2019

A virtual-reality training simulator for cochlear implant surgery

Blake Jones
The University of Western Ontario, bjones82@uwo.ca

Seyed Alireza Rohani
The University of Western Ontario, srohani4@uwo.ca

Nelson Ong
The University of Western Ontario, nong2@uwo.ca

Tarek Tayeh
The University of Western Ontario, ttayeh@uwo.ca

Hanif M. Ladak
The University of Western Ontario, hladak@uwo.ca

See next page for additional authors

Citation of this paper:

Jones, Blake; Rohani, Seyed Alireza; Ong, Nelson; Tayeh, Tarek; Ladak, Hanif M.; Chalabi, Ahmad; and Agrawal, Sumit K., "A virtual-reality training simulator for cochlear implant surgery" (2019). *Electrical and Computer Engineering Publications*. 591.
https://ir.lib.uwo.ca/electricalpub/591

Authors

Blake Jones, Seyed Alireza Rohani, Nelson Ong, Tarek Tayeh, Hanif M. Ladak, Ahmad Chalabi, and Sumit K. Agrawal

A virtual-reality training simulator for cochlear implant surgery

Blake Jones, Seyed Alireza Rohani, Nelson Ong, Tarek Tayeh, Ahmad Chalabi, Sumit K Agrawal*, Hanif M Ladak*

*co-senior authors

Abstract

Background and Objectives: Hearing loss is one of the most prevalent chronic conditions and can significantly impact an individual's quality of life. Cochlear implantation (CI) is a widely applicable treatment for severe to profound hearing loss, however CI surgery can be difficult for surgical trainees to master. Training environments that are safe, controlled, and affordable are needed. To this end, we present a virtual-reality (VR) cochlear implant surgical simulator developed with a popular, commercial game engine. **Method:** Unity3D was used to develop the simulator and model the delicate instruments involved. High-resolution models of human cochleae were created from images obtained from synchrotron-radiation phase-contrast imaging (SR-PCI). The physical-realism of the simulator was assessed via a comparison with fluoroscopic images of an actual cochlear implant insertion. Different resolutions of cochlear models were used to benchmark the real-time capabilities of the simulator with the number of frames per second (FPS) serving as the performance metric. **Results:** Quantitative analysis comparing the simulated procedure to fluoroscopic imaging revealed no significant differences. Qualitatively, the behaviour of the inserted and simulated implants were similar throughout the entirety of the procedure. The simulator was able to maintain 25 FPS even when experiencing an artificially high computational load. **Conclusion:** VR simulators provide a new and exciting avenue to enhance current medical education. Continued use of widely available and supported game engines in the development of medical simulators will hopefully result in lowered costs. Preliminary feedback from expert surgeons of the simulator presented here has been positive and future work will focus on evaluating face, content and construct validity.

Keywords: simulation, surgical training, cochlear implants, gaming engines

1. Introduction

One of the fundamental ways that humans interact with the world is through sound. Speech, tone and other verbal cues play a vital role in human communication. For these reasons, hearing loss can have devastating implications on an individual's quality of life. The anatomy that facilitates hearing is diverse in both function and form, as are the causes of hearing loss. Damage to any one of the tympanic membrane (eardrum), the ossicles (delicate bones of the middle ear), or the cochlea (spiral-shaped bone that transmits sound to the nervous system) may result in partial or complete loss of hearing.

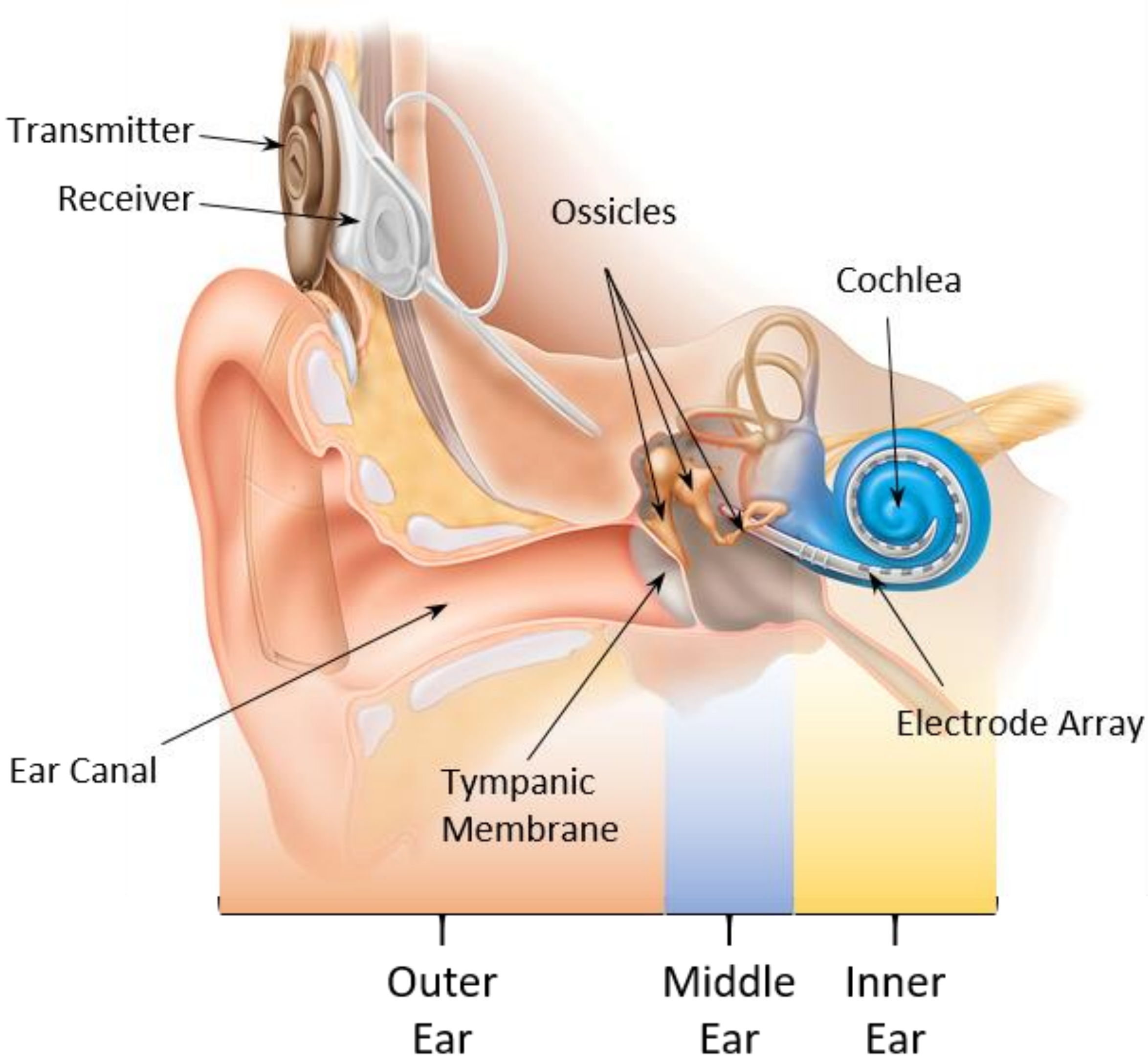

Figure 1. Ear anatomy and cochlear implant components. (Wikimedia Commons, 2008)

Cochlear implantation (CI) is a surgical procedure that aims to remedy severe to profound sensorineural hearing loss via the insertion of a thin, conducting electrode into the cochlea. Figure 1 presents a simplified view of a cochlear implant and relevant ear anatomy. The transmitter of the CI contains the microphone and processor. This sends the processed sound signals and energy to the implanted receiver via radio frequency transmission. The electrode permanently resides within the cochlea, where it directly stimulates the auditory nerve. Cochlear implantation differs from hearing aids in that it does not amplify sound from the environment, but instead directly translates sound into an electrical signal. Since the advent of the modern cochlear implant in 1977, over 300,000 devices have been implanted in patients around the world (NIDCD, 2017). With reports predicting that the cochlear implant market will continue its growth and reach 3.1 billion USD by 2025 (Grand View Research, 2017) the need for trained surgeons to perform these implants is increasing.

The cochlear implantation procedure consists of three parts: 1) incision and drilling of the mastoid process (bone located immediately behind the ear), 2) implantation of the internal receiver into the mastoid process, and 3) insertion of the electrode into the cochlea. Electrode insertion is especially important, since the final positioning of the electrode may have profound implications on patient-related outcomes (Finley *et al*, 2009). In addition, the insertion process is associated with risk to the fragile anatomical features of the cochlea which, if damaged, can worsen the hearing of a patient (Connell, Hunter, & Wanna, 2016). Surgeons must use meticulous surgical technique and tactile feedback to ensure atraumatic insertions.

The traditional method of surgical education is an apprenticeship approach that is rooted in William Stewart Halsted's motto of "See one, do one, teach one." Although refined over the years, the core of this pedagogical formalism has remained a cornerstone of surgical education. However, recent studies assessing the learning curve in new surgical techniques, particularly laparoscopic procedures, have found that the number of surgeries required for surgeons to perfect their abilities can range from 250-750 (Coles, Meglan, & John, 2011; Secin et al., 2012). It is therefore not surprising that Halsted's approach has been the topic of much criticism, and calls for new tools in medical education have been made (Kotsis & Chung, 2013).

One such tool that has become increasingly popular is computer-based surgical simulation. With the recent advancements in technology, computer simulation has become an effective option to help train surgical residents (Nagendran, Gurusamy, Aggarwal, Loizidou, & Davidson, 2013). In addition to providing a novel tool with which to train, the digital nature of these systems lends itself well to providing detailed, quantitative feedback, which can facilitate user learning.

A critical shortcoming of many early surgical simulators was the inability to provide a tactile response. The importance of a realistic sense of touch in surgical skill development has been noted in the literature, and incorporation of haptic devices into surgical simulators has become essential (Escobar-Castillejos, Noguez, Neri, Magana, & Benes, 2016; Prasad, Manivannan, Manoharan, & Chandramohan, 2016).

Although simulators show promise in facilitating higher throughput of well-trained clinicians, the technical burden of developing a new simulator can be high particularly as a result of the low-level languages many modelling frameworks are written in. To this end, this work presents a virtual-reality (VR) simulator for cochlear electrode insertion that incorporates haptic feedback to provide an immersive, realistic experience, which may augment surgical skill development during the preclinical stage of medical education which was developed entirely using a popular, commercial game engine.

2. Methods

2.1 System Usage

High resolution models of the cochlea are imported into Unity3D (Unity Technologies, San Francisco CA). To begin, the user may select from one of these preloaded options or import their own model. After selection, the chosen model and the electrode are rendered into the scene.

The haptic device is now the primary user interface with the system and has control over the position and rotation of the cochlea and electrode, which move synchronously with one another. By translating and rotating the haptic arm, the user can align the scene to their preferred orientation. The outward walls of the models can also be made transparent for increased visualization of the interior structures.

Upon positioning the scene to his/her satisfaction, the user can press a button on the haptic arm which detaches the cochlea model from the haptic device. The electrode now moves independently of the cochlea and electrode insertion can begin. Throughout the insertion, usage statistics and performance metrics are tracked to provide feedback. Performance metrics include: insertion time, number of successful insertions, failed attempts, average insertion depth, total time spent training, and number of simulator resets. The user continues inserting the electrode until they indicate they are done or would like to restart. After training is complete, the user may review the performance data that was collected and debrief on their experience.

2.2 System Design

The simulator was created using the popular game development platform Unity3D (Unity Technologies, San Francisco, CA). Unity3D is fully compatible with the Windows graphic APIs Direct3D and Direct X 12, and fully supports NVIDIA VRWorks. Unity3D served as a hub to integrate the various components, handle the graphic and haptic rendering, and perform the physical modelling. A diagram of the system architecture is shown in Figure 2.

Although Unity3D provides a GUI to streamline the development process, additional C# code was required to implement the flexible electrode, customize the VR experience, interface with the haptic device, and provide debriefing data. Unity3D relies on the Update and FixedUpdate functions to handle visual updates and physical calculations, respectively. Therefore, efforts were made to keep these functions as simple as possible and make use of event-driven programming to increase efficiency.

When the simulation begins, the Unity3D game-loop starts. This consists of repeated calls to Update and FixedUpdate. FixedUpdate calculates the position of all the meshes included in the scene and communicates with the haptic C# wrappers which, in turn, communicate with the device drivers provided by 3D Systems Inc (3D Systems Inc, Rock Hill SC) for their Touch haptic device. The Update function then visually renders the scene and provides the Oculus Rift (Oculus VR, Menlo Park CA) headset with the result.

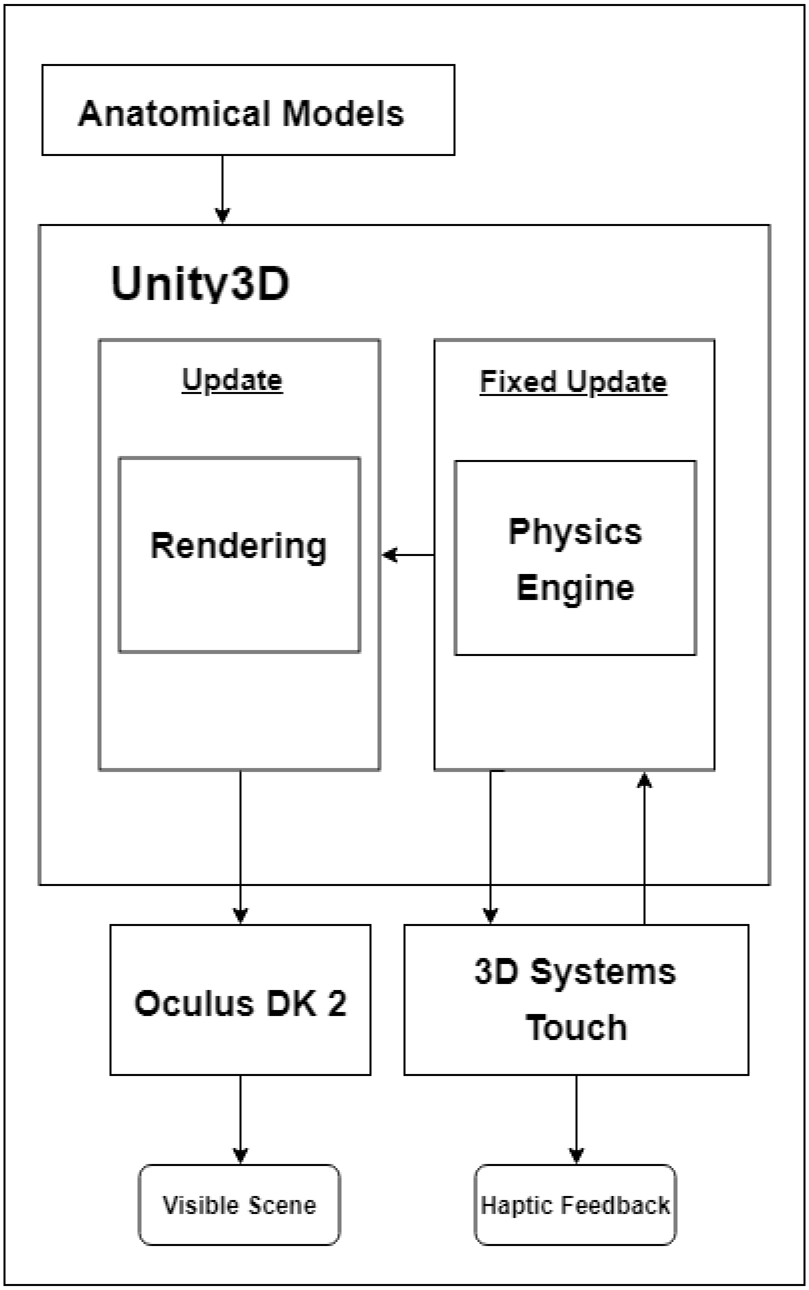

Figure 2. Simulator software design. Square boxes represent an internal component and rounded boxes represent output. The Oculus DK 2 and 3D Systems Touch boxes correspond to the VR headset and haptic devices, respectively (including their drivers and applications).

2.3 Hardware

The simulator was developed and tested primarily on an Asus ROG laptop with an Intel i7 quad-core processor and an NVIDIA GTX 960M GPU running Windows 10 as an operating system. The Oculus Rift DK2 was used to allow for an immersive user experience. The use of a VR environment has the additional benefit of providing a sense of depth, which not only improves the user experience of the simulator but also lends it additional realism. When a VR display was not present, the simulator was rendered on the laptop screen, which was a 15-inch, 1080p display. Unity3D provides extensive support for Oculus VR development. To allow for VR and non-VR use of the simulator, scenes included an additional camera exclusively for VR use. A monitor script checked for the presence of a VR device at run-time and adjusted the simulator accordingly.

Haptic support was enabled by the 3D Systems Touch device. This device provides six positional degrees of freedom (DOF) and three DOFs for haptic feedback. The Touch device was used to control the position and movement of the electrode and provide the tactile response upon contact with the cochlear anatomy. Unity3D does not natively provide haptic device integration, so C# wrapper functions were created to interface with the 3D System Inc plugins.

2.4 Cochlear Models

High resolution (9 µm voxel size) images of multiple cochleae were collected from a previously published study using synchrotron-radiation phase-contrast imaging (SR-PCI) (Elfarnawany et al., 2017). Using 3D Slicer (Fedorov et al., 2012) and Geomagic Studio (3D Systems Inc, Rock Hill SC) a 3D model of the cochlea showing clinically-important internal structures was developed. The open-source software Blender (Blender Foundation, Amsterdam Netherlands) was then used to create an external (outward-facing) surface. This was required since Unity3D only supports single-sided geometry. Therefore, a duplicated surface of the cochlea was created and fitted to the existing surface. The new surface then had its normal vectors flipped. Both the single-sided and double-sided models were imported into Unity3D. Using the Unity3D editor, mesh colliders were applied. The use of single-sided and double-sided mesh geometry allows the user to select whether the simulated cochlea's external faces are transparent. The mesh colliders were required for the detection of collision events with the electrode during the simulation. Anatomical features, such as the basilar membrane (an intracochlear membrane which should be kept intact during insertion), were segmented from the SR-PCI data independently, and developed into their own 3D models before being combined in Unity3D. By doing this, the different physical and haptic features could be applied to different anatomical structures in an efficient manner.

2.5 Electrode Modelling

Since Unity3D does not natively support bendable objects, a custom implementation is needed to simulate many medical procedures. The primary concerns of the electrode simulations were physical realism and efficiency. By modelling the physics of the electrode in real-time, trainees can visualize the electrode coiling into the cochlea, which is not possible during actual surgery due to the opaque bone.

To achieve these goals, the electrode was simulated using a mass-spring model. Mass-spring models have long been known to realistically simulate deformable objects and offer several computational advantages in the Unity3D framework (Basdogan, Ho, & Srinivasan, 2001;

Luboz, Blazewski, Gould, & Bello, 2009). The “masses” of the electrode object were represented by capsules (a Unity3D primitive object). This greatly reduced the number of draw calls required to render the electrode, since Unity3D supports GPU instancing to draw multiple copies of the same mesh (Unity3D, 2018). Each capsule was connected by a Unity3D spring-joint. Spring-joints have customizable physical properties, such as a spring constant and coefficient of damping. These properties were qualitatively tuned to best resemble an electrode by consulting a practicing ear surgeon with over a decade of experience in cochlear implant surgery.

In addition to the capsule and spring-joint system, the individual cells of the electrode had scripts and components attached. Each segment of the electrode had an associated rigid-body and a trigger collider. The rigid-body ensured the segment was governed by the physics engine, and the trigger collider provided the ability to raise collision events when specific areas of the electrode entered defined regions of the cochlea.

2.6 Assessment of the Simulated Electrode Model

The behaviour of a cochlear implant electrode during insertion has important ramifications for the overall success of the procedure. Limited imaging of an electrode during insertion exists due to the difficulty involved in acquiring such data. To validate the simulated electrode, images from a fluoroscopic video of an actual insertion into a human cochlea were obtained, courtesy of Med-El GmBH (Innsbruck, Austria).

A visual comparison with the fluoroscopy data was encouraging, however a quantitative assessment of the behaviour of the simulated electrode was desired. To accomplish this, a novel approach to compare the shape of the electrodes was developed. The method simultaneously considered three consecutive, inserted segments of the electrode and obtained an angle representing the local deformation of the electrode at a particular point.

As shown in Figure 3, the first angle was calculated by connecting the centroids of segments 1 and 2, and 2 and 3, respectively. The angle between these lines was then recorded. This process then continued using segment 2 as the initial electrode. The final angle was calculated between the lines formed by the centroids of segments 5 and 6, and 6 and 7, respectively. Therefore, every time an additional segment of the electrode entered the cochlea, this set of measurements could be repeated with the newly inserted segment allowing for an additional angle to be calculated. As a result, this method allowed for an evaluation of the electrode bending not only in its final configuration, but at several stages throughout the procedure.

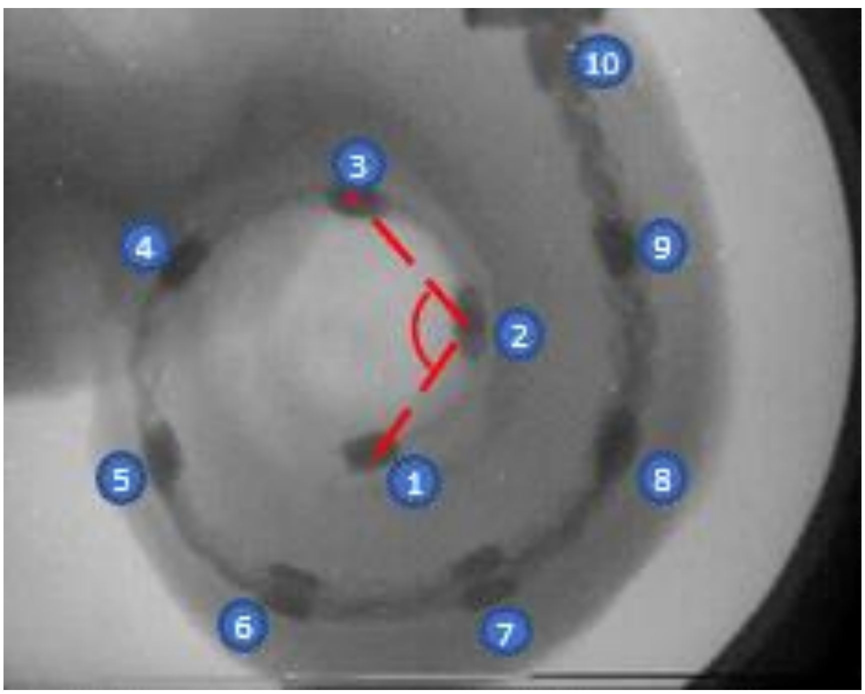

Figure 3. A fluoroscopically-obtained image of an electrode inserted into a human cochlea with the electrode segments numbered (segment 1 being the deepest).

Using this method, angles were measured for both the fluoroscopic data and simulation data. The results from multiple simulations were collected and averaged. Both sets of measurements were found to be normally distributed using the Shapiro-Wilk (Shapiro & Wilk, 1965) and Anderson-Darling tests (Stephens, 1974). Therefore, a paired *t*-test was used to evaluate the significant difference between the fluoroscopic and simulated measurements. Significance was set to $p < 0.05$. Apart from significance testing, the percent error from each measurement was assessed via the formula presented in Equation 1, where S is the value obtained from the simulated electrode and F is the value from the fluoroscopic data.

Equation 1: $$Percent\ Error = \frac{S-F}{F} \times 100\%$$

3. Results and Discussion

3.1 Simulator Performance

The primary goal of this study was to design a simulator to realistically model the behaviour of an electrode during cochlear-implant surgery. It is anticipated that this will improve the understanding of how manipulations performed to the electrode affect the position of the implant during insertion in real-time. As such, the real-time performance of the simulator was of paramount concern. Despite large strides forward in terms of performance, Unity3D has been known to struggle with complex mesh rendering (Cristina, Dapoto, Thomas, & Pesado, 2018). To assess the effectiveness of the simulator, a study of the frame rate as a function of the number of triangles rendered was conducted. Triangles are considered an important aspect of computational load as they form the base unit of graphic rendering. Furthermore, the number of triangles directly correlates with not only the visual fidelity of the model, but also the physical accuracy.

Figure 4 shows the recorded frame rate for three cochlear models with a varying number of triangles. Also included in the figure are the frame rate results for a simple cube mesh. To reduce the dependence of the frame rate on specific actions or events, a script was used to move the electrode forward into each of the models at a constant speed. Repeated measurements were taken and then averaged.

The frame rate results assisted in choosing an appropriate balance between visual fidelity of the cochlear models and simulator performance. The angle of insertion of the electrode into the cochlea and the speed at which it was inserted were chosen to maximize the number of collisions that the electrode would undergo. Therefore, all the models start at a very similar FPS but are quickly stratified according to mesh detail. The simple cube mesh illustrates the known ability of Unity3D to handle simple meshes and Unity3D primitives with greater efficiency than more complex meshes with a similar level of detail (Cristina et al., 2018).

A threshold of 20 FPS was deemed as an acceptable minimum to ensure the real-time functionality of the simulator. The smallest model provided a frame rate that was consistently above 20 FPS, even when Unity3D was experiencing an unrealistically high number of collisions. The 359,400 triangles that composed this model offered more than enough detail to accurately render intricate anatomical structures of the SR-PCI models. Figure 5 presents two screenshots from the simulator using cochlear models with a similar level of fidelity as the 359,000 triangle benchmark.

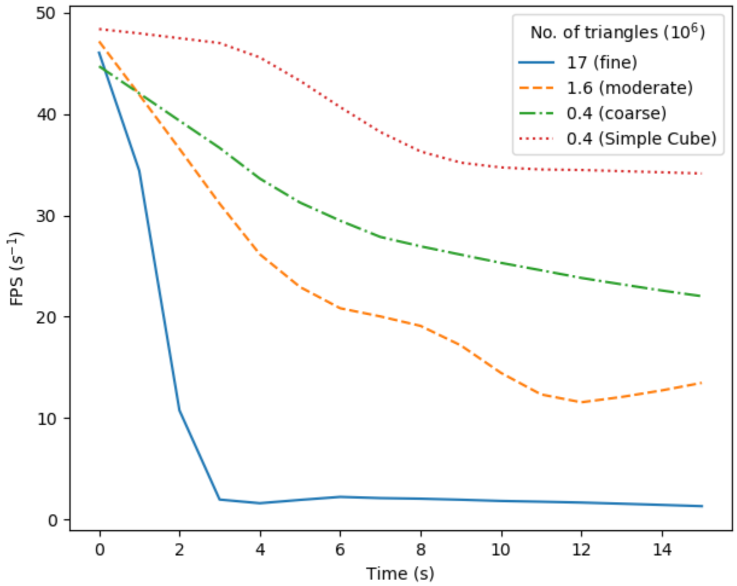

Figure 4. The FPS as a function of time for varying levels of mesh triangles and complexity. FPS is the frames per second of the simulator. Time is correlated to the number of collisions the simulator was modelling, kept consistent by the controller script (see text for description). As time progressed, the simulator was forced to handle an increasingly large number of active triangles and collisions, hence the declining FPS with time. The simple cube line indicates FPS using a simple cube mesh while all other lines result from cochlear meshes. Locally weighed scatter plot smoothing (LOWESS) with a factor of 0.45 was used.

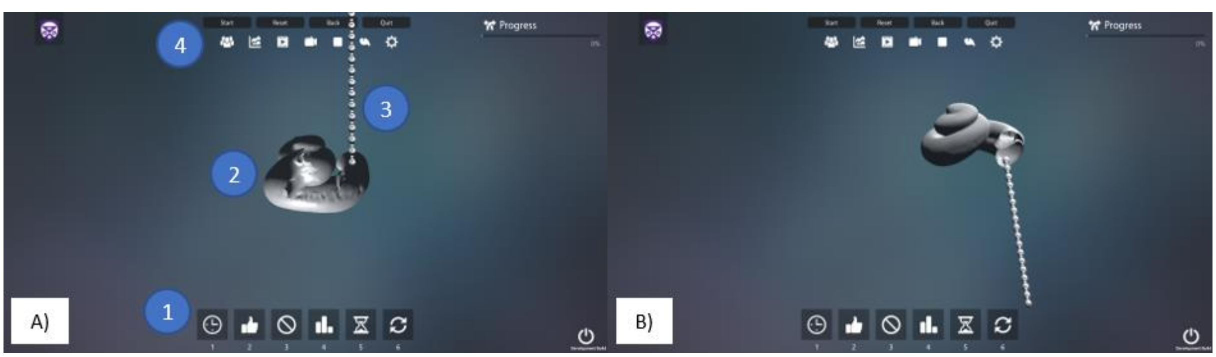

Figure 5. Screenshots of the training scene of the simulator. A) electrode and simple cochlea. The cochlea has its external faces made transparent allowing for observation of both the internal anatomy and the electrode during insertion. 1: Basic usage statistic display (insertion time, number of successful insertions, failed attempts, average insertion depth, total time spent training and number of resets), 2: Cochlear model, 3: The electrode, 4: UI options to modify simulator settings, begin/end a recording, select a different cochlea model, etc. B) The electrode with a more complex cochlear model that includes additional anatomical features. The external faces of the model have been rendered and thus the outward faces of the model are no longer transparent.

3.2 Simulated Electrode Model

The modelled electrode was an important component of the simulator. Our electrode model had a mean angular deformation of 127° and standard deviation of 19.4°, whereas the mean angular deformation of an actual electrode was 129° with a standard deviation of 17.5°. A paired t-test did not reveal a significant difference between the simulated and actual deformations at a p-value of 0.05 ($t=-0.14$, $p=0.89$, DOF=8). Figure 6 highlights the qualitative performance of the simulated electrode model. From the comparison of the simulated electrode to the fluoroscopic images, the behaviour of both is consistent throughout the procedure.

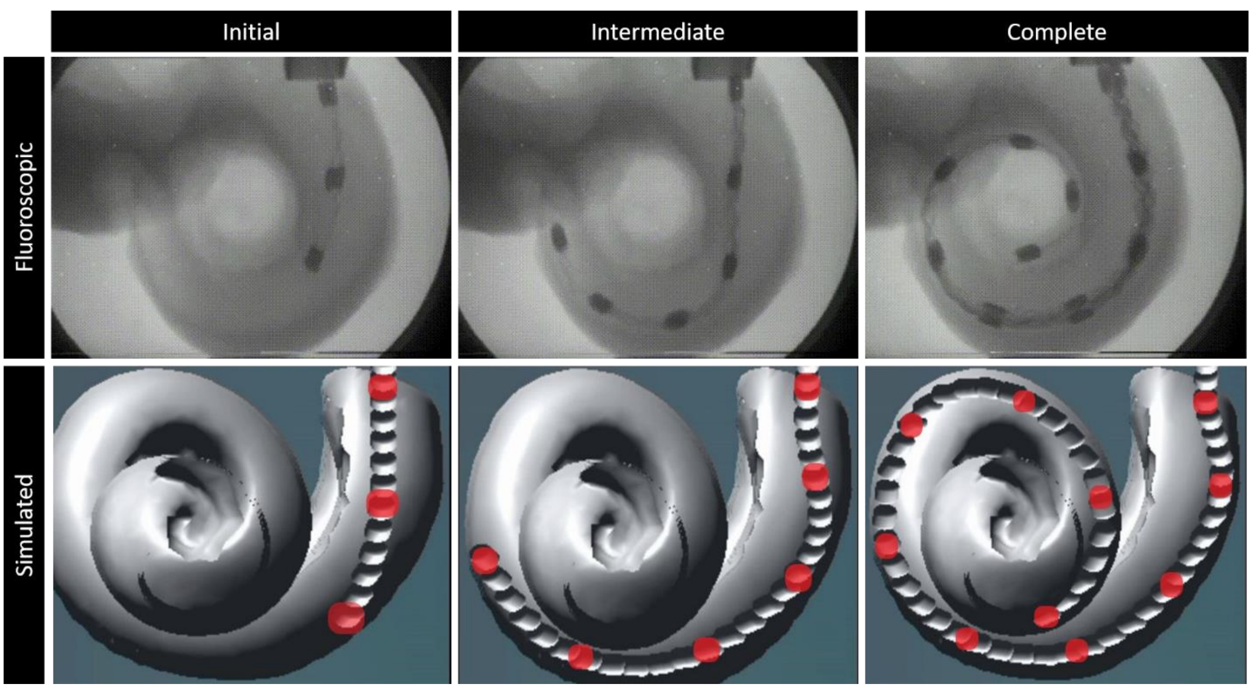

Figure 6. Video frames from a fluoroscopic insertion of an electrode compared to screenshots of an insertion of the simulated electrode at various insertion depths. The red circles indicate the position of the individual electrodes to facilitate easy visual comparison.

The kernel density estimation of the percent error for all measurements, calculated from Equation 1, is shown in Figure 7. The maximum deviation that occurred between the averaged measurements was 7.2% and the average deviation was <1%.

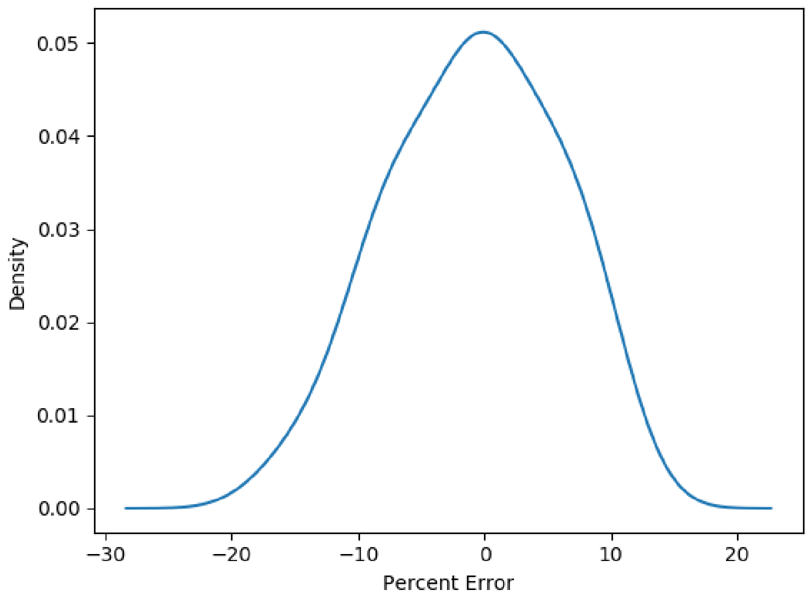

Figure 7. A Kernel Density Estimation plot of the percent difference between the fluoroscopic and raw simulation data with a mean and standard deviation of 0.83% and 6.57% respectively.

3.3 Debriefing Functionality

An extremely useful, but often underutilized, benefit of virtual surgical simulators is their ability to autonomously collect data as students are being trained. The simulator presented here tracks basic usage statistics, and provides instance-specific data on each electrode insertion attempt. Two categories of instance-specific data were collected: 1) force as a function of depth, and 2) depth as a function of time. These statistics are useful for allowing users to debrief after using the simulator and also serve as the basis for automatic error detection and future discriminant validity studies. This data can be examined in application or exported as a JavaScript Object Notation (JSON) file. In addition to numerical feedback, the system allows for users to record their attempts as a video for self or expert review.

3.4 Advantages and disadvantages of Unity3D

A major contribution of this paper was providing a benchmarked example of a surgical simulator that adopted popular gaming technologies. In 2007, Marks *et al* conducted one of the earliest examinations of the potential of game engines in relation to serious game development (Marks, Windsor & Wünsche, 2007). They concluded that the physical realism and technological abstraction that game engines provided was well suited for serious game or simulation development but that the performance was not yet at an acceptable level. More recently, the latency between Unity3D developed applications and VR headsets was probed and found to be comparable to low-level frameworks (Le Chénéchal & Chatel-Goldman, 2018). By providing quantitative outcomes of this system's performance we aimed to update the literature regarding the feasibility of medical simulation on a widely available game engine platform.

Unity3D was chosen for the development of the simulator due to its increasingly large presence in professional game development and for its reputation as being a user-friendly framework (Craighead, Burke, & Murphy, 2008). The powerful GUI, highly familiar workflow and wide developer base of Unity3D make it well positioned to reduce the technical cost of simulator development compared to other existing frameworks, like SOFA, discussed below.

Since version 5.0, the personal license of Unity3D supports automatic occlusion culling (Unity3D, 2015). Occlusion culling allows for the omission of objects that are not currently visible from the rendering loop. Depending on the particulars of the scene, this can provide a significant reduction in the rendering cost of a scene which often translates to a marked improvement in overall efficiency. Messaoudi *et al* found that the rendering pipeline was, in some instances, responsible for 92% of the CPU load. Another tool that can have profound implications to graphical processing costs is GPU instancing. The use of GPU instancing for this simulator meant that all the segments of the electrode could be drawn in one batch call, thereby preventing a large amount of overhead to draw copies of the same mesh.

Apart from graphical rendering, simulations often have an emphasis on realism, which requires accurate physical calculations. The fixed time step in Unity3D controls the time step between physical calculations. A smaller time step can result in more accurate calculations but can significantly impair system performance. Previously, our group has reported the impact of the fixed time step on Unity3D's frame rate (Huang, Agrawal, & Ladak, 2016). A fixed time step of 0.002 seconds was selected as an acceptable trade-off between accuracy and framerate.

An interesting area of Unity3D that was not utilized in this work is the third-party, performant frameworks that are beginning to interface with Unity3D through the use of application programming interfaces (APIs). Simulation Open Framework Architecture (SOFA) (Allard et al., 2008), a framework written in C++, has specialized in realistic surgical simulation software, runs smoothly in real-time, and is currently in the process of releasing a Unity3D integration. Currently, integration of Unity3D and SOFA is being led by InfinyTech3D (Nice, France) and the first proof-of-concept of SOFA-controlled deformable meshes has recently been released. Although not yet entirely mature, the ability to use third-party libraries like SOFA bodes well for Unity3D's continued role as a platform to develop serious games and simulators.

3.5 Future Work

Using the functional system put forth herein, future work will continue along two main paths in tandem: 1) validity and skill transference studies and 2) the development of standalone mass-spring framework that will be released as an asset on the Unity asset store for public use. Similar to work conducted previously in the lab for an audiological probe tube simulator (Koch *et al*, 2018), face and content validity will be assessed with feedback from additional otolaryngologists and medical students. After incorporating the feedback into the simulator, skills transference can proceed and the efficacy of the simulator can be quantitatively assessed.

4. Conclusion

The current work presented a fully implemented virtual-reality simulator of an electrode insertion into human cochleae. The simulator allowed users to repeatedly practice electrode insertions in a safe, controlled environment, with costs that did not increase with usage. The simulator further allowed for the real-time visualization of electrode behaviour during insertion and provided detailed performance metrics as well as the ability to replay video of previous attempts. Continued use of widely available and supported game engines in medical simulation will hopefully result in lowered costs of VR simulators. Preliminary feedback of the simulator from relevant professionals has been positive and future work will focus on evaluating its face, content and construct validity.

Funding Statement

Funding for this work was provided through a Collaborative Health Research Projects grant from the Canadian Institutes of Health Research and from the Natural Sciences and Engineering Research Council of Canada.

References

Allard, J., Cotin, S., Faure, F., Bensoussan, P., Poyer, F., Duriez, C., Delingette, H., & Grisoni, L. (2007). SOFA - an Open Source Framework for Medical Simulation. Studies in health technology and informatics, 125, 13-8.

Basdogan, C., Ho, C. H., & Srinivasan, M. A. (2001). Virtual environments for medical training: Graphical and haptic simulation of laparoscopic common bile duct exploration. *IEEE/ASME Transactions on Mechatronics*, *6*(3), 269–285. https://doi.org/10.1109/3516.951365

Coles, T. R., Meglan, D., & John, N. W. (2011). The role of haptics in medical training simulators: A survey of the state of the art. *IEEE Transactions on Haptics*, *4*(1), 51–66. https://doi.org/10.1109/TOH.2010.19

Connell, B. P. O., Hunter, J. B., & Wanna, G. B. (2016). The Importance of Electrode Location in Cochlear Implantation, (December), 169–174. https://doi.org/10.1002/lio2.42

Jeff Craighead, Jenny Burke, Robin Murphy. "Using the Unity Game Engine to Develop SARGE: A Case Study". Proceedings of the 2008 Simulation Workshop at the International Conference on Intelligent Robots and Systems (IROS 2008). September 2008.

Cristina F., Dapoto S., Thomas P., Pesado P. (2018) Performance Evaluation of a 3D Engine for Mobile Devices. In: De Giusti A. (eds) Computer Science – CACIC 2017. CACIC 2017. Communications in Computer and Information Science, vol 790. Springer, Cham. https://doi.org/10.1007/978-3-319-75214-3_15

Elfarnawany, M., Alam, S. R., Rohani, S. A., Zhu, N., Agrawal, S. K., & Ladak, H. M. (2017). Micro-CT versus synchrotron radiation phase contrast imaging of human cochlea. *Journal* *of Microscopy*, *265*(3), 349–357. https://doi.org/10.1111/jmi.12507

Escobar-Castillejos, D., Noguez, J., Neri, L., Magana, A., & Benes, B. (2016). A Review of Simulators with Haptic Devices for Medical Training. *Journal of Medical Systems*, *40*(4), 1–22. https://doi.org/10.1007/s10916-016-0459-8

Fedorov, A., Beichel, R., Kalphaty-Cramer, J., Finet, J., Fillion-Robbin, J.-C., Pujol, S., … Kikinis, R. (2012). 3D Slicer as an image computing platform for the quantitative imaging network. *Magnetic Resonance Imaging*, *30*(9), 1323–1341. https://doi.org/10.1016/j.mri.2012.05.001

Finley, C. C., Holden T.A., Holden L. K., Whiting B. R., Chole R. A., Neely G. J., Hullar T. E., Skinner, M. W. (2009). Role of electrode placement as a contributor to variability in cochlear implant outcomes. *Otology and Neurotology*, *29*(7), 920–928. https://doi.org/10.1097/MAO.0b013e318184f492

Grand View Research. (2017). *Cochlear Implants Market Analysis By Type of Fitting (Unilateral Implantation, Bilateral Implantation), By End-Users (Adults, Pediatrics), By Region (North America, Europe, APAC, Latin America, MEA), & Segment Forecasts, 2018 - 2025*. Retrieved from https://www.grandviewresearch.com/industry-analysis/cochlear-implants-industry

Huang, C., Agrawal, S. K., & Ladak, H. M. (2016). Virtual Reality Simulator for Training in Myringotomy with Tube Placement. *Journal of Medical and Biological Engineering*, *36*(2), 214–225. https://doi.org/10.1007/s40846-016-0124-1

Koch RW, Moodie S, Folkeard P, Scollie S, Janeteas C, Agrawal SK, Ladak HM. (2018). Face and Content Validity of a Probe Tube Placement Training Simulator. J Am Acad Audiol. DOI:10.3766/jaaa.17114. PMID: 30461412.

Kotsis, S. V., & Chung, K. C. (2013). Application of the “see one, do one, teach one” concept in surgical training. *Plastic and Reconstructive Surgery*, *131*(5), 1194–1201. https://doi.org/10.1097/PRS.0b013e318287a0b3

Le Chénéchal, M., Chatel-Goldman, J. (2018) HTC Vive Pro time performance benchmark for scientific research. ICAT-EGVE 2018, Limassol, Cyprus.

Luboz, V., Blazewski, R., Gould, D., & Bello, F. (2009). Real-time guidewire simulation in complex vascular models. *Visual Computer*, *25*(9), 827–834. https://doi.org/10.1007/s00371-009-0312-x

Marks, S., Windsor, J., & Wünsche, B. 2007. Evaluation of game engines for simulated surgical training. In Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia (GRAPHITE '07). ACM, New York, NY, USA, 273-280. DOI: https://doi.org/10.1145/1321261.1321311

F. Messaoudi, G. Simon and A. Ksentini, "Dissecting games engines: The case of Unity3D," 2015 International Workshop on Network and Systems Support for Games (NetGames), Zagreb, 2015, pp. 1-6. DOI: 10.1109/NetGames.2015.7382990

Nagendran M, Gurusamy KS, Aggarwal R, Loizidou M, Davidson BR. Virtual reality training for surgical trainees in laparoscopic surgery. Cochrane Database of Systematic Reviews 2013, Issue 8. Art. No.: CD006575. https://doi.org/10.1002/14651858.CD006575.pub3.

NIDCD. (2017). Cochlear Implants. Retrieved May 28, 2018, from the National Institute on

Deafness and Other Communication Disorders: https://www.nidcd.nih.gov/health/cochlear-implants

Prasad, M. S. R., Manivannan, M., Manoharan, G., & Chandramohan, S. M. (2016). Objective Assessment of Laparoscopic Force and Psychomotor Skills in a Novel Virtual Reality-Based Haptic Simulator. *Journal of Surgical Education*, *73*(5), 858–869. https://doi.org/10.1016/j.jsurg.2016.04.009

Secin, F. P., Savage, C., Abbou, C., de La Taille, A., Salomon, L., Rassweiler, J., Hruza, M., Rozet, F., Cathelineau, X., Janetschek, G., Nassar, F., Turk, I., Vanni, A. J., Gill, I. S., Koenig, P., Kaouk, J. H., Martinez Pineiro, L., Pansadoro, V., Emiliozzi, P., Bjartell, A., Jiborn, T., Eden, C., Richards, A. J., Van Velthoven, R., Stolzenburg, J. U., Rabenalt, R., Su, L. M., Pavlovich, C. P., Levinson, A. W., Touijer, K. A., Vickers, A., … Guillonneau, B. (2010). The learning curve for laparoscopic radical prostatectomy: an international multicenter study. The Journal of urology, 184(6), 2291-6.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 591–611.

Stephens, M. A. (1974). EDF Statistics for Goodness of Fit and Some Comparisons. *Journal of the American Statistical Association*, *69*, 730–737.

Unity3D. (2015). Unity 5.0 Release Notes. Retrieved May 28, 2018, from https://unity3d.com/unity/whats-new/unity-5.0

Unity3D. (2018). Unity - Manual: GPU instancing. [online] Available at:

https://docs.unity3d.com/Manual/GPUInstancing.html

Wikimedia Commons. (2008). Cochlea Implant. Retrieved May 29, 2018, from https://commons.wikimedia.org/wiki/File:Cochleaimplantat.jpg