

Multimodal Simulation of Laparoscopic Heller Myotomy Using a Meshless Technique

S. De¹, M. Manivannan¹, J. Kim¹, M. A. Srinivasan¹ and D. Rattner²

¹*Laboratory for Human and Machine Haptics,
Massachusetts Institute of Technology, Cambridge, MA 02139*

²*Division of General and Gastrointestinal Surgery,
Massachusetts General Hospital, Boston, MA 02114*

Abstract. In this work we focus our attention on developing a surgical simulator for performing laparoscopic Heller myotomy using force feedback. A meshless numerical technique, the method of finite spheres, is used for the purpose of physically based, real time haptic and graphical rendering of soft tissues. Localized discretization allows display of deformations in the vicinity of the tool tip as well as interaction forces at high update rates (kHz). Novel cutting algorithms are implemented using point-based representation of anatomical models. Graphical rendering is accomplished by using a recently developed volumetric rendering technique known as splatting.

1. Introduction

Over the past few years, laparoscopic esophageal myotomy (Heller myotomy) has emerged as the surgical treatment of choice for the management of a rare esophageal motility disorder known as achalasia (see Figure 1). During the laparoscopic procedure the outer longitudinal muscle layer and the inner circular muscle layer surrounding the lower esophageal sphincter (LES) are incised allowing it to open more easily. Benefits of this approach include five small incisions instead of one large abdominal incision, shorter hospital stay, reduced postoperative pain and a shorter recovery time of a few days.

The success of a laparoscopic surgeon depends heavily on his or her training. In this work we focus on the development of a multimodal virtual environment system that allows the user to perform laparoscopic Heller myotomy using his/her senses of vision as well as touch. Such a system will not only result in customized practice environments for surgical residents, but will also reduce the use of animals and cadavers that are currently used for such training.

One major requirement of such a surgical trainer is real time performance [1]. For real time visual display an update rate of 30 Hz is sufficient. We use a Phantom haptic interface device that records the position and orientation of the user's hands and conveys to him/her forces resulting from the interaction of the virtual laparoscopic tool with the computer model of the esophagus. For stable simulation, the haptic loop requires an update rate of about 1kHz. This imposes severe restrictions on the complexity of the models that can be rendered haptically.

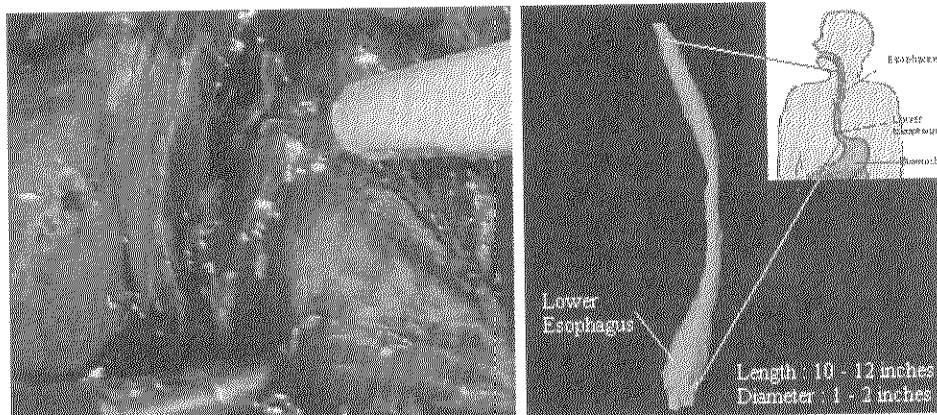


Figure 1. A snapshot of the Heller myotomy procedure is shown on the left. A model of the esophagus is shown on the right.

The material modeling of the soft tissue that comprises the esophagus is quite challenging. Soft tissues exhibit complex material properties [2]. They are nonlinear, anisotropic, viscoelastic, nonhomogeneous and layered. It is quite difficult to obtain *in vivo* material properties of the esophagus. We have performed *in vivo* experiments on the esophagus of pigs and developed linear elastic material models that have been used in our simulations. See [3] for details regarding the experiments and results.

Several researchers have applied finite element techniques for real time surgical simulations [1, 4, 5]. However, finite element techniques suffer from certain drawbacks in real time simulations. First, the contact between tool and tissues must occur only at nodal points. Therefore, to prevent loss of resolution, the density of nodal points should be sufficiently high. This requires extensive memory resources and high computational overhead. Second, cutting or tearing requires an expensive remeshing process during simulation. This means precomputed data of the object becomes, at least locally, invalid and all the data displayed to the user must be computed in real time. The computation time increases approximately as the cube of the number of nodal unknowns. This poses significant obstacles in real time applications, given the high rate of force updates required.

In [6] we presented a meshless numerical scheme, the method of finite spheres, for laparoscopic surgical simulation that does not suffer from these problems. In this technique nodal points are sprinkled locally around the surgical tool tip and the interpolation is performed by functions that are nonzero only on spheres surrounding the nodes. The governing partial differential equations of elasticity are applied at the nodal points only (a technique known as “collocation”). A force extrapolation technique was then used to obtain real time performance.

In this paper we use a point-based discretization of the geometric model and use the meshless technique to compute local deformations. We use splatting techniques to render our point-based geometrical model. Recently, a number of researchers have demonstrated the efficiency of splatting for rendering geometrically complex objects [7,8]. Splatting is a technique for representing continuous texture function of the surface of point-based models. With each point a reconstruction kernel (called a splat) is associated, which is usually a Gaussian distribution. Therefore, a single point is mapped to multiple pixels, and the resulting color of the pixel is the cumulative of all the contributing pixel colors. We have introduced splatting techniques to simulate cutting in this paper.

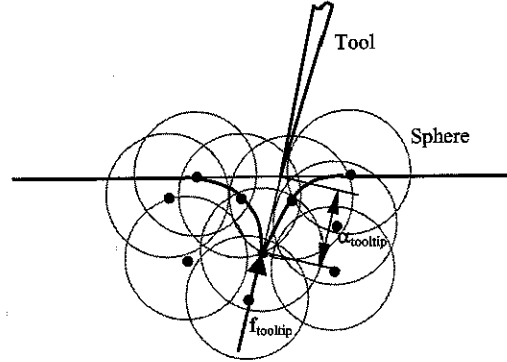


Figure 2. A laparoscopic surgical tool interacting with the surface of an organ. MFS nodes are distributed in the vicinity of the tool tip to obtain a localized discretization. $\alpha_{\text{tool tip}}$ and $f_{\text{tool tip}}$ are the prescribed displacement and reaction force at the tool tip, respectively.

In the next section we briefly introduce the MFS technique we have adopted in computing the deformation fields and reaction forces. In section 3 we discuss the technique we use to implement surgical cutting in the context of laparoscopic Heller myotomy.

2. The computation of deformation fields and reaction forces

In this section we briefly summarize the method of finite spheres. In this technique, we distribute nodal points around the surgical tool tip and define spherical “influence zones” around each node (see Figure 1). The approximation u_h of a variable u (e.g. displacement), using ‘ N ’ spheres, may be written as

$$u_h(\mathbf{x}) = \sum_{J=1}^N h_J(\mathbf{x}) \alpha_J \quad (1)$$

where α_J is the nodal unknown at node J . The nodal shape function $h_J(\mathbf{x})$ at node J is generated using a moving least squares technique

$$h_J(\mathbf{x}) = W_J(\mathbf{x}) \mathbf{P}(\mathbf{x})^T \mathbf{A}^{-1}(\mathbf{x}) \mathbf{P}(\mathbf{x}_J) \quad J=1, \dots, N \quad (2)$$

where

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

The vector $\mathbf{P}(\mathbf{x})$ contains polynomials ensuring consistency up to a desired order (in our implementation we have chosen $\mathbf{P}(\mathbf{x}) = \{1, x, y, z\}^T$ to ensure a first order accurate scheme in 3D, similar to bilinear finite elements). W_J is a quartic spline radial weighting function at node J . We assume linear elastic tissue behavior and satisfy the elasticity equations only at the nodal points to obtain the discretized set of equations

$$\mathbf{K} \mathbf{U} = \mathbf{f} \quad (4)$$

where \mathbf{K} is the stiffness matrix and \mathbf{f} is the vector containing nodal loads.

A node is placed at the tool tip ($\mathbf{x}_{\text{tooltip}}$) and we use a singular weighting function

$$\tilde{W}_{\text{tooltip}} = W_{\text{tooltip}} \|\mathbf{x} - \mathbf{x}_{\text{tooltip}}\|_0^{-p}; \quad (p > 0 \text{ is an integer and } \|\cdot\|_0 \text{ is the Euclidean norm}) \quad (5)$$

at this node to enforce that the displacement computed at the tool tip using equation (1) is the one prescribed at the tool tip ($\mathbf{U}_{\text{tooltip}}$).

The stiffness matrix in Eq (3) may be partitioned as

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{aa} & \mathbf{K}_{ab} \\ \mathbf{K}_{ba} & \mathbf{K}_{bb} \end{bmatrix} \quad (6)$$

corresponding to a partitioning of the vector of nodal parameters as $\mathbf{U} = [\mathbf{U}_{\text{tooltip}} \quad \mathbf{U}_b]^T$

where \mathbf{U}_b is the vector of nodal unknowns which may be obtained as $\mathbf{U}_b = -\mathbf{K}_{bb}^{-1} \mathbf{K}_{ba} \mathbf{U}_{\text{tooltip}}$.

The reaction force to be delivered to the haptic interface device is obtained as $\mathbf{f}_{\text{tooltip}} = \mathbf{K}_{aa} \mathbf{U}_{\text{tooltip}} + \mathbf{K}_{ab} \mathbf{U}_b$.

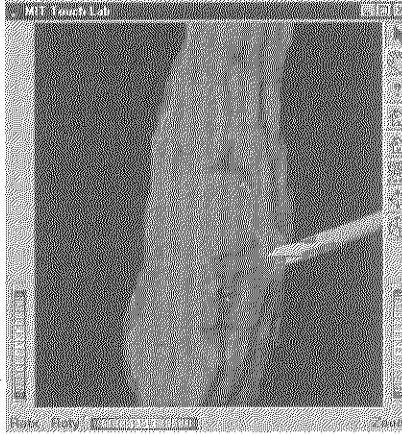


Figure 3. A snapshot of Heller myotomy simulation. Photorealistic textures are applied to the surface of an esophagus model.

3. Simulation of meshless cutting using splats

The concept of representing objects as a set of points and using these as rendering primitives has been introduced in a report by Levoy and Whitted [7]. Splatting, one of the recent image based volume rendering algorithms introduced by Lee Westover [8] in 1990, performs a front-to-back object-order traversal of the voxels in the volumetric dataset. Each voxel's contribution to the image is accumulated. Since this procedure is very similar to throwing a snowball (voxel's contribution to image) at a flat surface it is called splatting, where the amount of snow at the center of impact will be high and it drops off further away from the center. A reconstruction kernel determines the contribution of each voxel and the projection of this kernel into the image buffer is known as a footprint. The size of the footprint is proportional to the volume and the image to be generated and its shape is usually Gaussian.

This has the effect of increasing the contribution from voxels near the center of projection and reducing from those far from the center.

Use of points as geometric primitives is quite a deviation from the traditional use of triangular primitives that have connectivity information. Therefore, the primary challenge of point-based rendering is the reconstruction of this connectivity information from point clouds, i.e., reconstructing the continuous surface. The reconstruction algorithm should guarantee that there are no holes left on the surface [9]. While reconstruction algorithms guarantee that there are no holes left on the surface, cutting in point-based models involves deliberately creating discontinuities or holes by selecting regions of interest.

The first stage in the haptic rendering of cutting operation involves finding the point of contact or collision detection. Collision detection algorithms available in the literature are only for polygonal meshes. Haptic rendering or collision detection algorithms for point-based models are not known. We have developed a Z-buffer based [10] collision detection technique for point-based models.

To detect collision we shoot a ray from the camera to the tip of the force feedback device. The Z-buffer value at the intersection of this ray with the surface is used to decide whether there is indeed a collision. Collision occurs when: (a) the Z-buffer value of the tip is greater than that of the splats at the intersection while the back face is culled; and (b) the Z-buffer value is less than that of the splats at the intersection while the front face is culled.

Once collision is detected the tissue is deformed using the method of finite spheres. When the reaction force at the tool tip exceeds a critical value, the tissue is cut. Traditionally, cutting task is defined and simulated as splitting polygonal surfaces along a collection of marked edges and vertices [11]. Since point-based models do not have this connectivity information, we have developed special algorithms to find the neighboring points and move those points accordingly. After the tissue is cut, the layer beneath the surface is visible and the points are splat-rendered.

Figure 4 shows a schematic diagram of the point-based rendering paradigm and a snapshot of an esophagus model, generated using 4770 points and being cut using the scheme just described. Part of the code is used from Qsplat [9], a point rendering system that was designed to render large data sets produced by modern scanning devices.

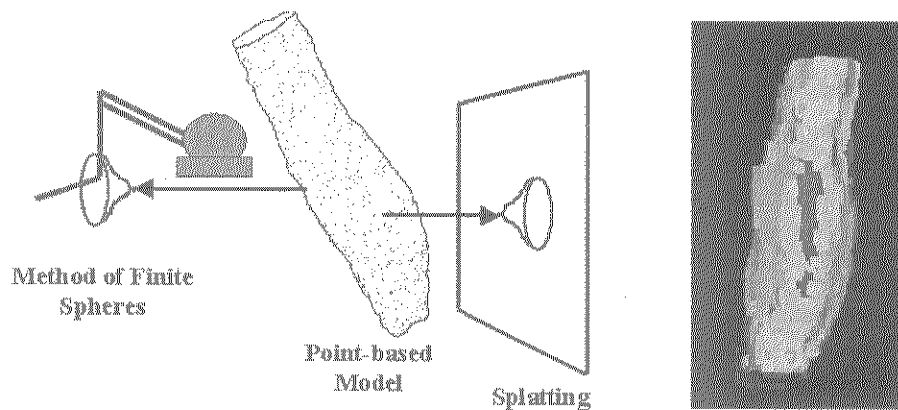


Figure 4. Schematic diagram of mashless cutting using splats (left). The actual cut esophagus is shown on the right.

4. Concluding remarks

In this paper we present a point-based paradigm for representing soft objects during multimodal medical simulations in general, and laparoscopic Heller myotomy, in particular. While a localized point based method of finite spheres is used to compute the deformation fields and reaction forces, a meshless cutting technique is implemented using splats. A new collision detection algorithm using the Z-buffer is also presented. The techniques are quite general and powerful. Further improvements in implementation may be achieved by coupling the localized method of finite spheres with a global discretization (which may be quite coarse) of the organ.

Acknowledgements: This work was supported by a grant from Harvard Center for Minimally Invasive Surgery.

References

- [1] C. Basdogan, C. H. Ho and M. A. Srinivasan, Virtual Environments for Medical Training: Graphical and Haptic Simulation of Laparoscopic Common Bile Duct Exploration, *IEEE/ASME Trans. on Mechatronics*, (2001) Vol. 6, No. 3.
- [2] S. De and M. A. Srinivasan, Thin Walled Models for Haptic and Graphical Rendering of Soft Tissues in Surgical Simulation, *Proceedings of MMVR 7 Conference* (1999) 94-99.
- [3] 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 MMVR 2002 Conference* (2002)
- [4] M. Bro-Nielsen and S. Cotin, Real Time Volumetric Deformable Models for Surgery Simulation Using Finite Elements and Condensation, *Proceedings of Computer Graphics Forum, Eurographics 96* (1996) 57-66.
- [5] S. Cotin, H. Delingette, J. M. Clement, V. Tassetti, J. Marescaux and N. Ayache, Geometric and Physical Representations for a Simulator of Hepatic Surgery, *Proceedings of MMVR 4 Conference* (1996)139-151.
- [6] S. De, J. Kim and M. A. Srinivasan, A Meshless Numerical Technique for Physically Based Real Time Medical Simulations, *Proceedings of MMVR 2001 Conference* (2001), 113-118.
- [7] M. Levoy and T. Whitted, The Use of Points as Display Primitives, *Technical Report TR 85-022, UNC at Chapel Hill, Dept. of Comp. Sc.* (1985).
- [8] L.A. Westover, Footprint Evaluation for Volume Rendering, *Proc. of SIGGRAPH Conference'91*, pp. 275-284, 1991.
- [9] S. Rusinkiewicz and M. Levoy, QSplat: A Multiresolution Point Rendering System for Large Meshes, *Computer Graphics (SIGGRAPH 2000 Proceedings)* (2000).
- [10] E.A. Karabassi, G. Papaioannou & T. Theoharis, Intersection Test for Collision Detection in Particle Systems, *ACM Journal of Graphics Tools*, 4(1), 1999, pp. 25-37.
- [11] C. Basdogan, C-H. Ho, and MA Srinivasan, Simulation of tissue cutting and bleeding for laparoscopic surgery using auxiliary surfaces, *Proceedings of MMVR (1999) Conference*, 1999.