



**UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**



**APLICACIÓN DE REALIDAD VIRTUAL SOBRE REDES EN ESTADO DE
REPOSO DEL CEREBRO A PARTIR DE IMÁGENES DE RESONANCIA
MAGNÉTICA FUNCIONAL**

POR

Javiera Alejandra Baeza Acuña

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción
para optar al título profesional de Ingeniera Civil Biomédica

Profesor(es) Guía
Pamela Guevara Alvez

Comisión
Jaime Pinto Vargas
Esteban Pino Quiroga

Agosto 2022
Concepción
(Chile)

© 2022 Javiera Alejandra Baeza Acuña

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento.

Agradecimientos

En primer lugar, quiero agradecer a mis padres Gicella y Victor por estar siempre ahí para mí dándome palabras de apoyo o algún consejo, apoyándome en mis decisiones, y por sobre todo entregándome su amor incondicional, lo que me ha llevado a ser quien soy.

Agradecer a mi hermana Antonia, quien también ha sido una gran compañera de carrera durante estos casi 6 años, siendo mi complemento y apoyo en mis momentos más difíciles, lo que me ha ayudado poder llegar hasta el final.

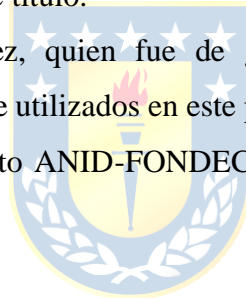
Agradecer a mi segunda familia mis amigos Sofía, Tabata, Ignacio, Sofia y Monserrat, por estar durante mi proceso de crecimiento, entregándome risas y buenos momentos cuando los necesitaba.

Agradecer a mi profesora guía Pamela Guevara, por brindarme parte de su tiempo para guiarme, confiar en mí para este proyecto, y darme consejos y palabras de apoyo cuando los necesitaba durante el desarrollo de esta memoria de título.

Agradecer a Hernán Hernández, quien fue de gran ayuda para resolver mis dudas y enseñarme sobre algunos de los software utilizados en este proyecto, siempre con buena disposición.

Finalmente agradezco al proyecto ANID-FONDECYT 1221665 y al centro ANID-BASAL FB0008 AC3E.

Muchas gracias a todos.



Resumen

El rápido avance de la tecnología ha cautivado a gran parte de la población, ya que nos permiten realizar cosas impensadas. Uno de los usos más importantes que tiene, es ocupar estos adelantos tecnológicos en los procesos de aprendizaje. Dentro de estas tecnologías que han aparecido en el último tiempo están las tecnologías inmersivas, tales como, realidad virtual o aumentada, las cuales al utilizarlas entregan múltiples beneficios al usuario y permiten a las personas realizar actividades que no se pueden hacer en el mundo real.

Es por ello que el objetivo principal de este proyecto es realizar una aplicación de realidad virtual que permita representar las redes en estado de reposo a partir de imágenes por resonancia magnética funcional (fMRI).

Para esto se utilizaron imágenes fMRI generadas por Thomas Yeo, quien realizó una parcelación de la corteza cerebral en 7 y 17 redes en el espacio de superficie de FreeSurfer y MNI 152 no lineal. Estos archivos fueron llevados a Python para separar cada una de las redes del Dr. Yeo en archivos individuales. Luego, los elementos segmentados fueron importados en 3D Slicer, el cual es un software que permite crear mallas 3D a partir de imágenes. Después de obtener los modelos de malla de cada red en estado de reposo, se importaron estos archivos en el software Blender, el cual permite segmentar y posprocesar las mallas. Al realizar todos estos pasos, se tiene una representación 3D de las 7 redes en estado de reposo del Dr. Yeo listas para implementar en un software de realidad virtual.

El desarrollo de la aplicación de realidad virtual se hizo usando el software Unity. En este programa se implementaron tres escenas: menú principal, instrucciones y principal. Esta última es en la que el usuario estará más tiempo, ya que contiene las redes en estado de reposo. En ella se puede visualizar la activación/desactivación cerebral, el usuario puede interactuar con objetos en la sala, y se entrega información importante sobre la app y las redes en estado de reposo.

La aplicación fue probada usando lentes Oculus Rift S. Se verificó el buen funcionamiento de los distintos componentes de la aplicación y la correcta visualización de las redes en estado de reposo. Lo anterior da a conocer que este software sirve para poder enseñar de una forma interactiva las redes en estado de reposo a la población.

Abstract

The rapid advance of technology has captivated a large part of the population, since it allows us to do unthinkable things. One of the most important uses is to use these technological advances in learning processes. Among these technologies that have appeared in recent times are immersive technologies, such as virtual or augmented reality, which when used deliver multiple benefits to the user and allow people to perform activities that cannot be done in the real world.

Therefore, the main objective of this project is to develop a virtual reality application to represent resting state networks from functional magnetic resonance imaging (fMRI).

For this we used fMRI images generated by Thomas Yeo, who performed a parcellation of the cerebral cortex into 7 and 17 networks in FreeSurfer and MNI 152 nonlinear surface space [1]. These files were brought into Python to separate each of Yeo's networks into individual files. Then, the segmented elements were imported into 3D Slicer, which is software that allows creating 3D meshes from images. After obtaining the mesh models of each network in a resting state, these files were imported into Blender software, which allows segmenting and post-processing the meshes. By performing all these steps, we have a 3D representation of Yeo's 7 resting state networks ready to implement in virtual reality software.

The development of the virtual reality application was done using Unity software. Three scenes were implemented in this program: main menu, instructions and main room. The latter is the one in which the user will spend more time, since it contains the networks in a resting state. In this scene the brain activation/deactivation can be visualized, the user can interact with objects in the room, and important information about the app and the networks in the resting state is provided.

The application was tested using Oculus Rift S glasses. The correct functioning of the different components of the application and the correct visualization of the networks in the resting state were verified. The aforementioned demonstrates that this software is useful to be able to teach in an interactive way the networks in a resting state to the population, thanks to all the implemented functionalities

Tabla de Contenidos

LISTA DE TABLAS	VIII
LISTA DE FIGURAS	IX
ABREVIACIONES	XI
CAPÍTULO 1. INTRODUCCIÓN	1
1.1. INTRODUCCIÓN GENERAL.....	1
1.2. OBJETIVOS.....	4
1.2.1 <i>Objetivo General</i>	4
1.2.2 <i>Objetivos Específicos</i>	4
1.3. ALCANCES Y LIMITACIONES.....	4
1.4. METODOLOGÍA	5
1.5. TEMARIO	5
CAPÍTULO 2. MARCO TEÓRICO.....	7
2.1. INTRODUCCIÓN	7
2.2. EL CEREBRO	7
2.3. fMRI.....	8
2.3.1 <i>rfMRI</i>	9
2.3.2 <i>Señal BOLD</i>	9
2.3.3 <i>Resting State</i>	10
2.3.4 <i>Default Mode Network</i>	11
2.3.5 <i>Conectividad Funcional</i>	11
2.4. REALIDAD VIRTUAL	11
2.4.1 <i>Realidad Virtual en Medicina</i>	12
2.4.2 <i>Hardware</i>	13
2.5. SOFTWARES A UTILIZAR	15
2.5.1 <i>Unity</i>	15
2.5.2 <i>3D Slicer</i>	17
2.5.3 <i>Blender</i>	18
2.6. FORMATOS DE ARCHIVOS.....	19
2.6.1 <i>Nifti</i>	19
2.6.2 <i>Archivos “.obj”</i>	20
2.6.3 <i>Archivos “.stl”</i>	20
2.6.4 <i>Archivos “.fbx”</i>	20
2.7. TRABAJOS PREVIOS.....	20
2.7.1 <i>Método Analítico: Softwares</i>	20
2.7.2 <i>Análisis de ROI</i>	21
2.7.3 <i>Análisis de Componentes Independientes (ICA)</i>	23
2.7.4 <i>Análisis con Grafos</i>	23
2.8. iBRAIN	24
2.9. DISCUSIÓN.....	25
CAPÍTULO 3. DISEÑO SOFTWARE	27
3.1. INTRODUCCIÓN	27
3.2. CONTENIDOS DEL SOFTWARE.....	27
3.3. REQUERIMIENTOS FUNCIONALES Y NO FUNCIONALES	28
3.4. ESTRUCTURA DE LA APP	28
3.4.1 <i>Diseño Escena “Menú Principal”</i>	29
3.4.2 <i>Diseño Escena “Principal”</i>	29
3.4.3 <i>Diseño Escena “Instrucciones”</i>	29
3.5. DISEÑOS MODELOS 3D.....	30
3.6. DISCUSIÓN.....	30

CAPÍTULO 4. CREACIÓN MALLADOS 3D	31
4.1. INTRODUCCIÓN	31
4.2. OBTENCIÓN DE LAS IMÁGENES.....	31
4.3. SEGMENTACIÓN DE IMÁGENES.....	31
4.4. CREACIÓN DE LAS MALLAS.....	32
4.5. OBTENCIÓN DEL MALLADO CEREBRAL.....	34
4.6. DISCUSIÓN.....	34
CAPÍTULO 5. DESARROLLO DE LOS MODELOS	36
5.1. INTRODUCCIÓN	36
5.2. SEGMENTACIÓN DE LAS REDES	36
5.3. POSPROCESAMIENTO DE LAS MALLAS	38
5.4. DISCUSIÓN.....	39
CAPÍTULO 6. DESARROLLO DE LA APP DE REALIDAD VIRTUAL	40
6.1. INTRODUCCIÓN	40
6.2. CREACIÓN DEL PROYECTO	40
6.3. CONFIGURACIÓN DEL OCULUS RIFT S.....	40
6.4. ESCENAS	41
6.4.1 Escena principal.....	42
6.4.2 Escena Menú principal.....	43
6.4.3 Escena Instrucciones.....	43
6.5. CREACIÓN DEL SHADER	43
6.6. MATERIALES DE LAS REGIONES CEREBRALES.....	45
6.7. INTERFAZ DE USUARIO	46
6.8. ANIMACIONES	47
6.9. TELETRANSPORTACIÓN	47
6.10. DISCUSIÓN.....	48
CAPÍTULO 7. RESULTADOS.....	50
7.1. INTRODUCCIÓN	50
7.2. MODELOS 3D CON SUS RESPECTIVOS MATERIALES	50
7.3. REDES EN ESTADO DE REPOSO IMPORTADAS EN LA ESCENA.....	50
7.4. APLICACIÓN RV.....	51
7.4.1 Escena principal.....	51
7.4.2 Escena Menú principal.....	52
7.4.3 Escena Instrucciones.....	53
7.5. ANIMACIONES.....	53
7.6. TELETRANSPORTACIÓN	53
7.7. DISCUSIÓN.....	54
CAPÍTULO 8. CONCLUSIONES	55
8.1. DISCUSIÓN.....	55
8.2. CONCLUSIONES.....	56
8.3. TRABAJO FUTURO.....	56
BIBLIOGRAFÍA	57
ANEXO A. TABLA RESULTADOS MALLADOS.....	62
A.1. RESULTADOS POSPROCESAMIENTOS DE LOS MALLADOS.....	62
A.2. MATERIALES ASIGNADOS A CADA UNA DE LAS REGIONES QUE COMPOEN LAS REDES EN ESTADO DE REPOSO	65
ANEXO B. IMÁGENES	67

Lista de Tablas

Tabla 3.1: Requerimientos Funcionales (RF) y Requerimientos No Funcionales (RNF) de la aplicación.	28
Tabla 5.1: Resultados posprocesamiento Red Somatomotora.....	38
Tabla 5.2: Resultados posprocesamiento de la malla cerebral	39
Tabla 6.1: Materiales asignados a cada región de la red Somatomotora.....	46
Tabla A.1: Resultados posprocesamientos DMN.....	62
Tabla A.2: Resultados posprocesamientos Red Somatomotora	62
Tabla A.3: Resultados posprocesamientos Red Visual	62
Tabla A.4: Resultados posprocesamientos Red Frontoparietal	63
Tabla A.5: Resultados posprocesamiento Red Límbica.....	63
Tabla A.6: Resultados posprocesamiento Red de Atención Dorsal	64
Tabla A.7: Resultados posprocesamientos Red de Atención Ventral	64
Tabla A.8: Materiales asignados a cada región de la DMN	65
Tabla A.9: Materiales asignados a cada región de la red de Atención Dorsal	65
Tabla A.10: Materiales asignados a cada región de la red frontoparietal.....	65
Tabla A.11: Materiales asignados a cada región de la red Límbica.....	66
Tabla A.12: Materiales asignados a cada región de la red Visual.....	66
Tabla A.13: Materiales asignados a cada región de la red de Atención Ventral.....	66



Lista de Figuras

Fig. 1.1: Historia RV	2
Fig. 2.1: División Estructural del Cerebro [43].	7
Fig. 2.2: División funcional del Cerebro [44].	8
Fig. 2.3: fMRI.....	9
Fig. 2.4: La señal BOLD	10
Fig. 2.5: Uso de aplicación de realidad virtual, usando como mando un smartphone [13].....	13
Fig. 2.6: Visor Oculus Rift S [47].	14
Fig. 2.7: Oculus Rift S 6DOF.....	15
Fig. 2.8: Controladores Oculus Touch.	15
Fig. 2.9: Interfaz de usuario de Unity.....	17
Fig. 2.10: Interfaz de usuario 3D Slicer.	18
Fig. 2.11: Interfaz de Usuario Blender.....	19
Fig. 2.12: Visualización gráfica de resultados obtenidos a partir de rfMRI utilizando el software CONN.....	21
Fig. 2.13: Resultados obtenidos a partir de fMRI haciendo uso de software GIFT	22
Fig. 2.14: Mapa de conectividad funcional de ROI.....	22
Fig. 2.15: Resultados de análisis a imágenes de rfMRI utilizando el método ICA.....	23
Fig. 2.16: Mapa de conectividad funcional del método de análisis con grafos	24
Fig. 2.17: Capturas iBraiN	25
Fig. 2.18: Visualización redes en iBraiN.....	25
Fig. 3.1: Tipos de Información mostradas en la App.	27
Fig. 3.2: Diagrama funcionamiento App.....	28
Fig. 4.1: Máscaras de Yeo.....	32
Fig. 4.2: Máscaras de Yeo separadas	32
Fig. 4.3: Herramienta Segment Editor I.	33
Fig. 4.4: Herramienta Segment Editor II.....	34
Fig. 4.5: Modelo 3D de máscara 1.	34
Fig. 4.6: Template de MNI 152 no lineal	34
Fig. 4.7: Modelo 3D del cerebro.	35
Fig. 5.1: Primer caso de segmentación en Blender.	36
Fig. 5.2: Segundo caso de segmentación en Blender.	37
Fig. 5.3: Resultados Segmentación Blender.....	37
Fig. 5.4: Aplicación del Diezmado en Blender.	38
Fig. 6.1: Componentes de XR Origin.....	41
Fig. 6.2: Configuración cascos y mandos RV	41
Fig. 6.3: PBR Graph.....	44
Fig. 6.4: Shader y fMRI.....	45
Fig. 6.5: Materiales Regiones DMN.....	45
Fig. 6.6: Modelo Manos	46
Fig. 6.7: Diagrama mandos Oculus Rift S [50].	46
Fig. 6.8: Árbol de combinación creado para la mano derecha	48
Fig. 7.1: PCC de la Default Mode Network con su material asignado.....	50
Fig. 7.2: Malla cerebro con su respectivo material en Unity.....	50
Fig. 7.3: Redes importada en la escena	51
Fig. 7.4: Resultados escena principal	52

Fig. 7.5: Resultado final Escena Menú Principal	52
Fig. 7.6: Resultado final Escena Instrucciones.....	53
Fig. 7.7: Animaciones manos virtuales	54
Fig. 7.8: Área de Teletransportación.....	54
Fig. B.1: PCC con su material asignado.....	67
Fig. B.2: Giro Temporal con su material asignado.	67
Fig. B.3: Corteza Lateral Parietal con su material asignado.	67
Fig. B.4: Hipocampo con su material asignado.....	67
Fig. B.5: Corteza prefrontal dorsolateral con su material asignado.	68
Fig. B.6: MPFC con su material asignado.....	68
Fig. B.7: Red Visual con su material asignado.	68
Fig. B.8: Red Somatomotora con su material asignado.	69
Fig. B.9: Red de Atención Dorsal con su material asignado.....	69
Fig. B.10: Red de Atención Ventral con su material asignado.	69
Fig. B.11: Red Límbica con su material asignado.	70
Fig. B.12: Red Frontoparietal con su material asignado.	70



Abreviaciones

Mayúsculas

TI: Tecnología inmersiva.

RV: Realidad virtual.

RA: Realidad aumentada.

MRI: Imágenes por resonancia magnética, *Magnetic Resonance Imaging*.

fMRI: Resonancia magnética funcional, *Functional Magnetic Resonance Imaging*.

DMN: Red neuronal por defecto, *Default Mode Network*.

BOLD: Dependiente del nivel de oxigenación de la sangre, *Blood Oxigenation Level-Dependent*.

rfMRI: Resonancia magnética funcional en estado de reposo, *Resting State Functional Magnetic Imaging*.

ICA: Análisis de Componentes Independientes, *Independent Component Analysis*.

ROI: Región de Interés, *Región of Interest*.

HCP: Proyecto conectoma humano, *Human Connectome Project*.

PCC: Corteza posterior cingulada, *Posterior Cingulate Cortex*.

ACC: Corteza anterior cingulada, *Anterior Cingulate Cortex*.

MPFC: Corteza prefrontal media, *Medial Prefrontal Cortex*.

FEF: Campos oculares frontales, *Frontal Eye Fields*.

UI: Interfaz de usuario, *User Interface*.

RF: Requerimiento Funcional.

RNF: Requerimiento No Funcional.

2D: Dos dimensiones (largo, ancho).

3D: Tres dimensiones (largo, ancho, alto).

4D: Cuatro dimensiones (largo, ancho, alto, tiempo).



Minúsculas

app: Aplicación, *Application*.

Capítulo 1. Introducción

1.1. Introducción General

“La tecnología inmersiva (TI) es una integración de contenido virtual con el entorno físico de una manera que permite al usuario interactuar naturalmente con la realidad combinada” [2], fusiona el mundo físico con una realidad que es simulada o real. Dentro de las tecnologías inmersivas las principales son: realidad virtual (RV), la cual utiliza un sistema de cascos montados en la cabeza y distintos dispositivos de entrada para proporcionar una sensación de inmersión en un mundo virtual totalmente inexistente al usuario; y la realidad aumentada (RA), la cual permite la superposición de elementos virtuales inexistentes con el entorno físico.

Los inicios de la RV datan de 1836 cuando Charles Wheatstone inventó el estereoscopio, el cual es un instrumento que permite dar la sensación de profundidad en una imagen a partir de dos fotografías. Estas imágenes son observadas de forma independiente por cada uno de nuestros ojos, generando una imagen en tres dimensiones (3D) en nuestro cerebro.

En 1929 se comenzó a utilizar la RV para recrear condiciones de vuelo reales para que soldados estadounidenses se entrenaran, este aparato es conocido como el Blue Box [3].

A mediados del 50' Morton Heiling creó “Sensorama”, el cual era una cabina de gran tamaño que combinaba distintas técnicas cinematográficas, y que tenía la intención de estimular los sentidos de la vista, olfato, tacto y oído [4].

En 1980 Thomas Furness desarrolló un simulador de vuelo llamado “Cockpit”, el cual permitía al piloto controlar un avión haciendo uso de gestos, movimientos oculares o palabras. Además, en esta cabina de entrenamiento se proyectaban mapas en 3D, datos de aeronáutica, imágenes infrarrojas y de radar en tiempo real.

La NASA en 1986 fue una de las instituciones pioneras en mostrar al público gafas de realidad virtual, el dispositivo se llamaba VIVED, contaba con visores que permitían un campo visual de 120°, incorporaba control por voz, sistema de reconocimiento de gestos por guantes, y un traje lleno de sensores para reconocer orientación espacial de la persona, movimientos y gestos.

Las décadas del 80' y 90' fue la época dorada de los videojuegos arcade. Sega en el 93' fue una de las primeras compañías en lanzar un dispositivo de RV para el público general, el cual incorporó una pantalla LCD, auriculares, y sensores para detectar el movimiento de la cabeza.

Nintendo en el '95 lanzó su primera consola llamada Virtual Boy, el cual utilizaba un joystick para jugar y unas gafas para crear un efecto 3D (monocromático).

Palmer Luckey en el 2012 materializó un proyecto interesante sobre RV "Oculus", el cual combinaba la visión de volumetría 3D en un entorno 360°. A partir de este momento se disparó el desarrollo de este tipo de TI, el cual tiene un pronóstico de crecimiento de gran nivel, ya que, empresas multinacionales como Google, HTC, Oculus VR, Samsung, Microsoft, entre otras, ya han apostado por esta gran tecnología [3], [4].

En la Fig. 1.1 se observan algunos de los dispositivos nombrados anteriormente.

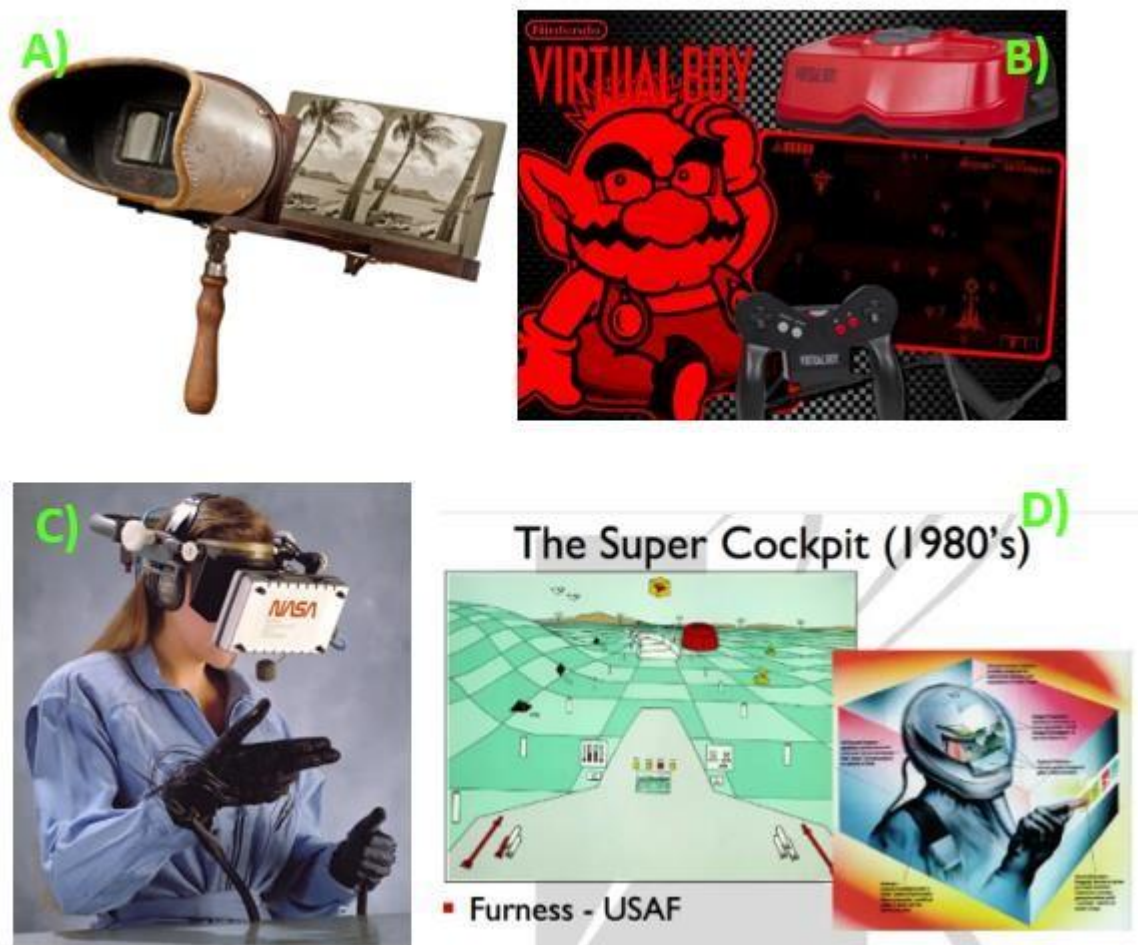


Fig. 1.1: Historia RV.

En A) se puede observar el estereoscopio de Wheatstone, el cual fue el puntapié inicial en la historia de la realidad virtual; B) Virtual Boy 1er dispositivo de RV de Nintendo; C) VIVED realidad virtual hecha por la NASA; D) Simulador de vuelo que permitía controlar un avión usando gestos, palabras o movimientos oculares [3].

Como se mencionó, la realidad virtual se ha utilizado para varias cosas entre ellas entretenimiento, educación, y la industria. En este trabajo el uso más importante de la RV es en el campo de la educación, ya que el usuario aprende con un tipo de aprendizaje llamado “inmersivo”, el cual tiene una serie de ventajas:

- Los estudiantes pueden poner en práctica sus conocimientos en un mundo ficticio aprendiendo de mejor forma, ya que retienen más la información.
- Las personas aprenden habilidades riesgosas sin estar en un ambiente estresante, como, por ejemplo, que un piloto aprenda a volar, o un médico pueda practicar cirugías complejas para prevenir complicaciones.
- Permite que los usuarios tengan acceso a experiencias o actividades a los que no pueden acceder en el mundo físico.
- Proporciona nuevas alternativas de aprendizaje.
- Da la oportunidad a las personas de aprender con los sentidos.
- Permite a los usuarios a aprender haciendo, ya que, por ejemplo, un estudiante de historia que estudia la antigua Italia, además de leer libros e investigar podría estar caminando por el Coliseo Romano.

Otro punto importante de este trabajo es cómo se comporta el cerebro humano. Durante varios años se ha investigado las funciones cerebrales ante estímulos o realizando tareas, pero hace no mucho se ha comenzado a investigar qué cosas hace nuestro cerebro mientras no hacemos nada, ya que durante muchos años en el campo de la neuroimagen se consideraba la actividad intrínseca del cerebro (a bajas frecuencias) como simple ruido.

En varias investigaciones se ha descubierto que el cerebro no descansa, ya que, aunque no estemos realizando ninguna tarea, hay una activación de ciertas redes neuronales (redes en estado de reposo) que permiten que el cerebro esté listo para funcionar ante un estímulo. Está comprobado que estas activaciones cerebrales no son al azar, estando bien organizadas y estructuradas. Estos patrones de conectividad han sido descritos gracias a imágenes de Resonancia Magnética funcional (fMRI), lo que ha permitido a los científicos aprender sobre cómo se comporta el cerebro humano mientras está funcionando.

Hoy en día se ha investigado mucho más sobre el estado de reposo y las redes que se activan mientras las personas no realizamos ninguna actividad. Esto ha permitido a la comunidad científica estudiar la actividad cerebral de niños y pacientes que no se pueden someter a experimentos largos

para analizar cómo funciona su cerebro. Hasta el momento las redes en estado de reposo más conocidas e investigadas son la Default Mode Network (DMN), la visual, la somatomotora, la de atención dorsal, la de atención ventral, la límbica y la frontoparietal [1], donde la DMN es la más estudiada. Si bien existen algunos videos y software que las representan como grafos, no se ha encontrado material educativo lúdico para que el público general pueda aprender sobre ellas.

Por lo tanto, se propone diseñar y desarrollar una aplicación de realidad virtual (RV) a partir de imágenes de resonancia magnética funcional, con el fin de enseñar de forma interactiva la DMN a la población.

1.2. Objetivos

1.2.1 Objetivo General

Desarrollar una aplicación de Realidad Virtual en Unity que represente las redes en estado de reposo a partir de datos reales de fMRI.

1.2.2 Objetivos Específicos

- Mediante procesamiento de imágenes de fMRI, obtener modelos 3D de las activaciones cerebrales de la Default Mode Network, red Visual, red Somatomotora, red de Atención Dorsal, red de Atención Ventral, red Límbica y la red Frontoparietal.
- Diseñar la aplicación, especificando objetos a mostrar, interacciones y secuencia de activación de las redes.
- Implementar la aplicación de realidad virtual, probando diferentes opciones de despliegue de la conectividad cerebral.

1.3. Alcances y Limitaciones

Se utilizará las máscaras de la parcelación a nivel de red publicada por el Dr. Thomas Yeo [1], las cuales se encuentran disponibles para su descarga en la web [5]. Estos archivos están en formato “.nii”, por lo que para su procesamiento se utilizará el lenguaje de programación Python. Se realizarán los mallados 3D de las redes en estado de reposo utilizando el software 3D Slicer. En el software Blender se hará la segmentación y posprocesamiento de las mallas de las redes en estado de reposo. Se trabajará con la versión gratuita de Unity, que permitirá desarrollar la aplicación de RV.

1.4. Metodología

El primer paso a seguir es realizar un estudio bibliográfico sobre fMRI, resting-state, default mode network, trabajos previos realizados sobre el tema, y aprender sobre Unity, 3D Slicer y Blender.

El desarrollo de este trabajo se realizará con la ayuda del lenguaje de programación Python en su versión 3.8 con el cual se procesarán las imágenes de fMRI y se obtendrán las máscaras (archivos “.nii”) de las redes en estado de reposo del Dr. Yeo de forma independiente. Luego se usará el software 3D Slicer en su versión 4.11.20210226 para convertir los volúmenes obtenidos a mallados “.obj”. Después se utilizará el software Blender en su versión 2.93.4 para segmentar y posprocesar la malla de la DMN y el resto de las redes en estado de reposo. Finalmente, se hará uso de Unity en su versión 2019.4.32f1 para el desarrollo de la app, en donde el usuario podrá moverse a través de una sala, en la cual podrá visualizar la activación de la default mode network y el resto de las redes en estado de reposo.

Lo anterior se desarrollará en un computador con un procesador Intel® Core™ i7-8700k CPU @ 3.70GHz, tarjeta gráfica NVIDIA GeForce GTX 1050 Ti y 32 GB RAM.

1.5. Temario

- Capítulo 1: En este capítulo se introduce el tema a tratar, y se definen los objetivos y la metodología a seguir para llevar a cabo este trabajo.
- Capítulo 2: Aquí se abarca el marco teórico, en el cual se realiza una revisión bibliográfica para comprender los conceptos teóricos a utilizar en este trabajo, y se muestra un compilado de los trabajos previos realizados.
- Capítulo 3: En este capítulo se mencionan los requerimientos funcionales y no funcionales de la aplicación, se explica el diseño del software, centrándose en la funcionalidad y estructura de la app, diseño de los modelos 3D, y que tipo de información se entregará en ella.
- Capítulo 4: Describe la obtención de los modelos de mallados que serán visualizados en la aplicación de realidad virtual.
- Capítulo 5: En este capítulo se explica cómo se desarrollaron los modelos 3D en Blender.
- Capítulo 6: Aquí se describe como se desarrolló la aplicación de realidad virtual en Unity.
- Capítulo 7: En este capítulo se muestran los resultados de los modelos de mallas

implementados en Unity. Además, se observan imágenes de prueba realizadas con lentes de RV.

- Capítulo 8: En este último capítulo se tiene la discusión, la conclusión y el trabajo futuro del trabajo realizado.



Capítulo 2. Marco Teórico

2.1. Introducción

A continuación, se presenta un estudio de los contenidos, conceptos, terminología, softwares a utilizar en este trabajo. Luego, se realiza una revisión de los trabajos previos, con el objetivo de estar al día en el estado del arte de las investigaciones en este campo, y las metodologías usadas por diversos investigadores.

2.2. El Cerebro

El cerebro humano es un órgano difícil de estudiar, ya que se encarga de controlar las funciones complejas del cuerpo humano. Estructuralmente se organiza en encéfalo, cerebelo y tronco cerebral, como se observa en la Fig. 2.1.

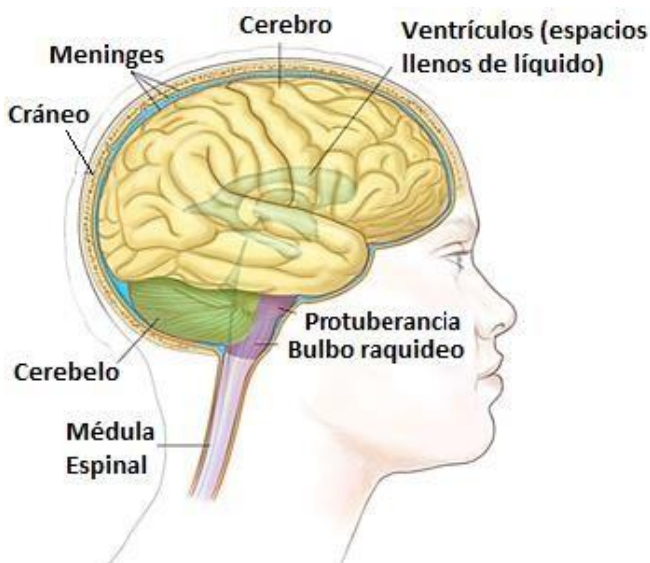


Fig. 2.1: División Estructural del Cerebro [43].

El encéfalo se encarga del control del tacto, la T°, la memoria, el pensamiento, entre otras. Este se subdivide en lóbulos frontal, parietal, occipital y temporal. Cabe destacar que es la estructura más grande que compone el cerebro.

El cerebelo ubicado en la región posterior de la cabeza, se caracteriza por tener una gran concentración de neuronas, se encarga principalmente de mantener el equilibrio, funciones motoras complejas, regula el tono muscular, entre otras. El cerebelo, al igual que el encéfalo, se divide en lóbulos, el anterior y el posterior.

El tronco cerebral consta de 3 estructuras que lo conforman, el bulbo raquídeo, la protuberancia y el mesencéfalo. Dentro de sus funciones se encuentran: regular el ritmo cardíaco y presión, la respiración, transmitir señales sensitivas y motoras, permite conectar el cerebro con la médula, y el encéfalo con el cerebelo, entre otras funciones.

Cada una de las diferentes estructuras que componen el cerebro a su vez se divide en distintas regiones que se encargan de procesar, enviar y recibir señales neuronales.

Funcionalmente el cerebro se divide en distintas áreas que se encargan de funciones específicas como el habla, la audición, visión, movimientos corporales, entre otros, como se muestra en la Fig. 2.2. Estas regiones se componen de distintas redes conectadas de forma directa e indirecta que trabajan sincrónicamente. Estos patrones de conectividad se activan cuando la persona está en estado de reposo o realizando tareas.

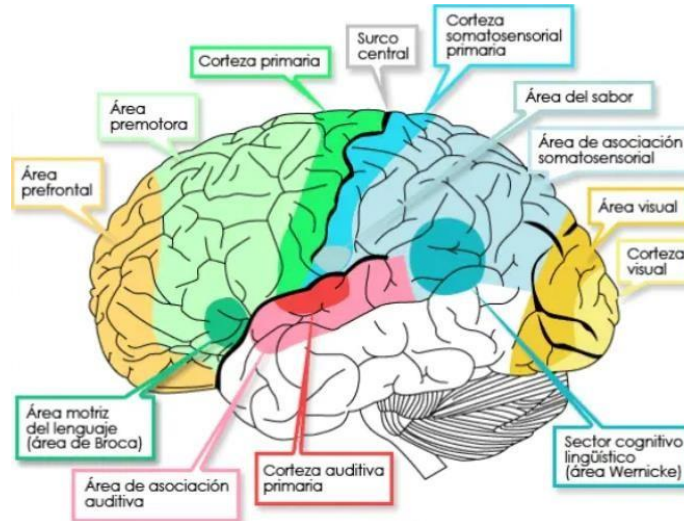


Fig. 2.2: División funcional del Cerebro [44].

2.3. fMRI

La estructura y la función del cerebro ha sido descubierta gracias al uso de distintos tipos de neuroimágenes como resonancia magnética, tomografía axial computarizada, magnetoencefalografía, resonancia magnética funcional, entre otras. La más importante para este trabajo es la fMRI, la cual es una técnica de neuroimagen no invasiva utilizada para investigar la función del cerebro y la cognición en individuos sanos y poblaciones con estados anormales del cerebro [6]. Este procedimiento se obtiene gracias a un resonador magnético, como se observa en la Fig. 2.3. La fMRI utiliza una señal de resonancia magnética para detectar cambios en el flujo de sangre, la que está relacionada directamente con la activación de las neuronas de una región específica del cerebro. Este

incremento y decremento de la señal dependiente del nivel de oxigenación de la sangre (BOLD) es la base para la creación de distintos mapas cerebrales, tanto en estado de reposo (rfMRI) como basada en tareas (fMRI).



Fig. 2.3: fMRI.

En la imagen se observa cómo los científicos miden nuestra actividad cerebral haciendo uso de un resonador magnético [45].

2.3.1 rfMRI

La resonancia magnética funcional en estado de reposo (rfMRI) permite estudiar la conectividad funcional en el cerebro adquiriendo datos de fMRI mientras los sujetos permanecen inactivos en el resonador, aprovechando el hecho de que las regiones cerebrales relacionadas funcionalmente se coactivan de forma espontánea [7]. Cabe destacar que estas activaciones de la señal BOLD se observan en baja frecuencia ($< 0.1\text{Hz}$).

2.3.2 Señal BOLD

La señal producida por los cambios dependientes del nivel de oxigenación sanguínea (BOLD) es la que se detecta por el resonador magnético. Su aumento y decremento depende de la activación cerebral, ya que, cuando se activa una región específica del cerebro, aumenta su consumo de oxígeno. Esto conlleva a un aumento en la concentración local de desoxihemoglobina, lo que gatilla un aumento del flujo sanguíneo de esa zona, aumentando la concentración de hemoglobina oxigenada y con esto aumenta la señal BOLD en esa zona específica, como se ilustra en la Fig. 2.4.

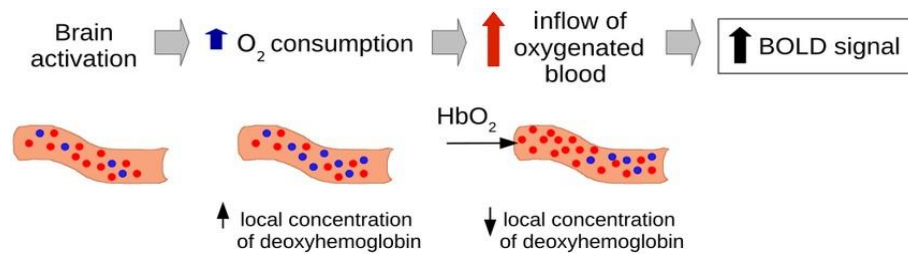


Fig. 2.4: La señal BOLD.

Detecta los cambios locales de oxigenación sanguínea mientras se produce la activación cerebral [46].

2.3.3 Resting State

El cerebro cuando no está haciendo nada nos dice mucho, ya que realmente cuando una persona no realiza ninguna tarea, ciertas regiones de su cerebro igualmente se están activando. Esto lo hace para estar preparado para funcionar ante la aparición de estímulos. A continuación, se nombran las redes que se activan durante el estado de reposo.

Red Visual: Conjunto de regiones que se encuentran en el lóbulo occipital, y se relacionan con el procesamiento visual.

Red Somatomotora: Está compuesta por dos regiones del cerebro que se encuentran delante y detrás de la fisura central, la cual separa el lóbulo frontal del parietal. Estas regiones controlan el sentido del tacto y nuestra posibilidad de poder mover nuestros músculos.

Red de Atención Dorsal: Compuesta por regiones que nos permiten mover los ojos, y procesan la información relacionada a nuestra posición y los objetos en el espacio. Se encarga de mediar la capacidad de las personas de dirigir deliberadamente o conscientemente nuestra atención al mundo.

Red de Atención Ventral: Se encarga de reorientar nuestra atención hacia estímulos destacados. De alguna manera etiqueta las cosas como sobresalientes y deja que nuestra atención sea dirigida. Se considera que esta red está involucrada en su mayoría en acciones involuntarias.

Red Límbica: Conjunto de regiones que se encargan de integrar señales de áreas emocionales de nuestro cerebro con nuestra corteza. Permite que la información emocional influya directamente en nuestra percepción, comportamiento y pensamiento.

Red Frontoparietal: Se activa cuando tratamos de pensar en algo de manera directa cada vez que intentamos resolver algún problema o razonar.

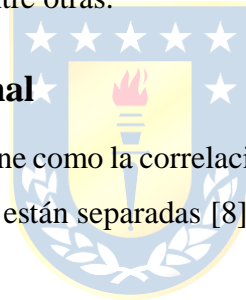
Red de Modo por Defecto: Está relacionada con el procesamiento relacionado con uno mismo, nuestra capacidad para recordar el pasado y proyectarnos hacia el futuro, nuestra capacidad de soñar despierto, y cualquier aspecto de nuestra percepción del pensamiento o comportamiento que se base en nuestros recuerdos de experiencias pasadas o conocimiento conceptual. Cabe destacar que es la red más conectada con el resto del cerebro, jugando un papel importante en la mediación de interacciones con otras redes cerebrales.

2.3.4 Default Mode Network

La red de modo por defecto es un grupo de regiones cerebrales que tienen menores niveles de activación cuando la persona se encuentra realizando tareas, y mayores niveles cuando estamos despiertos, pero en reposo. Algunas regiones que la componen son la corteza posterior cingulada, la corteza prefrontal medial, corteza parietal inferior, entre otras. Cabe destacar que varios investigadores mencionan que esta red está relacionada directamente con trastornos mentales como la esquizofrenia, la depresión, ansiedad, entre otras.

2.3.5 Conectividad Funcional

La conectividad funcional se define como la correlación temporal de los patrones de activación neuronal de regiones que espacialmente están separadas [8].



2.4. Realidad Virtual

Las Tecnologías Inmersivas integran contenido virtual con nuestro entorno físico, permitiéndonos interactuar con objetos en una realidad que puede ser simulada o real. Dentro de estos tipos de tecnologías más conocidas está la Realidad Virtual (RV). *“La realidad virtual es el término utilizado para describir un entorno tridimensional generado por computadora que una persona puede explorar e interactuar con él”* [9]. Esta persona se convierte en parte de este mundo virtual, y mientras se encuentre allí es capaz de manipular objetos o realizar un conjunto de acciones específicas.

Para que la RV funcione tal como se espera se tiene que contar con ciertos componentes elementales [10], tales como:

- ✓ *Entorno Virtual:* Es el mundo artificial compuesto por escenas y objetos creados por computadora, crea en el usuario la sensación de inmersión en él.
- ✓ *Usuario:* Es la persona que está inmersa en este entorno virtual, haciendo uso de un dispositivo llamado gafas, cascos o visores de realidad virtual.

- ✓ *La interfaz:* Permite la correcta comunicación entre el entorno virtual y el usuario, en otras palabras, traduce la información recibida por ambos.

En la actualidad existen 3 tipos de realidad virtual, y se clasifican según el nivel de inmersión que tenga el usuario en este mundo artificial [10].

Realidad virtual no inmersiva: Es el tipo más común de todos, se le dice no inmersiva dado que el usuario tiene mayor conciencia de que se trata de una realidad simulada. Se utiliza un monitor como ventana hacia este mundo virtual, la interacción se realiza gracias a un teclado, mouse, auriculares, entre otros. El uso más conocido de este tipo de RV es en el mundo de los videojuegos.

Realidad virtual semi inmersiva: Para poder lograr la experiencia semi inmersiva se tienen elementos virtuales apoyados de otros elementos físicos. Se caracteriza por tener tres pantallas o más en forma de cubo, las cuales rodean a la persona que está observando, además el usuario usa visores de RV. Este tipo de tecnología se utiliza más en el sector industrial y educación, ya que se necesita que el usuario no pierda el contacto con ciertos elementos del mundo real.

Realidad virtual inmersiva: Se le considera la más futurista de todas, y es la que ofrece al usuario la mayor sensación de inmersión. Está compuesto por una serie de dispositivos externos, tales como, un casco conectado a un computador (ocupa toda la visión del usuario), gafas, mandos, guantes de RV, entre otros. Además, posee una serie de sensores y controles de movimiento, que permiten al usuario no ser solo un simple espectador, si no, convertirse en un protagonista y decidir las acciones que quiere realizar dentro de este mundo virtual.

2.4.1 Realidad Virtual en Medicina

La realidad virtual está siendo utilizada en muchos ámbitos de nuestra sociedad, desde la industria hasta el entretenimiento, ya que su uso trae muchas ventajas, como poder visualizar casi cualquier objeto o desplazarnos a cualquier lugar. Por esta razón, también está siendo implementada en el campo de la educación. *“En la educación médica, la formación de los estudiantes mediante simulaciones es importante porque los procedimientos de diagnóstico son difíciles de aprender para los médicos”* [11], es por esto que actualmente se han introducido varios sistemas para que los estudiantes de ciencias médicas puedan visualizar imágenes anatómicas. Por ejemplo, la Anatomage Table y Lundstrom [11], que permite a los estudiantes realizar simulaciones de procedimientos médicos sin utilizar un cuerpo real.

En *“The comparison of teaching efficiency between virtual reality and traditional education in medical education: a systematic review and meta-analysis”* se indica que en los resultados de una

prueba realizada a estudiantes que fueron educados con realidad virtual y a los que recibieron educación tradicional, los primeros tienen una mayor tasa de aprobación, por lo que se concluyó en este trabajo que la enseñanza con RV mejora los resultados de aprendizaje en estudiantes de medicina [12].

Los autores de “*Virtual Reality Educational Tool for Human Anatomy*” [13] crearon un modelo tridimensional del cerebro a partir de tomografía computarizada, y para crear la experiencia virtual usaron Unity. Esta app permite al usuario tener una visita guiada por el interior y exterior del cráneo. Además, tiene un test de evaluación en donde se sitúa a la persona dentro de la cabeza y se le realiza una serie de preguntas, y finalmente se puede realizar un ensamblaje de huesos de forma interactiva. Cabe destacar que el objetivo de este trabajo fue generar una aplicación de realidad virtual que cualquier usuario pudiera ocupar. Es por esto que la aplicación fue desarrollada para los diferentes tipos de gafas y para Google Cardboard, el cual permite utilizar casi todos los smartphones como mando de RV, como se muestra en la Fig. 2.5



Fig. 2.5: Uso de aplicación de realidad virtual, usando como mando un smartphone [13].

2.4.2 Hardware

A. *Oculus Rift S*

Son unas gafas de realidad virtual con conexión para funcionar con un PC, que ofrece las mismas funciones que el Rift original, pero mejorando varias características [14]. Los lentes del Rift S se observan en la Fig. 2.6. Creados con la colaboración de Lenovo, fueron construidas sobre las bases del Rift original, añaden nueva tecnología de rastreo (Oculus Insight), la cual mejora la detección de los movimientos.

Los visores del Rift S poseen 5 sensores para seguir el movimiento, los cuales funcionan desde dentro hacia afuera, esto quiere decir que se potencia el seguimiento de la posición más cercana para mejorar la sensación de agarre y señalización de objetos.



Fig. 2.6: Visor Oculus Rift S [47].

La resolución de estas nuevas gafas es de un total de 2560x1440 píxeles, que se divide en 1280x1240 por ojo. Además, posee una pantalla LCD con una tasa de refresco de 80Hz. Los Oculus Rift S poseen un cable DisplayPort de 5 metros, una distribución mejor de su peso (ligeramente superior a 470gr), audio integrado, y se optimizó el bloqueo de la luz. Posee una característica conocida como 6DOF, se refiere a que el sistema puede seguir tanto la orientación del usuario (dirección donde mira) como la posición (donde se encuentra físicamente). El término proviene de la sigla Six Degrees of Freedom (seis grados de libertad), los cuales son la cantidad de grados de libertad que el sensor de seguimiento posee. Estos son: 3 de orientación (cabeceo, balanceo y guiñada) y 3 de posición (arriba/abajo, adelante/atrás e izquierda/derecha), como se observa en la Fig. 2.7. Los controladores que utiliza el Rift S son los Oculus Touch, como se observa en la Fig. 2.8, los cuales dan la sensación al usuario de que sus manos virtuales son las suyas, ya que posee la capacidad de identificar algunos movimientos de los dedos y replicarlos en la app de RV. Estos dispositivos se conectan de forma inalámbrica los lentes Oculus. Cabe destacar que estos controles utilizan pilas AA para que funcionen, los cuales poseen un alto consumo de batería [14], [15].

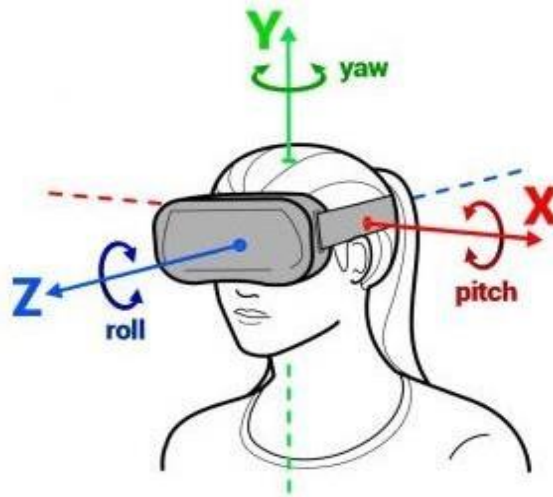


Fig. 2.7: Oculus Rift S 6DOF.

La 6DOF es una característica propia de los Rift S, el cual permite que el sensor de movimiento de los visores tenga 6 grados de libertad, los cuales son: adelante/atrás y balanceo (“roll”) en azul, izquierda/derecha y cabeceo (“pitch”) en rojo, arriba/abajo y guiñada (“yaw”) en verde [48].



Fig. 2.8: Controladores Oculus Touch.

Estos son los controladores que utiliza el Oculus Rift S para interactuar con el mundo virtual [47].

2.5. Softwares a Utilizar

2.5.1 Unity

Unity es un motor de juegos, el cual permite desarrollar videojuegos y animaciones para diversas plataformas, lo que se logra usando un editor visual y programación en C#. Usando este software se pueden lograr juegos o animaciones muy simples hasta resultados totalmente profesionales, como Pokémon Go, Monument Valley 2, Temple Run, Resident Evil: Umbrella Corps, entre otros. Una gran ventaja de este software es que se compila una sola vez y se puede publicar en celulares (Apple App Store y Google Play), dispositivos de escritorio (Windows, Mac y Linux),

consolas (PlayStation, Xbox, Nintendo Switch y Google Stadia) y en la web. Además, es la plataforma líder para crear contenido de realidad aumentada y realidad virtual. A continuación, se definirán conceptos esenciales que se utilizan en Unity:

- Asset: Es un elemento que se puede utilizar en un proyecto de Unity, pudiendo provenir de un archivo creado en Unity o fuera de este. Los creados fuera de este software pueden ser un modelo 3D, colores, texturas, una imagen, archivos de audio, o cualquier tipo de archivo que sea compatible con Unity [16].
- GameObject: “Cada objeto de un proyecto en Unity es un *GameObject*, desde los personajes y objetos, hasta las luces, cámaras y efectos especiales” [17]. No pueden hacer nada por sí mismos, ya que se les tiene que asignar propiedades antes de que puedan representar un personaje, un objeto, algún lugar o ser un efecto especial.
- Prefabs: Es un tipo de Assets reutilizable, ya que actúa como una plantilla que almacena un *GameObject* completo con todos sus componentes, formas, propiedades, entre otras características. Permite crear nuevas instancias de forma rápida en la escena [18].
- Shader: “Son Assets que contienen código e instrucciones para que la tarjeta gráfica ejecute” [19].
- Canvas: Es un *GameObject* que contiene todos los elementos de interfaz de usuario, tales como, botones, paneles, texto, imágenes, entre otros [20].

La interfaz de Unity es amigable con el usuario, está compuesta de 5 ventanas y una barra de herramientas [21], como se visualiza en la Fig. 2.9. A continuación, se definirán cada una de las partes de la interfaz de usuario:

- 1) Barra de herramientas: Contiene las herramientas básicas para manipular la vista de escena y los objetos que se encuentran dentro de ella, permitiendo al usuario rotar, escalar, moverse y arrastrar objetos. Además, contiene los botones de reproducción, pausa y pasos.
- 2) Ventana del proyecto: Permite al usuario acceder y gestionar los *assets* del proyecto, tales como, materiales, *prefabs*, modelos 3D o 2D, texturas, códigos en C#, archivos, entre otros documentos.
- 3) Ventana de Jerarquía: Contiene los *GameObjects* de la escena actual, por lo que está relacionada directamente con 5). Cada elemento en la escena tiene una entrada en la

jerarquía, ya que permite visualizar cómo los objetos están agrupados (cuáles son padres o hijos, y como depende un *object* del otro).

- 4) Ventana del Inspector: Permite observar y editar las características de un objeto seleccionado, por ejemplo, ubicación en el plano, escala, si está rotado o no, material del que está hecho, si este posee o no propiedades físicas, entre otras.
- 5) Vista de Escena: Permite al usuario visualizar y editar su escena, permitiendo mostrar un plano 2D o 3D dependiendo de la configuración inicial del proyecto.

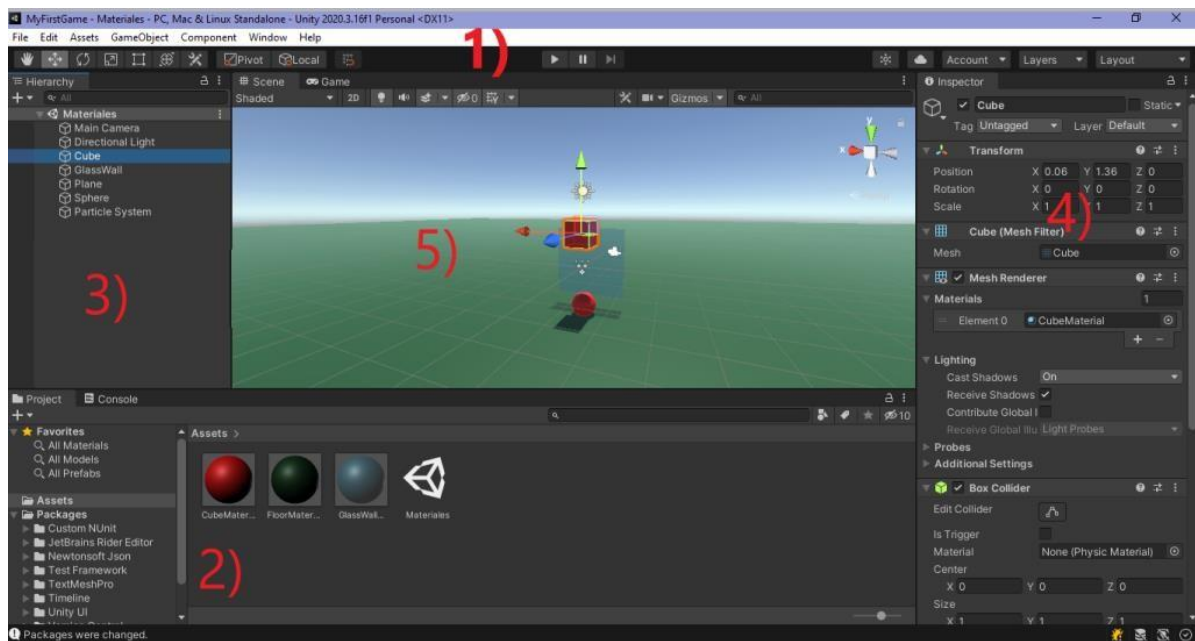


Fig. 2.9: Interfaz de usuario de Unity.

- 1) Barra de Tareas; 2) Ventana del proyecto; 3) Ventana de Jerarquía; 4) Ventana del Inspector; 5) Vista de Escena.

2.5.2 3D Slicer

Es un software gratuito, de código abierto y multiplataforma ampliamente utilizado para la investigación de imágenes médicas, biomédicas y relacionadas [22]. Se hará uso de este software, ya que las imágenes médicas que se utilizarán son de formato “.nii”. Slicer permite abrir este tipo de archivos, editarlos y guardarlos en formatos aceptados por Blender.

Dentro de la variedad de herramientas que contiene Slicer, sólo se utilizará *Volume Rendering*, *Crop Volume*, *Segment Editor*, y *Segmentations*, las cuales permitirán crear un modelo de malla a partir de imágenes Nifti. Cabe destacar que las herramientas anteriores se explicarán en más detalle en el próximo capítulo. En la Fig. 2.10 se muestra la interfaz de usuario de 3D Slicer.

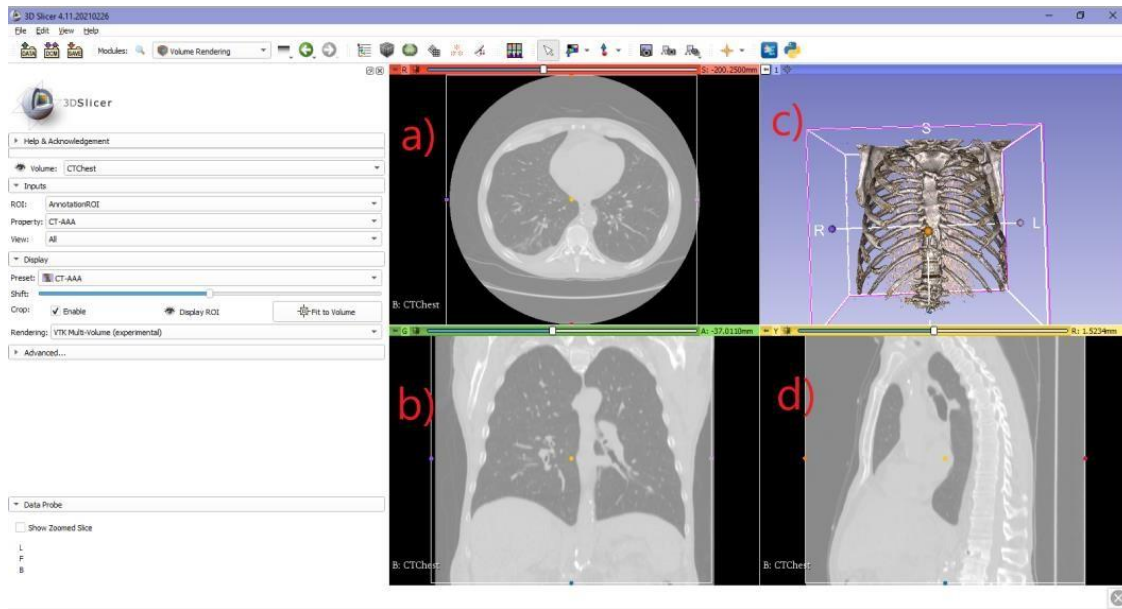


Fig. 2.10: Interfaz de usuario 3D Slicer.

a) Imagen en corte axial; b) Imagen en corte coronal; c) Imagen médica en 3D; d) Imagen en corte sagital.

2.5.3 Blender

Es un software de código abierto, dedicado al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales [23]. Es compatible con todas las versiones de Windows, macOS, GNU/Linux, Solaris, FreeBSD e IRIX. Dentro de los formatos de archivos que acepta Blender encontramos: OBJ, FBX, 3DS, PLY, STL, BLEND, entre otros.

La interfaz de usuario de Blender se divide en 3 partes:

- **Barra Superior:** Contiene distintos menús de Blender, el de archivo, de edición, renderizado, de ventana, ayuda, entre otros. Además, contiene las pestañas de escena y layout, las cuales permiten seleccionar la escena activa y la capa de visualización. Se observa de color verde en la Fig. 2.11.
- **Áreas en el centro:** Contiene la *vista 3D* la que permite entrar en el modo de edición y el modo objeto, incluye una serie de herramientas que se pueden utilizar en el objeto 3D. La *Escene collection* mantiene organizados toda la información dentro de Blender. Las *propiedades* contienen varias categorías que se pueden seleccionar a través de los iconos pequeños, permiten la edición de muchos datos activos, la escena y objetos. La *timeline* permite manipular los fotogramas para realizar la animación de la escena, y

utilizar los controles de reproducción, pausa y saltos a través de una secuencia de tiempo. Se observa de color rojo en la Fig. 2.11

- **Barra de estado:** Muestra atajos de teclado, mensajes de advertencia, información estadística, entre otros. Se observa en la Fig. 2.11 de color morado.

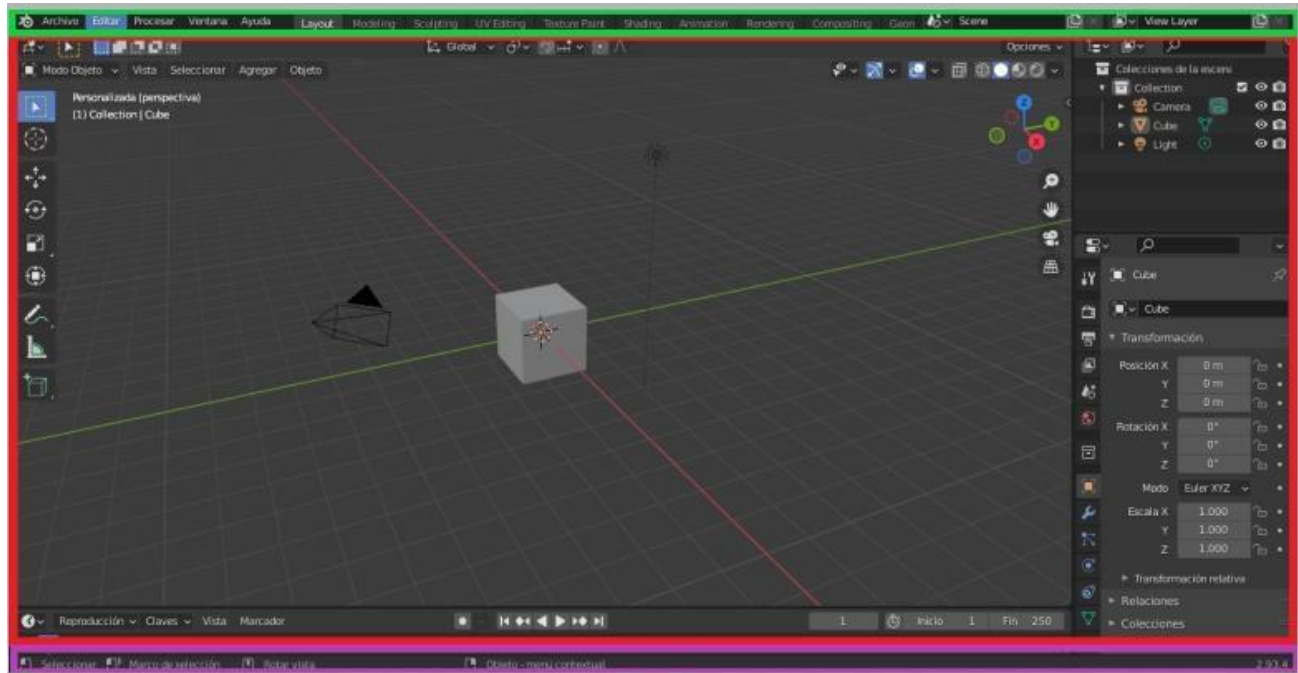


Fig. 2.11: Interfaz de Usuario Blender.

En la imagen se observan las componentes de la interfaz de usuario en Blender. El recuadro de color verde corresponde a la barra superior, el rojo son las áreas en el centro y la de color morado es la barra de estado.

2.6. Formatos de Archivos

2.6.1 Nifti

“*NIfTI (Neuroimaging Informatics Technology Initiative) es un grupo de trabajo patrocinado por el NIH (National Institutes of Health) para promover la interoperabilidad de las herramientas de software de neuroimagen funcional*” [24]. Los archivos NII son un tipo de NIfTI-1 Data Format. Contienen información tanto de la metadata la cual describe características de los datos, como el contenido, calidad, información y otros atributos [25], y una matriz de imagen que puede ser 2D, 3D o 4D. Esta dispone en filas y columnas numéricas el valor de cada píxel en el caso de dos dimensiones, y vóxeles en el caso de tres y cuatro dimensiones.

2.6.2 Archivos “.obj”

La extensión OBJ es conocida como Wave 3D Object File. Básicamente es un archivo de malla 3D, el cual permite almacenar coordenadas 3D (líneas poligonales y puntos), mapas de textura, archivos, caras poligonales, así como información de paleta de colores, y otra información de objetos.

2.6.3 Archivos “.stl”

Un archivo con extensión STL conocido como Stereolithography, es un formato nativo de la plataforma de software CAD de estereolitografía desarrollada por 3D Systems [26]. Una limitación de este formato es que contiene solo información geométrica del modelo 3D (tamaño y forma) codificada en una malla triangular, y no contiene datos de textura o color.

2.6.4 Archivos “.fbx”

La extensión FBX es un tipo de archivo que permite almacenar datos de las escenas, cámara y la geometría 2D y 3D, y animaciones. Una de sus grandes ventajas es que es compatible con varios software como Blender, Autodesk, Paint 3D, entre otros. Usualmente este tipo de archivos es utilizado para desarrollar juegos, RV, RA y películas [27].

2.7. Trabajos Previos

A continuación, se darán a conocer trabajos previos relacionados con métodos y softwares para realizar el análisis de resonancia magnética funcional en estado de reposo.

2.7.1 Método Analítico: Softwares

En el trabajo “Resting-State Functional MRI: Everything That Nonexperts Have Always Wanted to Know” [28], se describe una variedad de software que sirven para analizar las imágenes de rfMRI, de los cuales los más utilizados son:

- AFNI: Análisis de neuroimágenes funcionales, desarrollado principalmente para el análisis y visualización de múltiples modalidades de MRI: datos anatómicos, de MRI funcional y ponderada en difusión [29].
- CONN: Es un software multiplataforma de código abierto basado en Matlab/SPM para el cálculo, la visualización y el análisis de imágenes de resonancia magnética de conectividad funcional. Se utiliza para analizar datos de estado de reposo, así como

fMRI basada en tareas [30]. En la Fig. 2.12 se observa como CONN entrega de forma gráfica algunos resultados.

- MELODIC: Multivariate Exploratory Linear Optimized Decomposition into Independent Components utiliza el análisis de componentes independientes para descomponer uno o varios conjuntos de datos 4D en diferentes componentes espaciales y temporales [31].
- GIFT: Group ICA of fMRI Toolbox es una caja de herramienta de Matlab que implementa múltiples algoritmos para el análisis de componentes independientes y la separación ciega de fuentes de datos de imágenes de resonancia magnética funcional de un grupo y de un solo sujeto [32]. En la Fig. 2.13 se observa cómo se muestran los resultados obtenidos en GIFT.

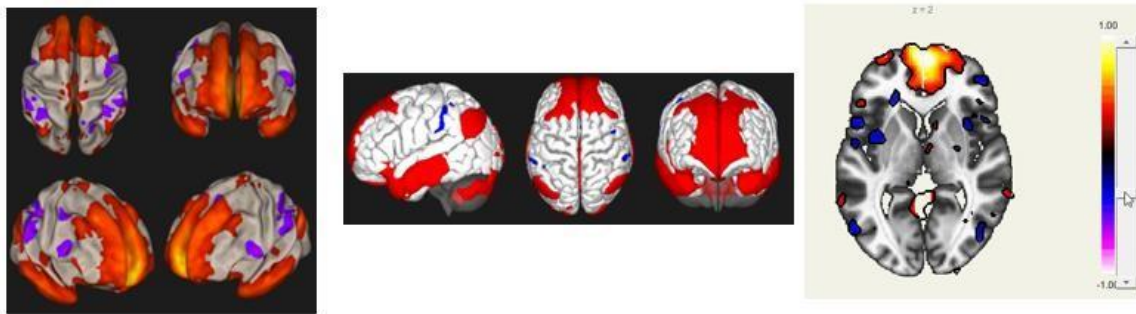


Fig. 2.12: Visualización gráfica de resultados obtenidos a partir de rfMRI utilizando el software CONN.

En las imágenes se observan las zonas que pertenecen a la DMN, obtenida a partir de una semilla en la corteza prefrontal medial (MPFC). Las imágenes de la izquierda y centro representan la DMN en 3D, mientras que en la derecha se representa en 2D y se va visualizando la región a través de cortes. Cabe destacar que las zonas de color más rojizo indican que ahí existe una mayor activación de la señal BOLD.

2.7.2 Análisis de ROI

El análisis de regiones de interés (ROI) es uno de los más utilizados para analizar conectividad funcional en estado de reposo [33]. Es un método basado en hipótesis, el cual relaciona regiones cerebrales con la actividad de la región semilla realizando una correlación cruzada entre las series temporales de la semilla y el resto del cerebro [28]. Como resultado se obtiene un mapa de conectividad funcional con todas aquellas regiones que tienen una alta correlación con la semilla seleccionada, como se muestra en la Fig. 2.14. La gran ventaja de este método es que el cálculo es sencillo y rápido, y la interpretación de los resultados se da de forma intuitiva. Una desventaja es que

el mapa obtenido depende de la ROI seleccionada, por lo que se tiene que escoger con cuidado la región semilla para evitar el sesgo.

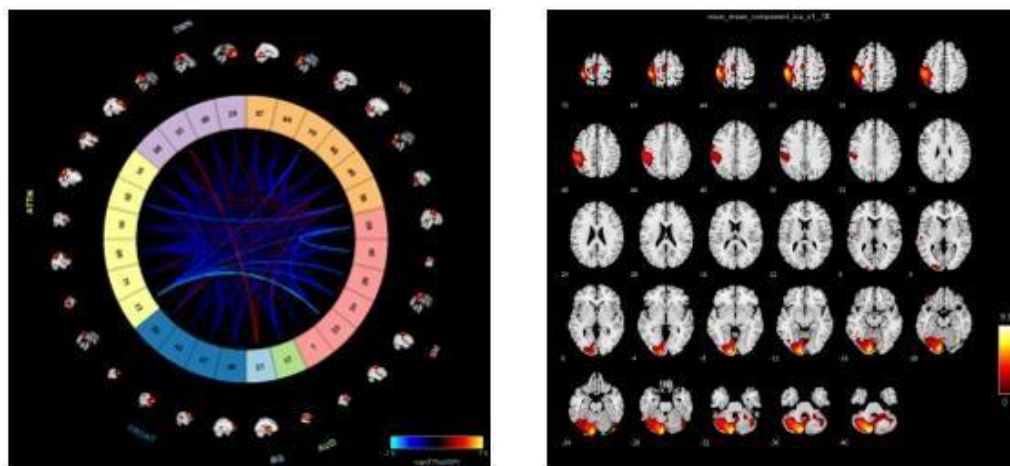


Fig. 2.13: Resultados obtenidos a partir de fMRI haciendo uso de software GIFT.

En la imagen de la izquierda se observa un conectoma del cerebro, el cual representa de forma gráfica las conexiones cerebrales entre las distintas áreas de la corteza cerebral. En la imagen de la derecha se observa un montaje con 29 cortes de un sujeto, que muestra los componentes del cerebro que pertenecen a una red en específico [32].

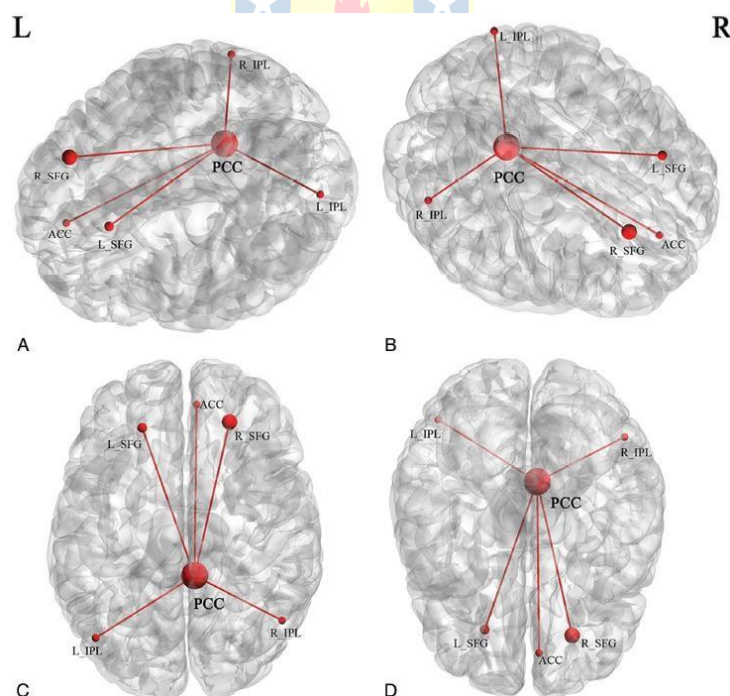


Fig. 2.14: Mapa de conectividad funcional de ROI.

En la imagen se observa que la región semilla es la corteza posterior cingulada (PCC) y están conectadas con la ROI las regiones de la corteza anterior cingulada (ACC), el gyrus frontal superior (SFG) y el lóbulo parietal inferior (IPL). Los nodos representan áreas del cerebro; las líneas conectividad funcional entre nodos.

a) Vista izquierda; b) Vista Derecha; c) Vista Superior; d) Vista Inferior [28].

2.7.3 Análisis de Componentes Independientes (ICA)

Este tipo de análisis emplea algoritmos matemáticos que descomponen la señal BOLD de los vóxeles del cerebro en sus series independientes espacial y temporal [34]. Cada red funcional (componente) incorpora un grupo independiente de neuronas que tienen actividad BOLD sincronizada. Cada componente se presenta como un mapa espacial que deriva de la correlación entre las series temporales de cada vóxel y el promedio de la serie temporal de toda la red cerebral [28].

De este análisis surgieron varias redes en estado de reposo conocidas actualmente, tales como la DMN, la red auditiva, la red visual, la red sensomotora, entre otras. El ICA se basa en los datos, a diferencia del análisis basado en semillas que se basa en ROIs elegidos por el investigador.

Las desventajas de este método es que los mapas generados son más complejos de explicar y analizar, y no se puede visualizar más de una red al mismo tiempo, como se muestra en la Fig. 2.15.

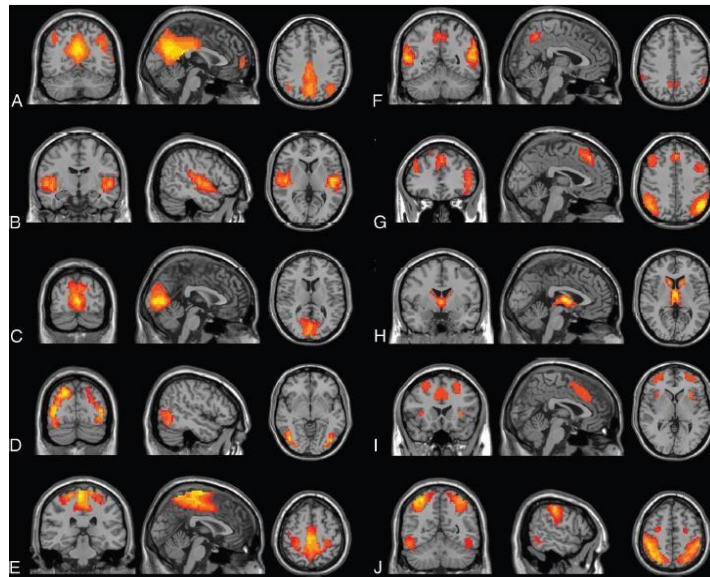


Fig. 2.15: Resultados de análisis a imágenes de rfMRI utilizando el método ICA.

Se observan: a) Default Mode Network; b) Red auditiva; c) Red visual medial; d) Red visual lateral; e) Red sensomotora; f) Red precuneus; g) Red de atención frontoparietal; h) Red de ganglios basales; i) Red de control ejecutivo; j) Red visoespacial [28].

2.7.4 Análisis con Grafos

El análisis con grafos entrega el marco teórico para analizar las redes cerebrales examinando su topología global y local [28]. Las redes neuronales son analizadas como grafos, en donde los nodos representan vóxeles y las aristas conexiones funcionales. En este método se tiene que utilizar algún tipo de atlas para dividir el cerebro en regiones, por ejemplo, el Automated Anatomical Labeling

(AAL) [35]. Una vez dividido el cerebro en distintas áreas se calcula la conectividad funcional entre 2 nodos mediante la correlación cruzada entre sus series temporales. Los resultados de este tipo de análisis se observan en la Fig. 2.16.

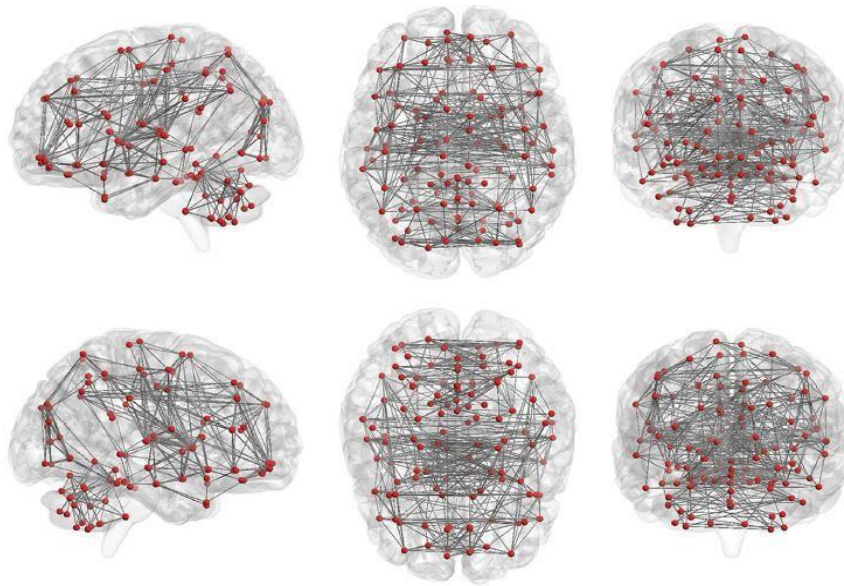


Fig. 2.16: Mapa de conectividad funcional del método de análisis con grafos.

Se utilizó el atlas AAL para dividir el cerebro en 116 regiones, 45 en cada hemisferio y 26 en el cerebelo. En rojo se observa cada nodo, y la línea negra representa la conectividad funcional entre dos nodos [28].

2.8. iBraiN

iBraiN es una aplicación móvil que permite visualizar las redes de conectividad funcional del Dr. Yeo. Este último realizó una parcelación a nivel de red y determinó que hay al menos 7 redes en un cerebro humano sano [1]. Los creadores de esta app crearon un mallado de la corteza cerebral a partir de una plantilla estructural promedio, y utilizaron las máscaras de parcelación a nivel de red publicadas por el Dr. Yeo [5] para crear mallados de las 7 redes funcionales cerebrales. Con lo anterior construyeron un modelo de mallas del cerebro completo [36], para luego crear una aplicación móvil que tiene fines académicos y de investigación. A continuación, en la Fig. 2.17 y Fig. 2.18 se muestran un par de capturas de la app.

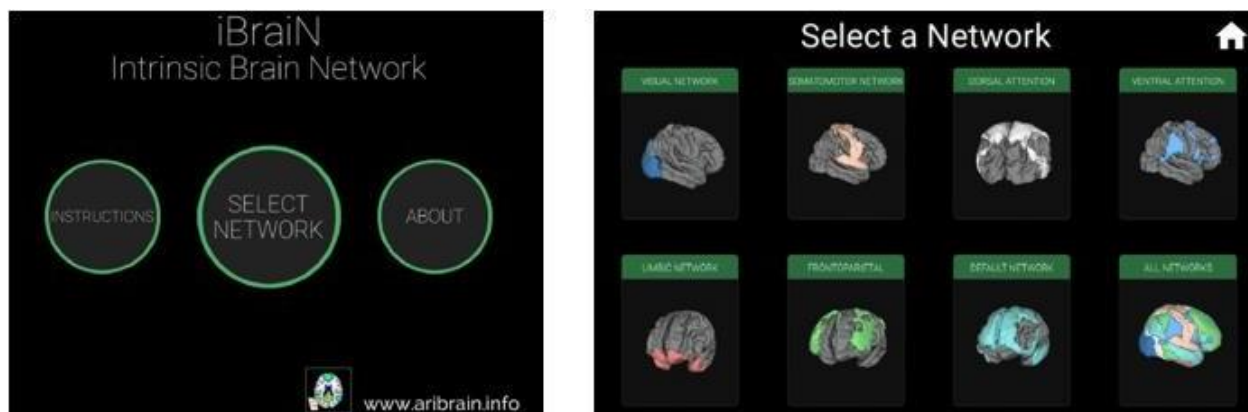


Fig. 2.17: Capturas iBrain.

En la imagen de la izquierda se observa el menú de inicio de iBrain en donde el usuario puede acceder a instrucciones, saber más sobre la app, y seleccionar que red de conectividad funcional desea visualizar. En la de la derecha el usuario tiene que seleccionar una de las 7 redes o elegir la opción de visualizar todas las redes en conjunto [36].



Fig. 2.18: Visualización redes en iBrain.

En la imagen se observa la interfaz de usuario (UI) de la app al seleccionar una red en específico, en donde se muestra el cerebro semitransparente y la red seleccionada de un color (imagen de la derecha), o la UI en el caso de seleccionar todas las redes juntas (imagen de la izquierda), donde la persona podrá navegar por el cerebro y dar click en algún color para más información sobre esa red [36].

2.9. Discusión

Unity es un software versátil, multiplataforma y de fácil uso, que cuenta con todas las herramientas necesarias que un motor de videojuego debe tener. Además, permite la compatibilidad

con una variedad de dispositivos de realidad virtual y el correcto funcionamiento de sus accesorios, entre ellos el Oculus Rift S, el cual será utilizado en este proyecto.

De acuerdo a la revisión bibliográfica, actualmente solo existen softwares y videos que representan las redes en estado de reposo como grafos, y ninguno la muestra de forma lúdica y educativa para el público general. Una excepción es iBraiN, que mostraba las redes en estado de reposo en una aplicación móvil, pero actualmente ya no está disponible para su descarga. Cabe destacar que en la sección de realidad virtual en medicina se menciona que los estudiantes tienen mejores resultados de aprendizaje cuando utilizan aplicaciones de realidad virtual, ya que no solo aprenden escuchando o leyendo, sino que tocando y visualizando estructuras que antes no se podían ver. También se mantienen más concentrados y adquieren un mayor conocimiento en comparación al aprendizaje tradicional. Es por esto que se trabajará en la creación de una app de realidad virtual en Unity que permita mostrar la DMN y las otras redes en estado de reposo de forma interactiva a partir de las máscaras del Dr. Thomas Yeo.



Capítulo 3. Diseño Software

3.1. Introducción

En este capítulo se presentan los requerimientos funcionales y no funcionales de la aplicación, las principales consideraciones de diseño de la app de realidad virtual, definiendo los contenidos e información que se presentará en ella, los que se relacionan directamente con las redes en estado de reposo.

3.2. Contenidos del Software

Este software contendrá 3 tipos de información, los cuales se mostrarán en forma de texto (en paneles informativos) o se mencionarán en videos. A continuación, se enlista la información.

- Información sobre la app: Aquí se mencionará el motivo de creación de esta aplicación de RV, de donde se obtuvieron los datos de fMRI, que es el resting state, que son las redes en estado de reposo, y cuales se enseñarán.
- Paneles informativos: En estos paneles se explicará una breve reseña sobre cada una de las 7 redes en estado de reposo a enseñar.
- Tótems Informativos: Estos contendrán las regiones que componen cada red en estado de reposo, y una etiqueta de color que representará cada región en la malla 3D.

Esta información será enseñada mediante modelos 3D de las redes en estado de reposo, componentes de interfaz de usuario, videos en la app, como se visualiza en la Fig. 3.1.

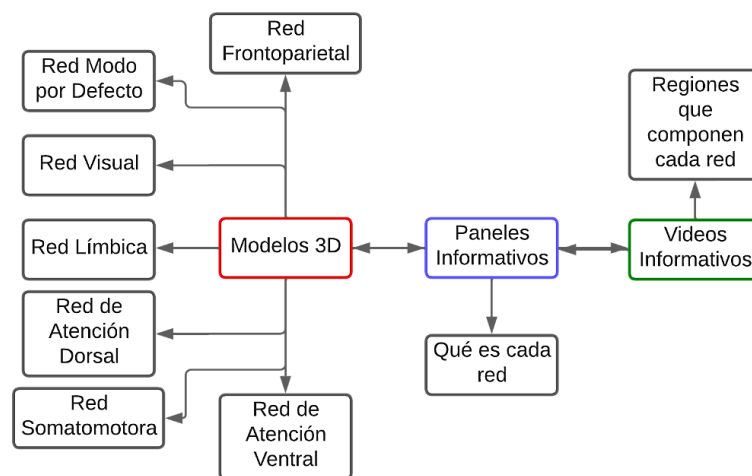


Fig. 3.1: Tipos de Información mostradas en la App.

En este software existirán 3 tipos de información mostradas al usuario, la enseñada a través de modelos 3D (en rojo), la mencionada en paneles informativos (en azul), y la mostrada en videos (en verde).

3.3. Requerimientos Funcionales y No Funcionales

En la Tabla 3.1 se encuentran los requerimientos funcionales (RF) y los requerimientos no funcionales (RNF) que se consideraron en esta aplicación. Los RF declaran los servicios que entregará el sistema, informan la manera en cómo éste reaccionará a entradas particulares, y en algunos casos declara lo que el software no debe hacer[37]. Los RNF son los “que no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de este” [37].

Tabla 3.1: Requerimientos Funcionales (RF) y Requerimientos No Funcionales (RNF) de la aplicación.

Requerimiento	Descripción Requerimiento
RF1	La aplicación debe tener un menú principal, el cual permita al usuario acceder a la sala de instrucciones, y a la sala principal en donde se apreciarán los modelos 3D de las redes en estado de reposo y el cerebro, videos y audios informativos, la activación cerebral y la opción salir de la app.
RF2	Debe tener una sala de instrucciones en la cual se le enseña al usuario a utilizar los controles del Oculus Rift S.
RF3	La aplicación debe representar la activación/desactivación cerebral de las redes en estado de reposo.
RF4	La aplicación debe permitir el desplazamiento del usuario al utilizar los lentes Oculus Rift S.
RF5	Debe tener audios, videos y textos informativos, para que la persona aprenda de forma lúdica.
RF6	En las escenas “Redes” e “Instrucciones” tiene que aparecer alguna opción que permita al usuario volver al menú principal.
RNF1	La aplicación debe ser fácil de usar para cualquiera persona que tenga acceso a lentes de realidad virtual.
RNF2	La interfaz de usuario debe ser amigable.
RNF3	La aplicación necesita de un dispositivo de realidad virtual para funcionar.
RNF4	La aplicación está configurada para que funcione con cualquier lente Oculus.

3.4. Estructura de la App

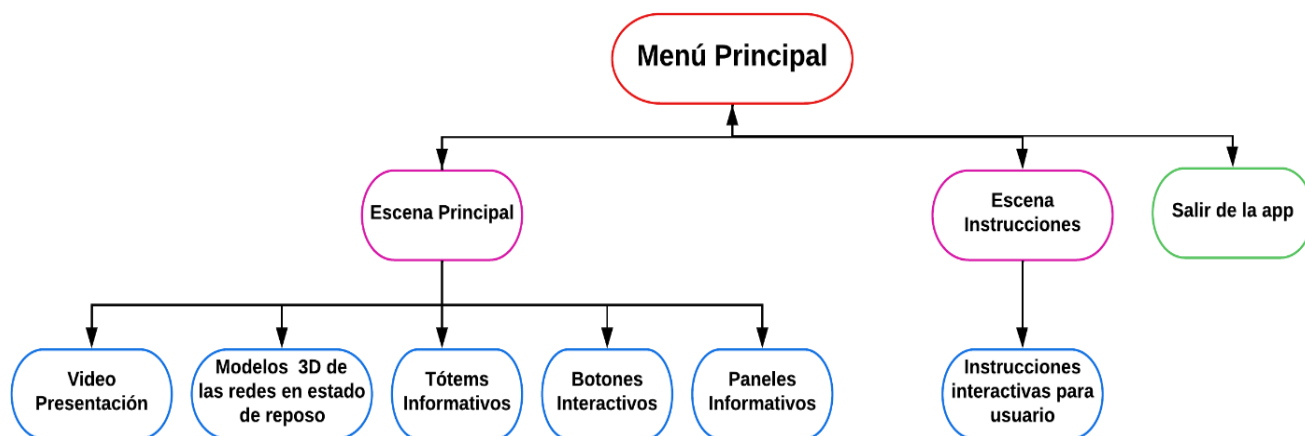


Fig. 3.2: Diagrama funcionamiento App.

La estructura de funcionamiento de esta aplicación de realidad virtual, se observa en la Fig. 3.2. En el diagrama se observa que el usuario iniciará en la escena “Menú principal”, en donde verá un mensaje en un panel informativo. Desde aquí la persona puede dirigirse a “Principal” e “Instrucciones”. Redes es la escena principal en la cual el usuario podrá observar distintos videos informativos, realizar interacciones con objetos, y aprender sobre las redes en estado de reposo. En Instrucciones se le enseña a la persona como utilizar los controles de RV dentro de la app, para poder desplazarse por el lugar e interactuar con distintos objetos virtuales.

Cabe destacar, que desde ambas escenas el usuario podrá volver al Menú Principal para ir nuevamente a “Redes”, “Instrucciones” o salir de la aplicación de realidad virtual.

A continuación, se mencionará en mayor profundidad el diseño de cada escena.

3.4.1 Diseño Escena “Menú Principal”

Esta es la primera escena con la que se encuentra el usuario. Su objetivo es guiarlo para que la persona vaya recorriendo de forma ordenada las otras salas. Para el diseño de esta escena se considerará un componente de interfaz de usuario (UI) con fondo negro, que contendrá 3 botones:

- Botón “Principal”: Permite al usuario ir a la escena principal, en la cual se podrá aprender de forma interactiva sobre las 7 redes en estado de reposo.
- Botón “Instrucciones”: Lleva al usuario a la sala instrucciones, para aprender a utilizar los controles de la app.
- Botón “Salir”: Permite salir de la app.

3.4.2 Diseño Escena “Principal”

Esta escena será la principal, la cual contendrá las mallas de las redes en estado de reposo distribuidas como un museo (la malla sobre una plataforma con buena iluminación), tótems con información alrededor de cada red entregada al usuario en forma de video o texto. En esta sala se representará cada una de las redes en estado de reposo en 3D, para que se entienda su estructura tridimensional.

3.4.3 Diseño Escena “Instrucciones”

En esta sala se le enseñará al usuario cómo ocupar los controles del Oculus en la app para desplazarse e interactuar con objetos. Para esto la escena contendrá un panel con instrucciones, y una mesa con una esfera y un cubo para practicar la interacción objeto/usuario.

3.5. Diseños Modelos 3D

En este software se tendrán 8 modelos 3D, 7 que representarán cada una de las redes en estado de reposo descubiertas por el Dr. Thomas Yeo, y uno que representará el cerebro. A continuación, se nombrará qué modelos se representarán y que contendrán.

- **Modelo redes en estado de reposo:** Representarán la forma de cada red en estado de reposo descubiertas por el Dr. Yeo.
 - ✓ Default Mode Network
 - ✓ Red Somatomotora
 - ✓ Red Límbica
 - ✓ Red de Atención Ventral
 - ✓ Red Visual
 - ✓ Red de Atención Dorsal
 - ✓ Red Frontoparietal
- **Modelo Cerebro:** Representará la forma de la corteza cerebral.

3.6. Discusión

En este capítulo se explicó cómo se diseñó esta aplicación de realidad virtual. Se detalló a través de un diagrama cómo funcionará la app, y que tipo de información se le entregará al usuario.

El diseño de la interfaz de usuario se hizo pensando para que fuera intuitiva y fácil de usar para las personas.

Cabe mencionar que el diagrama de funcionamiento de la app era más acotado, ya que, en un inicio se pensó enseñar a la población solo la Default Mode Network, pero a medida que se fue avanzando en el proyecto, se tomó la decisión de agregar las otras 6 redes del Dr. Yeo, con el objetivo de enseñar a las personas sobre todas las redes en estado de reposo como si estuvieran en un museo.

Capítulo 4. Creación Mallados 3D

4.1. Introducción

En este capítulo, se describen los pasos implementados para obtener mallas en 3D a partir de las máscaras del Dr. Yeo.

4.2. Obtención de las imágenes

El Dr. Thomas Yeo [1] realizó una parcelación cortical estimada por conectividad funcional intrínseca, utilizando datos de secuencias funcionales y estructurales de 1000 jóvenes sanos de la base de datos del Human Connectome Project (HCP) [38]. Esta información fue adquirida por escáneres Siemens 3T. Como resultados se obtuvo la parcelación de la corteza cerebral en 7 y 17 redes en el espacio de superficie de FreeSurfer y MNI 152 no lineal. Estas están disponibles para su descarga en [2]. Cabe destacar que para el desarrollo de este trabajo se utilizará el archivo “*Yeo2011_7Networks_MNI152_FreeSurferConformedImm_LiberalMask.nii.gz*” de la parcelación en 7 redes del Dr. Yeo, la cual es una máscara que contiene 8 etiquetas, que corresponden a cada una de las 7 redes.

4.3. Segmentación de imágenes

Para realizar la segmentación de las máscaras de Thomas Yeo se hizo uso de Python, en específico las bibliotecas de NiBabel, Matplotlib y Numpy, y la biblioteca *visualtools.multi_slice_viewer* para visualizar los distintos cortes de la imagen (desarrollada durante los cursos de imágenes de la carrera). Lo primero que se realizó fue cargar el archivo. Se obtuvieron sus dimensiones (256, 256, 256, 1) y sus intensidades máxima y mínima (7 y 0). Como la imagen Nifti era una matriz de 4 componentes, se le realizó un reshape para que esta quedara con una dimensión de (256, 256, 256) y pudiera ser visualizada usando *vt.multi_slice_viewer*, como se muestra en la Fig. 4.1. Luego se crearon 14 variables, siete variables llamadas “*mascX*”, las cuales son máscaras que contienen ceros y unos, donde la distribución de estos valores depende de la etiqueta de Yeo asignada (de 1 a 7). Otras siete variables las “*Yeo_X*” corresponden a la multiplicación de “*mascX*” con la imagen original. Los resultados se observan en la Fig. 4.2. Después se crearon 7 imágenes Nifti con la función *nib.nifti1.Nifti1Image(datos_imagen, imagen_afin, header)*, y finalmente se guardaron estos archivos haciendo uso de *nib.save()*.

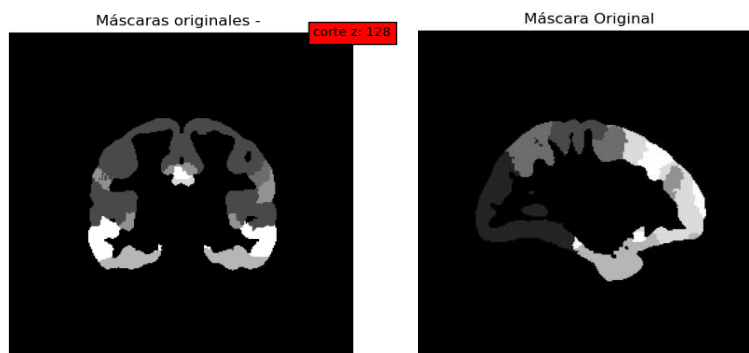


Fig. 4.1: Máscaras de Yeo.

La imagen contiene 8 valores, el 0 corresponde al fondo y del 1-7 representa a cada una de las 7 redes del Dr. Yeo.

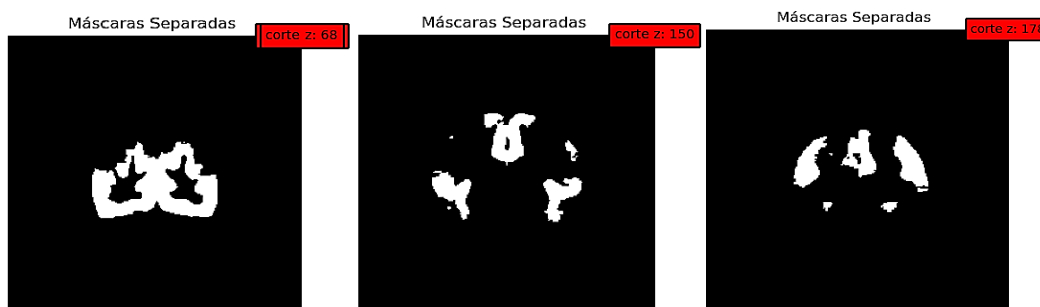


Fig. 4.2: Máscaras de Yeo separadas.

En la imagen se observan la máscara de Yeo separada por redes. A la izquierda se observa la red 1 corte coronal 68, al centro la red 4 corte coronal 150, y a la derecha red 6 corte coronal 178.

4.4. Creación de las mallas

En el punto anterior se obtuvieron archivos Nifti, los cuales están compuestos por información de metadata y una matriz 2D, 3D o 4D que contiene los valores para cada pixel o vóxel. Este tipo de datos no es aceptado por ningún software que permita modelar en 3D. Por esta razón se utilizó 3D Slicer, el cual permite transformar archivos Nifti a un formato aceptado por estos software de modelado 3D. Los pasos realizados en 3D Slicer se describen a continuación:

- Abrir archivos: Para abrir un archivo en 3D Slicer se tiene que apretar el botón *Load Data*. Se abrirá una ventana que permite cargar archivos individuales o un conjunto de datos. 3D Slicer acepta imágenes DICOM, NRRD, VTK, Nifti, PNG, GIPL, ultrasonidos, mallas superficiales o volumétricas, entre otros.
- Segment Editor: Permite seleccionar la región de interés. Esto se logra creando una etiqueta de color, llamada “Red” como se observa en la Fig. 4.3. Luego, se usó la herramienta *Threshold*, la cual permite realizar la segmentación usando distintos umbrales. Se elige un rango de umbral el cual pertenece a la malla, en este caso

cualquier valor mayor a 0, ya que los archivos obtenidos en el punto anterior son imágenes con valores 0 y 1. Con este paso quedará seleccionada solo la región de interés, como se observa en la Fig. 4.4.

- Segmentations: Se utilizó esta herramienta para segmentar solo el volumen perteneciente a la región de interés, y poder exportar el modelo a un archivo STL, OBJ, VTK, entre otros.
- Save: Permite guardar la malla creada, se elige el nombre, el formato y dónde se guardará el archivo.

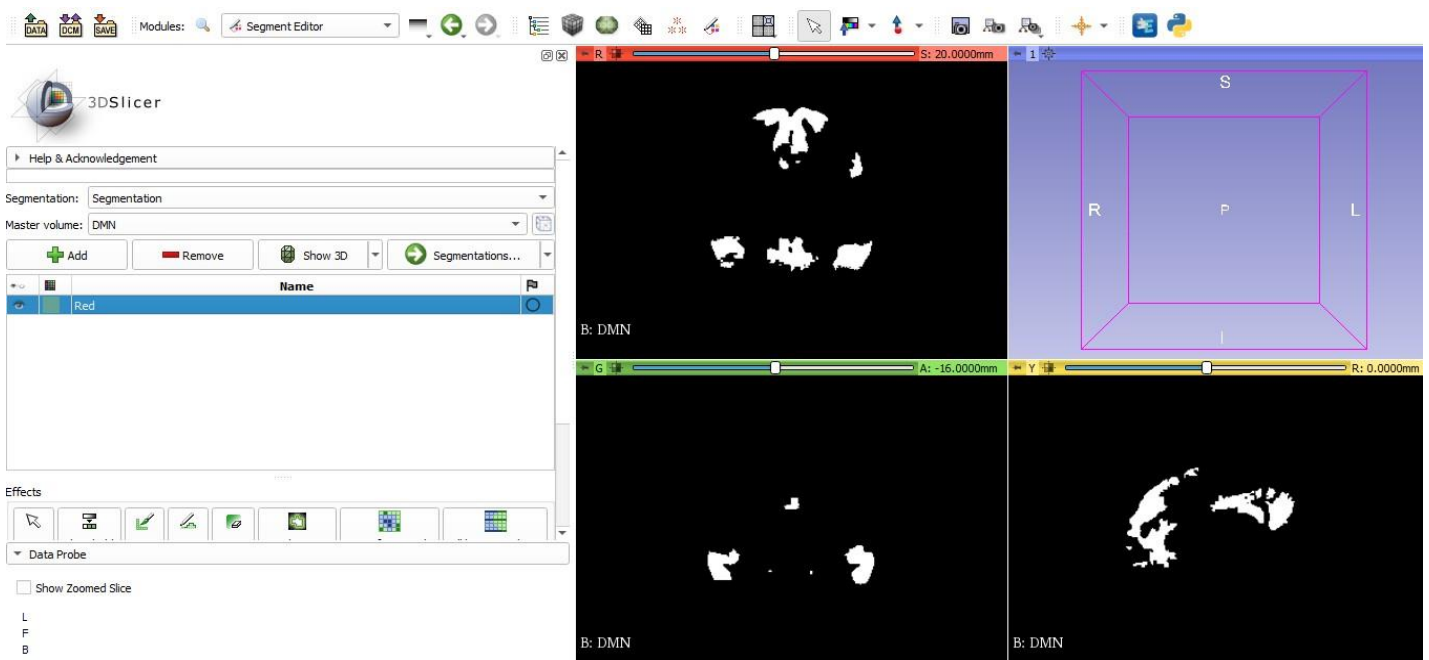


Fig. 4.3: Herramienta Segment Editor I.

En la imagen se observa la creación de la etiqueta de color “Red”, con la cual se etiquetará la región de interés.

Los pasos anteriores se realizaron en las 7 máscaras obtenidas en el punto anterior. Como resultado final se obtuvieron 7 modelos “.obj”, los cuales son archivos que contienen coordenadas 3D (líneas poligonales y puntos), mapas de textura, archivos, caras poligonales, así como información de paleta de colores, y otras características de objetos. En la Fig. 4.5 se observa el archivo “.obj” de la máscara 1.

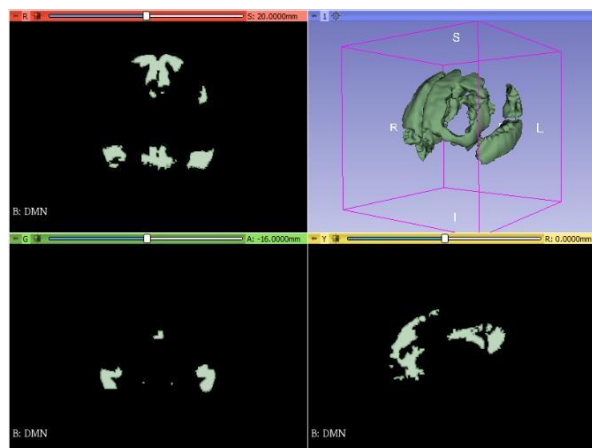


Fig. 4.4: Herramienta Segment Editor II.

Se hizo uso de “Threshold” para poder seleccionar solo la región de interés. En el cuadro azul se observa la malla 3D seleccionada (verde), que se obtiene al utilizar 3D Slicer.

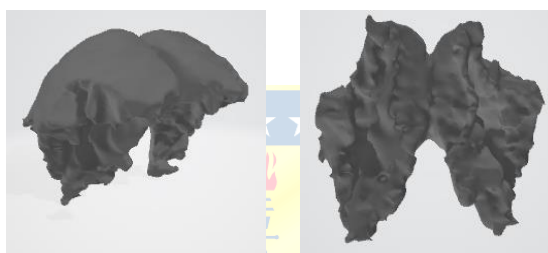


Fig. 4.5: Modelo 3D de máscara 1.

Malla 3D obtenida usando los archivos Nifti en el Software 3D Slicer

4.5. Obtención del Mallado Cerebral

Para que las redes en estado de reposo no se visualizaran en la app de realidad virtual como si solo estuvieran flotando, y con el objetivo de que el usuario pueda entender de mejor forma la ubicación de cada red en el cerebro, se decidió crear una malla 3D que represente la corteza cerebral completa. Para crear este modelo de malla, lo primero que se realizó fue obtener un template de MNI 152 no lineal (tamaño vóxel 1x1x1mm) de la página web “The McConnell Brain Imaging Centre” [39], Fig. 4.6.



Fig. 4.6: Template de MNI 152 no lineal.

Plantilla cerebral para población normal del centro McConnell [39].

Lo que se obtuvo fue una carpeta comprimida llamada *mni_icbm152_nlin_sym_09a*, de la cual se eligió el archivo *mni_icbm152_t1_tal_nlin_sym_09a_mask.nii*. Este fue llevado al software 3D Slicer, se realizaron los mismos pasos que se le aplicaron a las máscaras del Dr. Yeo para obtener los modelos de malla de las redes en estado de reposo. La malla final que se obtuvo se puede observar en la Fig. 4.7.

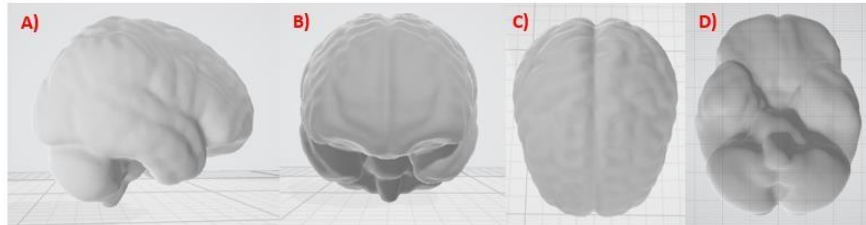


Fig. 4.7: Modelo 3D del cerebro.

En la imagen se observan distintas vistas de la malla 3D del cerebro obtenida de 3D Slicer.

A) Vista lateral; B) Vista anterior; C) Vista Superior; D) Vista Inferior.

4.6. Discusión

Los modelos de mallas de las redes en estado de reposo fueron obtenidos de 3D Slicer a partir de las máscaras de Yeo, y el modelo 3D del cerebro fue creado a partir de un archivo Nifti del Template MNI152 no lineal obtenido de la web.

Los mallados nombrados anteriormente están en bruto, por lo que se les tiene que someter a un post-procesamiento en un software de modelado 3D para mejorar los diseños obtenidos.

Cabe destacar que inicialmente se utilizaron más herramientas de 3D Slicer que las mencionadas, las cuales fueron *Volumen Rendering*, *Crop Volume*, *Segment Editor*, *Segmentations* y *Save*. Las cuales realizaban la creación de la malla 3D de cada ROI, pero en un tiempo total de 4-5 horas por mallado, lo que hizo que el proceso fuera muy lento y agotador, ya que se iba eligiendo de forma manual con una etiqueta la región de interés, y con otra el resto de la imagen (esto fue realizado corte por corte). Por lo que se estudió un poco más en profundidad las herramientas de 3D Slicer, y se encontró que existe un comando llamado *Threshold* en Segment Editor, el cual permite realizar la segmentación usando distintos umbrales. En el caso de este proyecto este comando es de gran utilidad, ya que los archivos obtenidos en la segmentación de imágenes son máscaras con valores 0 y 1, por lo que para segmentar la máscara solamente hay que utilizar un umbral de 1 para guardar en una etiqueta de color la región de interés (color verde). Cabe destacar que, si se utiliza *Threshold* durante la creación de mallas, se pasa de un tiempo total de procesamiento de los 7 mallados de 28-35 horas a un tiempo máximo de 35 minutos, lo que significa un gran cambio en el proyecto.

Capítulo 5. Desarrollo de los modelos

5.1. Introducción

En este capítulo se presentan los modelos 3D que serán utilizados en la app de RV. Estos se obtienen a partir de las mallas obtenidas en el capítulo anterior. Las mallas de la Default Mode Network y el resto de las redes en estado de reposo serán segmentadas en las distintas regiones que las componen, luego las regiones ya segmentadas serán posprocesadas. Los dos pasos anteriores se realizarán en Blender. Este software permite exportar los modelos como archivos. FBX, el cual es un formato compatible con Unity.

5.2. Segmentación de las redes

Al realizar la segmentación de las redes en estado de reposo se observaron dos casos de segmentación, el primero en donde la separación de una región es sencilla, ya que no se encuentra unida a otra en la malla, como es el caso de la corteza posterior cingulada, hipocampo izquierdo y derecho, el giro temporal derecho, y la corteza lateral parietal derecha en la DMN. Este tipo de segmentación se realizó en el modo edición de Blender, se seleccionan las regiones que no se quieren segmentar y se eliminan los vértices, como se observa en la Fig. 5.1.

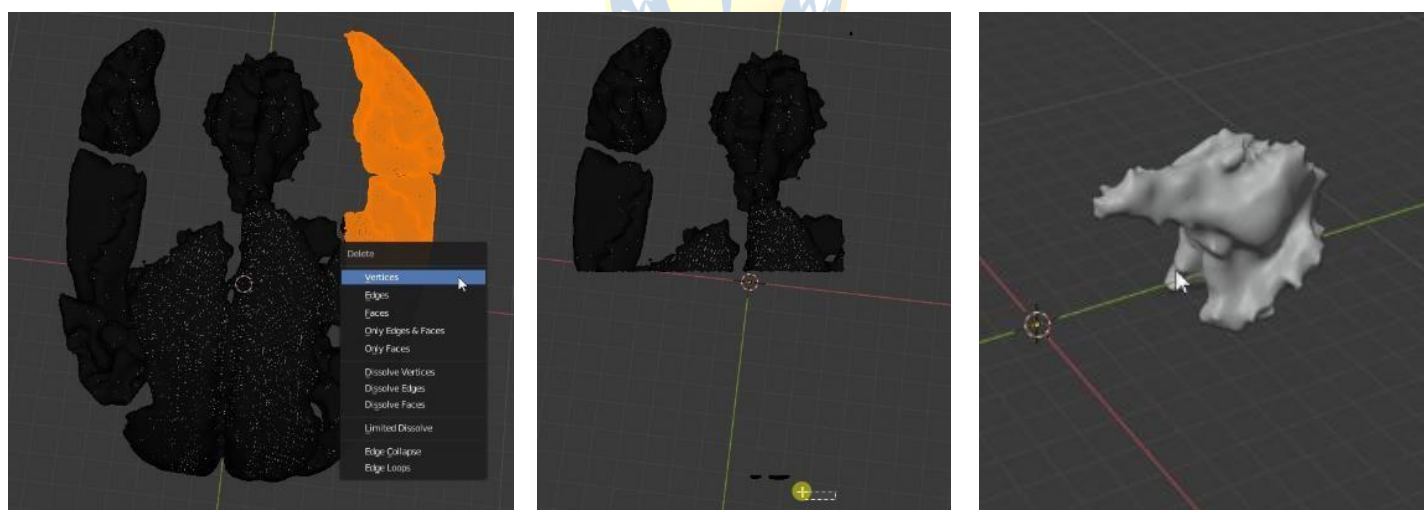


Fig. 5.1: Primer caso de segmentación en Blender.

En la figura se observa la segmentación del PCC, en la imagen de la izquierda se van seleccionando y eliminando los vértices de las regiones que no se quieren segmentar. En la del centro, se observa que ya se han eliminado algunas regiones; en la imagen de la derecha se visualiza el PCC segmentado.

El segundo caso de segmentación es cuando se quiere separar una región que está unida a otra en la malla, como es el caso del giro temporal izquierdo y la corteza lateral parietal derecha en la DMN. Para realizar la separación de las mallas se realizó el mismo paso anterior, y se incluye un paso más, ya que la región segmentada queda abierta en los puntos en los que se encontraba unida a otra en la malla. Este nuevo paso es el de completar la malla abierta, esto se realizó usando la herramienta *fill* de Blender, la cual permite crear nuevos polígonos. El resultado del segundo caso de segmentación se observa en la Fig. 5.2.

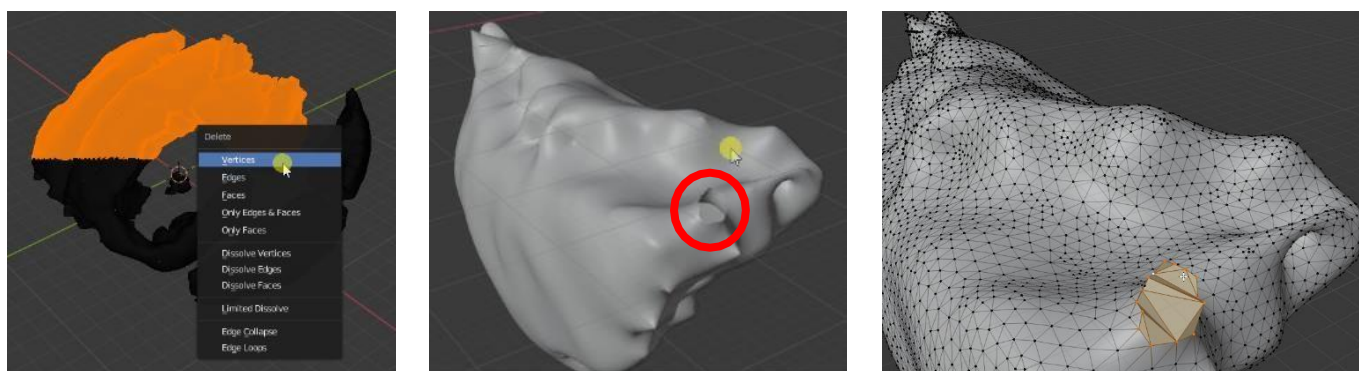


Fig. 5.2: Segundo caso de segmentación en Blender.

En la figura se observa la segmentación del giro temporal izquierdo. En la imagen de la izquierda se van seleccionando y eliminando los vértices de las regiones que no se quieren segmentar. En la del centro se observa el giro segmentado, pero una parte de la malla no está completa. En la imagen de la derecha se observa el giro temporal segmentado, y la malla completamente cerrada gracias a la herramienta *fill*.

Los pasos anteriores se tuvieron que realizar en todas las regiones que forman parte de las redes en estado de reposo, en la Fig. 5.3 se observan algunos resultados de la segmentación.

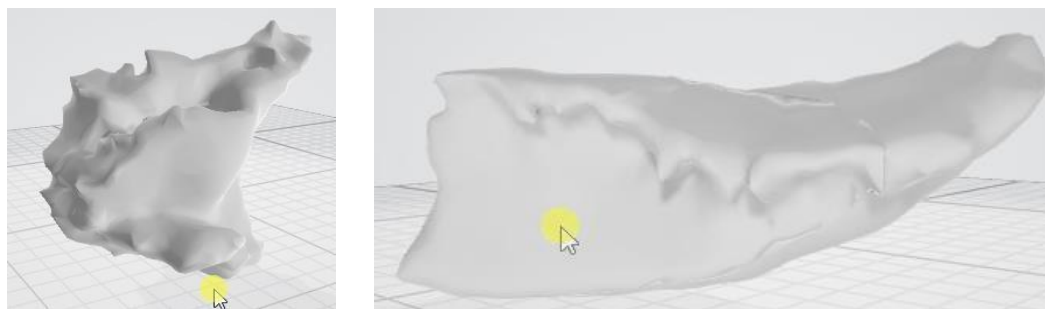


Fig. 5.3: Resultados Segmentación Blender.

En la figura se observa el PCC (izquierda) y el giro temporal izquierdo (derecha) segmentados.

5.3. Posprocesamiento de las mallas

A las mallas segmentadas en el punto anterior se les aplicó un diezmado y en algunos casos se les crearon caras, vértices y aristas. Un diezmado es la disminución de caras de un objeto 3D a tal punto de que se siga manteniendo la forma original del objeto, como se observa en la Fig. 5.4. Esto se realizó para mejorar el rendimiento gráfico en Unity. A continuación, se mostrará la Tabla 5.1 con los resultados del posprocesamiento de la red Somatomotora. En el anexo A.1 se observan las tablas con los posprocesamientos de las otras redes.

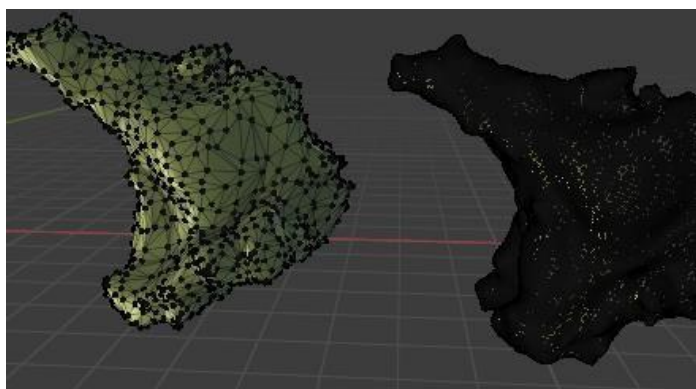


Fig. 5.4: Aplicación del Diezmado en Blender.

En la imagen se observa la aplicación del diezmado al PCC, se observa que se disminuyó el número de caras de la malla, pero se sigue manteniendo la forma.

A la izquierda está el PCC con el diezmado aplicado; a la derecha el PCC original.

Tabla 5.1: Resultados posprocesamiento Red Somatomotora

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Giro Poscentral Izquierdo	39344	0.45	17705
Giro Poscentral Derecho	40400	0.45	18180
Giro Precentral Izquierdo	31895	0.45	14353
Giro Precentral Derecho	27777	0.45	12500
Lóbulo Paracentral Izquierdo	10448	0.45	4702
Lóbulo Paracentral Derecho	12686	0.45	5709

Cabe destacar que a la malla del cerebro también se le realizó un posprocesamiento, el cual solo constó de un diezmado, ya que este modelo 3D contenía muchas caras. Los resultados de esto se observan en la Tabla 5.2.

Tabla 5.2: Resultados posprocesamiento de la malla cerebral

Malla Posprocesada	Total Caras	Razón Diezmado	Total Caras Finales
Cerebro	261792	0.1	26178

Luego, a la malla cerebral y a cada región que forma parte de las redes en estado de reposo se les creó un material. Todos los modelos 3D fueron exportados como .FBX, ya que este formato de archivo es compatible con Unity, y además, permite exportar los modelos con animaciones.

5.4. Discusión

Realizar la segmentación de las redes en estado de reposo fue una tarea que tardó un poco, ya que no es tan simple saber qué regiones pertenecen a cada red, dado que los resultados varían en los distintos trabajos que hablan sobre las redes en estado de reposo y atlas disponibles en la web. Las regiones que componen la Default Mode Network que más se repiten en los diferentes papers y atlas son: la corteza posterior cingulada (PCC), corteza parietal izquierda y derecha, hipocampo izquierdo y derecho, y la corteza prefrontal medial (MPFC). Las regiones segmentadas en este trabajo contienen las mencionadas anteriormente, y algunas otras que igualmente son nombradas en la web y en trabajos de investigación, pero no en la mayoría de estos. Cabe mencionar que la DMN es una de las redes en estado de reposo más estudiadas y de la que se tiene más información hasta el momento. Es por esto que su segmentación fue más fácil y tardó menos tiempo. En cambio, sobre las otras redes se tuvo que buscar esta información en muchos atlas, papers, revistas, trabajos y en la web, para poder realizar su segmentación.

Cabe destacar que el uso de la herramienta *fill* implica tiempo y trabajo manual, ya que en algunos casos completar la malla resultó más complejo.

Luego de realizar la segmentación, se realizó un posprocesamiento en Blender a cada región. Esto se realizó para optimizar el rendimiento gráfico en Unity. La herramienta que se utilizó fue *Decimate*, la cual permitió reducir el número de caras de cada objeto 3D manteniendo la forma original de este. La razón de decimado se fue variando hasta que se llegó a un valor óptimo en donde la malla siguiera con su forma original. Cabe destacar que otra forma de mejorar el rendimiento gráfico es a través de la retopología, la que consiste en construir la malla 3D a partir de polígonos más simples, y no necesariamente de la misma forma (cuadrados, triángulos, entre otros). Se optó por el decimado ya que este proceso era más eficiente en términos de tiempo (20 minutos v/s 4-5 horas).

Capítulo 6. Desarrollo de la App de Realidad Virtual

6.1. Introducción

En este capítulo se explicará cómo se realizó la creación del proyecto, la escena, los shader, la configuración del hardware, en Unity, los cuales son de gran importancia para la visualización de la aplicación de realidad virtual.

6.2. Creación del Proyecto

Para desarrollar la aplicación de realidad virtual lo primero que se tiene que realizar es crear un proyecto en Unity con la configuración de Universal Render Pipeline, asignarle un nombre y seleccionar en qué carpeta se guardará. Luego, se configura Unity para que se pueda visualizar el proyecto con lentes de realidad virtual y se instala el plugin de Oculus para poder utilizar los lentes de RV. Después se eliminó la cámara principal de la jerarquía y se incluyó la cámara RV, la cual imita el movimiento de los lentes. Finalmente, se diseñaron escenas, se importaron las regiones que pertenecen a cada red en estado de reposo y una malla 3D que representa el cerebro.

6.3. Configuración del Oculus Rift S

Para poder utilizar los visores y los mandos de realidad virtual en Unity estos tienen que ser configurados. Lo primero que se realizó fue descargar desde la ventana *Package Manager* de Unity el plugin de Oculus y el paquete de herramientas *XR Interaction Toolkit*, el cual permite crear cualquier entorno de realidad virtual o aumentada.

Luego de instalar los paquetes descargados, se eliminó la cámara principal que viene por defecto en Unity y se creó un *GameObject* vacío el cual se llamó *XR Origin*, que representará la zona donde se moverá el jugador, también conocida como plataforma de RV. A este objeto vacío se le asignaron los componentes que se observan en la Fig. 6.1. Los más importantes que se observan en la imagen son el *Camera GameObject* el cual representa a la cámara del visor de RV, y el *Locomotion System* que permite darle movimiento al jugador.

Para configurar la cámara de los cascos del Oculus Rift S, lo primero que se hizo fue crear dos objetos hijos al *XR Origin*, uno llamado *Camera Offset* que sirve para determinar las coordenadas de los lentes, y el otro nombrado *Main Camera* que es una cámara que entrega al juego la información visual de los lentes. A este último *GameObject* hijo se le agregó el componente *Tracked Pose Driver*, el cual permite imitar la posición y los movimientos de los lentes del Oculus Rift S.

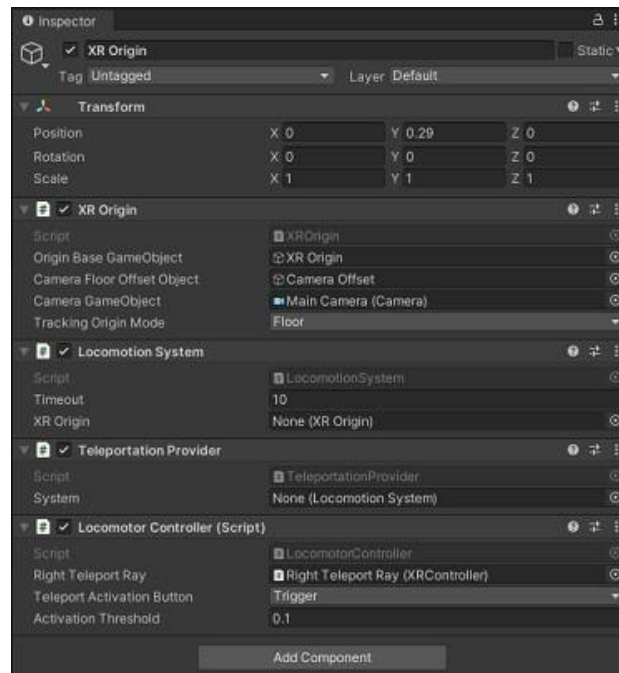


Fig. 6.1: Componentes de XR Origin.

En la imagen se observa los componentes que se le asignaron a los cascos de RV.

Para poder configurar los mandos de realidad virtual se le asignan dos GameObject hijos más a *XR Origin*, llamados *LeftHand Controller* y *RightHand Controller*. Se utilizan algunos componentes que vienen con el paquete *XR Interaction Toolkit*, y se les asignan a los controladores. Ahora el juego detectará los movimientos y la presión de los botones de los mandos del Oculus Rift S. El resultado final de la configuración de los lentes y los controladores de RV se observan en la Fig. 6.2.

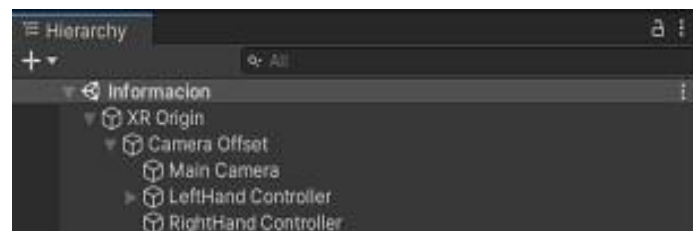


Fig. 6.2: Configuración cascos y mandos RV.

6.4. Escenas

Cuando se comienza a crear un juego o una aplicación, el punto inicial al usar cualquier plataforma es crear una escena, ya que serán la pantalla o pantallas que forman nuestra app. “Las escenas contienen los objetos de su juego. Pueden ser usadas para crear un menú principal, niveles individuales, y cualquier otra cosa. Piensa en cada archivo de escena, como un nivel único. En cada

escena, se colocan un ambiente, obstáculos, y decoraciones, el diseño esencial y la construcción del juego en pedazos” [40].

Para crear las escenas se utilizaron objetos disponibles en Unity, tales como cubos, cilindros, planos, luces, Canvas, objetos de interfaz de usuario, entre otros. Además, se buscaron materiales gratuitos en la Asset Store para incorporarlos en los distintos objetos del juego.

Se crearon un total de 3 escenas, la principal, instrucciones y menú principal.

6.4.1 Escena principal

La escena principal es donde el usuario de la app estará más tiempo. Aquí se encontrará una representación en 3D de las distintas redes en estado de reposo y el cerebro. Además, se desplegará información sobre sus componentes, y se podrá visualizar la activación cerebral de cada red, entre otros. El usuario se podrá trasladar por este lugar utilizando la mano derecha e interactuar con los distintos objetos con la mano izquierda.

Esta escena consta de un pasillo y una sala principal. El pasillo es el lugar donde se encontrará la posición inicial del usuario, y en él se encuentra un panel (componente UI de Canvas), con un video informativo, el cual contiene información sobre la app y los datos que se utilizaron para crear las mallas 3D. En la sala principal se encuentran distribuidas de una forma simétrica las 7 redes en estado de reposo, 7 bases cilíndricas (contienen cada red), 7 mallas del cerebro, 7 tótems informativos y 7 Canvas tipo menú (contienen 3 botones que permiten la activación/desactivación cerebral y resetear).

Los tótems informativos están compuestos por un video (menciona el nombre de la red seleccionada y qué regiones la componen) y 4 botones, 2 para rotar la malla de la red a la izquierda y derecha, y 2 más, uno para reproducir y otro para pausar el video.

Al presionar el botón “*Activación Red On*” se cambiará el material de las regiones de la red seleccionada, a materiales creados a partir del Shader que se explicará más adelante, lo que permitirá simular la activación cerebral de cada red. Si el usuario presiona “*Activación Red Off*”, lo que se observará es que todos los componentes de la red se volverán de color gris, lo que indica desactivación. Finalmente, el botón “*reset*” cambia el material de cada región cerebral al asignado inicialmente en la app.

6.4.2 Escena Menú principal

Se presenta una escena en 3D, la cual contiene un Canvas con 3 opciones de botones con los textos: **redes** (permite ir a “*escena principal*”), **instrucciones** (nos lleva a una sala en donde se dan las instrucciones de la app) y **salir** (nos permite salir del juego).

6.4.3 Escena Instrucciones

Esta sala contiene frente al usuario un panel con instrucciones. Inicialmente aparece un mensaje de bienvenida y el botón con el texto “comenzar”. Al presionar este botón se dan las indicaciones para que el jugador aprenda a desplazarse por el lugar haciendo uso del *Trigger* del mando de Oculus, y aparece un botón con la opción “siguiente”. La segunda instrucción le indica al usuario como puede interactuar con los objetos que están en la sala, los cuales son un cubo, una mesa y una esfera.

6.5. Creación del Shader

Se decidió crear un Shader, el cual es un Asset con instrucciones en código para que la tarjeta gráfica ejecute, y a partir de este Asset crear un material con distinto color. Se creó un PBR Graph (tipo de Shader dentro de Unity), que posee un PBR Master Node que representa la salida final del shader, al cual se le pueden conectar un máximo de 11 entradas las que van otorgando distintas características gráficas al shader [41]. Las 11 entradas al PBR Master Node son:

- **Vertex Position:** Permite seleccionar el sistema de coordenadas a utilizar.
- **Vertex Normal:** Define la normal absoluta del vértice del espacio del objeto por vértice.
- **Vertex Tangent:** Define la tangente absoluta del vértice del espacio del objeto por vértice.
- **Albedo:** Define el color del material.
- **Normal:** Define el valor de la normal del material a crear.
- **Emission:** Define qué valor toma el color de emisión.
 - A este nodo se le conectó un nodo de Efecto Fresnel y un nodo de Tiempo sinusoidal, los cuales permiten que se observe un efecto de respirado en el objeto (aumenta la intensidad del color y luego disminuye).
- **Metallic:** Define el grado de metalizado del material.
- **Smoothness:** Permite definir el grado de pulido del material.

- **Oclusion:** Define el nivel de oclusión ambiental del material.
- **Alpha:** Se utiliza para transparencia, si *Alpha* es mayor en valor que *AlphaClipThreshold* el material se comienza a hacer transparente.
- **AlphaClipThreshold:** Umbral utilizado para que ocurra transparencia en el material.

El PBR Graph se observa en Fig. 6.3.

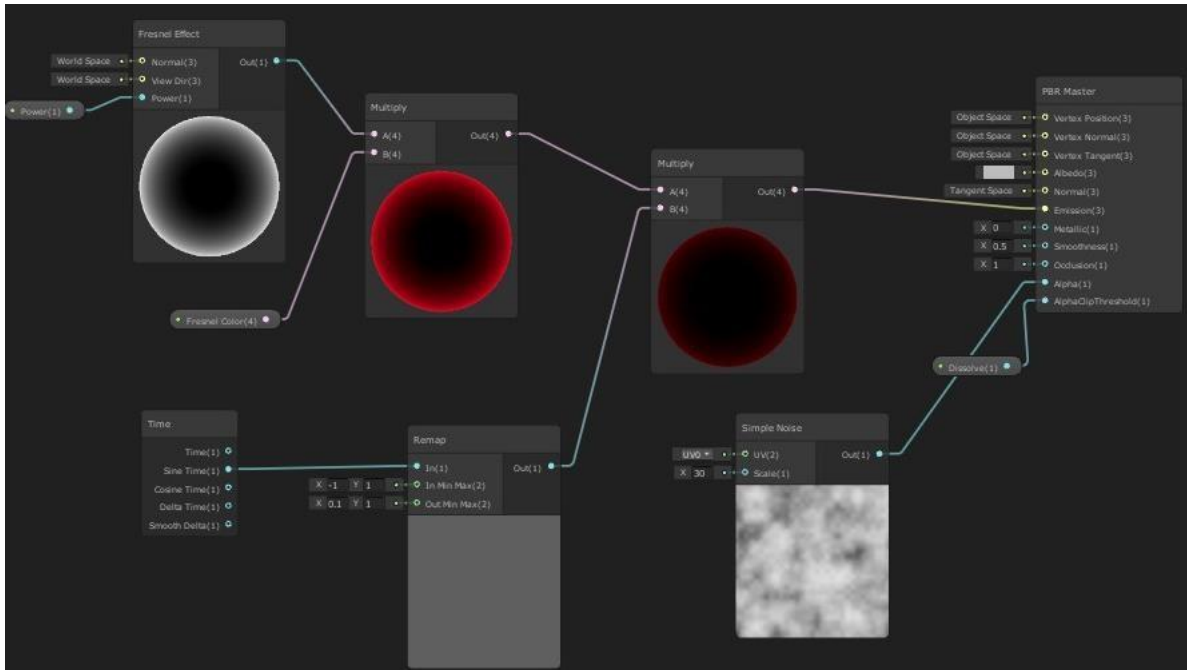


Fig. 6.3: PBR Graph.

En la imagen se muestran ciertas instrucciones gráficas que tendrá nuestro PBR Graph para que la tarjeta gráfica ejecute.

Finalmente se crearon 2 materiales a partir del Shader creado, uno llamado *Glow_Material_HA*, y el segundo llamado *Glow_Material_LA* como se observa en la Fig. 6.4 A), el primero representa el brillo de una región cuando tiene alta activación cerebral (en color rojo), y el segundo representa el brillo de una región que tiene una baja activación (en color amarillo). Lo que se quiere lograr con esto es mostrar de una forma sencilla lo que se observa en una resonancia magnética funcional, en la cual se representa con una escala de colores el nivel de activación cerebral al realizar alguna tarea o estar en reposo, como se observa en Fig. 6.4 B).

6.6. Materiales de las regiones cerebrales

Se decidió crear un material para cada región que pertenezca a una red en estado de reposo (Fig. 6.5), con el propósito de visualizar cada componente de una forma más fácil en el cerebro. En la Tabla 6.1 se muestra el color de cada material asignado de la Red Somatomotora, y en el anexo A.2 se observan las tablas para las otras redes en estado de reposo.

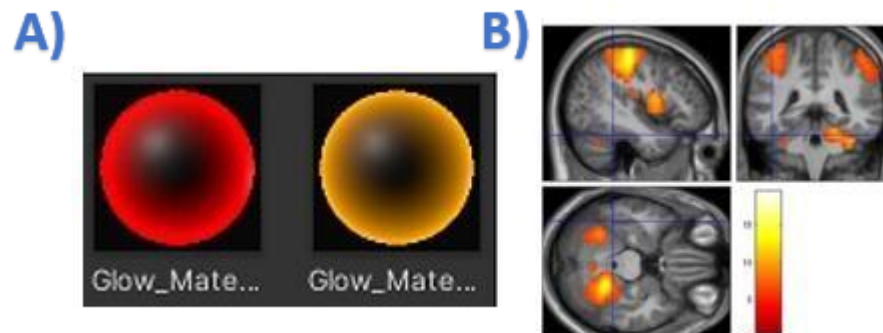


Fig. 6.4: Shader y fMRI.

En A) se observan los materiales creados a partir del Shader PBR, los cuales representarán el material de una región con alta activación cerebral (rojo), y el otro una con baja (amarillo). En B) se observa una fMRI, la cual representa la activación cerebral con una escala de colores, y este efecto es el que se quiere lograr usando el Shader [49].



Fig. 6.5: Materiales Regiones DMN.

En la imagen se observan los materiales asignados a algunas regiones de la DMN, en azul se representa la Corteza Lateral Parietal, en morado la Corteza Prefrontal Dorsolateral, en verde claro el Giro Frontal Superior, y en amarillo el Giro Temporal.

Tabla 6.1: Materiales asignados a cada región de la red Somatomotora

Red Somatomotora	
Región	Color
Giro Poscentral	Amarillo
Giro Precentral	Verde
Lóbulo Paracentral	Rojo

6.7. Interfaz de Usuario

Para navegar a través de la aplicación de RV se utilizan los visores de Oculus, y para poder interactuar con los objetos dentro de las escenas se usan los controladores del dispositivo. Para que el usuario sintiera una mayor sensación de inmersión dentro del software se incorporaron modelos de mano 3D obtenidos del plugin de Oculus (ver en la Fig. 6.6), con el objetivo de que el usuario observe las animaciones creadas al presionar botones (se explicaran más adelante).

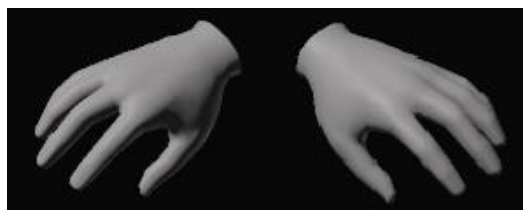


Fig. 6.6: Modelo Manos.

En la imagen se observan los modelos de manos virtuales obtenidas del plugin de Oculus.

Al presionar los botones de los comandos, el usuario puede desplazarse a través de la escena (Fig. 6.7 botón 6 mano derecha), interactuar/agarrar objetos (Fig. 6.7 botón 6 mano izquierda).

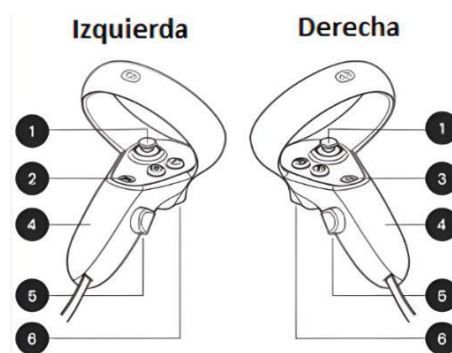


Fig. 6.7: Diagrama mandos Oculus Rift S [50].

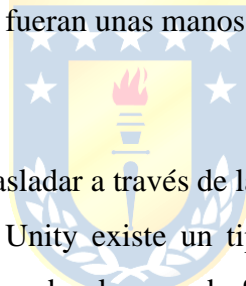
1. Palanca de mando; 2. Botón menú Oculus; 4. Botón Grip; 6. Botón Trigger.

6.8. Animaciones

Las manos virtuales que se obtuvieron del plugging de Oculus son como un bloque, no poseen movimiento, es por esta razón que se crearon animaciones para que el usuario tenga una sensación de inmersión mayor.

Para poder realizar esto se utilizó la ventana *Animator* de Unity. Lo primero que se realizó fue crear dos *Animator Controller* (“Right Hand Animator” y “Left Hand Animator”), los cuales son un asset que permite crear un conjunto de animaciones para un GameObject en específico [42]. Se utilizó la ventana *Animator* para poder crear las animaciones a través de un árbol de combinación. En la Fig. 6.8 se observa el árbol de combinación de la mano derecha. Se crearon 3 movimientos para cada mano activados por la presión de los botones *Trigger* y *Grip*: posición neutral, puño y pellizco. Una vez creada cada animación se le asignó a la mano izquierda el *Animator Controller Left Hand Animator*, y a la derecha el *Right Hand Animator*.

Ahora, si se presiona cualquiera de los botones nombrados anteriormente se observará en el juego que las manos se mueven como si fueran unas manos reales.



6.9. Teletransportación

Para que el usuario se pudiera trasladar a través de la aplicación de realidad virtual se decidió que se usará la teletransportación. En Unity existe un tipo de teletransporte interactivo llamado *Teleport Area*, el cual permite al jugador desplazarse de forma libre por cierta área designada sin experimentar las molestias originadas al caminar en la realidad virtual.

Para realizar esto, se utiliza el *Locomotion System* que viene en el XR Interaction Toolkit, el cual es un paquete que se puede instalar en Unity y permite crear experiencias de realidad virtual y realidad aumentada. El *Locomotion System* le entrega al usuario de RV movimiento, además, permite implementar la teletransportación.

Lo primero que se realizó fue elegir qué botón del controlador de Oculus permitiría el desplazamiento. Se eligió el *Trigger* de la mano derecha para realizar esto. Luego se seleccionó el área en la que se podía teletransportar el usuario, ya que puede moverse libremente, pero solo en lugares elegidos por el programador. Después se trabajó en los detalles gráficos para que el jugador pudiera observar hacia dónde estaba apuntado el rayo de teletransportación. Para realizar esto se creó

un GameObject que representará el lugar exacto hacia donde se desplazaría. Además, se le asignó un material a partir de un Shader creado, para otorgarle mejores características de diseño.

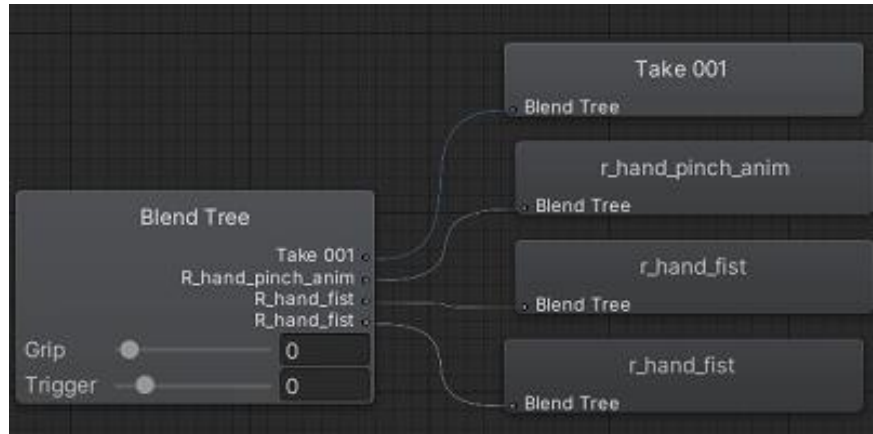


Fig. 6.8: Árbol de combinación creado para la mano derecha.

Este árbol se visualiza en la ventana Animator de Unity. Se observa que a la mano derecha se le crearon 3 movimientos: posición neutral (“Take 001”), pellizco (“Pinch”) y puño (“Fist”). La primera animación viene por defecto con las manos, y las otras dos se activan al presionar los botones *Trigger* y *Grip*.

6.10. Discusión

La creación de la escena es un proceso intuitivo, lo que no tiene mayor dificultad, pero toma tiempo, ya que se tiene que ir eligiendo detalles específicos que el usuario no nota al jugar, pero son indispensables en la app.

Se decidió crear un Shader con el objetivo de representar de una forma sencilla la activación cerebral para que sea fácil entender. Esto se hizo basándose en cómo se representa a través de etiquetas de colores la activación cerebral en la resonancia magnética funcional. La creación del PBR Graph tardó bastante tiempo, dado que se tuvo que aprender del funcionamiento del Shader desde cero. Cabe destacar que el efecto que otorga este Asset a cada malla, es importante para poder visualizar cómo va cambiando la activación de las regiones pertenecientes a las redes en estado de reposo.

Se decidió crear animaciones, dado que al animar las manos virtuales al realizar movimientos con las manos reales se aumenta la sensación de inmersión en la app del usuario

Dentro de las opciones que se investigaron en la web para realizar el desplazamiento del usuario dentro de la app de realidad virtual, las más utilizadas son: a través del movimiento de los joysticks de los controles del Oculus, y mediante teletransportación. En la primera opción el jugador se desplaza moviendo una de las palancas de los mandos (movimientos adelante, atrás, izquierda y derecha). Esta alternativa a veces causa mareos, dolores de cabeza, náuseas, entre otros efectos

secundarios en algunos usuarios, ya que la persona observa a través de los lentes un desplazamiento no continuo. Para poder moverse usando la segunda opción se tiene que tener un área de teletransportación (elegida por el programador), y para desplazarse el usuario simplemente apunta con uno de los controles hacia algún lugar (dentro del área de teletransportación) y presiona un botón para teletransportarse.

En esta aplicación se decidió elegir la segunda opción, con el objetivo de evitar malestares y mejorar la experiencia del usuario mientras este usando la app.

Cabe destacar que ambas opciones de desplazamiento fueron probadas por 3 personas, los 3 presentaron mareos al moverse usando la primera opción, y ningún efecto secundario al utilizar la segunda.



Capítulo 7. Resultados

7.1. Introducción

Aquí se mostrarán los resultados de la aplicación de realidad virtual creada en Unity. Se observará el resultado final de la app y como se visualiza usando los lentes Oculus Rift S.

7.2. Modelos 3D con sus respectivos materiales

Los resultados de las regiones de la DMN y el resto de las redes en estado de reposo con sus respectivos materiales asignados se observan en el anexo B. En la Fig. 7.1 y Fig. 7.2 se observa la corteza posterior cingulada (PCC) de la DMN y la malla del cerebro con sus respectivos materiales asignados.

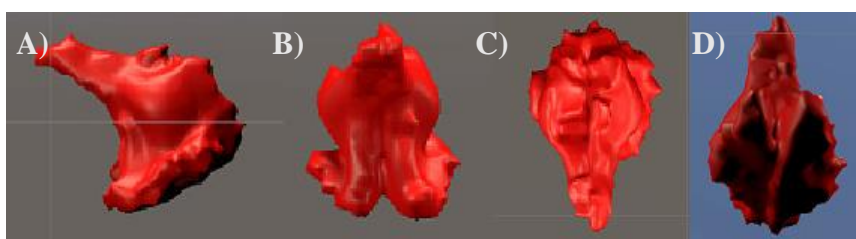


Fig. 7.1: PCC de la Default Mode Network con su material asignado.

En la imagen se observan distintas vistas de la malla de la corteza posterior cingulada en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior

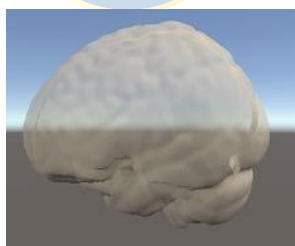


Fig. 7.2: Malla cerebro con su respectivo material en Unity.

En la imagen se observa modelo de malla creado, se le asignó un material con una transparencia 0.5 de color gris.

7.3. Redes en estado de reposo importadas en la escena

Luego de segmentar las regiones pertenecientes a la DMN, red visual, somatomotora, atención ventral, atención dorsal, límbica y frontoparietal, y asignarles un material, como se explica en los capítulos anteriores, estas se importan y posicionan en la escena. Los resultados se observan en la Fig. 7.3.

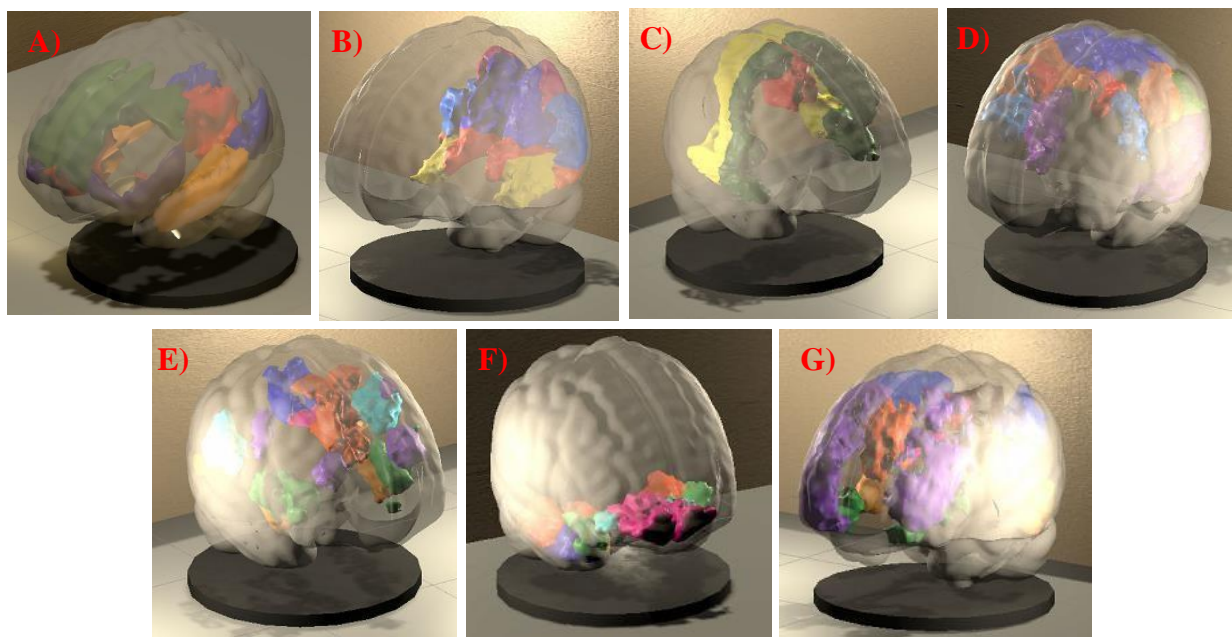


Fig. 7.3: Redes importada en la escena.

A) DMN; B) Visual; C) Somatomotora; D) Atención Dorsal; E) Atención Ventral; F) Límbica; G) Frontoparietal.

7.4. Aplicación RV

Luego de crear la aplicación de realidad virtual, se probó la app haciendo uso de los lentes Oculus Rift S. Cabe destacar que se permite al usuario desplazarse por la escena, y que interactúe con algunos objetos.

7.4.1 Escena principal

En esta escena el usuario podrá visualizar las distintas redes en estado de reposo en la aplicación de realidad virtual. En la Fig. 7.4 se observa parte de la escena principal y cómo se verá la activación de la Default Mode Network. En esta sala se muestran las mallas de los cerebros y los componentes de las distintas redes en estado de reposo, paneles en el suelo con información de las redes y la activación/desactivación cerebral, y botones interactivos. En los tótems informativos se menciona información sobre cuáles son las componentes de la red seleccionada en la escena.



Fig. 7.4: Resultados escena principal.

En la imagen de la izquierda se observa cómo es la sala principal de la app, en donde se visualizan las mallas 3D de cada red junto a sus tótems y paneles informativos. En la imagen de la derecha se muestra la DMN en la escena, al presionar el botón “Activación Red On”, el jugador visualizará en rojo las regiones de la DMN con mayor activación cerebral y en naranja las con menor.

7.4.2 Escena Menú principal

Esta es la primera escena que verá el jugador en la aplicación de realidad virtual, donde verá un Canva con el título “Menú Principal” y 3 botones interactivos que el usuario podrá presionar usando el botón Trigger. Si se presionan los primeros 2 botones se dirigirá al jugador a las otras 2 escenas restantes (principal e instrucciones), si presiona “Salir” se sale del juego. El resultado final se observa en la Fig. 7.5

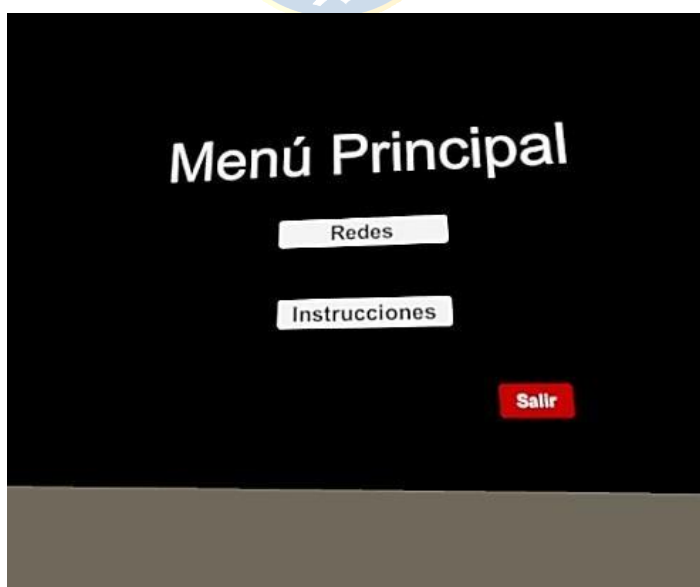


Fig. 7.5: Resultado final Escena Menú Principal.

7.4.3 Escena Instrucciones

Aquí se encuentra una sala pequeña, que contiene un panel con información de bienvenida al módulo de instrucciones, como se observa en la Fig. 7.6 A), al presionar el botón “Comenzar” se le mostrarán al usuario indicaciones para que él pueda desplazarse por el lugar (Fig. 7.6 B)), e instrucciones para que pueda interactuar con objetos, como se observa en Fig. 7.6 C).

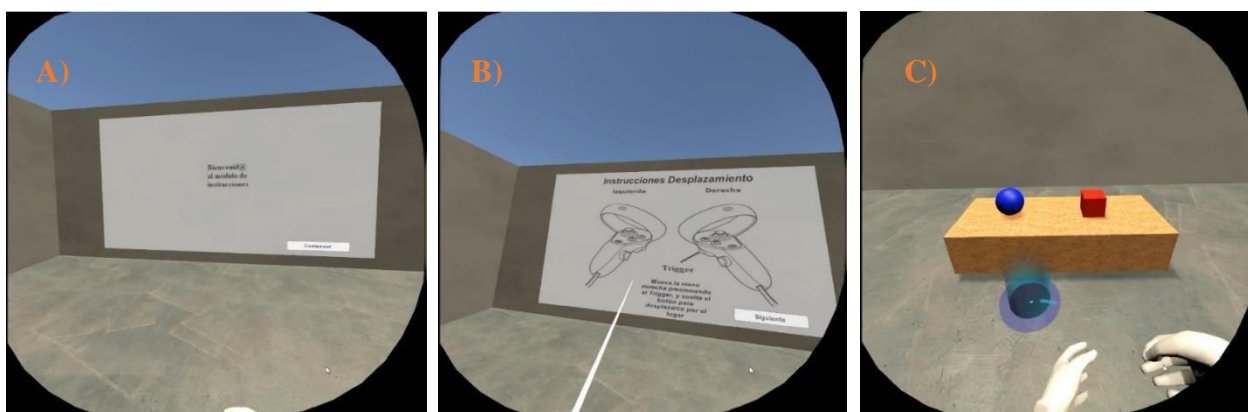


Fig. 7.6: Resultado final Escena Instrucciones.

En A) se observa el mensaje de bienvenida al módulo instrucciones; B) se muestran las instrucciones para desplazamiento; C) se observa una mesa con un cubo rojo y una esfera azul, las cuales el usuario utilizará para practicar las instrucciones de agarre.

7.5. Animaciones

En la aplicación se crearon 3 animaciones para las manos virtuales, como se observa en la Fig.

7.7. Estas animaciones se activan para indicarle al usuario la presión de *Grip* y *Trigger*:

- Posición neutra = Esta animación se activa cuando el jugador está en el juego sin presionar ningún botón del controlador de Oculus, esta animación es la que viene por defecto.
- Puño = Se activa cuando se presiona el botón *Grip* de la mano izquierda o derecha.
- Pellizco = Esta posición de las manos se observará en el juego cuando el usuario presione el botón *Trigger* de la mano izquierda o derecha.

7.6. Teletransportación

Se le permitió al usuario desplazarse por medio de la teletransportación. Para esto se configuraron algunas características en Unity, y se creó un GameObject que representará el lugar hacia dónde se iba a desplazarse. Se personalizó el rayo de teletransportación para que tuviera un mejor diseño, los resultados se observan en la Fig. 7.8.



Fig. 7.7: Animaciones manos virtuales.

En la figura se observa el resultado final de las 3 animaciones creadas: posición neutra (izquierda), puño (medio) y pellizco (derecha).



Fig. 7.8: Área de Teletransportación.

En la imagen de la izquierda se puede observar el GameObject creado, que representa el lugar al que se quiere teletransportar el usuario. En la figura de la derecha se observa el jugador en la app apuntando hacia donde se quiere desplazar.

7.7. Discusión

Al utilizar las gafas de realidad virtual, se observa de forma clara los componentes de las distintas escenas. Existe un buen funcionamiento de todos los componentes de la interfaz de usuario, y se observa de forma clara la información que se entrega al jugador en cada una de las salas. Se visualiza de buena forma y se entiende la posición de las componentes de cada red en el cerebro. El efecto de animación de las manos virtuales se observa casi al instante de presionar los botones de los mandos de Oculus, lo que aumentó la sensación de inmersión del jugador. El diseño del área de teletransportación y la creación del efecto gráfico del rayo de teletransportación, mejoraron la calidad del diseño gráfico del juego, dándole un efecto más futurista.

Capítulo 8. Conclusiones

8.1. Discusión

De acuerdo al estudio bibliográfico realizado, las técnicas de análisis de rfMRI más utilizadas son el software CONN, que realiza un análisis completo basado en semillas a los datos entregados al programa, y los métodos ICA, del cual surgieron varias redes en estado de reposo conocidas, y ROI, que entrega un mapa de conectividad funcional de fácil interpretación.

En los trabajos previos se menciona que los estudiantes tienen mejores rendimientos académicos cuando estos aprenden haciendo uso de la realidad virtual. Este punto es relevante, ya que el desarrollo de esta aplicación de RV para visualizar la DMN y el resto de las redes en estado de reposo será un gran aporte para que el público general visualice, interactúe y aprenda con algo que no es fácil de imaginar cómo lo es una red funcional del cerebro.

A los modelos de malla que representan al cerebro y el de las redes en estado de reposo se les realizó en Blender un diezmado, y una segmentación para dividir las redes en las diferentes áreas que las componen. En la web y los distintos trabajos que hablan de la red de modo por defecto, se mencionan muchas regiones que la componen, siendo las que más se repiten la corteza posterior cingulada (PCC), corteza parietal izquierda y derecha, hipocampo izquierdo y derecho, y la corteza prefrontal medial (MPFC), las cuales pertenecen a las regiones segmentadas en este trabajo. En el caso de las otras redes, se obtuvieron regiones segmentadas que igualmente coinciden con trabajos, papers y atlas, pero se debe destacar que, en comparación con la DMN, éstas no son tan estudiadas.

En el capítulo “desarrollo de los modelos” se menciona que se le realizó un posprocesamiento a cada región segmentada. Esto se realizó dado que con esto se optimiza el rendimiento gráfico del computador.

En el capítulo 6 se mencionó que se crearon distintas salas, shaders, animaciones, entre otros, con el objetivo de que el usuario tuviera una mejor experiencia en la aplicación de realidad virtual. Cabe destacar que antes de realizar la elección de cada una de las características mencionadas anteriormente se probaron distintas variantes en cada caso, con el objetivo de entregar lo mejor al usuario.

Cabe destacar que los pasos mencionados en este trabajo para poder crear una aplicación de realidad virtual parecen simples de seguir, pero se debe mencionar que no es tan fácil, ya que, para poder implementar algo simple se deben observar una gran cantidad de videos en la web y seguir muchos tutoriales, ya que, Unity es un software que se va actualizando constantemente y puede llegar

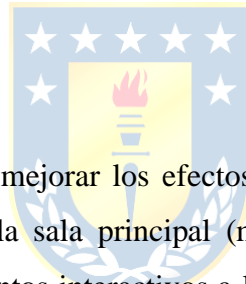
a tener más de una versión por año. Estas actualizaciones hacen que algunas características que funcionan en un tutorial no les funcionen a los programadores al minuto de probar algo nuevo, dado que las herramientas utilizadas van quedando obsoletas. Es por esto que lo más difícil de este proyecto ha sido la gran cantidad de tiempo que toma aprender sobre los distintos softwares y la visualización de largas horas de tutoriales.

8.2. Conclusiones

La aplicación fue probada usando los lentes de realidad virtual Oculus Rift S. Se verificó el buen funcionamiento de los botones y paneles interactivos dentro de la app, y la correcta visualización de las mallas 3D de las 7 redes en estado de reposo y las 3 escenas creadas.

Mientras el usuario esté usando la aplicación este visualizará videos y paneles informativos sobre las redes en estado de reposo, podrá rotar las mallas de cada red para visualizarla desde todos los ángulos, y a través de la presión de botones se visualiza la activación/desactivación cerebral.

Con lo anteriormente mencionado se dan por cumplidos los objetivos de esta memoria de título.



8.3. Trabajo Futuro

En un trabajo futuro se podría mejorar los efectos visuales de la activación/desactivación cerebral de cada red, y el diseño de la sala principal (museo) para dar un efecto más realista. Además, se podrían agregar más elementos interactivos a la aplicación, con el objetivo de que este software sea más lúdico para las personas.

Se podría implementar alguna escena o sala en la que el usuario pueda estar dentro del cerebro y visualizar la activación cerebral, y los componentes de cada red desde el interior de este.

Esta aplicación también podría ser desarrollada para PC y celulares Android e iOS, con el objetivo de que no solo las personas con acceso a lentes de realidad virtual puedan aprender sobre las redes en estado de reposo.

Cabe mencionar que en el futuro se podría complementar este trabajo con más información sobre las regiones que componen cada red, y agregar más datos sobre las resting state networks, que cada día están siendo más estudiadas.

Bibliografía

- [1] B. T. Thomas Yeo, F. Krienen, J. Sepulcre, M. Sabuncu, D. Lashkari, M. Hollinshead, J. Roffman, J. Smoller, L. Zöllei, J. Polimeni, B. Fisch, H. Liu, R. Buckner, “The organization of the human cerebral cortex estimated by intrinsic functional connectivity,” *Journal of Neurophysiology*, vol. 106, no. 3, pp. 1125–1165, Sep. 2011, doi: 10.1152/jn.00338.2011.
- [2] Ivy Wigmore, “Tecnología Inmersiva.” [Online]. Available: <https://www.techtarget.com/whatis/definition/immersive-technology>. [Accessed: Apr. 25, 2022]
- [3] “Realidad Virtual | 10 hitos en la historia de la VR.” [Online]. Available: <https://deusens.com/es/blog/hitos-historia-realidad-virtual>. [Accessed: Apr. 25, 2022]
- [4] “Historia de la Realidad Virtual.” [Online]. Available: <https://xperimentacultura.com/historia-de-la-realidad-virtual/>. [Accessed: Apr. 25, 2022]
- [5] Thomas Yeo, “Cortical Parcellation Estimated by Intrinsic Functional Connectivity,” 2013. [Online]. Available: https://surfer.nmr.mgh.harvard.edu/fswiki/CorticalParcellation_Yeo2011. [Accessed: Oct. 10, 2021]
- [6] jove, “fMRI: resonancia magnética funcional.” [Online]. Available: <https://www.jove.com/v/5212/fmri-resonancia-magnetica-funcional?language=Spanish>. [Accessed: Oct. 09, 2021]
- [7] S. M. Smith, C. Beckmann, J. Andersson, E. Auerbach, J. Bijsterbosch, G. Douaud, E. Duff, D. Feinberg, L. Griffanti, M. Harms, M. Kelly, T. Laumann, K. Miller, S. Moeller, S. Petersen, J. Power, G. Salimi-Khorshidi, A. Snyder, A. Vu, M. Woolrich, J. Xu, E. Yacoub, K. Uğurbil, D. Van Essen, M. Glasser, “Resting-state fMRI in the Human Connectome Project,” *Neuroimage*, vol. 80, pp. 144–168, Oct. 2013, doi: 10.1016/j.neuroimage.2013.05.039.
- [8] M. P. van den Heuvel and H. E. H. Pol, “Exploring the brain network: A review on resting-state fMRI functional connectivity,” *Psihiatria Biologica*, vol. 18, no. 1, pp. 28–41, 2011, doi: 10.1016/j.psiq.2011.05.001.
- [9] “¿Qué es la Realidad Virtual?” [Online]. Available: <https://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html>. [Accessed: May 01, 2022]
- [10] “¿Qué es la realidad virtual y cuáles son sus aplicaciones?” [Online]. Available: <https://pragm-a.com/que-es-la-realidad-virtual-y-cuales-son-sus-aplicaciones/>. [Accessed: May 01, 2022]

- [11] K. Kwon, J. S. Park, and B. S. Shin, “Virtual anatomical and endoscopic exploration method of internal human body for training simulator,” *Journal of Korean Medical Science*, vol. 35, no. 12, Mar. 2020, doi: 10.3346/JKMS.2020.35.E90.
- [12] G. Zhao, M. Fan, Y. Yuan, F. Zhao, and H. Huang, “The comparison of teaching efficiency between virtual reality and traditional education in medical education: a systematic review and meta-analysis,” *Annals of Translational Medicine*, vol. 9, no. 3, pp. 252–252, Feb. 2021, doi: 10.21037/atm-20-2785.
- [13] S. G. Izard, J. A. Juanes Méndez, and P. R. Palomera, “Virtual Reality Educational Tool for Human Anatomy,” *Journal of Medical Systems*, vol. 41, no. 5, May 2017, doi: 10.1007/s10916-017-0723-6.
- [14] Will Greenwald, “Oculus Quest 2 vs. Oculus Rift S: Which VR Headset Should You Buy?” [Online]. Available: <https://www.pcmag.com/comparisons/oculus-quest-vs-oculus-rift-s-which-vr-headset-should-you-buy#:~:text=The%20Oculus%20Rift%20S%20is%20a%20tethered%20VR%20headset%20designed,uses%20your%20PC%20for%20processing.> [Accessed: May 01, 2022]
- [15] Enrique Pérez, “Las nuevas Oculus Rift S mejoran la tecnología de rastreo y eliminan el lío de cables para ser unas gafas VR más fáciles de utilizar.” .
- [16] Unity, “¿Qué es un Asset de Unity?” [Online]. Available: <https://unity3d.com/es/quick-guide-to-unity-asset-store>. [Accessed: Dec. 02, 2021]
- [17] Unity Documentation, “GameObjects.” [Online]. Available: <https://docs.unity3d.com/Manual/GameObjects.html>. [Accessed: Dec. 02, 2021]
- [18] UnityDocumentation, “Prefabs.” [Online]. Available: <https://docs.unity3d.com/Manual/Prefabs.html>. [Accessed: Dec. 02, 2021]
- [19] Unity Documentation, “Assets Shader.” [Online]. Available: <https://docs.unity3d.com/es/530/Manual/class-Shader.html>. [Accessed: Dec. 02, 2021]
- [20] Unity, “Canvas.” [Online]. Available: <https://docs.unity3d.com/es/530/Manual/UICanvas.html#:~:text=El%20Canvas%20es%20un%20Game,uno%20en%20la%20escena%20ya.> [Accessed: May 04, 2022]
- [21] Unity Technologies, “Unity Documentation,” 2016. [Online]. Available: <https://docs.unity3d.com/es/530/Manual/LearningtheInterface.html>. [Accessed: Oct. 14, 2021]
- [22] 3D Slicer, “3D Slicer,” 2021. [Online]. Available: <https://www.slicer.org/>. [Accessed: Oct. 04, 2021]

- [23] Wikipedia, “Blender.” [Online]. Available: <https://es.wikipedia.org/wiki/Blender>. [Accessed: Oct. 10, 2021]
- [24] MathWorks, “Formato de archivo NifTI.” [Online]. Available: <https://la.mathworks.com/help/images/ref/niftiread.html>. [Accessed: Dec. 02, 2021]
- [25] PowerData, “¿Qué son los metadatos y cuál es su utilidad?,” Mar. 19, 2016. [Online]. Available: <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-son-los-metadatos-y-cual-es-su-utilidad>. [Accessed: Oct. 11, 2021]
- [26] DescubreArduino, “STL u OBJ: ¿Cuál deberías usar?” [Online]. Available: <https://descubrearduino.com/stl-u-obj/>. [Accessed: Oct. 10, 2021]
- [27] “Qué es el formato de archivo FBX.” [Online]. Available: <https://www.softzone.es/windows/como-se-hace/archivos-fbx/>. [Accessed: Jun. 13, 2022]
- [28] H. Lv, Z. Wang, E. Tong, L. Williams, G. Zaharchuk, M. Zeineh, A. Goldstein-Piekarski, T. Ball, C. Liao, M. Wintermark, “Resting-state functional MRI: Everything that nonexperts have always wanted to know,” *American Journal of Neuroradiology*, vol. 39, no. 8, pp. 1390–1399, Aug. 2018, doi: 10.3174/ajnr.A5527.
- [29] AFNI, “AFNI” [Online]. Available: https://afni.nimh.nih.gov/about_afni. [Accessed: Oct. 11, 2021]
- [30] NITRC, “CONN: functional connectivity toolbox.” [Online]. Available: <https://www.nitrc.org/projects/conn/>. [Accessed: Oct. 12, 2021]
- [31] FSL, “MELODIC.” [Online]. Available: <https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/MELODIC>. [Accessed: Oct. 13, 2021]
- [32] TReNDS, “Group ICA Of fMRI Toolbox(GIFT).” [Online]. Available: <https://trendscenter.org/software/gift/>. [Accessed: Oct. 13, 2021]
- [33] E. Proal, M. Alvarez-Segura, M. De La Iglesia-Vayá, L. Martí-Bonmatí, F. Castellanos, N. Kline, P. Green, R. Cowen, “Actividad funcional cerebral en estado de reposo: Redes En Conexión,” *Rev Neurol*, vol. 52, no. 1, pp. 3–10, 2011.
- [34] K. A. Smitha, K. Akhil Raja, K. Arun, P. Rajesh, B. Thomas, T. Kapilamoorthy, C. Kesavadas, “Resting state fMRI: A review on methods in resting state connectivity analysis and resting state networks,” *Neuroradiology Journal*, vol. 30, no. 4, pp. 305–317, Aug. 2017, doi: 10.1177/1971400917697342.

- [35] Cycon, “Anatomical Automatic Labeling.” [Online]. Available: <https://web.archive.org/web/20150211005316/http://www.cycon.fr/index.php/en/plateforme-en/freeware>. [Accessed: Dec. 02, 2021]
- [36] G. M. Rojas, J. A. Fuentes, and M. Gálvez, “Mobile device applications for the visualization of functional connectivity networks and EEG electrodes: iBraiN and iBraiNEEG,” *Frontiers in Neuroinformatics*, vol. 10, no. OCT, Oct. 2016, doi: 10.3389/fninf.2016.00040.
- [37] “Técnicas para identificar Requisitos Funcionales y No Funcionales” [Online]. Available: <https://sites.google.com/site/metodologiareq/capitulo-ii/tecnicas-para-identificar-requisitos-funcionales-y-no-funcionales>. [Accessed: Jul. 27, 2022]
- [38] HCP, “The Human Connectome Project.” [Online]. Available: <http://www.humanconnectomeproject.org/>. [Accessed: Dec. 03, 2021]
- [39] The McConnell Brain Imaging Centre, “ICBM 152 Nonlinear atlases version 2009.” [Online]. Available: <https://www.bic.mni.mcgill.ca/ServicesAtlases/ICBM152NLin2009>. [Accessed: May 03, 2022]
- [40] Unity Documentation, “Escena.” [Online]. Available: <https://docs.unity3d.com/es/530/Manual/CreatingScenes.html>. [Accessed: Dec. 12, 2021]
- [41] Unity, “PBR Master Node.” [Online]. Available: <https://docs.unity3d.com/Packages/com.unity.shadergraph@8.1/manual/PBR-Master-Node.html>. [Accessed: Dec. 12, 2021]
- [42] Unity, “Animator Controller.” [Online]. Available: <https://docs.unity3d.com/es/530/Manual/class-AnimatorController.html#:~:text=Un%20asset%20Animator%20Controller%20es,ciertas%20condiciones%20de%20juego%20ocurren>. [Accessed: May 10, 2022]
- [43] Instituto Nacional del Cáncer, “Encéfalo.” [Online]. Available: <https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/encefalo>. [Accessed: Dec. 02, 2021]
- [44] Slideshare, “Organización funcional del cerebro.” [Online]. Available: <https://es.slideshare.net/abemen/organizacin-funcional-del-cerebro>. [Accessed: Dec. 02, 2021]
- [45] Travis Dixon, “fMRI: An important technological technique used to study the brain.” [Online]. Available: <https://www.themantic-education.com/ibpsych/2019/02/04/brain-imaging-all-about-fmri/>. [Accessed: Apr. 27, 2022]

- [46] J. Arias, P. Ciuciu, M. Dojat, F. Forbes, A. Frau-Pascual, T. Perret, J. Warnking, “PyHRF: A Python Library for the Analysis of fMRI Data Based on Local Estimation of the Hemodynamic Response Function,” in *15th SciPy*, 2017, pp. 34–40, doi: 10.25080/shinma-7f4c6e7-006.
- [47] “Lentes de Realidad Virtual Oculus Rift S.” [Online]. Available: <https://mcegamer.com/producto/oculus-rift-s-lentes-de-realidad-virtual/>. [Accessed: May 01, 2022]
- [48] Dario, “6DOF.” [Online]. Available: <https://proyectoidis.org/6dof/>. [Accessed: May 01, 2022]
- [49] “Resonancia Magnética Funcional.” [Online]. Available: <https://www.mri.cl/resonancia-magnetica-funcional/>. [Accessed: May 02, 2022]
- [50] “Cómo capturar tu pantalla en realidad virtual.” [Online]. Available: <https://stormseeker.xyz/como-capturar-tu-pantalla-en-realidad-virtual/>. [Accessed: May 03, 2022]



Anexo A. Tabla Resultados Mallados

A.1. Resultados Posprocesamientos de los Mallados

Tabla A.1: Resultados posprocesamientos DMN.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
PCC	28052	0.14	3874
Corteza Lateral Parietal Izquierda	16966	0.14	2344
Corteza Lateral Parietal Derecha	13688	0.13	1778
Hipocampo Izquierdo	1724	0.15	258
Hipocampo Derecho	4176	0.15	626
Giro Temporal Izquierdo	24380	0.20	2344
Giro Temporal Derecho	21788	0.15	3268
Corteza prefrontal medial (MPFC)	24593	0.25	6187
Giro Frontal Medio Izquierdo	22030	0.30	6655
Giro Frontal Medio Derecho	13352	0.15	2025
Giro Frontal Superior	21149	0.20	4315
Corteza prefrontal Dorsolateral Izquierda	19142	0.25	4784
Corteza prefrontal Dorsolateral Derecha	12368	0.15	1854

Tabla A.2: Resultados posprocesamientos Red Somatomotora.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Giro Poscentral Izquierdo	39344	0.45	17705
Giro Poscentral Derecho	40400	0.45	18180
Giro Precentral Izquierdo	31895	0.45	14353
Giro Precentral Derecho	27777	0.45	12500
Lóbulo Paracentral Izquierdo	10448	0.45	4702
Lóbulo Paracentral Derecho	12686	0.45	5709

Tabla A.3: Resultados posprocesamientos Red Visual.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Cuneus Izquierdo	14298	0.45	6434
Cuneus Derecho	14941	0.45	6724
Giro Lingual Izquierdo	9274	0.45	4172
Giro Lingual Derecho	9151	0.45	4116
Giro Fusiforme Izquierdo	10417	0.45	4686
Giro Fusiforme Derecho	13398	0.45	6029
Polo Occipital Izquierdo	692	0.45	310

Polo Occipital Derecho	899	0.45	403
Giro Occipital Superior Izquierdo	11691	0.45	5259
Giro Occipital Superior Derecho	14497	0.45	6523
Giro Occipital Medio Izquierdo	13139	0.45	5911
Giro Occipital Medio Derecho	11750	0.45	5286
Giro Occipital Inferior Izquierdo	2808	0.45	1263
Giro Occipital Inferior Derecho	2217	0.45	996

Tabla A.4: Resultados posprocesamientos Red Frontoparietal.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Corteza Anterior Cingulada (ACC)	13070	0.30	3921
Corteza Posterior Cingulada (PCC)	7615	0.45	3425
Giro Temporal Superior Derecho	9584	0.25	2396
Giro Temporal Superior Izquierdo	9268	0.30	2780
Giro Frontal Inferior Derecho	19943	0.35	6980
Giro Frontal Inferior Izquierdo	16773	0.40	6708
Giro Frontal Medio Derecho	44861	0.35	15700
Giro Frontal Medio Izquierdo	39911	0.40	15964
Corteza Prefrontal Medial (MPFC)	19134	0.30	5740
Corteza Parietal Posterior Derecha	19839	0.20	3967
Corteza Parietal Posterior Izquierda	18750	0.20	3750
Precuneus	15001	0.45	6749

Tabla A.5: Resultados posprocesamiento Red Límbica.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Amígdala Derecha	6529	0.35	2285
Amígdala Izquierda	8968	0.35	3137
Corteza Entorrinal Derecha	5845	0.35	2045
Corteza Entorrinal Izquierda	4387	0.35	1534
Hipocampo Derecho	11310	0.35	3958
Hipocampo Izquierdo	10690	0.35	3740
Ínsula Derecha	2474	0.35	864
Ínsula Izquierda	2233	0.35	781
Corteza Orbitofrontal Derecha	15941	0.35	5578
Corteza Orbitofrontal Izquierda	16847	0.35	7580
Corteza Perirhinal Derecha	5564	0.35	1946
Corteza Perirhinal Izquierda	5483	0.35	1918

Tabla A.6: Resultados posprocesamiento Red de Atención Dorsal.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Campos Oculares Frontales Derecha (FEF)	7426	0.45	3341
Campos Oculares Frontales Izquierda (FEF)	7652	0.45	3443
Lóbulo Lateral Parietal Derecho	21550	0.40	8620
Lóbulo Lateral Parietal Izquierdo	20425	0.40	8169
Lóbulo Lateral Occipital Derecho	5931	0.40	2372
Lóbulo Lateral Occipital Izquierdo	7940	0.40	3176
Corteza Temporal Superior Medial Derecha	20655	0.40	8261
Corteza Temporal Superior Medial Izquierda	21027	0.40	8410
Giro Frontal Medio Derecho	13464	0.45	6058
Giro Frontal Medio Izquierdo	12435	0.45	5594
Corteza Posterior Parietal Derecha	20837	0.40	8333
Corteza Posterior Parietal Izquierda	16381	0.40	6552

Tabla A.7: Resultados posprocesamientos Red de Atención Ventral.

Región Segmentada	Total Caras	Razón Diezmado	Total Caras Finales
Corteza Anterior Cingulada (ACC)	5086	0.40	2034
Giro Frontal Inferior Derecho	16529	0.35	5785
Giro Frontal Inferior Izquierdo	14174	0.35	4959
Giro Frontal Medio Derecho	10936	0.25	2734
Giro Frontal Medio Izquierdo	9384	0.25	2346
Giro Temporal Superior Derecho	14191	0.35	4966
Giro Temporal Superior Izquierdo	15236	0.35	5331
Lóbulo Parietal Inferior Derecho	11508	0.35	4027
Lóbulo Parietal Inferior Izquierdo	13166	0.35	4608
Corteza Prefrontal Medial (MPFC)	28210	0.40	11284
Corteza Posterior Cingulada (PCC)	17116	0.40	6845

A.2. Materiales Asignados a Cada Una de las Regiones que Componen las Redes en Estado de Reposo

Tabla A.8: Materiales asignados a cada región de la DMN

Default Mode Network	
Región	Color
Corteza Posterior Cingulada (PCC)	Rojo
Corteza Lateral Parietal	Azul
Giro Temporal	Amarillo
Giro Frontal Medio	Verde Oscuro
Giro Frontal Superior	Verde Claro
Hipocampo	Fucsia
Corteza Prefrontal Medial (MPFC)	Naranja
Corteza Prefrontal Dorsolateral	Morado

Tabla A.9: Materiales asignados a cada región de la red de Atención Dorsal

Red de Atención Dorsal	
Región	Color
Campos Oculares Frontales (FEF)	Rojo
Lóbulo Parietal Lateral	Naranja
Lóbulo Occipital Lateral	Verde
Corteza Temporal Superior Medial	Morado
Giro Frontal Medio	Celeste
Corteza Posterior Parietal	Azul

Tabla A.10: Materiales asignados a cada región de la red frontoparietal

Red Frontoparietal	
Región	Color
Corteza Anterior Cingulada (ACC)	Rosado
Corteza Posterior Cingulada (PCC)	Rojo
Giro Temporal Superior	Amarillo
Giro Frontal Inferior	Verde
Giro Frontal Medio	Morado
Corteza Prefrontal Media (MPFC)	Naranja
Corteza Parietal Posterior	Celeste
Precuneus	Café

Tabla A.11: Materiales asignados a cada región de la red Límbica

Red Límbica		
Región		Color
Amígdala		Verde
Corteza Entorrinal		Amarillo
Hipocampo		Naranja
Ínsula		Celeste
Corteza Orbito Frontal		Rosado
Corteza Perirhinal		Azul

Tabla A.12: Materiales asignados a cada región de la red Visual


Red Visual		
Región		Color
Cuneus		Azul
Giro Lingual		Rojo
Giro Fusiforme		Amarillo
Polo Occipital		Lila
Giro Occipital Superior		Fucsia
Giro Occipital Medio		Celeste
Giro Occipital Inferior		Naranja

Tabla A.13: Materiales asignados a cada región de la red de Atención Ventral

Red de Atención Ventral		
Región		Color
Corteza Anterior Cingulada (ACC)		Fucsia
Giro Frontal Inferior		Verde
Giro Frontal Medio		Lila
Giro Temporal Superior		Amarillo
Lóbulo Parietal Inferior		celeste
Corteza Prefrontal Media (MPFC)		Naranja
Corteza Posterior Cingulada (PCC)		Azul

Anexo B. Imágenes

Default Mode Network

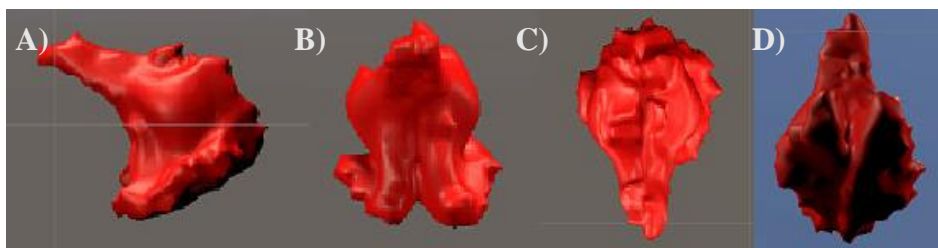


Fig. B.1: PCC con su material asignado.

En la imagen se observan distintas vistas de la malla de la corteza posterior cingulada en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.



Fig. B.2: Giro Temporal con su material asignado.

En la imagen se observan distintas vistas de la malla del giro temporal en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

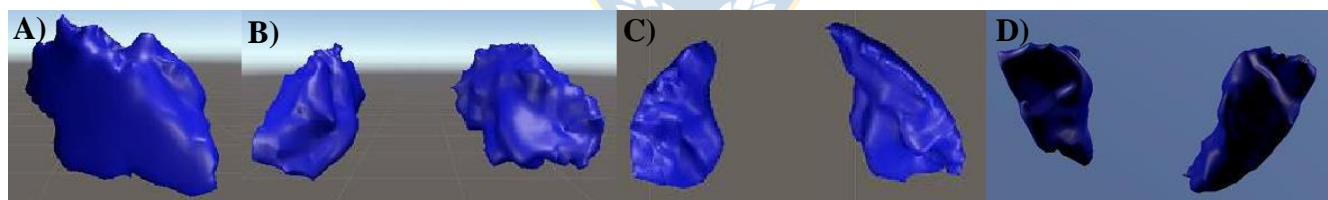


Fig. B.3: Corteza Lateral Parietal con su material asignado.

En la imagen se observan distintas vistas de la malla de la corteza lateral parietal en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.



Fig. B.4: Hipocampo con su material asignado.

En la imagen se observan distintas vistas de la malla del hipocampo en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

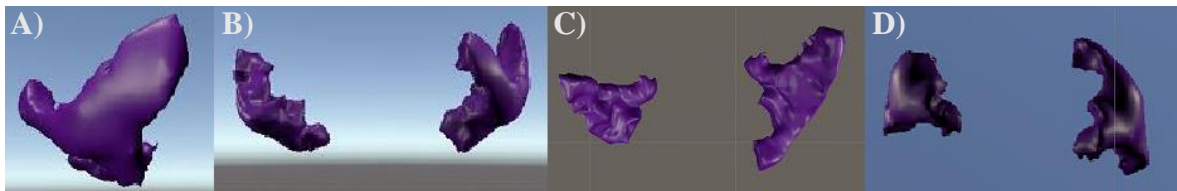


Fig. B.5: Corteza prefrontal dorsolateral con su material asignado.

En la imagen se observan distintas vistas de la malla de la corteza prefrontal dorsolateral en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

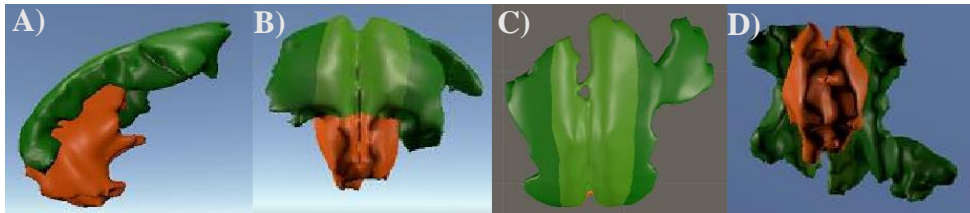


Fig. B.6: MPFC con su material asignado.

En la imagen se observan distintas vistas de la malla de la MPFC (naranja), el giro frontal superior (verde oscuro) y el giro frontal medio (verde claro) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

Red Visual

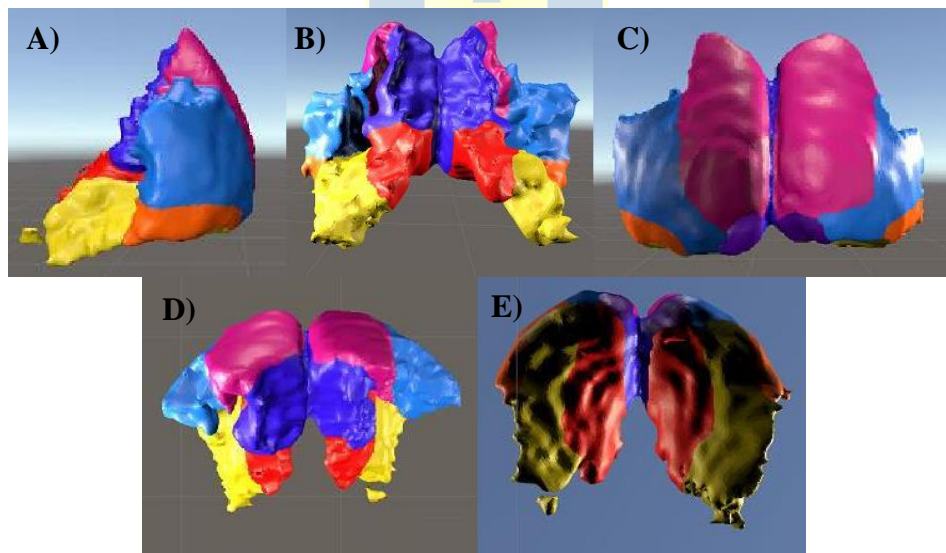


Fig. B.7: Red Visual con su material asignado.

En la imagen se observan distintas vistas de la malla del cuneus (azul oscuro), el giro fusiforme (amarillo), el giro lingual (rojo), giro occipital inferior (naranja), giro occipital medio (celeste), giro occipital superior (fucsia) y polo occipital (morado) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Anterior; D) Vista Superior; E) Vista Inferior.

Red Somatomotora

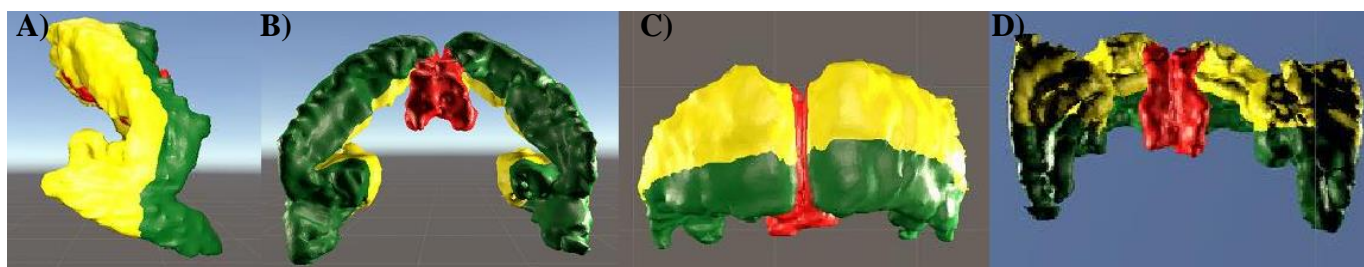


Fig. B.8: Red Somatomotora con su material asignado.

En la imagen se observan distintas vistas de la malla del giro precentral (verde), giro poscentral (amarillo) y el lóbulo paracentral (rojo) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

Red Atención Dorsal

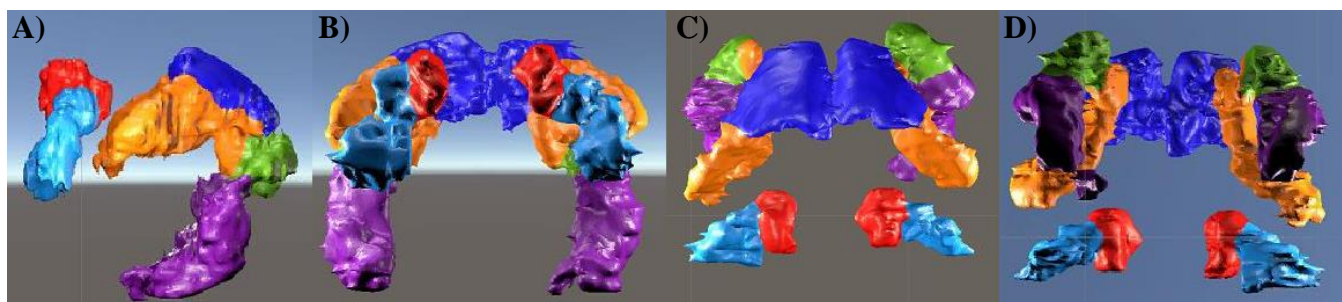


Fig. B.9: Red de Atención Dorsal con su material asignado.

En la imagen se observan distintas vistas de la malla de los campos oculares frontales (rojo), lóbulo lateral parietal (naranja), lóbulo occipital lateral (verde), corteza temporal superior medial (morado), giro frontal medio (celest) corteza posterior parietal (azul) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

Red Atención Ventral

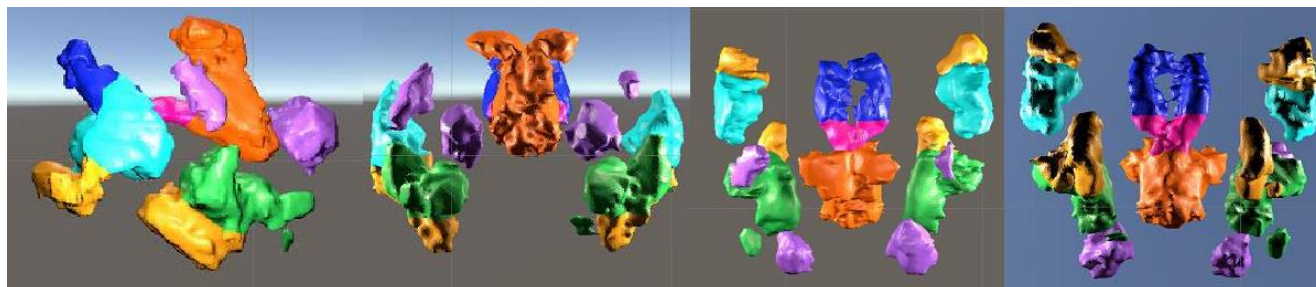


Fig. B.10: Red de Atención Ventral con su material asignado.

En la imagen se observan distintas vistas de la malla del ACC (fucsia), giro frontal inferior (verde), giro frontal medio (lila), giro temporal superior (amarillo), lóbulo parietal inferior (celest), MPFC (naranja) y PCC (azul) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

Red Límbica



Fig. B.11: Red Límbica con su material asignado.

En la imagen se observan distintas vistas de la malla de amígdala (verde), corteza entorrinal (amarillo), hipocampo (naranja), ínsula (celeste), corteza orbito-frontal (fucsia) y corteza perirhinal (azul) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

Red Frontoparietal

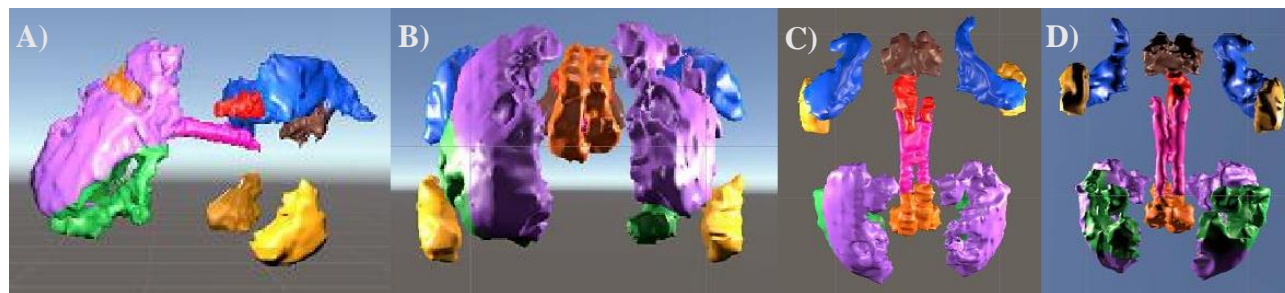


Fig. B.12: Red Frontoparietal con su material asignado.

En la imagen se observan distintas vistas de la malla de la corteza anterior cingulada (fucsia), corteza posterior cingulada (rojo), giro temporal superior (amarillo), giro frontal inferior (verde), giro frontal medio (lila), corteza prefrontal media (naranja), corteza parietal posterior (azul) y el precuneus (café) en Unity.

A) Vista Lateral; B) Vista Anterior; C) Vista Superior; D) Vista Inferior.

UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA
RESUMEN DE MEMORIA DE TITULO

Departamento : Departamento de Ingeniería Eléctrica
Carrera : Ingeniería Civil Biomédica
Nombre del Memorista : Javiera Alejandra Baeza Acuña
Título de la memoria : Aplicación de Realidad Virtual Sobre Redes en Estado de Reposo del Cerebro a partir de Imágenes de Resonancia Magnética Funcional
Fecha de la presentación oral : Jueves 25 de Agosto 2022

Profesor(es) Guía : Pamela Guevara Alvez
Profesor(es) Revisor(es) : Jaime Pinto y Esteban Pino
Concepto :
Calificación :

Resumen (máximo 200 palabras)

El objetivo principal de este proyecto es realizar una aplicación de realidad virtual que permita representar las redes en estado de reposo a partir de imágenes por resonancia magnética funcional (fMRI).

Para esto se utilizaron imágenes fMRI generadas por Thomas Yeo, quien realizó una parcelación de la corteza cerebral en 7 redes. Estos archivos fueron llevados a Python para separar cada una de las redes en archivos individuales. Luego, los elementos segmentados fueron importados en 3D Slicer para crear mallas 3D a partir de estos archivos. Después de obtener los modelos de malla de cada red en estado de reposo, se segmentaron y posprocesaron las mallas.

El desarrollo de la aplicación se hizo en Unity. Se implementaron tres escenas: menú principal, instrucciones y principal. En esta última se puede visualizar la activación/desactivación cerebral, el usuario puede interactuar con objetos en la sala, y se entrega información importante sobre la app y las redes en estado de reposo.

La aplicación fue probada usando lentes Oculus Rift S. Se verificó el buen funcionamiento de los distintos componentes de la aplicación y la correcta visualización de las redes en estado de reposo.