Virtual surgery simulation for medical training using multi-resolution organ models

Jung Kim1*
Changmok Choi1
Suvaran De2
Mandayam A. Srinivasan3

1School of Mechanical, Aerospace and Systems Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea
2Department of Mechanical, Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA
3Massachusetts Institute of Technology, Cambridge, MA, USA

*Correspondence to: Jung Kim, Department of Mechanical Engineering, #5201 KAIST, Science Town, Daejeon 305-701, Korea. E-mail: jungkim@kaist.ac.kr

Abstract

Background Real-time simulation of organ deformation is one of the biggest challenges in virtual surgery, due to the conflicting requirements of real-time interactivity and simulation realism. In this paper we propose a method to overcome this challenge by introducing a multi-resolution modelling technique.

Methods In our approach a reasonably coarse global model is locally enhanced, using a mesh subdivision and smoothing algorithm. The global model is based on a discretization of the boundary integral representation of three-dimensional deformable objects. Local refinements are provided at the tool-tissue interaction region by a local subdivision technique.

Results As an example, we have developed a deformable human kidney model generated from the Visible Human Dataset, with tissue properties determined from in vivo animal experiments. A mixed reality laparoscopic surgical training system has been developed, using an abdominal mannequin and force feedback devices.

Conclusions The use of precomputation and structural re-analysis techniques results in a very rapid computation procedure. Validation of the simulator is in progress. Copyright © 2007 John Wiley & Sons, Ltd.

Keywords physically-based organ model; surgical simulation; real-time computation

Introduction

Surgery simulation, using interactive anatomical models and multi-modal virtual reality technologies to simulate surgical procedures for practice, is a rapidly growing field (1–4). Surgeons can be trained in computer-generated environments where they would interact with three-dimensional (3D) anatomy without risks to patients or sacrifice of animals. Several surgery simulators have been already developed and tested to provide various levels of simulation for training, and these have proved a certain level of effectiveness of training methods based on simulation (5,6).

One of the technical challenges in surgery simulation is the development of a tissue model simulating responses to interventions using surgical instruments. Such computations are numerically expensive, because they involve discretizing complex 3D domains and solving large systems of coupled differential equations for realistic deformation. Furthermore, surgery simulators require real-time performance to provide visual and haptic feedback in addition to simulation realism. While the graphical frames are

Accepted: 7 May 2007
typically updated at about 30–40 Hz, stable haptic interaction requires a much higher update rate, around 500 Hz–1 kHz, for smooth sensations (7). This implies that the force computations need to be achieved within a few milliseconds, and thus severe limitations are imposed on the complexity of the model that can be rendered haptically in real time.

Real-time deformation models (8,9) tend to sacrifice the physical nature of the deformation to achieve real-time performance. ‘Physically-based’ techniques that take into account the underlying mechanics of soft tissue deformation using mass spring networks (10) or sets of finite volumes (11) have been developed. Included among them are techniques based on partial differential equations, such as the finite element method. Because these techniques can provide very realistic deformations but have poor computational efficiency, research efforts have been made to speed up the computations, e.g. condensation (11) or precomputation (12,13). Wu and Tendick (14) also modelled material non-linearities of soft tissue deformation to handle large deformations. Another well-known technique of numerically solving physical problems is the boundary element method (BEM) (15), 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 (16). They modelled quasi-static deformations using the boundary element method and achieved real-time performance using a simple structural re-analysis technique. To circumvent 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 (17). A comprehensive review of the real-time deformable model for surgery simulation can be found in Meier et al. (18).

A high-resolution model is usually necessary to provide greater realism. However, a uniformly high-resolution model is prohibitively expensive in terms of the number of computations per update and is not necessary, since the tools used in laparoscopic surgery are long and slender and interact only with local regions of the organs.

This calls for a multi-resolution approach for the real-time deformation modelling of laparoscopic surgical processes. Astley et al. (19) proposed a primitive multirate simulation technique for haptic interactions, using Norton equivalents. Cavusoglu et al. (20) developed a multirate simulation technique in the context of the finite element technique. This paper, we introduce a novel multi-resolution modelling technique for physically-based real-time surgery simulation, where a reasonably coarse global model is locally enhanced using a mesh subdivision and smoothing algorithm (Figure 1). The justification of using such an approach is as follows. When a tool–tissue interaction occurs, the ‘scene of action’ is restricted to a zone near the tool tip. Therefore, a high-resolution model is required for this region. The rest of the domain can be modelled using a relatively coarse model. The global model is based on a discretization of the boundary integral representation of 3D deformable objects. The use of precomputation and structural re-analysis techniques results in a very rapid computational procedure. The local refinements are provided in the vicinity of the tool–tissue interaction area.

Figure 1. Conceptual diagram of the multi-resolution approach for real-time surgery simulation

The BEM equations of linear elastostatics with \( n \) elements using a piece-wise constant interpolation (i.e. the displacements and tractions are assumed to be constant over each element) (15) may be written as:

\[
c u_i + \sum_{j=1}^{n} \left( \int_{\Delta_i} p^* \Phi^T d\Gamma \right) u_j = \sum_{j=1}^{n} \left( \int_{\Delta_i} u^* \Phi^T d\Gamma \right) p_j
\]

where \( \Delta_i \) is the surface of the \( i \)th element, and \( u^* \) and \( p^* \) are the ‘fundamental solutions’ (15). \( u, p, \Phi \) are displacement and traction vectors at a point \( x \) in the domain or on the boundary and the interpolation functions, respectively. The coefficient \( c \) in equation 1 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 as follows:

\[
U_{ik}^* = \frac{1}{8\pi Er} \left[ (3 - 4\nu)\delta_{ik} + \frac{\partial r}{\partial x_i} \frac{\partial r}{\partial x_k} \right]
\]

\[
p_{ik}^* = \frac{1}{8\pi(1-\nu)r^2} \left[ \frac{\partial r}{\partial n} \left( 1 - 2\nu \right) \delta_{ik} + 3 \frac{\partial r}{\partial x_i} \frac{\partial r}{\partial x_k} \right]
- \left( 1 - 2\nu \right) \delta_{ik} \left( \frac{\partial r}{\partial x_i} \frac{\partial r}{\partial x_j} - \frac{\partial r}{\partial x_j} \frac{\partial r}{\partial x_i} \right)
\]

where \( n \) is a unit outward normal to the surface, \( \delta_{ik} \) is the Kronecker delta, \( r \) is the distance from the point of application of the load to the point at which \( u \) is being computed, \( E \) is a Young's modulus, \( \nu \) is the Poisson's ratio, and \( n_{ik} \) are the direction cosines (see Figure 2).

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

\[
AY = F
\]

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 \( N \times N \) dense matrix. \( F \) is the known right hand-side vector containing boundary conditions.

In the classical BEM formulation, two computationally expensive steps are involved. The first step is the computation of the system matrix \( A \) from geometric and material properties of the object. The second step is the solution of the dense system in equation (3) (see Table 1 for the time needed to build the system matrix and solve the system equations for three different models). For example, for a model with 5000 polygons, it takes several days to solve equation (3) above!

Using acceleration techniques, such as the Fast Multipole Method (23), the boundary element equations may be solved quite efficiently. However, since the boundary condition on only a very small portion of the model actually changes during simulation, we use...
Table 1. Model sizes, precomputation times off-line and update times on-line for the three BEM models displayed in Figure 3 (as performed on a Windows NT workstation with a dual Pentium III 1 GHz processor)

<table>
<thead>
<tr>
<th>Model</th>
<th>Number of vertices</th>
<th>Number of polygons</th>
<th>Precomputation time off-line</th>
<th>Update time on-line (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cube</td>
<td>98</td>
<td>192</td>
<td>11 s</td>
<td>0.17</td>
</tr>
<tr>
<td>Kidney (graphic model)</td>
<td>346</td>
<td>688</td>
<td>7 min, 21 s</td>
<td>0.54</td>
</tr>
<tr>
<td>Kidney (created from the Visible Human Dataset)</td>
<td>2979</td>
<td>5145</td>
<td>208 h, 26 min</td>
<td>4.3</td>
</tr>
</tbody>
</table>

A technique similar to the one developed in (16) 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.

Let $A_0$ be the reduced 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 $A_0$, as follows:

$$A = A_0 + UV^T$$

(4)

where $U$ and $V$ are block matrixes representing difference between the original matrix and the updated matrix (16). The matrix $A_0$ and its inverse are precomputed and stored. The inverse of $A$ is given by the Sherman–Morrison–Woodbury (S–M–W) formula (24) as follows:

$$A^{-1} = A_0^{-1} - A_0^{-1}UC^{-1}V^TA_0^{-1}$$

(5)

where $C = I + V^TA_0^{-1}U \in \mathbb{R}^{3s \times 3s}$ is known as the ‘capacitance matrix’ and ‘$s$’ is the number of non-zero boundary conditions. For 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, ‘$s$’ = 3 and the computing cost for matrix inversion is $O(N)$ instead of $O(N^3)$.

The interaction forces at the surgical instrument are computed using the following formula:

$$F = a_{tip}p_c$$

(6)

where $a_{tip}$ and $p_c$ are the effective nodal area of the tool tip and the traction vector at node ‘c’ where the interaction occurs, respectively. Table 1 summarizes the computing costs of off-line (precomputation) and real-time computation times for the various geometrical models shown in Figure 3. We have chosen three models for our numerical experiments: (a) a cube; (b) a crude kidney model; and (c) a kidney model created from the Visible Human Dataset. The use of the S–M–W formula allows us to simulate even the most complex model at real-time rates.

Local enhancement of deformation

One of the major difficulties encountered during the display of deformation fields using a model with a fixed spatial resolution is the presence of visual artifacts, such as flatness, which appear in the vicinity of the tool–tissue interaction point experiencing large displacements. To overcome this problem, subdivision algorithms are frequently used (14,25). These algorithms, however, increase numerical complexity and require extra memory because they provide unnecessary detail. 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 have employed a local subdivision algorithm only in the vicinity of the tool–tissue interaction region. We have also developed a local smoothing algorithm to enhance the visual quality of the deformation field. As illustrated in Figure 4, the steps in this procedure are summarized as follows:

1. Detect collision between the surgical tool and the organ model (Figure 4a).
Figure 4. Steps in the local enhancement technique

2. Find neighbourhood triangles around the contact triangle that need local enhancement (Figure 4b).
3. Subdivide each triangle into a set of finer triangles (Figure 4c)
4. Read displacements of nodal points from the global model.
5. Smooth the deformation field using interpolation functions as described below (Figure 4e).
6. Compute displacements of points in refined triangles by using the interpolation functions (Figure 4f).
7. Display the entire object in its deformed configuration.

Among the many subdivision algorithms described in computer graphics literature (26,27), we have chosen the PN algorithm developed by Vlachos et al. (28). In this algorithm, a triangle is subdivided into a set of smaller triangles that have variations of normal vectors, as shown in Figure 4d. 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 data other than the position and normal vector of three vertices are necessary, it is computationally efficient compared to the other techniques and requires the information of only the neighbourhood polygons. The details of the PN triangulation algorithm are explained in Kim et al. (29).

When the surgical tool contacts a triangle, we define the ‘edge neighbourhood’ of that triangle as the set of triangles that share an edge with that triangle. The ‘vertex neighbourhood’ is defined as the group of triangles sharing
Figure 5. Subdivision areas: (a) a triangle in contact with a tool; (b) triangles sharing edges with the contact triangle; and (c) triangles sharing vertices with the contact triangle

Figure 6. (a) Interpolation functions in 1D; (b) 2D version of interpolation function

If the displacements computed at nodal points are \( \hat{u}_i \), \( i = 1, 2, \ldots, N \), then a smooth deformation field may be defined using the following interpolation formula:

\[
\mathbf{u}_{\text{interp}} = \sum_{i=1}^{N} \varphi_i(\mathbf{x}) \hat{u}_i
\]

where \( \varphi_i \) is a basis function at node ‘i’. In our work, we have used the Shepard partition of unity approach to generate the basis functions. In this approach, we first define, at each node ‘i’ a radial weight function \( W_i(\mathbf{x}) \), which is non-zero, on a sphere of radius \( r_i \) centred at node ‘i’. In this work, we have used quartic spline weight functions of the form:

\[
W_i = \begin{cases} 
1 - 6m^2 + 8m^3 - 3m^4 & 0 < m < 1 \\
0 & m > 1 
\end{cases}
\]

where \( m = (\|\mathbf{x} - \mathbf{x}_i\|/r_i) \)

The Shepard partition of unity function at node ‘i’ is defined by:

\[
\varphi_i = \frac{W_i}{\sum_j W_j}
\]

Notice that these functions are non-zero on the sphere of radius \( r_i \) centred at node ‘i’. Figure 6 shows one of these functions at a node on a 2D domain. The choice of the radius of influence, \( r_i \), at node ‘i’ is an important factor determining the number of nodal points included by node ‘i’. Figure 7 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 smoother deformation field is observed, as compared with (b).

Material parameters based on biomechanical soft tissue behaviour

The major advantage of using a differential/integral equation-based technique for deformation modelling is the ability to use a material model. The assumption of linear isotropic elasticity implies that only two independent parameters, Young’s modulus and Poisson’s ratio, are necessary to simulate the response of soft tissues. Therefore, characterization of tissue properties, especially in vivo, is an important issue in the development of a surgery simulator. Most of the techniques developed in the measurement of soft tissue properties are ex vivo (31). Tissue samples are removed from the organ of interest and tested with devices and procedures similar to those used for engineering materials. However, these data are not suitable for surgery simulation, due to the significant difference between in vivo and ex vivo tissue properties. After removing samples from a body, tissue conditions change drastically because of factors such as: (a) temperature (changes in viscosity); (b) hydration (drying up might change elasticity and...
Virtual surgery simulation using multi-resolution organ models

Figure 7. Smoothing using Shepard functions: (a) the discretized cube model; (b) deformation using a coarse mesh; (c) coarse model with local subdivision and smoothing using Shepard functions

viscosity); (c) breakdown of proteins (change in stiffness); and (d) loss of blood supply. Moreover, the boundary conditions of a sample are different from those of in vivo states. Therefore, the measurement of in vivo tissue behaviour is another important challenge for developing surgery simulators. Although this topic is beyond the scope of this paper, there are a few notable devices for recording in vivo behaviour of soft tissues, such as TeMPest (32).

We used the data from the in vivo mechanical properties of pig kidney tissues (33). The responses of soft tissues in intra-abdominal organs (including the kidney) of pigs were measured by using ramp-and-hold indentation experiments. The animals were first put under general anaesthesia and placed on the surgical table. Midline incision techniques were performed to approach the target organ by an experienced surgeon. The range of indentation amplitude (maximum 8 mm) and frequency (maximum 3 Hz) was chosen according to the region of interest and characteristics of the simulated procedures.

The global deformation model is based on a linear elastic material law. From the relationship of force ($F_z$) and displacement ($z$) in the case of normal indentation on a semi-infinite medium by a right circular indenter of radius $a$, the Young's modulus ($E$) can be computed from the equation (34):

$$E = \frac{(1 - \nu^2)F_z}{2a\delta_z} \quad (10)$$

where $\nu$ is the Poisson's ratio of the material. We have assumed a Poisson's ratio close to 0.5 to simulate the condition of incompressibility (31). Figure 8 shows the results of our indentation experiments on pig kidneys from seven subjects. The computed Young's modulus from the seven subjects is 21.3 kPa:

Implementation Issues

Hardware

We have implemented the technique to the simulation of a realistic organ model in a minimally invasive surgery setup, as shown in Figure 9a. The set-up includes a rubber abdominal wall, with surgical instruments and trocars inserted as in real operations. Underneath the abdominal wall, the surgical instruments are connected to Phantom Premium 1.0 (SensAble Technologies) haptic interface devices. The Phantom devices are connected to a Pentium III PC for this set-up. Laparoscopic surgeons attested to the fact that forces by the abdominal wall at the trocar are the dominant forces during surgery and are much greater than the smaller forces between the instruments 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 modelling only needed to be done for the forces between the instruments and the visceral organs.

Geometrical model

The creation of 3D anatomical models of human bodies had been very difficult until the National Library of Medicine (NLM) published a full set of 3D human anatomies named the Visible Human Datasets in 1994 (35). These complete, anatomically detailed, 3D representations of human bodies at 1 mm intervals 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. Figure 9b shows a human kidney model used in our
Figure 9. (a) Minimally invasive surgery simulator set-up. Two surgical instruments are connected to the phantoms under the abdominal mannequin. (b) Kidney model obtained from segmented images of the Visible Human Dataset. This model has a total of 107 K vertexes and 209 K polygons.

Figure 10. A snapshot of a palpation of a kidney model: (a) undeformed model; (b) deformed model. The model was created from the Visible Human Dataset and was optimized in 3D Studio Max. 

Simulation of deformation

Figure 10 shows the simulation of the interaction of a surgical tool with the kidney model. We used a Microsoft WinNT-based personal computer (Pentium III 2.7 GHz processor) with an NVIDIA graphics accelerator (TNT M64) and Phantom force feedback devices from SensAble Technologies Inc. The source code was written in C++, using the OpenGL library for graphics rendering and Ghost SDK for haptic rendering. An update of 40 Hz for the graphics loop and 1 kHz for the force feedback loop have been achieved. Since the kidney model obtained from the Visible Human Dataset is extremely detailed, we used the optimization function in 3D Studio Max to obtain a coarser model, having 6000 nodal points and 10 000 polygons. For rapid collision detection, we used an aligned bounding box technique with the local neighbourhood watch technique developed by Ho et al. (37).
Conclusions

In this paper, we have presented an efficient multi-resolution modelling approach for the simulation of tool–tissue interactions in real time surgery simulation. A coarse global model employing the boundary integral formulation is coupled to a refined local enhancement technique around the vicinity of the tool–tissue contact area. A graphical smoothing scheme using localized interpolation functions is used in the vicinity of tool–tissue interactions to improve visual display of deformation fields, while the global model is used to compute the tool–tissue interaction forces.

The advantage of the boundary integral formulation with the low-rank update provided by the S–M–W formula is a drastic reduction in computational time. However, the computational complexity is still linear with the number of nodal variables. While it is not possible to develop a sublinear algorithm which solves the elliptic differential equations on the organ model, complex models with a large number of degrees of freedom quickly reach the limit of what can be simulated on a general-purpose PC. The implication is that it becomes necessary to implement parallelism in the computations and distribute the computational burden on multiple processors.

The techniques described in this paper are equally applicable to a finite element-based discretization. The finite element technique has advantages over the boundary element technique in the modelling of non-homogeneous and non-linear media. A limitation of the current approach is that the refinement in the tool–tissue interaction region is purely geometric. A physically-based technique that overcomes this problem is described by De et al. (17). Some recent work in this area has been published by Lim and De (38,39). Another challenge is the incorporation of non-linearities in the model response. As stated earlier, the BEM is not well suited to handle non-linearities. A multi-resolution technique for non-linear soft tissue deformation modelling has been developed (38,39), not in the context of the boundary element method, but the method of finite spheres. Other limitations of the method include the inability to simulate topological changes, such as during surgical cutting and when multiple organs are in contact.

The challenge of developing deformation modelling for surgery simulation is the requirement of a broad spectrum of knowledge, including an understanding of computer algorithms, computer visualization, machine haptics, instrumentation, biomechanical experiments and computational mechanics, as well as familiarity with clinical considerations. It is difficult to develop a realistic simulation model that satisfies experts from all these areas simultaneously. One encouraging approach is that of utilizing an open source framework, such as GipSi, as proposed by Cavusoglu and Tendick. The latest release of the GipSi framework and relevant information are well documented in Cavusoglu et al. (40).

While encouraging advancements have been shown in this paper on improving computational speed and fidelity, we are now working on validation studies which are aimed at demonstrating the effectiveness of the simulator in the clinical setting.

Acknowledgments

This work was supported by the Harvard Centre for Minimally Invasive Surgery and partly by a Basic Research Fund of the Korea Institute of Machinery and Materials (KIMM).

References


