Fully Immersive Virtual Reality for Skull-base Surgery: Surgical Training and Beyond

Adnan Munawar[1*], Zhaoshuo Li[1], Nimesh Nagururu[2], Danielle Trakimas[2], Peter Kazanzides[1], Russell H. Taylor[1] and Francis X. Creighton[2]

[1]Johns Hopkins University, Baltimore, 21218, MD, USA.
[2]Johns Hopkins Medicine, Baltimore, 21209, MD, USA.

*Corresponding author(s). E-mail(s): amunawa2@jh.edu;

Abstract

Purpose: A fully immersive virtual reality system (FIVRS), where surgeons can practice procedures on virtual anatomies, is a scalable and cost-effective alternative to cadaveric training. The fully digitized virtual surgeries can also be used to assess the surgeon skills automatically using metrics that are otherwise hard to collect in reality. Thus, we present FIVRS, a virtual reality (VR) system designed for skull-base surgery, which combines high-fidelity surgical simulation software with a realistic hardware setup. **Methods:** FIVRS integrates software and hardware features to allow surgeons to use normal clinical workflows for VR. FIVRS uses advanced rendering designs and drilling algorithms for realistic surgery. We also design a head mounted display with ergonomics similar to that of surgical microscopes. A plethora of digitized data of VR surgery are recorded, including eye gaze, motion, force and video of the surgery for post analysis. A user-friendly interface is also designed to ease the learning curve of using FIVRS. **Results:** We present results from a user-study involving surgeons to showcase the efficacy FIVRS and its generated data. **Conclusion:** We present FIVRS, a fully immersive VR system for skull-base surgery. FIVRS features a realistic software simulation coupled with modern hardware for improved realism. The system is completely open-source and provides feature rich data in an industry standard format.

Keywords: Virtual Reality, Skull-base Surgery, Surgical Training, Dataset Generation

1 Introduction

Skull-base surgery requires surgeons to operate within the skull to remove bone and soft-tissue for various procedures. These procedures are among the most complicated surgical interventions [1] as operating within the skull requires safely navigating around sensitive structures including nerves and vessels that are often concealed by operable tissue sub-millimeter distances away. Furthermore, there can be significant anatomic variation between patients [2]. This requires high expertise by the surgeon, gained through training and practice, to avoid sensitive anatomies while still removing adequate operable tissue. Otolaryngology residents (who practice skull-base surgery) therefore undergo extensive training on human cadaver heads and anatomical phantoms as part of their training. However, feedback during this training is limited to metrics such as Objective Structured Assessment of Technical Skills (OSATS) which are subjective and limited in specificity [3].

While cadaver heads offer the most realistic training setup, they are consumable (once drilled) and difficult to obtain. Anatomical phantoms are a great alternative but are expensive and also consumable. Computer simulations on the other hand may offer the best alternative in a variety of ways [4]. First, the simulated anatomies can be rapidly modeled based on actual patient scans. Second, they are non-consumable and re-playable, and thus inexpensive (excluding the initial cost of the simulation setup). Third, these simulations can potentially generate invaluable ground truth data. This data can be used to ascertain 1) surgical competency, and 2) downstream computer vision algorithms and for training and validating artificial intelligence (AI) for various tasks.

Developing such a simulation, or a“simulation system”, on the other hand requires a great deal of work so that it can emulate a real surgical scenario as closely as possible. We use the word “simulation system” to emphasize that this system is more than a computer simulation as it involves the amalgamation of possibly several hardware components alongside software interfaces. In this work, we have developed a Fully Immersive Virtual Reality System (FIVRS) that consists of an immersive volumetric drilling simulator which can load actual patient Computer Tomography (CT) scans, a haptic device to control a virtual drill, a VR headset for an immersive stereoscopic view, foot pedal interfaces to power the virtual drill and change drill types, pupil detection and tracking hardware for gaze mapping and an extensive data processing pipeline for recording and analyzing real-time and offline data for surgical skill assessment. The entire system is based on open-source software and data formats that we have released to the community. To demonstrate the utility, performance, and efficacy of our system, we have conducted a pilot user-study involving 3 attendants and 4 residents who performed mastoidectomy, a procedure in skull-base surgery. We present the preliminary quantitative results of these studies. We also present qualitative results regarding the utility of the “simulation system”.

2 Related Work

There has been considerable prior work in the area of computer simulations for skull-base surgery. Early implementations of temporal bone dissection focus mainly on providing a simulation platform for training purposes [5–8]. A non-patient specific skull drilling simulator with multi-point haptic rendering was presented in [9], increasing the simulation realism. In addition to 6 degrees of freedom (DOF) haptic rendering, another notable aspect of this work was the application of local marching cubes algorithm for real-time updates and better visual fidelity. Further work by [10] and [11] greatly improved the visual fidelity of the anatomy and drilling operation. Patient-specific simulation, which enables customized simulation and pre-operative planning was presented in [12]. Recently, the OpenEar dataset provides cone-beam CT scans with micro-slicing to increase the resolution of simulated volumes [13] for finer details. A commercial simulator called Voxel-man (Voxel-Man, Martinistr. 52, Hamburg, Germany) simulates various types of surgeries, including skull-base surgery, and it has found some traction in teaching hospitals.

The mentioned prior works focus on several different aspects for improving the simulation realism, to modeling patient specific anatomy. On the other hand, FIVRS is a simulation system combining software *and* hardware components for immersive training. Furthermore, FIVRS generates and records extensive data in industry standard formats for quantitative skills assessment and incorporating AI to push the boundaries of skull-base surgery.

3 Methods

As discussed in the Sec. 1, the “simulation system” consists of several different hardware and software components which are presented in Fig. 1. The goal of selecting each component is to provide a realistic system to the operators, both in terms of interface (Sec. 3.1), and visual fidelity (Sec. 3.2) and to allow the generation and recording of synthetic data for downstream applications (Sec. 3.3).

We developed FIVRS on top of the Asynchronous Multi-Body Framework (AMBF) [14, 15]. AMBF allows for the real-time simulation of complex robots and environments, both surgical and non-surgical, and provides interfaces for integrating multiple haptic devices alongside the physics simulation. All scene objects, dynamic objects and haptic devices are defined in a front-end description format, called the AMBF Description Format (ADF). We also restructured and upgraded AMBF alongside FIVRS to create the necessary features required for its function. The upgrades include a new multi-pass rendering pipeline, computation and publishing of scene depth and video data in standardized Robot Operating System (ROS) formats, integration of volumetric model loading, drilling and rendering, and modular integration with several hardware and software components. The hardware components include virtual reality head mounted displays, footpedal interfaces, pupil trackers and 6 degrees of freedom (DOF) mice as shown in Fig. 1.

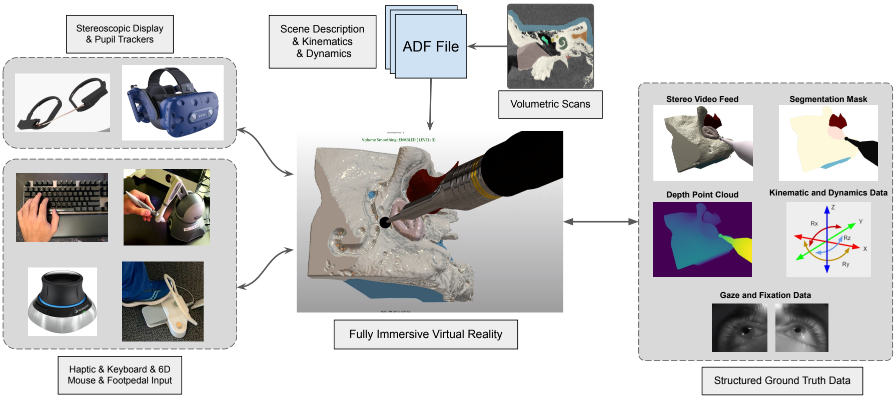


Fig. 1 System architecture: FIVRS comprises of several software and hardware components aimed at providing a realistic user interface, visual fidelity and extensive data generation and recording features. The simulated drill, camera and anatomy can be controlled via a haptic device, keyboard, 6D mouse or collaboratively. The ground truth data is generated and streamed asynchronously using Robot Operating System (ROS) middleware

FIVRS is built upon AMBF's hierarchical plugin pipeline and includes multiple plugin types of different hierarchies. The core of FIVRS is implemented as a simulator plugin, while some optional components such as the virtual reality head mounted display are implemented as object plugins. Details of various plugin types can be found in [15].

3.1 User Interaction and Interfaces

3.1.1 Head Mounted Display

The surgical field in skull-base surgeries is traditionally viewed via a high resolution stereo microscope. We modeled our system to be similar to this setup by using a custom virtual reality (VR) headset on an off-the-shelf mount with custom 3D parts for interconnection. FIVRS is currently only compatible with the Linux operating system and thus we conducted a survey to pick a compatible and suitable VR headset which was the HTC Vive Pro. After researching the state of existing libraries for interfacing VR headsets on Linux, we decided to create our own rendering and "presentation" pipeline based purely on OpenGL and Graphics Library Framework (GLFW).

The camera view in the simulation can be changed by moving the haptic device (while pressing a button on the handle) or the 6 DOF mouse as shown in Fig. 1. In the future, we plan on incorporating external sensors for moving the simulation view by tracking the HMD pose.

3.1.2 Modeling Drill Types and Bone Removal

Various drill types are used for mastoidectomy procedures with variations in burr types and sizes. The burr types include the cutting and diamond tips, and the common burr radii are 1, 2, 4 and 6 mm. The bone removal rate (BRR) consequently varies between the two drill burr types and burr radii. Also, the bit shaft length is typically adjusted to be much longer for smaller burr sizes. To emulate this variation, we modelled the 4 drill types (1, 2, 4, 6 mm) in FIVRS and the BRR for each virtual drill is assigned from surgeon feedback. The feedback was based on trial and error from varying the removal rate for each burr. Further work is required to quantitatively calculate and emulate BRR based on real tissue.

3.1.3 Modeling Auditory Feedback

In our previous work, we implemented haptic feedback to emulate the interaction of the drill with the anatomy. Rather than just modeling the interaction of the drill tip/burr, we also implemented shaft collision detection and force feedback. In our current work, we have incorporated audio feedback in addition to the haptic feedback to provide more realism. An audio signal that mimics a drill's sound is played when the drill is powered on using the footpedal interface and the cutting force alters the signal's pitch according to Eq. 1.

$$p = A_{audio} - \|\vec{F}_{collision}\| / \vec{F}_{max} \tag{1}$$

In the equation above, A_{audio} is the custom maximum audio amplitude, $F_{collision}$ is the force generated from the collision detection algorithm as described in [15] and F_{max} is maximum force threshold for the haptic device. Powering on the drill also adds a vibratory feedback to the haptic device according to Eq. 2, where $\vec{1}$ denotes a 3D vector of ones, A_{drill} is the custom maximum amplitude, f is the custom frequency, and t is the system time. Furthermore, an informative text overlay has been added to inform / warn the operator in case the drill interacts with the sensitive/forbidden anatomies.

$$\vec{F}_{haptic} = \vec{F}_{collision} + \vec{1} * A_{drill} * sin(f * t) \tag{2}$$

3.1.4 Graphical User Interface

In our previous work, the drilling simulator had to be instantiated using the Linux command line which required some familiarity with BASH. While this is not an issue for research purposes, we had to deploy the system with unattended access for attending physicians and residents. Thus we developed an intuitive Graphical User Interface (GUI) that supports the multitude of options, as shown in Fig. 5. A notable aspect of this GUI is that it instantiates as a standalone process and can execute, monitor and control the drilling simulator, pupil tracking software and data recorder as separate processes.

3.2 Visual Fidelity

One of the more challenging aspects of creating a computer simulation that is attractive to attendants and residents is its visual fidelity. While the visual

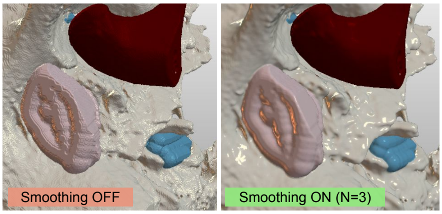


Fig. 2 Comparison between online volume smoothing turned on and off

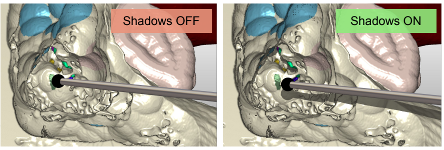


Fig. 3 Comparison between shadowing, to improve depth perception, turned on and off

output from our previous work [15] was acceptable, the users (surgeons and residents) suggested specific shortcomings which included 1) unnatural artifacts/edges on the anatomy, 2) difficulty in perceiving depth and 3) mediocre illumination (and shading). FIVRS and the infrastructure behind it is based on OpenGL and thus we implemented the improvements as OpenGL shaders. While more work needs to be done, the users were satisfied by the updates. Each implementation for improving the visual fidelity is discussed below.

3.2.1 Online Smoothing of the Iso-surface

The volumetric scans that we used for FIVRS comprise of the middle + inner ear region and are roughly $100^3 mm$ in dimensions with an approximate resolution of 500^3 points (voxels). This resolution, while suitable for discerning and segmenting the smaller anatomies within, is less than ideal for rendering without smoothening the surface as shown in Fig. 2. For better rendering quality, many software construct a static geometric mesh from the scan and apply surface smoothening.

Computing the surface mesh for the volumetric data (using algorithms such as marching cubes) is usually performed offline as it is a moderately expensive operation. This is not a problem for a constant grid of data and the advantage of creating a geometric mesh is that the resulting mesh can be further processed, i.e., surface smoothening. However in our case, the volumetric data is modifiable via simulated drilling operations and thus recreating the surface mesh is not ideal. Instead, we use a volumetric ray casting approach implemented on the GPU and summarized in Alg. 1.

The resulting normal $\vec{\eta}$ from the above algorithm is then used for lighting calculations and this simple approach produces noticeably improved visualization of the anatomy. Depending upon the sample threshold N, the rendering frame rate (FPS) is reduced. In our experience, running FIVRS on a computer with AMD Ryzen 5 3600, 32 GB DDR4 RAM, and a NVidia GTX 1080, the FPS dropped from intended 120 Hz to 102 Hz for $N = 3$. Please see the submitted media for a better comparison between the visual outputs.

3.2.2 Shadow Mapping

We implemented shadow mapping (both external object and self-induced) to the rendered volume to improve depth perception (Fig. 3). The improvement is especially noticeable when viewed stereoscopically (i.e., via HMD). Shadow

Algorithm 1 Online Volume Smoothing

```
$\delta\vec{p} = \vec{1} * (N-1)/2.0$                    ▷ $N$ := Sample Threshold $\in \mathbb{R}$
$\vec{\phi} = \vec{1}/[c_x, c_y, c_z]'$                  ▷ $c_x, c_y, c_z$ := Number of Voxels along X, Y and Z
$\vec{p}_{iso} := Raycast(\vec{v})$                      ▷ $\vec{v}$ := View Ray. $\vec{p}_{iso}$ refined by interval bisection
$\vec{\eta} = \vec{0}$                                   ▷ Smoothed Normal
for $x < N$ do
    for $y < N$ do
        for $z < N$ do
            $\vec{p}_{offset} = [x, y, z]'$
            $\vec{p}_{sample} = \vec{p}_{iso} + (\vec{p}_{offset} - \delta\vec{p}) * \vec{\phi}$
            $\vec{\eta} = \vec{\eta} + \nabla(\vec{p}_{sample})$      ▷ $\nabla$ := Gradient from Central Difference
        end for
    end for
end for
$\vec{\eta} = normalize(\vec{\eta})$
```

mapping on volumetric data rendered via ray-casting shares the same principles as geometric meshes, however the implementation is a bit different as the volume is not rendered using primitives (triangles or quads).

3.2.3 Lightmaps

Physically based rendering (PBR) is the state-of-the-art technique for achieving visual realism in computer simulations. This technique uses the reflectance equation instead of the simpler Blinn-Phong shading model to compute the fragment/pixel intensity (color). The reflectance equation produces spectacular visualizations, however, it is considerably more computationally expensive. While our ultimate goal is to implement PBR for AMBF and FIVRS, we were able to improve the visual fidelity by incorporating a computationally and algorithmically much simpler lighting model based on pre-baked lightmaps.

3.3 Data Generation and Management

3.3.1 Importing Anatomical Scans

Volumetric data from CT and MRI scanners is often output in the Digital Imaging and Communication (DICOM) format. In prior work, we collected a large data-set of patient CT scans and manually segmented the sub-cranial anatomies using 3D Slicer [16]. Manual segmentation is an effort intensive task and automated techniques such as [17] can be potentially beneficial. The resulting scans were then saved as Segmented Nearly Raw Raster Data (Segmented NRRD ".seg.nrrd") which is a superset of the "NRRD" format used by 3D Slicer. FIVRS can render either intensity based volumetric data (NRRD), or constant intensity pre-segmented volumes. The latter format is extracted from the 3D Slicer specific "seg.nrrd" format in which each segment is either one hot encoded into a separate grid layer or labeled incrementally (1,2,3 ... for different segments). We created a program that converts the data in ".seg.nrrd"

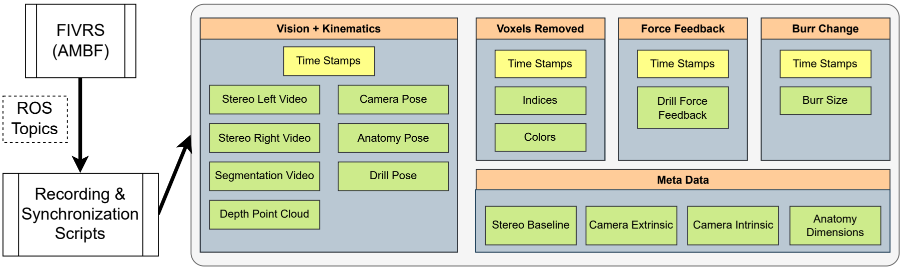


Fig. 4 Asynchronous data from FIVRS is streamed via ROS topics. External Python based scripts synchronize and record this data into “groups” of an HDF5 file. The “voxels removed” group contains the 3D indices of the removed voxels, time of removal and their associated labels (colors). The “force feedback” is the force applied to the haptics device, “burr change” is the change of burr sizes and the time stamps. The “vision + kinematics” and “meta data” are self-explanatory

to an array of images in either JPEG or PNG format as these raw images are easier to be incorporated into FIVRS. The spatial data from the “.seg.nrrd” files is encoded separately using ADF files.

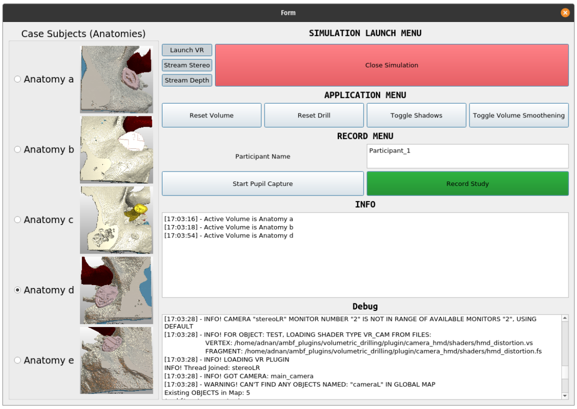
3.3.2 Ground Truth Vision Data

For FIVRS we render 4 simulated cameras and each of these cameras requires multi-pass rendering. The four cameras include a single frame left-right stereo for viewing on the HMD, separate left and right stereo cameras with different baseline (for recording and evaluation purposes) and a segmentation camera. For the single stereo HMD camera, multiple rendering passes include rendering for the left and right halves of the viewport, applying the inverse geometric and chromatic distortion filters, and finally rendering to the screen. For the segmentation camera, the rendering passes include computing the segmentation masks and then separately the depth point cloud. In addition to multi-pass rendering, three of these cameras (other than the single stereo camera) stream their video, and the depth point cloud in the case of the segmentation camera, over ROS.

We found that altogether this became computationally quite expensive and resulted in reduced frame-rates. In this work, we have optimized several components of AMBF’s rendering and depth computation pipeline to alleviate the computational load. Firstly, we implemented an optional presentation free mode for each camera, which allows any camera to only stream its video and not present/render to the screen. Secondly, we reorganized the depth computation algorithm from our previous work [15] to take place entirely on the GPU by leveraging floating point textures. These optimizations increase the overall frame-rate by 15 fps.

3.3.3 Gaze Mapping

Pupil tracking and gaze mapping can demonstrate what specific structures surgeons focus on. This data may further predict the operator’s workload.



○ Anatomy a
○ Anatomy b
○ Anatomy c
● Anatomy d
○ Anatomy e

Participant Name: Participant_1

```
[17:03:16] - Active Volume is Anatomy a
[17:03:18] - Active Volume is Anatomy b
[17:03:54] - Active Volume is Anatomy d
```

```
[17:03:28] - INFO! CAMERA "stereoLR" MONITOR NUMBER "2" IS NOT IN RANGE OF AVAILABLE MONITORS "2", USING
DEFAULT
[17:03:28] - INFO! FOR OBJECT: TEST, LOADING SHADER TYPE VR_CAM FROM FILES:
             VERTEX: /home/adnan/ambf_plugins/volumetric_drilling/plugin/camera_hmd/shaders/hmd_distortion.vs
             FRAGMENT: /home/adnan/ambf_plugins/volumetric_drilling/plugin/camera_hmd/shaders/hmd_distortion.fs
[17:03:28] - INFO! LOADING VR PLUGIN
INFO! Thread Joined: stereoLR
[17:03:28] - INFO! GOT CAMERA: main_camera
[17:03:28] - WARNING! CAN'T FIND ANY OBJECTS NAMED: "cameraL" IN GLOBAL MAP
Existing OBJECTS in Map: 5
```

Fig. 5 A intuitive GUI to launch, debug and test the simulation and its various components.

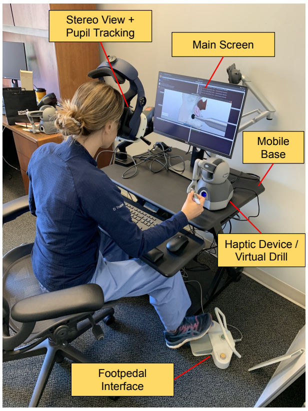


Fig. 6 Labelled hardware setup

To record this data, we incorporate the pupil eye tracking hardware (Pupil Labs GmbH, Sanderstr. 28, 12047, Berlin) into our HMD. The pupil tracking hardware outputs videos for each eye and associated time-stamps which we post-process to compute the operator's gaze, fixation and blinks.

3.3.4 Data Recording Pipeline

One of the notable features of FIVRS is its ability to generate multi-modal ground truth data. This data includes a pair of videos (from stereo cameras), real-time depth point cloud, segmentation mask, and kinematic and dynamics data of scene objects (cameras, drills, anatomy). We experimented with storing this data using ROS Bags due to their convenience. We discovered that ROS bags were not feasible for storing data, as the file size of a 640 × 480 stereo video, segmentation mask and a depth point cloud along with scene kinematics and dynamics data collected at $\approx 10fps$ for $60s$ was $\approx 15GB$. For a single extended mastoidectomy session that may last 20 minutes, the resulting file size of $\approx 300GB$ is almost impractical to manage. We ultimately chose the HDF5 data format as a similarly sized file ($\approx 10fps$ for $60seconds$) had a footprint of $\approx 100MB$. Furthermore for each individual study, we split the recorded data into multiple files based on a user-defined batch size. The pupil-tracking data is stored separately after synchronizing the timestamps with the HDF5 files. The data in each HDF5 file is organized as shown in Fig. 4.

4 Experiments and Results

We conducted a pilot study with 7 participants including 3 attending surgeons (P4, P5, P7), 3 residents and 1 medical student (P1-P3 and P6), who performed a cortical mastoidectomy on a setup shown in Fig. 6. The participant ages ranged from 23-37 years and included 4 males and 3 females. The purpose of the study was to perform an initial validation of the system and the recorded

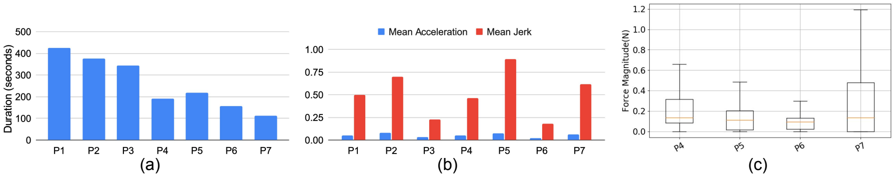


Fig. 7 Results for performing cortical mastoidectomy by participants P1-P7. b) Acceleration and Jerk are reported in Simulation Units. c) Force data for only 4 participants was collected (P4-P7)

data rather than characterizing the performance of the participants. The user-study was approved by our institutional review board under IRB00264318 and was conducted at both the medical campus as well as the engineering campus. Two of the three residents had significant prior experience with the procedure. To account for the learning curve and to establish familiarity with FIVRS, each participant interacted with the system and operated on multiple anatomies for at least 30 minutes. After the training sessions, the study participants operated on different anatomies and the data presented in Fig. 4 along with pupil tracking was recorded.

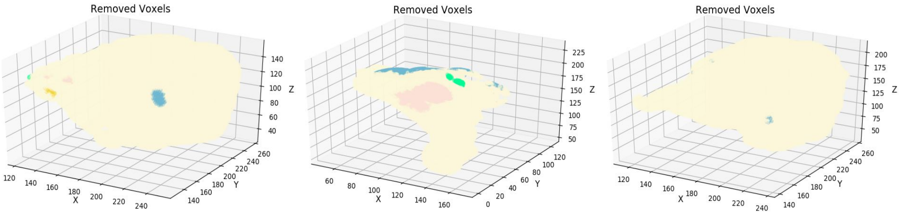
We present some results based on a common anatomy. Because of the small participant count and lack of a concerted effort for diverse demographics, the study is primarily limited to only showcase the utility of the system and the recorded data. Fig. 7 shows the temporal, kinematic and dynamics data for each participant. Fig. 8 shows an interesting metric which is the volume removed. This data will be invaluable for machine learning applications and other tasks such as predicting the volume to be removed for a given anatomy.

5 Conclusion and Future Work

We present FIVRS, a fully immersive VR system for skull-base surgery. In addition to providing a realistic "simulation system" for skull-surgery, FIVRS offers an extensive data generation and recording interface using an industry standard format. FIVRS is light-weight, modular and extensible and the infrastructure behind it is already being used for numerous research applications including simulation of other medical procedures, surgical navigation, and enforcing virtual fixtures for robot control with simulation in the loop.

For the future, our short term goal is to continue collecting data from surgeons to create an extensive data-set. We shall then use this data-set for several applications ranging from developing a quantitative metric for surgical skills evaluation, improving surgical workflow by presenting the predicted anatomy to be removed to the surgeons and integrating with robots (such as the ENT Robot from Galen Robotics, 1100 Wicomico St., Baltimore, MD) for safer cooperative control with simulation based virtual fixtures.

Acknowledgments. This work was supported by NSF OISE-1927354 and OISE-1927275, NIDCD K08 Grant DC019708, a grant from Galen Robotics,



P1 P2 P5

Fig. 8 Visualization of volume (voxels) removed. Colors other than bone (pastel) represent unintended removal of sensitive/critical structures

and an agreement between Johns Hopkins University and the Multi-Scale Medical Robotics Centre, Ltd.

Declarations

Conflict of interest. Russell Taylor and JHU may be entitled to royalty payments related to technology discussed in this paper, and Dr. Taylor has received or may receive some portion of these royalties. Also, Dr. Taylor is a paid consultant to and owns equity in Galen Robotics, Inc. These arrangements have been reviewed and approved by JHU in accordance with its conflict of interest policy.

References

[1] Scholz, M., *et al.*: Skull base approaches in neurosurgery. Head & Neck Oncology **2**, 1–9 (2010)

[2] Cousins, V.: Lateral skull base surgery: a complicated pursuit? The Journal of Laryngology & Otology **122**(3), 221–229 (2008)

[3] Laeeq, K., *et al.*: Pilot testing of an assessment tool for competency in mastoidectomy. Laryngoscope **119**(12), 2402–2410 (2009)

[4] George, A.P., De, R.: Review of temporal bone dissection teaching: how it was, is and will be. The Journal of Laryngology & Otology **124**(2), 119–125 (2010). https://doi.org/10.1017/S0022215109991617

[5] Kuppersmith, R.B., *et al.*: Building a virtual reality temporal bone dissection simulator. In: Medicine Meets Virtual Reality, pp. 180–186 (1997). IOS Press

[6] Wiet, G.J., *et al.*: Virtual temporal bone dissection simulation. In: Medicine Meets Virtual Reality 2000, pp. 378–384 (2000). IOS Press

[7] Agus, M., *et al.*: Mastoidectomy simulation with combined visual and

haptic feedback. In: Medicine Meets Virtual Reality 02/10, pp. 17–23 (2002). IOS Press

[8] Pflesser, B., *et al.*: Volume cutting for virtual petrous bone surgery. Computer Aided Surgery **7**(2), 74–83 (2002)

[9] He, X., Chen, Y.: Bone drilling simulation based on six degree-of-freedom haptic rendering. In: Proceedings of EuroHaptics, pp. 147–152 (2006)

[10] Sorensen, M.S., *et al.*: The visible ear simulator: a public pc application for gpu-accelerated haptic 3d simulation of ear surgery based on the visible ear data. Otology & Neurotology **30**(4), 484–487 (2009)

[11] Wong, D., *et al.*: Comparison of cadaveric and isomorphic virtual haptic simulation in temporal bone training. Journal of Otolaryngology-Head & Neck Surgery **43**(1), 1–6 (2014)

[12] Chan, S., Li, P., Locketz, G., Salisbury, K., Blevins, N.H.: High-fidelity haptic and visual rendering for patient-specific simulation of temporal bone surgery. Computer Assisted Surgery **21**(1), 85–101 (2016)

[13] Sieber, D.M., *et al.*: Openear image data enables case variation in high fidelity virtual reality ear surgery. Otology & Neurotology **42**(8), 1245–1252 (2021)

[14] Munawar, A., *et al.*: A real-time dynamic simulator and an associated front-end representation format for simulating complex robots and environments. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1875–1882 (2019). IEEE

[15] Munawar, A., *et al.*: Virtual reality for synergistic surgical training and data generation. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization **10**(4), 366–374 (2022)

[16] Ding, A.S., *et al.*: Automated registration-based temporal bone computed tomography segmentation for applications in neurotologic surgery. Otolaryngology-Head and Neck Surgery **167**(1), 133–140 (2021)

[17] You, C., *et al.*: Simcvd: Simple contrastive voxel-wise representation distillation for semi-supervised medical image segmentation. IEEE Transactions on Medical Imaging **41**(9), 2228–2237 (2022)