Team13 Final Report

Simulation Assisted Navigation for Skull Based Surgery

EN 601.656 Computer Integrated Surgery II

Team 13: Xinhao Chen, Zhaomeng Zhang

Mentors: Dr. Adnan Munawar, Dr. Manish Sahu, Max Li, Mohammad Salehizadeh, Dr. Peter Kazanzides, Dr. Pete Creighton, Dr. Danielle Trakimas, Dr. Deepa Galaiya, Dr. Russ Taylor

February 2023 to May 2023

Contents

1. Abstract

Skull base surgery is a complex and delicate procedure that requires real-time information to prevent critical regions from being damaged. Surgeons often rely on their experience and the support of endoscopes and microscopes, but the lack of stereo anatomy models and real-time navigation information makes this a challenging operation [1]. During this semester, our team successfully developed a system for augmenting stereo-microscope video with AMBF simulation to aid in skull base surgery. Surgeons are able to view both the AMBF simulation and microscopic view simultaneously through a VR headset, with the ability to change between different display modes and manipulate the view in various ways. While the basic functionality of the system has been accomplished, there are still some aspects that need to be optimized, particularly the latency time, which currently stands at 0.3 seconds. We plan to address this by directly processing and displaying microscopic video on the MockOR computer. Additionally, our team aims to improve the display effect of the smaller window and increase the clarity of the video displayed on the HMD (Head Mount Display), potentially by using a higher-performance device. Despite some remaining work that still needs to be done, we have decided to continue working on this project throughout the summer to ensure that we meet all of our goals and objectives.

2. Introduction

2.1 Background

Skull base surgery usually has delicate and complex surgical procedures, so surgeons hope to know the detailed real-time information of the operation so that the critical regions will not be hurt and damaged. For example, surgical drills intend to remove the bony tissue rather than soft tissue. If the soft tissue is beneath the bony tissue, surgeons need to know how far soft tissue is located from the current position.

In minimally invasive skull base surgery, surgeons normally rely on their experience with the support of endoscope and microscope to do the operation [1]. Due to the lack of stereo anatomy models of patients and real-time detailed navigation information, skull base surgery is considered a difficult operation for many surgeons. In order to solve the challenges in skull base surgery, we plan to provide immersive, detailed, and real-time navigation for surgeons. We need to develop a pipeline for augmenting stereo-microscope video with AMBF simulation for navigation during the operation. While doing the surgery, the surgeons should be able to pause the operation and load the simulated anatomy onto the main view to get an idea of how far they have drilled and what anatomies are nearby. They are also able to change the view of simulated anatomy like rotation, scaling, and slicing. Our project is built upon prior works, like Asynchronous Multi-Body Framework (AMBF) [1] and a fully immersive virtual reality system (FIVRS) [4].

2.2 Clinical Motivation

The skull base surgery is a specialized operation closely related to neurosurgery and otolaryngology, in which surgeons need to access the skull of a patient to treat the diseases [1]. Pituitary tumors, meningiomas, and chordomas are the diseases requiring skull base surgery [2]. In certain types of skull base surgery, the bony tissue must be removed by surgical tools in order to reach the target anatomy. However, it is a difficult and complicated procedure for surgeons for two reasons.

Patients usually have different anatomy structures in some surgical regions. For example, male patients usually have thicker and larger skull bones, but the skulls of females are generally thinner and lighter. And male eye sockets are squarer than females [3]. The sex is only one factor that can make an individual's skull different from others. These distinct skull structures increase the difficulty of doing the operation for surgeons. Therefore, surgeons hope to have a stereo anatomy model of patients as reference during the surgery, which is one motivation for our project.

Moreover, the real-time and detailed navigation information could greatly assist the skull base surgery and lower the risk of operation [1]. During the operation, the surgical drills intend to remove the bony tissue rather than soft tissue, and if the soft tissue is beneath the bony tissue, surgeons are not able to see it, so they need to know how far soft tissue is located from the current position of surgical drill in real-time for making sure sensitive tissue won't be hurt and damaged. To provide concrete and real-time operation navigation information for surgeons is the second motivation for our project.

2.3 Prior Work

A fully immersive virtual reality system (FIVRS) has been developed and designed for practice of skull base surgery. The system works on the fundament of Asynchronous Multi-Body Framework (AMBF) which is “an open-source 3D versatile simulator for robots and provides a real-time dynamic simulation of multi-bodies such as robots, free bodies, and multi-link puzzles.” [5] Based on AMBF, our mentors developed FIVRS in recent years. To run the FIVRS requires a lunch file that consists of a world file, input device file, and model file. These three files are three types of AMBF Description Format (ADF) Files [1]. According to Figure 1, the world file has env model file and world attributes that can be solver attributes, gravity, and dimensions. The env model file is formed by the model file which can be bodies, joints, sensors and actuators, and multiple model files can be loaded in the launch file [1]. The overall model loading flow for FIVRS is described in Figure 1.

Compared with AMBF, the rendering pipeline and AMBF Description Format (ADF) specifications are improved for AMBF+ [1]. Additionally, the modular plugin handling interface in AMBF+ allows users to do custom application development [1].

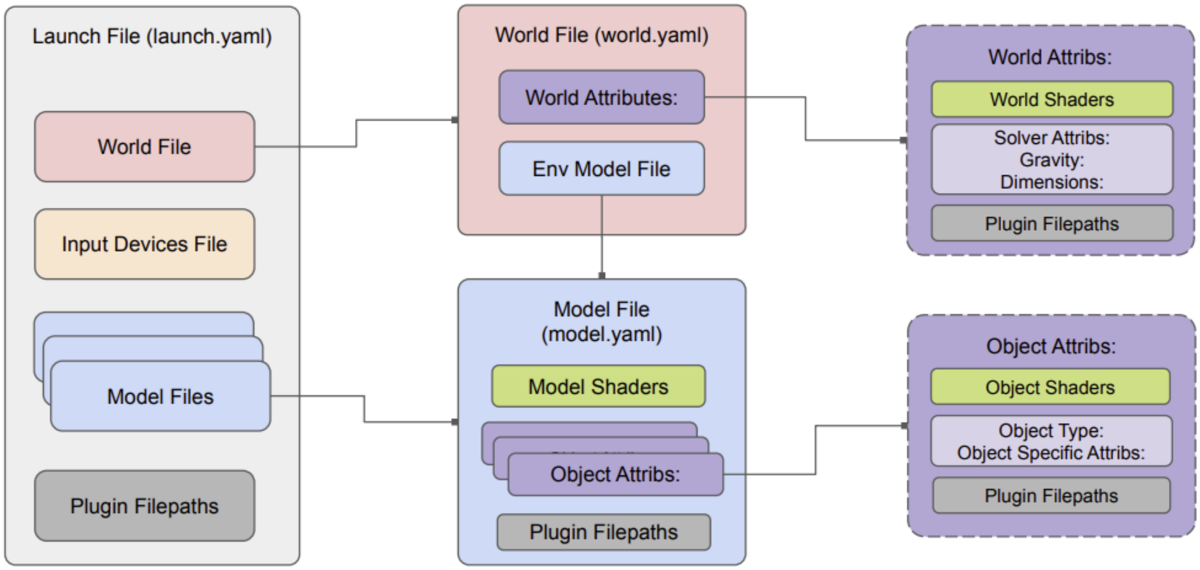


Figure 1 The overall model loading flow for AMBF+ [1]

Fully immersive virtual reality system allows users to load stereo anatomy models from actual Computer Tomography (CT) of patients and accepts various input devices, including stereoscopic display (VR headset), haptics device, keyboard, 6D mouse, and foot pedal (Figure). With these input devices, surgeons can have an immersive stereoscopic view of anatomy and control the virtual drill to do the operation in the simulation environment. In the meanwhile, the system is able to generate valuable data for downstream algorithm development and surgical assessment of surgeons [1].

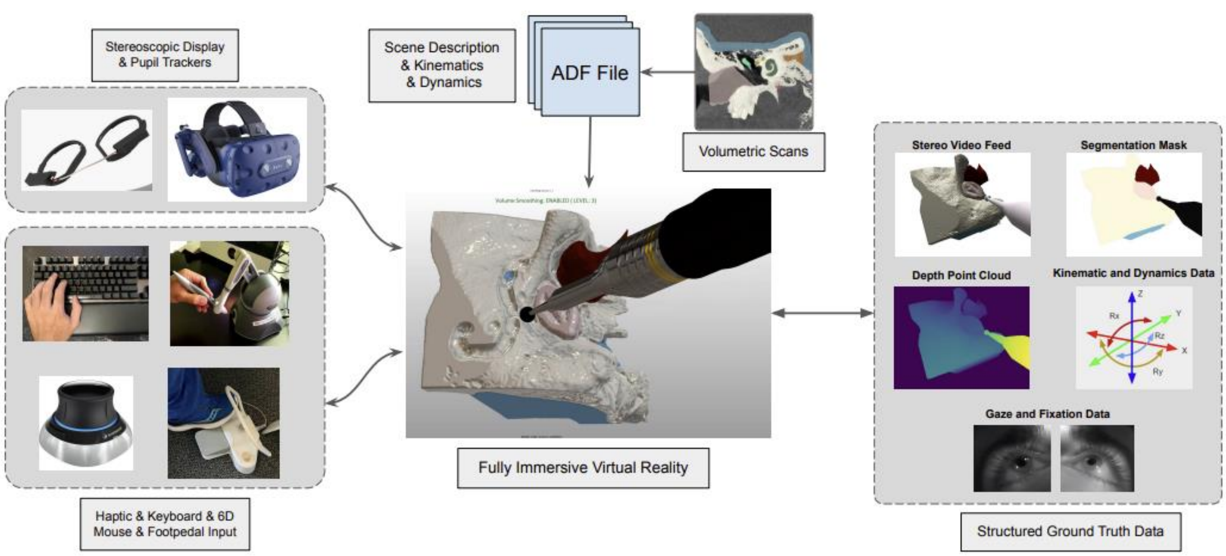


Figure 2 The fully immersive virtual reality system [4]

2.4 Goals

The project motivation and prior work drives us to develop a pipeline for augmenting stereo-microscope video with AMBF simulation for guidance in skull base surgery. In other words, the new system as much as possible to achieve that simulated operation simultaneously corresponds with real skull base surgery and provides guidance modalities to the surgeons to accomplish the surgery. Either real microscopic video or simulated stereoscopic view can be shown on VR headset to have an immersive view for surgeons. During the operation, they can pause the operation and load the simulated anatomy to get real-time and detailed surgical information. The view of simulated anatomy also can be changed, like rotation, scaling, and slicing.

Based on these goals, we have several milestones:

1) Process the stereo microscopic video on the computer.
2) Display the video in HMD while operating.
3) Minimize the latency time of showing video in HMD.
4) Integrate the video in HMD with AMBF.
5) Develop a user interface for the system.
6) Perform anatomy registration.
7) Update from the real operation to the simulation.
8) Conduct user study with surgeons.

The relationship between these milestones is that the former is the foundation for the latter except for the fourth and fifth milestones. As a result, we need to do the fourth and fifth milestones simultaneously and finish other works one by one.

Our deliverables are follows:

Minimum: Develop the pipeline to augment stereo endoscopic video with the AMBF simulation in an HMD.

Deliverables: A environment that can run AMBF codes, an AMBF plugin that can read the stereo video from stereo microscope and an AMBF plugin that can process the stereo microscopic video for HMD.

Expected: Perform anatomy registration and update from the real setup to the simulation. Minimize the latency time of the entire system. Design a user interface for the system.

Deliverables: A system with small latency time, a system can display both simulation and anatomy videos in HMD, a user interface for switch and control two videos and an updated system with registration.

Maximum: Conduct user study with surgeons, update the system according to the feedback and present the system and findings in a paper.

Deliverables: User study report, a system can be applied in skull-base surgery and present the system and findings in a paper.

2.5 Procedure Overview

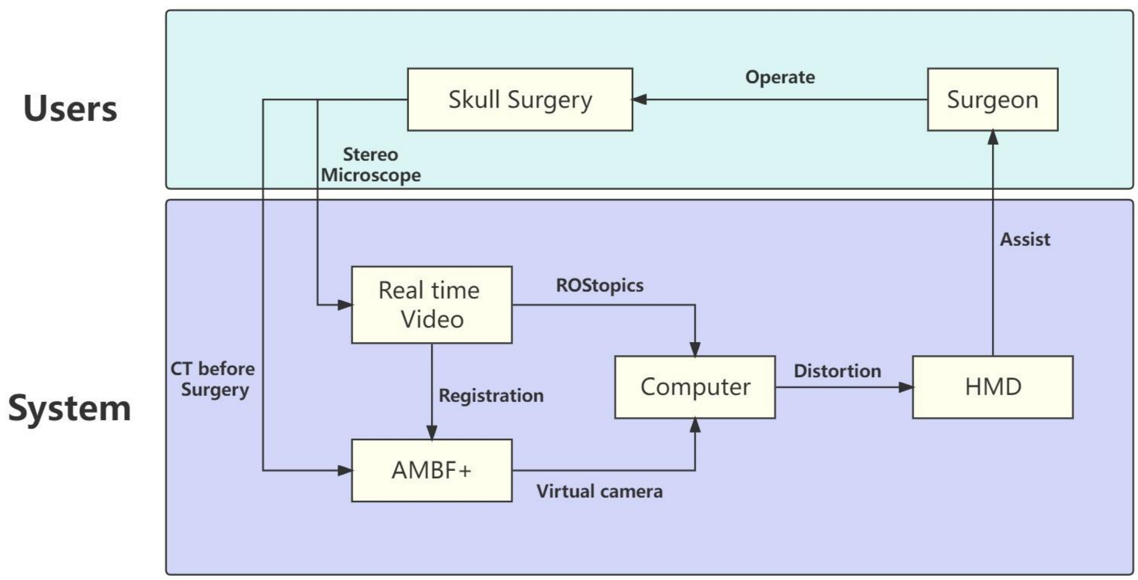


Figure 3 Workflow of the system

Figure 3 shows the workflow of our system. It can be divided into two parts, the users' part and the system part. As we mentioned in the motivation, skull base surgery is hard. The aim of our system is to assist surgeons by showing everything more clearly.

We have a stereo microscope to produce real time video from surgery. We also have CT scan before surgery to construct the model in AMBF+. We already have ICP (Iterative

Closest Point) registration through optical tracker system between real time video and simulation environments to connect them together.[6] This step enables the drill in AMBF+ to move and cut the same thing as the drill in real surgery. Then, we process both videos and transmit them to HMD. Surgeons can see both real time video and simulation video in HMD. They have many choices for simulation videos. They can see the model from any point in any angle or see slices instead of 3D models.

Our task mainly focuses on the system part. We already have all the hardware and AMBF+ platform. The task we need to do is to connect all parts together and make them a whole system. We will start with the ROS (Robot Operating System) and CV_bridge package to process videos. If we want to further decrease the latency time, we may explore additional protocols and software such as Gstreamer. CV_bridge package fills the gap between getting raw images from a camera driver and higher-level vision processing by OpenCV. It can process raw camera images into useful inputs to vision algorithms. Since AMBF+ is also based on ROS in C++, they can cooperate well.

We will use HTC VIVE Pro HMD to display videos. Built to meet the needs of today’s most demanding VR users, VIVE Pro is an easy-to-deploy VR system that provides the richest feature set for professional users. From seated environments to expansive, multi-user deployments, VIVE Pro delivers the highest fidelity, clearest audio, and most immersive VR experience.

3. Technical Approach

3.1 Reading videos

The system we aim to develop needs two kinds of videos, one from AMBF simulation platform and the other one from real microscope. To make the system work, we should develop two methods to read a group of stereo videos from each source. A group of stereo videos contains a video for the left eye and a video for the right eye.

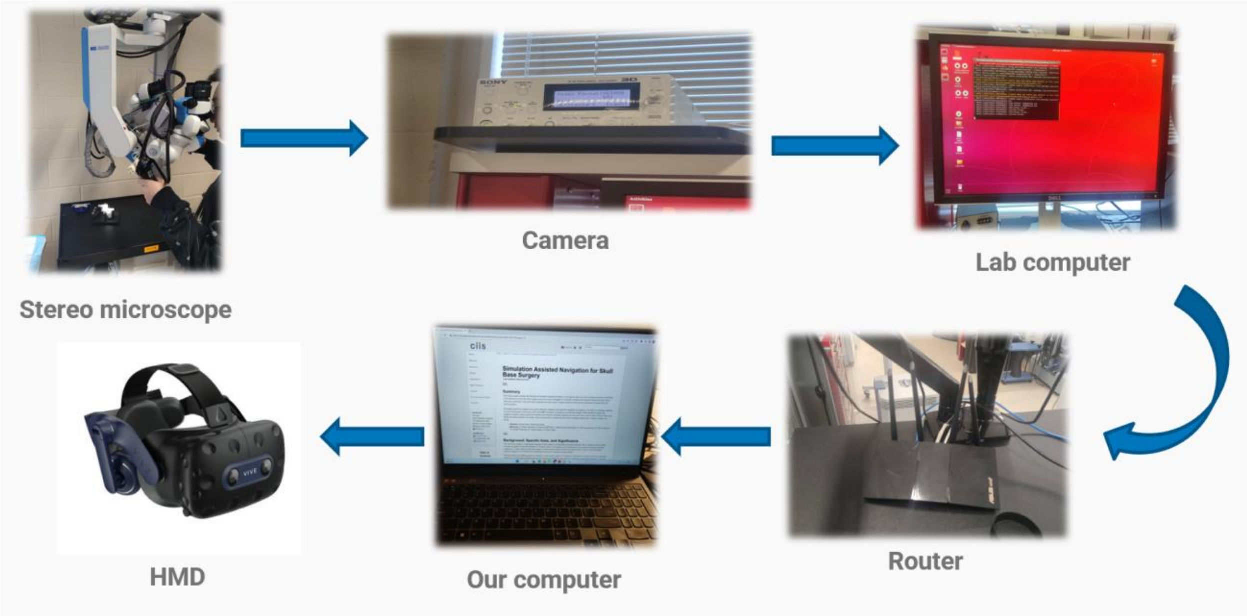


Figure 4 Hardware pipeline for reading videos

Figure 4 shows the hardware pipeline for reading videos. We first adjust stereo microscope to see models clearly. Then, the camera will record videos from microscope and transmit it to the lab computer. The lab computer is connected to a router, and we also connect our computer to the router through an internet cable. We do not use WIFI connection because the bandwidth may not be big enough for high resolution videos. Till now, videos from the real microscope are transmitted to our computer.

Meanwhile, the AMBF simulation platform is running on our computer to generate another group of videos. We process both groups and finally display them on HMD.

Figure 5 Operating Haag-Streit microscope

Figure 5 shows the stereo microscope. The type of it is the Haag-Streit microscope. Haag-Streit delivers breakthrough versatility and precision in microscope technology for ophthalmology. With over 140 years of experience in surgical tool manufacturing, they set the standard for optics, ergonomics, engineering, and imaging. This surgical microscope provides best-in-class stability and 3D perception.

To use it, we first need to remove the protective cover on it. And then open the green switch on the side. There are many buttons on both sides' handles, which can move the camera, adjust the light and focus. During the usage, we should take care not to touch the lens or they will be polluted and make the video vague. After using it, we need to close the switch and put the protective cover back. It is important to close the switch because the light of the microscope is strong and can produce huge heat after running for a long time.

Figure 6 Sony MCC-3000MT 3D Full HD Medical Video Camera

After adjusting the stereo microscope, we need to use a camera to record videos from it. The camera we use is “Sony MCC-3000MT 3D Full HD Medical Video Camera”, as shown in figure 6. Since everything is already set and connected, we just need to open the camera when using it and close it before we leave. One thing that needs to be noticed is the focal plane of the camera is determined by the microscope but not itself. The camera publishes videos as ROStopics.

The video recorded by Medical Video Camera is transmitted to the lab computer. Since a lot of research projects need that computer, it is not recommended to install packages and do video processes on that computer. A router is used to transmit videos from that computer to our own computer, as shown in figure 7.

Figure 7 Router

We need to connect both computers to the router through internet cables. A special type of internet cable is required to prove the speed of transmission.

It is easier to read videos from AMBF simulation platform than the ones from real microscope because the video is generated in the computer. AMBF simulation platform provides functions for setting camera at any position. We just need to set camera parameters in the file "./ADF/single_stereo_camera.yaml". The videos are published as ROStopics, which is the same as the real camera.

Since both groups of stereo videos are in ROStopics, we can subscribe to what we need to read them.

3.2 Processing videos

The videos we read from real microscope or AMBF simulation platform are designed to display on normal screens but not VR videos for HMD. If we directly display them in HMD, people cannot see anything clearly in it and will feel dizzy. That is why we need to process videos in proper type for displaying.

The method to process normal videos into VR type is giving distortion.[7] There are a total of 6 parameters that have to do with lens correction. First, there is the lens Field of View. It is a parameter that determines image perspective distortion. The actual lens correction parameters a, b and c, which are used to correct for barrel distortion, pincushion distortion and even wavy distortion, as shown in figure 8. The lens shift parameters d and e that correct for the lens optical axis not being in the image center.

(a) (b)

(c)

(d)

Figure 8 Distortion types [7] (a) Raw image (b) Barrel distortion

(c) Pincushion distortion (d) Even wavy distortion

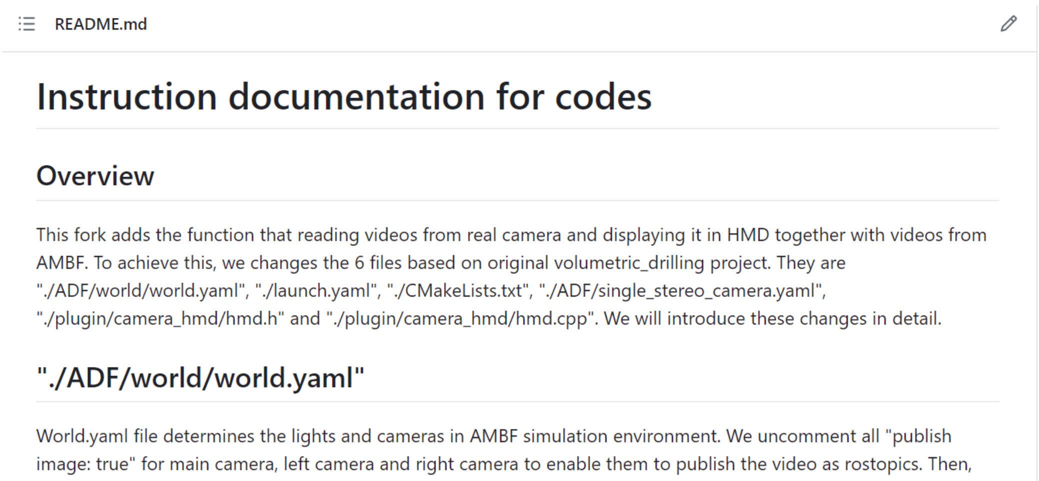
The Focal Length is a physical property of the lens. We can change it by operating the microscope. Together with the effective sensor or film size and the focusing distance it approximates the image Field of View. The Field of View together with the lens projection determines the image perspective distortion.

The lens distortion a, b and c parameters correspond to a third-degree polynomial describing radial lens distortion.

$$r_{src} = (ar_{dest}^3 + br_{dest}^2 + cr_{dest} + d)r_{dest}$$

This equation shows how it works. “rdest” and “rsrc” refer to the normalized radius of an image pixel. The center point of this radius is where the optical axis hits the image. Normalized means here that the largest circle that completely fits into an image is said to have radius equals 1. The fourth parameter d is determined by a, b, and c by d = 1 – a – b - c.

In our program, all parameters are set in file "./plugin/camera_hmd/hmd.cpp" and the distortion is written in file "./plugin/camera_hmd/shaders/hmd_distortion.fs". Detailed information of how we realize it in codes can be found in our documentation GitHub README.md [8].


README.md

Instruction documentation for codes

Overview

This fork adds the function that reading videos from real camera and displaying it in HMD together with videos from AMBF. To achieve this, we changes the 6 files based on original volumetric_drilling project. They are "./ADF/world/world.yaml", "./launch.yaml", "./CMakeLists.txt", "./ADF/single_stereo_camera.yaml", "./plugin/camera_hmd/hmd.h" and "./plugin/camera_hmd/hmd.cpp". We will introduce these changes in detail.

"./ADF/world/world.yaml"

World.yaml file determines the lights and cameras in AMBF simulation environment. We uncomment all "publish image: true" for main camera, left camera and right camera to enable them to publish the video as rostopics. Then,

Figure 9 Documentation in GitHub [8]

3.3 Displaying videos

After processing both groups of stereo videos, we need to display them in head-mounted display (HMD). It is a display device, worn on the head or as part of a helmet, that has a small display optic in front of each eye.

Figure 10 VIVE Pro headset [9]

The HMD we use for this system is the VIVE Pro headset. It has next-level graphics and sound for riveting PC-VR. Its purposeful and pragmatic ergonomics delivers smooth and comfortable immersion.

When using it, we need to connect it to our computer through two cables, one USB and one USB-PD. The HMD also needs an individual power source. It works as a second screen for the computer. Detailed information of all hardware we used for this system can be found in our documentation Hardware Setup Manual [9].


LABORATORY FOR
Computational
Sensing + Robotics

SIMULATION ASSISTED NAVIGATION
SYSTEM HARDWARE MANUAL

VERSION 1.0

EDITOR: XINHAO CHEN, ZHAOMENG ZHANG
TIME: 4/10/2023

Figure 11 Hardware Setup Manual [9]

To display processed video in the HMD, we should set it display in the second screen in full-screen mode. It can be set in file "./ADF/single_stereo_camera.yaml". There will be a normal video showing AMBF simulation platform in computer screen while VR views in HMD.

3.4 Registration

The registration method is mainly following Twin-S system.[6] It is based on optical tracker system.

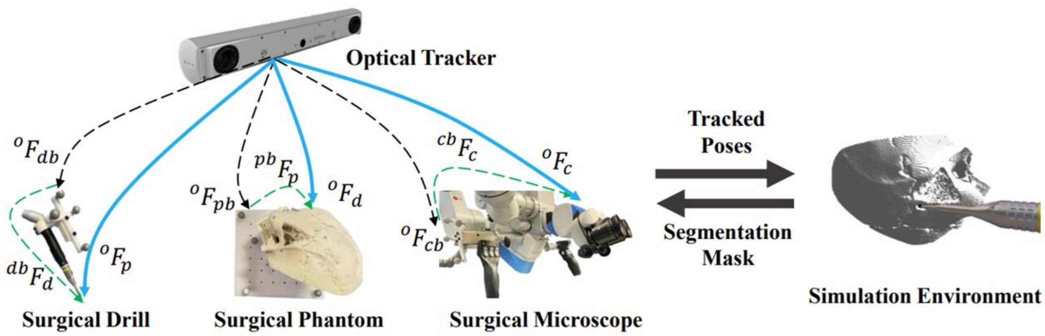


Figure 12 Overview of Twin-S for skull-base surgery [6]

Building a digital twin system for skull-base surgery requires precise modeling, tracking, and updating of the patient's anatomy, the surgeon's tool(s), and the surgical camera. In the paper, the system includes a surgical phantom to simulate the patient's anatomy, a surgical drill as the ablation tool, and a stereo surgical microscope.

For each object, there are 2 frames. They are the base coordinate and the local coordinate. The base frame is used to localize the position and orientation of the object while the local coordinate is related to the function of the object. Optical markers are fixed on the base frame.

The base frame of the surgical drill is at the tail of the drill shaft, together with the optical tracking markers. The drill local coordinate frame is on the drill tip center. For the rotation part of the transformation between these frames, there are only 2 degrees of freedom because the drill is symmetrical along its shaft. To find the shaft axis in optical tracker coordinate, they fix the drill on a robot arm end effector, where the drill holder aligns the drill shaft with the Z-axis of the robot to sufficient accuracy. Therefore, by commanding the robot to move along its Z-axis, we recover the direction of motion Q in the optical tracker coordinate.

They obtain the 3D structure of the surgical phantom using a CT scanner. The phantom is modeled as a binary volume of occupancy, where voxels corresponding to the bony tissues are marked as occupied and voxels representing air are marked as free space. To track the surgical phantom, they mount the phantom on a polycarbonate board with optical tracking markers, defining the base coordinate of the phantom. To calibrate the transformation from phantom base to phantom, they directly compute the transformation between the virtual model and the physical phantom via the point-to-plane ICP registration. They sample 380 points on the physical phantom surface using a tracked pointer tool for the purpose of calibration.

When surgeons perform drilling in the real world, the system updates the surgical phantom in real-time. They approximate the drill tip as a sphere. Twin-S detects collisions between the surgical phantom and the drill burr given the tracked positions. The voxels that collide with the drill tip sphere are moved and set to free space to simulate the tissue removal process.

The paper obtains the intrinsic parameters and distortion coefficients of the stereo camera using a ChArUco pattern. Rectification is then performed to obtain a projective camera model. To track the surgical camera, optical tracking markers are mounted on the handle of the camera, defining the base coordinate frame. A hand-eye calibration routine [10] is used to obtain the transformation. Given the tracked camera, Twin-S generates per-pixel segmentation masks and depth maps based on the object information, which can be used for different downstream applications.

With the help of mentors, we have already prepared everything for registration. However, due to the limited time, we have not integrated registration to our system. We plan to finish this in the summer holiday.

3.5 User interface

We designed four different modes for displaying videos, as shown in Figure 13. Actually, they are the combination of two groups of stereo videos.

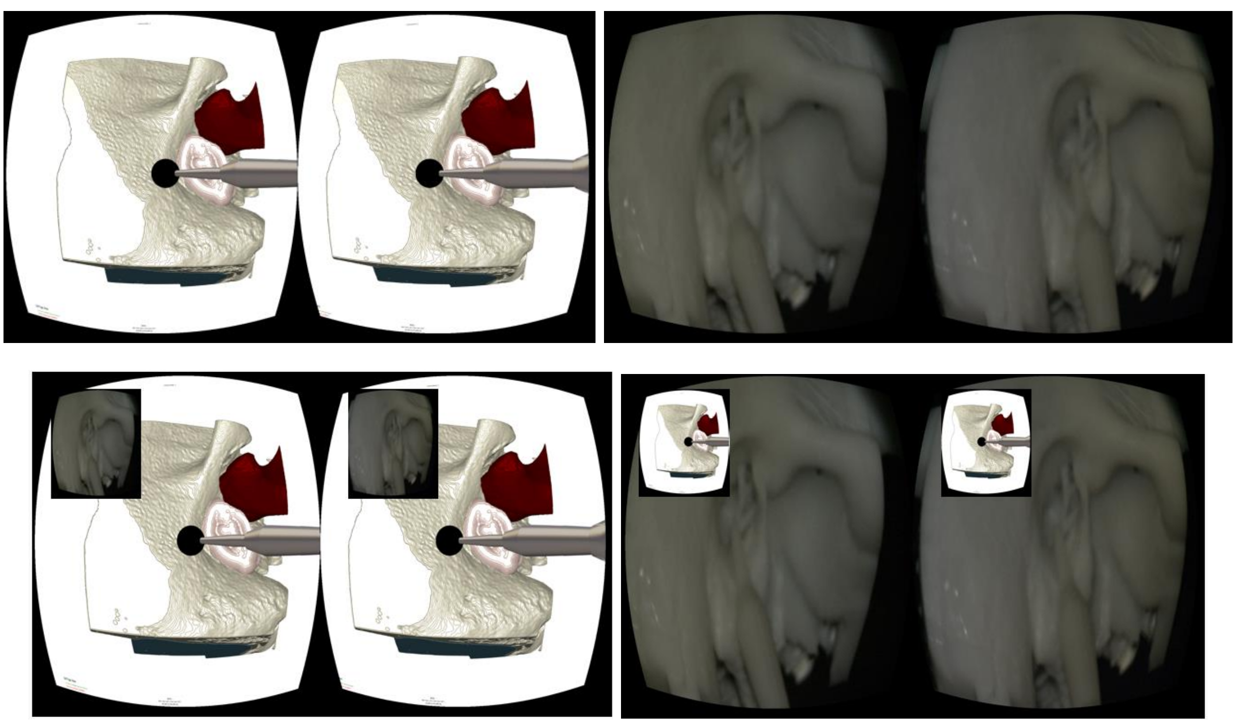


Figure 13 Four modes in HMD

We can only display one video on the whole screen of HMD. Or we can display two videos, one on the big screen while the other one on a small window on the side. Both videos can have 3D views in this mode.

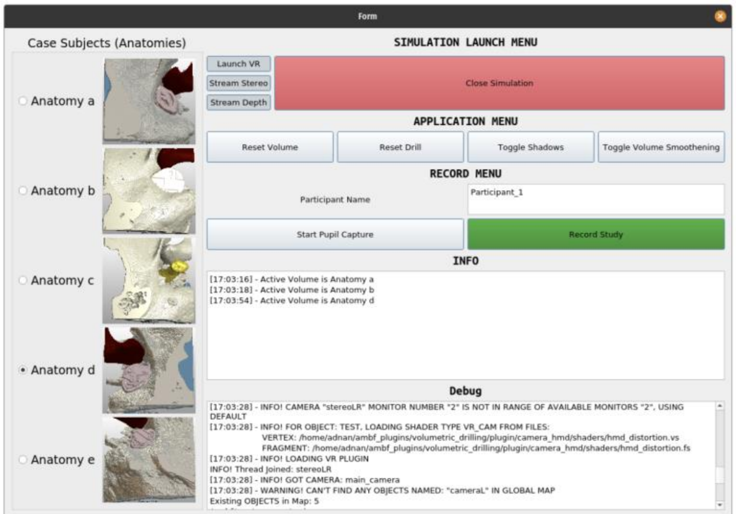


Figure 14 GUI (Graphical User Interface) of AMBF [4]

To switch between these modes, we can develop a GUI based on AMBF's GUI. Figure 14 shows the GUI of AMBF. We may replace the left part with four modes and keep other buttons the same. We may also set shortcuts on the keyboard to switch between these modes.

4. Evaluation and Experimental Setup

We designed three test plans for our system. They are latency time test, system test with models and system test with surgeons.

4.1 Latency time test

In this test, we will focus on the latency time of displaying videos from real microscope in HMD.

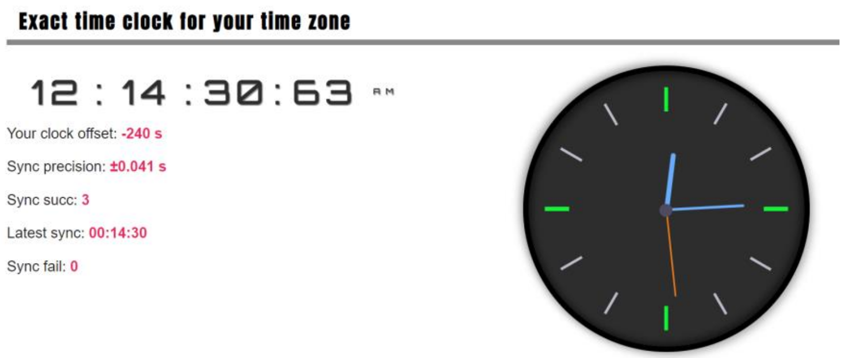


Figure 15 Time zone website

To do this test, we found a website which can show current time in milliseconds as shown in figure 15. We open that website on both a cell phone and our computer. Then, put the cell phone on a table and adjust the stereo microscope to show it. The video will go through the whole pipeline and be shown on our computer. We just need to minus the time in video to the one on the website and that is the latency time.

The ideal result is that users cannot feel these errors. For example, it is hard to perceive a latency of 0.15 seconds in video.[11] But it is hard to achieve. So, the standard of passing this test is that errors should be small enough for surgeons to do the surgery. For example, if the latency is about 0.3 second, although users can feel the delay, the system can still assist well.

4.2 System test with models

In this test, we will test the entire system by ourselves on a model. We will focus on checking whether the entire system can run well.

To do this test, one of us will keep watching HMD while the other one will try to move a drill on the model. If everything is working well, we will then move the camera in AMBF to a different position and different view angle to see whether the video is still good in HMD.

The ideal result is that we can see videos from both AMBF simulation platform and real microscope clearly in the HMD. We can also switch between the four display modes we have designed. If the system can pass this test, it means that the system is finished and can be pushed to user study.

4.3 System test with surgeons

For the maximum goal, we will submit IRB application for user study with surgeons. We will invite surgeons to use our system and give us some feedback. We will then update our system according to their suggestions.

For example, as introduced in the technical approach, surgeons have many choices about simulation videos. But we do not know which one is the most useful one when doing surgery. So, surgeons themselves will choose the video that can help. They can also choose how these videos are displayed.

This test will be conducted in the summer holiday.

5. Results

5.1 Results of latency time test

We did two latency time tests for different methods of video transmission.

The first one is using ROStopic to transmit raw video from stereo microscope.

Figure 16 Time offset in raw video test

Before calculating the latency time, we need to know the time offset between the cell phone and our computer. Figure 16 shows the result. There is a 0.01 second offset because the time on computer is faster than the one on cell phone.

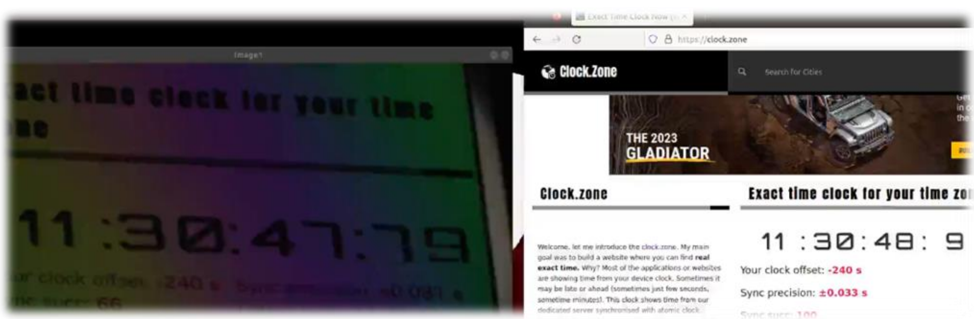


Figure 17 Latency time in raw video test

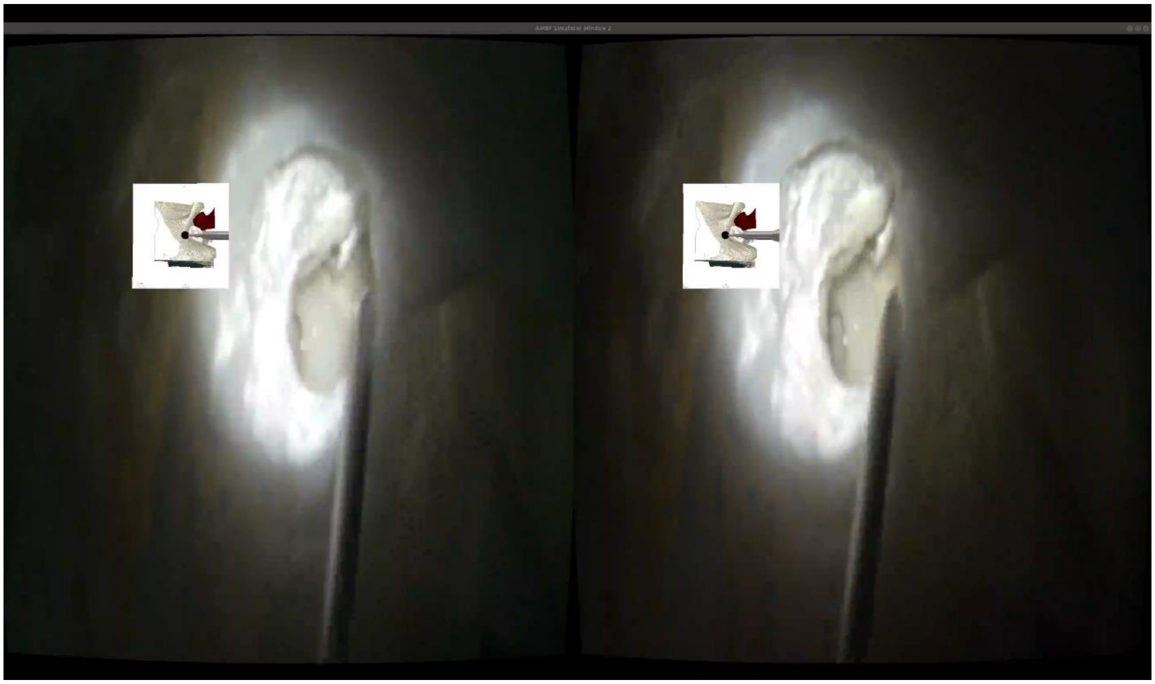
Figure 17 is the latency time in raw video test. The delay is between 0.27 to 0.32 second. Considering the 0.01 second offset, the latency time for raw video is about 0.3 second.

The stereo microscope publishes two ROStopics for each video, one raw video and one compressed video. We did another test to measure the latency time for reading compressed video and uncompress it.

We still need to know the time offset between the cell phone and our computer first. But this time the cell phone is 0.01 second faster than the computer. The final delay is still about 0.3 second.

5.2 Results of system test with models

The system displays process and display videos in HMD. During the whole test, we can see videos clearly in different modes. However, we cannot show the results here without the help of lens.



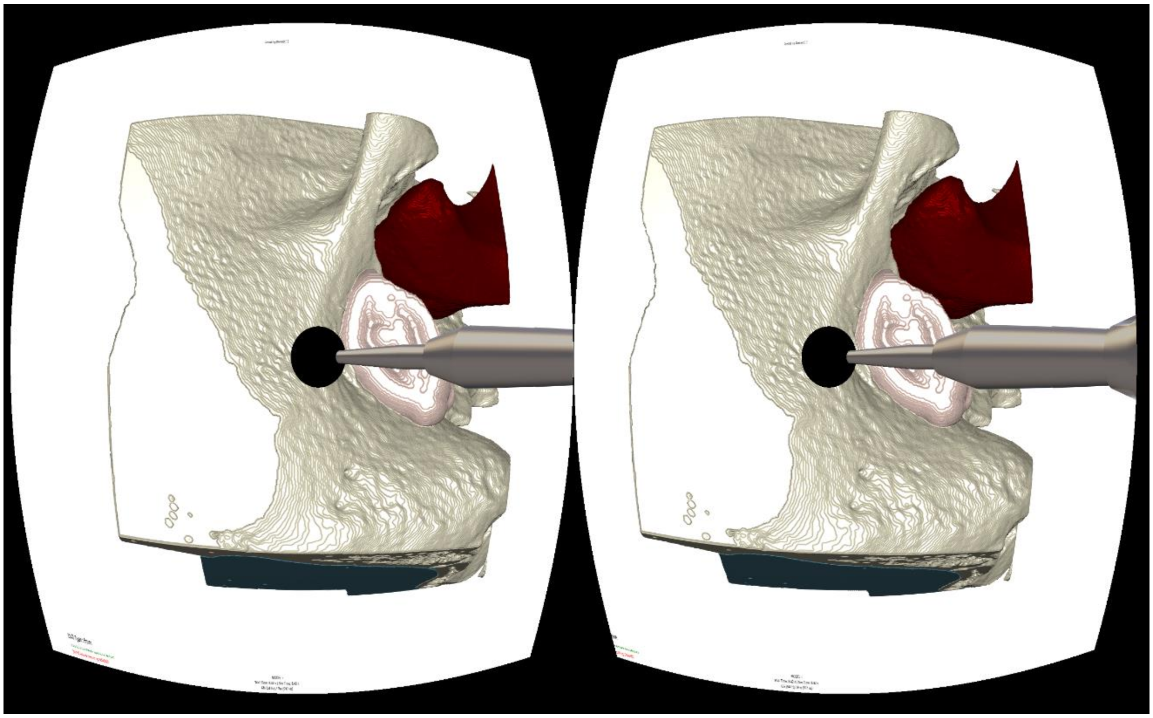


Figure 18 Test with models

Figure 18 shows the video displayed in HMD. It shows the video from a real microscope in main window while the video from AMBF simulation platform in small window. It has some distortion for both videos to fit the HMD. During the test, we rotate and translate the model. We also move a drill on the surface of the model. The video keeps clear during all processes.

In short, the system works well in displaying videos from AMBF and the real microscope on HMD.

6. Discussion

During this semester, we successfully developed a pipeline for augmenting stereo-microscope video with AMBF simulation for the navigation of skull base surgery. Surgeons are able to see both AMBF simulation and microscopic view from a VR headset. Additionally, they can change different modes for displaying both videos. In order to view the anatomy from different perspectives, they can rotate, scaling, and slicing the view on the ABMF simulation incorporating with real-time microscopic video to get more detailed information during the operation.

Overall, we accomplished the basic functionality of the system, but there are still some aspects that need to be optimized. Reducing the latency time is one of the major parts of our project. The current latency time of the system is about 0.3 seconds, but it is not ideal based on our mentor’s feedback, and it needs to be reduced as small as possible so that surgeons satisfied with it. One main reason caused the latency time is our hardware setup: the microscopic video is transmitted from MockOR computer to our laptop by router. If the microscopic video can be directly processed and displayed on the MockOR computer, the latency time of the system could be reduced. This is one of the future works that need to be done for our project.

There are two windows of videos displayed on the HMD, but the small window does not have an excellent display effect. The view of left and right eyes is not perfectly aligned with each other. This problem should be solved by adjusting the distortion coefficients and window scale parameters. Another aspect of the system that needs to be improved in the future is clarity. The video displayed on the HMD is not as clear as it is on the computer. This problem might be caused by resolution of HMD. Using the higher performance HMD could be one solution.

7. Progression Evaluation

7.1 Dependencies

Dependency	Status	Contingency Plan	Planned DDL	Hard DDL
Install ROS on computer	Completed	Already Done	2/13/2023	2/13/2023
Install AMBF on computer	Completed	Already Done	2/13/2023	2/13/2023
Install OpenXR, OpenHDM, OpenCV on computer	Completed	Use computer in lab, or ask mentors for help.	2/20/2023	2/24/2023
Access to MockOR	Completed	Ask mentors to have access to other labs.	2/20/2023	2/24/2023
Hardware Check (HMD, Phantom Omni, Haag Streit Microscope)	Completed	Try to fix them if fails ask mentors to buy a new one.	Continuous	Continuous
IRB Approval	Not Start	N/A	7/1/2023	7/15/2023

Figure 19 Dependencies

As shown in figure 19, we have six dependencies for our project. The first four dependencies are the foundation of all works. So, we finish them at the beginning of the semester. Checking hardware is a continuous task because we always need them to work.

For the goals in summer holiday, we need to have IRB approval for user study with surgeons. This dependency is planned to be finished in the beginning of July.

7.2 Deliverables

	Deliverables	Key Milestones	Status
Minimum	A environment that can run AMBF codes.	Set up environment and learn AMBF codes.	100%
	An AMBF plugin that can read the video from stereo microscope.	Read the video from stereo microscope to computer in proper format.	100%
	An AMBF plugin that can process the stereo microscopic video for HMD.	Process the stereo microscopic video in the computer.	100%
Expected	A system with small latency time.	Minimize the delay of showing videos.	100%
	A system can display both simulation and anatomy videos in HMD.	Display all videos in HMD	100%
	An user interface for switch and control two videos.	Develop an user interface for the system.	100%
	An updated system with registration.	Perform anatomy registration.	80%
Maximum	An user study report.	Conduct user study with surgeons.	Plan to work in summer holiday
	A system can be applied in real skull-base surgery.	Update from the real operation to the simulation.	Plan to work in summer holiday

Figure 20 Deliverables

Figure 20 shows the deliverables and the status of our project. We have finished all minimum parts and most of the expected parts. For the registration task, we have already prepared everything for registration with the optical tracker system. However, due to the limited time, we have not integrated registration to our system. We plan to finish the rest in the summer holiday.

The maximum parts will also be done in the summer holiday. We will first finish our system and apply for IRB approval. Then, we will conduct user study with surgeons and finally update our system according to their feedback.

7.3 Schedule Adherence

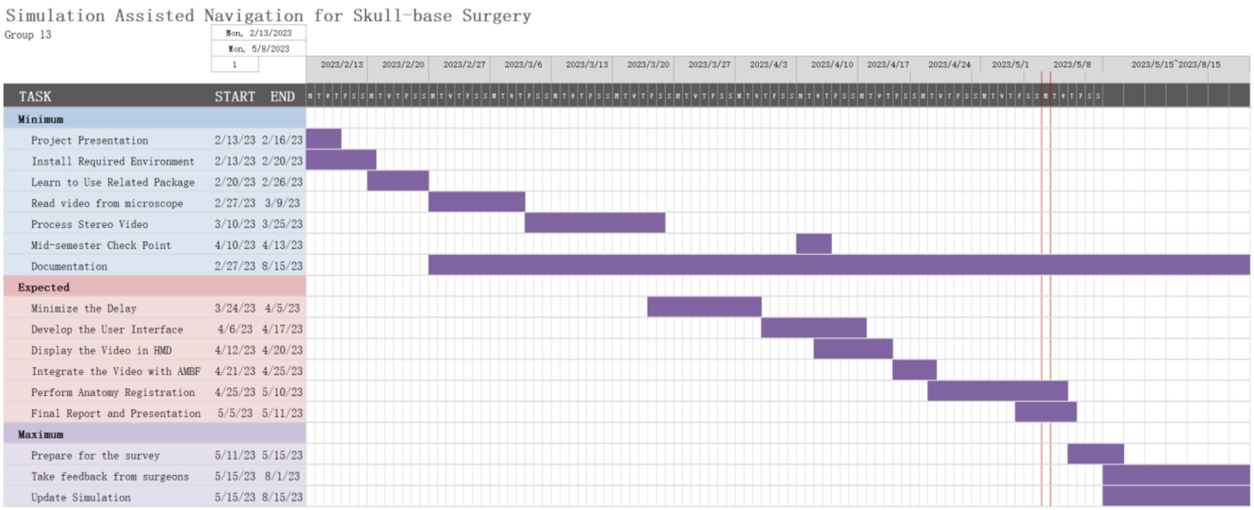


Figure 21 Gantt chart

Figure 21 shows our timeline in the Gantt chart. According to the timeline we made at the beginning of this semester, we will finish all minimum and expected goals before May 11^{th} and all maximum goals before August 15^{th}.

All minimum and most of the expected goals are finished on time. The only task that we are unable to finish is the registration task. This is because the debugging process of hardware and software takes more time than we expected. We believe we can finish it in one more week after May 11^{th}.

7.4 Management and Roles

For the management plan, we have weekly regular group meetings with Dr. Adnan and other mentors on Wednesday 3:00 pm to 4:30 pm. Except for that, we had additional meetings with Dr. Adnan when needed. All meetings are held in person. We usually use Microsoft teams to communicate.

All documentation is stored in Microsoft teams, GitHub, Google Drive or CIS2 wiki page.

Our group consists of two people, Xinhao Chen and Zhaomeng Zhang. The responsibilities for each person are listed here:

Xinhao Chen:

- Read videos from microscope

- Minimize the delay

Zhaomeng Zhang:

- Process videos to fit HMD
- Design user interface

Together:

- Set up environment
- Display videos in HMD
- Write all reports and prepare for presentation

7.5 Skills learnt

- Learned knowledge about OpenGL and ROS
- Improved C++ programming skill
- Understood the mechanism of VR distortion video
- Improved collaboration and communication skills

8. Final Remarks

Overall, our project went very well during this semester, and we completed all the minimum and most parts of expected deliverables that were initially established at the beginning of the semester. Our team members have good relationships and cooperate well with our mentors. Each member of our team has a clearly defined role and responsibilities, which has enabled us to complete our weekly tasks on time and consistently make progress towards achieving our project objectives. Despite some remaining work that still needs to be done, we have decided to continue working on this project throughout the summer to ensure that we meet all our goals and objectives.

Through our work on this project, we have acquired an understanding of various concepts, such as ROS, OpenGL, and Lens Correction Model, and have improved our C++ programming skills. In addition, our communication, cooperation, and problem-solving abilities have also been enhanced. Moreover, we have had the opportunity to work with hardware that we have not had previous experience with, including an advanced microscope and HMD. Overall, we are grateful for the enriching experience that we have had while working on this project and are proud of the work that we have accomplished thus far.

9. Reading List

• Virtual reality for synergistic surgical training and data generation [1]

• Twin-S: A Digital Twin for Skull-base Surgery [2]

• Fully immersive virtual reality system [3]

• Online document of ROS [13], Learn OpenGL [12], and Lens Correction Model [7]

• Calculating Stereo Pairs [7]

References

[1] Munawar, A., Li, Z., Kunjam, P., Nagururu, N., Ding, A. S., Kazanzides, P., ... & Unberath, M. (2022). Virtual reality for synergistic surgical training and data generation. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 10(4), 366-374.

[2] Scholz M, Parvin R, Thissen J, Löhnert C, Harders A, Blaeser K. Skull base approaches in neurosurgery. Head Neck Oncol. 2010 Jul 5;2:16. doi: 10.1186/1758-3284-2-16. PMID: 20602753; PMCID: PMC2913918.

[3] Skull base surgery. Skull Base Surgery | Johns Hopkins Medicine. (2019, November 19). Retrieved February 14, 2023, from https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/skull-base-surgery

[4] Adnan Munawar, Zhaoshuo Li, Nimesh Nagururu, Danielle Trakimas, Peter Kazanzides, Russell H. Taylor and Francis X. Creighton. Fully Immersive Virtual Reality for Skull-base Surgery: Surgical Training and Beyond

[5] Locketz GD, Lui JT, Chan S, Salisbury K, Dort JC, Youngblood P, Blevins NH. Anatomy-Specific Virtual Reality Simulation in Temporal Bone Dissection: Perceived Utility and Impact on Surgeon Confidence. Otolaryngol Head Neck Surg. 2017 Jun;156(6):1142-1149. doi: 10.1177/0194599817691474. Epub 2017 Mar 21. PMID: 28322125.

[6] Shu, H., Liang, R., Li, Z., Goodridge, A., Zhang, X., Ding, H., ... & Unberath, M. (2022). Twin-S: A Digital Twin for Skull-base Surgery. arXiv preprint arXiv:2211.11863.

[7] Lens correction model. Lens correction model - PanoTools.org Wiki. Available at: https://wiki.panotools.org/Lens_correction_model (Accessed: May 7, 2023).

[8] "Volumetric_drilling_real_camera/README.md at Master Jackeyhappie/volumetric_drilling_real_camera." GitHub, https://github.com/jackeyhappie/volumetric_drilling_real_camera/blob/master/README.md.

[9] courses:456:2023:projects:456-2023-13:project-13 – CIIS Wiki. Available at: https://ciis.lcsr.jhu.edu/doku.php?id=courses%3A456%3A2023%3Aprojects%3A456-2023-13%3Aproject-13 (Accessed: May 7, 2023).

[10] Furrer, F., Fehr, M., Novkovic, T., Sommer, H., Gilitschenski, I., Siegwart, R.: Evaluation of Combined Time-Offset Estimation and Hand-Eye Calibration on Robotic Datasets. In: Field and Service Robotics, (2018)

[11] Yuhas, Daisy. "Speedy Science: How Fast Can You React?" Scientific American, Scientific American, 24 May 2012, https://www.scientificamerican.com/article/bring-science-home-reaction-time/.

[12] "Introduction," LearnOpenGL. [Online]. Available: https://learnopengl.com/Introduction. (Accessed: 08-May-2023)

[13] "Robot operating system," ROS. [Online]. Available: https://www.ros.org/. (Accessed: 08-May-2023)