International Conference on Control, Automation and Systems 2008
Oct. 14-17, 2008 in COEX, Seoul, Korea

Volume-based Haptic Model for Bone-drilling

Kimin Kim[1] Ye-Seul Park[2], and Jinah Park[3]
[1, 2, 3] Information and Communications University, Daejeon, Korea
(Tel : +82-42-866-6257; E-mail: kiminkim@icu.ac.kr, ystopia@icu.ac.kr, jinah@icu.ac.kr)

Abstract: For bone surgery haptic simulation, we need to consider both volume cutting deformation and stable force feedback. In this paper, we propose a volume-based haptic model where both bone and a tool are represented by voxel-based models so that we can handle shape deformation of bone efficiently. For fast and stable haptic rendering, we propose that the tool is represented by a signed-distance field and the bone surface is tracked by a set of points, called a point shell. In order to handle shape deformation of bone while drilling, fast reconstruction of point shell points are performed locally based on the voxel properties of bone and the tool.

Keywords: haptic rendering, voxel sampling, volume manipulation, surgery simulation

1. INTRODUCTION

Recently, with advances in computing environment as well as in haptic device technology, many researches on medical simulation with haptic feedback are proposed. Since the characteristics of bone-related surgery such as craniofacial, maxillofacial, or dental operation are well suited with haptic application, bone surgery simulation is receiving increasing attention.

Since bone surgery involves in removal of some portions of bone in volume, the bone is naturally represented by voxel-based models and the haptic feedback force is computed directly from the intersecting volume data [1][2]. Although this direct rendering allows a fast computation, it suffers from unstable feedback force. The instability comes from the fact that the computed feedback force is sharply changed when the interaction point enters a voxel or leaves from a voxel. Another problem source is the unpredictable amount of intersecting volume data which can fluctuate while user moves around the haptic device. It may well exceed the maximum renderable stiffness of the device, and cause undesirable buzz.

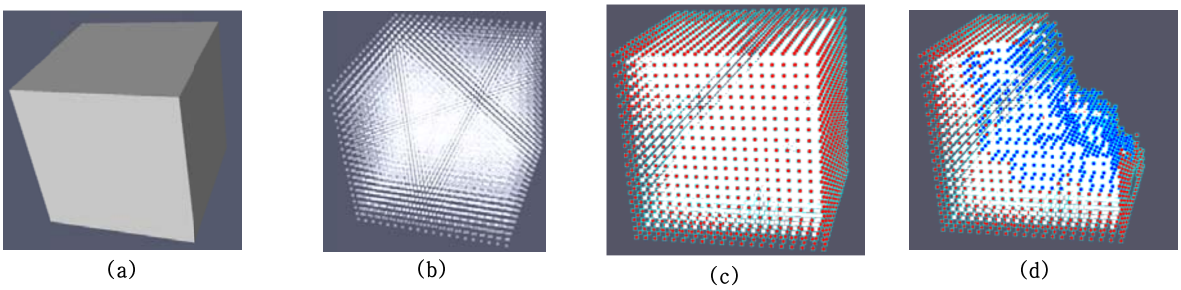
Alternative approach to direct rendering is to compute the feedback force indirectly from the surface which is reconstructed from volume data [3][4]. This approach generates smooth feedback force, but it is often cumbersome and computationally intensive to reconstruct the surface locally as bone parts are removed. Moreover, since the cost of reconstruction depends on the number of contact polygons, it is even harder to meet the haptic cycle under multi-contact.

The important part of bone surgery simulation is the choice of the bone representation so that it can handle both the visual effect of bone removal and the fast computation of stable haptic feedback. Our goal is to generate smooth and stable feedback force for bone surgery simulation during bone drilling. We propose to apply virtual coupling[5] onto the voxel-based haptic rendering approach which also handles shape deformation (i.e. bone part removal) in haptic cycle.

2. METHOD

Our approach is inspired by the method of Barbič et al.[6] which resolves penalty-based contact forces between two objects using a point-based representation (a point shell) for one object and a signed-distance field for the other. In their method, a soft object with reduced degree of freedom is represented by the point shell. In bone drilling application, however, the object being deformed (bone) is a hard object and the degree of freedom for its deformation (cut-away) is unlimited. Therefore, the reduced deformable model in [6] is not a suitable representation for bone drilling simulation.

In order to handle bone part removal efficiently, we use voxel-based models for both bone and the tool. Moreover, for stable haptic rendering, the bone surface is tracked by a point shell [7], and the tool is represented by signed-distance field. The contact force is approximated by a penalty-based method, which utilizes the point shell points of bone and distance field of the tool. After resolving the contact force, we calculate the feedback force using static virtual coupling [6], which reduce the force discontinuity originated from the limited voxel resolutions.



(a) (b) (c) (d)

Figure 1.(a) A cube model; (b) Density of the cube; (c) Point shell of the cube; (d) Changes of density and point shell

2.1 Bone representation

Bone to be drilled is a volumetric object which consists of voxels. Each voxel has three properties – density, stiffness, and a voxel type. The density shows the amount of remain of bone for each voxel, and is used for visualization. The stiffness property of each voxel is used for determining the magnitude of feedback force. The voxel type describes status (i.e., free space, surface element, or interior) of each voxel. We use this property to estimate surface of bone to regenerate a point shell upon drilling. From the volumetric bone model, we preprocess volume data to generate initial point shell points on the surface. For each point, its surface normal is also computed so that it is possible to obtain the direction of force as fast as possible when a tool collides with the bone.

The other approach that we represent bone is to use minimum distance as a property of each voxel. Minimum distance of voxels of bone is an exact way to obtain penetration depth in the situation when the bone is not removed by drilling. But in our bone drilling simulation, the bone part removal should be handled and remains of bone need to be updated into the new bone for controlling. In that case, the voxel of bone with minimum distance takes much time complexity for updating itself. Therefore, our approach uses a property, density. To update an alternative bone with density is an efficient way for changing the point shell which will be covered in 2.4 section.

As Figure 1 shows, in an alternative bone as a volumetric cube (a), the density (b) and point shell (c) of the cube are to be changed as a result of bone removal, drilling.

2.2 Tool representation

In the real bone surgery, surgeons operate on a patient using a variety of tools, which have different shapes and sizes according to their objective. For bone surgery simulation, various tools can be used such as Figure 2 representing the tip of each burr, condenser and drill from top to bottom as well as its color representation of distance field.

For bone-drilling, the tip of burr is modeled as a volumetric object. In our approach, each voxel has two properties – occupancy and minimum distance from a surface. The occupancy property of the tool volume is used against the density property of the bone for bone part removal simulation. The signed distance field is used for fast computation of the penalty force by sampling the distance value of voxel without calculating the penetration depth of a tool. The most significant difference between previous work on bone surgery simulation and our approach is that we apply the distance field to the tips of tools rather than bone. This leads that the collision detection and penetration depth computation are simple and easy to implement compared to mesh-based approaches.

The distance field is generated from Closest Point and Distance Transform algorithm proposed by Mauch et al.[8] (Figure 2 right). In case of a burr, a sphere is set as a tip. A cylinder is proper to be the tip of a condenser as well as a cone is a simplification on shape of the tip of the drill. Particularly, the cone includes the locus of the drill bits in order to detect the collision.

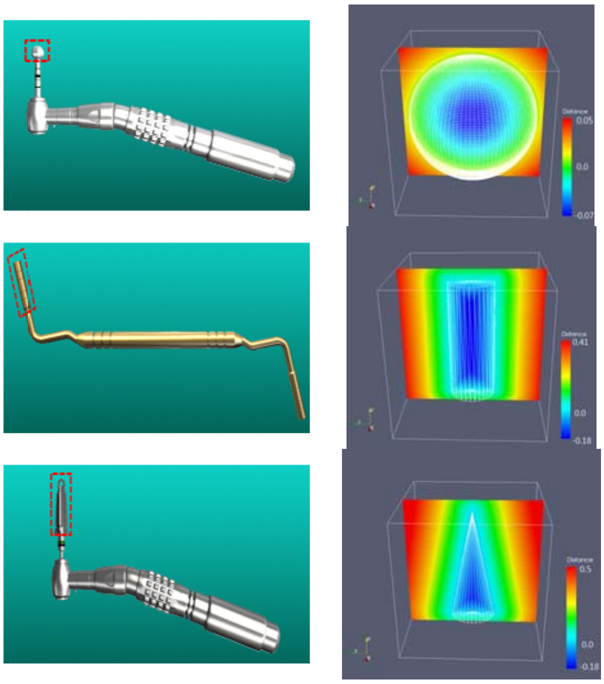


Figure 2.(Left) 3D models of tools (Right) Distance field of their tips, dotted circle in left models.

2.3 Voxel-based haptic rendering

Our haptic rendering algorithm do not consider surface model but only volume model. Given volume model of bone and distance field of tool, we calculate the feedback force to be conveyed to a user.

At first, we get the manipulandum position and orientation. Then the following steps are executed during haptic cycle; 1) collision detection 2) contact force computation 3) virtual coupling force computation 4) Finding a new position of virtual tool 5) Feedback force computation. The following sections will describe theses steps in detail.

2.3.1 Collision detection

The bounding box of tool is updated as the simulation object move and bounding box overlap between bone and a tool is performed. If the bounding boxes overlapped, every point shell of bone is queried against the distance field. However, the point shell and distance field have different coordinate system. Thus, every point shell point in world coordinate is transformed into volume space of tool using the following equation.

$$P_v = (V_{Num} \ - 1\ ) \times \frac{(P_x - V_{min})}{V_{size}}, \qquad (1)$$

where the P_v and P_x are the volume coordinates of tool and the position of point shell, respectively. The range of this value is [0, … V_{num}]. V_{min} and V_{size} are the minimum value of volume and the size of the volume in world coordinate, respectively.

At transformed position, P_v, we query value of distance field by using trilinear interpolation. Since we applied signed distance field for tools, a negative sign of this value means that objects are collided. If we detect the collision between bone and a tool, contact force is calculated.

2.3.2 Contact force computation

Once collision is detected, the following equation is used to calculate the contact force, which is summation of penalty forces of point shell points in contact:

$$F_c = \sum F_i = \sum k_i d_i N_i, \qquad (2)$$

where k is the stiffness value and d is a distance field value. Both k and d are interpolated from the 8-neighber voxels near each point for better precision. N is the normal vector of the point shell point.

Since we can get the smooth feedback force only in the case of shallow penetration of a tool, the contact force cannot be conveyed to the device directly. Fig. 3 shows this problem. When the tool is in free space, no force is generated (a). As the tool moves the inside of bone, the magnitude of sampled distance field value is increased compared with previous one. And then feedback force is increased. (b)-(c). But if the tool penetrates bone over the medial axis of distance field, the feedback force is reduced.(c)-(d). Moreover, as the distance field pass through bone, we cannot detect the collision(d). And no feedback force will be generated.

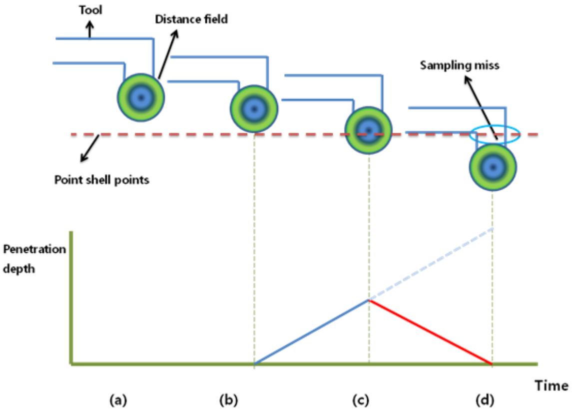


Figure 3. A problem of direct rendering with distance field

Notice that depending on the number of point shell, it may well exceed the maximum renderable stiffness of the device. This can easily make a device instable. To address this problem, we adopt the indirect method, called virtual coupling. Virtual coupling is a spring connected between the manipulandum and the simulation object. Since feedback force is only involved in this spring force, we can stabilize the feedback force by adjusting the parameter of this spring.

2.3.3 Virtual coupling computation

For the virtual coupling, we should trace the position of manipulandum($P_{manipulandum}$) and the position of tool in simulation world(P_{tool}), separately. Then the virtual spring is connected with them. The following equation shows this virtual coupling force.

$$F_{vc} = k_d(P_{manipulandum} - P_{tool}), \qquad (3)$$

where we calculate the $P_{manipulandum}$ by scaling the position of haptic device in accordance with the size of simulation world, the P_{tool} by solving net force equilibrium equation. The following section will describe how to find the position of the tool(P_{tool}).

2.3.4 Finding a new position of virtual tool

When the summation of the contact force(F_c) and the virtual coupling(F_{vc}) is zero, called net force equilibrium, the simulation object is anchored. If a user moves the manipulandum, this equilibrium status will be broken. The simulation object will move to the new position where the net force is zero. If we consider both positional rotational force, this relation can be represented 6x6 linear system[6]. However, we only consider the positional force in this paper. The following equation show this relations.

$$F_c + F_{vc} + \left(\frac{\partial F_c}{\partial x} + \frac{\partial F_{vc}}{\partial x}\right)\Delta x = 0, \qquad (4)$$

where Δx is displacement of the position of simulation object. $\frac{\partial F_c}{\partial x}$ and $\frac{\partial F_{vc}}{\partial x}$ is the gradient of contact force and virtual coupling force with respect to Δx respectively. Since the x, y, z coordinate involved in equation (3), we get the 3 linear equations. Every haptic cycle, Δx is resolved by Gaussian elimination. Then we find new position of simulation object by adding Δx to current position of simulation object.

2.3.5 Feedback force computation

Once the position of simulation object is calculated by solving the linear equation system of (3), we finally compute the feedback force, which is conveyed to a user. The feedback force is as same as the opposite force of virtual coupling force between new simulation position and position of manipulandum (Newton's 3rd law). The important matter on calculating feedback force is that the maximum exertable force and maximum renderable stiffness should be considered. The haptic device we used is PHANTOM Premium 1.5, which has the maximum exertable force of 8.5N and the maximum renderable stiffness of 0.6 N/mm. In order to generate stable feedback force, we should limit magnitude of feedback force below 8.5N. and the stiffness of the virtual coupling below 0.6. In our system, the magnitude of feedback force is limited to 8.5N. And we got the stable force while the k_d value was below 0.64.

2.4 Volume cutting

To handle shape deformation of bone while drilling, the volume cutting procedure is performed at every haptic cycle. The first step of this procedure is to find overlapped voxels by comparing the occupancy value between bone and a drill (Fig 1.), and then we apply difference Boolean operation to the density of bone with the occupancy of the tool. In accordance with the density of bone, the voxel type of bone also is updated. Finally, the point shell is adjusted. The blue dots in Fig, 1 (mid) are the point shell points before cutting. The point shell points on removed part of bone are deleted. (The open dots in Fig. 1 (right) are the deleted ones.) The hole in the point shell is filled by using a flood-fill algorithm. We consider the collided voxels as the seed voxels, and check 26-adjacent voxels around each seed voxel. If the voxel type of an adjacent voxel is surface element, we generate a point shell point. The blue dots in Fig.1 (right) are the new points generated.

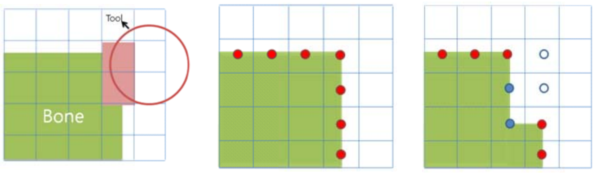


Figure 4. Schematic illustration of the volume cutting procedure

3. EXPERIMENT

The simulation have been tested on a Intel® Core™ 2 Duo CPU E8400 @ 3.00 GHz processor PC with a GeForce 8800 GTS graphics card. We have developed our system in Windows platform with CHAI 3D[9], libraries for computer haptics. And Phantom premium 1.5 was used to display the feedback force. We have conducted a preliminary experiment on simplified 3D bone and a drill as shown in Fig. 4. The volume resolutions, for bone and a tip of a tool, are 24x24x24 and 12x12x12, respectively. The voxel sizes for bone and a tool are 0.05 and 0.02, respectively.

In order to cover the entire bone, 2,168 points are generated. As bone was drilled, several points was generated and deleted. The maximum number of points reached up to about 2,300. Haptic rendering and volume cutting was processed within 1 kHz. And visual rendering ran at 60 Hz. Figure 5 shows that we were able to generate continuous and stable haptic feedback by using the method proposed in this paper. When we were drilling, the feedback force fluctuates because Boolean operation eliminates the collided voxel immediately as shown in Figure 6. This result indicates that we need another force interpolation algorithm when the drill eliminates the one voxel.

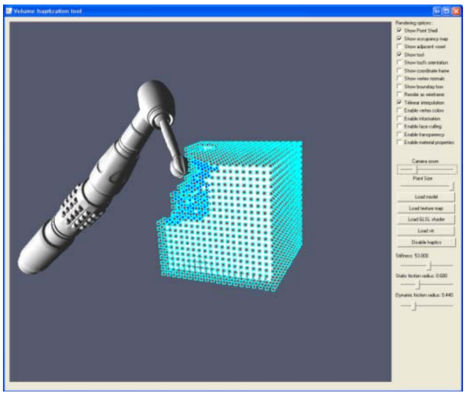


Figure 4. Screen shot of our system during bone removal: square block is volumetric bone covered by point shell points.

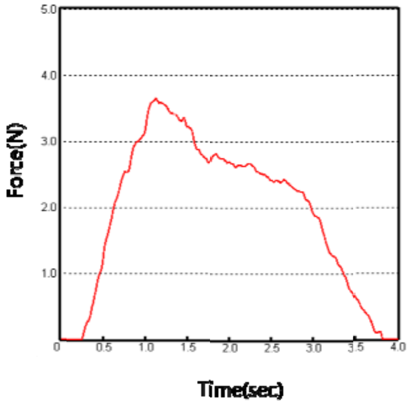


Figure 5. Feedback force from the time when a drill contacted with bone to the time when the drill separated from bone.

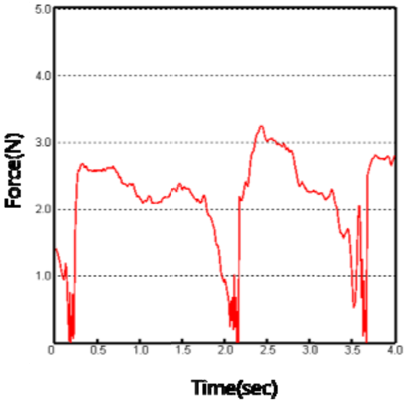


Figure 6. Feedback force while drilling

4. CONCLUSIONS AND FUTURE WORK

For bone drilling simulation, we have described a volume-based haptic model. To calculate the contact force as fast as possible, bone is covered by a point shell and tool is represented by distance field. We also adapt the static virtual coupling for stable force generation. Finally, volume cutting is implemented by applying Boolean operation to the density of bone with the occupancy of the tool, and the deformed shape is tracked by applying flood-filling algorithm.

This research is the partial work of dental implant simulation system to be developed. We aim at realistic haptic feedback of various drills and a condenser in order to provide dentists with surgical training and education. Currently, we are investigating the data structure, which can manage frequent addition and remove of point shell points. And we are going to append polygonization technique to our system that enables the volume model to convert the surface model. The important future work is to model accurate force between bone and drill bits with regard to resistance and vibration forces of drill.

ACKNOWLEDGEMENT

This research was supported by the Ministry of Information and Communication, Korea, under the Information Technology Research Center support program supervised by the Institute for Information Technology Advancement (IITA-2008-C1090-0804-0002)

This work was also supported by the MKE(Ministry of Knowledge Economy), under international joint R&D program(No. 10030198)

We would like to thank CEO Jai Kim at PRIMPO CO., Ltd. for supporting 3D models of dental instruments.

REFERENCES

[1] D. Morris, C. Sewell, N. Belevins, F. Barbagli, and K. Salisbury. “A collaborative virtual environment for the simulation of temporal bone surgery”, Proc. MICCAI, 2004.

[2] M. D. Tsai, M. S. Hsieh, and C. H. Tsai, “Bone drilling haptic interaction for orthopedic surgical simulator”, Computers in Biology and Medicine, 1709-1718, 2007

[3] A. Petersik, B. Pflesser, U. Tiede, K. H. Hohne, and R. Leuwer. Haptic volume interaction with anatomic models at sub-voxel resolution. In IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 66-72, 2002

[4] M. Eriksson, and H. Flemmer, and J. Wikander, “A Haptic and Virtual Reality Skull Bone Surgery Simulator”, Proc. World Haptics, 2005

[5] J.E. Cogate, M.C.Stanley, and J.M.Brown. Issues in the Haptic Display of Tool Use. In Proc. Of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Pages 140-145. IEEE, 1995.

[6] J. Barbič, and D. James, “Time-critical distributed contact for 6-DoF haptic rendering of adaptively sampled reduced deformable models”, Proc. SCA, 171-180, 2007

[7] W. McNeely, K.D Puterbaugh, and J. J. Troy. “Six degree-of-freedom Haptic Rendering Using Voxel sampling”, Proc. ACM SIGGRAPH, 401-408, 1999

[8] Mauch Sean, A Fast Algorithm for Computing the Closest Point and Distance Transform, Technical Report, http://www.its.caltech.edu/~sean/

[9] www.chai3d.org