# An Augmented Reality Application for Surgical Planning and Navigation

*A Project Report*

*submitted by*

**PRAKRUTI CATHERINE GOGIA**

*in partial fulfilment of the requirements*
*for the award of the degree of*

**MASTER OF TECHNOLOGY**

**DEPARTMENT OF ELECTRICAL ENGINEERING**
**INDIAN INSTITUTE OF TECHNOLOGY MADRAS.**

**JUNE 2016**

# THESIS CERTIFICATE

This is to certify that the thesis titled **An Augmented Reality Application for Surgical Planning and Navigation**, submitted by **Prakruti Catherine Gogia**, to the Indian Institute of Technology, Madras, for the award of the degree of **Master of Technology and Bachelor of Technology**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.


**Prof. 1**
Research Co-Guide
Prof. M. Manivannan
Dept. of Applied Mechanics
IIT-Madras, 600 036

**Prof. 2**
Research Co-Guide
Prof. Kaushik Mitra
Dept. of Electrical Engineering
IIT-Madras, 600 036


Place: Chennai

Date: 10 June 2015

# ACKNOWLEDGEMENTS

I would like to thank my mentors, Prof. M. Manivannan and Prof. Kaushik Mitra for providing invaluable guidance during the course of this project. I am constantly inspired by Prof. Manivannan's drive to bring applications from the realm of theory into the real world. The lab environment he has fostered and nurtured keeps his students working hard to make meaningful contributions. Prof. Mitra's enthusiasm to explore new technologies and getting his students up-to-date with cutting-edge research is admirable. I thank him for encouraging me to step outside the department and try something interdisciplinary.

I would like to thank Mr. Prabhu Vasulingam for support with the hardware aspects of the project; I aspire to be half as ingenious as he is someday. Many thanks to Ravali, Joseph and the other project associates and students at Touch Lab for their support and company. A special shout out to Mr. Paul for arranging meetings with doctors which proved insightful and shaped my understanding of the problem. I am truly indebted to my band, the Acafellas who kept me sane and made sure I always had an enjoyable music-filled evening to look forward to (after a long day of working on this project!). A very special thank you to Akash who has been a constant companion, motivator and competitor in my life. He has impacted my life positively ever since I've known him. Lastly, I would like to thank my lovely family for helping me every step of the way.

# ABSTRACT

KEYWORDS: Augmented Reality; Neurosurgery; Surgical Navigation

Neurosurgeons perform complex procedures where precision is paramount. Moreover, the anatomy of each of the patients is different. Thus, surgical planning and in-situ navigation assistance assume importance in this field.
Current surgical navigation systems track instruments and use a 2D screen to make a surgeon aware of position and orientation. This suffers from the drawback of loss of attention of the surgeon as she switches between the screen and the patient, and also of viewing only 2D projections of a 3D object. In addition, some frame-based systems cause considerable discomfort to the patient.
Another point of concern is the cost of commercially used systems, a cost which eventually trickles down to the consumer. An Augmented Reality - based application has been proposed to overcome these drawbacks at the surgical planning stage as well as to serve as a navigating aid during surgery. The implementation details of a preliminary proof of concept are detailed in the present thesis.

The result of the present thesis is a pipeline to be able to augment content on any 3D object that can fit on a desktop, view the augmented content on the Oculus Rift and manipulate it using a UI. Objects with regular geometry are found to be better subjects for augmentation. It is also observed that a hand-input based UI is convenient for Augmented/ Virtual Reality Applications. The future course of the project would involve customizing the system for a specific surgery scenario with inputs from surgeons as well as investigating the use of markerless Augmented Reality.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# NOTATION

| | |
|---|---|
| $u, v$ | co-ordinates of a pixel in an image |
| $x, y$ | Varying coordinates on the x and y axes |
| $M$ | Matrix for the calculation of corner points |
| $k$ | Discrete time index |
| $\overline{x_{old}}$ | Old moving average without considering present data point |
| $\overline{x_{new}}$ | New moving average by considering present data point |
| $\epsilon$ | Threshold |

# CHAPTER 1

# INTRODUCTION

## 1.1 Motivation

Augmented Reality (AR) related technology has progressed considerably in the last few decades with an increase in the resolution of displays, precision of tracking algorithms as well as the development and release of consumer-grade virtual reality (VR) and augmented reality devices.
Augmented Reality has made its presence felt in marketing, gaming and also to a smaller extent in fields like medicine and defence. The field of surgery in particular has seen growing acceptance of AR and VR for simulators, pre-surgical planning and even as in-surgery aids.
In simple terms, augmented reality enables the overlaying of virtual graphical content over the real world. AR devices are mostly present in the form of head-mounted displays (HMDs) and heads-up displays (HUDs). They are often used to provide in-context and 3D guidance and to make information of patient diagnostics, vitals and the pre-planned path of surgery available to the doctor right in place. Medical Augmented Reality applications have been developed as early as the year 1996 and have been evolving ever since, see Meola *et al.* (2016) for a review. We chose to address the problem of pre-surgical planning and in-surgery guidance. Current market alternatives suffer from one or many of the following drawbacks.
Pre-surgical planning is only recently catching on and is proving to be very effective due to the following reasons. A rehearsal of a surgery can be performed and a number of eventualities explored. This is especially important as every patient has anatomical uniqueness which can be better navigated after training.
Current systems using cross-sectional imaging modalities require the mental transformation of a number of 2D images to the underlying 3D anatomy. Studies show that surgeries performed with 3D visualization systems gave surgeons more confidence that those performed with 2D slice viewers according to Bhayani and Andriole (2005) and

Mert *et al.* (2012). Augmented Reality with HMDs that overlay anatomical models on existing structures make the interface even more real and manipulable.

Having a head-set on at all times and all information projected in it ensures that the surgeon does not have to switch gaze and more importantly attention to check patient vitals/ recenter the image. The overlay of 3D Models on the skin helps doctors make important decisions such as where to place the craniotomy while performing open brain surgery.

The current system includes the implementation of an AR application on the Oculus Rift. A custom 3D model of a brain is augmented onto a mannequin head. This augmentation tracks changes in the displacement and orientation of the head. The model's parameters such as transparency, hiding/ displaying separate parts can be accomplished using a UI.

The Oculus Rift is essentially a VR device and the ïňĄrst part of the project involves equipping the Rift to support video see-through AR. The second part involved the design of the software for tracking and augmentation of a 3D model on a real-world object. The final part of the project involved the design of the UI (User Interface) to manipulate the augmented model. In a previous work Natekar and Manivannan (2015),a method to create a 3D model from MRI dicom images was proposed. The model created out of patient data will be used as the augmentation material.

## 1.2 A Brief Review of Alternate Systems for Surgical Planning and Navigation

A review of Surgical Planning and Navigation technologies that use Augmented Reality as well as those without follows. The drawbacks and advantages of each system, as well as major challenges that need to be addressed are mentioned.

### 1.2.1 Laproscopy and Endoscopy

AR is been used as an in-surgery navigation guide for laproscopy and endoscopy Fuchs *et al.* (1998). These systems use either an optical see-through system or a video see-through system for augmented content. They usually require an extra light-emitting device attached to the tool. The key challenges to be addressed are the registration of the body structures with their imaged counterparts and calculation of the distance of the tumour/ point of interest from the tool tip. The narrow field of view also is a challenge. These applications augment information on a 2D screen with the images from the laproscope.

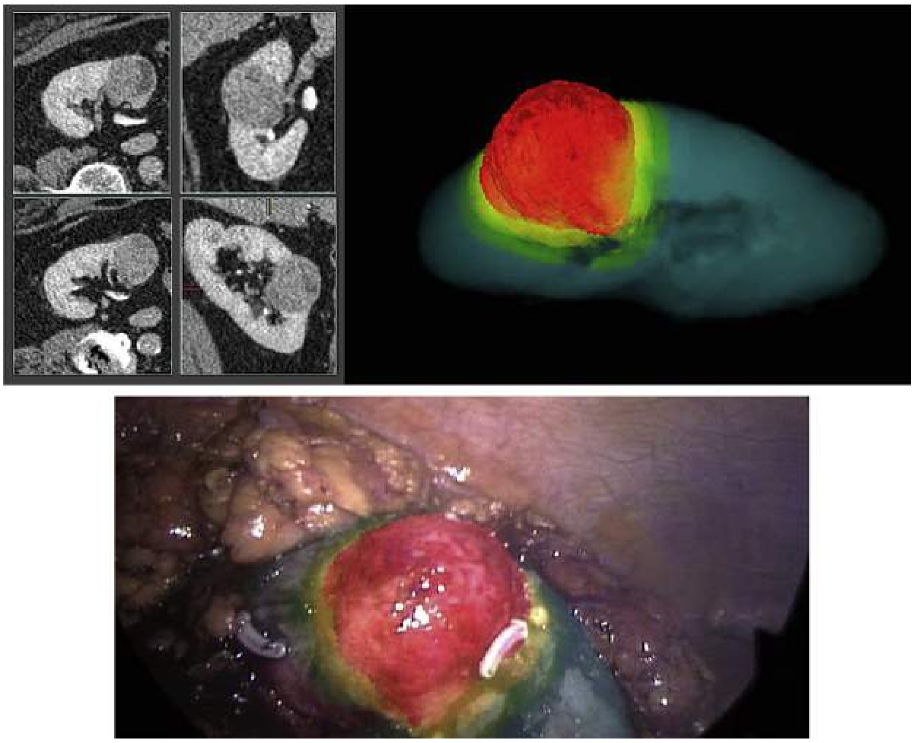
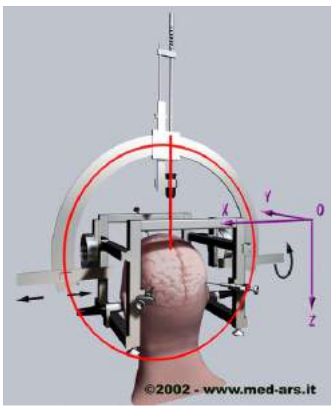
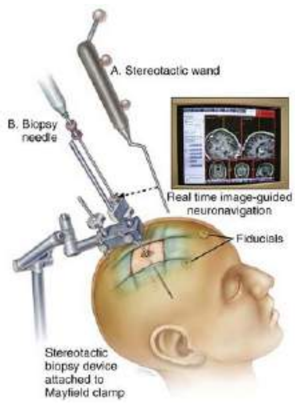


Figure 1.1: AR application for Laproscopic surgery navigation




(a) Frame-based Stereotactic System




(b) Frameless Stereotactic System

Figure 1.2: Frame and frame-based system

### 1.2.2 Stereotaxy: Frameless and Frame-based

The current systems useful for surgical planning and navigation which do not use AR or VR are Frameless Stereotaxy and Frame-based Stereotaxy.

A frame-based system for surgical navigation has several benefits, it uses a frame fixed to the head that acts as a fixed coordinate system. This coordinate system enables very precise navigation and hence is minimally invasive. This technique is used for biopsies. However, it generally provides targeting of only a single point rather than orientation to three-dimensional anatomy. Further, it functions primarily to direct a probe to a pre-selected target, not to find the location of an independently positioned instrument on an image. In addition, the frame itself is a constraint during surgery.

The technique involves taking intra-operative MRIs, before and after the craniotomy. This is because the craniotomy causes a small but in some cases, significant change in the orientation of the brain. Taking an intra-operative MRI is cumbersome and is the major drawback of the system.

Frameless Stereotaxy uses different digitizers to create a co-ordinate system and ascertain the position of a point within space. Optical, sonic and mechanical digitizers have been documented. These may require fiducial markers on the subject, but do not suffer from the drawbacks of the frame-based system. However, each of these methods display cross-sectional 2D views on a screen and are also considerably expensive systems.

### 1.2.3 Commercial Products: Surgical Theater and Brainlab

Surgical Theater is a start-up based out of Cleveland, OH. Its flagship product, SuRgical Planner (SRP) creates realistic models of patient-specific anatomy and surgical tools. The surgeon can perform a rehearsal of the surgery in an immersive VR environment using an Oculus Rift. The second product offered by Surgical Theater is Surgical Navigation Advanced Platform (SNAP), which offers tool-tracking and a sectional view of the anatomy.

Brainlab introduces instrument tracking within surgery and shows sagittal, axial and coronal views on a 2D screen. Brainlab neuronavigation tracks tools in real time and

(a) Surgical theater's system (b) Brainlab's system

Figure 1.3: Commercial products for surgical navigation

displays their position in relation to the patient data. Surgeons are guided through procedures, helping to keep incisions small and minimize damage to healthy structures.

The end goal of the project can be briefly summarized to be of developing an economical alternative to SRP.

## 1.3 A mandate for an ideal system

Apart from the review of technologies above, we conferred with doctors at CMC Vellore and Tamil Nadu Government Multi Super Specialty Hospital in Chennai. At both these places, the most urgent need the doctors spoke of was of a reduction in cost as compared to the current systems.

Another pressing need is to reduce the number of MRIs taken intra-operatively, as is done in existing navigation systems.

Thus we list down a mandate for a working Neurosurgical Navigation system as follows.

1. *Low-cost* - Affordable in the Indian context.
2. *3D visualization and In-place* - To prevent switching between consulting a 2D screen and have a detailed reference in place.
3. *Minimal use of markers/ frame/ wires* - The system should not be an impediment to the patient comfort/ interaction between the patient and the surgeon.
4. *Accurate tracking and registration* - Images obtained from various imaging modalities need to be tracked and registered.

The present system uses augmented reality as it satisfies the requirements to an extent.

## 1.4 Problem Definition

Various imaging modalities capture 3D information that is useful while performing neurosurgery. Examples of this information include; the location of the tumour, eloquent areas of the nervous system, veins and other structures that the surgeon may want to avoid damaging.

In current systems, tool tips are tracked and a 2D screen displays the relative position of the tool and anatomical structure to aid navigation. A surgeon has to make the jump from a number of 2D views to uncover the spatial relation between the 3D structures. Moreover, the surgeon has to constantly consult the screen while his hands operate at the surgery table.

Employing a system that displays the 3D model, overlaid over the anatomical structure eliminates these problems. Thus, the broad problem definition is to build a system capable of tool tracking, performing registration between images and rendering augmented content. Such a system can give unique guidance and insight to the surgeon in-situ.

## 1.5 Contribution of the Thesis

To assist the larger goal of building a surgical planning and navigation system, this thesis presents a proof of concept, which allows the augmentation and tracking of a 3D anatomical model (human brain) on a real world object (mannequin head). Moreover, we provide a method to augment any desktop-sized 3D object with a corresponding model and enable its viewing in the HMD. User input using a gamepad and leap motion controller are investigated and in conclusion, the project is demonstrated with all the UI and Augmented elements.

## 1.6 Overview of the Thesis

After a brief introduction to Augmented Reality and discussion of alternatives in Chapter 1, various aspects of implementation are discussed in Chapter 2, results are discussed in Chapter 3, future scope and limitations are discussed in Chapter 4.

# CHAPTER 2

# IMPLEMENTATION OF PROPOSED SYSTEM

## 2.1 Workflow

The designed systems takes in a customised 3D model of the brain and augments it on a mannequin head. The UI (User Interface) makes it possible to manipulate the augmented model of the brain. To achieve this objective, the system is divided into modules

1. Model Scanning Module
2. Camera Mount Module
3. Unity Application Module - Model Augmentation
4. Leap Motion Controller - UI Module

The project pipeline is briefly described in 2.1.

## 2.2 A description of the tools used

The project aims to implement an Augmented Reality system using a Head Mounted Device (HMD). We now present a comparison of several HMDs both intended for VR and AR use. Discussion on the other software tools used follows. Parameters necessary for comfortable AR viewing are discussed in the end.

### 2.2.1 A comparison of Head Mounted Devices (HMDs)

AR can be implemented using a video see-through approach or an optical see-through approach. Optical see-through devices are more intuitive in AR use, however due to video see-through considerations even VR headsets could be considered. In this project we will discuss two VR headsets; the Oculus Rift and the HTC Vive and two AR headsets; Meta Glass and Microsoft Hololens.

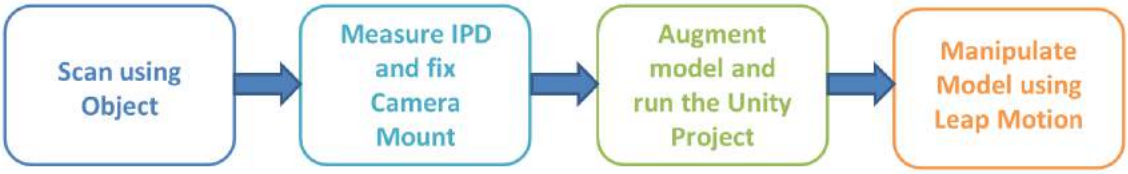



Figure 2.1: Project Pipeline

Table 2.1: Comparison of VR Headsets (Consumer Versions)

| | Oculus Rift | HTC Vive |
|---|---|---|
| Tethered | Yes | Yes |
| Host OS | Windows, Linux, OSX | Windows, Linux, OSX |
| Price | $599 | $799 |
| Resolution | 1080 x 1200 per eye | 1080 x 1200 per eye |
| Field of View | 110° | 110° |
| Positional Tracking | Yes | Yes |
| Refresh Rate | 90 Hz | 90 Hz |

Thus, we see that these two headsets have very similar specifications except for the cost. The added cost for the Vive is due to its wireless controllers. A comparison between the two can be found at `http://www.wareable.com/vr/htc-vive-vr-headset-releas`

Table 2.2: Comparison of AR Headsets

| | Meta Glass | Microsoft Hololens |
|---|---|---|
| Tethered | Yes | No |
| Price | $2985 | $3000 |
| Resolution | 1080 x 1200 per eye | 1080 x 1200 per eye |
| Field of View | $90°$ diagonal | $\approx 40°$ diagonal |
| Positional Tracking | Yes | Yes |

A comparison between the two can be found at `https://vbandi.net/2016/03/04/hololens-vs-meta-2/`. It seems that though Hololens has a narrower field of view its technology for tracking is more robust than the Meta Glass. It is also wireless which may be an important benefit in applications.

### 2.2.2 System Components

The following components were included in the system design

1. Computer System
   Virtual environment rendering requires high amount of computation and graphical processing. It is very difficult to acquire a seamless VR experience on an average computer setup. The modeling and rendering takes high amount of computer resources and hence the system is assembled in order to cater for the high computational need. The system has Intel(R) Xeon(R) CPU E3-1220 v3 @3.10GHz 3.10GHz at core as CPU. It is a 64-bit Instruction Set with 4 cores and 4 number of threads. The system is equipped with Nvidia GTX 980 graphical processing unit (GPU).

2. Oculus DK2
   The Oculus DK2 is a VR HMD.

3. Unity
   Unity is a cross-platform game engine developed by Unity Technologies and used to develop video games for PC, consoles, mobile devices and websites.

4. Logitech Webcam C920
   The Logitech Webcam is a high resolution video conferencing webcam which was used for implementing the video see-through.

5. Vuforia
   Vuforia is an SDK for creating Augmented Reality Experiences. It provides robust tracking and object recognition functionalities.

6. Leap Motion Controller
   Leap Motion technology is designed to allow users to control their computers with hand gestures alone. This is used for the UI as the Oculus occludes one's vision.

7. Maya Maya is a 3D modeling software, this was used to create some of the models used in the Unity project.

## 2.3 Camera Module and Mount

For the purpose of the project, the Rift is turned into a video see-through AR device. This requires two camera feeds that act as the left and right eyes to display the surrounding environment. The brain performs stereopsis and we experience the world in 3D. The graphical content is then augmented on each eye's feed. The current system was built on the same lines as `http://willsteptoe.com/post/66968953089/ar-rift-part-1`.

### 2.3.1 Camera Selection

To maintain a reasonable resolution and field of view, appropriate cameras need to be chosen. The Rift has a field of view (FOV) of 90° horizontal and 100° vertical depending on the IPD. We are limited by the resolution and FOV of the Rift.

The cameras chosen were Logitech C920 cameras with the following specifications.

Table 2.3: Logitech Camera Specifications

| Specification | Value |
|---|---|
| VFOV | 70.42 ° |
| HFOV | 43.30 ° |
| FPS | 30 |
| Resolution | 1280 x 720 |
| Auto-focus | Yes |

An added advantage is that since the camera circuit board's height is not significant, it does not further contribute to the offset from our actual eye.

### 2.3.2 The Camera Mount

The mount was made from an acrylic sheet and designed to fit over the front surface of the Oculus. It uses a snapping mechanism to latch onto the Oculus Rift. The distance between the two camera centers is adjustable and can be changed to suit a user's IPD.

The camera is placed in a configuration with the left eye camera in landscape mode and right eye camera in portrait mode. The reasons for this are discussed in 2.5.2. The optical axes of both cameras are fixed perpendicular to the surface of the Rift i.e. in parallel mode with symmetric frusta. The users who tested the device did not experience any difficulty with fusing the left and right images with this setup. See `https://en.wikipedia.org/wiki/Stereopsis` for more information on image fusion.

However, symmetric frusta and parallel camera placement is not the best way to create stereo. This leads to the viewing plane being located at infinity and causes everything before the viewing plane to pop out. The article at `http://doc-ok.org/?p=77` explains why parallel cameras with skewed frusta are the best option for good stereo.

Another alternative is the *toed-in configuration*, this could be explored in a further iteration of the project.

The camera mount is displayed in Figure 2.2.

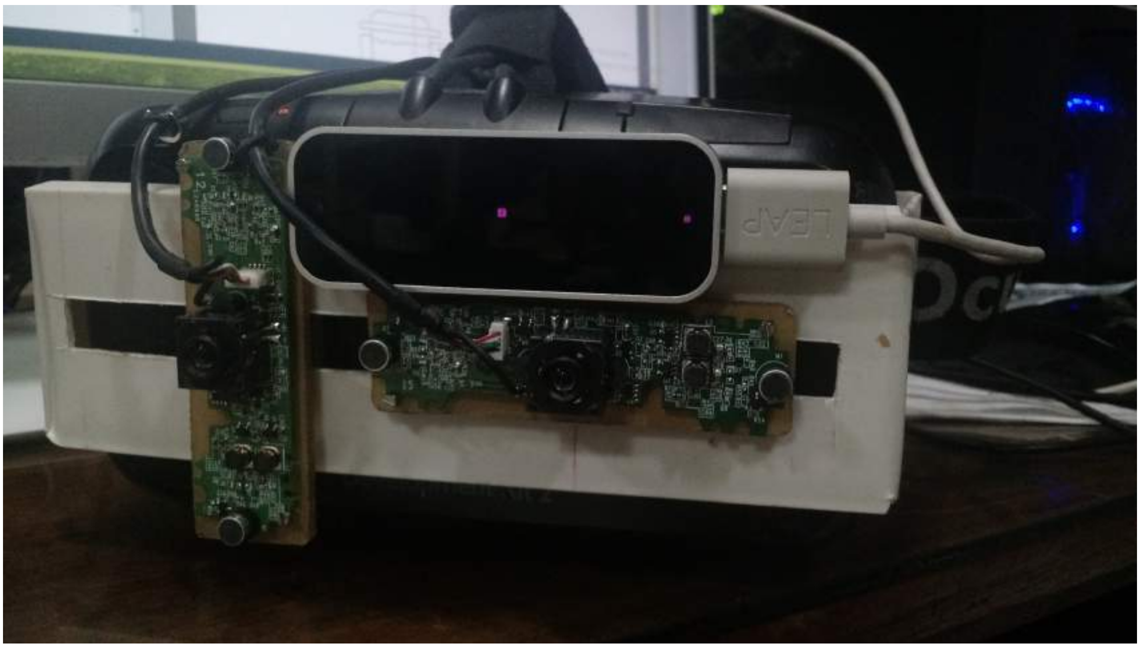


Figure 2.2: The camera mount with the two Logitech C920s and the Leap Motion

## 2.4 Model Scanning Module

The purpose of the scanning module is to scan a 3D object in order to use it as a subject to augment content on. Thus, the 3D Object needs to have a fair amount of interest points that ensure repeatable and accurate recognition and tracking. We use marker-based AR in this application.

### 2.4.1 Vuforia Object Scanner

Vuforia provides a scanning application that can be used to create scans of 3D objects. The scanning procedure is described in (this link).

A brief description of how this technology may be implemented is given below. A database is created during the scan that extracts corners and then classifies them according to their strength based on a metric.

Harris corners are used because we seek to identify the same points across multiple views of a scene. Corners are relatively stable features across multiple views. The rough definition of a corner in an image is a point around which the gradient is high in both directions.

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u,y+v) - I(x,y)]^2$$

Where $w(x, y)$ is a weighting function and $I$ is the intensity at a point. To find corners we need to maximise

$$\sum_{x,y}[I(x+u,y+v) - I(x,y)]^2$$

Using a first order approximation, the above term can be written as

$$\approx \sum_{x,y}[I(x,y) + uI_x + vI_y - I(x,y)]^2$$

We can rewrite it as

$$\sum_{x,y} u^2{I_x}^2 + 2uvI_xI_y + v^2{I_y}^2$$
$$\sum_{x,y} \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

$$Let M = \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}$$

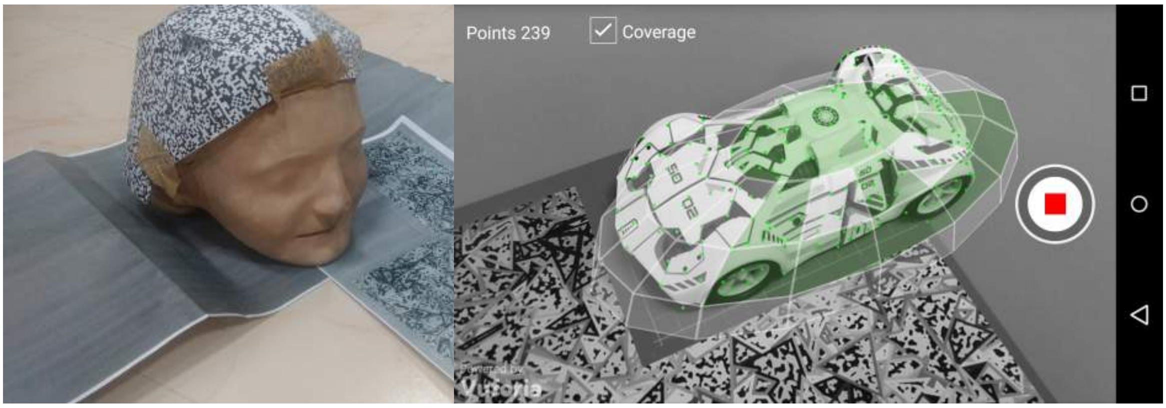
The corner points are those for which the principal component ellipse for $M$ is nearly a large circle. This corresponds to large and approximately equal eigenvalues $\lambda_1 \approx \lambda_2$. A measure that incorporates both these requirements is the cornerness property $R$

$$R = detM - k(trace(M))^2$$
$$detM = \lambda_1\lambda_2$$
$$trace(M) = \lambda_1 + \lambda_2$$

Where k is an empirically determined constant; k = 0.04 - 0.06

Other methods of corner detection can also be implemented eg. Shi-Tomasi Corners. Every time a scan is made a database of corners and their relative location in a 3D volume is recorded. Now, almost the same points will be detected repeatedly regardless of the view. However we need to map corresponding points. This is usually done by calculating a descriptor such as SIFT (Scale Invariant Feature Transform) and comparing descriptors in different views. (Disclaimer : Vuforia does not publish details of how the object scanner functions).

Once the 3D object is scanned, the database is uploaded to the Vuforia Developer




(a) Scanning Environment (b) An example of a scan

Figure 2.3: Scanning module

site under Target Manager to convert it to a Unity Package. The scanning process is shown in Figure 2.3.

## 2.5 Unity Application Module - Model Augmentation

This module is the Unity Project that performs the augmentation of the model. The Unity Scene is a level that holds the objects of a game. The scene hierarchy is shown below. The AR/VR Sample from the Vuforia Digital Eyewear Samples was used as a template for the project. The sample, however, was designed to use the feed only from one camera. A modification to the code was made to include feed from a second camera for true stereo and wider FOV.

### 2.5.1 Virtual Cameras

The scene has two virtual cameras that show the left and right eye perspectives. These are attached to the OVR Camera rig and are tracked by the Oculus Rift. We can project different content to both the eyes by putting the objects we want in separate layers; $Cam1$ and $Cam2$.

The camera feeds are projected onto quads in Unity, with the right eye quad named Webcam2 using the script WebcamTexture in A. The quads do not cover up the entire view of the screen in the Oculus as the FOV of the camera is not large enough. This can be remedied in a future iteration by adding a wide angle lens and correcting for

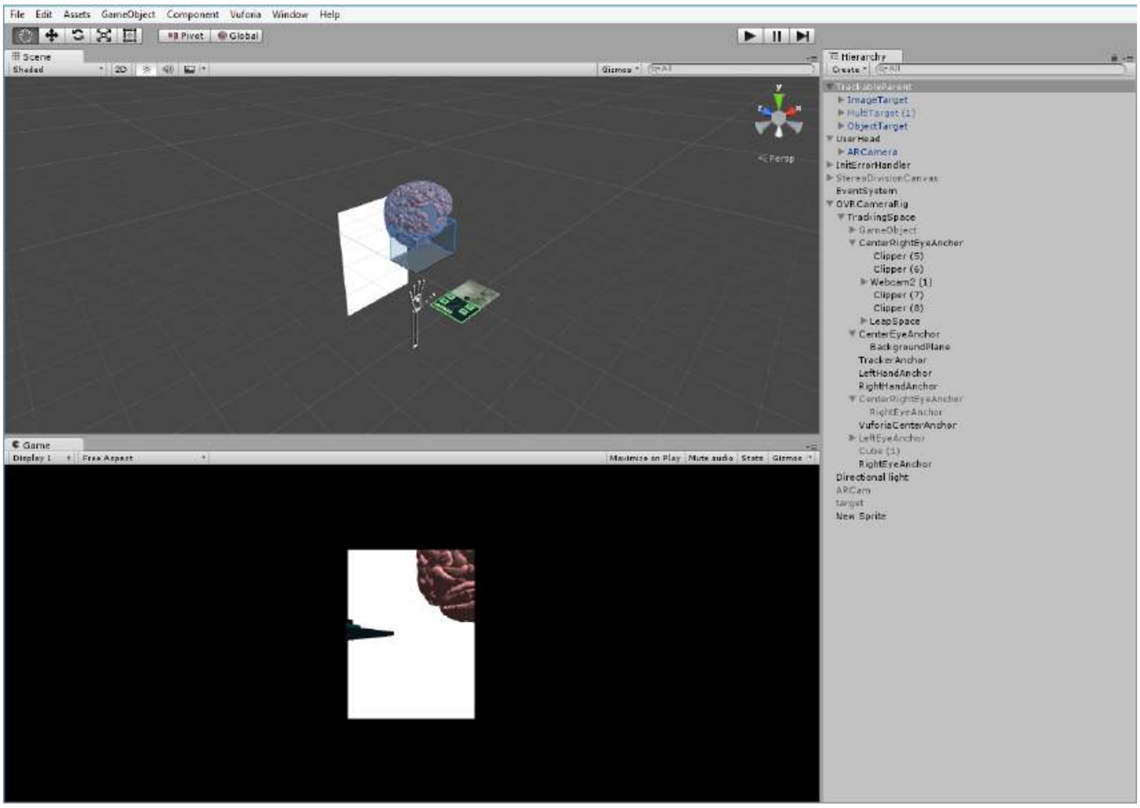



Figure 2.4: The scene hierarchy in Unity

distortion.

The following scripts are implemented on the left eye. The VRIntegrationHelper script helps with Oculus integration and augmentation, the VideoBackgroundBehaviour script displays the camera feed on the Background plane and the HideExcessAreaBehaviour script makes certain that the augmented model does not exceed the bounds of the background plane. On the right eye camera the augmented model is prevented from exceeding the bounds by placing occluding planes at the near plane of the camera.

The important parameters for the Virtual Camera are as follows

**Projection** : Perspective

**Frustum** - Near Plane : 0.01, Far Plane : 1000

**Culling Mask** : Cam1 included in CenterEyeAnchor, Cam2 included in CenterRightEyeAnchor

**Field of View** : 90

**Target Eye**: Left / Right


(a) An example target (b) Augmentation on target

Figure 2.5: Example augmentation

### 2.5.2 Augmentation

The Vuforia SDK uses an AR Camera, which views the camera feed from CentreEyeAnchor. The augmentation is implemented using the feed from a single camera on the left eye. The augmentation in this case is a brain model, constructed from an MRI image. The MRI was obtained from a forty year old male diagnosed with epilepsy. The tumour is located in the right inferior parietal lobule involving both the supramarginal and angular gyri. The sensory gyrus was pushed anteriorly lobe. An example of marker-based augmentation is shown in 2.5.

When the object of interest / model is detected, the plane is offset to intersect the augmentation at an appropriate angle. The scale of the plane has to be appropriate to keep the apparent size of the object intact. A script to make the right eye quad follow the left eye background plane is given in A where the scale factor is calculated as

$$Scalefactor = Z_{final}/Z_{initial}$$

Vuforia's coordinate system uses the camera feed in landscape mode (this is not an editable property), and thus the cameras are placed in different modes. The right eye camera could not be placed in landscape mode as this would exceed the normal IPD.

### 2.5.3 Model Flicker

The augmented model position flickers around the actual position. The Script ModelFlicker.cs solves this using a combination of a moving average and thresholding. The goal is to prevent flickering around a point but moving the object smoothly when actual motion is detected.

Let $x_k$ be the current measured position, let $\overline{x_{new}}$ be the position as calculated by the moving average algorithm. Let $\overline{x_{old}}$ be the old moving average.

The current position is updated as follows, $\epsilon$ is the threshold.

$$x_k \rightarrow \overline{x_{new}}$$

$$if$$

$$|\overline{x_{new}} - \overline{x_{old}}| > \epsilon$$

$$else$$

$$x_k \rightarrow \overline{x_{old}}$$

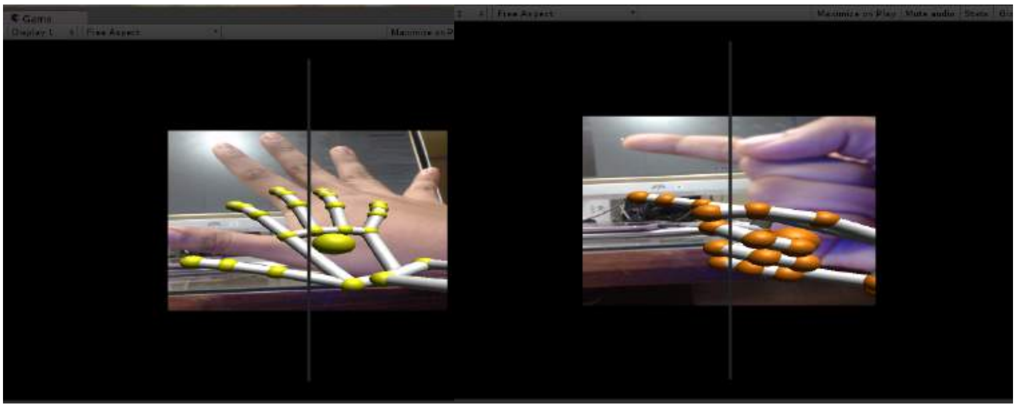
### 2.5.4 Stereoscopic disparity

Stereoscopic disparity is the difference in views that the two eyes see of an object due to the offset between them.

To simulate the phenomenon for a virtual object, we need to offset the virtual cameras. However, an internal setting in the OVR Camera Rig prevents this.

A solution to this was found by duplicating each virtual object and placing them in separate layers. The camera centers lie on the camera axis. The effect of offsetting a camera in one direction can be simulated by offsetting virtual objects in the opposite direction.

The camera axis direction is calculated by using the tracking information of the Oculus. The offset distance is the user's IPD. The corresponding script is in A.



(a) Hand Gesture 1 (b) Hand Gesture 2

Figure 2.6: Gestures tracked by Leap

## 2.6 UI Module - Leap Motion Controller

The Leap Motion Controller is a input device, it is convenient for VR as one's vision is occluded. It has 3 infrared LEDs which take in light outside the visible spectrum $\approx 850$ nanometers. Two cameras capture stereoscopic images. The workings of the leap motion algorithm have not been detailed but it does not calculate a depth map, according to (this link).

An image target is used to display the UI. It currently has functions to display parts of the brain, change the transparency and view the deep veins in the brain. The UI also possesses functionality to switch on a night mode where only the model is visible but is still tracked.

The UI also enables rotation of the model and displays relevant information in 3D text boxes for the surgeon's benefit.

The leap motion is mounted on the Oculus and a model of the hand is overlaid on the screen. The user is guided by the feedback of the model as well as elements like the lighting up of a button on being actuated using a finger.

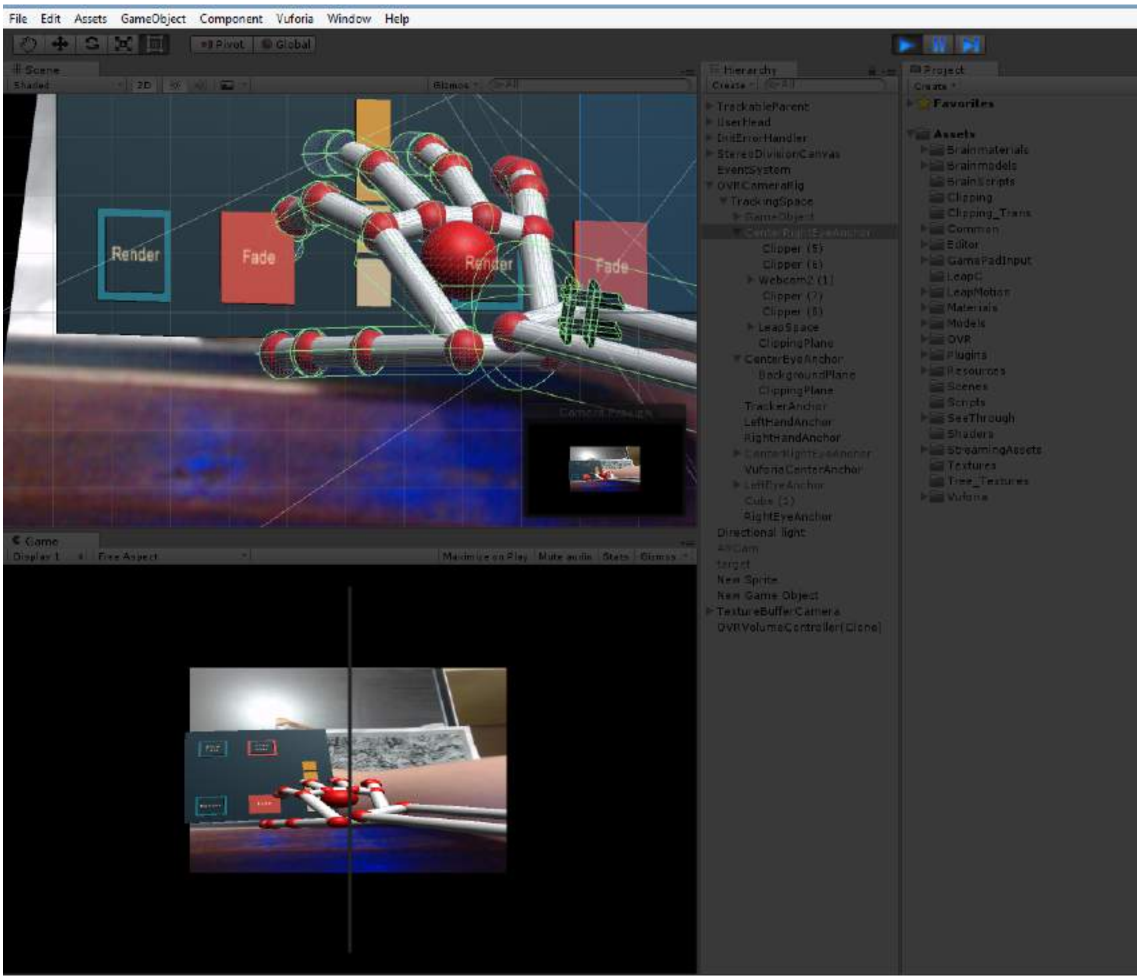



Figure 2.7: Actuation of a button using the Leap Motion

# CHAPTER 3

# RESULTS

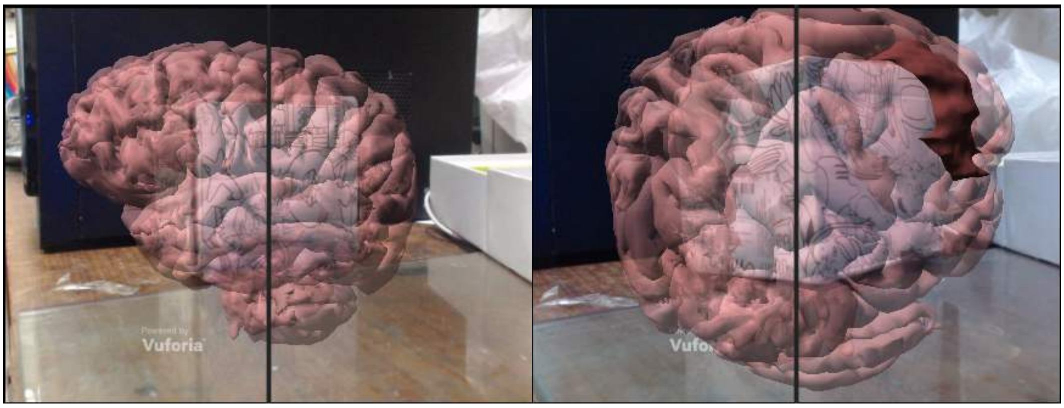
## 3.1 Augmentation on a cube

From Figure 3.1 we see the augmentation on the cube. Users found that since the cube is an object with regular geometry, the augmentation on it performed better than that on the mannequin head in the following section. This geometry did not need an object scanning step. The cube augmentation used a Mutlitarget prefab provided by Vuforia. One had to provide pictures of the 6 faces in the right orientation and download the multitarget database as a Unity Package from the Vuforia Developer Site. The steps to do this are detailed in `https://developer.vuforia.com/library/articles/Training/Multi-Target-Guide`

## 3.2 Augmentation on a mannequin

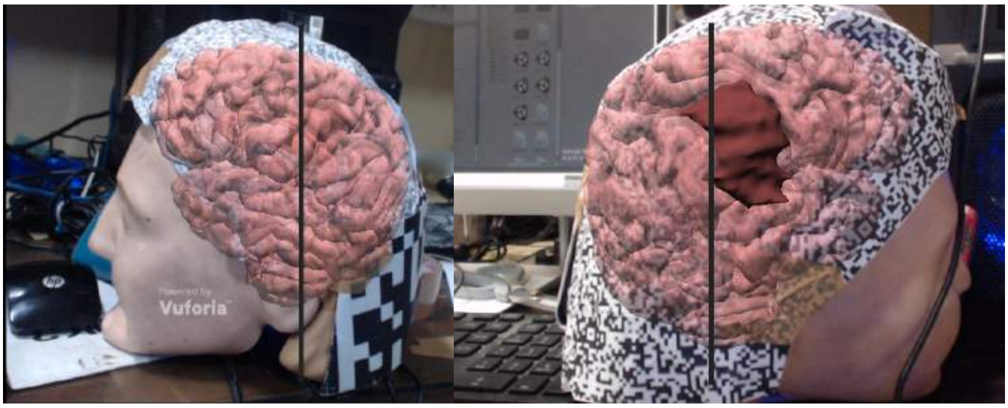
Augmentation was performed on a mannequin head that was covered with markers. The object had irregular geometry and thus was scanned using the Vuforia Object Scanner. It was initially found that the scaling was not appropriate and hence the object was not tracked properly and this is a problem with almost all objects that are larger than 100mm x 150mm (the size of the image target).

Possible solutions or work-arounds to this are by breaking up the object into parts or by using a larger target image to scan the object. Thus, when an A3-sized image was used as a scanning target the object tracking improved considerably.




(a) Left side view of augmentation (b) Right side view of augmentation

Figure 3.1: Augmentation on a Cube




(a) Left side view of augmentation (b) Right side view of augmentation

Figure 3.2: Augmentation on Mannequin

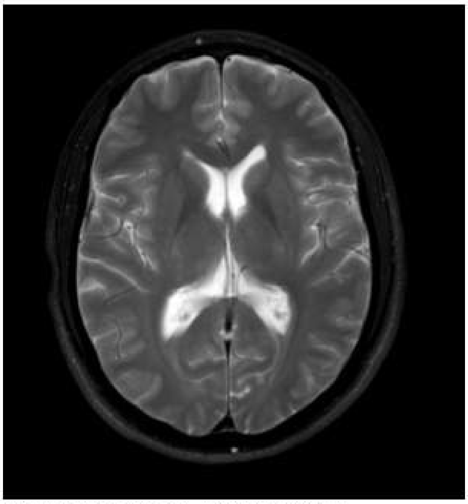
A few screenshots of the system are shown below.

The 3D Brain Model is created from the MRI images. Shown are the T1 weighted MRI images of a 40 year old male, with a tumour on the right inferior parietal lobule in 3.3. The sensory gyrus was pushed anteriorly.

The brain model created from the MRI images is shown in 3.4

An example of the stereo projections of the real world is shown in 3.5. As is evident the object appears slightly shifted due to the distance between the two cameras. Also, note that the right camera is in portrait mode.

T1 weighted MRI images for model-1



*Drawing 1: Model_1 Axial_view_1*

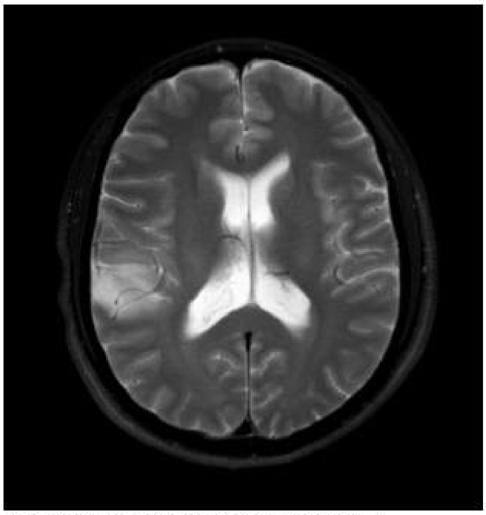


*Illustration 19: Model_1 Axial_view_2*

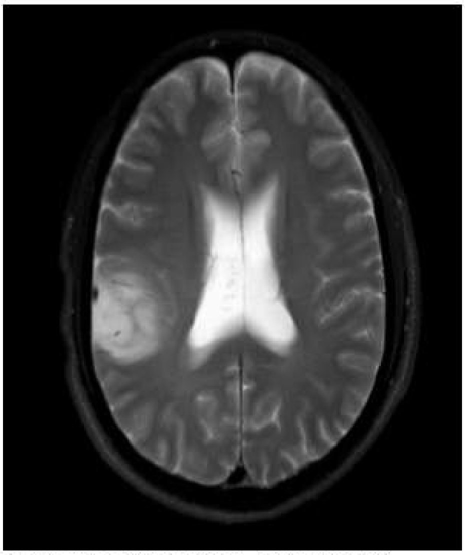


*Illustration 18: Model_1 Axial_view_3*

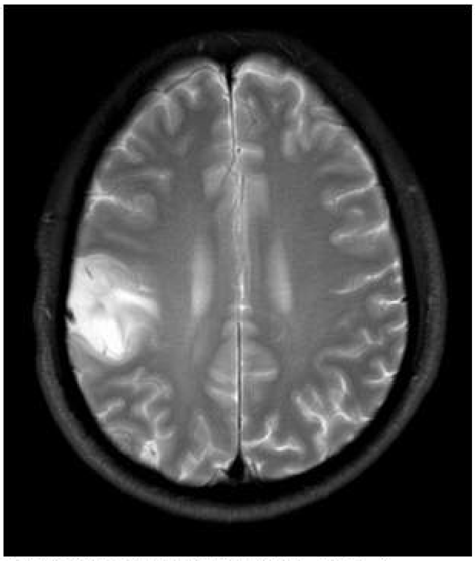


*Illustration 17: Model_1 Axial_view_4*

Figure 3.3: The T1 weighted MRI (taken from Natekar and Manivannan (2015))

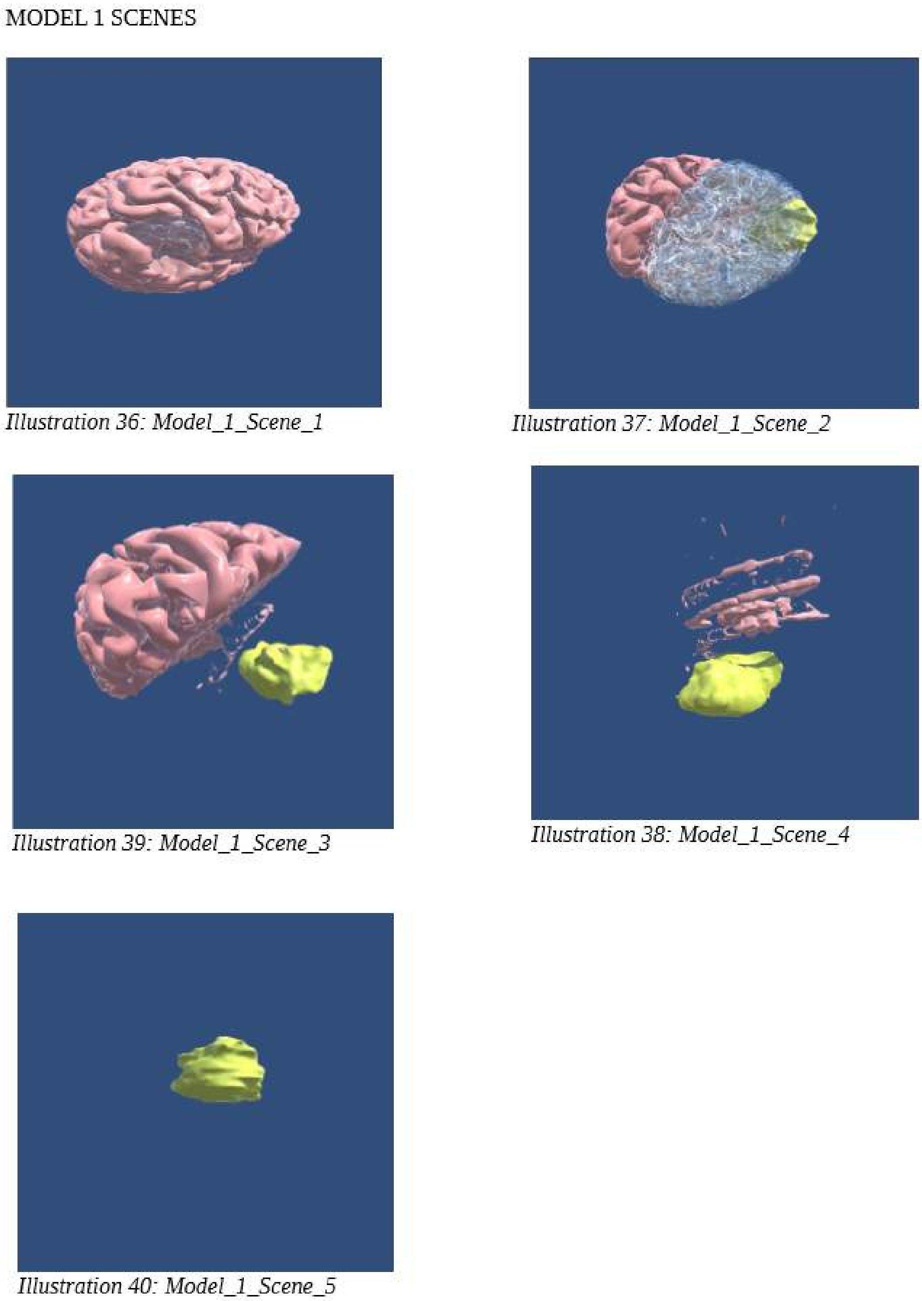
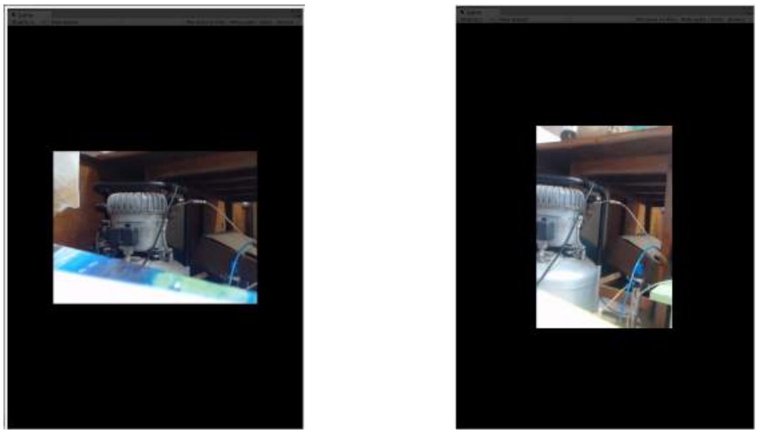



Figure 3.4: 3D Model Created from MRI (taken from Natekar and Manivannan (2015)



(a) Left eye view of world (b) Right eye view of world

Figure 3.5: The stereoscopic view of the world in the Rift

The User Interface can be used for pre-planning and training before a surgery, it has the following features

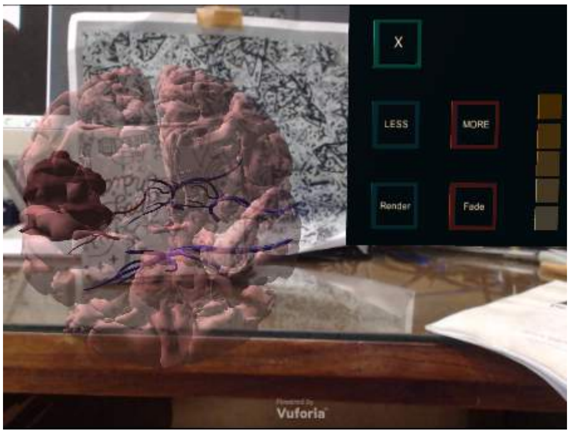
**Night Mode** - To focus only on the augmented model and not be distracted by the real world. It is enabled by a tap gesture (little finger and thumb) of the hand. 3.6

**Rotation and Zoom** - This enables local model rotation along all three axes.

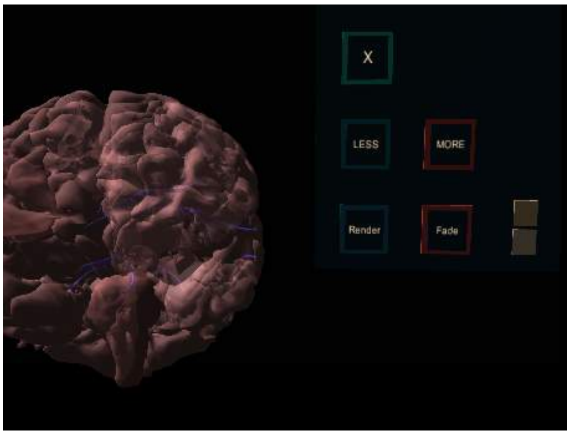
**Fade/ Render** - This function can change the opacity of the entire mode and in the planning mode helps to see inner structures such as veins, central brain parts clearly. 3.7

**Less and More** - Makes parts of the brain completely transparent. This helps in viewing for eg. only the tumour without the surrounding brain tissue.

**Patient info** - Vital patient statistics during surgery, or important medical history records can be displayed during the planning stage. The best points of entry can be annotated during the planning stage and then consulted during surgery. See 3.8.

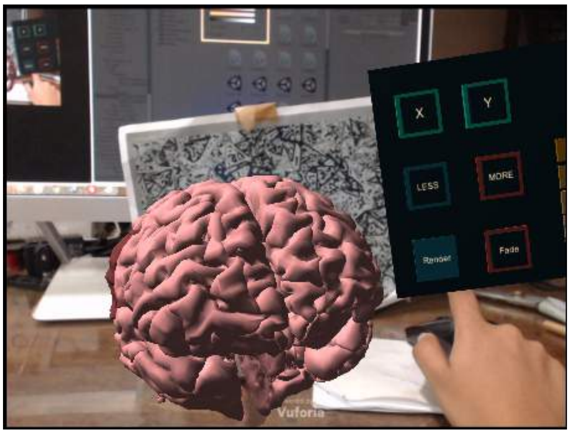


(a) Day Mode

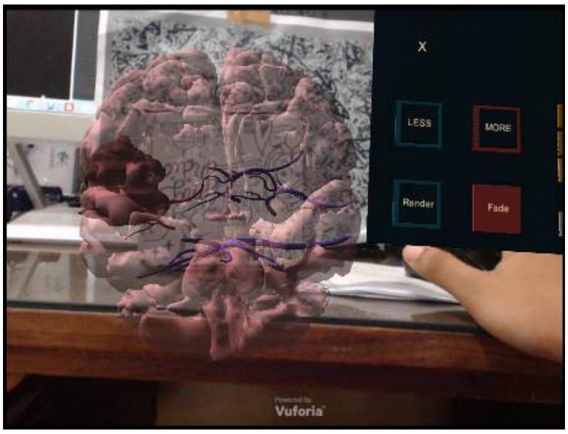


(b) Night Mode

Figure 3.6: Night Mode example




(a) Render button activated, opacity increased




(b) Fade button activated, opacity decreased

Figure 3.7: Varying Opacity for the entire model

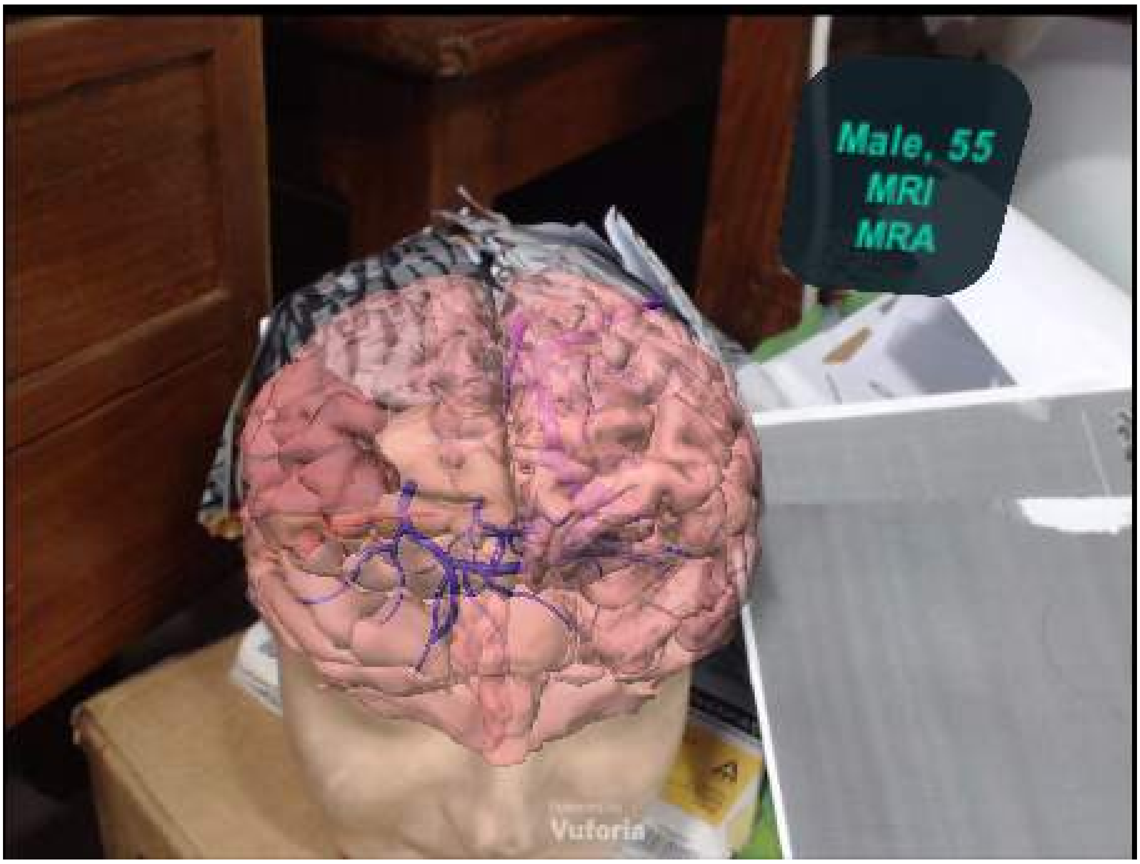



Figure 3.8: Patient info displayed along with Model

# CHAPTER 4

# Conclusions, Future Work and Limitations

As part of the thesis, the problems addressed are the following

1. Turning the Oculus Rift into an Augmented Reality Device
2. Setting up a pipeline to scan and augment any Desktop-sized 3D object with markers.
3. A User Interface to manipulate augmented content.

The present thesis thus, accomplishes one step in the project plan to create an AR-based surgical navigation and planning system. The final cost of the system implemented is under INR 50,000 (non-inclusive of the computer system).

## 4.1 Future Work

The current application would need to be validated by the doctors with regard to the accuracy of augmentation and subjective measures such as the ease of use, helpfulness of the application in planning the procedure.

A limitation of the current work is that camera feed from only one camera is being used for augmentation. Another requirement is that the resolution and FOV of the camera device needs to be higher. A solution to the lower FOV is to use a wide-angle lens and then correct for distortion. The camera resolution and Oculus resolution both need to be higher in order to obtain better levels of precision. With the current system, small text appears blurred.

An optical see-through or hybrid setup as described in Rolland and Fuchs (2000) can also be considered. Our implementation of video see-through may lack some realism as real world content is projected on screens and loses an important depth cue, stereoscopic disparity. Hybrid or optical see-through approaches can be considered for greater resolution and more realism.

The current project relies extensively on the use of visual markers for AR. This can be an impediment in some scenarios. For such scenarios, markerless AR should be investigated.

As far as the goals of the larger project are concerned, suggestions for future areas of exploration are

## 4.2 Designing for various imaging modalities

Currently this project focuses on creating a model out of MRI scans and registering it to the view from a stereo camera. However, various imaging technologies such as ultrasound, laproscopy based images are used to gain more information about the patient. Registration between a number of these would help the surgeon gain unique insight.

## 4.3 Designing for different organs

There is a wide variety in the usage scenarios even within neurosurgery. This occurs because the various parts of the nervous system face different risks and display different anatomical structures. Therefore a surgical navigation application for the brain would be different from a spinal surgery one.

## 4.4 Designing for different scenarios

Within brain surgery itself, doctors mostly operate with a microscope. Thus the headset should be designed so as to allow comfortable use with a microscope.

# APPENDIX A

## CSharp Functions Written

```
using UnityEngine;
using System.Collections;

public class OffsetQuad : MonoBehaviour
{
        public Transform BGPlane;
        public Transform Augmentation;
        private float scale;
        private Transform QuadTrans;
        private Vector3 prevScale;
        public Transform Image;
        // Use this for initialization
        void Start ()
        {



        }

        // Update is called once per frame
        void Update ()
        {
                //if a trackable is detected
                if (Augmentation.GetComponent<MeshRenderer>
                   ().enabled |
                   Image.GetComponent<MeshRenderer> ().enabled )
                {
                        //find the scale between the positions
                           of the webcam quad and bg plane
```

```
                scale = ((BGPlane.localPosition.z) /
                   (transform.localPosition.z));

                //change the parameters of the webcam
                   quad
                prevScale = transform.localScale;
                transform.localPosition =
                   BGPlane.localPosition;
                transform.localScale = prevScale *
                   scale;

            }

        }
}


using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using UnityEngine;
using VR = UnityEngine.VR;



public class SetStereoOffset_Brain : MonoBehaviour {

        public float stereoshift;
        private float actualshift;
        private Transform objTrans;
        private Vector3 LeftEyePos;
        private Vector3 RightEyePos;
        private Vector3 dir;
        public Transform CubeOrig;
```

```
public GameObject Augmentation;
public Transform Anchored;

// Use this for initialization
void Start () {
        //Initialize Object Trans
        objTrans = gameObject.GetComponent<Transform>
           ();

        //normal parameter value = -64
        actualshift = -1 * stereoshift;

        VR.VRSettings.renderScale = 3f;

}

// Update is called once per frame
void Update () {
        if (Augmentation.activeSelf)
           {//GetComponent<MeshRenderer> ().enabled) {
                //gameObject.GetComponent<MeshRenderer>
                   ().enabled = true;
                gameObject.SetActive(true);
                transform.position = CubeOrig.position;
                transform.rotation = CubeOrig.rotation;



                RightEyePos =
                   VR.InputTracking.GetLocalPosition
                   (VR.VRNode.RightEye);
                LeftEyePos =
                   VR.InputTracking.GetLocalPosition
                   (VR.VRNode.LeftEye);
```

```
                    //camera axis
                    dir = (LeftEyePos -
                       RightEyePos).normalized;

                    //transform dir to world coordinates
                    dir =
                       transform.InverseTransformDirection(dir);

                    //translate current game object
                    transform.Translate(dir*actualshift,
                       Space.Self);

            } else {
                    //
                    gameObject.SetActive(false);

            }
        }


}
```

# REFERENCES


1. **Bhayani, S. B.** and **G. L. Andriole** (2005). Three-dimensional (3d) vision: does it improve laparoscopic skills? an assessment of a 3d head-mounted visualization system. *Reviews in urology*, **7**(4), 211.

2. **Fuchs, H.**, **M. A. Livingston**, **R. Raskar**, **K. Keller**, **J. R. Crawford**, **P. Rademacher**, **S. H. Drake**, **A. A. Meyer**, *et al.*, *Augmented reality visualization for laparoscopic surgery*. Springer, 1998.

3. **Meola, A.**, **F. Cutolo**, **M. Carbone**, **F. Cagnazzo**, **M. Ferrari**, and **V. Ferrari** (2016). Augmented reality in neurosurgery: a systematic review. *Neurosurgical review*, 1–12.

4. **Mert, A.**, **K. Buehler**, **G. R. Sutherland**, **B. Tomanek**, **G. Widhalm**, **G. Kasprian**, **E. Knosp**, and **S. Wolfsberger** (2012). Brain tumor surgery with 3-dimensional surface navigation. *Neurosurgery*, **71**, ons286–ons295.

5. **Natekar, P.** and **M. Manivannan** (2015). *Affordable Immersive Virtual Reality based pre-surgical planning tool for Neurosurgery*. Ph.D. thesis, Indian Institute of Technology, Madras.

6. **Rolland, J. P.** and **H. Fuchs** (2000). Optical versus video see-through head-mounted displays in medical visualization. *Presence*, **9**(3), 287–309.