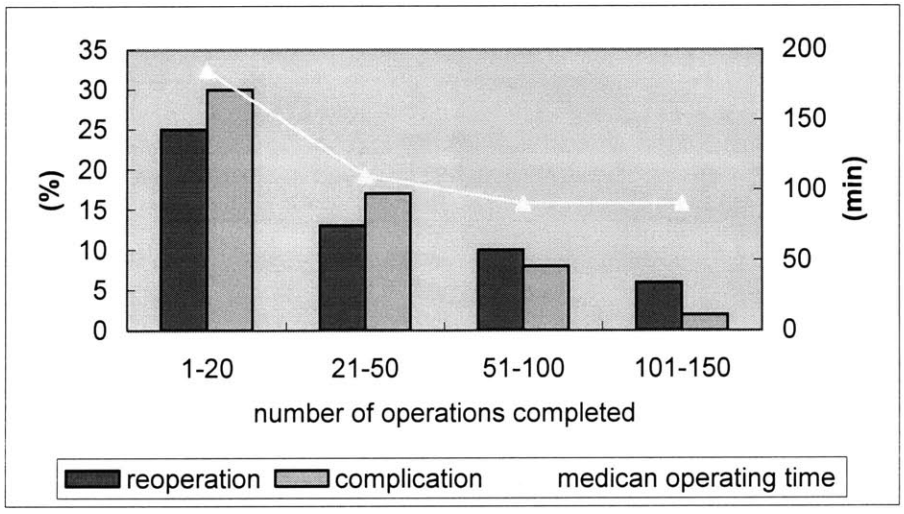

Figure 2-1 "Learning curve" in surgical training of esophageal procedures. Experience can significantly reduce the complications and operation time. This is the clinical data from Australian hospitals [11]; Total 11 surgeons performed over 280 esophageal procedures. The complication rate, reoperation rate and medical operation time for this operation were significantly reduced over the number of operations.

2.2 Visible Human Datasets

The creation of the Visible Human Dataset (VHD) by the National Library of Medicine (NLM) has allowed the area of surgical simulation to advance rapidly in the last ten years. They created two sets of data completely describing one male and one female cadaver in terms of MRI, CT and anatomical cross sections illuminated with visible light. These data sets, in JPEG format, are available (see Figure 2-2) via a public FTP site (nlmpubs.nlm.nih.gov). These complete, anatomically detailed, three-dimensional representations of the human bodies at a one millimeter interval (Table 2-1) were developed to provide a common reference point for the study of human anatomy, to test

medical imaging algorithms, and to test code designed to index and present large image libraries through computer networks.

Since the VHDs are just sets of two dimensional images, the segmentation and reconstruction of anatomical structures is necessary to build three dimensional models with good resolution for the real time processing. Segmentation has been performed using the marching cube technique developed by Lorensen [12] in order to produce triangulated surface models of individual organs. A typical model (Figure 2-3) is comprised of polygons with a very high spatial density, and requires post-processing in order to reduce its complexity to something computationally tractable for real time simulation. The detail procedures are well documented in their website (www.nlm.nih.gov).

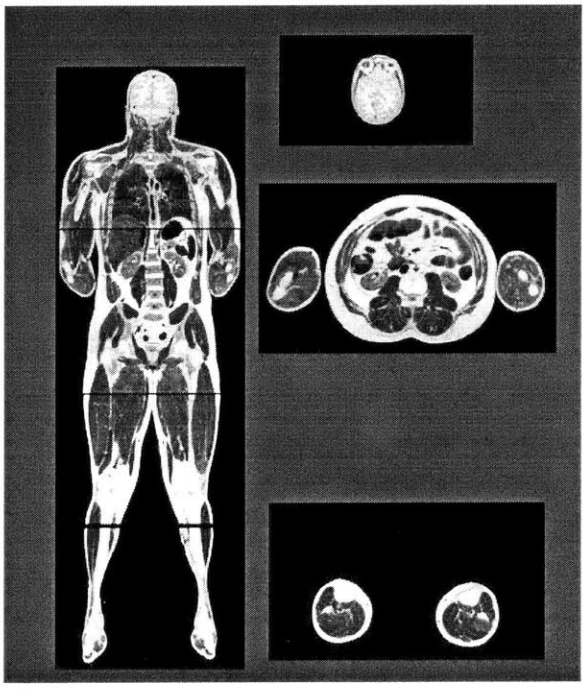

Figure 2-2 Selected images from the Visible Human Male Dataset created by NLM.

Table 2-1 Fact sheet of the Visible Male and Visible Female Datasets

		Visible Human Male	Visible Human Female
Released Year		1994, 2000	1995
MRI Image	Resolution	4 mm	4 mm
	Pixel/image	256 p x 256p	256 p x 256p
	Format	12 bit grey	12 bit grey
CT Image	Resolution	1 mm	1 mm
	Pixel/image	512p x 512p	512p x 512p
	Format	12 bit	12 bit
Anatomical Images (No. of slices)		1 mm (1871)	0.3mm(5189)
		2048p x 1216p	2048p x 1216p
		24 bit	24 bit
Data Size		15 Gbyte	40 Gbyte

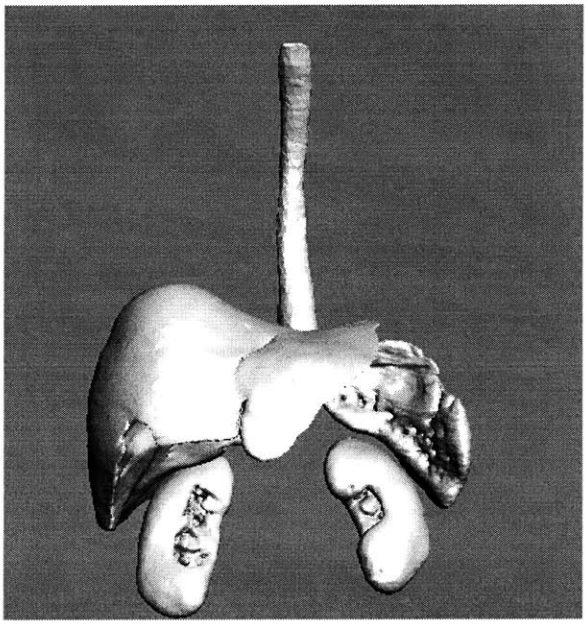

Figure 2-3 Segmented and graphically rendered organs from the Visible Human Datasets by the Visible Human Production. This scene includes triangular models of the esophagus, liver, kidneys and spleen. They are rendered by 3D StudioMax.

2.3 Haptics

2.3.1 Haptic rendering (Computer Haptics)

Haptic rendering refers to a process that compares the kinematic information of a virtual tool tip with geometrical data of simulated objects and computes a force to send back to the user [13]. It also displays various kinds of surface properties such as friction and texture of virtual objects.

During the simulation, the user manipulates the instrument and feels reaction forces when the tool collides with the simulated organs. Although there are many varieties of surgical instruments in reality, they can be modeled as a point, a line, an object or combination of these three entities in virtual environments. It is generally accepted that for haptic interactions to seem natural, the data update rate should be at least a few hundred Hz. This means that computational time for each haptic loop should take about a millisecond or less. Because anatomical models have complex surfaces as well as material properties, the development of a fast haptic rendering technique for anatomical becomes an important topic in medical simulations.

2.3.2 Haptic device (Machine Haptics)

The typical functions of a force-feedback haptic interface are to sense the position of the user's hand and reflect forces back to the user. In the past few years, different types of haptic interfaces have been developed for different purposes. As a wide variety of devices are used or under development, we will limit our review only to pure force feedback devices. Depending on how the interfaces are set up, they can be categorized as ground-based or body-based. In general, the ground-based interface rests on a floor or tabletop, whereas the body-based interface is a mobile device worn directly on human hands. Currently available ground-based displays have high enough spatial resolution (typically 25 micron) for surgical simulation, and are ideal for simulating surgical tools, since they generally interface with the user via a small stylus. Current body-based haptic displays lack the fidelity required for surgical simulation.

The devices shown in Figure 2-4 and Figure 2-5 are force feedback haptic devices widely used in surgical simulation. They evolved from a robotic mechanism so they generate and transmit forces to a user by motor at specific haptic update rates (500-1 KHz). Reviews of haptic interfaces can be found in Burdea, 1996 [14] and Biggs and Srinivasan 2001 [15].

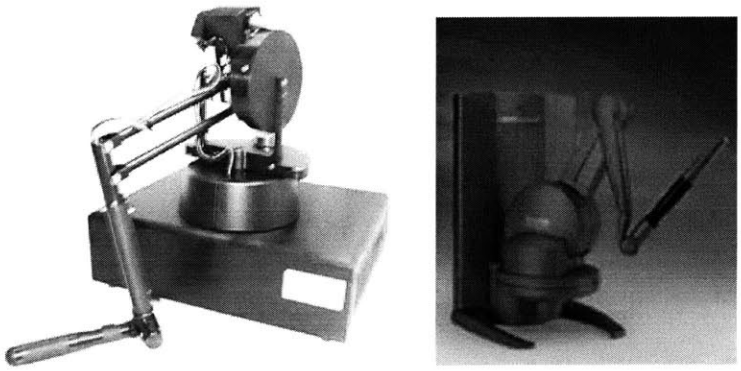

Figure 2-4 PHANToM by SensAble Technology, Inc.

Figure 2-5 Virtual Laparoscopic Impulse Engine by Immersion Cooperation

2.3.3 Is haptics is useful for training?

With the addition of a haptic device, simulation becomes even more challenging since a haptic loop requires a much higher update rate than a visual loop. Furthermore, adding haptics dramatically increases the cost and complexity of a simulator. The previous work [16] suggests that improved training may be worth the added complexity. They conducted experiments measuring training effectiveness with haptics and without. A dual station experimental platform was built to investigate how much haptic fidelity was required in order to achieve a given level of training effectiveness. Analogous laparoscopic surgical tasks were implemented in a virtual and a real station, with the virtual station modeling the real environment to various degrees of fidelity. After measuring subjects' initial performance in the real station, different groups of subjects were trained on the virtual station under a variety of conditions and tested finally at the real station. The results showed that haptics enhanced the training considerably and led to a more consistent training effect. Also, training with haptics seemed to become more important for surgical tasks that require more accuracy and delicate tool control. Therefore, the results seem to support the idea that an effective surgical simulation is one that provides both graphic and haptic feedback.

2.4 Scope of the Thesis

Figure 2-6 shows the technical components in VR-based surgical simulation or simulation of tool tissue interactions. The two major research areas are real time rendering and the modeling of surgical environments. The real time rendering area consists of two major tasks: collision detection and collision response. Collision detection determines the occurrence of contact between the tools and the organ geometry, while collision response (tissue model) computes proper response of models such as deformation fields and interaction forces. Both parts are highly dependant on modeling of organs and surgical tools. The modeling part includes 1) the creation of a fully three-dimensional representation of a segmented organ or body region. 2) the photorealistic textures of

anatomical objects 3) the modeling of surgical instruments and 4) the *in vivo* material properties of the soft tissues of interest.

The research in this thesis focuses on two fundamental fields of tissue modeling. This thesis will cover the characterization of the *in vivo* material properties of soft tissues including animal testing, and the development of collision response generating haptic and visual responses. The following paragraph briefly summarizes the other parts, which already have been investigated deeper than the topics we will cover in this thesis.

Other aspects of the problem of surgical simulation, beyond the scope of this thesis, are reviewed elsewhere. Collision detection has been studied in computer graphics and Computer Aided Design areas. Collision detection methods for haptic rendering have been studied intensively by Ho and Srinivasan [13]. The first two components in the modeling are readily available from the VHD and medical videos such as laparoscopy. The instrument model specifies a surgical task which includes the graphic description of the shape and the type of tool-tissue interactions. Generally, the instruments in virtual environments can be modeled as a point (point-based rendering [17]), a line (ray-based rendering [18] or a 3D object [19]. They provide a sufficient level of modeling of the real tool for surgical simulation.

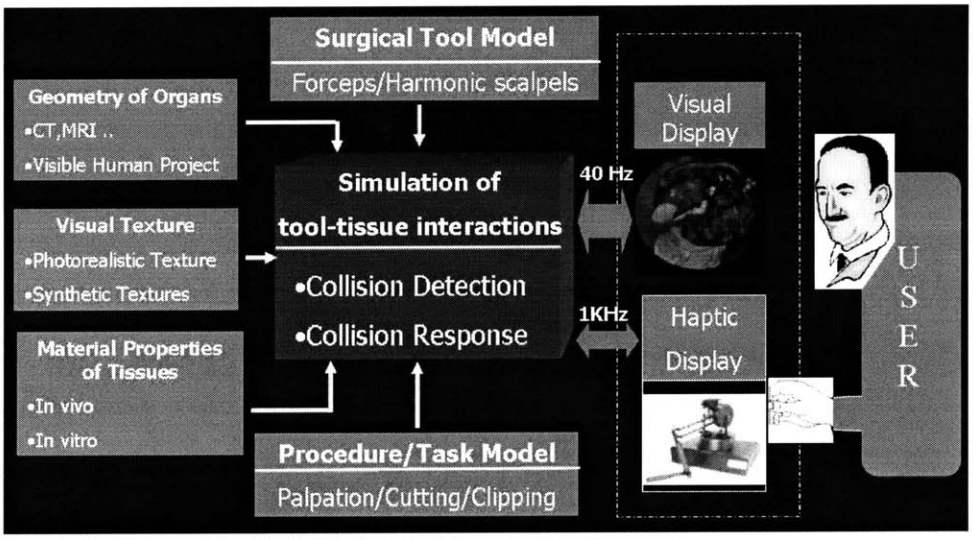

Figure 2-6 Technical components of a VR-based surgical simulator.

Chapter 3 Real Time Simulation of Tool-tissue Interactions

We begin this chapter with a literature review of real time simulation of deformable objects. The current trend of deformable modeling is to apply techniques that consider the underlying physics of deformation. This kind of modeling is called "physically based techniques". We describe one kind of physically based techniques, the point collocation based method of finite spheres (PCMFS). It assumes local deformation around the tool-tissue contact region. Realistic simulations of tool-tissue interactions are presented using these techniques and are compared with solutions using the Finite Element Method (FEM) in terms of speed and accuracy. The integration of the proposed scheme into a surgical simulator is also presented at the end of the chapter.

3.1 Previous Work

The modeling of deformable objects has been widely studied in the computer graphics and computer aided design communities [20-22]. Delp at el. [23] demonstrated simulation of gunshot injuries on the human lower body and their surgical treatment. Basdogan and Srinivasan [24] developed a deformable organ model which computes the reaction forces proportional to the penetration depth of surgical tools and displays visual deformations locally by using spline functions. These techniques are "geometrically based" because they used implicit surfaces like Bezier/B-spline surfaces or free form

deformation surfaces [25] to represent the deformable objects. They are quite rapid and visually appealing but they do not account for the physics of the deformation as embodied in the partial differential equations governing the behavior of soft tissues.

In contrast, the "physically based" approaches consider the underlying physics of deformation in the modeling. Mass–spring models are widely used in computer animation because they are relatively easy to implement. Cover *et al.* [26] developed the first laparoscopic gall balder surgery simulator using surface-based mass–spring models. Kuhnapel and Neisius [27] used a mass-spring model as a part of their KISMET simulation system for endoscopic surgery. Although mass-spring models are relatively computationally inexpensive, it is difficult to extract parameters for the hundreds and thousands of individual springs, masses or dampers from experiments. Additionally, the construction of an optimum three-dimensional network of springs, masses and dampers is a complicated process. It is reported that the geometrical structure and topological arrangement of springs/dampers strongly influence the material behavior and may generate undesired isotropy [28].

Techniques based on continuum mechanics, such as the finite element method, are more appealing due to their proved accuracy. Additionally, only a few material parameters (as few as two for linear isotropic elasticity) are required in the modeling and they can be obtained rather conveniently from experiments. Finite element models have been extensively used in engineering analysis. However, they are very computationally expensive, so research effort has been concentrated on developing techniques of speeding up the finite element method for real time performance. One of these efforts is to use a so-called "waterbed model" proposed by De and Srinivasan [29] which models deformable objects as thin-walled membrane structures filled with fluid. Bro-Nielson developed a simulation system based on three dimensional linear elastic finite element models [30]. Real time performance was achieved by the use of condensation, and precomputation of the system stiffness matrix. Fast finite element models were also used by the INRIA group for hepatic surgery simulation [31]. Wu and Tendick [32] modeled

material nonlinearities in FEM to handle large deformation for 3D deformable bodies. Another well known technique of numerically solving physical problems is the boundary element scheme in which the governing integral equations are discretized on the boundary of the domain using piece-wise polynomial approximations. One of the earliest applications of the boundary element technique to the field of real time simulation of deformable objects is the work of James and Pai [33, 34]. They modeled quasi-static deformations using the boundary element method and achieved real time performance using a simple structural reanalysis technique. However, the approach is limited to linear models which are homogeneous and isotropic.

Mesh based techniques such as the finite element/boundary element methods are not ideally suited for real time surgical simulation. The contact between tool and tissue must occur at a special set of points (nodal points). Therefore, to prevent loss of resolution, the density of nodal points should be sufficiently high leading to an increase in the problem size and consequently computational time (see Figure 3-1). Moreover, during surgical cutting, the element edges have to be oriented along the cut and this requires, at least locally, a constant remeshing operation which is computationally expensive. These mesh-based schemes also require an expensive numerical integration operation for the computation of the system stiffness matrices.

To overcome the complexities of mesh generation and the consequent constraints imposed on the computation, a radically simplified technique known as the method of finite spheres was proposed [35]. In this method, the continuum is represented as a collection of particles or "nodes" which serve as the computational primitives. The particles possess finite "regions of influence" or "fields" which smear out their effects and coordinate their motions during simulation. To meet the requirements of real time performance, a specialized version of this technique, the point collocation-based method of finite spheres (PCMFS) was developed [36]. In the following sections, we present an application of this technique to the simulation of surgical tool-soft tissue interactions in real time.

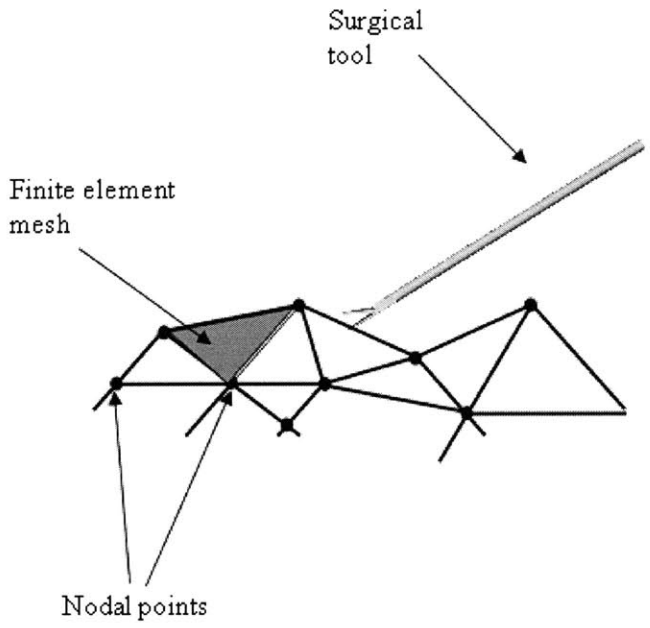

Figure 3-1 In a mesh-based modeling technique the entire organ has to be meshed. One of the problems associated with such a technique is that it is quite difficult to handle the situation in which the surgical tool does not land at a nodal point.

3.2 Mathematical Theory

The PCMFS is a novel numerical technique for the solution of partial differential equations without the use of a computational mesh as in FEM/BEM. In this scheme, computational particles are sprinkled on the domain (see Figure 3-2). At each particle a spherical influence zone is defined. At every particle an approximation function is defined which is nonzero only on the spherical influence region and zero everywhere else on the domain. These functions are then used to solve systems of differential equations on the domain using a technique known as "point collocation".

We are interested in solving the following linear elasticity problem on a domain Ω,

$$\partial_{\varepsilon}^{\mathrm{T}}\boldsymbol{\tau}(\mathbf{u}) + \mathbf{f}^{\mathrm{B}}(\mathbf{x}) = 0 \text{ in } \Omega \tag{3-1}$$

subject to prescribed displacements on portion Γ_u of the boundary

$$\mathbf{u}(\mathbf{x}) = \mathbf{u}^S \qquad \text{on } \Gamma_u \tag{3-2}$$

and forces on portion Γ_f of the boundary

$$\mathbf{N}\boldsymbol{\tau}(\mathbf{u}) = \mathbf{f}^S \qquad \text{on } \Gamma_f \tag{3-3}$$

In these equations, $\mathbf{u(x)}$ and $\boldsymbol{\tau}$ are the displacement and stress vectors, $\mathbf{f}^S$ is the prescribed traction vector on the boundary Γ_f, u^S is the vector of prescribed displacements on the boundary Γ_u (note that the domain boundary $\Gamma = \Gamma_u \cup \Gamma_f$), $\mathbf{f}^B$ is the body force vector (including inertia terms in a dynamic analysis), ∂_ε is a linear gradient operator, and $\mathbf{N}$ is the matrix of direction cosine components (n_x, n_y, n_z) of a unit normal to the domain boundary (positive outwards).

For a three-dimensional analysis the various vectors are as follows:

$$\begin{aligned}
&\mathbf{x} = \begin{bmatrix} x & y & z \end{bmatrix} \\
&\mathbf{u}(\mathbf{x}) = \begin{bmatrix} u(\mathbf{x}) & v(\mathbf{x}) & w(\mathbf{x}) \end{bmatrix} \\
&\boldsymbol{\tau} = \begin{bmatrix} \tau_{xx} & \tau_{yy} & \tau_{zz} & \tau_{xy} & \tau_{yz} & \tau_{zx} \end{bmatrix} \\
&\partial_\varepsilon = \begin{bmatrix} \partial/\partial x & 0 & 0 \\ 0 & \partial/\partial y & 0 \\ 0 & 0 & \partial/\partial z \\ \partial/\partial y & \partial/\partial x & 0 \\ 0 & \partial/\partial z & \partial/\partial y \\ \partial/\partial z & 0 & \partial/\partial x \end{bmatrix} \\
&\mathbf{N} = \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & n_z \\ 0 & n_y & 0 & n_x & n_z & 0 \\ 0 & 0 & n_z & 0 & n_y & n_x \end{bmatrix}
\end{aligned} \tag{3-4}$$

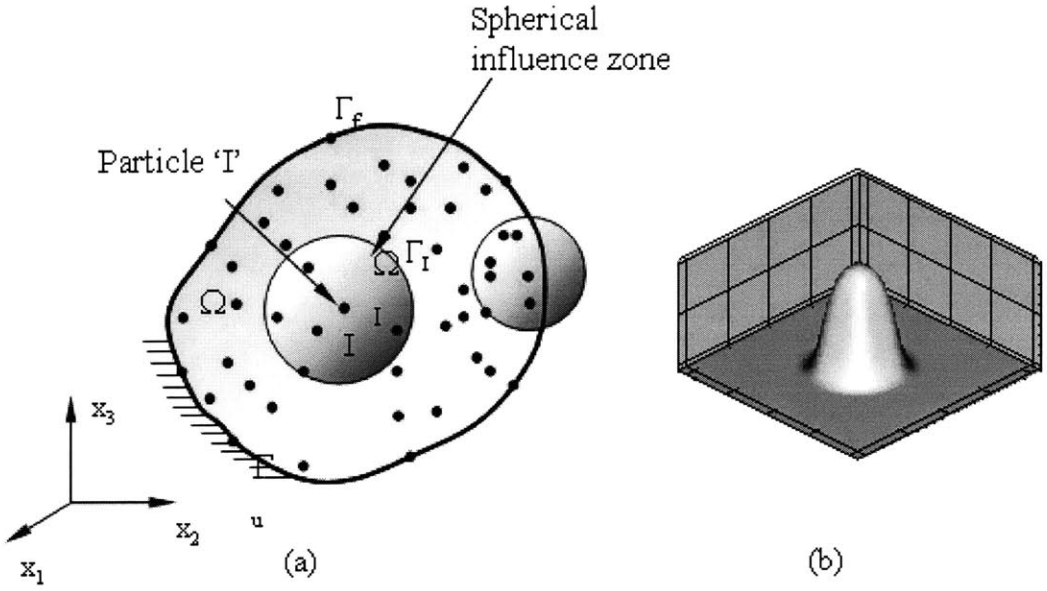

Figure 3-2 (a) A 3D domain is discretized using particles with finite (spherical) influence zones (I is the spherical influence zone at particle I). (b) The interpolation function h_I at particle I is nonzero only on the sphere at particle I.

3.2.1 Point collocation technique

In the point collocation technique the displacement solution **u(x)** is approximated by $\mathbf{u}_h$ and the governing partial differential equations are applied at the nodal points. The discrete set of equations may be written as

$$\left[\partial_\varepsilon^T \boldsymbol{\tau}(\mathbf{u}_h)\right]_{\mathbf{x}=\mathbf{x}_I} + \mathbf{f}^B(\mathbf{x}_I) = 0 \text{ in } \Omega \tag{3-5}$$

subject to displacement boundary conditions

$$\mathbf{u}_h(\mathbf{x}_I) = \mathbf{u}^S \qquad \text{on } \Gamma_u \tag{3-6}$$

and force boundary conditions

$$\left[\mathbf{N}\boldsymbol{\tau}(\mathbf{u}_h)\right]_{\mathbf{x}=\mathbf{x}_I} = \mathbf{f}^S(\mathbf{x}_I) \qquad \text{on } \Gamma_f \tag{3-7}$$

where $\mathbf{x}_I$ is the position vector of node I.

For linear stress-strain relationship this results in a set of linear algebraic equations. For nonlinear differential equations, a Newton-Raphson scheme is used [37]. For problems involving dynamics, an important criterion is the choice of stable time integration schemes satisfying the CFL condition [37]. Note that the point collocation approach avoids the computationally burdensome numerical integration required in mesh-based schemes such as the finite element methods.

3.2.2 Approximation functions

The approximation $\mathbf{u}_h$ of the variable $\mathbf{u}$, using 'N' particles, may be written as

$$\mathbf{u}_h(\mathbf{x}) = \sum_{J=1}^{N} \mathbf{H}_J(\mathbf{x})\alpha_J = \mathbf{H}(\mathbf{x})\mathbf{U} \tag{3-8}$$

where $\mathbf{U} = [\alpha_1 \alpha_2 \alpha_3 ...]^T$ is the vector of nodal unknowns and $\alpha_J = \lfloor u^J \; v^J \; w^J \rfloor$ is the vector of nodal unknowns at node J.

The nodal shape function matrix for node J is

$$\mathbf{H}_J(\mathbf{x}) = \begin{bmatrix} h_J(\mathbf{x}) & 0 & 0 \\ 0 & h_J(\mathbf{x}) & 0 \\ 0 & 0 & h_J(\mathbf{x}) \end{bmatrix}. \tag{3-9}$$

where

$$h_J(\mathbf{x}) = W_J(\mathbf{x})\mathbf{P}(\mathbf{x})^T \mathbf{A}^{-1}(\mathbf{x})\mathbf{P}(\mathbf{x}_J) \qquad J=1,\ldots,N$$

with

$$\mathbf{A}(\mathbf{x}) = \sum_{I=1}^{N} W_I(\mathbf{x})\mathbf{P}(\mathbf{x}_I)\mathbf{P}(\mathbf{x}_I)^T \tag{3-10}$$

The vector $\mathbf{P}(\mathbf{x})$ contains polynomials ensuring accuracy up to a desired order (if we choose $\mathbf{P}(\mathbf{x}) = \{1,x,y,z\}^T$ for example, a first order accurate scheme is ensured in 3D,

similar to bilinear finite elements). $W_J(\mathbf{x})$ is a radial weighting function at particle 'J' which is nonzero only on the sphere at particle J (e.g., quartic spline).

With this approximation, the deformation equations can be rewritten in a matrix form

$$\mathbf{KU} = \mathbf{f} \tag{3-11}$$

where **K** is the stiffness matrix (nonsymmetrical but banded) and **f** is a known vector containing nodal loads.

3.3 Implementation of PCMFS

A key assumption in the modeling is that the surgical tool interacts ***locally*** with the organ. Especially if the size of the organ is large compared to the tool tip, it may be assumed that the deformation zone is localized within a "region of influence" of the surgical tool tip. Therefore, it may be sufficient to sprinkle the computational particles in the vicinity of the tool-tip and to perform local computations using the PCMFS. This technique results in a dramatic reduction in the solution time for massively complex organ geometries.

The localization results in zero displacements being assumed on the periphery of the "region of influence" of the surgical tool-tip, which is chosen after careful observation of videos of actual surgical procedures. The stiffness matrix, **K**, in equation **(3-11)** may be partitioned as

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{aa} & \mathbf{K}_{ab} \\ \mathbf{K}_{ba} & \mathbf{K}_{bb} \end{bmatrix} \tag{3-12}$$

corresponding to a partitioning of the vector of nodal parameters as $\mathbf{U} = [\mathbf{U}_{\text{tooltip}}\ \mathbf{U}_{\text{b}}]^{\text{T}}$ where $\mathbf{U}_{\text{tooltip}}$ is the applied displacement vector at the surgical tool tip and $\mathbf{U}_{\text{b}}$ is the vector of unknown displacements which can be obtained from

$$\mathbf{U}_{\text{b}} = -\mathbf{K}_{\text{bb}}^{-1}\mathbf{K}_{ba}\mathbf{U}_{\text{tooltip}} \tag{3-13}$$

The force vector delivered to the haptic device is computed as

$$\mathbf{f}_{\text{tooltip}} = \mathbf{K}_{\text{aa}}\mathbf{U}_{\text{tooltip}} + \mathbf{K}_{\text{ab}}\mathbf{U}_{\text{b}} \tag{3-14}$$

Since $\mathbf{K}$ is built locally, the size of the matrix is quite small. A point to note is that even though linear elastic behavior is assumed, the technique is very general (no precomputations are required) and a consistent linearization scheme may be adopted for iterative solution of problems involving nonlinear constitutive behavior. The most computationally expensive part is to calculate $\mathbf{K}_{\text{bb}}^{-1}$ using matrix inversion algorithms. We used an iterative matrix inversion technique named the Generalized Minimal Residual Algorithm (GMRES) [38] instead of using direct factorization. It is possible to reduce the solution time to order n^2 while the direct inverse technique requires order n^3 where n is a problem size.

We have implemented this modeling scheme for the solution of the problem of indenting a hemisphere (10 cm diameter) at the pole and compared the results with a FEM solution using a commercial software package, ADINA(www.adina.com). In Figure 3-3, we compare the displacement solution obtained for the hemisphere problem using the PCMFS scheme (N = 34 particles) with the FEM (4045 nodal points). The displacement profile computed using PCMFS is observed to be quite accurate in the vicinity of the tool-tissue interaction point. However, not surprisingly, the error increases

with distance from the tool tip. It is to be noted that with 34 nodes, it is not possible to generate more than two finite elements in 3D.

Table 3-1 Comparison of computational time for a FEM model and a localized PCMFS model of a hemisphere being indented at the pole. A 500 MHz P II machine was used for the simulations.

	FE model (ADINA)	The PCMFS model
Number of Nodal points(n)	4045	34
Degrees of freedom (3n)	12135	102
Solution time (sec)	11.88	0.0015
Total simulation time(sec)	18.32	0.0036

Table 3-1 summarizes the result of a comparison of solution times. The total time is assumed to be composed of the time to generate the stiffness matrix and time to solve the system of equations. While the FEM solution takes more than 10 seconds, the PCMFS solution takes only a few milliseconds. This is, of course, not a very fair comparison since the entire volume of the hemisphere is meshed using volumetric finite elements while only a few PCMFS particles are sprinkled around the tool tip. However, as we observed in the last paragraph, the solution accuracy of the two techniques is quite comparable, at least in the vicinity of the tool tip.

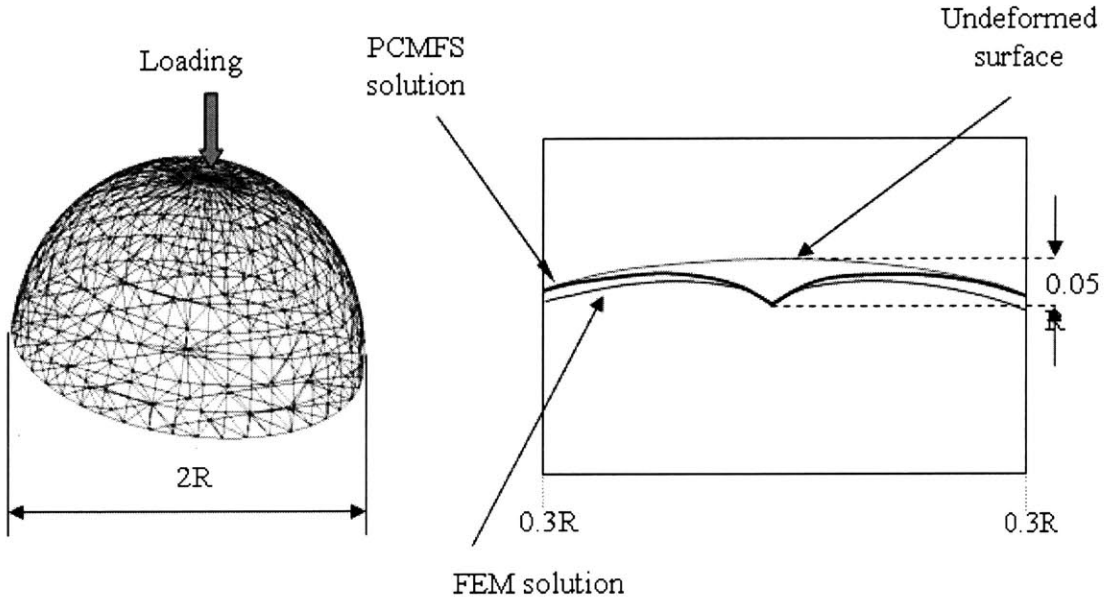

Figure 3-3 Deformation field computed when the PCMFS technique is used for the simulation of a surgical tool tip interacting with a hemispherical object is shown. The undeformed surface as well as the deformation field obtained when a finite element software package (ADINA) is used to solve the problem is also shown.

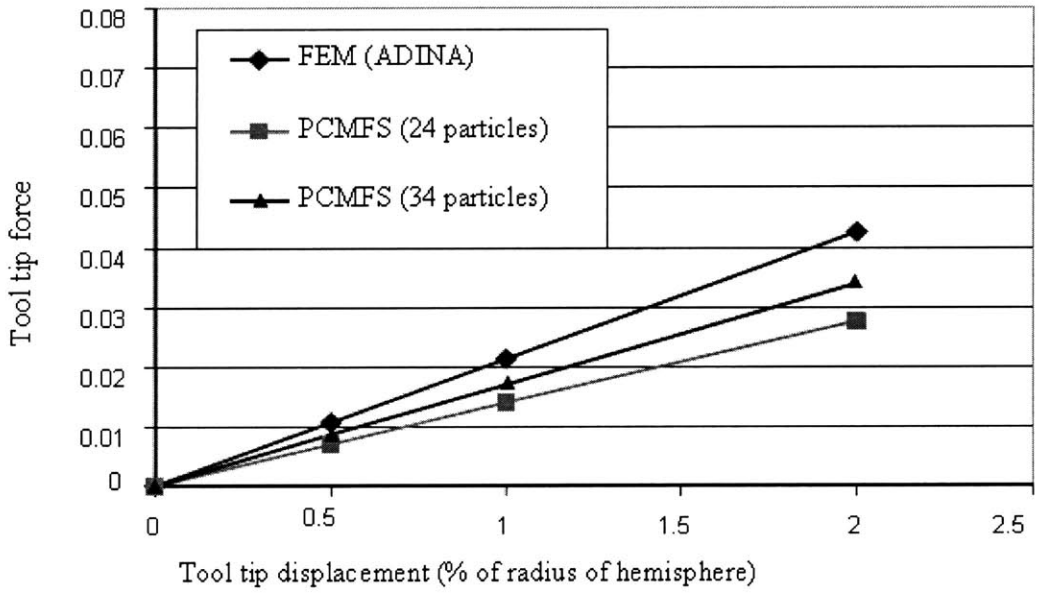

Figure 3-4 Forces computed at the tool tip in the indentation problem (see Figure 3-3).

In Figure 3-4, we compare the reaction forces at the tool tip computed using both the techniques. Since only part of the domain is modeled using PCMFS, the computed force is less than that obtained using the finite element method. However, with increase in the number of points used for discretization there is significant improvement in the computed reaction forces. Since the PCMFS provide the comparable interaction forces and deformation for real time display with less computational costs, we can use the modeling scheme to the development of tissue model for real time simulation without precomputation as in FEM based techniques, which are highly limit the flexibility of tissue models.

3.4 Examples

In this section we describe a typical surgical simulation scenario. Four major computational tasks need to be performed in real time,

1. The detection of the collision of the tool tip with the organ model
2. The sprinkling of the PCMFS particles
3. The computation of the displacement field
4. The computation of the interaction force at the tool tip

A PCMFS particle is placed at the collision point. The other particles are placed by joining the centroid of the triangle with the vertices and projecting on to the surface of the model using the surface normal of the triangle in a manner shown in Figure 3-6. The locations of the PCMFS particles corresponding to a collision with each and every triangle in the model are pre-computed and stored and may be retrieved quickly during simulation.

The haptic update rate is dependent on a specific haptic device. We used a PHANToM from Sensible Tech. in this project and the device had a 1 KHz update

frequency. Since the underlying tissue model had still slower responses than the update rate, the force extrapolation scheme and the haptic buffer were implemented in order to achieve the required update rate (see Figure 3-5). The model thread runs at 200 Hz to compute the interaction forces and send them to the haptic buffer. The haptic thread extrapolates the compute forces to 1 KHz and displays them through the haptic device. The special data structure like Semaphore is necessary to prevent the crash of the variables during a multithreading operation.

We present the simulation of the palpation of a liver as an example (see Figure 3-7). We use a three-dimensional liver polygonal liver model consisting of 20000 polygons. Since this is a large model compared to the tool tip dimensions, we employed the assumption of localized deformations and used total 34 PCMFS particles around the surgical tool tip. If the number of the particles is reduced below 34, the computed solution lost accuracy while too many particles lead the excess computational overhead into the simulator.

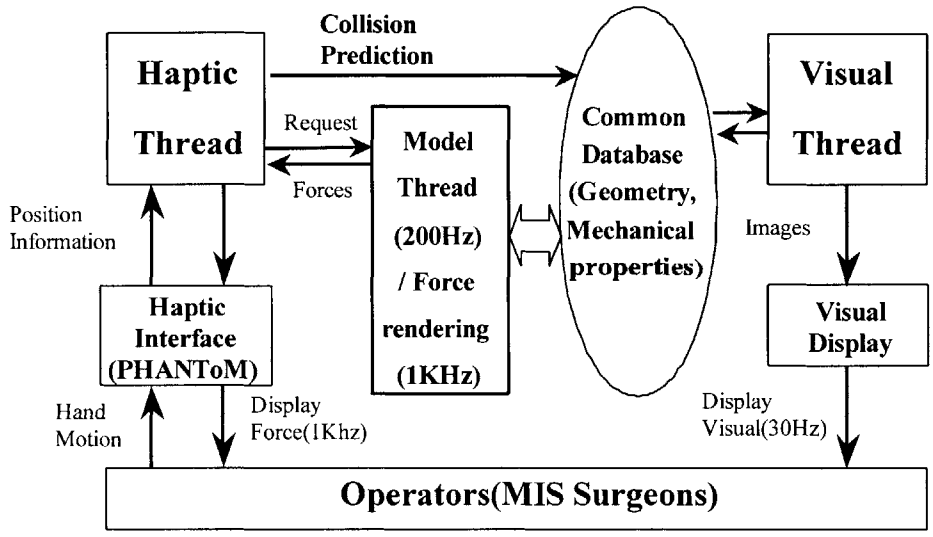

Figure 3-5 Software architecture of PCMFS implementation for surgical simulation

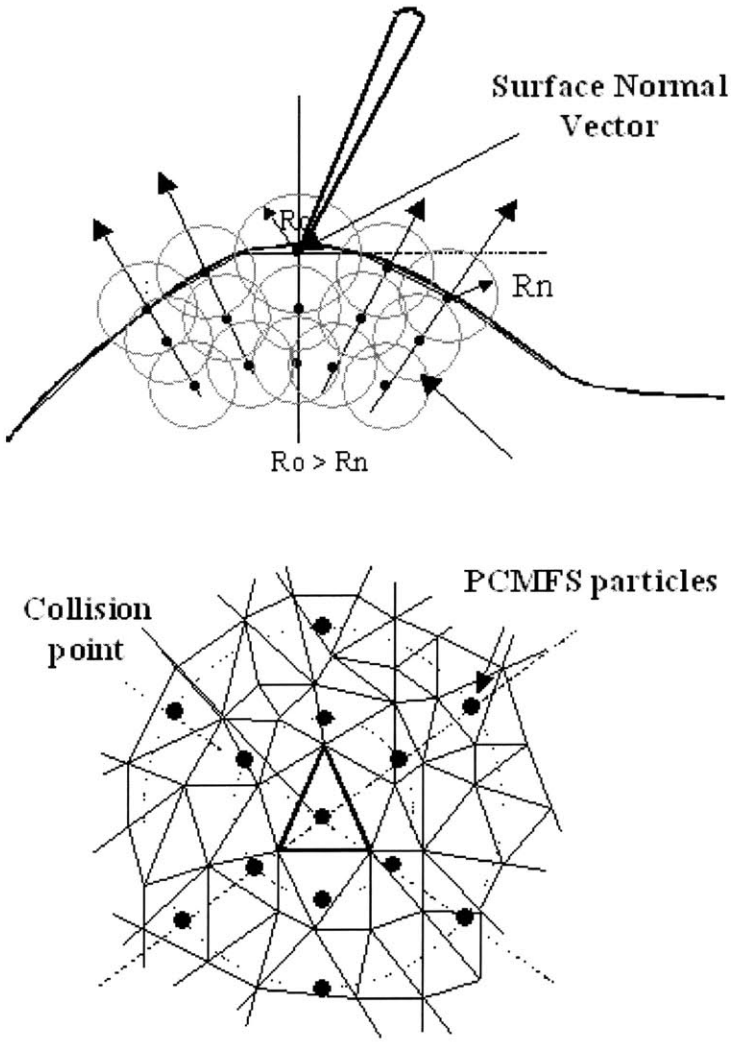

Figure 3-6 Placement of the PCMFS nodes once a collision point has been determined. In the top panel, we show a section through the polygonal surface model of an organ. The bottom shows a part of the top view.

We have proposed a novel real time modeling technique, the point-collocation-based method of finite spheres (PCMFS) for the simulation of tool-tissue interactions. For very complex tissue geometries we have proposed the use of a localized version of this technique assuming that the deformations die off rapidly with increasing in distance

from the surgical tool tip. The major advantage of this technique is that it is not limited to linear tissue behavior and real time performance may be obtained without using any precomputations.

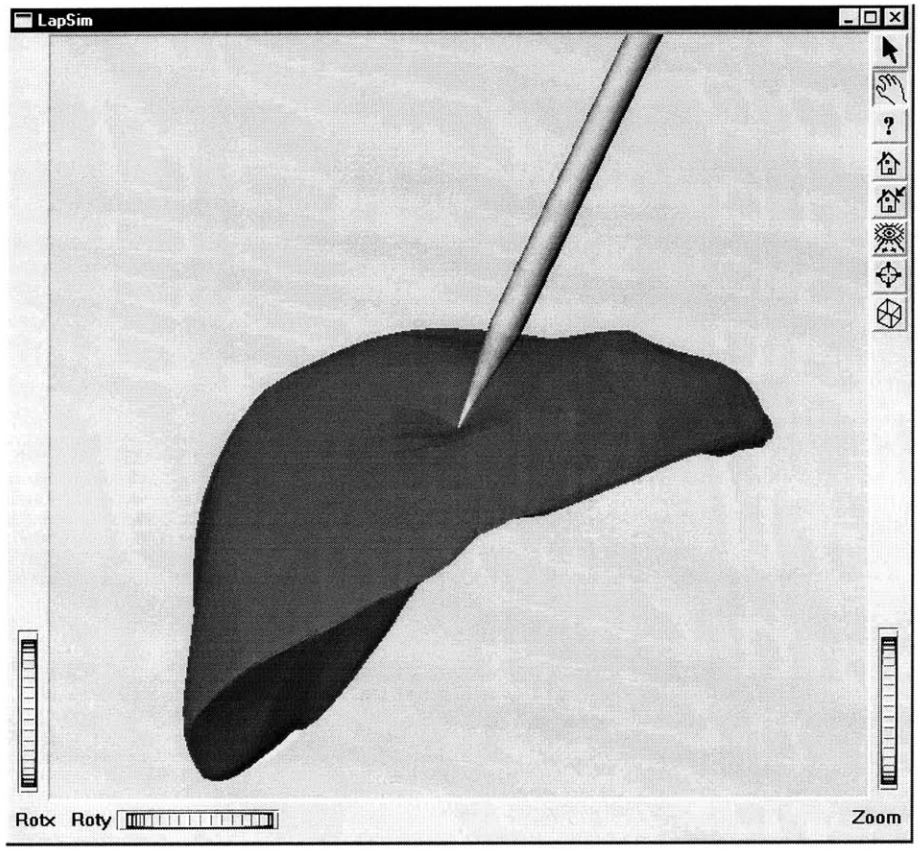

Figure 3-7 Snapshot of a liver palpation task. The deformation field, as well as tool tip reaction force is computed in real time using the PCMFS technique.

Chapter 4 Development of a Physically - based Hybrid Organ Model

In this chapter we propose a numerically efficient physically based tissue modeling scheme by introducing a hybrid modeling approach, where a reasonably coarse global model is locally enhanced with a local model with a finer resolution. The global model is based on a discretization of the boundary integral representation of three-dimensional deformable objects. The use of precomputation and structural reanalysis techniques result in a very rapid computation procedure. Local refinements are provided in the vicinity of the tool-tissue contact region by a local subdivision technique or a local tissue model that considers underlying the mechanics of a local area. This technique results in interactive visual as well as haptic rendering rates for reasonably complex models.

4.1 Motivations

There is a fundamental trade-off between fidelity and interactivity in real time simulation. The higher fidelity requires a higher resolution model and it leads to latency in real time computation. One of the examples is that the solution time for computing deformation from the tissue model increases as a cubic or at least square order of the problem size even for a linear elastic model. The typical anatomical model created from the VHD has usually more than 100,000 polygons for a single organ. Furthermore, the simulator should render

the visual and haptic scene of deformed organs within the real time limits of each modality. Therefore the techniques developed so far have focused on how to circumvent the computational formidability without scarifying fidelity of the simulation.

When we develop the deformable model simulating tool-tissue interactions, we consider a few aspects in surgical simulation. First, assigning a single, high resolution over an entire object is not an efficient solution due to limited computing resources and time. Fortunately, it is unnecessary to provide a higher resolution over an entire scene during simulation. The only part requiring higher resolution is the vicinity of the tool-tissue contact region experiencing large deformation and translation. Second, the visual rendering of the organ does not use information internal to the object. Finally, we can use a certain level of precomputation to speed up real time computation.

From these observations and design criteria, we present a hybrid approach for simulating tool-tissue interactions in surgery simulation to balance computational speed and accuracy of computations as shown in Figure 4-1. The justification of using such a hybrid approach is the following. When a tool-tissue interaction occurs, the "scene of action" is restricted to only the zone in the vicinity of the tool tip. Therefore, a high-resolution model is required for this region and this high-resolution model is governed by user's choice of tool-tissue contact location. The rest of the domain can be modeled using a relatively coarse model.

There is some related work in hybrid modeling with different terminologies. Astley and Hayward proposed a primitive multirate simulation technique for haptic interactions [39] using Norton equivalents. Cavusoglu and Tendick developed a multirate simulation technique in the context of the finite element technique [40]. Zhang at el. proposed a level of detail modeling technique with haptic subdivision [41]. Several researchers have implemented wavelet transform techniques for multiresolution modeling [34, 42]. Wavelets offer a natural multiresolution modeling approach, but are not yet computationally efficient for applying in deformation modeling.

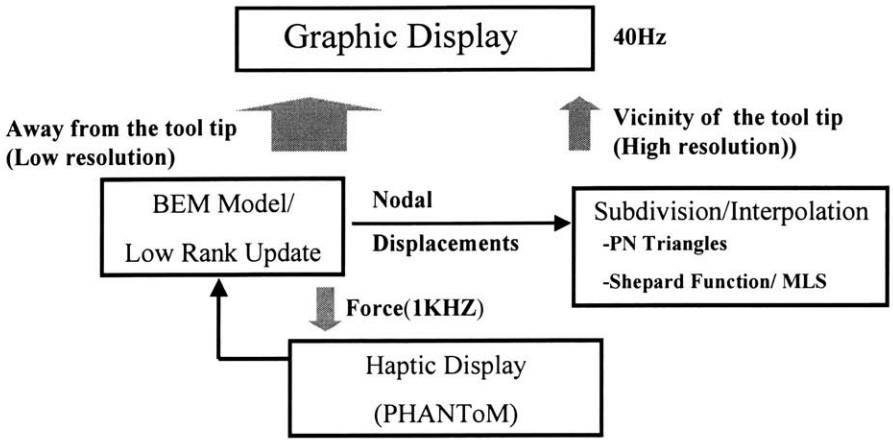

Figure 4-1 Concept of the hybrid modeling in simulation of deformation

4.2 Global Deformation Model

The global model computes the overall deformation of the organ model including rigid body motion for our global model. We selected the boundary element technique. BEM is based on the discretization of the integral equations of motion posed on the surface of the model and has significant advantages over volumetric techniques such as the finite element method (FEM) for the modeling and analysis of linear elastostatic problems [43].

Anatomical organs are usually available as surface models on which texture mapping is used to render realistic graphics. Voxel-models are also available, but they require much higher computational effort and special graphics hardware. Moreover, the boundary element technique reduces the dimensionality of the problem by one and generates a dense, but smaller set of equations compared to the finite element technique. Using accelerated iterative solvers like the GMRES [38], the boundary element equations may be solved quite efficiently. However, we use a technique similar to the technique in [34] and precompute the inverse of the stiffness matrix for a predefined set of boundary conditions and use rapid update techniques for real time computations. In the following paragraphs we will discuss the details of the formulation.

The displacement ($\mathbf{u}$) and traction ($\mathbf{p}$) vectors at a point $\mathbf{x}$ in the domain or on the boundary may be represented by their three Cartesian components

$$\begin{aligned} \mathbf{u} &= \mathbf{u}(\mathbf{x}) = (u_x, u_y, u_z)^T \\ \mathbf{p} &= \mathbf{p}(\mathbf{x}) = (p_x, p_y, p_z)^T \end{aligned} \tag{4.1}$$

Using piece-wise constant elements (i.e., the displacements and tractions are assumed to be constant over each element), the BEM equations of linear elastostatics with 'n' elements is given by [44]

$$c \quad \mathbf{u}_i + \sum_{j=1}^{n} \left(\int_{\Delta_j} p^* \Phi^T d\Gamma \right) \mathbf{u}_j = \sum_{j=1}^{n} \left(\int_{\Delta_j} u^* \Phi^T d\Gamma \right) \mathbf{p}_j \tag{4.2}$$

where Δ_i is the surface of the i^{th} element, u^* and p^* are the 'fundamental solutions' [44]. The coefficient 'c' depends on the smoothness of the boundaries and can be found in the literature (for a smooth boundary, $c = 0.5$).

The fundamental solutions for a three-dimensional isotropy body are,

$$\begin{aligned} \mathbf{u}^*_{ik} &= \frac{1}{8\pi E r}[(3-4\nu)\delta_{lk} + \frac{\partial r}{\partial x_l}\frac{\partial r}{\partial x_k}] \\ p^*_{lk} &= \frac{1}{8\pi(1-\nu)r^2}[\frac{\partial r}{\partial n}\{(1-2\nu)\delta_{lk} + 3\frac{\partial r}{\partial x_l}\frac{\partial r}{\partial x_k}\} - (1-2\nu)\delta_{lk}\{\frac{\partial r}{\partial x_l}n_k - \frac{\partial r}{\partial x_k}n_l\}] \end{aligned} \tag{4.3}$$

where n is a normal to the surface, δ_{lk} is the Kronecker delta, r is the distance from the point of load to the point under consideration, E is a Young's modulus, ν is the Poisson's ratio and n_{lk} are the direction cosines.

Satisfying this equation at the center of each element and incorporating the boundary conditions, we obtain the following system of linear algebraic equations:

$$\mathbf{AY} = \mathbf{F} \tag{4.4}$$

where **Y** is a vector of length N (N=3n) and contains the unknown deformations and tractions at the centroids of the boundary elements. **A** is an N x N dense matrix. **F** is the known right hand side vector containing boundary conditions and depth of tool indentation. The solution of this system is symbolically represented as,

$$\mathbf{Y} = \mathbf{A}^{-1}\mathbf{F} \tag{4.5}$$

Up to this point, two computationally expensive steps are involved. The first one is to build the system matrix A from geometric and material properties of the object. The second is to compute the inverse of **A.** (See Table 4-1 for the time for building the system matrix and the time for solving the system equations for three different models). For example, a model with about 5,000 polygons, which is not unusual in the simulation, takes several days to solve the above equation. Therefore it is expedient to precompute these in a look-up table for use in simulation. If $\mathbf{A}_0$ is the matrix corresponding to a set of predefined boundary conditions, then, corresponding to a new set of boundary conditions, the matrix **A** may be expressed as a low-rank update of $\mathbf{A}_0$

$$\mathbf{A} = \mathbf{A}_0 + \mathbf{U}\mathbf{V}^T \tag{4.6}$$

where **U** and **V** are matrices defined exactly as in [33]. The matrix $\mathbf{A}_0$ and its inverse are precomputed and stored.

The inverse of **A** is given by the Sherman-Morrison-Woodbury(S-M-W) formula [45] as follows.

$$\mathbf{A}^{-1} = \mathbf{A}_0^{-1} - \mathbf{A}_0^{-1}\mathbf{U}\mathbf{C}^{-1}\mathbf{V}^T\mathbf{A}_0^{-1} \tag{4.7}$$

where $\mathbf{C} = \mathbf{I} + \mathbf{V}^T\mathbf{A}_0^{-1}\mathbf{U} \in \mathbf{R}^{3s\times 3s}$ is known as the "capacitance matrix" and s is number of nonzero boundary conditions. In case of localized contact, s is small and this ensures a computational complexity that scales linearly with the number of unknowns in the system. When a single point contact is assumed, s would be three and the computing cost for the matrix inversion would be $O(3^3N)$ instead of $O(N^3)$.

The interaction forces for haptic rendering are computed using the following formula,

$$\mathbf{F} = a_{tip}\mathbf{p}_c \tag{4.8}$$

where a_{tip} and $\mathbf{P}_C$ are the effective nodal area of the tool tip and the traction vector at node 'c' where the interaction occurs, respectively. Because the interaction forces are another set of boundary conditions for our model, the traction vector $\mathbf{p}_c$ is an element of the solution $\mathbf{Y}$ and can be computed without extra computational cost. As seen in Table 4-1, the update times using S-M-W formula provides a possible order of computing times for force feedback in the case of decent size models. Table 4-1 summarizes computing costs of off-line (precomputation) and real time computation times for various geometrical models. We selected three objects for numerical experiments: a) a cube, b) a graphic kidney model, and c) a kidney model created from the Visible Human Dataset as shown in Figure 4-2. As we expected, the solution time increases dramatically as the number of polygons increases. However, the update times using the S-M-W formula provide proper computing time for real time application even in the most complicated case (the model from the Visible Human Dataset).

4.3 Local Deformation Model

One of the major difficulties encountered during the display of deformation fields using a model with a single, fixed spatial resolution is that visual artifacts such as flatness appear in

Table 4-1 Modeling, computing and update times for three BEM models displayed in Figure 4-2 (A winNT workstation with dual Pentium III 1GHz is used)

Model	Points/ Triangle	Modeling Time	Solution Time	Update Time
Cube	98/192	0.11s	11s	0.17 msec
Kidney (Graphic model)	346/ 688	1.51s	7m21s	0.54 msec
Kidney (Created from the VHP dataset)	2979/ 5145	1m 32s	208h 26m	4.3 msec

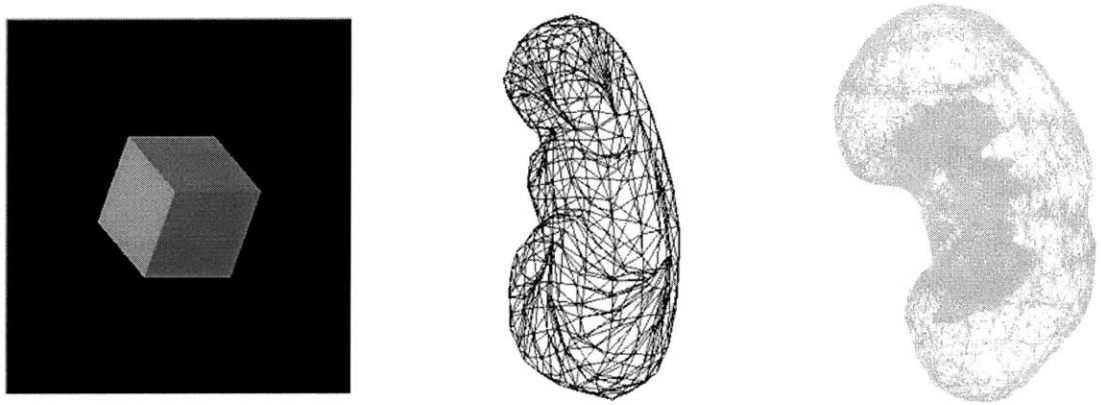

Figure 4-2 Models chosen for numerical experiments a) a cube b) a graphic kidney model c) a kidney model created from the Visible Human Dataset.

the vicinity of a tool–tissue interaction point experiencing large displacements. To enhance or refine this, subdivision algorithms are frequently used in many techniques [46]. These subdivision algorithms, however, increase numerical complexity and require extra memory because they provide unnecessary detail to an entire object. The regions far from the tool-tissue interaction region do not require the high resolution that results from the subdivision

process. To minimize the computational overhead induced by subdivision algorithms in real time, we employed a subdivision algorithm only around the necessary region – around the tool-tissue contact region, while the other areas were kept in the original resolution. We also applied a local interpolation algorithm to enhance the visual quality of the deformation. As illustrated in Figure 4-3, the steps in this procedure are summarized as follows:

1. Detect a collision between a tool and an organ model.
2. Find neighborhood triangles around the contact triangle that need local enhancement.
3. Subdivide each coarse triangle into a finer set of triangles.
4. Read displacements of nodal points from the global model. Usually, nodal points are vertices or centroids of each triangle.
5. Build interpolation functions generating smooth deformation fields.
6. Compute displacements of points in refined triangles by using the interpolation functions.
7. Render the deformed entire object.

In the following paragraphs we explain the subdivision algorithm that generates a set of triangles which have a smaller size with varying normal distribution from a larger, flat triangle. Among various subdivision algorithms in literature (e.g., Loop [47], Catmull-Clark [48]), we choose an algorithm called the PN algorithm developed by Vlachos *et al.* [49]. It divides a triangle into a set of triangles, which have variations of normal vectors. The geometry of a PN triangle is based on one cubic Bezier patch. The set of subtriangles matches the position and normal vector at the vertices of the flat triangle. Since no additional data other than the position and normal vector of three vertices are necessary, it is computationally efficient compared to the other techniques requiring the information of neighboring polygons.

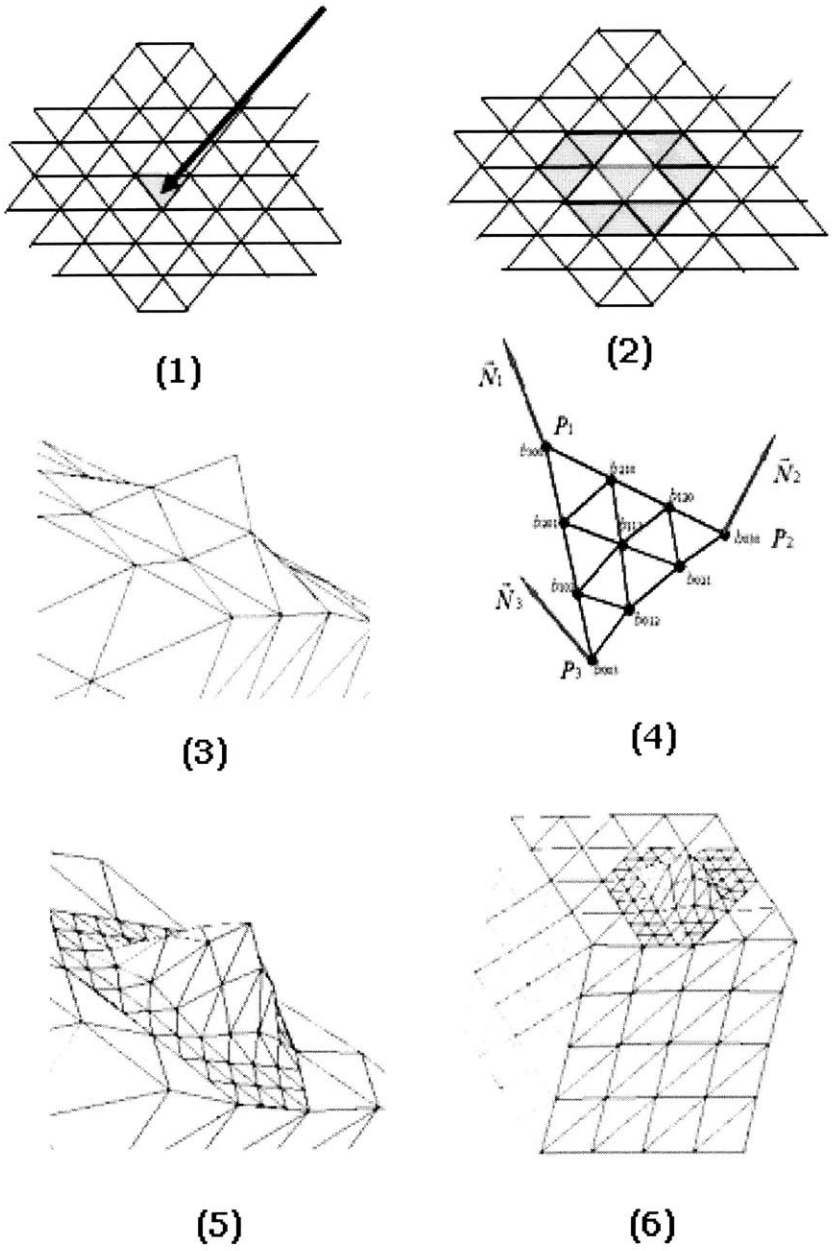

Figure 4-3 Procedures in the local enhancement technique 1) detect a collision and a corresponding. 2) Find neighborhood polygons around the tool-tissue contact region. 3) Deformation of the BEM model without the local enhancement. We can see the graphic artifacts such as sharp edges due to its insufficient resolution. 4) Inputs and output of the PN triangle algorithm. 5) Closed view of the enhanced deformation 6) The deformation of BEM model with the local enhancement technique.

We define the "edge neighborhood" of triangles around a contact triangle as those that share the same edge with the contact triangle. "Vertex neighborhood" is also defined as the group of triangles sharing a vertex with the contact triangle. Figure 4-4 shows the definition of the two groups of triangles. (The vertex neighborhood triangles are identical to the group of triangles called "the first ring" in [41].) When the contact occurs in one triangle, the triangles in the neighborhood are selected and subdivided. We can repeat this process if finer resolution is required.

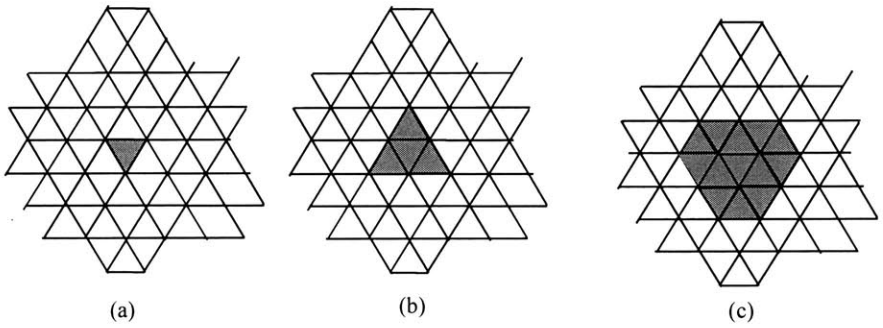

Figure 4-4 Definition of subdivision area (a) A triangle in touch with a tool (b) Triangles sharing edges with the contact triangle (c)Triangles sharing vertices with the contact triangle.

Although the PN triangulation algorithm significantly reduces the storage requirements and is easy to implement, the flatness from original triangles still remains in subdivided triangles and it is quite noticeable. Hence the extra smoothing or interpolation process is necessary to remove the flatness. The interpolation of a variable **u(x)** is:

$$\mathbf{u}^h = \sum_{i=1}^{N} a_i(\mathbf{x})\hat{\mathbf{u}}_i \tag{4.9}$$

where $\hat{\mathbf{u}}_i$ is the value of **u(x)** and a_i is a basis function at node 'i'. The selection of a set of basis functions determines the performance of the interpolation. In our application, we require local approximations, high convergence rate, and, of course, computational

efficiency. We select two techniques for generating the basis functions: the moving least squares technique and the Shepard partition of unity approach [50]. The moving least squares functions reduce to the Shepard functions as a special case.

Both these techniques use weight (or window) functions ($W_i(x)$) which are radial functions with compact support (i.e. these functions are highly localized, being nonzero in only a small region of the domain). They allow the localization of the approximation and result in a banded matrix. The influence of a node is governed by a decreasing radial weight function, which vanishes outside the domain of influence of the node. For better understanding, we demonstrate one dimensional cases of the interpolation functions (see Figure 4-5). They generate smooth curves from a set of nodal points. If increasing the radius of influence r, the curve becomes smoother but loses the locality of the approximation.

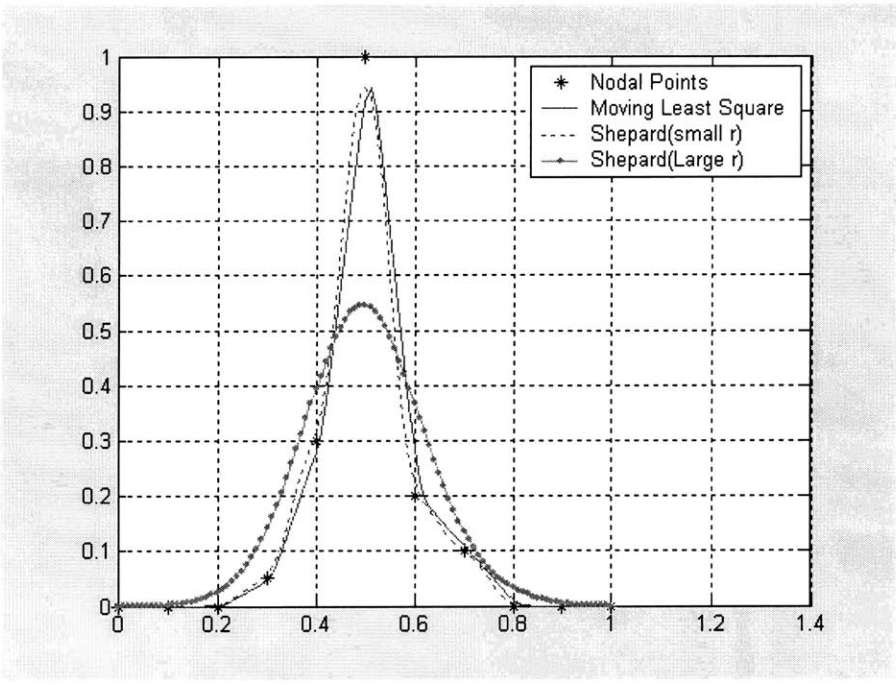

Figure 4-5 Interpolation functions for a one dimension. They generate smooth curves from a set of nodal points. If increasing radius of influence r, the curve becomes smoother but loses the locality of the approximation.

Using a set of nodal points $\{\mathbf{x}_i\}$ i =1 ,2...,N the approximation $\mathbf{u}^h(x,y,z)$ is

$$\begin{aligned} \mathbf{u}^h &= \Phi^T \cdot \hat{\mathbf{u}} \\ \hat{\mathbf{u}} &= [\mathbf{u}_1, \cdots \mathbf{u}_N] \end{aligned} \tag{4.10}$$

If moving least squares functions are used, then

$$\begin{aligned} \Phi^T &= \mathbf{p}^T(\mathbf{x})\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{p}^T &= [1 \quad x \quad y \quad z \;] \\ \mathbf{P} &= \begin{bmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_N & y_N & z_N \end{bmatrix} \\ \mathbf{A} &= \mathbf{P}^T\mathbf{W}\mathbf{P}, \quad \mathbf{B} = \mathbf{P}^T\mathbf{W} \\ \mathbf{W} &= \mathrm{Diag}(w_1, \cdots w_N) \end{aligned} \tag{4.11}$$

On the other hand, if the Shepard partition of unity approach is taken, the Shepard function at node 'i' is defined as

$$\Phi_i = \frac{W_i}{\sum_j^N W_j} \tag{4.12}$$

For both cases, the choice of a radial weight function W_i determines the degree of continuity and differentiability of the approximation as well as the cost of computation. We choose quartic spline weight functions,

$$\begin{aligned} W_i &= \begin{Bmatrix} 1 - 6m^2 + 8m^3 - 3m^4 & 0 < m < 1 \\ 0 & m > 1 \end{Bmatrix} \\ &where \quad m = (\|\mathbf{x} - \mathbf{x}_i\| / r_i) \end{aligned} \tag{4.13}$$

The choice of the radius of influence r_i at node 'i' is also important as it determines the number of nodal points included in the influence of node 'i'. Because the moving least squares method offers higher convergence rates but is computationally more expensive than the Shepard function, we selected the Shepard function for real time implementation. Figure 4-6 compares the deformation of a cube using (b) a coarse model and (c) a coarse model with local subdivision and smoothing using Shepard functions. In (c) a smooth and visually plausible deformation field is observed compared with (b). We would like to note that our technique is for purely geometric smoothing since no mechanics or material properties are considered in the computation as shown in [51] [32]. However, this leads to a significantly improved visual display of the deformation field.

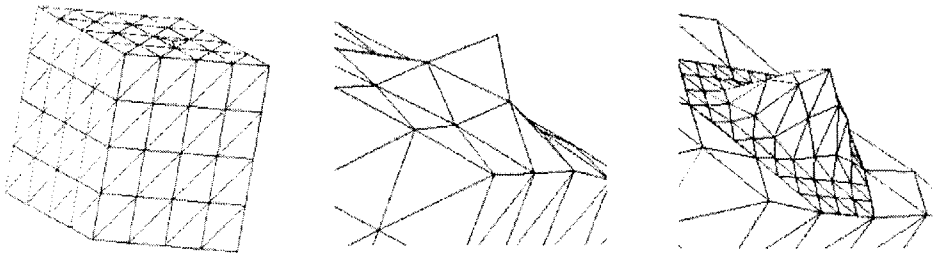

Figure 4-6 Comparison of visual deformations. (a) Unmodified cube model (b) Deformation with the coarse model (c) Coarse model with local subdivision and smootheing using Shepard functions

We have implemented the technique to a kidney model from real anatomical images. Figure 4-7 shows the pulling simulation of a kidney model for a MIS simulator generated using the hybrid modeling technique. We used a geometrical kidney model created by the Visible Productions LLC (www.visiblep.com). The company generated the models by manually segmenting the NLM's Visible Human male data set. Because it contains very highly detailed geometry, we reduced the model with the OPTIMIZATION function in 3D Studio Max.

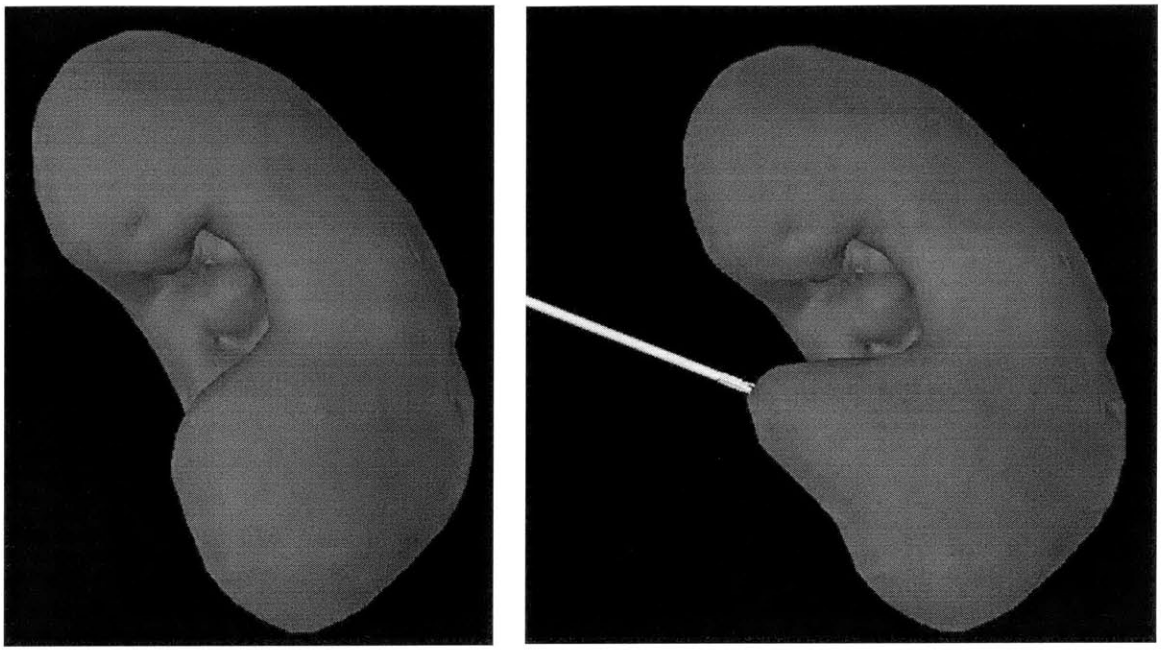

Figure 4-7 Snapshot of a palpation on the kidney model. It generates simulated force responses as well as visual responses of tool-tissue interactions. The model is created from the Visible Human Dataset and optimized in the 3D studio Max. It has about 6K points and 10K triangles.

4.4 Discussion

In this chapter, we have presented an efficient hybrid modeling approach for the simulation of tool-tissue interactions. A coarse global model employing the boundary integral formulation is coupled to refined local enhancement techniques around the vicinity of the tool-tissue contact region. A graphic smoothing scheme using localized interpolation functions such as the Shephard functions are used. The global model is used to compute the tool-tissue interaction forces for haptic rendering. The local enhancement technique makes it possible to avoid the use of a high-resolution model over an entire domain, which is computationally inefficient. The techniques described in this chapter are equally applicable to the finite element based discretization. It is well known that the finite element technique has advantages over the boundary element technique in the modeling of nonhomogeneous and nonlinear media.

A limitation of the current approach is that the refinement in the tool-tissue interaction region is purely geometric. One of the possible approaches is to use the local deformation model capturing mechanics of the local region rather than the entire region. In short, the local model computes the deformation fields in local area and intersection forces based on the boundary conditions from the BEM global model [52].

Chapter 5 In Vivo Measurements of Intra - abdominal Tissues

This chapter will begin with the importance of the *in vivo* soft tissue properties to the development of realistic surgical simulators. We will also review previous work done in the tissue property measurements. The following sections will describe hardware and software system of the measurement system we developed. The details of animal preparation and the experimental protocol will be presented. We made several sets of *in vivo* measurements on pig organ tissues including liver, kidney, spleen and esophagus. The experimental results will be presented at the end of the chapter.

For providing valid bounds of our measurements, we were mainly measured in the behaviors of the tissues when subjected to low frequency (less than 7Hz) and large indentation stimulations (less than 10mm) , which may occur in real operations.

5.1 Introduction

The material properties of various tissues have been measured *ex vivo* for decades [53, 54]. However, after removing samples from the living state, the conditions of tissues change drastically from due to 1) temperature (changes in viscosity) 2) hydration (drying up might change elasticity or viscosity) 3) break-down of proteins (change in stiffness) 4) lost of blood supply. Moreover, the boundary conditions of the sample are different from *in vivo* states and therefore its force-displacement relationship will change when the sample is cut away from the rest of the organ. As an example, Figure 5-1 shows the static force responses

of liver at 6, 12, 24 hours after death of animal. The responses to the same indentation depth increase (become stiffer) as time progresses. Significant changes of viscoelastic properties were also observed.

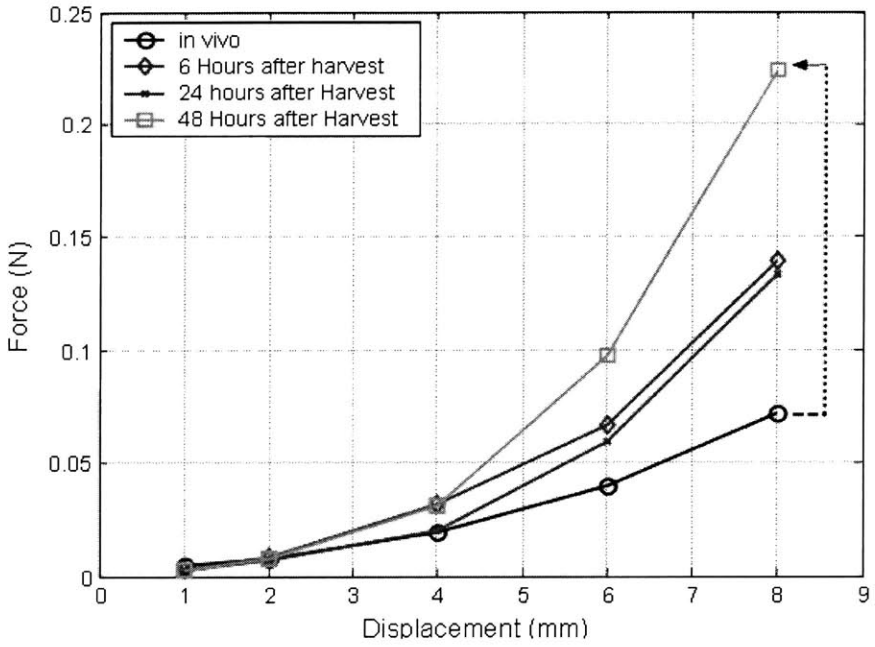

Figure 5-1 ***Ex vivo*** **and** ***in vitro*** **steady state force responses of pig liver. The liver was harvested after the pig was euthanized and stored in a saline solution. As time progressed, the force response was increased.**

Recently, considerable research has been done on measurements of mechanical properties of tissues *in vivo* [55-58]. Usually mechanical indenters are used to characterize the relationship between applied indentations and replied forces from the tissues. For example, the Dundee Single Point Compliance Probe [56] is designed to measure force response of organs to indentation stimuli during an open surgery. As a hand-held probe (Figure 5-2), it can indent up to a maximum depth of 6mm and measure the corresponding reaction forces. Force-displacement curves of the organs can be produced from the corresponding data and stiffness of the tissues inferred. It has been used to obtain force-displacement relationship data of the human liver. They showed significantly non-linear responses and variation in stiffness both between organs as well as between healthy and

diseased tissues. This device is useful for taking quasi-static measurements on tissue but it would be difficult for us to acquire dynamic data due to its hand-held nature.

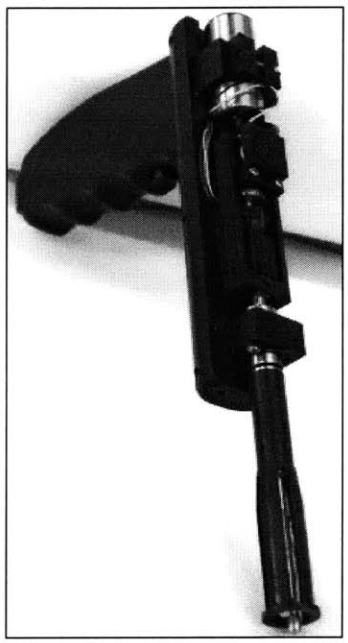

Figure 5-2 Dundee single point compliance probe

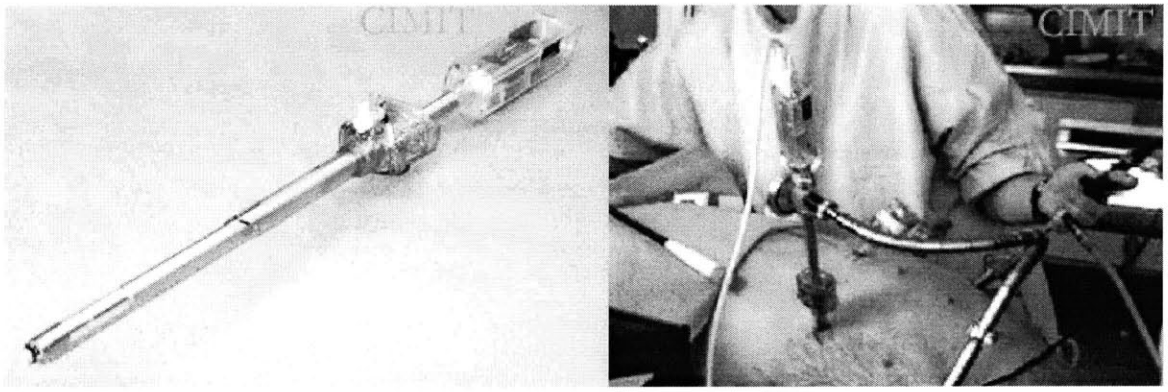

Figure 5-3 (left) TemPeST 1-D device developed by Ottensmeyer. (Right) *in vivo* animal testing using the device laparoscopically.

Ottensmeyer [59] designed a device to measure mechanical properties named TeMPeST 1-D. From the experiments on pig livers performed laparoscopically, he measured solid organ impedance with 500μm maximum amplitude and a 100Hz bandwidth. Under the "simi-

infinite" assumption, a Young's modulus was estimated around 10-15kPa from the measurements. In vivo material properties of organs were also measured using modified versions of laparoscopic instruments (Hannaford et al., 1998). The Force Feedback Endoscopic Grasper (FREG, see Figure 5-4) makes use of standard laparoscopic instruments with the tool shaft of the instrument mounted onto the slave subsystem and the tool handle attached to the master subsystem. Using either teleoperation or software, the FREG is able to control grasping forces at the tool tip. The FREG was used with software control in experiments to palpate tissues at 1Hz. Quasi-static stiffness was inferred for the porcine colon, small bowel, spleen, stomach and lung. They also tested a force-displacement hysteresis behavior with the same device. Rosen *et. al.* (1999) instrumented a laparoscopic grasper with force/torque sensors to measure force and torque at the surgeon hand/tool interface when the instrument was used to perform laparoscopic surgeries on pigs. This force/torque data was correlated with visual feedback from the endoscopic camera to create a database for surgical simulation and the optimization of performance in robotic surgeries.

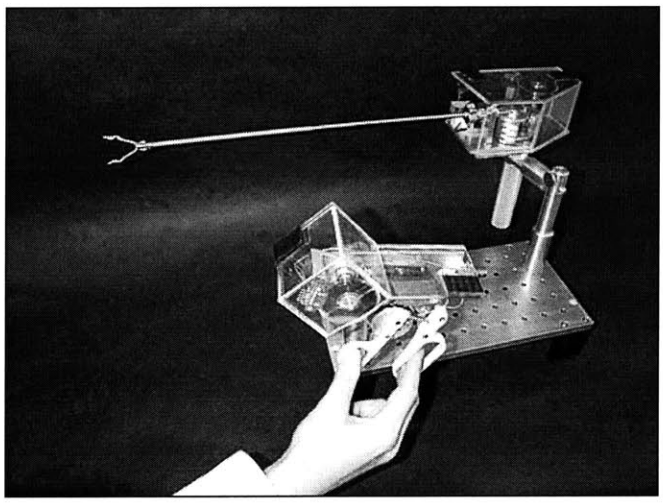

Figure 5-4 The Force feedback Endoscopic Grasper (FREG) with a Babcock grasper.

Brouwer *et. al.* (2001) inferred the mechanical properties of various abdominal tissues using a uniaxial tensile machine that grasped the abdominal tissues and measured the forces corresponding to different stretch ratios. In addition to these instrumented

devices, they also used a robotic device in their tissue measurements. It was programmed to indent the various abdominal tissues at constant velocity. Contact forces corresponding to displacements of suturing needles through abdominal tissues were also measured.

Although useful, the measurements in the previous work are not entirely useful for our application in a sense that the indentation depth, frequency bandwidth and boundary conditions that they consider are different from our requirements. Our group has performed animal experiments to characterize *in vivo* force–displacement relationship of the liver and lower esophagus of pigs when subjected to mechanical indentation by the robotic device [60] for the purpose of development of the Minimally Invasive Surgery simulator.

One technique worth mentioning here is "the Truth Cube" developed by Kerdok *at el.* [61]. They made a silicone rubber cube with an exact pattern of embedded Teflon beads that was tested under controlled loading conditions while CT images were taken. The known material properties, geometry, and carefully controlled boundary conditions resulted in a complete set of volumetric displacement data. They compared these data with the FEM model to validate results. This technique can be used to establish a standard for use in validating soft tissue models if they extend the technique to more general 3 dimensional shapes.

5.2 Measurement System

5.2.1 Measurement system design

We used a robotic device to deliver a mechanical indentation on the tissue due to the ease of handling and programming the motion required for the various modes of indentation. The indentation stimuli were delivered using the haptic interface device, Phantom Premium-T 1.5 (SensAble Technologies, www.sensable.com) that was programmed to perform as a mechanical stimulator. The Phantom has a nominal position resolution of 30 μm and a frequency bandwidth that significantly exceeds the stimulus frequencies that were being employed in this experiment. Reaction forces were measured using a six-axis force transducer, Nano 17 (ATI Industrial Automation) that was attached to the tip of the Phantom. The transducer has a force resolution of 0.781 mN along each of the three

orthogonal axes when connected to a 16-bit A/D converter. The indenter was a 2 mm diameter flat-tipped cylindrical probe that was fixed to the tip of the Phantom with the force transducer mounted in-between to accurately sense the reaction forces as shown in Figure 5-5.

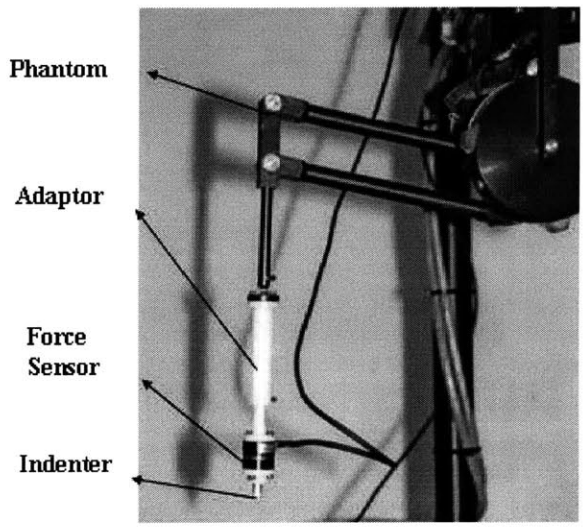

Figure 5-5 A phantom equipped with the indenter and a NANO 17 force transducer. The adaptor was machined to hold the indenter/sensor assembly

5.2.2 Validation of the testing device

The main idea of the measurement system for this study was to use a robotic device as a mechanical indenter generating both static and dynamic motions. However, the dynamics of the device were as a black box in the system so we should investigate whether the dynamics of the device distorted the measurements. To validate this, we developed a lumped parameter model to represent the behavior of the device both in free motion and in contact with soft tissues. Because most mechanical devices behave as low pass filters from a signal-processing viewpoint, it is important to clarify the mechanical bandwidth, which limits the dynamic range of the measurements. This model also predicts the maximum controllable frequency of the device without distortion of data. The model requires the parameters to represent the dynamic behaviors of actuators, linkage structure and contact stiffness with soft material. In Figure 5-6, the model consists of two concentrated masses connected by linear springs. To validate our model, we measured of the frequency response of the robot using an external force transducer. Figure 5-6 also shows the frequency

response from simulation and experiments. The low frequency region (less than 30 Hz), we can see that the dynamics of the robot does not influence the measured data. It also shows that the structural resonance of the device is much higher than the planned indentation frequency as illustrated. Therefore we conclude that the performance of the device is adequate over the frequency range of interests (0-7Hz).

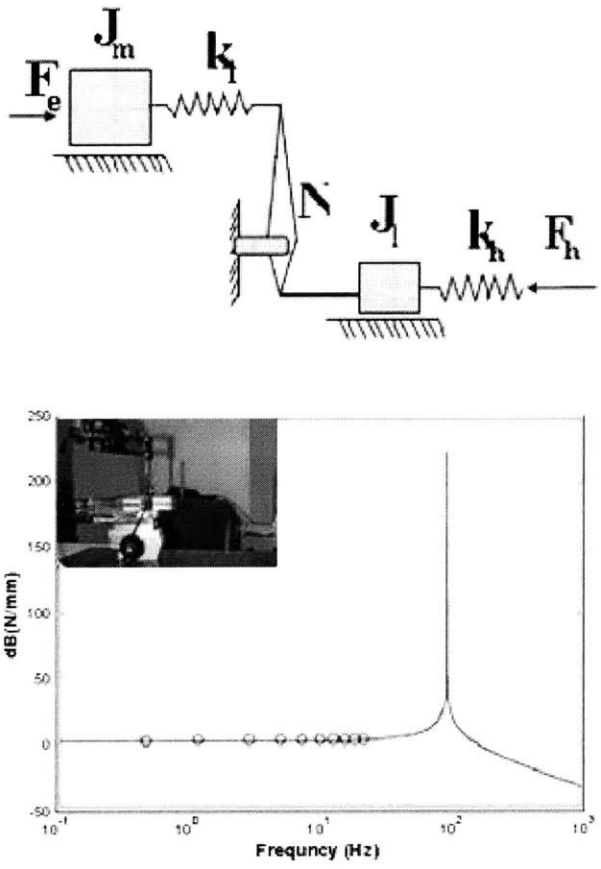

Figure 5-6 (Upper) The lumped parameter model of the phantom. The numerical values were obtained by the measurement and from the manufacturer. (Lower) The frequency response of the phantom. Circles represent the experimental data.

5.2.3 GUI and motion controller

A graphic user interface (see Figure 5-7) was coded using MS Visual C++ v6.0 and GHOST v4.0 from SensAble Technology. It uses an Active X component to communicate with the sensor at 1 KHz. It also stores both measured forces and positions of the indenter

at 1K Hz on hard disk for further analysis. It provides a convenient interface to generate the trajectories for the indenter, monitor the measuring system, control the motion and perform post-processing. We can also jog the robot to approach the measuring point using jogging buttons. They provide small Cartesian motions at each click.

The interface allows the generation of four different indenting motions: ramp-and-hold, sine and two chirp waveforms (they were not used in out experiments) as shown in Figure 5-8. We can easily change the frequencies/amplitudes of the trajectories and links them to a single button to expedite the measurement.

A PID controller drives the robot to the desired trajectory from obtained the motion generator.

$$F_{control} = K_p(e) + K_d(\frac{de}{dt}) + K_i \int edt \tag{5.1}$$

where $F_{control}, K_P, K_d, K_I, e$ are the driving forces for the robot, the proportional gain, the derivative gain, the integral gain and the error between the desired trajectory and the current position, respectively. The gains were adjusted by a careful tuning process.

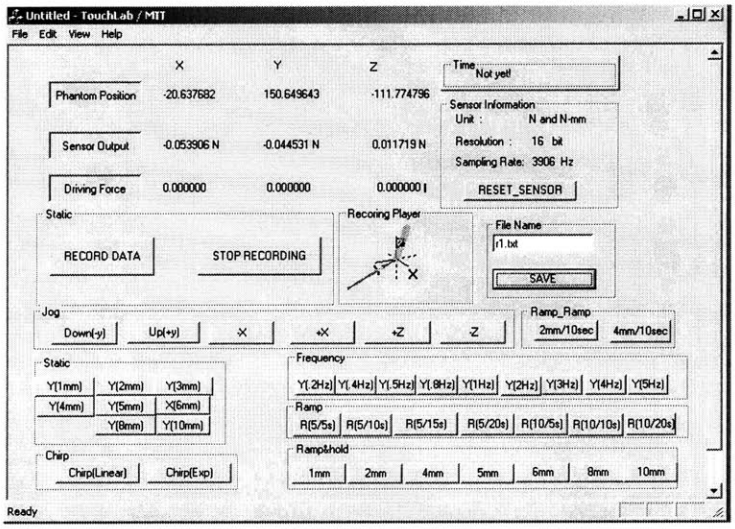

Figure 5-7 Snapshot of the Graphic User Interface (GUI) developed for the experimental control. It provided jog motion and programmed motion for the indentations.

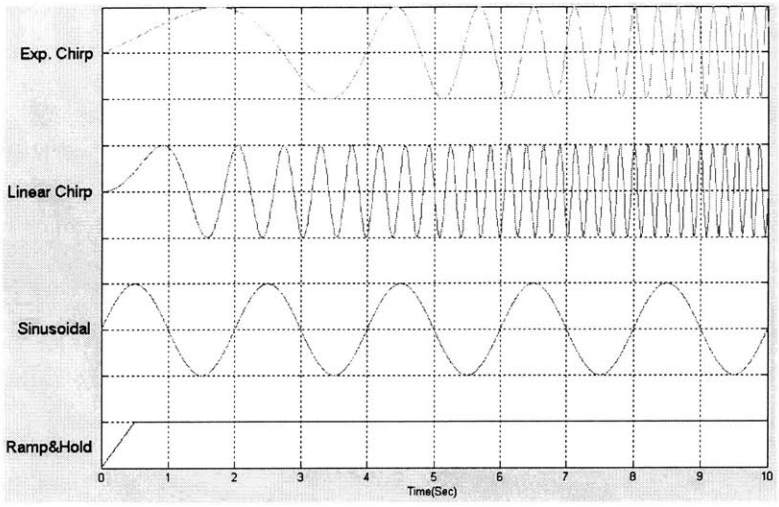

Figure 5-8 Four waveforms supported by the measurement system

5.3 Animal Preparation and Experimental Procedures

We conducted a series of experiments on the intra-abdominal organs of pigs (see Figure 5-10) including liver, kidney and spleen at the Harvard Center for Minimally Invasive Surgery in collaboration with surgeons from the Massachusetts General Hospital (MGH). An experimental protocol described below was developed in cooperation with the Animal Research Center at Dartmouth College, and was approved by the Committee on Animal Care at M.I.T (Approval number 00-032, Title: *In Vivo* Measurement of Biomechanical Characteristics of Organ and Soft tissues in Abdominal Region during a Laparoscopic Surgery). Eleven female pigs were used for open surgical activities (The weights and tested organs of pig subjects are listed in Appendix A). Anesthesia was induced with 20mg/Kg ketamine and 2mg/kg xylazine via. intra-muscular injection. 0.04 mg/kg atropine was also administered to reduce bradycardia and salivation. The pigs were intubated and placed on assisted ventilation using 100% oxygen, with 1-5% halothane or isoflurance to maintain anesthesia. The depth of anesthesia was monitored continuously by monitoring heart and respiratory rate, as well as corneal reflex. An intravenous catheter was inserted to

feed fluids and injectable anesthetics and analgesics were available as needed. The protocol called for euthanasia of the pig before recovery from anesthesia, accomplished with concentrated KCl solution administered intravenously. After euthanasia, organs could be harvested for the *ex vivo* measurements.

The pigs were first put under general anesthesia and placed on the surgical table (see Figure 5-9). A midline incision was then made at its abdominal region and dissection carried out on the anatomical structures to expose the organs. The surgical table was adjusted such that indentation stimuli were normal to the organ surface. The indentation sites were kept sufficiently moist throughout the experiment to ensure that the organs were maintained as closely as possible to their natural state. The indentation sites were chosen to be at the thickest part. The tip of the Phantom, with the indenter attached was then lowered into the abdominal region with care taken for each indentation stimulus to be delivered to the same site on the organs (see Figure 5-11).

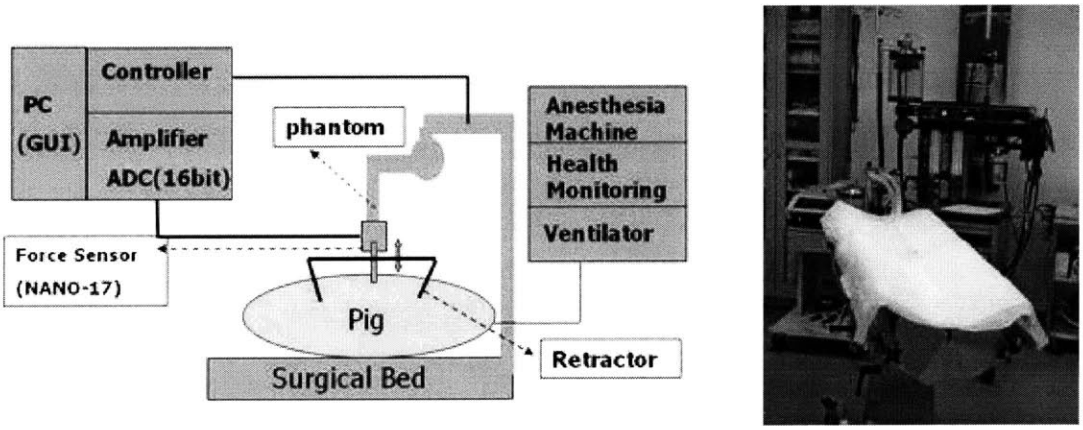

Figure 5-9 (Left) Schematic of the experimental setup (Right) Surgical table with a life support system at the Harvard Center for Minimally Invasive Surgery

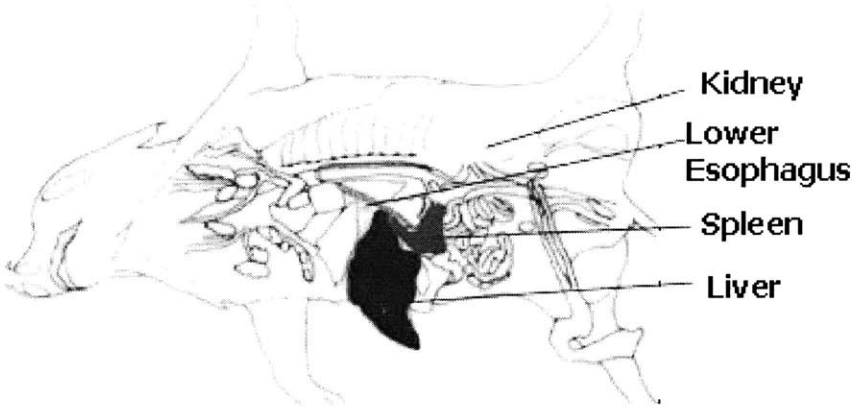

Figure 5-10 Anatomical diagram of pig intra-abdominal organs for the testing

Figure 5-11 View of animal testing during open surgical setup of pig spleen

5.4 Experimental Results

5.4.1 Force responses against a ramp-and-hold stimuli

Figure 5-12 and Figure 5-13 show the *in vivo* force responses of liver and kidney to ramp and hold indentation stimuli. Indentations were made to a perpendicular direction with respect to tissue. We can observe two distinct characteristics from these sets of the

responses: nonlinear elasticity and viscoelasticity. The responses show a typical behavior of viscoelastic material known as "stress relaxation".

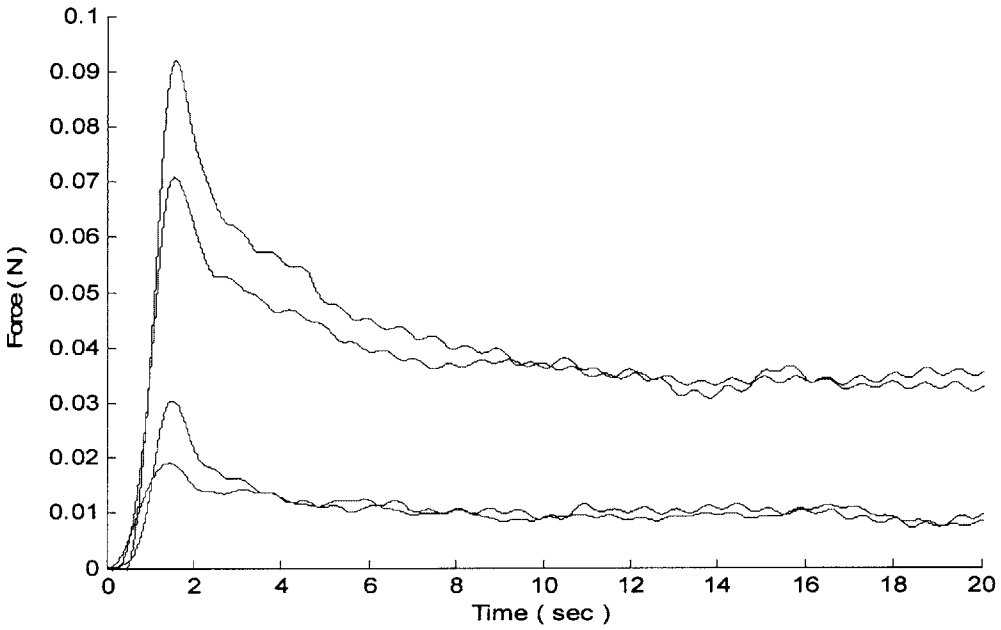

Figure 5-12 *In vivo* force responses of pig liver to ramp and hold indentation stimuli. From the top, 8mm, 6mm, 4mm, 2mm amplitudes were used. The ripple is due to cardiac action of the subject.

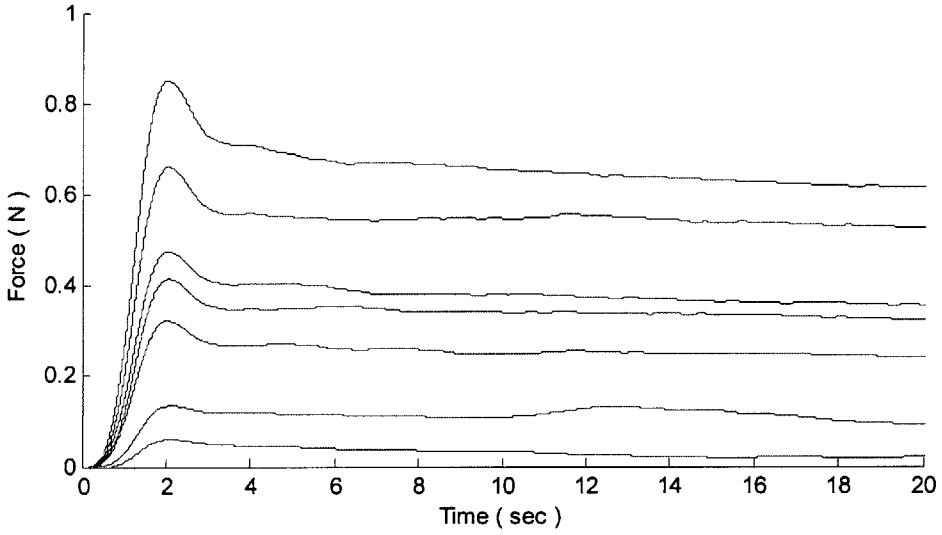

Figure 5-13 In vivo force responses of pig kidney to ramp and hold indentation stimuli From the top, 8mm, 6mm, 5mm,5mm, 4mm,2mm and 1mm amplitudes were used.

5.4.2 Static responses

Because all measurements were taken in suspension of the ventilation system, periods lower than 40 seconds might cause the damage to animal subjects so it was unsafe to take the steady state responses over more than a minute. But we could observe that the responses converged to steady state values within 10% error after about 40 seconds, although this depends on the relaxation time constants of the tissues. Figure 5-14 shows static force responses of four pig organs. The data were taken from the ramp-and-hold indentation experiments after the data converged to the steady state value. Nonlinearities are observed from the data and variation in individual subjects is also observed. Kidney is stiffer than the other organs and this is also found in *in vitro* data in the literature [53]. The response of the esophagus is more linear than the responses of other organs. Table 5-1 lists the maximum and minimum values of each indentation experiment.

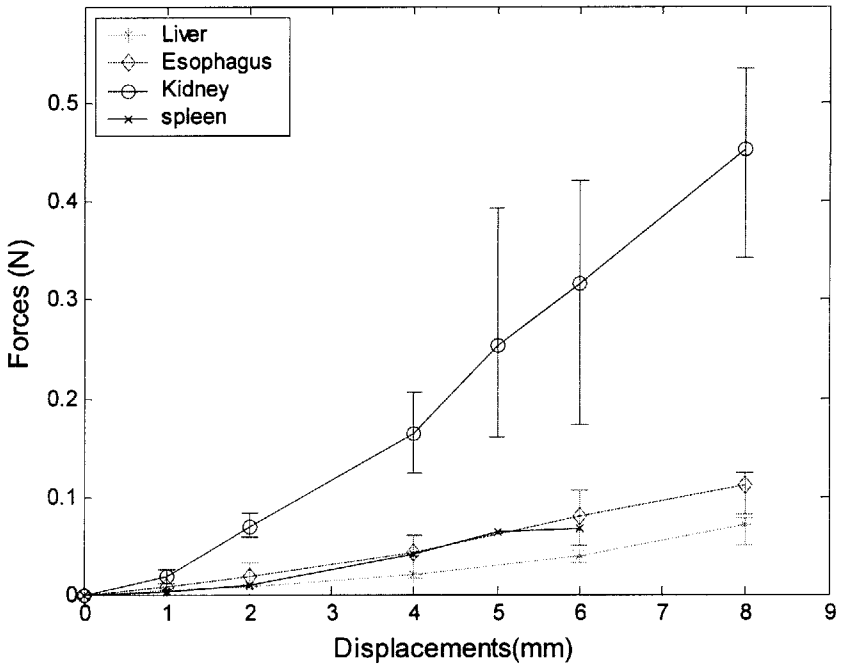

Figure 5-14 Static force responses of the intra-abdominal organs measured from in vivo conditions Vertical bars represent variations of each animal subjects.

Table 5-1 Variations and mean values of the static force responses (Unit: N).

Indentation depth	Liver			Esophagus			Kidney			Spleen		
	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
1 mm	0.0026	0.0048	0.0054	0.0027	0.0072	0.0121	0.0054	0.0189	0.0258	0.0016	0.0030	0.0043
2 mm	0.0078	0.0079	0.0100	0.0104	0.0182	0.0321	0.0594	0.0689	0.0839	0.0078	0.0094	0.0110
4 mm	0.0166	0.0196	0.0209	0.0199	0.0427	0.0604	0.1245	0.1632	0.2066	0.0397	0.0411	0.0426
5 mm	0.0323	0.0213	0.0268	0.0402	0.0544	0.8761	0.1608	0.2542	0.3947	0.0645	0.0645	0.0646
6 mm	0.0443	0.0396	0.0317	0.0497	0.0791	0.1058	0.1732	0.3164	0.4228	0.675	0.0682	0.0689
8 mm	0.0498	0.0718	0.0779	0.0817	0.1114	0.1230	0.3431	0.4541	0.5360			

5.4.3 Sinusoidal responses

Figure 5-15 presents the responses to sinusoidal indentation stimuli of the organs. The force-displacement profiles are elliptically shaped and enclose a non-zero area. The shape of the force-displacement profiles is a consequence of the viscoelastic nature of the material and the area enclose by the loop is known as the hysteresis [62]. This non-zero area represents the amount of viscous energy dissipated per cycle of the indentation.

Figure 5-16 shows the frequency responses of the tissues measured at 0.5,1,2,3,4,5 Hz indentations with 2 mm peak to peak amplitude after 2mm pre-indentation. The stiffness of kidney decrease as the indentation frequency increases while the responses of esophagus and liver are relatively flat over the frequency range. It is hard to say there is a significant characteristic feature in the plot.

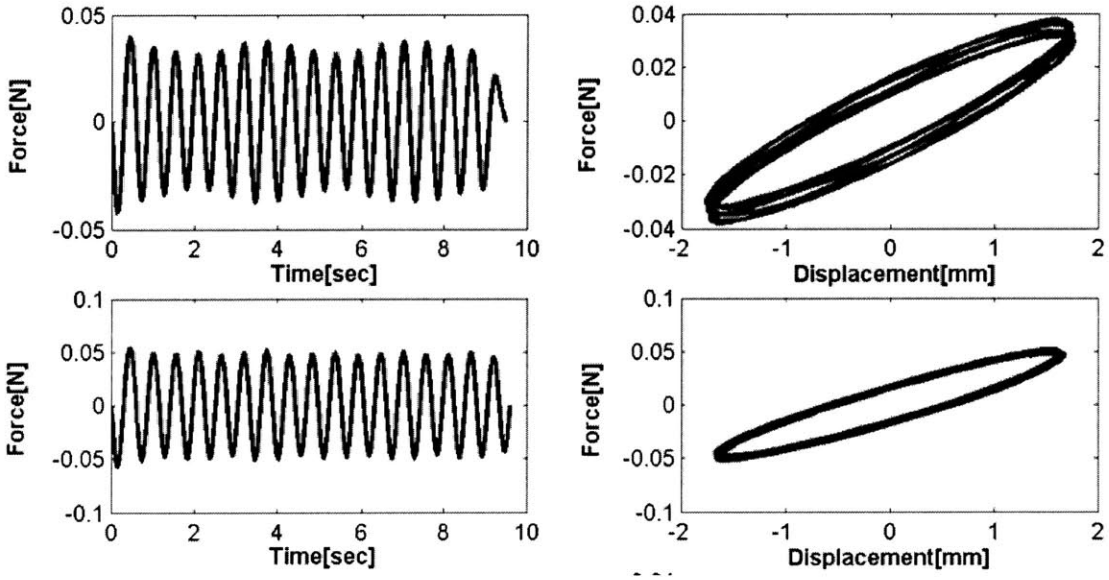

Figure 5-15 Typical frequency responses at 2Hz (left column) of liver and corresponding force-displacement plots (right column). Plots on the left of figures show the force responses with time while plots on the right show the force-displacement profiles of the corresponding indentation stimuli.

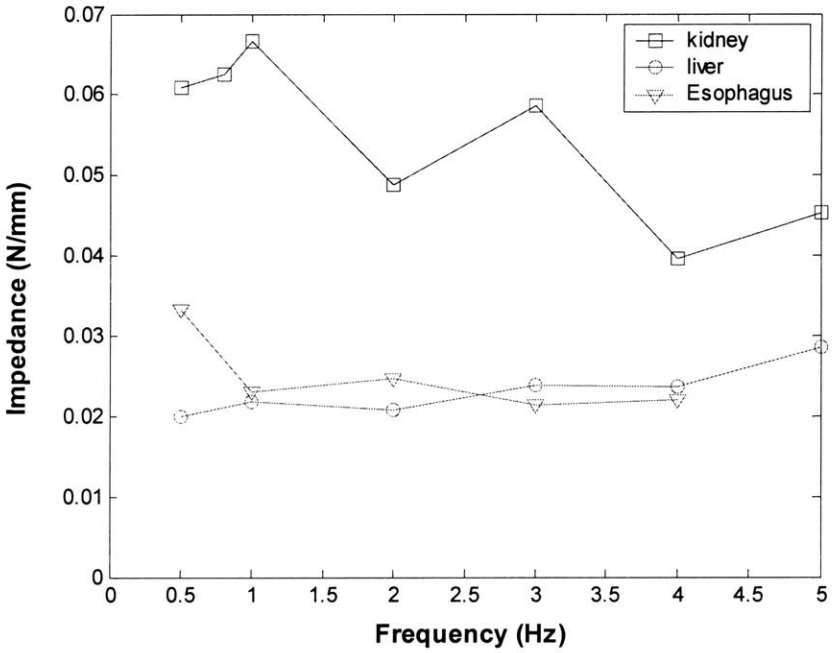

Figure 5-16 Impedances of the organs (magnitude/ indentation depth) to sinusoidal indentation stimuli. (Data from four animals and mean values are presented.)

5.5 Discussion

This chapter provides an overall framework for the measurement of *in vivo* mechanical properties of intra-abdominal organs for surgical simulation. The robot was programmed to deliver various forms of indentation stimuli to organs in live conditions. A 2 mm flat-tipped cylindrical probe was attached to the tip of the robot with a force transducer in-between to accurately measure the reaction forces. An experimental protocol was developed for the systematic measurement of the mechanical properties of biological tissues for surgical simulation. Ramp and hold as well as sinusoidal indentation stimuli were used in this study. These indentation stimuli were delivered to the organs and reaction forces measured. Conditions for the stimuli were designed such that they closely mimicked the conditions in an operating room.

The problem encountered was that of unwanted motion due to breathing and cardiac movements. The two kinds of motions are superimposed on the measurements and even some breathing motions could drive the indenter over maximum indentation amplitudes. The solution that was used in these experiments was to suspend ventilation during the measurement. The animals effectively had their breath held for periods of about 20-25 seconds. After finishing the measurement, the ventilation was restored to allow the animal to recover for 2-3 minutes. Sometimes, spontaneous breathing occurred during the measurements and the animals required longer resting periods in between tests.

Variations were small from the same animal measured within one hour interval. However, the data varied more as the anesthesia time progressed. There are significant variations in experiments conducted on different animals as shown in Table 5-1. Because each experiment had different conditions such as weight of pig, depth of anesthesia or healthiness, it is understandable to have such variations in different animals.

There is another category of tissue property measurement techniques known as the non-invasive technique in literature [63]. After applying an external deformation to the body, the internal strain fields are measure by imaging techniques such as magnetic resonance imaging (MRI), or ultrasound. They can calculate the mechanical properties like

Young's or shear moduli from the measured strain fields. Although these non-invasive techniques have been improved rapidly over the last few decades, the techniques do not provide proper ranges yet both in frequency and displacement required for surgical simulation. In addition, computational complexities finding the parameters from the images are very expensive.

Chapter 6 Characterization of Properties of Soft Tissues

The goal of tissue characterization is to find the parameters by correlating of the mathematical model with the experimental data. These parameters then can be used in an off-line simulation using FEM packages for further analysis or in real time tissue models. From the tissue responses from the animal experiments described in the last chapter, two dominant characteristics are observed: nonlinear elastic and viscoelastic responses. These phenomena cannot be modeled by a simple linear elastic model so that more advanced models are necessary to represent this behavior. Assuming an isotropic, homogeneous and incompressible material, we developed a few tissue models that capture the behavior of soft tissues.

We determined the parameters of three biomechanical models in which vary in how accurately they describe a soft tissue behavior. First, we determined the parameters of a linear elastic model assuming a semi-infinite medium assumption. Second, sets of parameters of more accurate biomechanical models were determined by using a curve-fitting technique. Finally, we develop an algorithm to find the parameters of a continuum mechanics model, which is capable of describing the tissue behavior in a more comprehensive way.

6.1 Previous work

Soft tissue characterization and modeling has been investigated in order to understand mechanisms of traumatic injury. For example, Farshad *et al.* [64] characterized the material parameters of pig kidney based on uniaxial tension tests on *ex vivo* samples.

They fit their measurements to a nonlinear elastic Blatz model and a four-term viscoelastic model. They were mainly interested in a tissue behavior under high speed or impact loading conditions, which occur in accidents, so their measurements were mainly *ex vivo* measurements. Recent developments in medical instruments and surgical simulation motivated the tissue characterization under the lower speed loading conditions occurring in surgical operations. Davies *et al.* [65] developed a two dimensional mathematical model for pig spleen tissue from uniform compression tests using a large-area indenter. They solved the nonlinear constrained boundary value problems numerically with an exponential stress-strain law and a finite element model (FEM). By varying the model size, they showed that the forces measured at the indenter were insensitive to the size of the model. Miller [66, 67] developed a three-dimensional, hyperelastic, viscoelastic constitutive model for brain and abdominal organ tissues. The model was developed from a strain-energy function with time dependent constants.

Fitting our data to a mechanical model required a different approach. Our indentation data lacked many of the features that make ex vivo samples convenient to calibrate. Typically our samples had non-uniform cross-sectional area, and non-uniform strain across any given cross section. An analytic solution based on the boundary value problem was not a good candidate, given the complexity of the material behavior, the organ geometry, and the three-dimensional deformation imposed on the surface.

To circumvent these difficulties, inverse finite element estimation has been investigated for characterization of soft tissue properties [68]. This method estimates unknown material parameters for a selected material law by minimizing the least-squares difference between predictions of a finite element model and experimental responses as shown Figure 6-1.

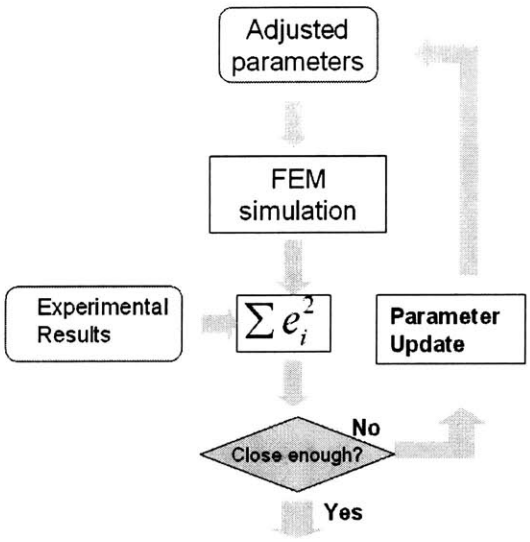

Figure 6-1 Schematic of the conventional inverse FEM estimation

6.2 Estimation of the mean elastic modulus

In spite of its limitations, a linear elastic medium provides the basis of tissue modeling and many tissue models in the current VR based simulators have used this linear elastic theory due to its simple implementation [31]. In such a case, only the mean elastic modulus, similar to the definition of Young's modulus is necessary to describe the mechanical behavior of soft tissues. From the relationship of force and displacement in the case of normal indentation by a right circular indenter (see Figure 6-2), the approximated mean elastic modulus can be computed by the following equation,

$$E = \frac{(1-\nu^2)F_z}{2a\delta_z} \tag{6.1}$$

E, ν, F_z, a, δ_z are the mean elastic modulus, Poisson's ratio, measured force, indenter radius, and indented depth, respectively. Assuming an incompressible material puts Poisson's ratio is close to 0.5. Table 6-1 lists the mean elastic modulus of various intra-abdominal organs of pigs from the animal experiments.

Table 6-1 Estimated Young's modulus and standard deviations. The moduli were computed from the average responses of each organ. The parameters of esophagus cannot be determined by this method due to its size and hollow structure

Organ	E (kPa)
Liver	3.22 ± 0.633
Spleen	4.88 ± 0.091
kidney	20.80 ± 4.54

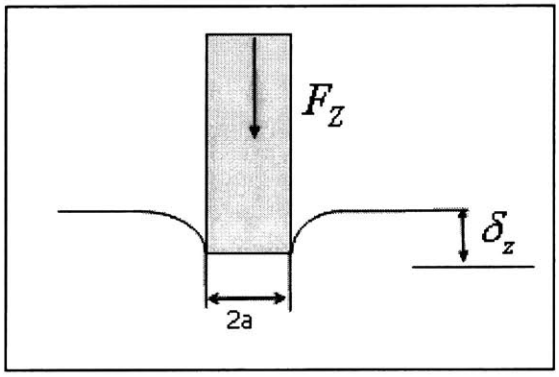

Figure 6-2 Modeling of the indentation experiments based on the "semi-infinite medium"

6.3 Lumped Parameter Models

As we mentioned earlier, the model based on the linear elasticity has limitations in modeling tissue behavior. Therefore, we developed a few lumped parameter models capturing the soft tissue behavior both in static and dynamic states using force-displacement relationships without internal models.

6.3.1 Nonlinear static elastic model

Recently, several attempts have been made to simulate nonlinear deformation of soft tissues in surgical simulations. From the various nonlinear elastic models (nonlinear constitutive equations) developed so far, we selected an exponential material model, because the exponential curve looks very natural for describing the experimental data.

One of the material models using an exponential relationship is the so-called Blatz model, which uses two separate exponential terms as follows,

$$P = \frac{\gamma}{\alpha + 1}(\lambda e^{\alpha(\lambda^2 - 1)} - \frac{1}{\lambda^2} e^{\alpha(1/\lambda - 1)}) \tag{6.2}$$

where P is the mean pressure at the indenter tip (force/area) and λ is compression ratio (undeformed length/ deformed length). α, γ are two independent parameters in the model and are obtained by using least square curve fitting procedures. Figure 6-3 shows that the prediction of this model matches the nonlinear response data we have obtained for the liver and lower esophagus of pigs.

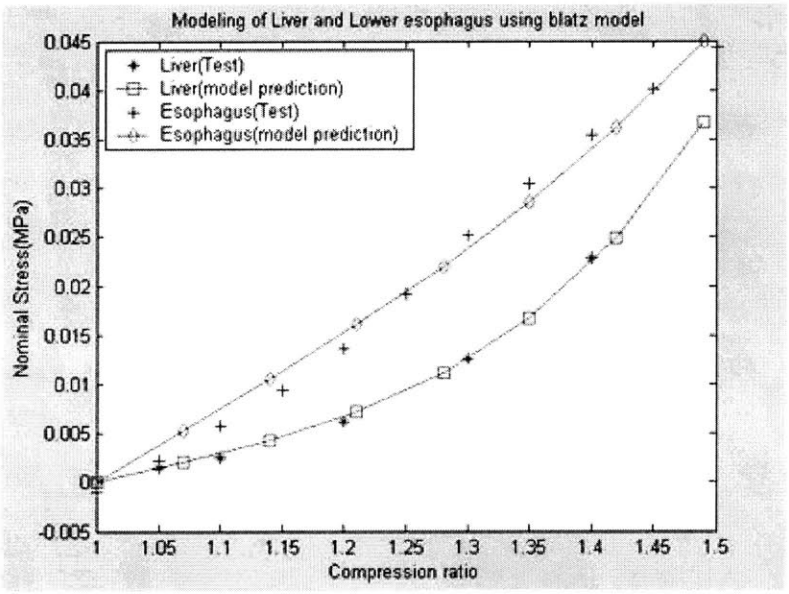

Figure 6-3 Static responses of liver and lower esophagus. The Blatz model can predict the behavior of soft tissues well. Effective area and length are used to convert the force-displacement relationship into the constitutive relationship.

6.3.2 Viscoelastic models

When a body is deformed rapidly and the indentation depth is maintained constant afterward, the corresponding forces decrease over time and settle to a steady state value. This phenomenon is called stress relaxation and is one of the typical features of viscoelastic materials. Linear or nonlinear lumped parameter models are often used to

predict the viscoelastic behavior of materials. The Kelvin model (a spring in parallel with a damper and a spring), also known as the standard linear solid is capable of modeling general linear viscoelastic behavior, including the stress relaxation observed in the experiments. The expression for the relaxation function G (t) can be written as

$$G(t) = k_1[1-(1-\frac{k_2}{k_1})e^{\frac{b}{k_1}t}]\,1(t) \tag{6.3}$$

where 1(t) is a unit step function. To determine the parameters of this assumed viscoelastic model, the experimental results on the pig liver were used. To fit the mathematical model to the experimental data, a least square optimization technique was employed. We can also use the real indentation profile without the assumption of a step input but the differences between the two cases are negligible. The estimated parameters have the following values.

k_1 = 0.008 (N/mm), k_2 = 0.007 (N/mm), b = 0.130 (sec N/mm)

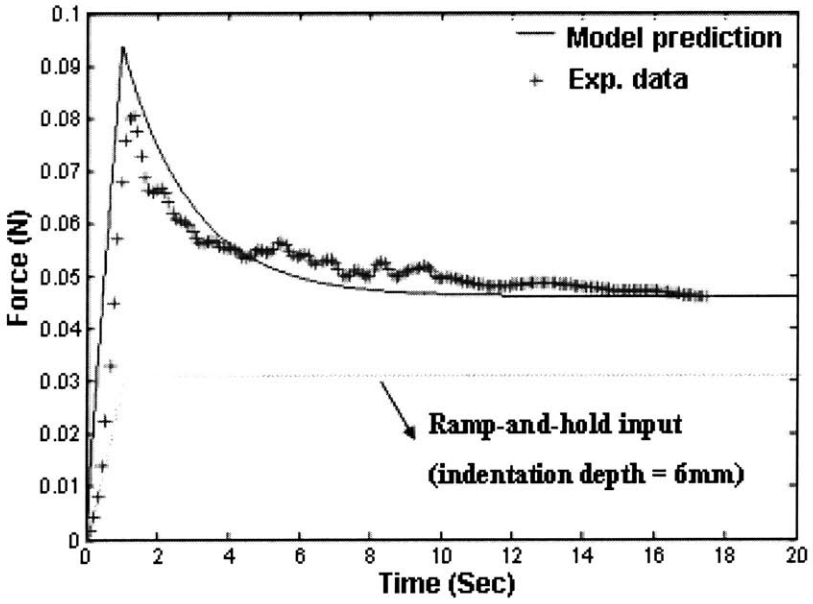

Figure 6-4 Experimental data and the predicted data by the 1st order Kelvin model

6.3.3 Nonlinear Lumped Parameter Model

Although the Kelvin model can predict the stress relaxation phenomenon, it is not capable of fully matching the force responses of a nonlinear viscoelastic body. For

example, the peak value of the force from the model does not match the experimental data well. To mdel the behavior more accurately, we use a system of nonlinear springs in parallel.

$$
\begin{aligned}
&F = F_0 + F_1 + F_2 + ... \\
&F_0 = k_{01} x^{k_{02}} \\
&\begin{cases} F_1 = k_{13}(x - x_1)^{k_{12}} \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = k_{11}(x - x_1)^{k_{12}} - x_2 \end{cases} \qquad \begin{cases} F_2(t+\tau) = k_{23}(x - x_3)^{k_{22}} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = k_{21}(x - x_3)^{k_{22}} - x_4 \end{cases}
\end{aligned} \tag{6.4}
$$

The parameters are determined using a nonlinear parameter estimation technique and listed in Table 6-2. As shown in Figure 6-5, the model more closely describes the viscoelastic behavior of the pig liver under various loading conditions than the single Kelvin model.

Table 6-2 Parameters for the nonlinear spring model of liver and esophagus

	K_{01}	K_{02}	k_{11}	k_{12}	k_{13}	K_{21}	k_{22}	k_{23}
Liver	0.002	1.680	0.025	2.236	0.007	0.080	2.184	0.0005
Esophagus	0.004	1.610	0.025	1.897	0.001	0.000	0.0000	0.0000

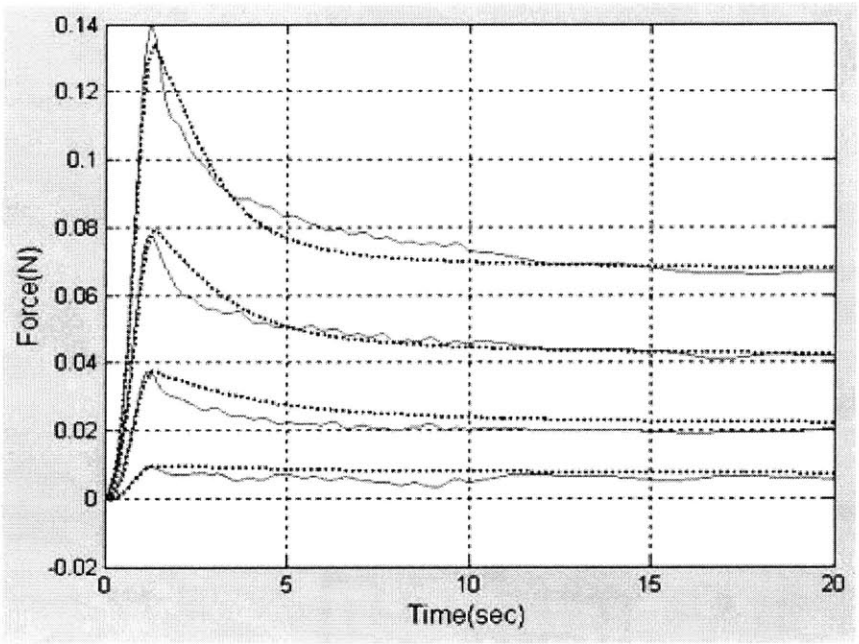

Figure 6-5 Viscoelastic responses of experiments of liver and model prediction

6.4 Inverse FEM Parameter Estimation

For lumped parameter models, it is relatively easy to find the parameters whatever models are used because they do not integrate a geometrical model between the input and output. Therefore, the resulting parameters are not able to be extended to a model considering continuum mechanics. We present the most complex and most comprehensive characterization method in this section.

6.4.1 Material models

We used the quasi-linear viscoelasticity (QLV) framework proposed by Fung [54]. This model has been successfully applied in a number of different tissues, such as lower limb tissue [69], ligament/tendon [70] and pig aortic valve [71]. This approach assumes material behavior can be decoupled into two effects: a time-independent elastic response, and a linear viscoelastic stress relaxation response. These models can be determined separately from the experiments. The stresses in the tissues, which may be linear or nonlinear, are linearly superposed with respect to time.

The three-dimensional constitutive relationship in the framework of QLV is given by,

$$S(t) = G(t)S^e(0) + \int_0 G(t-\tau)\frac{\partial S^e(E(\lambda))}{\partial \tau}d\tau \tag{6.5}$$

where S(t) is the second Piola-Kirchhoff stress tensor and G(t) is called the reduced relaxation function. $S^e(E(\lambda))$ is called the pure elastic response of the material and can be nonlinear or linear. The reduced relaxation function G(t) is a scalar function of time and can be often expressed by the Prony series,

$$G(t) = G_0\left(1-\sum_{i=1}^{N}\overline{g}_i^P\left(1-\exp\left(-\frac{t}{\tau_i}\right)\right)\right) \tag{6.6}$$
$$G_0 = G(0)$$

where $\overline{g}_i^P$'s are the Prony series parameters.

For the nonlinear elastic response, we have used an incompressible hyperelastic material representation, which is commonly used for elastomer modeling. The material properties of a hyperelastic material can be determined by a strain energy function W. Ideally, W is defined with only as many parameters as are required in order to make a FE model. There are many specific material models that could be used, depending on how to approximate the strain energy function. We selected two models: the neo-Hookean model and the Mooney–Rivlin model, both of which are widely used in soft tissue simulations.

The strain energy function of the three-dimensional incompressible Mooney–Rivlin model is given by [72]

$$W = C_{10}(I_1 - 3) + C_{01}(I_2 - 3) \tag{6.7}$$

where C_{10}, C_{01} are material parameters (having unit of stress) and I_1, I_2 are principal invariants. If we neglect the dependency of the second invariant term, we can obtain another widely used material model named the neo-Hookean model.

$$W = C_{10}(I_1 - 3) \tag{6.8}$$

A detailed description of the material models and formulation can be found in the review paper by Boyce [72].

Since the analytical solution considering the above material law and experimental conditions is very difficult, the Finite Element Method (FEM) [37] has been widely used in simulation. Through modeling of indentation experiments with the QLV approach, the final outcome of the FE simulation can be simply expressed as,

$$\begin{aligned} F_s &= FEM\_SIMULATION(p_i) \\ p_i &= [\tau_i, g_i, C_{ij}] \end{aligned} \tag{6.9}$$

where F_s is the simulated force and p_i is the material parameter containing the viscoelasticity and nonlinear elasticity. The goal of the characterization is to determine these parameters for a proposed material law by minimizing the errors between the simulated and the associated experimental measurements. This process is also called the inverse calculation because it is the opposite of an ordinary simulation (that is, solving for forces or displacements given material parameters and boundary conditions).

Instead of estimating all required parameters in one step, we separated the characterization process into two stages. In the first step the viscoelastic parameters were determined from the normalized force responses against ramp-and-hold indentation from the experiments. With the viscoelastic parameters estimated in the first stage, the inverse FEM parameter estimation method was used to determine the remaining elastic parameters. The validity of estimating these parameters separately has been investigated by Richard *et al.* [73]. They found that the stress relaxation response could be determined without regard to the particular form of the constitutive law.

This separation of parameters is also similar to the work by Kauer *et al.* [68]. However, they fixed the time constants of the stress relaxation (0.1s, 1s, 10s, and 100s), and fit their magnitudes, then determined the rest of the parameters using an inverse FEM calculation. In our work, the parameters in the viscoelastic model were estimated directly from the normalized force-displacement data in the experiments. The overall procedures of the characterization are illustrated in Figure 6-6. The advantages of this separation will be discussed in the following sections.

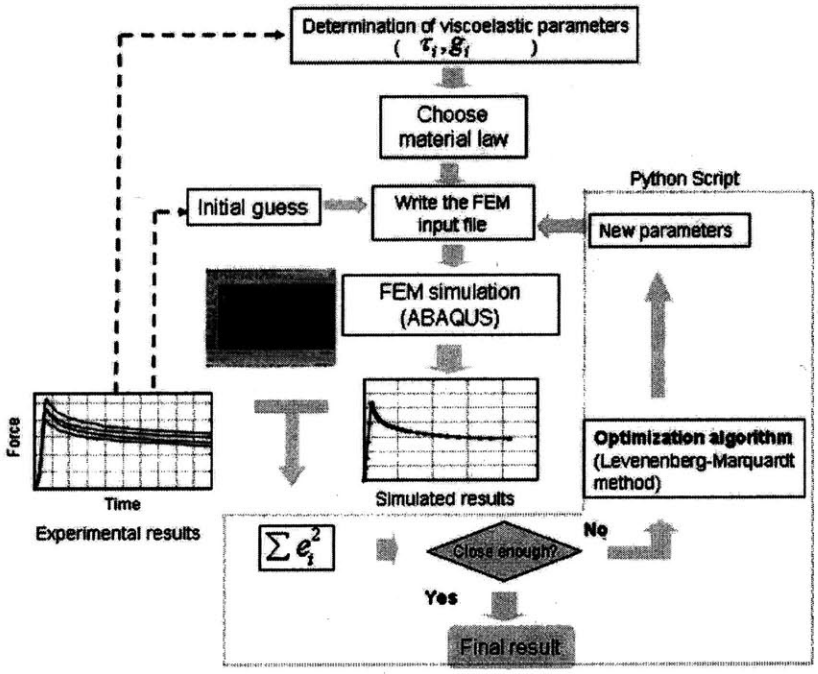

Figure 6-6 Flow chart for the inverse FEM parameter estimation algorithm

6.4.2 Determination of viscoelastic parameters

In this section we develop a three-dimensional linear viscoelastic model of the soft tissue based on the force-displacement experiment data. It has been successfully used for the modeling the human fingerpad [74].

The simplest lumped parameter model that can capture the viscoelastic behavior of a solid is the three parameter linear solid model whose transfer function may be written as

$$\frac{F(s)}{\delta(s)} = F_{ss}\prod_{i=1}^{n}\left(\frac{1+\alpha_i\ \tau_i s}{1+\tau_i s}\right) \tag{6.10}$$

where $F(s)$ and $\delta(s)$ are the Laplace transformed force and displacement variables, F_{ss} is the steady state value of the force response. τ is the relaxation time constant and α is the ratio of the initial response of the system to a step in displacement to the steady state value ($\alpha>1$). Incidentally, this model has the similar to the Kelvin model. The point indenter is a good choice for model parameter estimation since it is the closest approximation of a punch on an elastic half space. From linear elastostatics we know that the total force, P, required to indent a frictionless circular cylindrical punch, of radius 'a', into an isotropic elastic halfspace, by a distance D is given by

$$P = \frac{8aG(G+3K)}{4G+3K}D \tag{6.11}$$

where G is the rigidity modulus and K is the bulk modulus.

The correspondence principle [75] may now be invoked to obtain the corresponding quasistatic viscoelastic solution of a flat-faced circular cylindrical punch indenting a viscoelastic halfspace,

$$P(s) = \frac{8as\overline{G}(s)(\overline{G}(s)+3\overline{K}(s))}{4\overline{G}(s)+3\overline{K}(s)}D(s) \tag{6.12}$$

where $\overline{G}(s)$ and $\overline{K}(s)$ are the step response rigidity and bulk moduli, respectively.

The implication of assuming an almost incompressible tissue is that the bulk modulus is very large compared to the rigidity modulus. Hence, the material primarily relaxes in the

deviatoric mode. This assumption allows us to simplify the above equation as

$$P(s) = 8as\,\overline{G}(s)D(s) \tag{6.13}$$

From the above equations, the step response rigidity modulus may be written as,

$$\overline{G}(s) = \frac{F_{ss}}{8as}\prod_{i=1}^{2}\left(\frac{1+\alpha_i\tau_i s}{1+\tau_i s}\right) \tag{6.14}$$

Using the above model and the experimental data, we can find g_i, τ_i by using lsqnonlin.m, which is a built-in function for the nonlinear curve fitting in MATLAB (www.mathworks.com). This approach allows the rigidity modulus to be expressed as a Prony series expansion in the time domain. Table 6-3 lists the computed Prony series parameters of the selected experiments. The Prony parameter can be used directly in the representation of viscoelastic modeling in many FEM packages (see Figure 6-7).

Table 6-3 Prony series parameters and relaxation time constants from the normalized force data on liver and kidney of pig with a squared norm of errors

Organ	**Depth (mm)**	τ_1 (sec)	τ_2 (sec)	g_1	g_2	$\sum\varepsilon^2$
Liver	6	1.537	6.090	0.2866	0.2022	0.0754
Kidney	5	0.741	6.171	0.2054	0.2725	0.0653
Kidney	4	0.747	6.021	0.2305	0.2105	0.0867

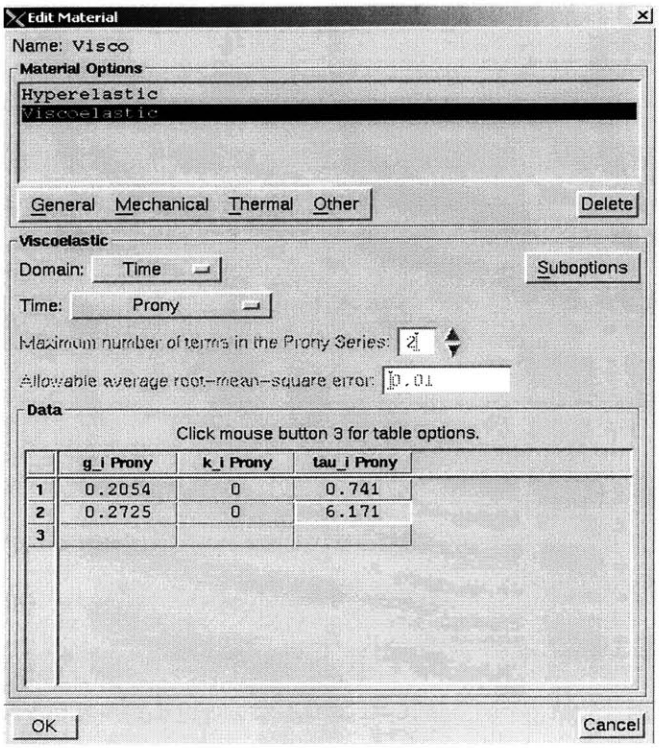

Figure 6-7 Parameter input panel in ABAQUS. Using this panel, the viscoelastic parameters can be integrated in FEM simulation for further analysis.

6.4.3 Inverse FEM parameter estimation using the optimization algorithm

In our application, the compared quantities are the simulated forces from the FEM simulation and the associated experimental forces at the indenter. Therefore, we can minimize a nonlinear sum of squares given by

$$E = \sum_{i=1}^{m} (F_s(t_i) - F_e(t_i)) \bullet (F_s(t_i) - F_e(t_i)) \qquad (6.15)$$
$$t_i = (t_1, t_2, \cdots t_m)$$

where Fe, Fs, ti, m are measured forces, simulated forces, time and the total number of data points, respectively. Among several optimization algorithms that could be used, we adopt the nonlinear least square optimization known as the Marquardt-Levenberg algorithm [76].It updates the parameters iteratively depending on the norm of $J^T J$ and

the Marquadt parameter λ.

$$
\begin{aligned}
&H = (J^T J + \lambda I)^{(i)} \\
&\triangle \vec{p} = -H^{-1}[J^T \cdot (f_s(\vec{p}^{(i)}) - f_e)] \\
&\triangle \vec{p} = \vec{p}^{(i+1)} - \vec{p}^{(i)} \\
&J = [\frac{\partial F_s}{\partial p_1}, \frac{\partial F_s}{\partial p_2} ...]
\end{aligned}
\tag{6.16}
$$

where $\vec{p}^{(i)}$ is a vector containing the estimated parameters and J is the Jacobian vector for the corresponding iteration (i). Because the parameters are contained implicitly in the FEM simulation, the Jacobian vector J should be computed numerically with respect to each parameter variation. This procedure may be repeated until the difference becomes smaller than a certain tolerance value.

Although the Levenberg-Marqudt has been used successfully in finite strain applications [77] [78], the entire process is computationally expensive. The FEM simulation, in which a few hours per run, is repeated as many times as necessary until the simulated results match the experimental results. The Jacobian vector is calculated at each iteration but this requires perturbing one parameter, running the entire FEM simulation, and measuring the effect of the perturbation. To construct the vector, this must be repeated for each parameter. Thus, for five free parameters, the FE model must be solved six times per iteration (the sixth solution is the reference to which the perturbations are compared). The entire characterization process takes several iterations to converge so the computational time is very large and hence it is better to reduce the number of parameters as much as possible as in our approach.

It was this computational expense that led us to identify the viscoelastic parameters before undertaking the inverse FEM simulation. This allowed us to iterate using only one (neo-Hookean) or two (Mooney-Rivlin) free parameters. This approach also avoided potential numerical errors due to poor scaling of the Jacobian that would have arisen due to the 10,000 to 100,000-fold difference between the relaxation time constants and elastic constants.

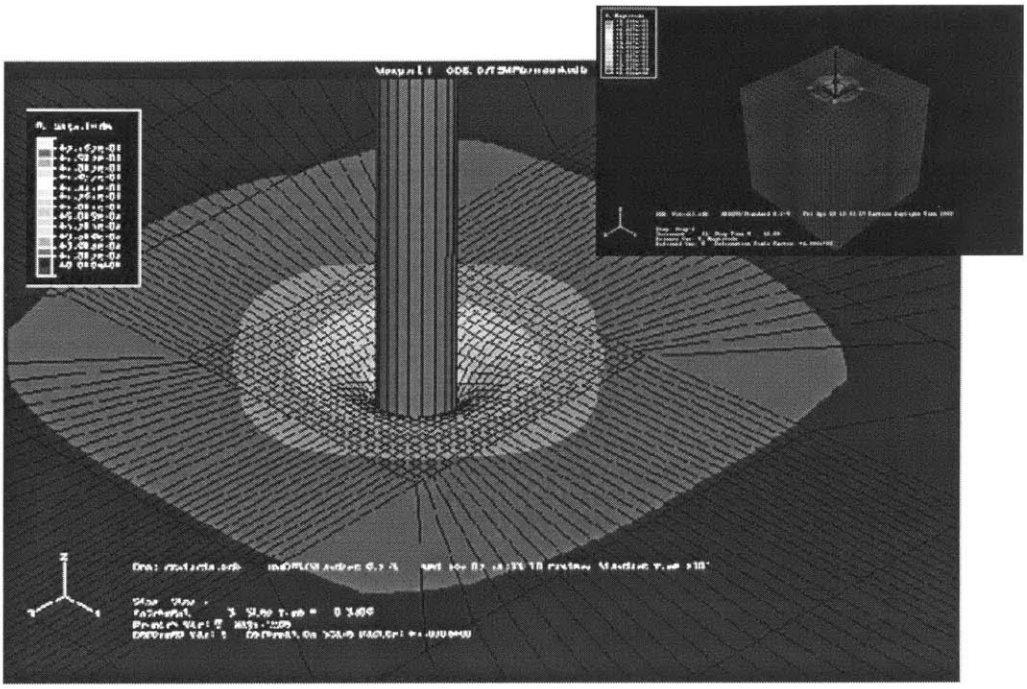

Figure 6-8 FEM simulation of the experiment developed with ABAQUS. It uses 4820 nodal points and 4400 eight nodded hexagonal elements. The dimension of the model is 100 mm x 100mm x 100mm. It predicted the forces at the instrument considering nonlinearity and contact mechanics. The color represents the stresses around the tool-tissue contact area.

Figure 6-8 shows the developed model in ABAQUS/Standard Version 6.3.1, a commercial finite element software produced by Hibbit, Karlsson & Sorensen, Inc.. In our experiments we noted that the deformation field appeared to be insensitive to the organ geometry. This made sense, since our indentations were small (millimeters) compared to the organs (centimeters). Accordingly, we simplified our analysis by modeling a sub-domain of the organ. The size and the mesh density of the model are carefully adjusted to ensure a well-conditioned solution. We used a set of eight nod hybrid brick elements provided by the ABAQUS mesh generator. Nonlinearities from both the material model and from large geometric deformation were allowed. The contact between the indenter and tissues was modeled with a contact mechanics module in ABAQUS and the non-uniform element density over the model was used to improve accuracy of the contact region.

We implemented the Levenberg-Marquardt algorithm with the Python language, which is a way to control the inputs and outputs of ABAQUS. After the selection of

initial parameters and the first simulation, the program reads the output files and computes the next parameter set. Then it generates new input files with the updated parameters and starts a new simulation. The program is synchronized with the FEM simulation so it repeats automatically until the algorithm finds out the parameters matching the simulated and experimental forces closely enough.

6.4.4 Numerical simulation results

Two force measurements were selected to test the algorithm. The viscoelastic parameters were estimated from the normalized force response and put into the ABAQUS database for the viscoelastic modeling. Due to the requirement of a reasonable initial guess, the mean static force response measurement in the experiments was used. We inputted these static responses of the organs to the simulation using the "Evaluate material function" which treated them (incorrectly) as the results of a uniaxial test with uniform strain,

$$\sigma_{eff} = 2C_{10}(\lambda - \frac{1}{\lambda^2}) \quad \text{(neo-Hookean material)} \tag{6.17}$$

$$\sigma_{eff} = 2(C_{10} + \frac{C_{01}}{\lambda})(\lambda - \frac{1}{\lambda^2}) \quad \text{(Mooney-Rivlin material)} \tag{6.18}$$

where σ_{eff} is effective stress and λ is effective compression ratio.

With these approximate initial values in hand, we used our Python code to iterate the FE model and update the parameters automatically. The parameters reached convergence with four or five iterations. To test the convergence of the algorithm, we tried the FE simulation starting from a range of other values. The algorithm converged to the final parameters provided the initial guess was within about 250% of the value we used. Deviations greater than 200% led to no convergence or convergence to physically meaningless (e.g., negative) values. Table 6-4 and Table 6-5 present the initial parameters and converged parameters for both hyperelastic material models. With a good guess from a priori knowledge of the parameters, the parameters converged with three or four iterations in most of cases as shown in Figure 6-9 (Neo-Hookean) and Figure 6-10 (Mooney-Rivlin).

Table 6-4 Initial and estimated parameters in the neo-Hookean material model

Experimental Condition		Initial parameters	Number of iterations	Final parameters	$\sum \varepsilon^2$
Target organ	Indentation Depth	C10 (Pa)		C10 (Pa)	
Kidney	5mm	2387.3	3	1872.74	0.005406
Liver	6mm	198.23	4	452.92	0.003312

Table 6-5 Initial and estimated parameters in the Mooney-Rivlin model

Experimental Condition		Initial parameters		Number of iterations	Final parameters		$\sum \varepsilon^2$
Target organ	Indentation Depth	C10 (Pa)	C01 (Pa)		C10 (Pa)	C01 (Pa)	
Liver	6mm	83.18	84.76	3	322.96	161.47	0.00285
Kidney	5 mm	682.31	700.02	2	868.66	467.11	0.00314

Figure 6-11 shows the predicted forces from the FE simulation with the estimated parameters and experimental forces for the selected experiments of pig liver and kidney. The force responses of the hyperelastic model in ABAQUS match the experimental data well. Because it is hard to present visually which model can predict the forces better between neo-Hookean and the Mooney–Rivlin model, we plot the static responses of the pig liver against static loads with two material models we used in our work in Figure 6-12 using the estimated parameters. Both models perform well in a low indentation range but the Mooney-Rivlin model shows better results than the neo-Hookean model in a higher deformations.

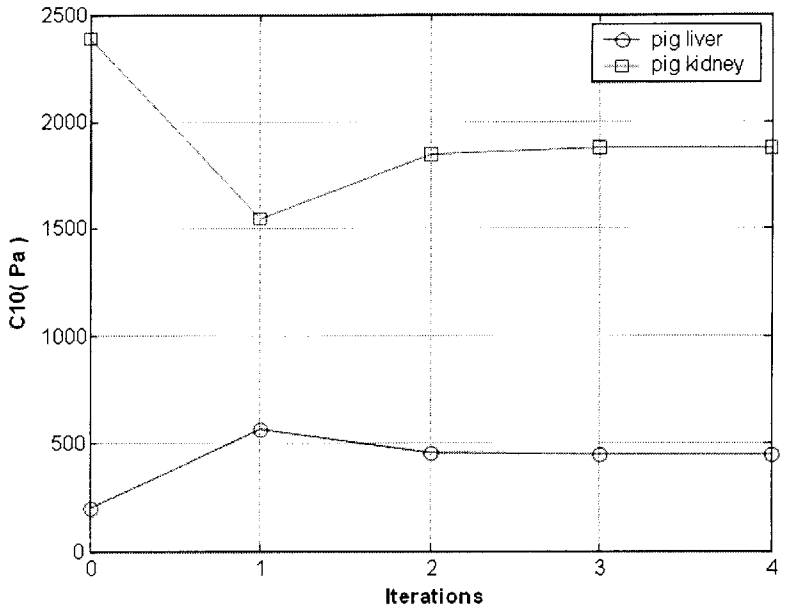

Figure 6-9 Convergence of the material parameters C10 for the neo-Hookean model for pig kidney and liver

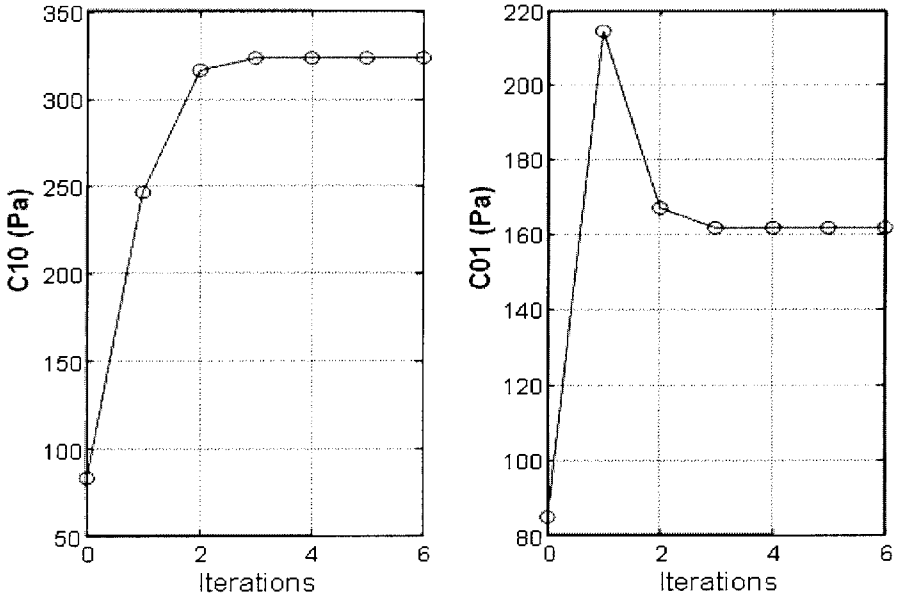

Figure 6-10 Convergence of the parameters of the Mooney-Rivlin model for pig liver

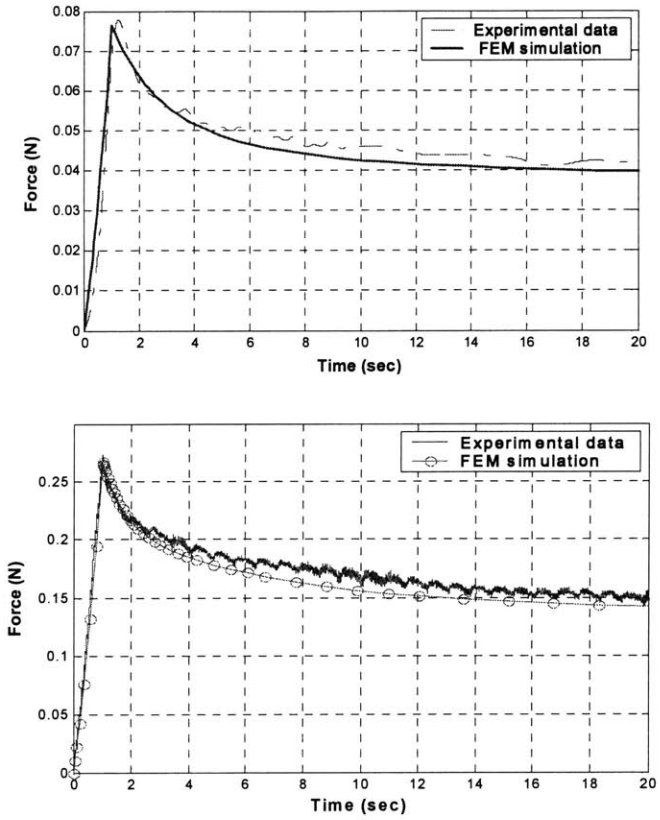

Figure 6-11 Force responses of the FE simulation and the experiments. (Upper) liver with 5mm indentation. (Lower) kidney with a 6mm indentation. The data from the experiment were filtered out to remove noisy properties by the 3nd order butterworth filter. The responses in (b) shows this noisy signal from the experiments.

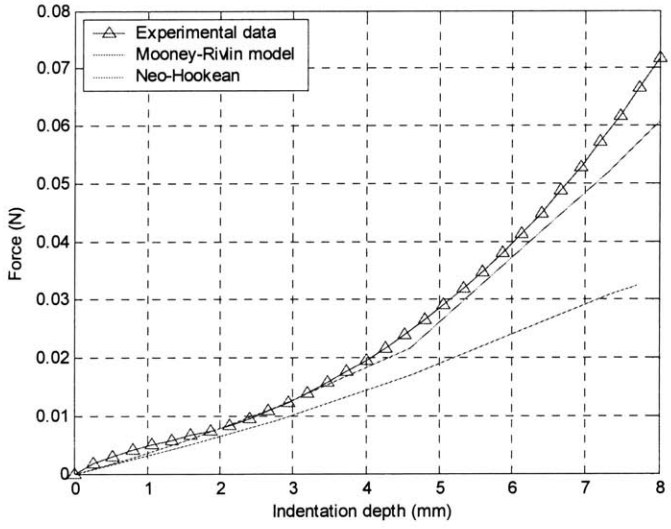

Figure 6-12 Static force responses of a pig liver and predicted responses by the FE simulation. Parameter values used in this figure are C10 = 198.23 for the neo-Hookean model , C10 = 322.96 and C01 = 161.48 for the Mooney-Rivlin model.

6.4.5 Special case

It is important to note that the modeling approach presented so far is appropriate only for solid organs such as liver or kidney. These kinds of tissues consist of dense connective collagen tissues and can be treated as approximately isotropic and homogenous [56]. However, hollow organs such as the esophagus must incorporate geometrical information in the simulation, so we should incorporate information of organ geometry and boundary conditions. Figure 6-14 shows the modeling of esophagus structure. We modeled the esophagus as a tubular structure 60 mm in length. Because the abdominal esophagus is tightly connected with the diaphragm at the upper end and the fundus (the upper part of stomach) at the lower end as shown in Figure 6-13, both ends can be modeled as fixed ends. A 30mmHg internal pressure is also modeled in the simulation. This information is obtained from the medical publications [79] [80]. Figure 6-15 shows the simulated response of the esophagus. The Young's modulus of the esophagus was estimated from the inverse FEM modeling to be 5.222kPa. The relaxation time constant and the 1st order of Prony parameter are 6.372 and 0.363sec, respectively.

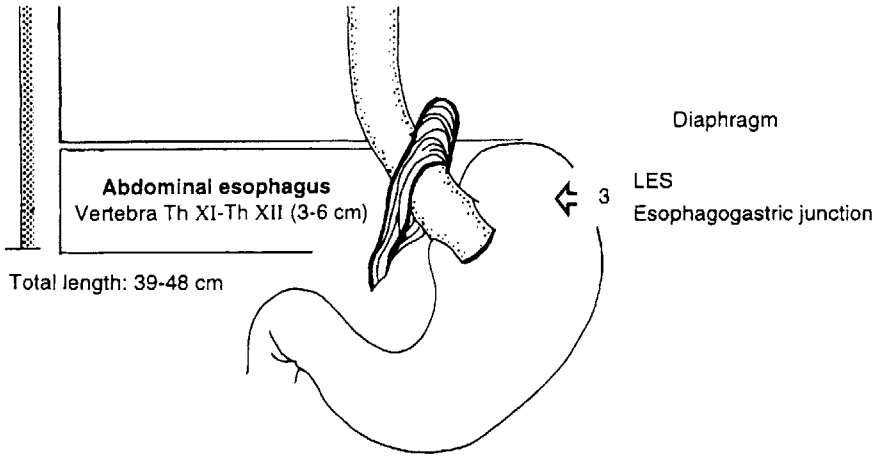

Figure 6-13 Esophagus and its anatomical location

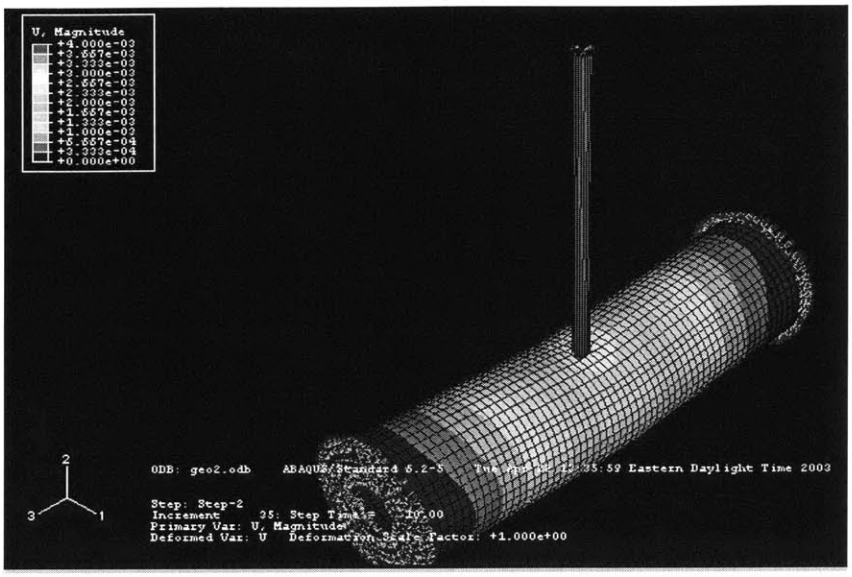

Figure 6-14 FEM modeling of esophagus muscle tube. The inner and outer diameters came from the Visible Human Datasets

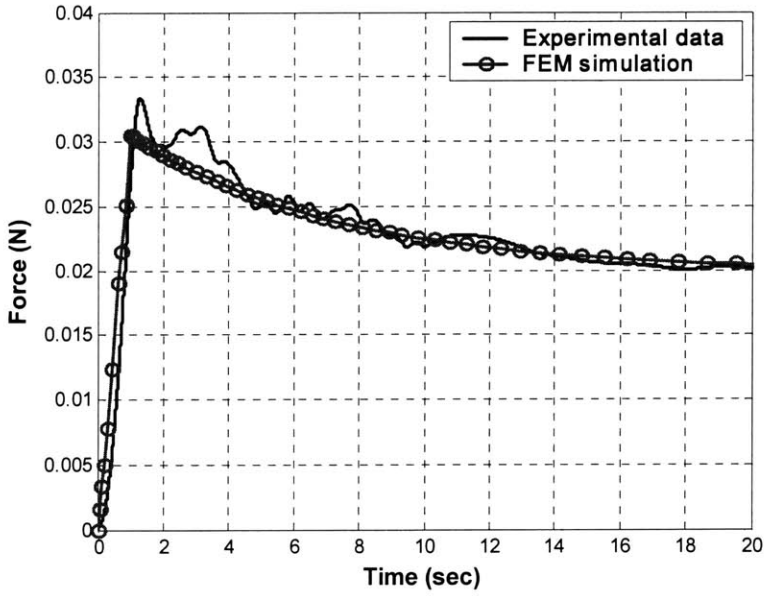

Figure 6-15 Simulated and experimental responses of the esophagus indentation considering geometry

6.4.6 Limitations

Another limitation of the approach is uniqueness. Like other applications that use optimization, it is hard to guarantee the uniqueness of the solution due to the locality of the algorithm. One of the possible solutions to this difficulty, which we used, is to optimize from a range of initial guesses, and to check to see that the final estimate is physically plausible. *Ex vivo* tests of organ properties remain helpful for making the initial guess and checking the final estimate. If the initial guesses are chosen without considering *a priori* knowledge, the algorithm may generate a physically unacceptable solution and lead to failure to converge in the FE simulation. Therefore accumulating as much knowledge as possible is still important in order to reach a physically meaningful solution as well as reducing the number of iterations to achieve convergence. Another important issue is the sensitivity of the algorithm to noise from the various sources in the experiments. Moulton et al. [78] investigated the noise sensitivity of the Levenberg-Marqudt algorithm by adding various noise levels to the measurements. They showed that the variability of the parameters is of the same magnitude as the noise in the measurement.

6.5 Discussion

In this chapter, we have characterized the mechanical properties of intra-abdominal organs from the *in vivo* animal experiments by using three different methods. First, the Young's moduli of the tissues were estimated based on the assumption of a linearly elastic, semi-infinite body. Second, the parameters in lumped parameter models were determined using curve fitting techniques. Finally, the algorithm for the most comprehensive continuum mechanics model is developed. In the framework of the QLV model, we estimated the viscoelastic and hyperelastic material parameters in two stages. To calibrate the parameters to the experimental results, we developed a three dimensional FE model to simulate the forces at the indenter and an optimization program that updates new parameters and runs the simulation iteratively.

Key assumptions in our approach are that the organs are incompressible, homogenous, and isotropic, and that the deformations we imposed were small compared to the size of the organ. The inverse FEM technique can be extended easily to include

more general features of soft tissue behavior if necessary. The approach may be extended to increasingly complex organs by building layered organ models and estimating the material properties of each layer with this algorithm.

With these limitations in mind, the material models presented in this study offer two basic uses in a VR-based medical simulation. First, they can be used directly in the simulator to compute visual deformations and interaction forces that are displayed in real time. Although real-time computation of complex tissue behavior is challenging, there is some encouraging work [32]. Second, the mathematical models presented here can be used as a standard for the evaluation of new real time algorithms for computing deformation [61].

Chapter 7 Example: Development of a Surgical Simulator for Esophageal Procedures

From chapter 3 through chapter 6, we have presented two important aspects in modeling of tool-tissue interactions: the real time tissue model and the characterization of in *vivo* tissue properties. These techniques are integrated to build an initial demonstration of a surgical simulator, which when fully developed could help laparoscopic surgeons to practice specific medical procedures.

In the following section, we will mainly discuss topics not coved by this thesis but which are important components to build a surgical simulator. In section 7.1, we will introduce anatomical information of the lower esophagus. The hardware setup and modeling of the haptic device are followed in section 7.2 and 7.3. The details of building organ geometry and surface texture are explained in the next two sections. In section 7.6, we present haptic rendering algorithm used in this demonstration. The progressive cutting algorithm is presented to simulate the cutting of the esophagus skin in section 7.7. Finally, the results are presented in the last section.

7.1 Esophagus and Related Procedures

The esophagus is the tubular organ connecting the pharynx to the stomach and is responsible for the transport of the bolus (chewed food in the alimentary canal) to the stomach. Two different layers of muscles form the exterior of the esophagus. The outer

muscle layer is aligned longitudinally along the esophagus while the second layer is circumferential the esophagus. The esophagus is divided into three sections, with the upper third composed of striated muscles, the middle third a combination of smooth and skeletal muscle and the lower third entirely of smooth muscles. A layer of mucosa forms the innermost layer of the esophagus. Contractions along the esophagus in the form of peristaltic waves form the mechanism to force the bolus through the esophagus into the stomach. The esophagus is able to bulge and expand to allow the bolus to move through it.

The lower third of the esophagus lies within the abdominal cavity and as referred to as the lower esophagus. It is tubular in nature, about 15 mm in diameter, 40 mm in length and is suspended between the diaphragm and the stomach. The region of the esophagus just above the stomach is known as the esophagogastric junction or the esophagogastric vestibule. This region is normally closed tightly to prevent regurgitation from the stomach. Malfunctioning of the esophagogastric junction causes coughing and heartburn due to reflux of acid to the esophagus. The common way to treat this is a procedure called a laparoscopic Nissen fundoplication. In this procedure the weakened lower esophageal sphincter is supported by wrapping the top portion of the stomach around the lower esophagus as a valve. An alternative esophageal procedure is Heller's myotomy. Usually the failure of lower esophageal sphincter (LES) causes a serious swallowing problem. To treat this, the muscle structure should be incised to reduce the blockage pressure. The outer layer and longitudinal muscle is incised and this circular muscle structure is cut carefully. If too much force is introduced on the light reddish inner layer, perforation of the wall occurs.

To provide a functional trainer for these procedures, the pulling and cutting operation of the lower esophagus area should be simulated. Our goal in this chapter is to develop the model demonstrating cutting and pulling of the lower esophagus area.

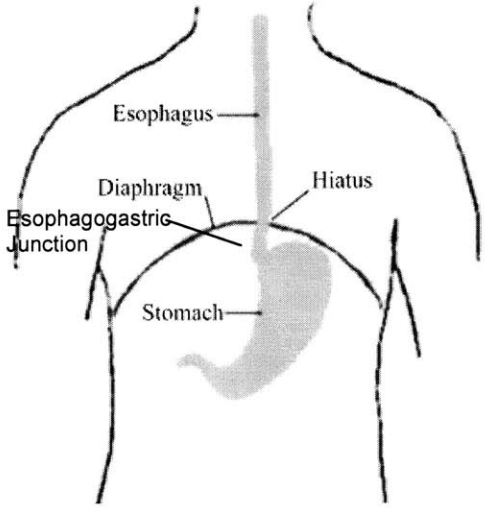

Figure 7-1 Esophagus and neighborhood organs. We are interested in the lower part of esophagus, connected to the upper part of stomach.

7.2 Hardware Setup

Figure 7-2 shows the final surgical simulation setup built in the TouchLab/MIT, which was originally built for measuring training effectiveness [16]. The setup includes the rubber abdominal wall, with surgical instruments and trocars inserted at a similar position to those in real operations. Underneath the abdominal wall, the surgical tools are connected to Phantom (SensAble Technologies) haptic interface devices. The Phantom devices were connected to a Pentium III PC for this setup. Laparoscopic surgeons attested to the fact that forces by the abdominal wall at the trocar are the dominant forces during surgery, much greater than the smaller forces between the tools and the organs. It would be technically challenging to simulate both the abdominal wall forces and contact forces at the same time. Therefore, a rubber abdominal model was used so that the modeling only needed to be done for the forces between the tools and the intra-abdominal structures. This system was found to be an effective setup for a laparoscopic simulation, under the assumption that the purchased rubber abdominal model is a model of reproduction.

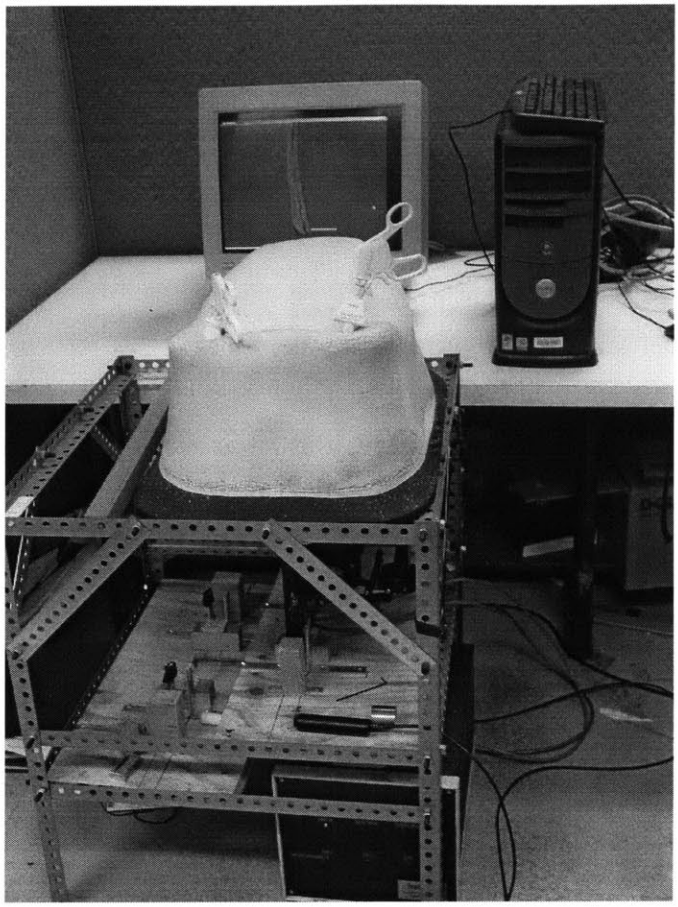

Figure 7-2 Simulator setup

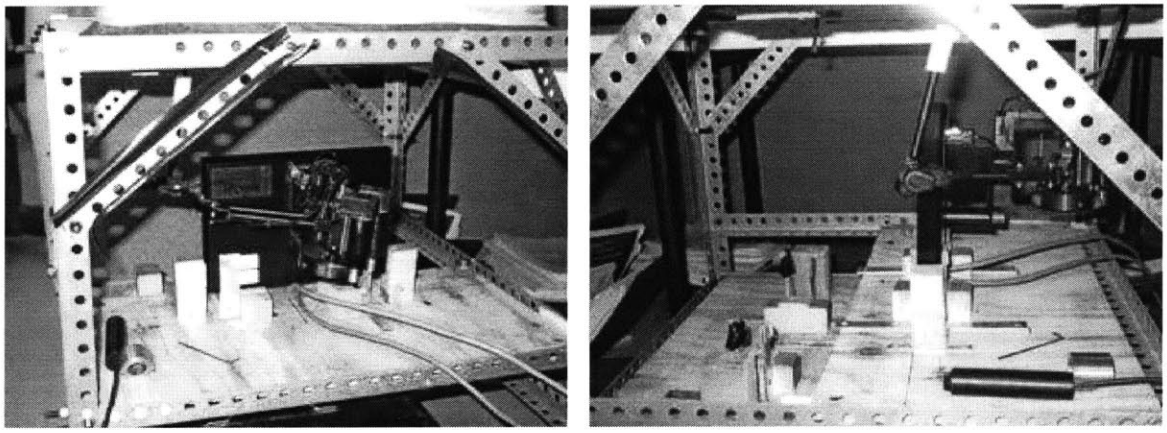

Figure 7-3 A phantom (a 1.0A model) connected to the end of the surgical tool.

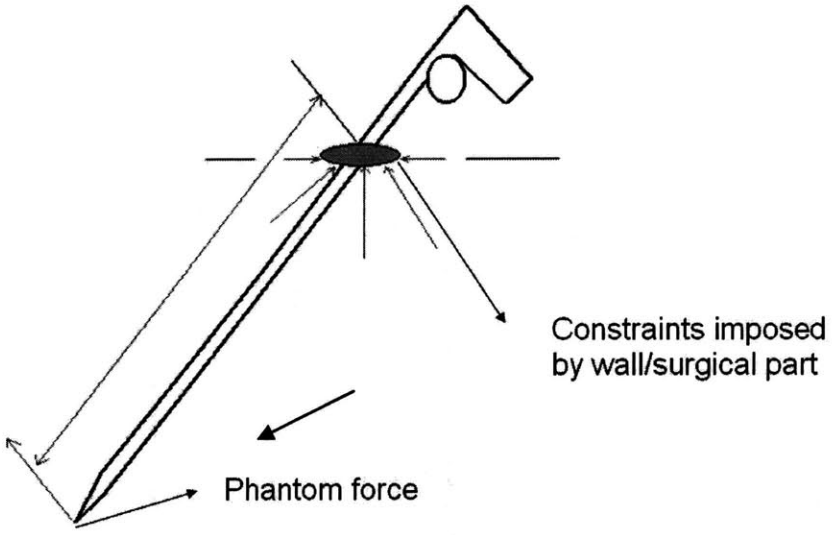

Figure 7-4Force applied by "virtual" forces from the Phantom and "real" forces from the abdominal wall [16].

7.3 Modeling of a Haptic Device

The transparency [81] of the haptic device is a very important performance measure in order to understand bi-directional interactions between a human operator and a virtual environment. Unlike visual interactions, the haptic interactions involve energy exchange between two sides through the device. If the device is not transparent enough, the force information may degenerate or be distorted.

We have characterized the phantom dynamics to check the transparency of the phantom within our working range (below 8 Hz) with a two–port network model which is a widely used tool describing stability and performance in bilateral teleoperation [82].

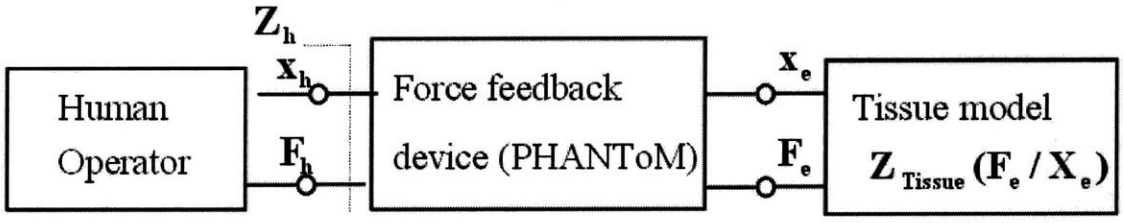

Figure 7-5 A two-port model for haptic interactions consisting of a human operator and a virtual environment.

Transparency can be restated as the quality in which velocities and forces are passed between the human operator and the virtual environment [81]. The haptic interaction can be represented with a two port model as in Figure 7-5, in which a "black box" relates between force and position (velocity) at the two terminal points [81].

The transfer function matrix between the human and the virtual environment can be given,

$$\begin{bmatrix} F_h \\ x_e \end{bmatrix} = \begin{bmatrix} G_1 & G_2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_h \\ F_e \end{bmatrix} \tag{7.1}$$

The impedance function Z_h can be computed from the transfer function matrix and it determines the transparency of the haptic device.

$$Z_h = \frac{F_h}{x_h} = (G_1(s) + G_2(s) Z_{tissue}) \tag{7.2}$$

where G_1 is called a input impedance and G_2 is a force transmission ratio. They are frequency dependent functions. Since Z_h is equal to unity in the case of a perfect transparent device, the conditions for the transparency are G_1 is very small and G_2 is close to unity within our working frequency range. If the conditions are fulfilled, the impedance that the human feels can be given as,

$$z_h = z_{\text{tissue}} \qquad at \ f < f_{\text{working\_limit}} \tag{7.3}$$

To estimate G_1, G_2 in the working frequency range (0 - 8 Hz), we developed models of the Phantom T model. For the input impedance G_1, the relationship between the force and the position at the human side is modeled as the second order system as shown in Figure 7-6 (left).

$$G_1 = \frac{F_h}{x_h} = ms^2 + bls + kl \tag{7.4}$$

For the force transmission ratio G_2, the mechanical structure of the phantom can be modeled as a connected two-spring system (see Figure 7-6(right)). The transfer function of G_2 is given as,

$$G_2 = \frac{F_h}{F_e} = \frac{k_l k_h}{J_m J_l s^4 + (J_m k_c + J_l k_l + N^2 J_m k_l)s^2 + k_l k_h} \tag{7.5}$$

where J_m, J_l, k_l, k_c, N are the inertia of the motor, the inertia and stiffness of the link, the stiffness of the soft tissue, and the transmission ratio, respectively.

With the numerical values, the two transfer functions are given as

$$\begin{aligned} G_1 &= 9.641\times10^{-5}s^2 + 0.0026s + 0.032 \\ G_1 &= \frac{3.9\times10^5}{3.97\times10^{-9}s^4 + 13.71s^2 + 3.9e^5} \end{aligned} \tag{7.6}$$

For more accurate measurements, we measured G_1, G_2 experimentally by using sinusoidal input functions as shown in Figure 7-7 and Figure 7-8. From these plots, we can see that the phantom can fulfill the transparency conditions we mentioned earlier. The limitation of this model is that we do not consider the instruments connected to the phantom. If we include this part in the modeling, it would provide a better representation of the system dynamics.

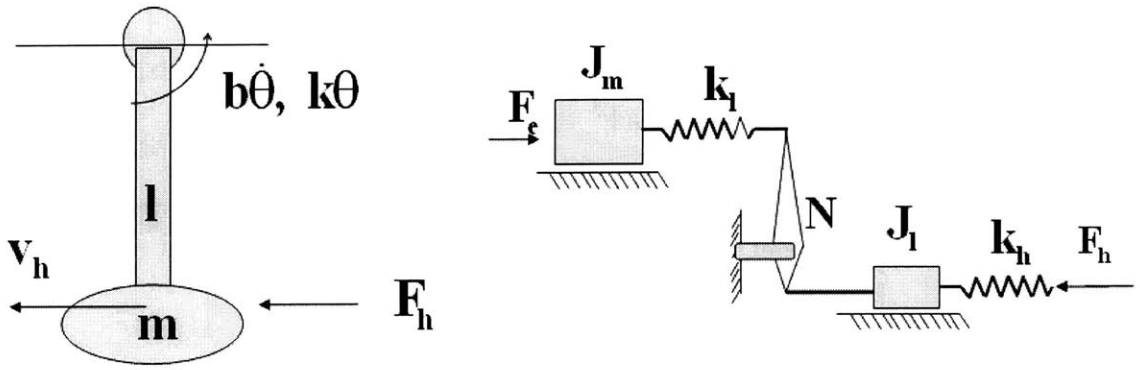

Figure 7-6 Lumped parameter model for the input impedance (Left) and the force transmission ratio (Right). Numerical values are obtained from the paper by [83] or measured.

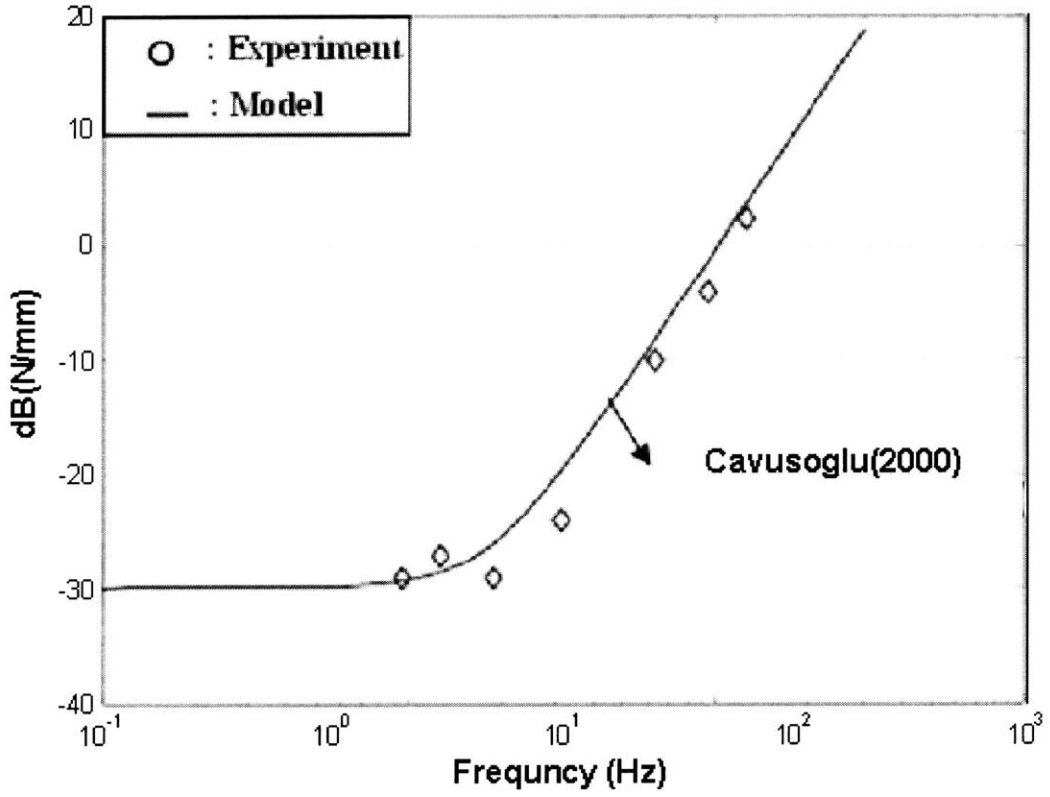

Figure 7-7 Frequency responses of the input impedance function. Under a low frequency region (100 rad/sec = 15 Hz), the amplitudes are around 30dB, which is less than 3.1%. The circles represent experimental measurements while the line is represented by the model prediction.

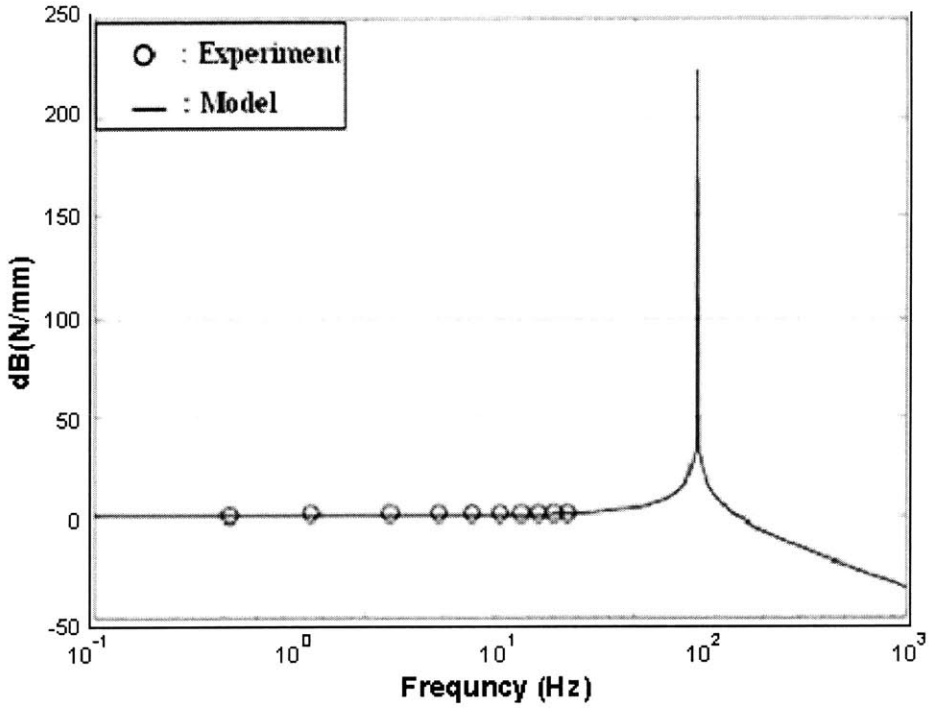

Figure 7-8 Frequency responses of the force transmission function. Under the lower frequency, the ratio is very close to unity.

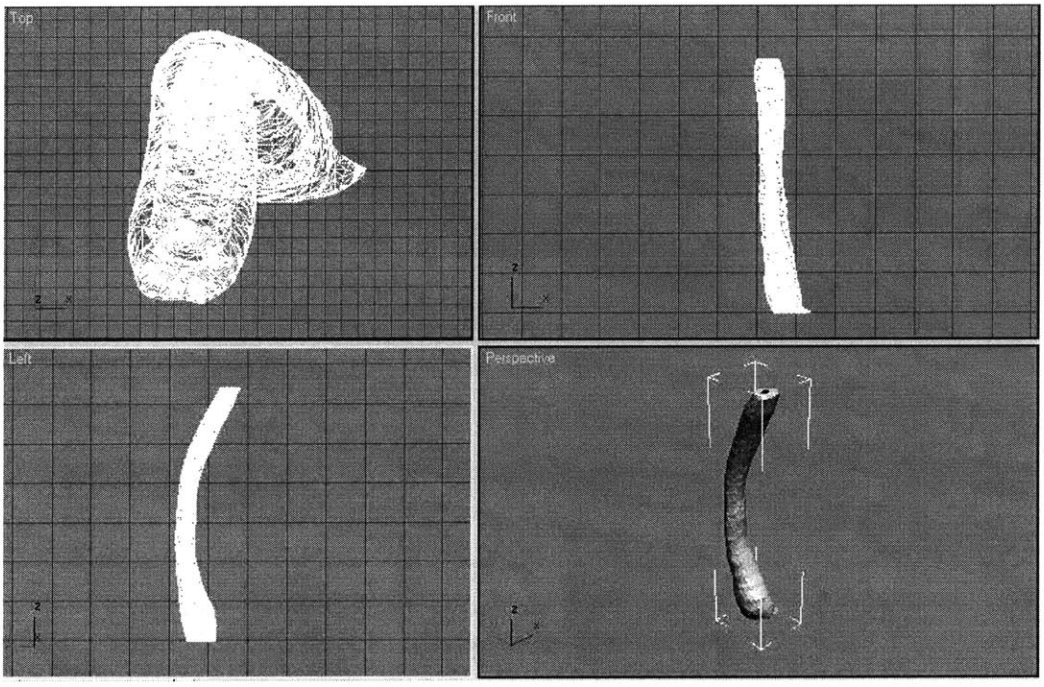

Figure 7-9 Segmented esophagus model from the Visible Human Production (http://visiblep.com). It is opened with the 3D Studio Max. From the top left in a counterclockwise manner, the top view, the front view, the perspective view and left view, respectively. (Images are not in the same scale)

7.4 Organ Geometry

We used the segmented esophagus model from the Visible Human Dataset as shown in Figure 7-9. Because the original model from the Visible Production had too much surface detail, we simplified the model in 3D StudioMax by Discreet (www.discreet.com) as shown in Figure 7-10. The original model contains 10658 vertices and 21146 triangles. The simplified model has 1836 vertices and 3726 triangles.

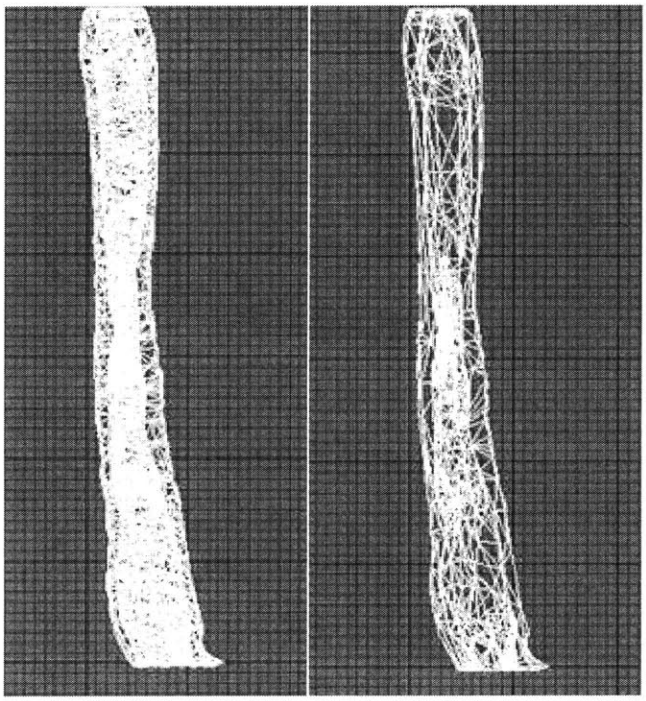

Figure 7-10 Comparison of the esophagus models before the simplifying operation. Before (Left) and after the operation. (Right) The operation was performed in 3D StudioMax.

7.5 Texture Mapping of Anatomical Objects

Since the graphic model is based on a mesh representation in current graphic hardware, high resolution texture maps have been used to visualize surface details in the simulation without much computing cost. However, texture mapping [84] on the surface of the anatomical model is not a trivial problem because the underlying geometry is not planar or rectilinear. Figure 7-11 shows the concept of the texture mapping.

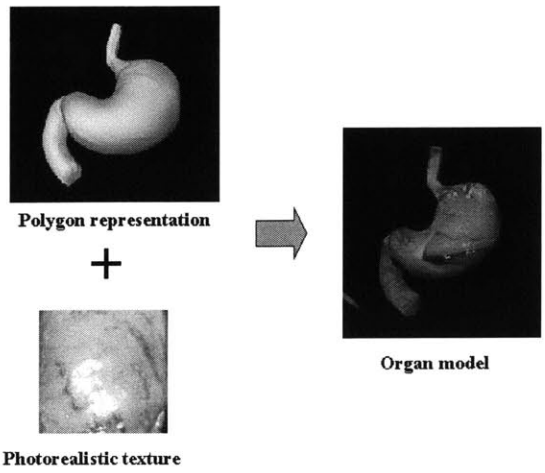

Figure 7-11 Illustration of texture mapping. It adds realism to the model with less computational complexity.

Environment mapping is an efficient technique to compute the reflection vectors without adding real time computations. Although this technique was developed around the 1970s [85], it only became available in PC level workstations recently due to advances in computer graphics hardware. In general, human beings perceive the properties of surfaces by the degree to which light is absorbed and reflected. To simulate this fully, complicated modeling of light is required to specify the direction of secondary rays from the surfaces. Some realistic effects, such as glistening, require this computation and are difficult to simulate in real time surgical simulation due to the computational burden. The environment mapping approach uses a simple form of reflection-vector dependent texturing and pseudo-ray tracing computations. In other words, by projecting 3D environments surrounding the object onto a 2D environment map, reflections can be approximated with some degree of accuracy, which significantly reduces real time computations. Environment mapping has several advantages over traditional texture mapping techniques. First, the connectivity on a projected texture is maintained, because normal vectors change smoothly from one texture to another. Second, it is not necessary to flatten the whole surface of the anatomical object. Just one triangle is flattened at a time, rather than flattening neighborhoods. Finally, topology of the model is maintained. The local texture orientation is implicitly maintained, so that no uneven scaling or stretching occurs as in traditional texture mapping techniques.

With the current graphics hardware, two kinds of environment mapping techniques are available depending on the shape of a map: sphere mapping and cube mapping. The sphere mapping is less computationally expensive, but has view point dependency which produces warping or distortion. This section describes the performance evaluation obtained through analyses, simulation and timing measurements for the techniques. From Table 7-1 summarizing the results, we observe that cube mapping and sphere mapping using GEForce 2 shows better performance in rendering organ models with glistening effects. If we simulated the same effect using conventional techniques, it would be very difficult to achieve real time performance due to the huge computational burden of ray tracing. We also observe that the software emulated environment mapping technique is not to be recommended for rendering of complicated anatomy objects due to its poor performance as shown in Table 7-1. The graphics cards with hardware support of environment mapping are three times faster than graphics cards which do not support an environmental mapping technique in the simulation of complicated effects.

Table 7-1 Rendering Performance of Environmental Mapping. We used GEForce 2 from NVIDIA, which provides hardware-supported environment mapping features and E&S Tornado 3000 from E&S, which has only software emulated environment mapping techniques.

Object	Grid Size	Rendering capacity per second		
		Cube Map (GeForce2) MTriangles/sec	Sphere Map (GeForce2) MTriangles/sec	Sphere Map (E&S Tornado 3000) MTriangles/sec
Liver	17536	1.714	1.121	0.387
Stomach	51264	1.645	1.477	0.377
Lung	27648	1.714	1.593	0.388
Kidney	16512	1.710	1.408	0.371
Intestine	68456	1.581	1.656	0.412

We generated a surface image of the human esophagus from a real esophageal laparoscopic video and encoded it as a required format for the environment mapping. Figure 7-12 shows the three different kinds of the esophagus model, which are rendered

differently. The algorithm for the texture mapping is listed in the following table. It shows very realistic deformation and biological effects in real time. There are a few limitations to environment mapping, such as no self-reflection properties, so careful consideration for obtaining and implementing is required to apply it in surgical simulation.

Table 7-2 Algorithm for texture mapping using environment mapping

Anatomical Environment Map Algorithm

OpenGL Environment Map Setup Usage
Step 1. Binding to a environment map texture
Step 2. Loading environment map textures
 	
OpenGL Environment Map Rendering Usage
 	
Step. 1. Enabling an environment map texture
glEnable(GL_TEXTURE_ENV_MAP);
Step. 2. Generated environment map coordinates (explicit or implicit)
glTexCoord3f(vx, vy, vz); // (vx,vy,vz) is unnormalized direction vector
Step. 3. Generate anatomical object with textures

Figure 7-12 Three kinds of rendered esophagus geometrical models. (Left) Polygonal model without textures. (Middle) Model with traditional texture mapping. We can see graphic artifacts such as extension marks. (Right) Model with the sphere environmental mapping technique.

7.6 Haptic Rendering

Because the deformation and forces could not be computed without the detection of collision location and penetration vectors (see Figure 7-13), the proper selection of collision detection algorithm is also very important in surgical simulation; this section mainly discusses collision detection in haptic rendering. A good collision detection algorithm not only reduces the computational time, but also helps in correctly displaying interaction forces to the human operator to make the haptic sensing of virtual objects more realistic. There is substantial literature in computer graphics on fast determination of collision points in various cases [86, 87]. In general, the purpose of the collision detection in graphics is to avoid the overlap between objects and to simulate the behavior of objects following the overlap. In contrast, the purpose of collision detection in haptic rendering is to check collisions between the probe and virtual objects to compute the interaction forces [17]. Although they have different objectives, they share the same technical components to find a collision point in real time: 1) partition of the object into set of bounding volumes, 2) data hierarchy and search algorithms, 3) localization of the contact area for successive collision detection.

We implemented the Axis Aligned Bounding Box Hierarchy (AABB) algorithm [88] because using the simplest bounding volumes can keep update overhead as small as possible. It uses a set of three dimensional boxes with the same axis of orientation for computational simplicity. The data structure representing the bounding box hierarchy is organized as a binary tree. The leaf nodes in the tree are the boxes containing only one triangle. Afterward, two boxes of a certain level are enclosed by a box of the next higher level [89]. We also use a neighborhood watch algorithm to localize the contact point after the first collision [13]. Although global search algorithms like AABB are very efficient techniques, it is still numerically inefficient to repeat them to find successive collision points. The neighborhood watch algorithm is to build neighborhood relationships for each primitive such as lines, vertices, and polygons, and then save them into a look-up table format. The successive collision point is searched only in the local areas of the current collision point. Another important function of the neighborhood watch algorithm is to compute the progressive cutting vector for cutting simulation.

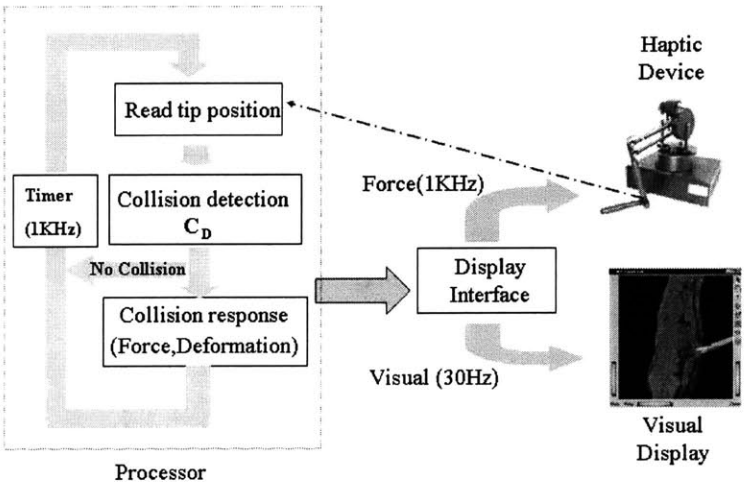

Figure 7-13 Collision detection and collision response. The simulator repeats two key processes in every time step

7.7 Progressive Cutting Algorithm

Cutting is a common procedure encountered in surgery but it is very difficult to be simulated in a physically-based way because the change in topology requires remeshing of the model and the numerical complexity associated with this such as elimination of the old model, recreation of the new model, computation of the stiffness matrix, and real time display of cut regions.

Volumetric mesh cutting schemes linked with physically based model have been investigated extensively [90]. Bielser *et al.* [91] used a lookup table to reduce the complexity involved in a creation of new elements for each cut element. Mor and Kanade [92] generated a minimal set of new elements to replace modified tetrahedral elements during progressive cutting. Nienhuy and Stappen [93] proposed a technique where the scalpel nodes are snapped to the trajectory. This method ensures that the mesh size remains small and few short edges are created. Although these are very important steps to reach the goal of accurate cutting simulation, it is still challenging for these schemes to provide enough realism for the simulation without unrealistic appearances. Besides the numerical complexity involved in the computations, each scheme suffers from visual artifacts such as bumpy appearance of the cut area or holes.

The minimum requirement of the cutting simulation is interactivity. Instead of using volumetric cutting schemes, we used a surface-based cutting technique based on

pure computers graphics techniques. It modifies graphic models following a cutting path and display visual feedback in real time as the progress of the cut shape is generated without increasing the number of polygons.

Heller's myotomy involves the cutting the lower esophagus junction down to cut the underlying muscle structure, so the simulation of cutting is essential. We implemented a progressive cutting algorithm and applied in the esophagus model. This algorithm was coupled with the collision detection algorithm and neighborhood database explained in section 7.6. It computes a progressive cutting path on the surface with a small amount of lag and detects an initial seed point of the cutting path. The flowchart is shown in Figure 7-14.

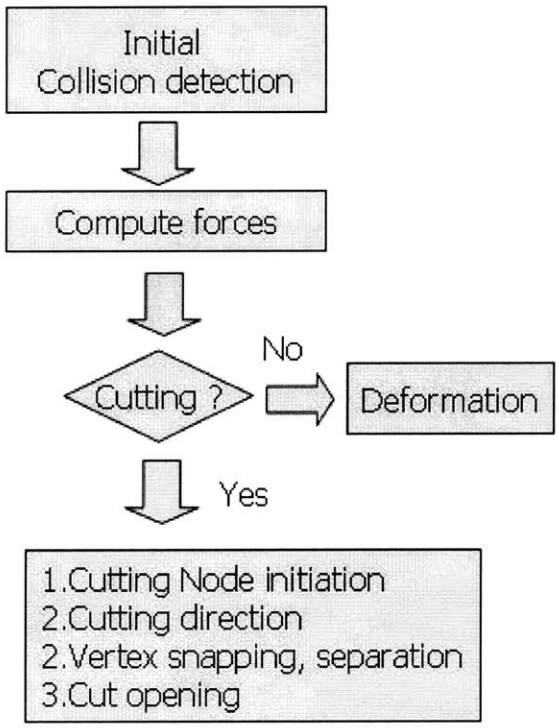

Figure 7-14 Flowchart of the progressive cutting algorithm

After initial contact between the cutting tool and the organ geometry, the nearest vertex is snapped to the contact point. As the cut progressed on the surface, the collision detection algorithm monitors the trajectory of the cutting tool. Whenever the tool crosses the one of the edges of a triangle, the successive cutting vector is computed and one of the points in

the cut edge is snapped into the path. The moved vertex is split into two vertices and they are displaced with respect to the cut opening vector, which is perpendicular to the cutting vector, to display the cutting opening as shown in Figure 7-15.

$$\begin{aligned} \vec{V}_{cut\ opening(Right)} &= \vec{V}_{cutting\ vector} \bullet \vec{V}_{surfase\ normal} \\ \vec{V}_{cut\ opening(Left)} &= -\vec{V}_{cut\ opening(Right)} \end{aligned} \tag{7.7}$$

The original vertex moves (right side vertex) along the tool opening vector and the newly split vertex (left side vertex) moves in the negative direction of the tool opening vector as shown in Figure 7-15. Until the tool leaves the surface, this progressive node snapping is repeated.

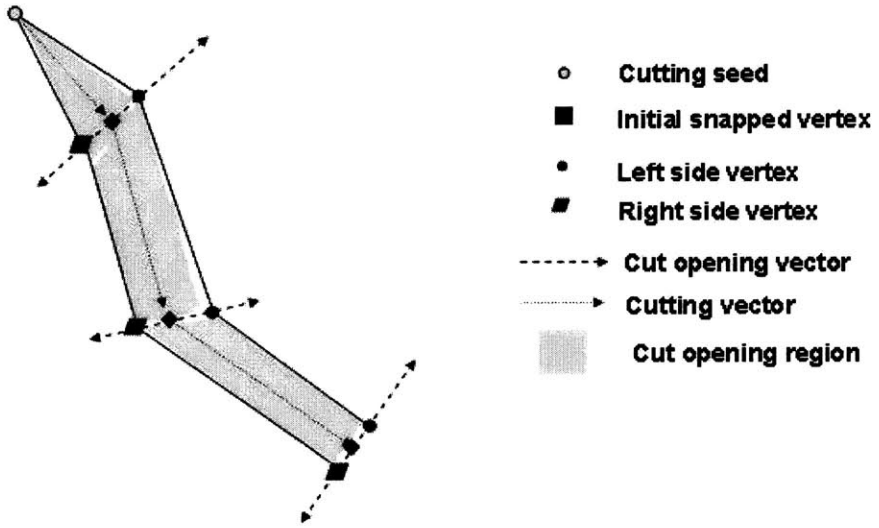

Figure 7-15 Node snapping and splitting during the progressive cutting. The shaded part represents the cut opening area.

To improve visual realism, we added a low resolution model with similar geometry and a different surface under the original model. Since different surface is shown in the cut opening, it generates cutting effects on the organ model. For a clearer demonstration, we tested the algorithm in a two dimensional cube shape as shown in Figure 7-16. This algorithm works in three-dimensional cases which we will present in the next section.

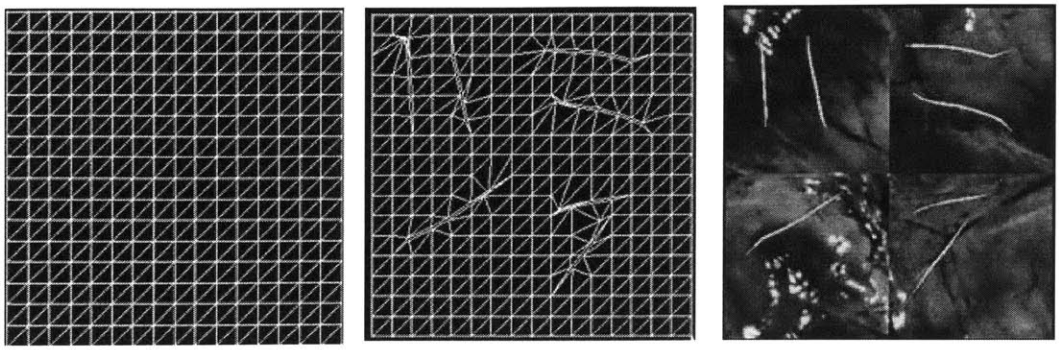

Figure 7-16 Comparison of cutting simulation. (Left) Uncut shape. (Middle) A wire-frame model after cutting. The nodes are snapped and split to create cut opening. (Right) A cut shape with texture mapping.

7.8 Demonstration and discussion

Figure 7-18 presents the deformation simulation of lower esophagus. As the user manipulates the force-feedback laparoscopic tools and interacts with the esophagus model, the associated deformations of the organs are displayed on the computer monitor and the reaction forces are fed back to the user through haptic interfaces. If the users grip the skin of the model, they can feel the gripping force. The user also feels the forces that arise from pulling and pushing the model during the simulation. The images in Figure 7-18 are example images of the simulation of deformation. It should be noted that the material properties used in the simulation were obtained using the techniques presented in Chapters 5 and 6. We used a mean value of eight animal subjects (see Figure 7-17) and the Young's modulus for the tissue model is 5.22 kPa.

We have implemented the visual cutting algorithm in the previous model to the esophagus model as shown Figure 7-19. If the cutting mode is detected, the cut opening is displayed following cutting tool path. The cutting mode can be determined in two ways: use of the special cutting tool or exceeding the predetermined limit of interaction forces with other instruments. The interaction forces were still computed from the elastic model based on the penetration depth of the instrument. Noted that the limit force and the cutting force cannot be determined in our experiments because our only nondestructive measurement of tissues was performed

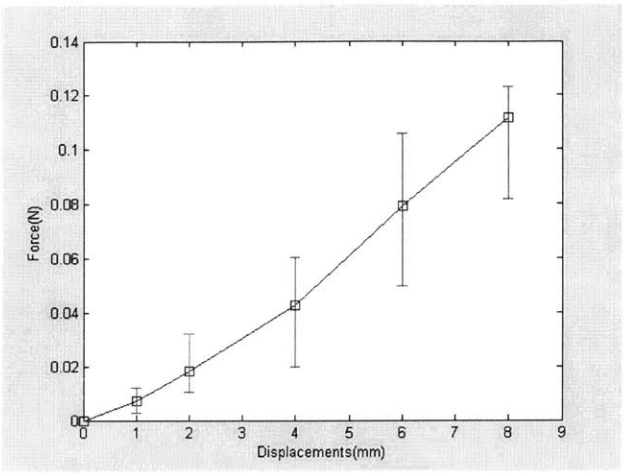

Figure 7-17 Static force responses of lower esophagus. The line with square marks represents median data. Data from eight pigs are used in this plot.

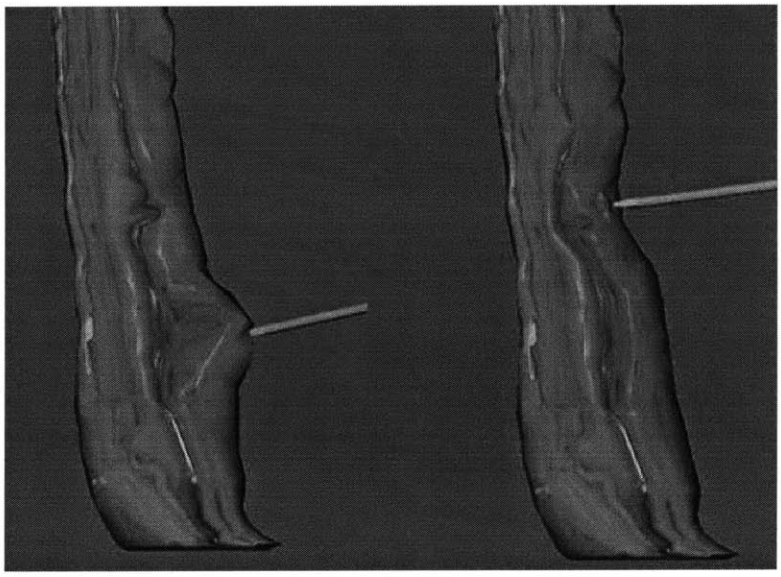

Figure 7-18 Deformation of the lower esophagus model. (Left) pulling by a gripper (Right) palpating of the model.

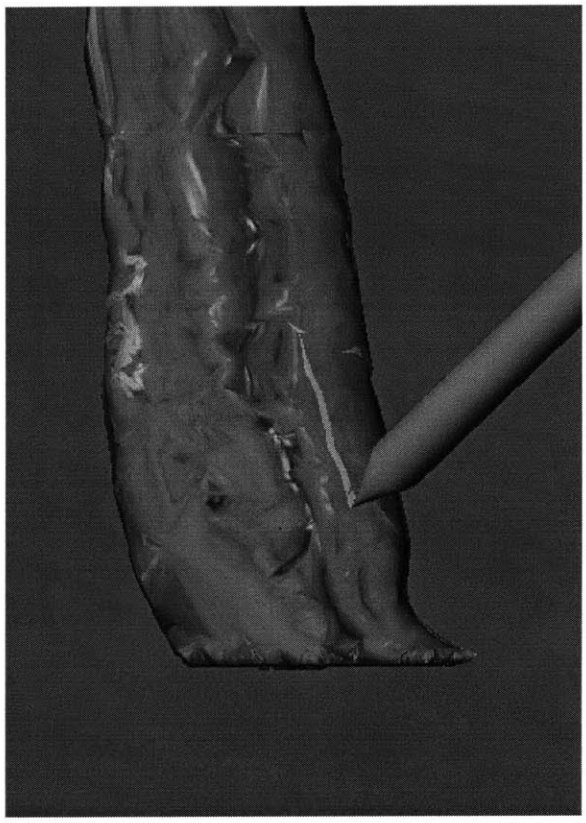

Figure 7-19 Cutting simulation on the lower esophagus model.

Chapter 8 Conclusions and Future Suggestions

This chapter summarizes the conclusions in this thesis and gives suggestions on further research directions and possible extensions. The following paragraphs cover all the topics addressed in this thesis. The overall contribution of this thesis can be divided into two parts; the development of tool tissue interaction models and the characterization of *in vivo* mechanical properties of soft tissues.

8.1 Overall Issues

We developed a soft tissue model for a virtual reality based simulator providing realistic haptic and visual responses to a user in real time. We characterized the soft tissue properties from *in vivo* animal experiments using several biomechanics models and the estimation algorithms. The final product of this thesis is a prototype of a MIS simulator. We developed a three dimensional esophagus model based on the Visible Human Dataset for simulation, that supports limited opening of skin tissue and mid-level (up to 10mm) deformation caused during gripping and palpation. In real MIS scenarios, more surgical procedures are performed so more complex algorithms are required to simulate such as those for scraping and excising of pathological tissues.

Unlike visual interfaces, the haptic device needs to be extended to simulate more general surgical instruments such as surgical scissors and harmonic scalpels. In addition, a haptic device with full force and torque feedback is necessary to increase the realism of the procedure being simulated. Furthermore, the computing architecture and scheduling

algorithms must improve to handle more complex algorithm both in collision detection and collision responses. Currently, the Digital Signal Processor (DSP) based coprocessor boards are rapidly advancing both in lower cost and computing power, so they could be integrated into the simulator as a computing engine.

One of the questions that remains not fully answered is how much realism asd detail we need for effective surgical training, although surgeons generally believe that a good trainer is one that is capable of reproducing the actual operative conditions in order to immerse the trainee in a virtual world that is an accurate representation of the real world. It is possible to learn the surgical practices with a simple unrealistic model but it may lead to negative training transfer. However, due to the learning abilities and perceptual limitations of the human sensory, motor, and cognitive system, perfect simulation is unnecessary, regardless of the existence of such a model. Thus, actual needs to for effective surgical simulation in virtual environments should be investigated, which will guide choices as to how far we should integrate the complex biomechanics of soft tissues.

8.2 Real Time Simulation of Tool–tissue Interactions

There are two key real time tasks in this simulation: collision detection and collision response. This thesis is dedicated mainly to the collision response area and assumes collision detection provides sufficient accuracy and functionality. Depending on the region of interest of a final deformation, we have developed two kinds of real time tissue models that simulate tool tissue interactions in real time. A local deformation model simulates the haptic and visual deformation based on the point-based discretization of the region surrounding a tool–tissue contact point, while a hybrid model integrates the BEM based global deformation model with the local model or only enhancements of visual resolution. Both of these schemes provide real time performance for haptic and visual rendering.

A possible extension of the tissue model is the modeling of nonlinearities and nonhomogeneous mediums. Many anatomical structures consist of multiple layers with different material properties, so the modeling of these would be necessary for some

surgical procedures. Furthermore, a tissue model integrated with geometrical modification algorithms for cutting simulation could improve the realism and functionality of the simulation. In collision detection, much remains to be developed for more realistic simulation of tool-tissue interactions such as detection of self-collision, organ-organ collision, or effective torque rendering in addition to force rendering in the current haptic rendering algorithms. Since a certain level of precomputation is unavoidable for real time rendering, effective strategies for precomputation of the global model can also be further improved. There are several techniques which could accelerate the calculation of solving the deformation equations such as the precorrected-FFT method [94].

Voxel-based modeling could be an alternative way to circumvent many difficulties in the current continuum mechanics based model. Because all properties are assigned to each volume cell (voxel), we could simplify the computation procedures for rendering, such as solving a large system of partial differential equation. For example, the interaction forces at each area are recorded as a form of "stiffness map" and are rendered in real time without much effort. However, generation of the stiffness map from experiments or the development of special graphics hardware for volume rendering should be solved to be considered as an alternative.

8.3 Soft Tissue Property Measurement and Characterization

We designed an experimental system to measure the *in vivo* material properties of pig intra-abdominal organs in a surgical setup. An experimental protocol was developed for the systematic measurement of the mechanical properties of biological tissues for surgical simulation. Ramp and hold as well as sinusoidal indentation stimuli were used in this study. These indentation stimuli were delivered to the organs and the reaction forces were measured. Conditions for the stimuli were designed such that they closely mimicked the conditions in an operating room. To extract meaningful information for both real time modeling and off-line analysis from the experimental data, we developed several algorithms to determine material parameters matching the simulated data with the experimental ones. Under the framework of the quasilinear viscoelastic material

assumption, we identified the viscoelastic and hyperelastic parameters as well as the linear elastic parameter.

The effects of death of the animal on the mechanical properties of organs were determined through comparing measurements on the respective organs under different experimental conditions. However, this change in the mechanical properties of the organs might be due to degeneration of the tissues or loss of blood pressure or a combination of both. To more quantitatively infer the effects of tissue degeneration on the mechanical properties of biological tissues, *in vitro* experiments can be carried out with the organs suitably perfused with fluids to simulate blood flow. This would decouple the change in mechanical properties due to tissue degeneration and loss of blood pressure. Possibly, future experiments to determine the mechanical properties of biological tissues could even be carried out on cadavers if experimental results were to show that tissue degeneration does not significantly change the mechanical properties of biological tissues.

Tissue property measuring devices for *in vivo* properties also needs improvement in the following areas. First, the *in vivo* measurements have been taken from indentation testing with a small radius indenter both in a mid-line incision or a minimally invasive setup due to the ease of testing. A device recording the forces and the deformation during gripping or cutting of tissues is necessary to understand the behavior of soft tissues and collect data for simulation of other surgical procedures. For example, the initiation of tearing or cut openings of tissues could be tested using this device. Second, the current devices can measure only single point at a time. A device that is capable of measuring multiple points or whole regions could provide more information for tissue characterization. The device integrated with a three dimensional imaging device could also be useful in further analysis.

Appendix A Labels and weight of pig subjects

Pig label	A1	A2	A3	A4	B1	B2	B3	B4	B5	B6	B7
Weight	32	32	45	43	37	42	32	34	32	37	32
Liver	O	O	O	O	O	O	O	O	O		O
Esophagus	O	O	O	O	O	O		O			O
Kidney							O	O	O	O	O
Spleen							O	O	O	O	

References

[1] R. D. Howe and Y. Matsuoka, "Robotics and Surgery," *Annual Review of Biomedical Engineering*, vol. 1, pp. 211-240, 1999.

[2] M. A. Srinivasan and C. Basdogan, "Haptics in Virtual Environments: Taxonomy, Research Status, and Challenges.," *Computer & Graphics*, vol. 21, pp. 393-404, 1997.

[3] M. J. Mack, "Minimally Invasive and Robotic Surgery," *JAMA (Journal of American Medical Association)*, vol. 385, pp. 568-572, 2001.

[4] M. R. Treat, "Surgeon's Perspective on the Difficulties of Laparoscopic Surgery," in *Computer Assisted Surgery*, Taylor, Ed. Cambridge, MA: MIT Press, 1996.

[5] S. Gibson, "Simulating Arthroscopic Knee Surgery using Volumetric Object Representations, Real-Time Volume Rendering and Haptic Feedback," Mitsubishi Electric Research Laboratories, Cambridge, MA TR96-19, 1996.

[6] U. G. Kuhnapfel and B. Neisius, "Realtime Graphical computer simulation for Endoscopic Surgery," Medicine Meet Virtual Reality(MMVR 94), 1994.

[7] C. Basdogan, C. Ho, and M. A. Srinivasan, "Virtual Environments for Medical Training: Graphical and Haptic Simulation of Laparoscopic Common Bile Duct Exploration," *IEEE/ASME Transactions on Mechatronics*, vol. 6, pp. 269-285, 2001.

[8] F. Tendick, M. Downes, T. Goktekin, M. C. Cavusoglu, D. Feygin, X. Wu, R. Eyal, M. Hegarty, and L. W. Way, "A Virtual Environment Testbed for Training Laparoscopc Surgical Skill," *Presence*, vol. 9, pp. 236-255, 2000.

[9] G. Szekely, M. Bajka, C. Brechbuhler, and e. al., "Virtual Reality Based Surgery Simulation for Endoscopic Gynaecology," Proceedings of MMVR'99, 1999.

[10] M. Bro-Nielsen and e. al., "PreOp Endoscopic Simulator: a PC-based Immersive Training System for Bronchoscopy," Proceedings of Medicine Meets Virtual Reality(MMVR) 7, 1999.

[11] D. R. Hunt and V. L. Wills, "Laparoscopic Heller Myotomy for Achalasia," *Auto. N.Z. Journal of Surgery*, vol. 70, pp. 582-586, 2000.

[12] W. E. Lorensen and H. E. Cline, "Marching Cubes : High Resolution 3D Surface Construction Algorithm," *Computer Graphics*, vol. 21, pp. 163-169, 1987.

[13] C.-H. Ho, "Computer Haptics: Rendering Technique for Force-feedback in Virtual Environments," in *Mechanical Engineering*. Cambridge: Massachusetts Inst. of Tech., 2000, pp. 127.

[14] G. Buedea, *Force and Touch Feedback for Virtual Reality*. N.Y.: John Wiley and Sons, Inc., 1996.

[15] J. Biggs and M. A. Srinivasan, "Haptic Interfaces," in *Handbook of Virtual Environments*, K. Stanney, Ed. London: Lawrence Earlbaum, Inc., 2002.

[16] H. K. Kim, D. W. Rattner, and M. A. Srinivasan, "The Role of Simulation Fidelity in Laparoscopic Surgical Training," Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003, 2003.

[17] C. Ho, C. Basdogan, and M. A. Srinivasan, "Efficient Point-Based Rendering Techniques for Haptic Display of Virtual Objects," *Presence*, vol. 8, pp. 477-491, 1999.

[18] C. Basdogan, C. Ho, and M. A. Srinivasan, "A Ray-Based Haptic Rendering Technique for Displaying Shape and Texture of 3D Objects in Virtual Environments," ASME Winter Annual Meeting, 1997.

[19] S. Ho, S. Sarma, and Y. Adachi, "Real-time interference analysis between a tool and an environment," *Computer Aided Design*, vol. 33, pp. 935-947, 2001.

[20] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer, "Elastically Deformable Models," *Computer Graphics*, vol. 21, pp. 205-214, 1987.

[21] D. Terzopoulos and K. Waters, "Physically Based Facial Modeling , Analysis and Animation," *Journal of Visualization and Computer animation*, vol. 1, pp. 73-80, 1990.

[22] G. Celnker and D. Gossard, "Deformable Curves and Surface Finite Elements for Free-form Shape Design," Proceeding of SIGGRAPH 1991 Conference, 1991.

[23] S. Delp, P. Loan, C. Basdogan, and J. Rosen, "Surgical Simulation : An Emerging Technology for Training in Emergency Medicine," *Presence*, vol. 6, pp. 147-159, 1997.

[24] C. Basdogan, C. Ho, and M. A. Srinivasan, "Force Interaction in Laparoscopic Simulation Haptic Rendering of Soft Tissue," Proceeding of MMVR'6 Conference, 1998.

[25] W. Hsu, J. Hughes, and H. Kaufman, "Direct manipulation of free-form deformation," *Proceeding of ACM SIGGRAPH'92*, vol. 6, pp. 147-159, 1992.

[26] S. Cover, N. Ezauerra, and J. O'Brien, "Interactively Deformable Models for Surgery Simulation," *IEEE Computer Graphics Application Magazine*, pp. 68-75, 1992.

[27] U. G. Kuehnapfel and B. Neisius, "CAD-Based Graphical Computer Simulation in Endoscopic Surgery," *Endoscopic Surgery*, vol. 1, pp. 369-378, 1993.

[28] C. Paloc, F. Bello, R. I. Kitney, and A. Darzi, "Online Multiresolution Volumetric Mass Spring Model for Real Time Soft Tissue Deformation," Proceeding of MICCAI 2002 Conference, 2002.

[29] S. De and M. A. Srinivasan, "Thin walled models for haptic and graphical rendering of soft tissues in surgical simulation," *Proceeding of MMVR, '7 Conference*, pp. 94-99, 1999.

[30] M. Bro-Nielsen, "Finite Element Modeling in Surgery Simulation," *Proceeding of IEEE*, vol. 86, pp. 490-503, 1998.

[31] S. Cotin, H. Delingette, and N. Ayache, "Real Time Elastic Deformations of Soft Tissues for Surgery Simulation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, pp. 62-73, 1999.

[32] X. Wu, M. S. Downes, T. Goktekin, and F. Tendick, "Adaptive Nonlinear Finite Elements for Deformable Body Simulation Using Dynamic Progressive Meshes," *Computer Graphics Forum*, vol. 20, pp. 349-58, 2001.

[33] D. James and D. K. Pai, "ArtDefo, Accurate Real Time Deformable Objects," Computer Graphics (ACM SIGGRAPH 99 Conference Proceedings), 1999.

[34] D. L. James and D. K. Pai, "Multiresolution Green's Function Methods for Interactive Simulation of Large-Sacle Elastostatic Objects," *ACM Transactions on Graphics*, vol. 22, pp. 47-82, 2003.

[35] S. De and K. J. Bathe, "The Method of Finite Spheres," *Computational Mechanics*, vol. 25, pp. 329-345, 2000.

[36] S. De, J. Kim, and M. A. Srinivasan, "A Meshless Numerical Technique for Physically Based Real Time Medical Simulations," Proceeding of MMVR 2001 Conference, 2001.

[37] K. J. Bathe, *Finite Element Procedures*: Englewood Cliffs, NJ, Prentice Hall, 1996.

[38] Y. Saad and M. H. Schultz, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, pp. 856-869, 1986.

[39] O. R. Astley and V. Hayward, "Multirate Haptic Simulation Achieved by Coupling Finite Element Meshes through Norton Equivalents," Proceedings of the 1998 IEEE International Conference on Robotics & Automation, Belgium, 1998.

[40] M. C. Cavusoglu and F. Tendick, "Multirate Simulation for High Fidelity Haptic Interaction with Deformable Objects in Virtual Environments," Proceedings of the IEEE International Conference on Robotics and Automation, 2000.

[41] J. Zhang, S. Payandeh, and J. Dill, "Haptic Subdivision: An Approach to Defining Level-of-detail in Haptic Rendering," , Proceedings of the 10th Symp. on Haptic Interfaces for Virtual Env. & Teleoperator Systems, 2002.

[42] A. Khodakovsky, P. Schroder, and W. Sweldens, "Progressive Geometry Computation," SIGGRAPH 2000,Computer Graphics Proceeding, 2000.

[43] C. Monserrat, U. Meier, M. Alcaniz, F.Chinesta, and M. A. Juan, "A New Approach for the Real-time Simulation of Tissue Deformation in Surgery Simulation," *Computer Methods and Programs in Biomedicine*, vol. 64, pp. 77-85, 2001.

[44] C. A. Brebbia, J.C.F.Telles, and L. C. Wrobel, *Boundary Element Technique: Theory and Applications in Engineering*. New York: Springer-Verlag, 1984.

[45] G. H. Golub and C. F. V. Loan, *Matrix Computations. (2nd Ed.)*. Baltimore: John Hopkins University Press, 1989.

[46] J. Warren and H. Weimer, *Subdivision methods for Geometric Design:A Constructive Approach*. San Francisco: Morgan Kaufmann Publishers, 2002.

[47] C. T. Loop, "Smooth subdivision surfaces based on triangles," in *Department of Mathematics*: University of Utah, 1987.

[48] E. Catmull and J.Clark, "Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes," *Computer Aided Design*, vol. 10, pp. 350-355, 1978.

[49] A. Vlachos, J. Peters, C. Boyd, and J. L. Mitchell, "Curved PN Triangles," Symposium on Interactive 3D Graphics, 2001.

[50] S. De and K. J. Bathe, "Towards an Efficient Meshless Computational Technique: the Method of Finite Spheres," *Engineering Computations*, vol. 18, pp. 170-192, 2001.

[51] J. Kim, S. De, and M. A. Srinivasan, "Physically Based Hybrid Approach in Real Time Surgical Simulation With Force Feedback," MMVR 2003, 2003.

[52] J. Kim, S. De, and M. A. Srinivasan, "A Hybrid Modeling Scheme for Virtual Reality Based Medical Simulation," Second M.I.T. Conference on Computational Fluid and Solid Mechanics, 2003.

[53] H. Yamada, *Strength of Biological Materials*. Baltimore: Williams and Wilkins Company, 1970.

[54] Y. C. Fung, *Biomechanics: Mechanical Properties of Living Tissues*. New York: Springer-Verlag, 1993.

[55] J. Rosen, B. Hannaford, M. MacFarlane, and M. Sinanan, "Force Controlled and Teleoperated Endoscopic Grasper for Minimally Invasive Surgery - Experimental Performance Evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 46, pp. 1212-1221, 1999.

[56] F. J. Carter, T. G. Frank, P. J. Davies, D. Mclean, and A. Cuschieri, "Biomechanical Testing on Intra-abdominal Soft Tissues," *Medical Image Analysis*, 2000.

[57] I. Brouwer, J. Ustin, L. Bentley, A. Sherman, N. Dhruv, and F. Tendick, "Measuring In Vivo Animal Soft Tissue Properties for Haptic Modeling in Surgical Simulation," Proceedings of MMVR Conference, 2001.

[58] R. J. Gulati and M. A. Srinivasan, "Determination of Mechanical Properties of the Human Fingerpad, In Vivo, Using a Tactile Stimulator," MIT, Touch Lab RLE TR-605, 1996.

[59] M. P. Ottensmeyer, "*In vivo* Measurement of Solid Organ Viscoelastic Properties," Proceedings of MMVR Conference, 2002.

[60] B. K. Tay, S. De, N. Stylopoulos, D. W. Rattner, and M. A. Srinivasan, "*In vivo* Force Response of Intra-abdominal Soft Tissue for the Simulation of Laparoscopic Procedures," Proceedings of the MMVR Conference, 2002.

[61] A. E. Kerdok, S. M. Cotin, M. P. Ottensmeyer, A. Galea, R. D. Howe, and S. L. Dawson, "Truth Cube: Establishing Physical Standards for Real Time Soft Tissue Simulation," Int. Workshop on Deformable Modeling and Soft Tissue Simulation, Germany, 2001.

[62] B. K. Tay, "In vivo Characterization of the Mechanical Behavior of Intra-abdominal Tissues for Surgical Simulation," in *Mechanical Engineering*. Cambridge: Massachusetts Institute of Technology, 2002.

[63] M. P. Ottensmeyer, "Minimally Invasive Instrument for In vivo Measurement of Solid Organ Mechanical Impedance," in *Mechanical Engineering*. Cambridge: Massachusetts Institute of Technology, 2001.

[64] M. Farshad, M. Barbezat, P. Flueler, F. Shmidlin, P. Graber, and P. Niederer, "Material Characterization of the Pig Kidney in Relation with the Biomechanical Analysis of Rental Trauma," *Journal of Biomechanics*, vol. 32, pp. 411-425, 1999.

[65] P. J. Davis, F. J. Carter, D. G. Loxburgh, and A. Cuschieri, "Mathematical Model for Keyhole Surgery Simulation: spleen capsule as an elastic membrane," *Journal of Theoretical Medicine*, vol. 1, pp. 247-262, 1999.

[66] K. Miller, "Constitutive Modeling of Abdominal Organs," *Journal of Biomechanics*, vol. 33, pp. 367-373, 1999.

[67] K. Miller, K.Chinzei, G.Orssengo, and P.Bednarz, "Mechanical Properties of Brain Tissue in-vivo : Experiment and Computer Simulation," *Journal of Biomechanics*, vol. 33, pp. 1369-1376, 2000.

[68] M. Kauer, V. Vuskovic, J. Dual, G. Szekely, and M. Bajka, "Inverse Finite Element Characterization of Soft Tissue," MICCAI 2001, 2001.

[69] Y. P. Zheng and A. F. T. Mark, "Extraction of Quasi-Linear Viscoelastic Parameters for Lower Limb Soft Tissues from Manual Indentation Experiments," *Journal of Biomechanical Engineering*, vol. 121, pp. 330-339, 1999.

[70] G. A. Johnson, G. A. Liversay, S. L. Woo, and K. R. Rajagopal, "A Single Integral Finite Strain Viscoelastic Model of Ligaments and Tendons," *Journal of Biomechanical Engineering*, vol. 118, pp. 221-226, 1996.

[71] E. O. Carew, E. A. Talman, D. R. Boughner, and I. Vesely, "Quasi-Linear Viscoelastic Theory Applied to Internal Shearing of Porcine Aortic Valve Leaflets," *Journal of Biomechanical Engineering*, vol. 121, pp. 386-392, 1999.

[72] M. C. Boyce and E. M. Arruda, "Constitutive Models of Rubber Elasticity: A Review," *Rubber Chemistry and Technology*, vol. 73, pp. 504-523, 2000.

[73] M. Richards, A. S. Wineman, E. Alsberg, J. A. Goulet, and S. A. Goldstein, "Viscoelastic Characterization of Mesechymal Gap Tissue and Consequence for Tension Accumulation During Distraction," *Transaction of ASME, Journal of Biomedical Engineering*, vol. 121, pp. 116-123, 1999.

[74] J. Cysyk, S. De, and M. A. Srinivasan, "A finite Element Model of the Human Fingerpad for Dynamic Simulations," , vol. Journal of Biomechanics.

[75] A. C. Pipkin, *Lectures on Viscoelasticity Theory*. New York: Springer-Verlag, 1986.

[76] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge: Cambridge University Press, 1988.

[77] P. Seshaiyer and J. D. Humphrey, "A Sub-Domain Inverse Finite Element Characterization of Hyperelastic Membranes Including Soft Tissues," *Journal of Biomechanical Engineering, Transaction of the ASME*, vol. 125, pp. 363-371, 2003.

[78] M. J. Moulton, L. L. Creswell, R. L. Actis, W. K, Myers, M. W. Vannier, B. A. Szabo, and M. K. Pasque, "An Inverse Approach to Determining Myocardial Material Properties," *Journal of Biomechanics*, vol. 28, pp. 935-948, 1995.

[79] R. M. Berne, M. N. Levy, B. M. Koeppen, and B. A. Stanton, *Physiology*, 4th ed. Danvers: sMA:Mosby, Inc., 1998.

[80] E. M. O. Lemme, G. R. Domingues, V. L. C. pereira, C. G. Firman, and J. Pantoja, "Lower Esophagus Sphincter Pressure in Idiopathic Achalasia and Chagas Disease-related Achalasia," *Disease of the Esophagus*, vol. 14, pp. 232-234, 2001.

[81] R. J. Adams and B. Hannaford, "Stable Haptic Interaction with Virtual Environments," *IEEE Transaction on Robotics and Automation*, vol. 15, pp. 465-545, 1999.

[82] J. E. Colgate and J. M. Brown, "Factors affecting the Z-width of a Haptic Display," IEEE international Conference of Robotics and Automation, Los Alamos, 1994.

[83] M. C. Cavusoglu, D. Feygin, and F. Tendick, "A Critical Study of the Mechanical and Electrical Properties of the PHANToM Haptic Interface and Improvements for High Performance Control," *Presence: Teleoperators and Virtual Environments*, vol. 11, 2002.

[84] F. Neyret and M.-P. Cani, "Pattern-Based Texturing Revisited," SIGGRAPH 99 Conference Proceedings, 1999.

[85] J. F. Blinn and M. E. Newell, "Texture and Reflection in Computer Generated Images," *CACM*, vol. 19, pp. 542-547, 1979.

[86] M. Lin and S. Gottschalk, "Collision Detection between Geometric Models: A survey.," Proc. of IMA Conference on Mathematics of Surfaces, 1998.

[87] A. Gregory, M. Lin, S. Gottschalk, and R. Talor, "H-COLLIDE: A Framework for Fask and Accurate Collision Detection for Haptic Interaction," IEEE Virtual Reality Conference, 1999.

[88] G. v. d. Bergen, "Efficient Collision Detection of Complex Deformable models using AABB Trees," *Journal of Graphics Tools*, pp. 1-14, 1998.

[89] D. Bielser, *A Framework for Open Surgery System*: Hartung-Gorre Verlag, 2003.

[90] C. Bruyns, S. Senger, A. Menon, K. Montgomery, S. Wildermuth, and R. Boyle, "A Survey of Interactive Cutting Techniques and a New Method for implementing generalized Interactive Mesh Cutting Using Virtual Tools," *Journal of Visualization and Computer Animation*, vol. 13, pp. 21-42, 2002.

[91] D. Bielser and M. H. Gross, "Interactive Simulation of Surgical Cut," Proceeding of Pacific Graphics 2000, 2000.

[92] A. B. Mor and T. Kanade, "Modifying Soft Tissue Models : Progressive Cutting with minimal new element Creation," Proceeding of Medical Image Computing and Computer Assisted Intervention(MICCAI), 2001.

[93] H.-W. Nienhuys and A. F. v. d. Stappen, "Combining Finite Element Deformation with Cutting for Surgery Simulation," Proceeding of the Eurographics 2000, 2000.

[94] J. R. Phillips and J. K. White, "A Precorrected-FFT method for Electrostatic Analysis of Complicated 3-D Structures," *IEEE Trans. on Computer-Aided Design*, vol. 16, pp. 1059-1072, 1997.