VOLUME BASED CUTTING FORCE SIMULATION FOR MUSCULOSKELETAL SURGERY

[a]Ming-Dar Tsai, [b]Shyan-Bin Jou and [c]Ming-Shium Hieh

[a]Institute of Information and Computer Engineering,
Chung Yuan Christian University, Chung-Li, Taiwan/R.O.C.
Email:tsai@ice.cycu.edu.tw

[b]Department of Information Management Technique
Nanya Institute of Technology, Chung-Li, Taiwan/R.O.C.
Email:jou@earth.ice.cycu.edu.tw

[c]Department of Orthopaedics and Traumatology,
Taipei Medical College Hospital, Taipei, Taiwan/R.O.C.
Email:shiemin@mail.tmc.edu.tw

ABSTRACT

Volume based surgical simulation typically provide visual verification of surgical plans; however, cutting force prediction allows a surgeon to verify feed rates and paths of surgical tools. In this paper, we present several force computation models used in musculoskeletal surgeries to evaluate bone cutting. Forces on surfaces of a virtual surgical tool are computed to simulate the instantaneous force on the tool that can be oscillating or rotating. The virtual tool is triangulated and rendered with isosurfaces of tissues. Overload surfaces of the tool are highlighted to indicate abnormal feed rate and moving paths of the tool.

1. INTRODUCTION

The main purpose of surgical simulators is considered as predicting geometric changes of anatomic structures during performing surgical procedures. However, evaluating physical properties is also important for many surgical fields. Currently, several surgical simulators evaluate tissue elasticity or plasticity to compute tissue resistance (e.g., [1-4]) that helps detecting and then preventing overload on anatomic structures and damages on tissue surfaces [5,6]. Most simulators manipulate a deformable surface model to evaluate physical constraints by a finite element method [1,2] or mass-spring model [3,4], and simultaneously compute geometric deformation for visualization.

Contrast to the tissue elasticity or plasticity, musculoskeletal surgery use removing operation to cut rigid bones and bring topology rather surface changes on anatomic (bone) structures. Therefore, geometric simulation for musculoskeletal surgeries include the following topology changes: sectioning a bone, identifying whether a bone is divided as separate two, fusing separate bones together, removing a bone and repositioning a bone. Instead of surface models, volume model is considered suitable to simulate musculoskeletal surgeries because it contains interior not only surface information. Three approaches of manipulating voxel-represented objects have been discussed. One approach extends the contents of each voxel; for example, 6 links were employed to represent relations between a voxel and its six-face neighbors [7]. Local manipulations such as sectioning a structure or fusing two structures are easily implemented by adding and deleting the links. Another approach uses 2D pointer array to manipulate a structure [8]. The array element is a list of triplets, each of which records the depth of a boundary voxel of the structure, a unit normal vector and 6 facing codes indicating which voxel faces of the voxel are boundary. Using the surface normal and face codes can quickly obtain a 3D image. Adding a translation to the depth of the boundary voxels in the lists, the structure in the pointer array can be translated along the depth direction. The last approach assigns a structure code and face codes to a voxel to represent structure behavior [9]. An extended 3D seed and flood algorithm was introduced to identify, reposition and remove a set (structure) of the same code voxels, and join multiple structures. Distance-levels on each voxel were used to model isosurface changes during structure manipulation [9].

Two aspects of research have discussed physical properties in volume modeling. One considers that there exist spring links connecting voxels and computes structure deformation by manipulating the links [10]. This is suitable to simulate tissue deformation such as stomach or gallbladder deformation. The second aspect of research focuses on computing the resistance on a point inside a volume. This can be used to obtain interactive haptic feedback in volume modeling [11,12,13], and manipulate individual atoms in molecular modeling [14]. This can be also used to simulate a virtual surgical tool that operates on anatomies, such as knee palpation [15] or head section [12]. The one-point simplicity for force computation may compute every quickly, however is not precise to simulate the musculoskeletal procedures that use large area of tool surface

to remove bones.

The goal of this research uses a material removing mechanism to compute resistance on local area of a tool during sectioning and drilling bones. There are several reasons to do this research. A bone structure is usually complex and contains several kinds of tissues (cortical or cancellous bone, or tumors) with different stiffness and hardness. The load on a tool may change abruptly, even the tool sections or drills in the same speed. The tool may section in an oscillating way or drill in a rotating way. The instantaneous load of some oscillating or rotating angle on some flutes of the tool may overload even the total force to the tool is comfortable. This over-loads on a flute or the tool may induce undesired results (e.g., cracks of tissues or tumors in a bone, a tool break).

In this study, we introduce the machining models that are already applied in CAM (Computer Aided Manufacturing) systems for evaluating machining forces of rotating tools [16,17]. By manipulating the volume data, we compute the geometric changes of bone structures to obtain the contact conditions of the tool with the bone structure. Then, using the contact and feed rate data of the tool, and tissue data, we can compute the load on the tool and the bone structure. In the following, Section 2 introduces the system structure and describes briefly the geometric simulation methods by manipulating volume data. Section 3 introduces the force computation models for a rotating drill, an oscillating saw and a general tool. Section 4 shows a simulation example of knee surgery wherein a surgeon intends to remove a tumor and insert a graft bone instead. He first uses an oscillating saw to section a window-shaped bone for opening the tibia, then uses a dissector to section a soft tumor, and last use rotating drills to fix the fixture for fusing the returned window bone and the tibia. The example shows the overload part or flutes of the saw and the instantaneous force of the rotating drill.

2. SYSTEM ARCHITECTURE

Figure 1 shows the system architecture. A user wears a shuttle eyeglass to observe stereographic images and uses a surgical instrument attached with a 6 dimensional degree tracker to interact with the system. The system includes the interface module, data conversion module, isosurface reconstruction module, rendering module, (geometric) simulation module and force computation module.

2.1 Interface Module

The interface module provides virtual tools, menus and data bars. Through the menus, users can choose a volume to simulate, determine a simulation function to operate, input bone grafts and prostheses that have been designed by a AutoCAD system and change parameters of the shading model about light and material properties. Through the bars, the users can determine perspective conditions in-conditions including viewing position and angle, parity of two eyes.

The tracker is attached to one end of a surgical tool to simulate a virtual instrument. Based on the position and attitude of the tracker and the geometric data of the tool, the system can compute the intersections between the tool with the volume for simulating geometric changes of anatomic structures and computing forces on the tool.

2.2 Data conversion module

For manipulating voxel-represented objects, every voxel is assigned three 6-bit distance-levels to simulate tissue surface changes, six 1-bit face codes indicating whether the voxel faces are boundary and one byte indicating the tissue type and structure number. Totally, 4 bytes of memory are used for each voxel. Bone grafts and prostheses are designed by the AutoCAD system first, then converted to our voxel-represented objects.

Currently, the system has dealt with gray-value volumes from CT (computer tomographic) slices and bordered volumes from MRI (magnetic resonance imaging) slices. The conversions are effective in these volumes. Manipulating the three distance-levels and six face codes can also simulate changes of tissue surfaces. These manipulations are employed in the below “section” and “fusing” simulations.

2.3 Isosurface reconstruction module and rendering module

The distance-levels are used computed sample points of tissue surface (isosurfaces). In contrast to thresholding techniques that determine a sample point on an isosurface by one over-threshold voxel and one under-threshold voxel, one distance-measured voxel can determine a sample point [18]. In our system, a distance-level is interpreted as a sample point on a main axis. It is convenient in using the marching cube algorithm to obtain triangulated isosurfaces by the sample points on the main axes [19].

All tools are triangulated to obtain a series of vertices associated with surface normals. These polygonal tools and isosurfaces of tissues are rendered with the OpenGL libraries. Although the isosurfaces need not to be reconstructed when changing perspectives, it may take tens of seconds for reconstructing the isosurfaces after simulating a surgical procedure. Therefore, we only reconstruct the isosurfaces of operated part of a volume to achieve fast response. Although the tool may visually intersect with the anatomic structure, it did not remove any voxel of the structure. The system removes part of the structure by recognizing the sectioned part and removing the part after a series of sections as described below.

2.4 Geometric simulation module

The following simulation functions are extended from the

algorithms reported in [9].

The "section" function first interprets the positions and attitudes of a tracker as swept sectioning surfaces, computes distance-levels for sectioned boundary voxels, and assigns a structure code to the voxels. The "recognition" function uses an efficient 3D seed and flood algorithm to assign the voxels the same structure code inside a closed boundary composed of voxels with the structure code (sectioned boundaries) and voxels with different tissue codes (natural boundaries). Different from straightforward seed and flood algorithms, voxels along X-axis are directly traversed and not stacked in the algorithm [20]. Therefore, voxels for recursive are considerably reduced.

The "removal" function sets all voxels of a structure as air voxels that can also be implemented by the 3D seed and flood algorithm. The "fusion" function re-recognizes one anatomic structure (separate bone, prosthesis or bone graft) into another to join them into together. The structures may contact each other and no new structure voxels are generated. New structure voxels may be generated to help fusing. In this case, this function generates closed boundary voxels between two user-specified curves on fusing structures. Sometimes, bone grafts are implanted to help fusing. The system then recognizes the voxels inside the new boundary voxels with the old structures as one structure.

The "collision test" function detects collision among bones, prosthesis, vessels and the nervous cord. He proposed an efficient collision detection method that maps all objects into a map of regular cells, then detects collisions if objects exist in other object's spaces [21]. This grid intersection method is not adopted to detect collisions herein because the distance when a structure is placed onto another structure in a "fusing" simulation must be computed during collision test. We adopt an efficient ray traversal algorithm to detect the collisions (whether bone or nerve voxels exist on the path of a moving anatomic structure or surgical instrument). This algorithm is the most efficient one because it has the fewest additions and comparisons [22].

The "reposition" function translates all voxels of a structure to another position by first implementing a "collision test" to detect collisions, than pushing the structure into a series of stacks and clearing the structure by the seed and flood algorithm before popping the structure to a new position. The three components of the translating vector are not limited integers. That means the system allows an unaligned translation that usually occurs when a structure is moved along the slice direction.

2.5 Force computation module

Although the force on the tool and geometric changes of the anatomic structure actually occurs at the same time, we compute the cutting force after implementing a geometric simulation (for section or fixation). The force computation evaluates the force by the volume data; however, it does not affect the volume model such as geometric simulation did.

Our force computation models approximate a tool as finite surface elements. We use a (contact) point to represent a surface element. A contact point is contacting if it is inside any voxel of the anatomic structure. We consider the tool is removing tissues at a surface element if its contact point is contacting, then a force computation for the surface element computation is implemented. After all contact points of the tool are computed; the load on part or whole of the tool can be obtained by summing the force from all surface elements.

Contact points of a simple-shaped tool can be determined by the attitudes of the tool. For a complex tool, we use the vertices of polygons for rendering as the contact points. In Section 3, we introduce the force computation models of two simple-shape tools: oscillating tools for sectioning and rotating tools for drilling, and the model of the general non-simple tool.

After the force computation, the overload vertices or flutes of the tool are highlighted to warn uncomfortable results of the surgical procedures. The system can also show the instantaneous forces of oscillating or rotating tools at another window.

3 FORCE COMPUTATION MODEL

The force acting on a tool is the sum of outside forces such as gravitation, the friction force retarding the moving of the tool, the stiffness force normal to the tool, and the shear force removing tissues. When the tool is removing tissues of bone especially in high speed, the last two terms of forces are large enough to omit others.

3.1 Force computation model for rotating tool

Figure 2 shows a typical rotating drill. q-axis represents the drill direction, u-axis and v-axis are perpendicular to the q-axis. A surface element of a flute is contacting, therefore a resistance acts on the surface element of the flute. We first determine the chip volume removed by flute at the surface element. It is calculated as $w \times h \times r$. w is the thickness along the flute, is the division of the (equally divided) thickness along the feed direction to $\cos(b)$. b is the helix angle of the flutes. h is the chip height on the flutes is the division of the feed rate of the tool to the rotation speed, the number of the flutes and $\cot(b)$. r is the thickness along the radial direction, it is equal to R (the local radius of the surface element) * $\sin(c)$. c is the angle of the tool.

We consider there are two forces acting on the surface element: a stiffness force Ft and a shear force Fh that are proportional to the chip volume. The proportional ratios only depend on the tissue type. Ft acts along the reverse of the surface normal of the element. Fh acts along the reverse of the moving direction of the element that is per-

pendicular to the direction of *Ft*. *Ft* and *Fh* are divided into three components along the three main axes. By summing up the force components of all surface elements of a flute, we can calculate the forces along three main axes on the flute. By summing up the forces of all flutes, we can calculate the force on the tool.

3.2 Force computation model for oscillating tool

Figure 3 shows a typical oscillating tool for sectioning tissues of bone. u-axis represents the feed direction, v-axis represents the oscillating direction. The chip volume removed by a surface element of a flute is calculated as $w \times h \times r$. w is the thickness along the flute, is the division of the (equally divided) thickness along the feed direction to $\cos(d)$. d is the angle of the flutes. h is the height on the flute, is the division of the feed rate of the tool to the oscillating speed and $\cot(d)$. r is the tool thickness that is considered the same everywhere.

We consider there are two forces acting on the surface element: a stiffness force *Ft* and a shear force *Fh* that are proportional to the chip volume. The proportional ratios only depend on the tissue type. *Ft* acts along the reverse of the surface normal of the element. *Fh* acts along the reverse of the moving direction of the element that is perpendicular to the direction of *Ft*. *Ft* and *Fh* are divided into three components along the three main axes. By summing up the force components of all surface elements of a flute, we can calculate the forces along three main axes on the flute. By summing up the forces of all flutes, we can calculate the force on the tool.

3.3 Force computation model for general tool

In the system, a virtual tool is triangulated for rendering, and then triangle vertices can be used as contact points. The chip volume on a surface element (represented by a vertex) of a general tool can be calculated as $A \times h$. A is the area of the surface element, is determined by the average of the triangles that share the vertex (contact point). h is the height of chip volume that is determined by the dot product of the moving speed of the tool on the vertex and the surface normal of the vertex.

A stiffness force *Ft* and a shear force *Fh* are considered as proportional to the chip volume. The direction of *Ft* is along surface normal, *Fh* is along the moving direction. *Ft* and *Fh* are then divided into three components along the three main axes. By summing up the force components of all surface elements, we can calculate the force on the tool.

4 IMPLEMTATION EXAMPLE

Figure 4 shows the rendering results of a knee open osteotomy for removing a tumor inside the proximal tibia. The volume is constituted by 28 CT slices with 256*256 resolution. The following simulation is implemented under a P-III 800 with 256 Mbyte main memories without special graphics accelerator.

Figure 4(a) shows the proximal tibia is being sectioned by an oscillating saw. Because the saw approaches the tibia too fast, there occurs overload on the left two flutes that has already sectioned the tibia. That suggests the user must slow the feed rate of the tool before contacting with the bones. Figure 4(b) shows a window-shaped bone fragment has been sectioned, recognized and repositioned away. The tumor also appeared. A dissector is ready to dissect the tumor. The dissector is not simple and vibrating (rotating or oscillating) tool, therefore it is used manually to section rather soft tissues, such as tumors. Figure 4(c) shows that the tumor is being dissected by the dissector. However, the left middle part of the tool contacts with a hard cortical bone. Therefore, there occurs overload at the local area. This suggests the path of the dissector should be modified to avoid contacting with the cortical bone.
This simulation of the musculoskeletal surgery indicates the position and size of the window is a good choice for opening the knee, the tumor can be completely removed and the graft bone is suitable to fill up the tumor space. The feed rates and paths can be confirmed and than modified to obtain comfortable ones. All simulation responses are in 2 second that shows our simulations can achieve the interactive requests for ordinary medical volumes. A response includes completing a geometric simulation function, a force computation (for section and drill simulations) reconstructing the isosurfaces of the operated part of the volume and rendering a stereographic image.

5 DISCUSSIONS AND FUTURE WORK

We have developed the force computation models for evaluating the loads on the surgical tools for sectioning or drilling procedures of musculoskeletal surgeries. Comparing to the one-point force model, our models calculate the resistance on multiple surface elements of the tool to simulate precise results. By the geometric and physical evaluations, our surgical simulation system can verify not only the geometric morphology but also the feed rate and moving paths of the tools. These help the verifications of surgical plans, and the training of residents and students. Although the simulation results seem reasonable, we still have to verify the force computation models by experiments.

We hope to extend the simulation system with tactile responses in the future. Before combining the haptic devices, we must improve the speed of rendering isosurfaces. Although the force computation practically can achieve the refresh-rate requirement of haptic devices, the rendition of large amount of isosurfaces is still difficult to achieve the visual real-time requirement. Decimation techniques to reduce triangles of isosurfaces or other volume visualization techniques such as accelerated techniques can be tried to save the rendition time. Besides, developing force

computation models for simulating other surgical procedures, such as sectioning deformable soft tissues is also required.

ACKNOWLEDGMENT

The authors would like to thank National Science Council for financially supporting this research under Contract No. NSC-86-2213-E-033-036, NSC-87-2213-E-033-005 and NSC-89-2320-B-038-019-M08.

REFERENCE

[1] Koch, R.M., Gross, M.H. , Carls, F.R. , Buren, D.F.V. , Fankhauser, G. and Parish,Y.I.H., "Simulating Facial Surgery Using Finite Element Models", *ACM SIGGraph Computer Graphics*, Vol.30, No.5, pp. 421-42, 1996.

[2] Hunter, I.W., Jones, L.A., Sagar, M.A., Lafontaine, S.R. and Hunter. P.J "Ophthalmic Microsurgeical Robot and Associated Virtual Environment", *Computer in Biomedicine*, Vol. 25, No. 2, pp. 173-182, 1995.

[3] Cotin, S., N., Delingette and Ayche, N. , "Real-Time Elastic Deformations of Soft Tissues for Surgery Simulation", *IEEE Transaction on Visualization and Computer Graphics*, Vol. 5, No. 1, pp.62-73, 1999.

[4] Cover,S. et al., "Interactively deformable models for surgery simulation", *IEEE Computer Graphics and Applications*, Vol. 13, No6, pp. 68-75, 1993.

[5] O'Toole R.V., Plater R.R., Kummel T.M., Blank W.C., Cornelius N.H., RobertsW. R., Bell W.J. and Raibert M., "Measuring and Developing Suturing Technique with a Virtual Reality Surgical Simulator", *Journal of America College of Surgeons*, Vol.189. No.1, pp.114-127, 1999.

[6] Seipel S., Wagner I.V., Koch S. and Schneide W., "Oral implant treatment planning in a virtual reality environment", *Computer Methods and Programs in Biomedicine* 57(2) 95-103, 1999.

[7] Frisken-Gibson, Sarah. F., " Using Linked Volumes to Model Object Collision, Deformation, Cutting, Carving, and Joining", *IEEE Transaction on Visualization and Computer Graphics*, Vol. 5, No. 4, pp.333-348, 1999.

[8] Udupa, J.K. and D., Odhner, "Fast Visualization, Manipulation and Analysis of Binary Volumetric Objects", *IEEE CG&A*, Vol. 11, No.5, pp.53-63, 1991.

[9] Tsai, M.D., Hsieh, M.S., Chang,W.C. and Wang, S.K., 1996, "Volume Manipulation Algorithms for Simulating Musculoskeletal Surgery", *Pacific Graphics '96*, pp.220-234, 1996.

[10] Gibson,S., Samosky,J., Mor,A., Fyock,C., Grimson,E., Kanade,T., Kikinis,R., Lauer,H., l McKenzie,N., Nakajima,S., Ohkami,H., Osborne,R., Sawada,A., " Simulating Arthroscopic Knee Surgery Using Volumetric Object Representations, Real-Time Volume Rendering and Haptic Feedback", *First Joint Conference of Computer Vision,VR and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery*, pp. 369- 378, 1997.

[11] Cotin,S. , Delingette,H. , Clement,J.M. , Soler,L. , Ayache,N. , Marescaux,J., "Geometrical and physical representations for a simulator of haptic surgery", *Proceedings of Medicine Meets Virtual Reality IV* , 1996.

[12] Avila,R. , Sobierajski,L., "A haptic interaction method for Volume Visualization", *Proc. Visualization '96*, pp. 197-204, 1996.

[13] Massie,T.M. , Salisbury,J.K., "The PHANToM Haptic Interface: A Device for Probing Virtual Objects", *ASME Haptic Interfaces for Virtual Environment and Teleoperator Systems in Dynamic Systems and Control* (Chicago, Now, 6-11), vo1. 1, pp. 295-301, 1994.

[14] Brooks F.P., Ouh-Yougn M.,Bratter J.J. and Kilpatrick P.J., "Project GROPE: Haptic Displays for Scientific Visualization", *ACM SIGGraph Computer Graphics*, pp.177-186, 1990.

[15] Burdea G., Langrana N., Lange K., Gomez D. and Deshpande S., "Dynamic Force Feedback in a Virtual Knee Palpation, *Journal of Artificial Intelligence in Medicine,* Vol.6, pp.321-333, 1994.

[16] Taketa,S., Tsai,M.D., Inui,M., Sata,T., "A Cutting Simulation System for Machining Ability Evalution Using a Workpiece Model", *Annuals of the CIRP*, Vol.38, No.1, pp.417-420, 1989.

[17] Tsai, M.D., Takata, S., Inui, M., Kimura, F., Sata, T., 1990, "Prediction of Chatter Vibration by Means of a Model Based Cutting Simulation System", *Annuals of the CIRP*, Vol. 39,No.1, pp.447-450, 1990.

[18] Sealy, G. and Novins, K., "Effective Volume Sampling of Solid Models using Distance Measures", *IEEE Computer Graphics International'99*, pp.26-32, 1999.

[19] Lorensen, W.E. and Cline, H.E., "Marching Cubes: A High Resolution 3D Surface Construction Algorithm", *ACM SIGGraph Computer Graphics*, Vol.21, No.4, pp.163-169, 1987.

[20] Feng, L. and Soon, S.H., "An Effective 3D Seed Fill Algorithm", *Computer & Graphics*, Vol.22, No.5, pp.641-644, 1998.

[21] He, T. and Kaufman, A.,"Collision Detection for Volumetric Objects", *IEEE Visualization'97*, pp.159-166, 1997.

[22] Cohen D. and Kaufman, A., 1997, "3D Line Voxelization and Connectivity Control", *IEEE CG&A*, Vol. 17, No. 6, pp.80-87, 1997.

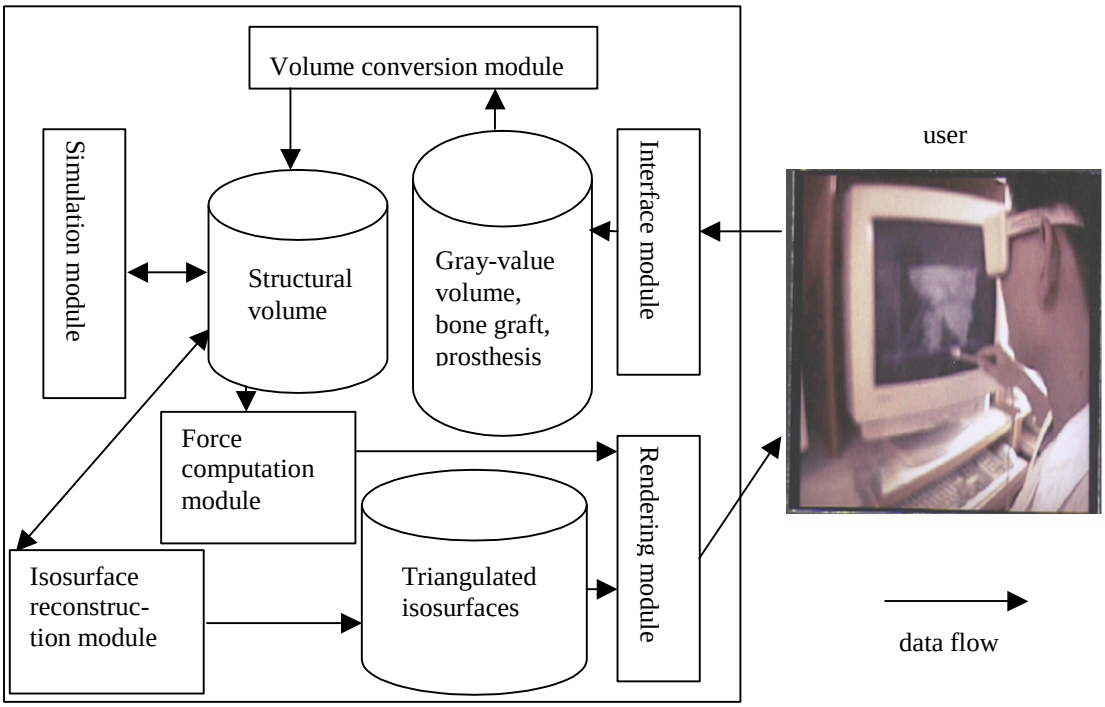


Figure 1 System achitecture

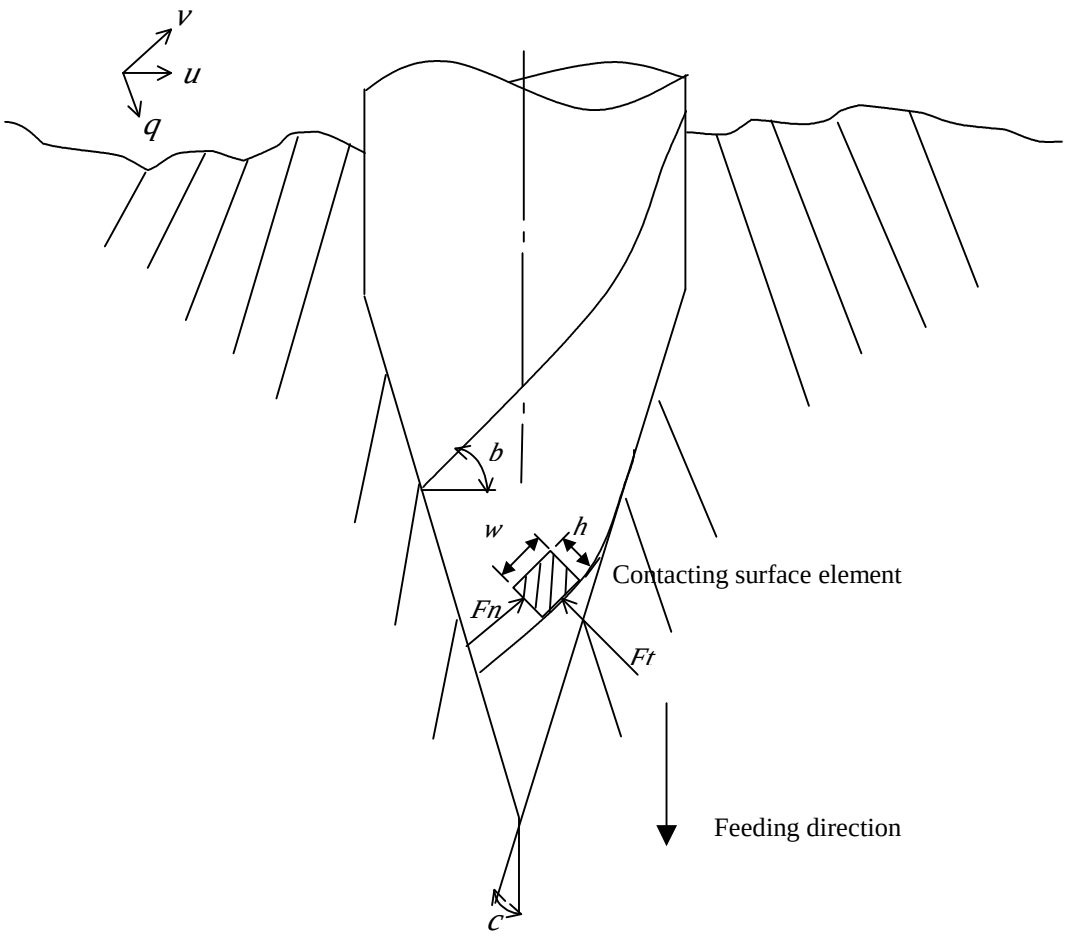


Figure 2 Force computation model of a rotating drill

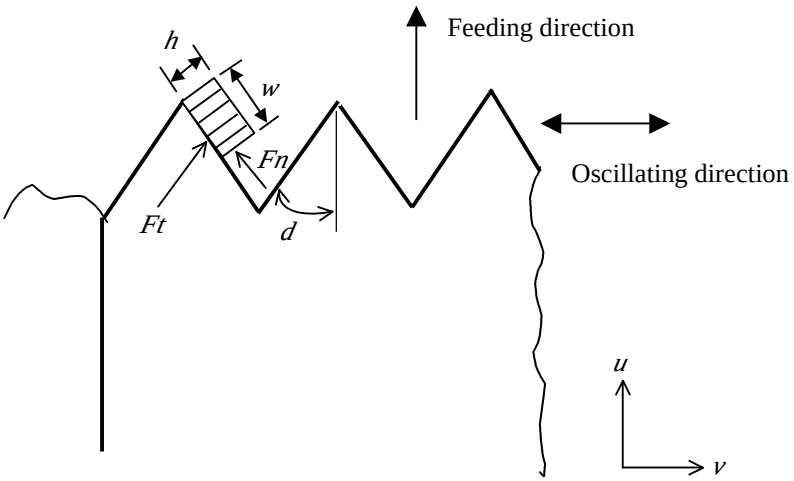
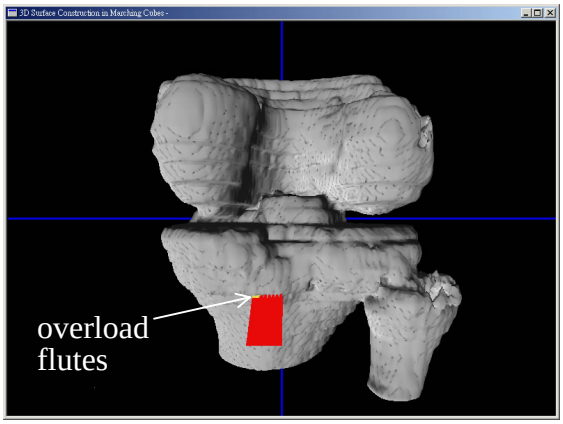
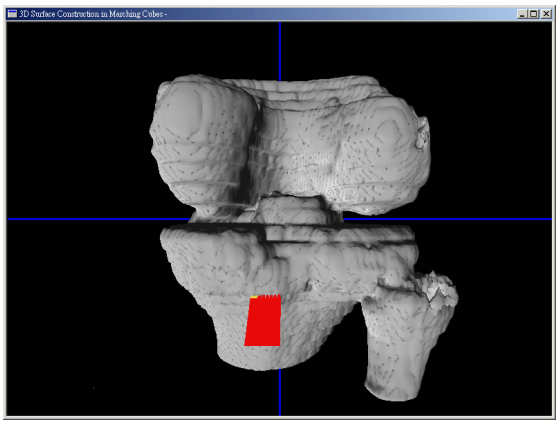


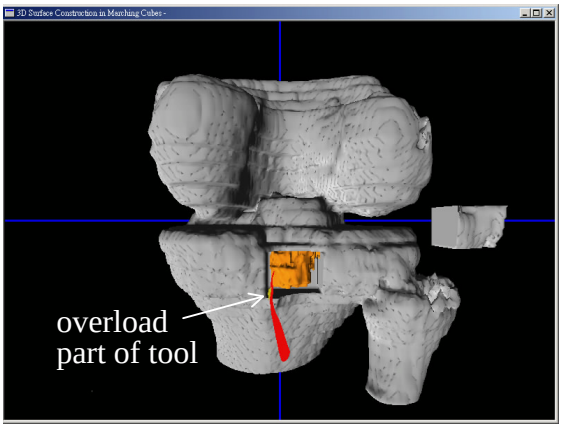
Figure 3 Force computation model of an oscillating saw



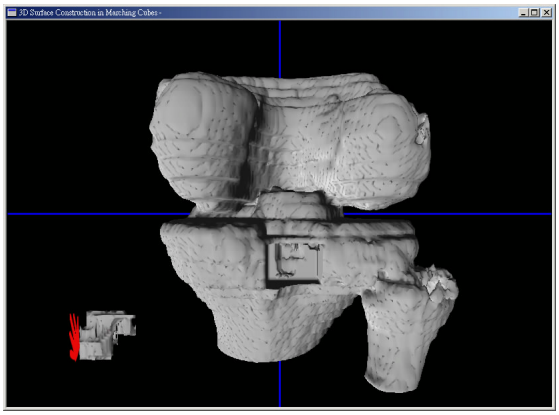
(a) Proximal tibia is being sectioned by an oscallating saw.



(b) Window-shaped bone is sectioned and re-positioned away.

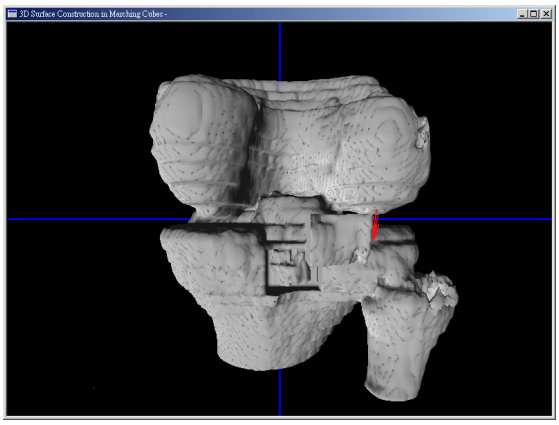


(c) Tumor is being sectioned by a dissector.

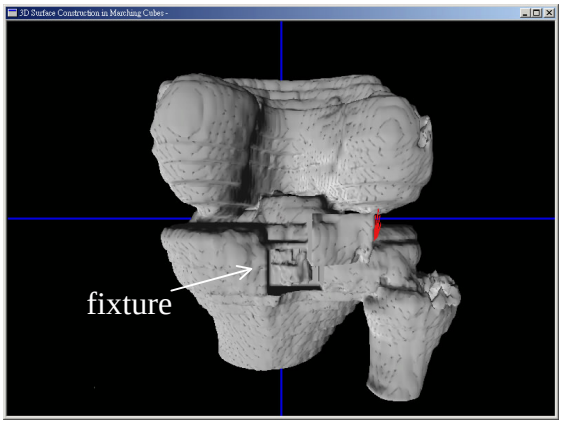


(d) Bone graft is prepared to fill up space of removed tumor

Figure 4 Open osteotomy for removing tumor in tibia (continuation)

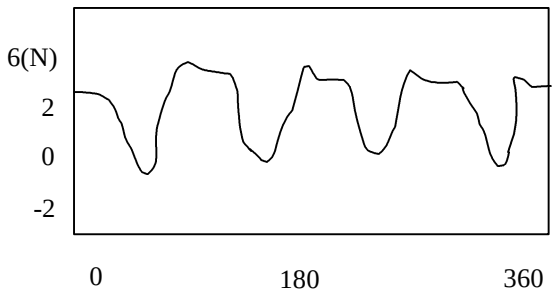


(e) Bone graft is implanted, window shaped bone is being repositioned to original position

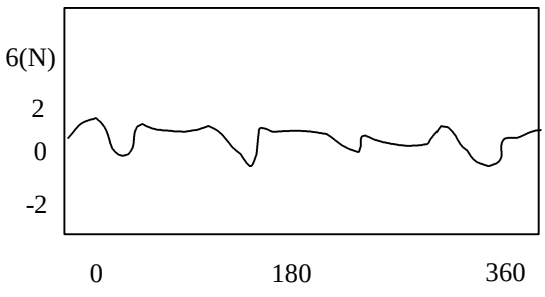


(f) Window shaped bone is already repositioned, rotating drills fixes the fixture

Figure 4 Open osteotomy for removing tumor in tibia



Rotational angle (deg.)
force along feed (q-axis) direction



Rotational angle (deg.)
force along u-axis direction

Figure 5 Instantaneous forces of a drill during one rotation