Haptic Rendering of Volumetric Anatomic Models at Sub-voxel Resolution

Andreas Petersik, Bernhard Pflesser, Ulf Tiede, Karl Heinz Höhne

Institute of Mathematics and Computer Science in Medicine, University Hospital Eppendorf, Hamburg, Germany

petersik@uke.uni-hamburg.de

Abstract

In this paper, a new approach for haptic rendering of high resolution voxel-based anatomic models is presented. For visualization the surface location is determined by a ray-casting algorithm at sub-voxel resolution. Since the same algorithm is used for the haptics as well, a very high level of detail is achieved for haptic feedback. Both graphical and haptic representation is congruent.

The interaction forces are calculated based on a collision detection between an arbitrary sized sphere-shaped tool and an arbitrary complex anatomic model. Forces are calculated at an update rate of 6000 Hz and sent to a 3-Degree-of-Freedom (3-DOF) Phantom device [7]. Compared to point-based haptic rendering, the unique combination of the sphere-based approach in combination with sub-voxel rendering provides more realistic and very detailed tactile sensations.

1 Introduction

Today haptic rendering is mostly based on traditional computer graphics methods where objects are represented by polygons only. However, using a surface based representation to create a model for medical applications, the knowledge about the interior structures of the organs is lost. This knowledge is very important when simulating interactive cutting operations such as required for surgery simulations.

In addition we realized that today 3-DOF haptic rendering is mostly point-based, i.e. only one point is used to calculate collisions and forces. This induces several problems:

- Discontinuities (e.g. sharp edges) on the surface can lead to discontinuities in the haptic display
- The virtual tool can reach points which can not be reached by the simulated real world tool

Our approach uses a tool-based collision detection which was inspired by [1]. For collision detection and force calculation we used a sphere-shaped tool which simulates a drill as it is used in petrous bone surgery.

1.1 Previous work

The voxel-based approach to haptic rendering presented in [1] enables 6-DOF manipulation of a modestly sized rigid object within an arbitrary complex environment of static objects. One drawback of this method is that the haptic rendering is at voxel level accuracy only. Our approach differs from this work primarily in the surface representation. While our model is also using a voxel representation, the exact location of the surfaces is calculated by a ray-casting algorithm at sub-voxel resolution. That leads to smooth surfaces while even small surface details can be represented.

2 Methods

Mainly the following three aspects are involved in our approach:

1. The collision force on the surface of the objects can be computed by an algorithm which was inspired by [1].
2. To calculate correct forces for deeper tool-object penetrations, a modified proxy algorithm [4,5] was used.
3. In contrast to other representations, our model does not include an explicit representation of the surface. It is rather calculated by a ray-casting algorithm at sub-voxel resolution [2]. Such calculations are computationally expensive and thus they should be minimized to the extend absolutely necessary.

2.1 Representation of the tool

The tool is represented by a number of points which are distributed at preferably equal distances over the tool surface, together with the inward pointing surface normals (Fig. 1). In our case all inward pointing normal vectors are ending at the center point of the sphere. To get an adequate representation of the shape while reducing computation to a minimum, we used 26 sample points on the surface. Depending on the size of the sphere more point samples might be required.

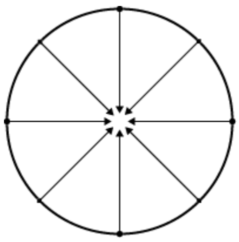

Fig. 1: Representation of the tool

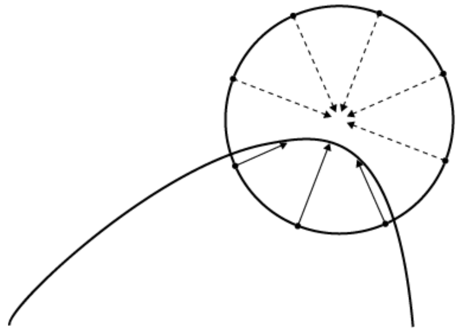

Fig. 2: Calculation of collision force

2.2 Haptic rendering of the surface

When a collision occurs, two parameters of the collision force must be calculated:

- Direction and
- Magnitude of collision force

First of all, the surface points of the sphere are checked, whether they are inside or outside the object. All surface points which are inside the volume are traced in direction of the inward pointing normal until the surface is found (Fig. 2) or the sphere center point is reached. All found vectors are added and the direction of the resulting vector is the direction of the force vector which must be applied to the haptic device. The problem of calculating the magnitude of the force vector is still under investigation; so far, we use a heuristic method which averages the magnitudes of the surface vectors. This leads to deeper penetration while more tool points are in contact with an object.

2.3 Proxy object algorithm

Any haptic device has a limited force which can be applied to it. When the user pushes harder, it is possible that the physical position of the tool immerses completely into the virtual model. In this case a calculation of the force direction as described above is not possible anymore because the sphere center point is in the object itself. To overcome that limitation a modified proxy object [5,6] was implemented. In free space, the position of the proxy object and of the device is identical. When the haptic device moves into an object, the proxy remains on the object’s surface. When the proxy position is known, the resulting force vector is proportional to the difference vector between the proxy and the device position.

In order to update the proxy position while the device is moving, the distance between the device position and the proxy must be locally minimized by regarding the surface constraints. Since searching for the local minimum would be computationally very expensive in our model, a new algorithm was implemented. Whenever more than a certain number of surface sample points are in contact with an object, the way back to the proxy is traced until the number of contacts is below the limit or until the proxy is reached. At that location the force vector is calculated as described above and the proxy is set to that position.

2.4 Implementation

The system was implemented on a Compaq SP750 workstation. The workstation has two Pentium III Xeon processors at 866 MHz and is equipped with 2GB RAM. As haptic device we used a 3-DOF Phantom Premium 1.0A [7]. Our system was running under SuSE Linux 7.1. For connecting the device directly to the system, we used the open-source PHANToM Linux-driver from [4].

The haptic process runs at an update rate of 6000Hz. The most time consuming calculation is the calculation if a point is in contact with the objects. By using 26 sample points on the sphere, 60 such calculations are executed at every haptic update on average.

3 Results

With our approach we achieve high quality haptic rendering of arbitrary complex anatomic models [9] as shown in figure 3. The perception of spatial relationships is greatly enhanced with haptic feedback. Even very small surface details or small objects like nerves and vessels can be sensed realistically due to our sub-voxel based approach. As a major application of our method we are working on a system for the simulation of middle ear surgery (fig. 4). Here, the haptic device is used to simulate drilling into the mastoid bone [8]. Using a sphere-based collision detection proved to be much more realistic than using point-based haptic interaction. Drilled holes and tubes in the bone could be intuitively probed with the haptic device.

4 Conclusions and future work

The approach presented here overcomes several problems of point-based haptic rendering. The shape based collision detection makes interactions more realistic, especially for the use in surgery simulations. Another important advantage is that surface details can be sensed as expected from the graphical representation. Sub-voxel rendering leads to both a graphical and haptical realistic and high detailed display.

While the current implementation is limited to sphere-shaped tools, future implementations will extend this to more general shapes. The calculation of the collision force magnitude will be investigated further to improve haptic rendering at locations with many tool object intersection as they appear in deep clefts.

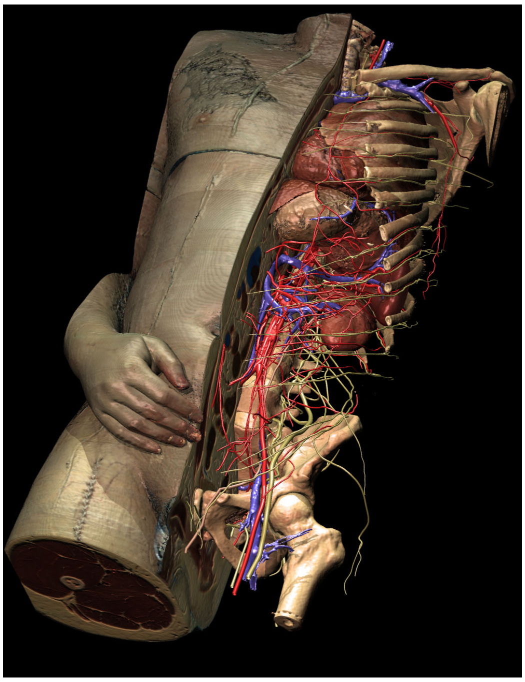

Fig. 3: Sub-voxel rendered anatomic model

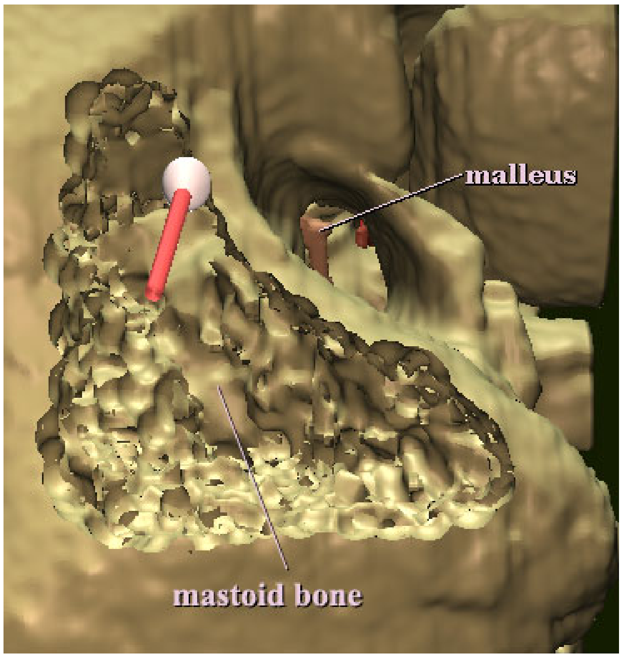

Fig. 4: Simulation of middle ear surgery

References

1. William A. McNeely, et al. Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling. Computer Graphics (SIGGRAPH ’99 Proceedings), pages 401-408
2. U. Tiede, T. Schiemann, K. H. Höhne: High quality rendering of attributed volume data. In David Ebert et al. (eds.): Proc. IEEE Visualization '98, IEEE Computer Society Press, Los Alamitos, CA, 1998, 255-262.
3. PHANToM Driver for Linux: http://decibel.fi.muni.cz/phantom/
4. D.C. Ruspini, K. Kolarov and O. Khatib. The haptic display of complex graphical environments. Proc. of ACM SIGGRAPH, pages 345-352, 1997
5. C. Zilles and K. Salisbury. A constraint-based god object method for haptics display. In Proc. of IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems, 1995.
6. K.H. Höhne, M. Bomans, A. Pommert, M. Riemer, C. Schiers, U. Tiede, G. Wiebecke: 3D-visualization of tomographic volume data using the generalized voxel-model. Visual Comput. 6, 1 (1990), 28-36.
7. SensAble Technologies Homepage: http://www.sensable.com
8. B. Pflesser, U. Tiede, K. H. Höhne, R. Leuwer: Volume based planning and rehearsal of surgical interventions. In Heinz U. Lemke et al. (eds.): *Computer Assisted Radiology and Surgery, Proc. CARS 2000*, Excerpta Medica International Congress Series 1214, Elsevier, Amsterdam, 2000, 607-612.
9. K.H. Höhne, B. Pflesser, A. Pommert, M. Riemer, R. Schubert, T. Schiemann, U. Tiede, U. Schumacher: A realistic model of human structure from the Visible Human data. *Meth. Inform. Med. 40*, 2 (2001), 83-89.