Haptic and Visual Simulation of a Material Cutting Process
A Study Focused on Bone Surgery and the Use of Simulators for Education and Training

MAGNUS G. ERIKSSON
Haptic and Visual Simulation of a Material Cutting Process – A Study Focused on Bone Surgery and the Use of Simulators for Education and Training

Magnus G. Eriksson

Licentiate thesis

Academic thesis, which with the approval of Kungliga Tekniska Högskolan, will be presented for public review in fulfilment of the requirements for a Licentiate of Engineering in Technology and Health. The public review is held at Kungliga Tekniska Högskolan, Brinellvägen 83 in room B442 at 10.15 am on the 9th of June 2006.
Haptic and Visual Simulation of a Material Cutting Process – A Study Focused on Bone Surgery and the Use of Simulators for Education and Training

A prototype of a haptic and virtual reality simulator has been developed for simulation of the bone milling and material removal process occurring in several operations, e.g. temporal bone surgery or dental milling. The milling phase of an operation is difficult, safety critical and very time consuming. Reduction of operation time by only a few percent would in the long run save society large expenses. In order to reduce operation time and to provide surgeons with an invaluable practicing environment, this licentiate thesis discusses the introduction of a simulator system to be used in both surgeon curriculum and in close connection to the actual operations.

The virtual reality and haptic feedback topics still constitute a young and unexplored area. It has only been active for about 10-15 years for medical applications. High risk training on real patients and the change from open surgery to endoscopic procedures have enforced the introduction of haptic and virtual reality simulators for training of surgeons. Increased computer power and the similarity to the successful aviation simulators also motivate to start using simulators for training of surgical skills.

The research focus has been twofold: 1) To develop a well working VR-system for realistic graphical representation of the skull itself including the changes resulting from milling, and 2) to find an efficient algorithm for haptic feedback to mimic the milling procedure using the volumetric Computer Tomography (CT) data of the skull. The developed haptic algorithm has been verified and tested in the simulator. The visualization of the milling process is rendered at a graphical frame rate of 30 Hz and the haptic rendering loop is updated at 1000 Hz. Test results show that the real-time demands are fulfilled. The visual and haptic implementations have been the two major steps to reach the overall goal with this research project.

A survey study is also included where the use of VR and haptic simulators in the surgical curriculum is investigated. The study starts with a historical perspective of the VR and haptic topics and is built up by answering different questions related to this topic and the implementation of simulators at the medical centres. The questions are of general concern for those developing surgical VR and haptic simulators.

Suggested future work includes modelling, development and validation of the haptic forces occurring in the milling process and, based on this, implementation in the simulator system. Also, further development of the simulator should be done in close cooperation with surgeons in order to get appropriate feedback for further improvements of the functionality and performance of the simulator.

Keywords: Surgical simulation, virtual reality, haptic feedback, surgical training, medical simulators, metrics, 3D visualization

Language: English
Acknowledgements

The research presented in this thesis is funded by CTV (Center for Technology and Health Care). The work has been conducted at the Mechatronics Lab at the Department of Machine Design at KTH in Stockholm, Sweden.

I would like to express my gratitude to all people that have been involved in the project, especially my supervisor, professor Jan Wikander, for all help with the research and editing of the papers. I want to thank my co-supervisor professor Hans von Holst for introducing me to the medical and surgery areas. I also want to thank Li Tsai, Christian Hogman and the rest of the team at Simulatorcentrum, Karolinska Universitetssjukhuset, for giving positive and good feedback during our meetings.

Mark Dixon and Daniel Evestedt at SenseGraphics AB have been a very good and invaluable help during my start up period. Thanks guys for answering all my “stupid” questions and giving me good ideas.

Professor Court Cutting and Aaron Oliker at the Graphics lab at the NYU Medical Center, New York City, USA, are acknowledged for their kindness and supporting me during my three months of work in NYC.

Further on I would like to thank my ex-roommate and ex-supervisor PhD Henrik Flemmer and my roommate Fredrik Roos. Finally I also want to thank my family and friends.

Stockholm, May 2006

Magnus G. Eriksson
List of appended publications

Paper A


Paper B


Paper C

Other publications


Contents

1. Introduction ........................................................................................................... 1
   1.1 Background ........................................................................................................... 1
   1.2 Overall goals ....................................................................................................... 2
   1.3 Education of surgeons ....................................................................................... 5
   1.4 Various possible VR haptic and milling applications ........................................ 6
      1.4.1 Temporal bone surgery .................................................................................. 6
      1.4.2 Craniofacial surgery, for example, on the jawbone .................................... 6
      1.4.3 Dental tooth milling ....................................................................................... 7
      1.4.4 Vertebral operating procedures ................................................................... 7
      1.4.5 Freeform design ........................................................................................... 8
   1.5 Equipment and implementation ....................................................................... 8
   1.6 Research issues ................................................................................................. 10
      1.6.1 Graphic rendering with real-time demands .................................................. 10
      1.6.2 Haptic fall-through problem (presentation of various solutions) ............... 12
      1.6.3 Verification of the developed proxy-based haptic algorithm ....................... 16
2. Summary of appended papers ............................................................................. 20
   2.1 Paper A: A Haptic VR Milling Surgery Simulator Using High-Resolution CT Data ........................................................................................................... 20
   2.2 Paper B: A Virtual and Haptic Milling Surgery Simulator ................................ 20
   2.3 Paper C: The Use of Virtual Reality and Haptic Simulators for Training and Education of Surgical Skills .............................................................. 21
3. Conclusion, discussion, and future work ........................................................... 22
4. References ......................................................................................................... 25
1. Introduction

This thesis examines the development of a haptic and virtual reality (VR) simulator. The simulator has been developed for simulating the bone milling and material removal process occurring in several operations, such as temporal bone surgery or dental milling. This project is an extension to the research done by Flemmer (2004) as part of the “Skullbase Project” at the Mechatronics Lab, the Royal Institute of Technology (KTH), Stockholm, Sweden. The project was funded by the Center for Technology and Health Care (CTV), an organization developed out of collaboration between KTH and the Karolinska Institutet, Stockholm, Sweden.

1.1 Background

Virtual reality and haptic feedback are still relatively new and unexplored areas, only emerging in approximately the last 10–15 years for medical applications. In the 1980s the aviation industry saw the possibilities of using increased computer power to develop training simulators and ushered in a new technology era. The first haptic device was developed in the early 1990s and the first surgical VR training simulator was an abdominal simulator developed in 1991 by Satava (1993).

Both the high risks of training on real patients and the shift from open surgery to endoscopic procedures have spurred the introduction of haptic and virtual reality simulators for training surgeons. Increased computer power and similarities with the successful aviation simulators have also motivated the introduction of simulators for surgical training.

The main reasons for using haptic and VR simulators are as follows:

1. Surgical techniques are undergoing a major shift from open surgery to more endoscopic procedures that minimize patient recovery time. Jolesz (1997) says that limited visibility through “keyholes” during endoscopic procedures and through small incisions of diminishing size increases the need for intraoperative image guidance. Monitor-based navigation systems are used with endoscopic surgery, so there is a natural progression from this real-world situation to practicing in a virtual environment using the same equipment.

2. Simulators will create new training opportunities for surgical procedures that are impossible to train for using current methods. Also, qualitative methods for measuring operating skills can be implemented using a computer-based tracking system to evaluate specific surgical performance.

3. Pre-operation planning using a simulator will reduce errors and make the surgeon feel safer when entering the real operating room to perform the task.

4. It will be possible to train and simulate specific complications, which is impossible today when the resident is dealing with real patients.
5. In the simulator it will be possible to test and evaluate completely new operating methods; this is impossible today out of concern for patient safety.

6. Moving the training of residents from the operating room to simulators would reduce operating room costs, costs that are very high today. Dawson and Kaufman (1998) claim that up to $1500/h is being charged for the use of some operating rooms. Moving training for surgical procedures from the operating room to a simulator in a lecture room would thus offer considerable economic advantages.

7. With the introduction of simulators into the curriculum, it will also become easier and more natural to initiate robot-assisted surgery. Using robot-assisted surgery would increase the precision and safety of operations and also decrease the operating time.

The simulator prototype developed and presented in this Licentiate research is primarily intended for skull base surgery. To remove cancerous tumors in certain locations in the human head, the surgeon must not only open up a hole in the temporal bone, but a path along the inside of the temporal bone must also be made. Today, the surgeon mills this path very carefully using a small hand-held mill, so that the tumor can be reached without affecting the brain more than necessary or damaging other vital parts of the head located near the tumor. Typically, this path is located in a region where the temporal bone is geometrically complicated and surrounds neurons, brain tissue, and critical parts of the nervous system. Hence, the milling phase of an operation of this type is difficult, safety critical, and very time consuming. Reducing operating time by even a few percent would in the long run produce considerable savings.

In the interests of reducing operating time and providing surgeons with an invaluable practicing environment, this licentiate thesis discusses the introduction of a simulator system to be used in both the surgery curriculum and in close connection with actual operations. Prior to a real operation, invaluable knowledge regarding potential complications and other vital factors can be gained by first performing the operation on the simulator. For simulations of a sensitive operation like the one described, the surgeon needs both high-quality visual and tactile feedback.

1.2 Overall goals

In earlier research, a prototype master–slave system for telerobotic surgery of the described type was developed by Flemmer (2004). The work presented here describes an extension of that system, in terms of developing a simulator system based on a virtual-reality representation of the human skull from which both haptic and visual feedback to the surgeon is generated. A future vision is that the same master unit will be used for both systems.

The research focus has been twofold: 1) to develop a properly-functioning VR system for the realistic graphic representation of the skull itself, including the changes resulting from
milling, and 2) to find an efficient algorithm for haptic feedback to mimic the milling procedure using the volumetric computer tomography (CT) data of the skull. When the mill (also graphically depicted) interacts with the bone and cuts away material, the visual rendering must manage to update the representation of the skull in real time without artifacts or delays. For this, an updating rate of approximately 30 Hz and a latency of less than 300 ms are needed to create a realistic visual impression [Mark et al. (1996)]. The corresponding demand for haptic rendering is an update frequency of 1000 Hz [Mark et al. (1996)].

Meeting these real-time requirements is a matter of general concern, since computational workload is much larger when rendering a deformable rather than a non-deformable object in real-time. Also, realistic haptic rendering in six degrees of freedom for complex interactions (not only point contacts) is very demanding. Different methods for graphic and haptic rendering are discussed in the papers included in this thesis, both from a computational workload and from a performance point of view.

Visual and haptic implementation are two major steps towards the overall goal of this research project, to develop an appropriate haptic and virtual-reality system for training and educating surgeons who practice bone milling. Another future goal is to connect the VR system with the already developed telerobotic surgery system to control a real operation situation with the help of VR interaction. The complete system is presented in Figure 1. The operating system would give the surgeon more information and the ability to perform safer operations by using visual and haptic feedback together with a telerobotic system. The surgeon manipulates the master device, which controls both the VR representation of the skull and the robotic slave performing the milling operation. This is a vision of the operating room of the future: the surgeon can control the operation procedures with the help of telerobotics and haptic and visual feedback, for greater safer and time efficiency.
After implementing realistic, “good enough” visualization and basic haptic functionality, planned ongoing research will address the further modeling, development, and validation of the haptic forces occurring in the milling process and, based on this, in implementing the simulator system. This includes an expansion from the current three degrees of freedom point contact haptic model to a full six degrees of freedom and more complex contact geometries. Another challenging problem comprises the stability problems occurring when two stiff objects collide (the mill and the skull). Modeling material removal and the feedback forces of the milling process also need further research.

The outcome of our cooperation with Centrum för Teknik i Vården (CTV) at KTH and Simulatorcentrum at Karolinska University Hospital will be very valuable to future research. The simulator will be further developed in close cooperation with the surgeons who will use it; they will conduct psychophysical experiments and give feedback for the further development of the simulator’s performance.

Various ideas about how to develop a simulator were discussed and tested in the first part of this research project. It was difficult to find an efficient start-up process for the project. Even with close contacts with surgeons and those responsible for simulator-based education, it is difficult to draw conclusions regarding the most important types of procedures needing simulators and regarding their requirements and specifications. Finding appropriate software and hardware for the project was also challenging, partly because of our lack of experience in the field, but also because computer-based tools for simulating such complex visual and
haptic processes are lacking. Despite these initial obstacles, we now have a properly functioning simulator that can be used for further development and for improving both visual and haptic feedback, as well as overall system layout.

1.3 Education of surgeons

Dawson and Kaufman (1998) state that the education of surgeons is the same as it has been for hundreds of years, functioning according to the maxim, “See one, do one, teach one.” Accordingly, the novice “sees, does, and teaches” on patients who enter the front door of the teaching medical center. This puts the patient into an unavoidably risky situation, in which he or she is the subject on which the novice learns. The situation is both ethically and economically unacceptable if there are other ways to teach and learn surgical skills. Earlier, there were no alternatives; now, however, in this information technology era, there are alternatives, and the use of medical simulators can change the whole surgical education model. Alternatives to using simulators are to use cadavers, plastic models, and animals, but these have many drawbacks, such as high cost, ethical problems, and difficulties drawing qualitative conclusions from the training results [Nelson (1990) and Totten (1999)].

Scerbo (2005) briefly describes some of the benefits that medical VR simulators offer in the surgical curriculum: 1) they allow students to acquire and refine their skills without putting patients at risk, 2) they provide immediate performance feedback and objective measures of performance, 3) they allow students to encounter and interact with rare pathologies, and finally 4) they reduce the need for animal and human cadaver labs.

VR simulators might also be used in selecting medical students or young graduates based on aptitude for surgical skills. It may also be possible to use this type of simulator to check the psychomotor skills of older, experienced surgeons, to ensure their competence to continue practicing [McCloy and Sone (2001)]. Surgeons now train for a fixed period of time; future surgeons may have a variable residency program, depending upon how quickly they attain competence by using a simulator [Fried et al. (2004)].

The ability to simulate specific complications is one of the most important capacities of a simulator; that, and the ability to train physicians without putting patients at risk. With a simulator, teaching would not occur in the operating room, but would make use of a simulator instead.

Both Ahlberg (2005) and Ström (2005) have demonstrated in different various studies that haptic feedback enhances performance in the training phase of skill acquisition in image-guided surgery simulator training. More consistent and much safer procedures will be performed if haptic feedback is integrated into image-guided surgery training.

VR and haptic surgery simulators have been developed or are currently under development for abdominal trauma surgery, laparoscopic cholecystectomy, neurosurgery, endoscopic sinus surgery, temporal bone dissection, arthroscopic surgery of the knee and shoulder, vascular anastomosis, coronary stent or cardiac lead placement, and gastroscope training [Bloom et al. (2003), Bro-Nielsen et al. (1998), Dawson et al. (2000), Eriksson et al. (2006),
Muller and Bockholt (1998), O'Toole et al. (1999), Smith et al. (1999), Tanaka et al. (1998), Tseng et al. (1998), Weghorst et al. (1998), Wiet et al. (2002)]. All of these simulators have been well received and are considered to have great potential; however, most have not yet been adequately tested for validity or for effectiveness as teaching tools. A detailed discussion of this is presented in paper C.

1.4 Various possible VR haptic and milling applications

Our developed simulator is not designed for any one specific operation, but rather can be used in training for different sorts of surgical milling operations. The only limitation is that the object to be manipulated must comprise volumetric data—e.g., derived from CT or magnetic resonance imaging (MRI)—uploaded into the simulator. The simulator has been developed so that it is possible both to add and remove material, which can be useful when training for milling operations. The rest of this section presents different possible VR and haptic milling operations for which simulator training can increase safety and decrease operating time. None of the concepts below is presented in detail, because the exact application of the simulator has yet to be established; rather, the concepts presented can be regarded as illustrating the flexibility and range of possibilities for using the simulator.

1.4.1 Temporal bone surgery

In temporal bone surgery the surgeon very carefully mills a path in the skull bone with a small hand-held mill, so that the tumor can be reached without affecting the brain more than necessary or damaging other vital parts of the head located near the tumor. Typically, this path is located in a region where the skull bone is geometrically complicated and is surrounded by neurons, brain tissue, and critical parts of the nervous system. Hence, the milling phase of such an operation is difficult, safety critical, and very time consuming. Training in a simulator could help the surgeon perform safer operations. Different research groups are developing simulators for training surgeons to perform these operations. The VOXEL-MAN Project (2006), the Stanford Biorobotics Lab (2006), and the CRS4 Visual Computing Group (2006) are the most successful groups that have advanced the development of temporal bone surgery simulators.

1.4.2 Craniofacial surgery, for example, on the jawbone

Craniofacial operations have become increasingly common and it has been necessary to find new ways to educate and train surgeons to perform them. One common procedure is cleft lip surgery, in which surgeons today use 3D computer visualization and animation programs for education and pre-operative planning [NYU Medical Center (2006)]. Introducing haptic and 3D navigation would increase the realism even more. In a craniofacial surgery simulator,
the created and modified data could easily be exported to a CAD program and printed using a 3D printer to create a real physical model of the organ to be manipulated. A literature survey indicates that no simulators using haptics have been developed for craniofacial surgery training.

1.4.3 Dental tooth milling

Dental training and education currently use plastic teeth for practicing the milling process. The resident mills the plastic teeth and the motions are tracked and evaluated using a computerized system [DentSim (2006)]. An instructor tries to evaluate how well a novice has performed by looking at the results afterwards. This methodology could be changed by using a simulator. Dental residents could practice by themselves in a virtual environment, over and over again, directly getting qualitative feedback from the program as to their skills level. The benefits are those mentioned above, as well as the ability to model the tactile feeling of manipulating a tooth with caries, which is currently impossible using plastic teeth. Plastic teeth provide just one kind of force feedback; in a simulator, however, it would be possible to apply caries to a specific region, deriving a different force feedback there than is experienced touching a clean part of the tooth. One major drawback of using a VR simulator for training in dental milling is the introduction of a completely new element into the dentist arena. The dentist performing a real procedure on a patient never uses a monitor or 3D visualization for navigation. Using a VR simulator could well cause more problems than benefits, and there is a risk that while residents could be trained to be brilliant VR dentists, they may not learn the correct skills for executing a real procedure. Despite this, several groups and companies are developing VR dental simulators, including the Korea Institute of Science and Technology (2006), Simulife Systems (2006), Novint/VRDTS (2006), and the Stanford Biorobotics Lab (2006).

1.4.4 Vertebral operating procedures

Vertebral operations are very risky, high-precision procedures. One such operation is the strengthening of the spine with titanium nails. In it, the surgeon must carefully find the exact location of the free space between two vertebrae, and then mill a corridor through which to insert the nails, one on each side of the spinal marrow. The milling path is depicted in Figure 4 below. Using the hand-held mill, the surgeon must perform the procedure very carefully to avoid hurting the spinal marrow or the nerve fibers located near where the nail will be placed. This complicated operation is a difficult one to let surgeons practice, and a specially developed simulator could greatly facilitate training. For this application it would also be interesting to develop a telerobotic system with which to perform the operation, controlled by an educated surgeon; such a robot system would increase the precision of the process, making it safer for the patient. However, the implementation of a robot system is beyond the research scope of this thesis.
There is a paradigm shift occurring in the operating room, from open surgery to the introduction of endoscopic techniques [Karolinska Universitetssjukhuset (2006)]. These techniques allow for easier diagnostic methods, safer and faster operations, faster rehabilitation, and decreased risk of infection. Using endoscopy instruments requires that the surgeon be able to navigate the instrument and manipulate the organs with the aid of a 3D camera and monitoring system. Developing a VR and haptic training simulator for vertebra fracture operations would make it possible for surgeons to train for this procedure, which is impossible today using the open surgery method. A literature survey indicates that this is a new idea, and no other research teams are working on such an application. The simulator can be regarded as a concept that combines previously developed laparoscopic (see, e.g., Mentice 2006, Surgical Science 2006), and bone milling simulators.

1.4.5 Freeform design

The simulator can also be used for freeform design. The ability to add and remove material and change the size and shape of the tool makes the simulator a functional sculpting system. The created VR model can be exported to a CAD program and printed using a 3D printer to create a real physical model. This can be useful for industrial design or other art, visualization, and computer interaction applications. Sensable Technologies (2006) has developed “Freeform,” a VR and haptic training program along these lines.

1.5 Equipment and implementation

For our application, a SenseGraphics H3D API scene graph (2006) is connected to the Sensable Technologies OpenHaptics toolkit (2006) for the control of the master device. The basic function of the scene graph is to describe both the visual and physical attributes of the VR environment. All the graphics and haptics are represented in the same scene graph. The advantage of this structure is that additional graphics and force modules can be implemented in the same software structure and share the same data. This enables vital real-time
interaction between different data in the scenario, which is crucial when the surface structure of the skull bone changes.

The C++ programming language is used for the low-level programming in the H3D API, while the X3D and Python scripting languages are used to build up the scene graph. The software consists of two different threads updated at 30 and 1000 Hz, respectively. The first thread represents the graphics loop and the second the haptic loop. Updates from the graphics loop are transferred at each sample to the haptic rendering loop to provide the force feedback. How the different loops run and how they share data is described in Figure 6.

The OpenHaptics toolkit is used to control the PHANToM Omni haptic device (2006). The forces are small at < 3.3 N [Hansson et al. (1996)], which is lower than the maximum force handled by the Omni, and the workspace of the haptic device is sophisticated enough to realistically mimic a real surgery. One limitation of this device is the limited number of actuated degrees of freedom (DOF) that can be used in a force model. The Omni delivers 6-DOF position information ($x$, $y$, $z$, pitch, roll, and yaw) from the sensors, but can only control the actuators in three DOF ($x$, $y$, and $z$), meaning that only forces, not torques, can be sent back to the user. Another limitation of the Omni compared to other haptic devices from Sensable is the poor stiffness of the device, which is very important in giving a realistic feeling when interacting with stiff materials, such as bone. The PHANToM Omni has a stiffness in the $x$-axis direction of 1.26 N/mm, while the PHANToM Desktop has a stiffness in that direction of 1.86 N/mm. Thus stiffness is 48% better with the Desktop model, but the price of the Omni is preferable.

![Figure 6. An overview of the haptic and graphic threads](image-url)
1.6 Research issues

1.6.1 Graphic rendering with real-time demands

Several methods are available for volumetrically representing a discrete 3D data matrix acquired from a CT or MRI scan. These are ray-casting [Levoy (1998)], 3D texture mapping [Cabral et al. (1994)], and Marching cubes [Lorensen (1987)]. In the case when the mill (also included in the graphics) interacts with the bone and cuts away material, the visual rendering must manage to update the representation of the skull in real time without artifacts or delays. For this, an updating rate of approximately 30 Hz and a latency of less than 300 ms are needed to give a realistic visual impression [Mark et al. (1996)].

The ray-casting algorithm is a volume-rendering method, it uses the volume data directly and the images are produced from projection of 3D voxel information into 2D pixel images. All the voxels located in the viewing line are used to generate the image.

The 3D texture mapping algorithm is another popular volume-rendering method. The 3D texture mapping method used in a number of research projects for the bone milling application is presented in Agus et al. (2002), Wiet et al. (2002), and Todd and Naghdy (2004). With these volume-rendering methods, little or no effort is required to visualize something that is similar to the skull; however, the visual impression is poor, mostly because a low-resolution volumetric dataset (e.g., $64 \times 64 \times 64$) has to be used to speed up the computations. In this application, where the user is focusing on one particular part of the object for a long time, high-resolution datasets (e.g., $512 \times 512 \times 176$, as can be taken from a CT scan) are needed for a realistic view. To do the real-time rendering using a volume-rendering method, normally very expensive graphics boards are required. Another disadvantage of both the ray-casting and texture mapping volume-rendering methods is that they produce several visual artifacts, making it annoying to look at one spot for a long period of time.

Due to real-time demands, and to achieve computational efficiency and more accurate visualization, a surface-rendering method is used. The Marching cubes algorithm is the most popular surface-rendering algorithm. It is a very efficient rendering method using voxel density values to produce a high-quality visualization of the surface. In the developed simulator, the dataset from the CT scan is implemented in a matrix structure in which the original Marching cubes algorithm is applied for data management and for generating a 3D model of the skull bone, based on a predefined density isovalue. This value indicates the object density level that defines the surface. This is done by comparing the isovalue with the voxel density values (the attenuation values taken from the CT scan) of which the volumetric object consists. The object to be rendered is then built up of cubes that consist of one density value in each corner. Depending on the relationship between the isovalue and the voxel values, different vertices are created along the edges of each cube using linear interpolation. At every vertex a normal vector is calculated. The created vectors of normals and vertices generate triangles using the GL_TRIANGLES function (an example is presented in Figure 7). Taken together, the triangles form the surface of the object. There
are 15 possible ways that the triangles can be constructed for one cube (see Figure 8), and
different rules are applied to connect the different triangles creating the surface of the object.
The normals of a triangle are calculated based on the voxel density gradient of each vertex of
the triangle; the GL_TRIANGLES function interpolates the normal values along each
border of a triangle to give a smooth graphic rendering of the surface.

To perform efficient rendering, a tree node structure and cached lists are used. The octree
structure is used to avoid traversal of empty regions using macrocells that contain the
min/max values (the coordinate and density value of each voxel) of their children nodes.
Whether or not to traverse a region is determined by comparing the isosurface value (used in
the Marching cubes algorithm) with the stored min/max values and also by comparing the
coordinates given to each child node with the location of the tip of the mill. This improves
the computation time and thus the real-time performance.

For realistic visual presentation of the material removal process, regeneration of the iso
surfaces at the local interaction volume only is important for real-time performance. Since
the surface is represented by the voxel density values, these values can easily be modified to
depict the removal of material. The simple method currently used is for voxel density values to decrease as a function of interaction time when removing material. Hence, when material is being added, the voxel density values will increase as a function of time. These time-dependent material removal rates will be further investigated and evaluated in future work. It is likely that an energy-based method will be applied to mimic a real situation. In this case, the transferred energy from the mill would be calculated and the material removal rate modeled as a function of transferred energy.

After changing the voxels’ density values, the Marching cubes algorithm is applied to the locally modified data at each frame in the graphics loop. This procedure updates the triangles only for the region that has changed. A new look of the surface is generated based on the new voxel values in the locally modified volume.

1.6.2 Haptic fall-through problem (presentation of various solutions)

A probe–proxy-based method is often used in haptic algorithms. The probe is the real position in the 3D space of the haptic device while the proxy is the virtual position of the device remaining on the surface of the manipulated object. A force is sent back to the haptic device based on the distance between the probe and the proxy.

The problem of “haptic fall-through” occurs when the haptic algorithm fails to perform the collision detection and the proxy falls through the surface. This is a well-known haptic problem, and the user recognizes it when the proxy is falling inside the object and there is no force feedback to the haptic device. The force will become zero when using a spring model-based force algorithm, because the distance between the probe and the proxy will be zero.

In our case, first a geometry-based haptic rendering method was tested using the rendered triangles for haptic feedback. When not removing material this method works well, but when updating the surface during milling there was a serious problem of haptic fall-through and lack of force feedback.

Therefore, a proxy-based haptic algorithm was developed and implemented to maintain a virtual milling tip position on the surface after a collision has occurred. In this algorithm, the voxel density values are used for haptic rendering, instead of the surface information as in geometry-based methods. Quite apart from solving the fall-through problem, another advantage is that this method will make it possible to use more sophisticated force algorithms in future work and to use density values as a basis for force modeling.

Different voxel-based haptic methods were developed, implemented, and tested before finding the most appropriate one for implementation. The rest of this section briefly describes the different tested methods.

As mentioned above, the first test was the geometry-based algorithm in which a haptic surface command was applied to the graphically rendered triangles. In this case the surface geometry information was used for collision detection and a spring force model was applied,
based on the distance between the probe and the proxy. The force command is directly called from the OpenHaptics software produced by Sensable Technologies (2006). This was an easy solution to arrive at, but the fall-through problem was serious and a better method had to be found. Fall-through happens when milling, because of pushing a rendered triangle and removing it to be able to graphically render a new one. At the moment when the triangle disappears there is no geometry to generate the haptic feedback, and fall-through occurs. The reason for this is the difference in updating frequencies between the haptic and graphic threads. The graphic thread is updated at 30 Hz while the haptic thread is updated at 1000 Hz. The removal of the triangles occurs in the graphic loop, so there will be updates of the haptic loop many times before new triangles are rendered in the next graphic update. Another problem with this solution is that the surface normals will be calculated based on triangles generated from surface information rather than the density values, and this gives a non-smooth haptic surface.

The second method developed and tested was a non-proxy, voxel-based method; the method is described in Figure 9.

### Haptic non-proxy voxel based method:

When collision (based on checking the voxel density values inside the sphere representing the tip of the mill): A force will be sent back to the device. The magnitude of the force is based on the voxel density value which is assumed proportional to the stiffness. (Illustrated as the length of the force vectors below). The force direction is based on the position of the voxels that are inside the sphere (the milling tip) relative to the center of the sphere. (Illustrated as the direction of the force vectors below).

\[
F_i = k_i \cdot (r - r_i) + c_i \cdot \frac{d}{dt} (r - r_i)
\]

Where \( F_i \) is the force from one voxel, \( k_i \) is the stiffness (density dependent value), \( r \) is the radius of the mill and \( r_i \) is the distance from a voxel to the center of the mill, \( c_i \) is an arbitrary damping coefficient.

All the different forces from each voxel are added as a vector sum to give the total force:

\[
F_{\text{tot}} = \sum_{i=1}^{n} F_i
\]

This total force, \( F_{\text{tot}} \), is sent to the OpenHaptics HD API for HAPTIC RENDERING at 1000 Hz.

---

**Figure 9.** The non-proxy based haptic rendering method
The method works very well if the mill is not pushed too hard against the surface. The surface feels smooth and very realistic and the stiffness can easily be adjusted by changing the stiffness factor. One drawback of the method is that the magnitude of the force is dependent of the resolution of the dataset. Another large problem with this algorithm occurs when the mill is pushed too hard so the entire sphere lies below the surface. In this situation, the vector sum of the forces becomes zero and the force feedback does not exist anymore. A solution to this problem is to have such a high stiffness constant for the surface that it is impossible to break through. However, this produces too much disturbance and uncontrolled vibration due to the limited force capabilities and bandwidth of the Omni device. This is an issue for future research.

We did not use the algorithm described above, but instead an algorithm developed by Vidholm and Agmund (2004) was implemented, tested, and evaluated. The method is depicted in Figure 10 and described in the following paragraph.

Discrete sample points on the surface of the tool sphere are used, $p$ being the proxy position and $x$ the probe position. The sample points in contact with the current object are used to define the normal component, $e_0$. The tangential direction, $e_1$, is constructed by projecting $x-p$ onto the plane defined by $e_0$. Then the proxy is moved a step in this tangential direction, as calculated at each frame in the haptic loop; a force proportional to the $x-p$ vector is sent back to the haptic device.

There is one drawback to using this method. When the surface is complex, as is typical when milling, it will be possible to move the proxy into the object where the proxy will remain (see Figure 11). The algorithm never checks whether the proxy is inside or outside of the object after moving it in the tangential direction. When the proxy is completely inside the object, the sum of all sample points will become zero; thus the normal of the surface becomes zero and the proxy will not move out of the object. The underlying concept of the method is good, but the method needs to be modified to fulfill the criteria of keeping the proxy on the surface of the manipulated object.
Based on these findings we developed a new algorithm differing in two main ways. First, the surface normals are calculated differently. Second, the algorithm is extended, incorporating a method that checks whether the proxy is inside or outside the surface after moving it in the locally estimated tangential direction.

The developed haptic rendering algorithm maintains the proxy in a position where the voxel density equals the density value used for isosurface generation. If the density value at the position of the haptic device is less than the density value of the isosurface, then the proxy position is updated to be the same as that of the haptic device. Otherwise, the proxy needs to be updated to minimize the distance between the haptic device and the proxy, while maintaining the requirement that the proxy remain in a position of equal density value as that of the isosurface. The distance between the probe and the proxy must be minimized so as to give the correct direction of the force to be sent back to the haptic device. The general approach of the algorithm is to update the proxy position using a two-step movement. First, the proxy is moved in a direction tangential to the surface by calculating the gradient at the center point of the sphere, based on the voxel density values and using vector projection as mentioned above. When the new tangential proxy position has been found, the point of intersection with the isosurface is derived by first computing the voxel gradient at this new location to determine a normal vector. Then the proxy is moved step by step along this normal vector towards the surface. In every step, the density value is computed to check whether the proxy is inside or outside the isosurface. The steps are performed iteratively until a point either inside or outside the surface is found, indicating an intersection with the surface. Linear interpolation between the last two points used will approximate the point that intersects the isosurface. By computing the gradient at this point, the proxy can finally be moved away from the surface by the radius of the proxy in this direction to ensure that the proxy is located entirely outside the surface.

The haptic force is then computed using a spring function between the haptic device (the probe) and the proxy. If the user has activated the milling mode, then a small random variation in the final force is added to simulate the vibration of the mill. The frictional coefficient can also be changed to produce different tactile sensations when touching the surface of the object. The algorithm is described in detail in Paper B and is verified by tests presented in the next section.
1.6.3 Verification of the developed proxy-based haptic algorithm

The developed haptic algorithm described above has been tested and verified for four different cases. A 3D cube has been modeled from a generated high-resolution volumetric dataset and used for the tests in the haptic milling simulator (see Figure 12).

![Figure 12. The cube used for the verification tests](image)

As a first verification analysis, the algorithm was tested using this basic geometry in the non-milling mode, i.e., there was no material removal. The virtual tip of the mill was dragged and pushed along a side of the cube. For the different test cases, the proxy position, the probe position (the real position of the haptic device), the spring distance between the probe and the proxy, and the actual modeled haptic force to the device were logged for analysis. The globally defined dimensions of the cube were also known and used for analysis. The four different test cases were as follows:

1. A stiff surface: high spring constant (between the probe and the proxy) and medium frictional coefficient (surface friction) (upper left in figures 13-15).
2. A soft surface: low spring constant and medium frictional coefficient (upper right).
3. A surface with high friction: high frictional coefficient and medium spring constant (lower left).
4. A surface with low friction: low frictional coefficient and medium spring constant (lower right).

Figure 13 presents graphs of the proxy and probe positions relative to the cube side for the four different cases. The results indicate that the proxy follows the surface very well in all cases. The force applied to the cube is different in all cases, as can be see in Figure 15, so it is hard to draw any conclusions about the probe position in the different cases. With a soft surface, the probe falls deeper into the material than in the case of a stiff surface, even though the force is smaller; this is as expected.
Figure 13. The proxy and probe positions relative to the pushed side of the cube in the four different verification cases.

Figure 14 depicts the length of the modeled spring between the probe and the proxy in the four different verification tests. In the cases with a low and a medium frictional coefficient, it is evident that the length of the spring is almost the same as the distance the probe falls into the object (compare these with the results presented in Figure 13). In the case with a high frictional coefficient, the length of the spring is greater than the distance the probe has fallen into the object. Hence, it is verified that a frictional surface gives a greater spring length (higher force), even though the probe is not pushed deeper into the object.
The force in the different cases is directly proportional to the length of the spring \((F = k \cdot (p_{proxy\_pos} - p_{probe\_pos}))\), as is clearly illustrated in Figure 15. In the soft surface case (using a low spring constant), it is evident that the force is significantly lower than in the other cases, even though the probe is pushed deeper into the object.
Based on the tests described above, it is verified that the principle of the haptic force algorithm works properly and produces the expected results for a specific 3D object built up from high-resolution volumetric data. Future work will include more extensive and general verification of cases with more complex geometries and involving material removal.
2. Summary of appended papers

2.1 Paper A: A Haptic VR Milling Surgery Simulator Using High-Resolution CT Data

This paper presents the underlying concept and development of a haptic and virtual-reality milling simulator using high-resolution volumetric data. The paper discusses graphic rendering as performed from an isosurface generated using Marching cubes and a hierarchical storage method to optimize for fast dynamic data changes during the milling process. A stable proxy-based haptic algorithm is used to maintain virtual tip position on the surface, avoiding haptic fall-through. The system is intended for use in educating and training surgeons for milling operations, such as complicated temporal bone operations used, for example, to remove brain tumors.

Agus et al. (2002), Pflesser (2002), and Sewell et al. (2005) are some of the other researchers also dealing with this problem. All this research in this field is still at an early stage, the solutions are deficient, and much more development is needed. The simulator developed here differs from other developed probes in its use of high-resolution datasets, which produce very realistic 3D visualizations of the milling process. The developed haptic rendering algorithm also improves on previous ones, by producing greater stability and reducing haptic fall-through problems.

This paper was presented at the 14th MMVR conference in Los Angeles, 2006.

2.2 Paper B: A Virtual and Haptic Milling Surgery Simulator

This technical report is an extension to paper A. In it both the developed graphic algorithms and the haptic method are presented in detail, in close connection with an explanation of how the C++ programming code was developed. The report describes how it is possible to import a DICOM file containing patient-specific data. This volumetric data is the only data used for both the graphic and haptic rendering of the milling process. The graphic rendering process is described in detail. An octree-based tree structure is used for efficiently managing the discrete data that is manipulated when milling. The tip of the mill is modeled as a sphere and a rapid bounding box algorithm is used to determine which voxels are affected by the milling. The density values of the voxels are used for calculating the surface normals; this produces a smoother surface than is possible using the more common geometry-based algorithms. Cache lists and the surface-based Marching cubes algorithm are used for efficient visualization and graphic rendering of the medical object, such as a skull or a tooth.

The developed haptic rendering algorithm uses the voxel density values for collision detection and for calculating the force sent back to the haptic device controlled by the surgeon using the simulator. A proxy-probe method is developed for realistic haptic feedback and avoidance of the commonly known “fall-through” problem when milling and
removing material. The developed technique keeps the proxy sphere on the surface of the object when milling. The distance between the proxy and the probe is used in the spring-based force feedback model.

The skull object used in this paper has the 512*512*174 resolution of the CT datasets, which gives 2,420,458 triangles rendered at a frame rate of 30 Hz. The haptic rendering loop is updated at 1000 Hz. The simulations have been tested on a Pentium 4, 3.2 GHz dual-processor PC with a Quadrow FX1400 graphics card; various tests indicate that the real-time demands are fulfilled for the given computational workload.

Apart from manipulating the VR representation of the object by milling/material removal, it is also possible to use cut planes, zoom, and rotation in the simulator to explore interesting regions of the data. Sound and vibration effects are also implemented to give a more realistic feeling to the milling process.

2.3 Paper C: The Use of Virtual Reality and Haptic Simulators for Training and Education of Surgical Skills

This is a survey study investigating the use of VR and haptic simulators in the surgical curriculum. The paper starts with a historical overview of VR and haptics, and of how interest arose in their application in the surgical field. Medical centers were late to start using VR simulators compared to aviation training centers, and the paper discusses the reasons for this. The current level of development and the forces driving the ongoing development of simulators are discussed.

The paper answers a range of questions concerning VR and haptic simulators and their use in medical centers. Why do medical centers need such training simulators? Particular types of surgical simulators have been developed and are in use today—why especially these and not others? Is there a need for simulators to simulate other kinds of operations? Who will use surgical training simulators? What is required of a simulator? How is simulator-based training correctly performed? Which is preferable, 2D or 3D visualization? Has the time arrived for telerobotic systems in real operation rooms? These questions are of concern for all those developing surgical VR and haptic simulators.
3. Conclusion, discussion, and future work

The research has focused on developing a haptic and VR surgical milling simulator. A paradigm shift is occurring in surgical education, from training on real patients to introducing haptic and VR simulators instead. This topic is still new and relatively unexplored, so much more research must be done to reach the goal of realistic simulators that mimic real operation procedures in a sufficiently realistic way. The simulator developed thus far in this project marks a first step towards training for temporal bone surgery, craniofacial surgery, dental tooth milling, and vertebral surgery using VR and haptic methods. Patient-specific data can easily be implemented and manipulated by the user in our simulator; both haptic and graphic real-time demands have been considered and verified for haptic feedback in three degrees of freedom.

The haptic force model used today is a simple spring model. Earlier research (Flemmer 2004) found that the position and relative impact velocity of the mill tip, the mill dimensions and rotational speed, and material data regarding the skull bone are important parameters for a realistic force model. Incorporating all these parameters would require far more complex force models than the currently used spring model. Using a voxel-based method would supply a basis for calculating a more realistic reflected force. Voxel density values can be regarded as a measure of the stiffness of the volume in which the mill is interacting with the material.

In future work we plan to investigate in greater accuracy the performance of various force models; after implementation, these force models will be benchmarked using both a teleoperator system and psychophysical experiments with real hand-held milling. Contact models of situations in which the mill is both turned on and turned off will also be tested and analyzed. The latter operational case is of great interest, as in some surgical situations the surgeon switches the mill off and pushes it carefully against a particular bone area, so as to judge the remaining bone thickness before breaking through.

When comparing different existing haptic milling algorithms presented in the literature, one conclusion is that modeling the mill/bone contact forces is difficult and that several completely different solutions exist, e.g. Agus et al. (2002), Peng et al. (2003), Pflesser (2002), Sewell et al. (2005), Vidholm and Agmund (2004), Wiet et al. (2002), and Zhengyi and Chen (2003). One can conclude from these sources that no algorithm exists that is able to generate sufficiently accurate force feedback for the milling case. Hence, it is necessary to develop and evaluate new algorithms for force feedback in this specific application. Future research will deal with these problems.

For calculating and visualizing the material removal rate during milling, an energy-based approach similar to that of Agus et al. (2002) will be implemented. As a basis for the removal of material, the attenuation value from the CT scan can be used as a measure of how much energy a voxel can resist before being removed. Combining the energy applied by the mill with the specific voxel attenuation values allows one to determine when the voxels should disappear and the milling process proceed in the material. The material removal rate depends...
on mill size and rotational speed and on the voxel density value. Currently, a simple time-dependent material removal rate is used.

There are three known ways to construct a haptic algorithm based on geometrical information for detecting collision between the probe and the VR object to be felt. These interaction methods are based on modeling the probing object as a point, a line segment, or a 3D object [Basdogan and Srinivasan (2001)]. Depending on the method chosen, different algorithms for generating the haptic feedback can be used. If a point-based interaction model is used, only a force is sent back to the 3-DOF haptic device. If a line segment is used, both forces and torques can be fed back to the haptic device; in this case, the user needs a 6-DOF haptic device to obtain correct feedback.

In the bone milling case, the probe is best represented as two 3D objects (a sphere and a cylinder) to obtain realistic force and torque feedback. This is especially important when milling in a hole, where collision detection is vital along the whole length of the probe for generation of realistic feedback (see Figure 16).

![Figure 16. The probe represented as a 3D object](image)

Most haptic systems in use today employ a point-based haptic collision detection algorithm. Lack of computer power and the complexity of the algorithms explain why more realistic models are not used. The present study uses a method based on voxel density values and one discrete point for detecting collision with the isosurface. This approach makes it easy to implement a greater number of discrete points along the border of the modeled mill, check all these points for collision with the isosurface, and then summarize all the partial forces to one total force. With this method both forces and torques can be calculated and sent back to the haptic device. However, the computational workload will be heavy and the real-time haptic demands cannot be met using current computer power.

The matter of multiple-point collision detection will be further investigated in future work. An algorithm to control this situation will be developed and implemented in the simulator. The required computational power will be investigated, based on the haptic and graphic rendering real-time demands.

In the developed simulator prototype it is possible to use cut-planes, zoom, and rotation to explore interesting regions of the modeled 3D object. Sound effects have also been
implemented to provide more realistic feedback to the user. In future research, we will investigate new 3D visualization concepts. To explore whether the PHANToM Omni haptic device reduces touch realism, even though its force models are accurate, we will evaluate how high a stiffness the haptic device can deliver.

Implementing more objects in the virtual scene and visualizing the dust produced during material removal will be done; this will improve the feedback to the user, and more closely mimic a real surgical situation. The performance of the simulator will be tested and validated by, for example, surgeons and dentists.

Future research will also focus on investigating the economic and ethical benefits of using simulators instead of real operations in training and educating surgery residents.
4. References


NYU Medical Center (2006), http://www.med.nyu.edu/research/cuttic01.html, accessed 2006-03-06.


