Emulation of surgical fluid interactions in real-time

Donald Stredney[1,2], Bradley Hittle[1], Hector Medina-Fetterman[4], Thomas Kerwin[1], Gregory Wiet[2,3]

1 Ohio Supercomputer Center, Columbus, OH
2 Ohio State University Department of Otolaryngology, Columbus, OH
3 Nationwide Children's Hospital, Columbus, OH
4 Research Consultant, Columbus, OH

Corresponding Author:
Thomas Kerwin[1]

Email address: kerwin@osc.edu

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

Emulation of surgical fluid interactions in real-time

Donald Stredney (1, 2), Bradley Hittle (1), Hector Medina-Fetterman (4), Thomas Kerwin (1), Gregory Wiet (2, 3)

1 Ohio Supercomputer Center, Columbus, OH
2 Ohio State University Department of Otolaryngology, Columbus, OH
3 Nationwide Children's Hospital – Columbus, OH
4 Research Consultant, Columbus, OH

Abstract – The surgical skills required to successfully maintain hemostasis, the control of operative blood, requires considerable deliberate practice. Hemostasis requires the deft orchestration of bi-dexterous tool manipulation. We present our approach to computationally emulate both irrigation and bleeding associated with neurotologic surgical technique. The overall objective is to provide a visually plausible, three dimensional, real-time simulation of bleeding and irrigation in a virtual otologic simulator system. The results present a unique simulation environment for deliberate study and practice.

Index Terms – Computer Aided Instruction, Computer Graphics, Computer Performance, Data Processing, Haptic Interfaces, Image Process, Medical Simulation, Parallel Algorithms, Parallel Processing, Stereo Vision, Virtual Reality

Introduction

The surgical environment is complex and exacting, requiring both a comprehensive knowledge of the regional anatomy as well as precise execution of interventional technique. A critical component of surgery is hemostasis, the control of operative blood flow. Hemostasis is both fundamental and essential to a wide variety of surgeries. To minimize blood loss to the patient,

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

the surgeon must remain attentive to the amount of blood flow. Throughout the surgery, controlling the amount of bleeding preserves a clear view of the operative site and reduces patient morbidity and mortality.

In neurotologic surgery, an iteration of techniques to "identify and expose" are conducted to maintain surgical awareness through identification of key anatomical landmarks. Once certain of the appropriate surgical location or depth of exposure, the surgeon uses a drill to deliberately remove thin layers of bone to expose structures encountered in the next operative level. Hemostasis is persistently sustained through proper control of both irrigation and suction. Irrigation is also used to mitigate the transfer of heat caused by the removal of bone, especially near critical neural structures. Ultimately, hemostasis requires the deft orchestration of bi-dexterous tool manipulation to successfully proceed from superficial to deep operative exposure.

We present our approach to emulate both irrigation and bleeding associated with neurotologic surgical technique. The overall objective is to provide a visually plausible, three dimensional, real-time simulation of bleeding and irrigation in a virtual otologic simulator system. The technique that we focus on is known as a mastoidectomy.

Related Work

Simulation of fluid dynamics has been a topic of interest since the early days of computing. Real-time, realistic blood simulation is relatively new, enabled in part by developments in highly parallelized computation on modern graphics processing units (GPUs). It is beyond the scope of this paper to provide an exhaustive review of the extensive literature related to fluid simulation. We review selected modern methods that have been intended for interactive blood

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

simulation. We follow the system of Qin [1] who characterized efforts to simulate bleeding as either grid-based or particle based.

Sweet et al. [2] proposed an image-based method utilizing texture mapping bleeding movies onto properly arranged surfaces. A Lattice Boltzmann Method (LBM) approach to real-time fluid simulation and rendering was demonstrated by Li et al. [3]. Other attempts to do the same type of simulation have been limited by having only portions as real-time, such as 2D blood sprites when hitting structures or by having pre-determined responses. Basdogen et al. [4] used a wave equation with a height field. They focused on realism and real-time performance.

Muller and others [5] developed an interactive approach to simulate virtual bleeding using smooth particle hydrodynamics (SPH). At the time of publication, that approach could model up to 3000 particles. In 2007, Pang et al. [6] presented a variation of SPH using a Physics Processing Unit PPU using a GPU marching cubes algorithm to accelerate rendering performance. More recently, Chong et al. [7] employed SPH to simulate hemorrhage in human injury, specifically ballistic trajectories. However, to the best of our knowledge, their simulation was not rendered in real-time as particle smoothing was handled in post-processing.

Previous temporal bone surgery simulators such as those described by Morris [8], Agus [9], Zirkle [10] and others do not incorporate water rendering or effects. We have previously introduced techniques utilizing 2D, grid based approach for representing real-time bleeding Kerwin, et al [11] 2009. A 2D method is substantially less computationally intensive; however it does not provide correct flow around 3D surfaces and cannot be realistically displayed in a stereo visual environment.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

Methods

Requirements

Our targeted application is the neurotologic procedure known as a mastoidectomy: the removal of portions of the mastoid bone. The mastoid includes the organs of hearing and balance, the motor nerve providing facial movement and expression, as well as the major vessels supplying and draining blood to and from the brain. Mastoidectomy is the initial step for a variety of surgical interventions to treat middle and inner ear pathologies, such as infection and tumors. Training for these techniques is traditionally performed in a dissection lab, using up excised cadaveric specimens. To be successful at the procedure, residents must integrate knowledge of regional anatomy with manual technique.

The requirements for emulating mastoidectomy in a virtual surgical simulation include:

- A visual update rate while drilling of at least 20 Hz, 15 Hz while rotating and reorienting the bone.
- Stereoscopic visual output to replicate the binocular microscope used in the surgery.
- A minimum 1 kHz update rate for haptic display.
- A minimum of 60Hz simulation update rate for fluid calculation.

In correct practice of a mastoidectomy, the surgical wound is irrigated with saline solution to clear blood and the bone debris from drilling in order to ensure visibility of the work area. The basic capability of controlling the amount of saline and suction with the tool in the non-dominant hand must be deliberately practiced to become proficient. Co-ordination of the suction/irrigation device with precise control of the drill that is held in the dominant hand is essential. There are some advanced physical simulators that incorporate bleeding. [12] However, setup and cleanup of such systems can be more burdensome than virtual computer simulation. Our goal is to emulate the orchestration of both suction/irrigation and drilling on a

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

mastoid bone with bleeding in a virtual computer simulation environment, into a single coherent experience (Figures 1 and 2).

Overall Architecture

We use direct volume rendering on the GPU to display the mastoid bone altered during the surgery. Haptics and real-time removal are processed on the CPU and the changes in the bone data due to removal from the drill are re-uploaded to the GPU for re-rendering. In order to integrate fluid interaction into our simulator, the representation of the collision conditions for the fluid simulation must be updated in real-time as well.

We use FLEX, a library created by NVIDIA and available to for free. FLEX provides fluid, fabric, and soft body real-time simulation for graphics applications, although we only use it for fluid interaction at this time. Collisions with polygonal objects like the instruments used in surgery are straightforward. However, in order the FLEX fluid simulation to interact with the mastoid bone, we must generate a signed distance field (SDF) representing distance to the bone in 3D space.

Signed Distance Field Creation

We must update the SDF of the mastoid bone in real-time to keep it up-to-date with the internal representation which changes due to drilling. This update need not be as fast as the haptic update rate, since small changes in the form of the bone might not be immediately apparent to the user. We have found that updating the SDF at the same rate as the visuals presents latency to the system. We have found that an SDF update rate of 200Hz is sufficient for plausible visual fidelity.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

The precision of the SDF very close to the edge of the object is important to maintain a physically plausible simulation, but values far away from the edge are much less important. Therefore, we use an algorithm that grows the distance field from the edge of the object.

All the following steps are implemented in CUDA, running on the GPU. Computing the full size SDF is not possible at the rates that we require, so we down-sample the volume so that every dimension is smaller than 128 voxels. Next, the distance is set for the boundary voxels (voxels that have a zero and non-zero voxel adjacent) to be zero while all other voxels are set to a distance value higher than the maximum length inside the volume (effectively a maximum possible distance). This boundary is then diffused outwards using a repeated local distance kernel, updating each voxel based on the minimum distance from its neighbors. The kernel makes use of shared memory on the GPU to reduce the number of global memory fetches. Global memory has substantially more latency than shared memory. The shared memory used per block is the size of the block, with a two voxel extended boundary in each dimension. Each thread corresponds to a voxel, and the volume is divided up into blocks each containing some number of threads. Inside the blocks, each thread copies data corresponding to its own voxel to shared memory and the boundary threads copy the boundary voxels. Then a 3x3 kernel performs the distance calculation, reading from shared memory. This greatly reduces the total memory lookup time as compared to using global memory directly. However, each execution of this kernel only expands the valid values in the distance field by one voxel in each direction. We perform 15 iterations of the kernel in order to get sufficient data for the FLEX simulation to work properly. After that, using the mask, we scale the values to correspond to the original (not subsampled) volume and test for location using the volume mask. The location test is used to set the sign of the distance field.

Bleeding

In traditional environments of training, residents are often introduced to temporal bone drilling through the use of 3D printed models or through cadaveric bones. However neither synthetic models nor cadaveric methods provide experience with controlling bleeding.

The regional anatomy involved in this application presents two types of bleeding: large scale and small scale. The first is associated with relatively large vascular arterial structures i.e., the carotid artery, and venous structures, i.e., the jugular vein, and sigmoid sinus. Violation of these structures results in considerable bleeding that must be immediately addressed. Small-scale bleeding comes from microvessels that transverse the compact bone. When exposed by the operative drill, they tend to gradually "ooze" or exude into the surgical wound. Because of the small size of the vessels the slow emission can be often attenuated by applying the diamond drill. This technique is effective because the diamond bit ablates the bone (as opposed to the "chipping" of the cutting burr). The resulting micro debris tends to pack into the openings and staunch the flow. Different burrs provide different rates of removal (e.g., the diamond burr removes bone more slowly than a tungsten-carbide burr), providing options that add to the realism of the system.

To identify soft tissue within the dataset, we seed-fill voxels starting from the sigmoid sulcus and determine if the voxel is in a cavity (empty) or if it is part of the bone. Termination conditions for the seed-fill are modified by parameters related to radius from bone structures and size of the fluid particles. This labeled mask is then used while drilling, allowing us to remove and insert bleeding particles on a voxel basis into the simulation.

Results

Visual quality

Although the results from our real-time rendering system are far from being photorealistic, the fluid moves similarly to the real fluid and has specular highlights, adding to the feeling of immersion in the simulation.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

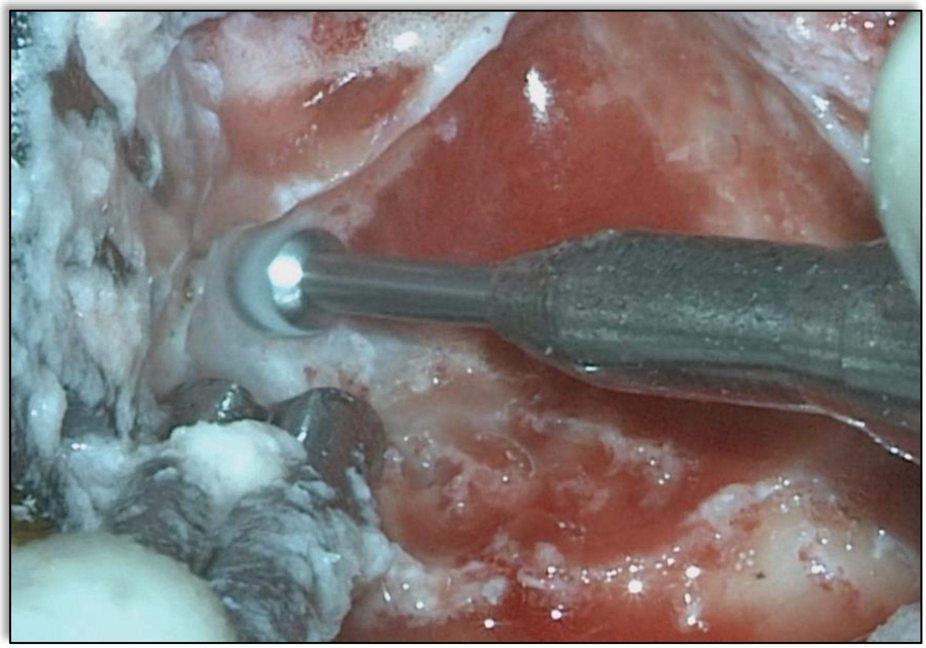

Figure 1: Image from actual surgery

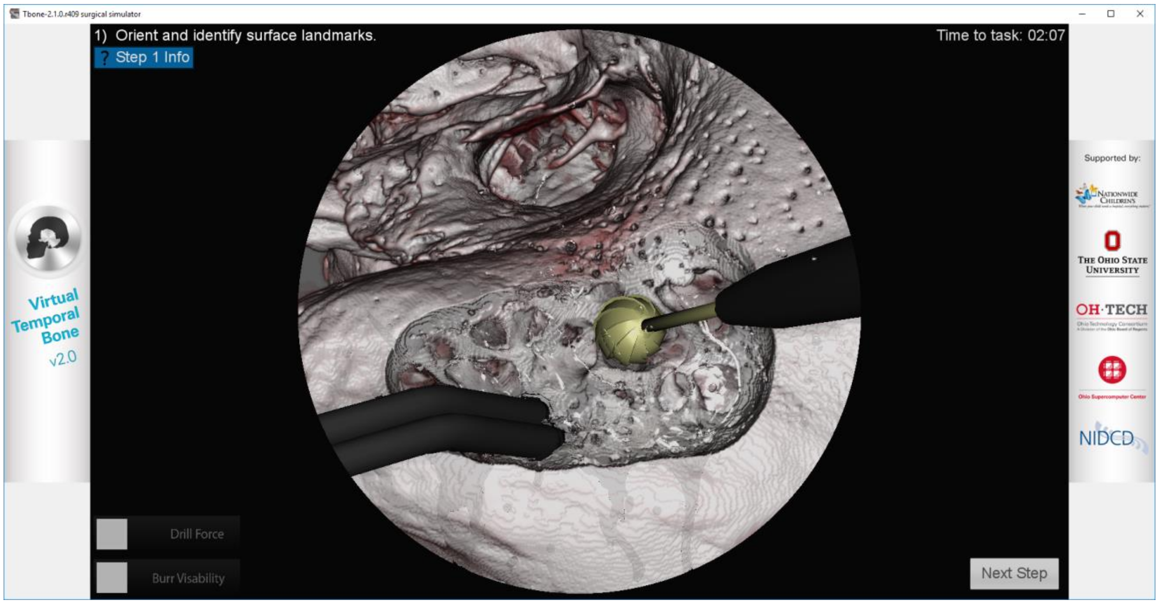

Figure 2: View from simulation showing drill and suction/irrigation device with saline

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

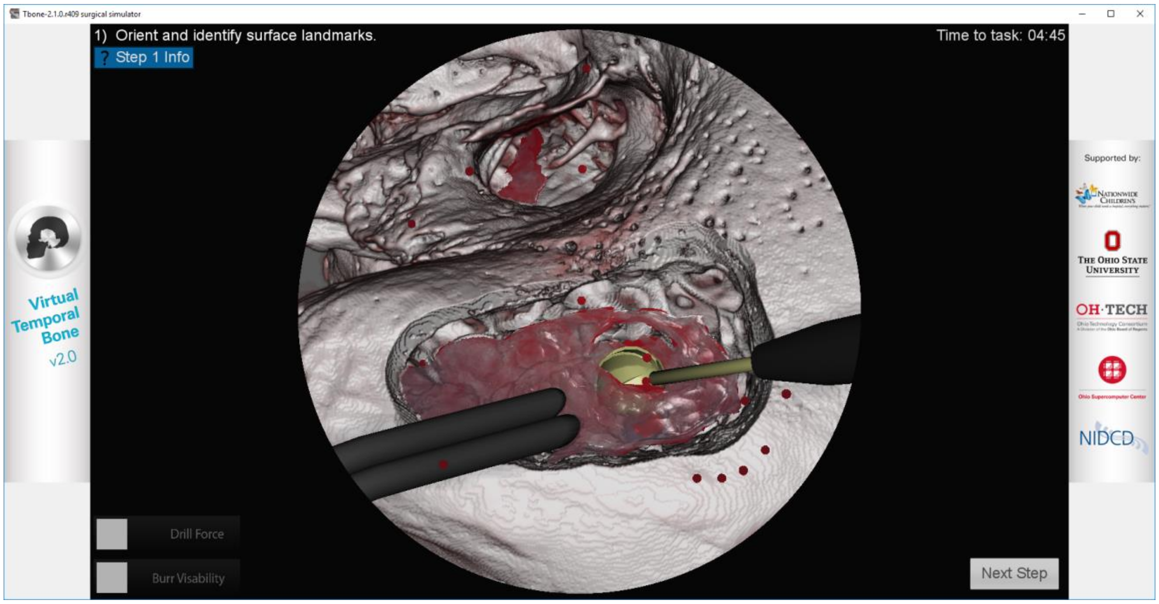

Figure 3 Image from simulation showing bleeding mixed by tool interaction

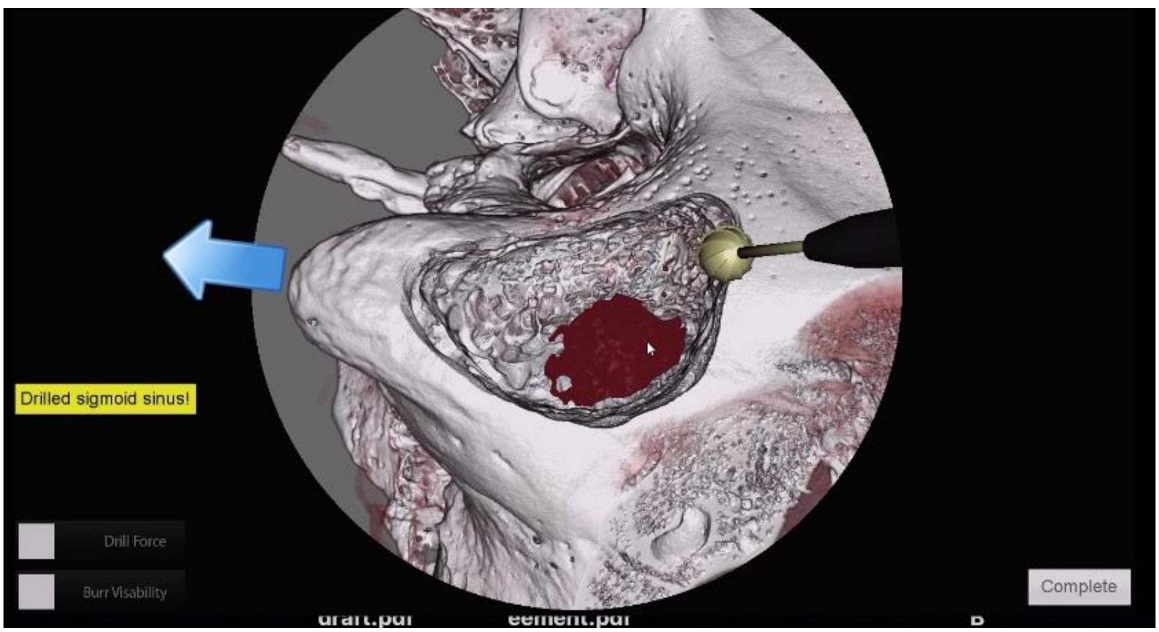

Figure 4: View from simulation immediately following violation of the sigmoid sinus

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

Performance

We tested the performance of the algorithm on Quadro cards using Kepler, Maxwell, and Pascal architectures. Using the newer architectures dramatically improves calculation time for the larger volumes.

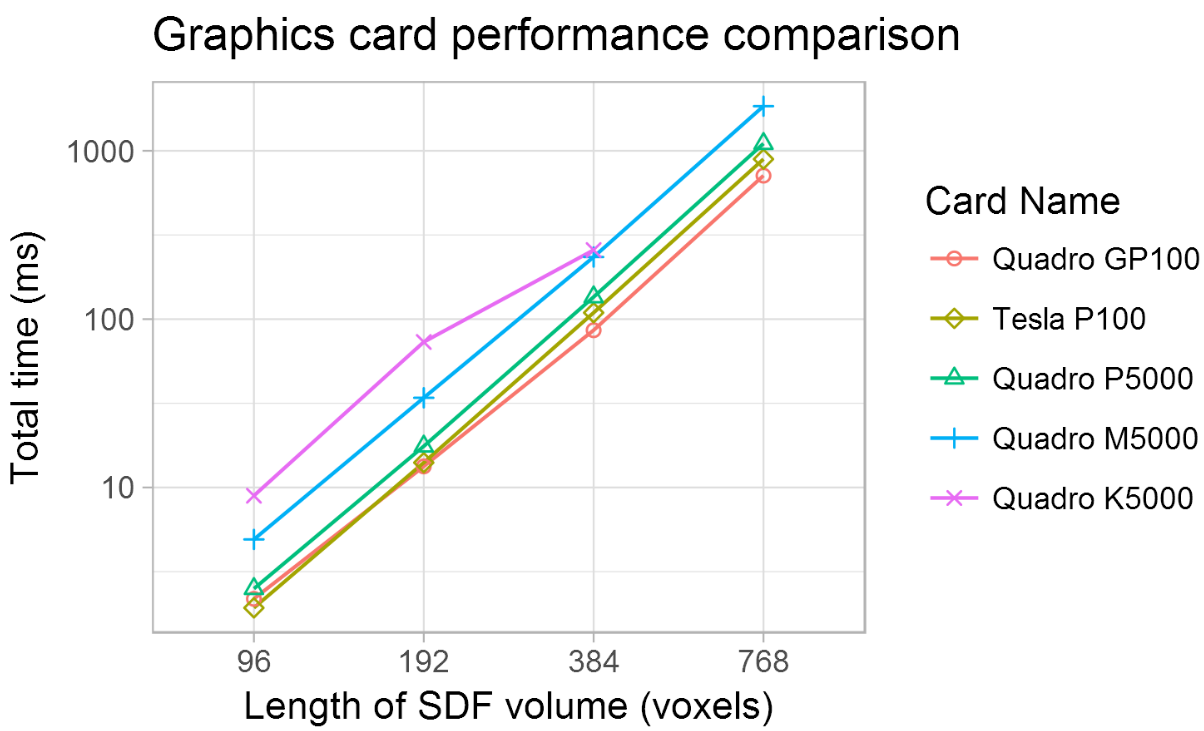

Figure 5: Calculation times for SDF at different resolutions using four different graphics cards. The vertical axis is time in seconds with a logarithmic scale; the horizontal axis is the size of the width dimension of a volumetric data.

Figure 5 shows the approximate SDF calculation with 15 shells using different GPU's at different resolutions. The subsampling step returns the original volume for all of these tests, not a subsampled dataset. The Quadro K5000 was unable to calculate a 768-cubed volume due to the algorithm requiring at least 4.5GB of memory for a 768-cubed dataset due to temporary buffers.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

A video showing the fluid effects is available at https://youtu.be/IKs_KsgXCLM

Discussion and Conclusion

Along with being a training environment, one of the goals of our system is to be a reproducible assessment tool. As a part of that assessment goal, we allow the reviewer to playback a recorded training session. There was some concern that due to the non-deterministic nature of the fluid algorithm, we would have discrepancies between playbacks in the visual output of the fluid. In fact, discrepancies do exist, but observers are hard-pressed to tell the difference between two different playback instances.

The performance of the system is quite acceptable at the 128-cubed voxel level. Higher resolution volumes, as shown in Figure 5, are prohibitive to use for real-time interaction.

We have presented our design and implementation of algorithms to provide a physically plausible integrated multi-modal representation of the neurotologic operative field, including the associated fluids. The results present a unique simulation environment for deliberate study and practice. Through independent exploratory practice, a novice gains time to consolidate their knowledge with performance and gain confidence in their abilities. By emulating the variant of bleeding, the system presents a less predictable environment, which user's find more realistic and engaging. It also presents an environment that provides immediate consequences to mistakes; allowing them to hone skills, explore improvisation, and attempt problem solving.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

References

1. Qin J., Pang W-M., Chui Y-P., Xie Y-M., Wong T-T., Poon W-S., Leung K-S., Heng P-A. 2007. Hardware-accelerated Bleeding Simulation for Virtual Surgery. In: Computational Biomechanics for Medicine II.

2. Sweet R., Porter J., Oppenheimer P., Hendrickson D., Gupta A., Weghorst S. 2002. Simulation of Bleeding in Endoscopic Procedures Using Virtual Reality. Journal of Endourology 16:451–455. DOI: 10.1089/089277902760367395.

3. Li W., Wei X., Kaufman A. 2003. Implementing lattice Boltzmann computation on graphics hardware. The Visual Computer 19:444–456. DOI: 10.1007/s00371-003-0210-6.

4. Basdogan C., Ho CH., Srinivasan MA. 1999. Simulation of tissue cutting and bleeding for laparoscopic surgery using auxiliary surfaces. Studies in Health Technology and Informatics 62:38–44.

5. Muller M., Schirm S., Teschner M. 2004. Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. Technology And Health Care: Official Journal Of The European Society For Engineering And Medicine 12(1):25–31.

6. Pang W-M., Qin J., Chui Y-P., Wong T-T., Leung K-S., Heng P-A. 2007. Orthopedics surgery trainer with PPU-accelerated blood and tissue simulation. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007:842–849.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.3334v2 | CC BY 4.0 Open Access | rec: 23 Feb 2018, publ: 23 Feb 2018

7. Chong K., Jiang C., Santhanam A., Terzopoulos D., Benharash P., Teran J., Eldredge JD. 2015. Video: Numerical simulation of hemorrhage in human injury. In: American Physical Society,. DOI: 10.1103/APS.DFD.2015.GFM.V0107.

8. Morris D., Sewell C., Barbagli F., Blevins N., Girod S., Salisbury K. 2006. Visuohaptic Simulation of Bone Surgery for Training and Evaluation. IEEE:48–57.

9. Agus M., Giachetti A., Gobbetti E., Zanetti G., Zorcolo A. 2004. Hardware-Accelerated Dynamic Volume Rendering for Real–Time Surgical Simulation. In: Workshop in Virtual Reality Interactions and Physical Simulations (VRIPHYS 2004).

10. Zirkle M., Roberson D., Leuwer R., Dubrowski A. 2007. Using a Virtual Reality Temporal Bone Simulator to Assess Otolaryngology Trainees. Laryngoscope 117(2):258–263. DOI: 10.1097/01.mlg.0000248246.09498.b4.

11. Kerwin T., Shen H-W., Stredney D. 2009. Enhancing realism of wet surfaces in temporal bone surgical simulation. IEEE transactions on visualization and computer graphics 15:747–758.

12. Adermann J., Geißler N., Bernal LE., Kotzsch S., Korb W. 2014. Development and validation of an artificial wetlab training system for the lumbar discectomy. European Spine Journal 23:1978–1983. DOI: 10.1007/s00586-014-3257-3.