



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**Simulación computacional de cortes en  
objetos rígidos con retroalimentación  
háptica para entrenamiento en  
microneurocirugía**

**TESIS**

Que para obtener el título de  
**Ingeniero en Computación**

**P R E S E N T A**

Mauricio Garduño Magaña

**DIRECTOR DE TESIS**

M. en I. Sergio Teodoro Vite



**Ciudad Universitaria, Cd. Mx., 2018**

# *Agradecimientos*

*Quisiera expresar mi gratitud a través de estas líneas a las personas cuya colaboración fue de gran importancia para la elaboración de este trabajo. Agradezco al CCADET, a la UIDT y en especial al M. en I. Sergio Teodoro Vite, quien fue mi profesor y posteriormente mi director, su orientación, seguimiento y motivación fueron piezas fundamentales para el desarrollo y culminación de este trabajo.*

*Deseo reconocer el interés y las sugerencias recibidas por parte del Dr. Miguel Ángel Padilla Castañeda respecto a mi trabajo, también, agradezco al Dr. Aldo Francisco Hernández Valencia por su colaboración y retroalimentación.*

*No hay palabras suficientes para agradecer a las siguientes personas, su comprensión, tiempo y amistad fueron, son y serán apreciadas por mi persona, agradezco a mis amigos quienes me ayudaron emocionalmente, en especial a Iván y a Daniel. Agradezco a mi papá Arturo, a mi mamá Gloria, y a mi hermano Arturo, todo esto no hubiera sido posible sin su apoyo incondicional y cariño.*

*Les agradezco de todo corazón.*

## Contenido

CAPÍTULO 1 Introducción .....	1
1.1 Ambientes Virtuales aplicados a medicina .....	1
1.1.1 Metaverso de la virtualidad .....	2
1.1.2 Ambientes Virtuales .....	3
1.1.3 Simulación en medicina.....	4
1.2 Definición del problema .....	6
1.3 Objetivo .....	7
1.4 Organización de la tesis.....	7
CAPÍTULO 2 Craneotomía .....	8
2.1 Clasificación.....	9
2.1.1 Craneotomías supratentoriales .....	10
2.1.2 Craneotomías infratentoriales .....	10
2.1.3 Craneotomías combinadas supra e infra tentoriales .....	11
2.1.4 Ejemplos .....	12
2.2 Instrumentos quirúrgicos .....	16
2.2.1 Microscopio.....	16
2.2.2 Instrumentación microquirúrgica .....	17
2.3 Entrenamiento .....	19
CAPÍTULO 3 Reconstrucción de modelos 3D a partir de imágenes de CT .....	23
3.1 Software para la reconstrucción de modelos .....	23
3.1.1 OsiriX.....	24
3.1.2 Blender .....	25
3.2 Reconstrucción a partir de imágenes de CT .....	25
3.2.1 Escáner CT.....	25
3.2.2 Unidades y escala de Hounsfield.....	27
3.2.3 Reconstrucción de imágenes .....	28
3.3 Técnicas de mallado a partir de volúmenes de imágenes .....	30
3.3.1 Volume rendering .....	30
3.3.2 Datos volumétricos.....	31
3.3.3 Clasificación de algoritmos de Volume Render .....	32

3.4 Visualización y refinamiento.....	36
CAPÍTULO 4 Simulador de entrenamiento para craneotomía .....	39
4.1 Metodología .....	39
4.2 Carga de modelos .....	43
4.3 Posicionamiento.....	45
4.4 Marcaje (Interacción háptica).....	46
4.5 Generación del modelo marcado .....	51
4.6 Algoritmo de corte.....	52
4.7 Generación de modelo resultante tras aplicarse el corte .....	55
4.8 Carga de interfaz, visualización y métrica .....	56
4.9 Audio.....	58
CAPÍTULO 5 Pruebas y resultados .....	59
CAPÍTULO 6 Conclusiones y trabajo a futuro .....	65
Referencias bibliográficas .....	67
Tabla de Figuras.....	70

# CAPÍTULO 1

## Introducción

### 1.1 Ambientes Virtuales aplicados a medicina

Los ambientes virtuales hoy en día han encontrado un amplio campo de aplicación en diversas áreas del conocimiento, en parte, gracias al crecimiento acelerado de las tecnologías de la información y a la alta disponibilidad de dispositivos de hardware para la comunicación y el procesamiento de datos. Estos campos incluyen a la medicina, la física, las ciencias de la tierra, la aeronáutica, astronomía, entre muchas otras. Este fenómeno ha dado lugar a un concepto muy genérico, que engloba a las tecnologías que coexisten en el mundo de aplicaciones reales y aquéllas que se representan mediante una computadora, llamado “metaverso de la virtualidad”.

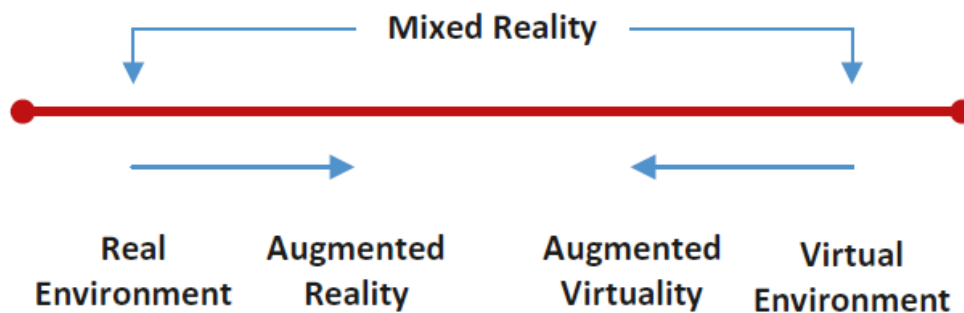


Figura 1. Continuo de la Realidad-Virtualidad de Milgram (1994). Imagen extraída de (Peddie, 2017, p. 13).

### 1.1.1 Metaverso de la virtualidad

El metaverso de la virtualidad es la convergencia entre la realidad física y una realidad virtual, que se representan como una línea continua, sobre la cual se relacionan tecnologías definidas por los medios físicos de interacción entre el usuario y la máquina. Así, podemos decir, que los intermedios de ese medio continuo forman una realidad mezclada, compuesta por las tecnologías de realidad aumentada y virtualidad aumentada, como se muestra en la Figura 1.

- Realidad física (Figura 2a): En este extremo, se incluyen los simuladores que trabajan con maniqués HPS (*Human Patient Simulator*, por sus abreviaturas en inglés), y que tienen por objetivo representar casos específicos para diversos fines, como puede ser el entrenamiento.
- Realidad Aumentada (Figura 2b): Se trata de una tecnología, donde dominan los componentes del mundo real y sobre el cual se le sobreponen elementos virtuales. Generalmente estos sistemas trabajan con un elemento de visión, como puede ser una cámara web o un teléfono inteligente, y un elemento distintivo, llamado marcador, sobre el cual se ubica el modelo tridimensional virtual.
- Virtualidad Aumentada (Figura 2c): Es una tecnología que ha significado la transición entre el mundo virtualizado y el aumentado, en el cual dominan los elementos virtuales sobre los reales. Este tipo de sistemas toman elementos del mundo real usando dispositivos de seguimiento, para incorporar elementos de interacción del mundo real y sobreponernos en el mundo virtual.
- Ambientes virtuales (Figura 2d): Engloba las tecnologías que abstraen los elementos del mundo real y los convierten en un ambiente totalmente artificial generado por computadora. En este sentido, el usuario pierde noción de la interacción con el mundo real para transportarse a un espacio digital con visualizaciones 3D y otros estímulos, como puede ser el táctil y el aural.

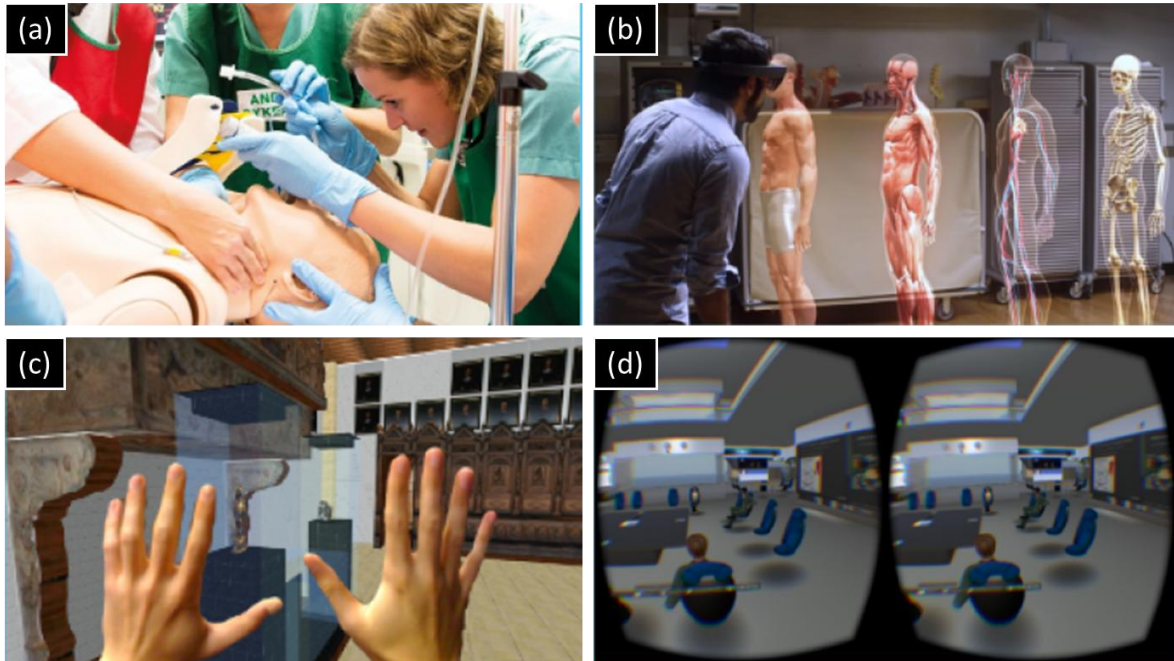


Figura 2. Ejemplo de aplicaciones del metaverso de la realidad-virtualidad. (a) SimMan3D, Laerdan, Norway, USA, Canada. (b) Microsoft HoloLens. (c) Poster: A Virtual Body for Augmented Virtuality by Chroma-Keying of Egocentric Videos. Frank Steinicke. (d) Second Life con el uso de Oculus Rift.

### 1.1.2 Ambientes Virtuales

Un ambiente virtual es un medio compuesto por simulaciones de eventos o sucesos que encontramos en la vida cotidiana o significan una representación de un espacio imaginario. Shannon y Johannes (Shannon & Johannes, 1976) definen a la simulación como “el proceso de diseñar un modelo de un sistema real y realizar experimentos con éste, con la finalidad de comprender el comportamiento del sistema o evaluar nuevas estrategias para su funcionamiento”. Asimismo, el propósito de un ambiente virtual es llevar estos eventos a un entorno controlado, lo cual permite que en este medio virtual se puedan poner en práctica diversos experimentos y pruebas para así poder realizar un estudio completo de los efectos generados, sin la necesidad de replicarlo en un ambiente real.

Una característica principal de los ambientes virtuales es el concepto de inmersión, una propiedad relacionada con la sensación de estar presente o experimentar un evento de manera realista. La inmersión se consigue al tener el

mayor número de sentidos estimulados de forma adecuada; por ejemplo, mediante estímulos visuales y táctiles a la vez.

Las aplicaciones de los ambientes virtuales, como se mencionó anteriormente, se ha venido dando en diversos campos; de acuerdo con Riva (Giuseppe Riva, 2000), los avances tecnológicos enfocados en ambientes virtuales son mejor aprovechados en el campo médico que en cualquier otra área, esto es debido a que las tecnologías como la robótica, el procesamiento de imágenes, el cómputo gráfico, la telepresencia, entre otras, se están convirtiendo en técnicas familiares para el diagnóstico, terapia, enseñanza y capacitación de personal médico.

### **1.1.3 Simulación en medicina**

Los simuladores aplicados a la medicina, principalmente para entrenamiento médico, son una herramienta muy práctica, debido a su alta disponibilidad en comparación con cirugías reales, las cuales involucran interacciones complejas y conllevan riesgos de mortalidad o de daño permanente al paciente. Debido a esto, los simuladores médicos actuales se enfocan en el entrenamiento de padecimientos de alta incidencia, que ayuden a los aprendices a enfrentar situaciones de emergencia.

Un punto clave para el desarrollo de un simulador computarizado enfocado al entrenamiento médico, es la capacidad de entregar al usuario inmersión y presencia. La inmersión está directamente relacionada con el ambiente virtual y al cómo es que el usuario lo percibe durante una simulación; mientras que la presencia es la capacidad del usuario para interactuar con el ambiente de manera realista y convincente (Teodoro et al., 2017, p. 2).

Para los médicos y cirujanos, el objetivo final de usar ambientes virtuales es la representación de elementos reales en un sistema parcial o totalmente computarizado de una manera idéntica a su contraparte natural. La característica clave de los entornos virtuales para los profesionales es el alto nivel de control de la interacción con la herramienta sin las limitaciones habitualmente encontradas en



los sistemas informáticos y la experiencia enriquecida proporcionada al paciente (G Riva, 2003, p. 525).

Las aplicaciones relacionadas con la cirugía se dividen principalmente en cinco usos: entrenamiento quirúrgico, diagnóstico, planificación, asistencia y control postoperatorio. Para el caso de entrenamiento, los ambientes virtuales pueden ser usados como herramientas didácticas y experimentales; por ejemplo, la enseñanza de la anatomía es principalmente ilustrativa, donde los usos de aplicaciones de visualización poseen un gran potencial como medio de aprendizaje. A través de la representación tridimensional de volúmenes masivos de información y bases de datos, los médicos y los estudiantes pueden entender importantes principios fisiológicos o anatomía básica (Alcañiz, Botella, Perpiñá, Baños, & Montesa, 2000).

Existen muchos ejemplos de sistemas que emplean la realidad virtual y la robótica para simular condiciones médicas diversas (Teodoro et al., 2017, p. 1). Algunos ejemplos son el simulador de cirugía de la próstata de la UNAM, México (Padilla et al., 2009), el simulador de reparación de aneurismas cerebrales para entrenamiento médico (Luengas C., López, & Sánchez, 2017, p. 513 - 525), el simulador de cirugía para aneurismas cerebrales de la Universidad de Illinois, EE.UU. (Alaraj et al., 2015), y *NeuroTouch*, del Consejo Nacional de Investigación de Canadá, este último tiene como objetivo desarrollar un simulador para procedimientos basados en craneotomía con retroalimentación háptica y gráfica para su implementación por universidades y hospitales para entrenamiento de neurocirugía (Delorme, Laroche, DiRaddo, & Del Maestro, 2012).

Uno de los propósitos del entrenamiento virtual, es el establecer un nuevo estándar para la enseñanza de la medicina a través del manejo de recursos tecnológicos, al tiempo que se obtienen ventajas respecto a otros métodos de aprendizaje, ya que no se compromete infraestructura médica y, por lo tanto, se afianza la seguridad del paciente, y garantiza un bajo costo relativo al entrenamiento al no requerir tiempo de quirófano (Teodoro et al., 2017, p. 1).

## 1.2 Definición del problema

Un estudio realizado por Vickers et al. en su libro "*The surgical learning curve for laparoscopic compared to open radical prostatectomy: a retrospective cohort study*" concluyó que se necesitan alrededor de 750 operaciones para mejorar la habilidad en el desarrollo de procedimientos quirúrgicos por parte de un médico; en este mismo estudio se encontró que los pacientes que fueron operados por médicos con una experiencia de entre 750 y 10,250 procedimientos tendieron a presentar menos complicaciones durante la cirugía, en comparación con un médico que poseía menos experiencia. Esto significa que los médicos requieren de una preparación muy extensiva para poder desarrollarse de manera exitosa durante un proceso quirúrgico. Sin embargo, para que esto suceda, se requiere que los aprendices posean las condiciones de experimentación adecuadas con ambientes lo más cercanos a el procedimiento quirúrgico a realizar, lo que en muchas ocasiones resulta en limitaciones de accesibilidad en muchos sentidos. Por ejemplo, la práctica se ve limitada por el número de veces en que los residentes pueden asistir o entrenarse en una cirugía in-vivo, por ello, en ocasiones los médicos recurren a ensayos sobre cadáveres y prácticas sobre animales. En este sentido, un ambiente virtual permite realizar un número ilimitado de pruebas sin restricciones de materiales e infraestructura, entre otros.

En el Hospital General de México, particularmente en la Unidad de Neurología y Neurocirugía, surge la necesidad de contar con un simulador para el entrenamiento médico para craneotomía, debido a que es un procedimiento de alta incidencia y que representa un paso importante para el tratamiento quirúrgico de varios padecimientos, tales como: remoción de tumores, reparación de aneurismas cerebrales y hematomas. Por lo tanto, en esta tesis, se propone el diseño, desarrollo y validación de un sistema de realidad virtual para el entrenamiento médico en la especialidad de micro-neurocirugía, con especial énfasis en el procedimiento invasivo de corte en hueso en las regiones fronto-parietal y temporal del cráneo (también conocida como craneotomía abierta y que se aplica al 90% de las cirugías), usando tecnologías de realidad virtual con retroalimentación háptica, con un

enfoque médico para poder lograr un buen grado de inmersión durante una simulación.

### **1.3 Objetivo**

El objetivo principal de este trabajo de tesis es el desarrollo de un simulador computarizado para el entrenamiento médico en el procedimiento de corte de hueso craneal (craneotomía abierta). Con la ayuda de este sistema, se pretende hacer que el neurocirujano se instruya, refuerce o mejore sus habilidades, de manera que, ante un eventual procedimiento real, se encuentre preparado para ejecutar de manera óptima, correcta y adecuada el procedimiento quirúrgico.

### **1.4 Organización de la tesis**

Este trabajo está estructurado de la siguiente forma:

En el capítulo 2, se profundizará con el tema del procedimiento de corte de hueso craneal, conocido como craneotomía, en el cual se describe el proceso quirúrgico y los elementos involucrados, tales como instrumental quirúrgico y preparativos clínicos. En el capítulo 3, se describe una técnica de reconstrucción de modelos tridimensionales a partir de imágenes de angio-tomografía computarizada de estructuras óseas, principalmente cráneo, que se empleó para la generación de modelos de prueba en el ambiente virtual. También se abordan las herramientas de software empleadas para la obtención de dichos modelos. En el capítulo 4, se reporta la metodología completa del desarrollo del simulador de craneotomía, su arquitectura general, el flujo del programa, algoritmos de corte y otras técnicas empleadas para la visualización del sistema. Finalmente, en los capítulos 5 y 6 se presentan las pruebas y resultados respectivamente, obtenidos tras las sesiones de prueba del sistema completo en el Hospital General de México, donde los especialistas y aprendices en neurocirugía hicieron uso del simulador para dar retroalimentación de su funcionamiento.

# CAPÍTULO 2

## Craneotomía

La craneotomía es un proceso quirúrgico en donde una sección del cráneo es temporalmente removida para tener acceso a una sección específica del cerebro. Se trata regularmente de una operación crítica realizada en pacientes que sufren de lesiones cerebrales o traumatismos en el área encéfalo craneana. Este procedimiento ha evolucionado a lo largo del tiempo para minimizar el área del corte óseo y maniobrar sobre las áreas de interés con menor daño. Sumado a esto, permitir en algunos casos, alcanzar con seguridad áreas profundas y al mismo tiempo afectar lo mínimo posible la estética craneofacial.

Generalmente una craneotomía debe ser precedida por un estudio que proporciona una imagen del cerebro (una tomografía computarizada o resonancia magnética, por ejemplo), que ayudará al cirujano a planear el acceso en una ubicación precisa para remover parte del hueso en un ángulo apropiado. El colgajo<sup>1</sup> óseo que debe ser retirado dependerá del problema que se requiere tratar y de la aproximación que se empleará.

Respecto al área craneal que se separa quirúrgicamente, habitualmente se les llama “craneotomías osteoplásticas”, cuando se conserva el colgajo óseo para reponerlo durante el cierre de la operación y su técnica no implica la exéresis<sup>2</sup> suplementaria de otras partes craneales, en su lugar temporalmente se ubican placas de algún metal como el titanio y tornillos u otra forma de fijación (alambre,

---

<sup>1</sup> Colgajo: Esta palabra es la traducción a secas de *flap* la cual se usó en repetidas ocasiones en las lecturas que sirvieron de base para esta redacción. Hace referencia a dos cosas, cierta área excedente que puede colgar, como lo es el corte de piel realizado en las craneotomías; el área ósea cortada, *bone flap*, también se traducía como *solapa* o *aleta*.

<sup>2</sup> Exéresis. Extirpación. Extirpación de un órgano o de un tejido corporal.

sutura, etc.) y luego se reintegra intacta la parte del cráneo, momento en el cual se retiran las placas. Son nombradas “craniectomías osteoclásticas”, cuando no se cierra la ventana ósea, o si se emplea la técnica de extirpar el hueso pedazo a pedazo, con una gubia u otro instrumento con acción similar, desechando posteriormente las piezas. Algunas craneotomías pudieran ser llamadas mixtas, pues tienen elementos osteoplásticos y osteoclásticos como parte de su descripción técnica (Pereira Riveron, 2007, p. 2).

## 2.1 Clasificación

Por la localización de la craneotomía y del objetivo intracraneal en relación con la tienda del cerebelo<sup>3</sup>, las craneotomías se pueden clasificar como supratentoriales, infratentoriales o combinadas. Con relación al plano medio sagital craneal, pueden definirse como unilateral (que abarca sólo un lado) o bilaterales (que abarca ambos lados). En atención a su altura en el cráneo, muchos clasifican las craneotomías en:

- Basales: porque son ubicadas en las regiones inferiores del cráneo y generalmente incluyen el hueso de las fosas anterior, media o posterior
- Altas: ya que se realizan en los alrededores de la sutura sagital craneal
- Intermedias: son localizadas entre las dos mencionadas anteriormente o
- Mixtas: son más amplias y cada vez menos favorecidas.

La amplitud del corte craneal dependerá del objetivo de la operación. En ocasiones es suficiente que el tamaño de la apertura de la piel sea alrededor de 5 [mm] y la perforación craneal sea mínima (2 a 4 [mm]) y realizada con una barrena o craneótomo. Otras veces será suficiente un agujero de trépano<sup>4</sup> realizado con

---

<sup>3</sup> La tienda del cerebelo, *tentorium cerebelli* o *cerebellar tentorium*, es una extensión de la duramadre que separa el cerebelo de la porción inferior del lóbulo occipital del cerebro.

<sup>4</sup> Trepano: Especie de taladro quirúrgico pequeño con una sierra circular para perforar los huesos, especialmente los del cráneo.

puntas y fresas, o con una trefina<sup>5</sup> de 20 ó 25 [mm] de diámetro (Pereira Riveron, 2007, p. 2 - 3).

### **2.1.1 Craneotomías supratentoriales**

Son diseñadas para acceso a lesiones en las áreas encefálicas ubicadas sobre la tienda del cerebelo y permiten entradas anteriores, laterales, posteriores, superiores o inferiores.

Las craneotomías supratentoriales para accesos anteriores y anterolaterales, generalmente requieren incisiones de partes blandas en arco, unilaterales o bilateralmente y por detrás de la línea de implantación del cabello. La apertura ósea incluye generalmente a los huesos frontal y esfenoides, así como el temporal y la parte anteroinferior del parietal, para permitir al cirujano diferentes opciones de ángulos para visualización.

Estas craneotomías pueden ser empleadas en lesiones cerebrales frontales y algunas ubicadas en el lóbulo temporal. También son útiles para tratar lesiones de la región quiasmática<sup>6</sup>, selar y supraselar del cerebro, base craneal anterior y hasta las partes anteriores de la fosa media (estas partes forman parte principalmente el área delantera de la cabeza) (Pereira Riveron, 2007, p. 4).

### **2.1.2 Craneotomías infratentoriales**

Son diseñadas para tratar lesiones axiales o extra-axiales en la fosa craneal posterior, es decir, en estructuras ubicadas por debajo de la tienda del cerebelo. Algunas de estas lesiones pueden formar parte, o estar incorporadas dentro del tejido de las estructuras del sistema nervioso central (cerebelo y tronco cerebral).

El tipo de lesión y su localización que aborda esta craneotomía son, por ejemplo: línea media, hemisferio cerebeloso, tronco cerebral, región del *clivus*<sup>7</sup>,

---

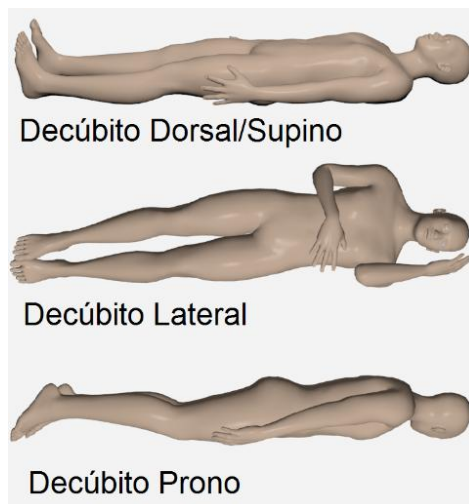
<sup>5</sup> Trefina: Es un instrumento quirúrgico con una hoja cilíndrica. Puede ser de una o de varias dimensiones y diseños dependiendo del uso para el que va a ser utilizado.

<sup>6</sup> El quiasma óptico: Es la parte del cerebro donde se entrecruzan parcialmente las fibras axónicas de los nervios ópticos.

<sup>7</sup> Clivus: En anatomía humana, recibe el nombre de clivus una formación ósea situada en la base del cráneo. Está formado por la unión del cuerpo del hueso esfenoides con la porción media del hueso occipital.

agujero occipital y su área colindante, tienda del cerebelo, tronco basilar o arterias cerebelosas; éstas determinarán el tipo y extensión de la craneotomía. La forma osteoclástica es muy favorecida en esta región, por ser más difícil en ella la preparación y manipulación del colgajo óseo (Pereira Riveron, 2007, p. 21 - 22).

La posición del paciente en una craneotomía juega un papel importante y representa un paso previo al proceso de corte de hueso. La posición más común suele ser en decúbito prono con flexión craneal para exponer bien el área de la fosa superior, o también en decúbito lateral, como se muestra en la Figura 3. Cuando el objetivo de la operación es lateral, el paciente puede colocarse en decúbito supino y la cabeza se rota contra lateralmente y se flexiona, para facilitar el acceso (Pereira Riveron, 2007, p. 21 - 22).



*Figura 3. Posiciones del paciente para un procedimiento de craneotomía: decúbito dorsal, supino, decúbito lateral y decúbito prono. Poses realizadas en Blender con ayuda de modelos generados con MakeHuman.*

### **2.1.3 Craneotomías combinadas supra e infra tentoriales**

En ocasiones se tratan lesiones que se extienden tanto por encima como por debajo de la tienda del cerebelo, o que están en la parte anterior del tronco cerebral; en el *clivus*; o en el borde anterior de la tienda del cerebelo. En esos casos puede ser necesaria una craneotomía combinada o simultánea supra e infratentorial; que incluya la fosa posterior lateral y las regiones laterales suboccipital y subtemporal (Pereira Riveron, 2007, p. 26).

### 2.1.4 Ejemplos

A continuación, se describen algunos ejemplos de las clasificaciones previamente mencionadas.

- La *craneotomía pterional, frontopeterional* (Yasargil, 1984, p. 215 - 233). Tiene por objetivos, el tratamiento microquirúrgico de los aneurismas, así como tumores. Esta aproximación es la más común en realizarse ya que aproximadamente el 90% de las craneotomías se realizan en esta zona. Se coloca la cabeza del paciente en una posición óptima, en la que generalmente el punto más alto es el pómulo acorde al lado de la cirugía. No es una craneotomía completamente osteoplástica, es mixta, pues, aunque se conserva el colgajo óseo, se requiere la exéresis de la parte externa del risco esfenoidal, parte del hueso cigomático y del techo orbitario. Yasargil (Yasargil, 1984) menciona que se deben realizar cuatro agujeros de trépano para obtener una pieza libre de hueso, pero Lawton (Lawton, 2011, p. 29 - 31) sugiere un solo agujero de trépano y el resto del corte que sea realizado por un craneótomo o alguna otra herramienta de corte como se muestra en la Figura 4, con esto se evita menos daño al colgajo óseo para que su estado final sea mejor estéticamente.

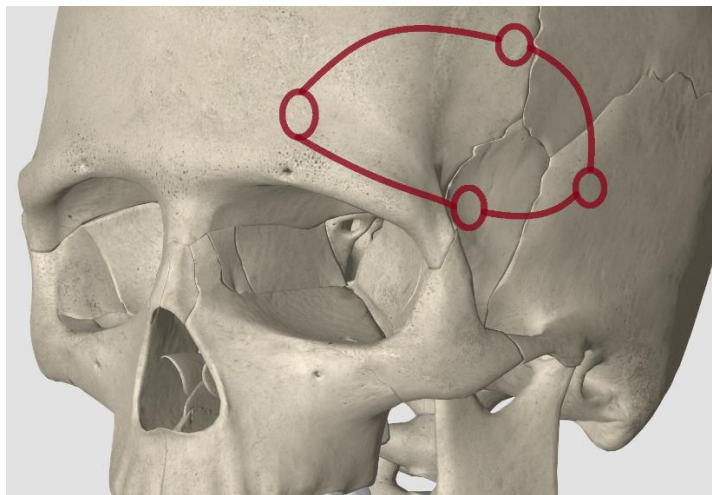
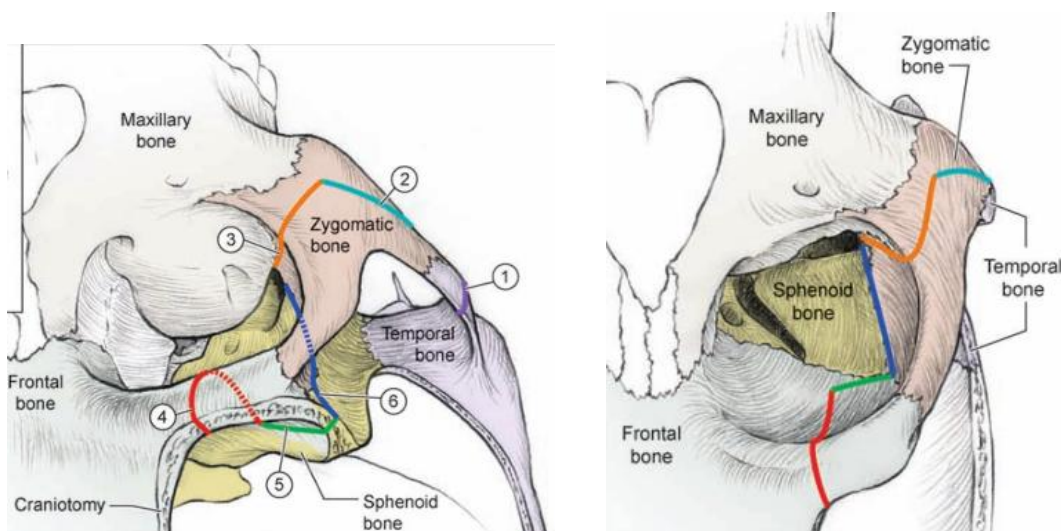


Figura 4. Craneotomía Frontotemporal de Yasargil. En la craneotomía se hacen 4 agujeros de trépano para obtener la pieza del cráneo que se removerá. Cráneo generado con el software ORGANON.



- La craneotomía por *aproximación orbitocigomática* (Lawton, 2011, p. 32 - 36), emplea la misma posición del paciente que en la craneotomía pterional, para exponer la unidad orbitocigomática y extraerla. Esta unidad se extrae de varias maneras:
  1. Con el colgajo óseo como una pieza integrada. Esta forma da un mejor resultado estético, pero la osteotomía es más difícil de realizar.
  2. Separación del colgajo óseo como dos piezas. Es la técnica estándar y más recurrida.
  3. Separación del colgajo óseo como dos piezas, con una unidad orbital modificada que no incluye al hueso cigomático. Es usada cuando no tiene sentido la exposición de la resección cigomática.

La unidad orbitocigomática es liberada por una serie de seis osteotomías hechas con una sierra de vaivén como se puede visualizar en la Figura 5. Con la órbita y el hueso cigoma eliminados, se abre un amplio corredor a las cisternas silviales y carotídeas.



*Figura 5. Osteotomías Orbitocigomáticas. La unidad orbitocigomática es liberada por una serie de 6 osteotomías en: (1) Raíz cigomática, (2) Cuerpo cigomático, (3) Fisura orbital inferior, (4) Techo orbital medio, (5) Techo orbital posterior y (6) Pared orbital lateral. Imagen extraída de (Lawton, 2011, p. 35). Osteotomías Orbitocigomáticas. Diferente ángulo de las 6 osteotomías. Imagen extraída de (Lawton, 2011, p. 35).*

- La craneotomía de *aproximación anterior interhemisférica* (Lawton, 2011, p. 37 - 39), se usan para tratar aneurismas. Por esta razón, la fisura interhemisférica anterior es abordada sólo en un lado, y la craneotomía es enfocada en ese lado. El lado derecho se elige en estos aneurismas para evitar complicaciones resultantes de la craneotomía sobre el hemisferio dominante. En la Figura 6A, podemos ver la posición del paciente en esta aproximación y en la Figura 6B podemos ver el área marcada que será la guía para la incisión de piel y de hueso. La craneotomía generalmente se hace con un solo agujero de trépano, que cruza la línea media y expone el seno sagital superior. El corte se extiende hacia abajo hasta el suelo de la fosa craneal anterior, y lateralmente a la línea temporal superior. El corte posterior se extiende hasta la sutura coronal (Figura 6C). La tabla interna del hueso frontal en la línea media inferior se perfora hasta quedar plana, de manera análoga a la perforación del hueso esfenoides para un abordaje *pterional*. Esta eliminación adicional de hueso aumenta la visualización a lo largo del piso de la fosa craneal anterior para acceder al control proximal de los aneurismas de la arteria pericallosal de baja altura (Figura 6D).

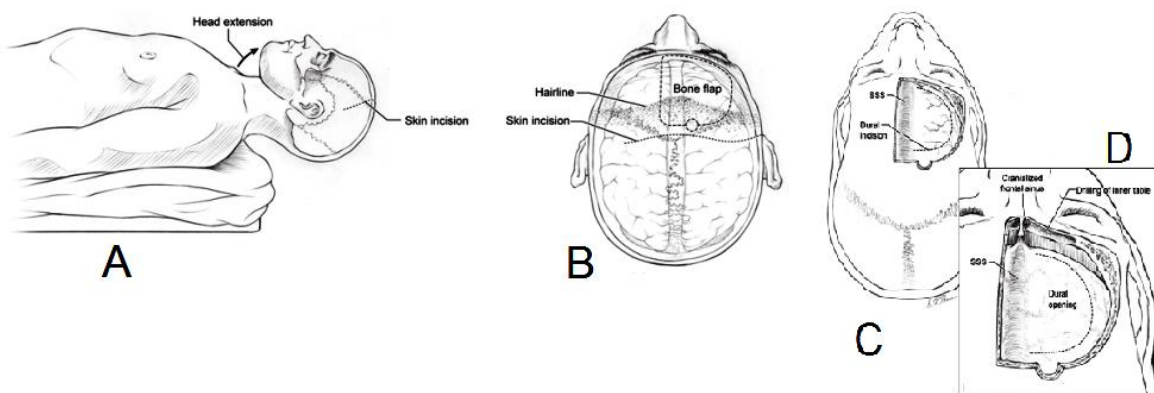


Figura 6. Existe una modificación en el proceso quirúrgico anterior el cual se usa para abordar los aneurismas de la arteria pericallosal en una región diferente, el enfoque *interhemisférico anterior modificado* utiliza la gravedad para retraer el hemisferio derecho. Imagen extraída de (Lawton, 2011, p. 38).

A diferencia de su versión original la cual indica que el paciente debe estar recostado en postura decúbito supino (Ver Figura 3) y el cuello extendido de manera suave, esta modificación requiere que el paciente este recostado de manera supino, pero se hace uso de refuerzos (tipo cojines) posicionados debajo del hombro izquierdo, la cabeza es rotada 90 grados a la derecha con la línea media sagital paralela al piso y rotado 45 grados hacia arriba (el posicionamiento se encuentra de manera ilustrada en la Figura 7).

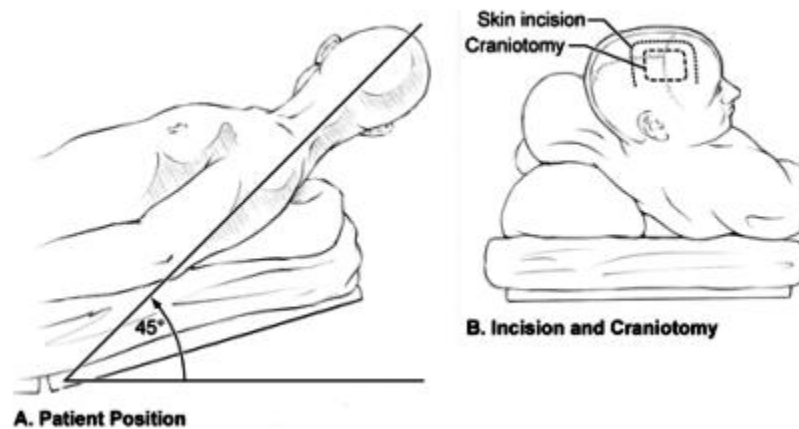
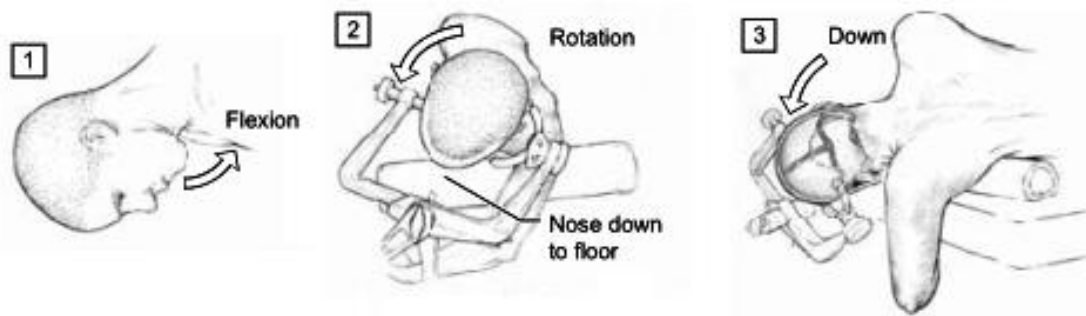


Figura 7. Posición del paciente recostado en decúbito supino, se hace uso de refuerzos bajo el hombro izquierdo para poder levantarlo 45 grados y la cabeza es rotada 90 grados a la derecha. Imagen extraída de (Lawton, 2011, p. 38).

- La craneotomía de *aproximación de enfoque distante*, también denominada de abordaje suboccipital lateral, cercamiento lateral extremo y exposición transcondilar extrema lateral (Lawton, 2011, p. 40 - 42). Se utiliza posición prona de tres cuartos con el paciente colocado con el lado de la lesión hacia arriba. El brazo dependiente cuelga sobre el extremo de la mesa, acunado en un refuerzo acolchonado. Tres maniobras posicionan la cabeza óptimamente: flexión hasta que el mentón esté a un dedo del esternón; rotación de 45 grados lejos del lado de la lesión, trayendo la nariz abajo hacia el piso; y la flexión lateral 30 grados hacia abajo hacia el suelo. Estas maniobras sitúan el *clivus* perpendicular al suelo, permitiendo al neurocirujano mirar el eje de las arterias vertebrales y basilares y así trabajar

entre los nervios craneales inferiores. En la Figura 8 podemos ver las tres maniobras realizados para el posicionamiento adecuado de la cabeza.



*Figura 8. Tres maniobras posicionan la cabeza óptimamente: flexión en el plano anteroposterior hasta que el mentón esté a un dedo del esternón; rotación de 45 grados lejos del lado de la lesión, trayendo la nariz abajo hacia el piso, y la flexión lateral 30 grados hacia abajo hacia el suelo. Imagen extraída de (Lawton, 2011, p. 41).*

## 2.2 Instrumentos quirúrgicos

### 2.2.1 Microscopio

Dos principios ópticos, la ampliación y la perspectiva estereoscópica, son importantes para el neurocirujano relacionados con el microscopio que usarán para la visualización interna durante el procedimiento quirúrgico. La importancia del microscopio quirúrgico en lo que se refiere a los procedimientos neurológicos no es la ampliación que proporciona, sino la clara perspectiva visual que proporciona según Yasargil (Yasargil, 1984, p. 208).

- Ampliación. El agrandamiento de los objetos en el campo operatorio es la función más reconocida, pero en realidad, es la menos importante del microscopio quirúrgico. Los principios ópticos relacionan la ampliación final obtenida a través de cualquier microscopio, con la lente de aumento y la ampliación de las piezas oculares.
- Perspectiva estereoscópica. En algunos procedimientos neuroquirúrgicos se realizan en la superficie del campo operatorio, y en éstos la ampliación y la profundidad de campo son consideraciones primarias. Sin embargo, algunas operaciones tienen lugar en un pequeño espacio en la base del cerebro a través de una incisión muy pequeña y en estos casos es más importante que

mantenga una visión binocular bien iluminada en los huecos del campo. La perspectiva estereoscópica es entonces la función más útil de los microscopios quirúrgicos para estas situaciones. El microscopio permite la visión estereoscópica en pequeños espacios reduciendo la distancia interpupilar necesaria para la visión binocular. La distancia entre las lentes anteriores del tubo binocular del microscopio es de sólo 16 [mm], mientras que la distancia interpupilar media es de alrededor de 60 [mm] (Yasargil, 1984, p. 208).

### **2.2.2 Instrumentación microquirúrgica**

La aplicación de técnicas microquirúrgicas a la neurocirugía ha requerido el desarrollo de instrumentación que explote adecuadamente las ventajas que ofrece el microscopio de operación. Como la instrumentación es básicamente una extensión de las capacidades físicas humanas, los principios del diseño del instrumento deben tener en cuenta las necesidades físicas del cirujano, así como el trabajo a realizar (Yasargil, 1984, p. 210). El cirujano debe de tener desarrollada mediante entrenamiento cierta destreza motriz en la(s) muñeca(s) para así poder operar de manera óptima las herramientas que serán de ayuda para realizar el procedimiento quirúrgico.

Los instrumentos son requeridos para proporcionar estabilidad al paciente, al cirujano y al campo operativo (Yasargil, 1984, p. 211). Algunos de estos instrumentos son:

- Inmovilizadores de cabeza. Esta herramienta permite que la cabeza se fije en una variedad de posiciones y evita la presión sobre las prominencias óseas y las órbitas oculares.
- Retractores de tejidos blandos. Retractores de resortes, son compuestos principalmente de ganchos afilados y una pinza unidos con un muelle para proporcionar una retracción constante de los tejidos blandos con facilidad para reposicionamiento.

- Retractores de cerebro. Es una herramienta para proporcionar una retracción suave pero constante del cerebro en casi cualquier dirección. Este retractor está sujeto por un poste metálico fijado a la barra lateral de la mesa de operaciones.
- Respaldo de brazo. Es un brazo con resorte montado sobre una articulación esférica con el propósito de estabilizar los antebrazos o manos del cirujano durante la disección microquirúrgica.

El cirujano requiere, además, instrumentación que le maximice la movilidad en la disección precisa. La disección debe realizarse en aperturas profundas estrechas con instrumentos delicados pero fuertes y que no interfieran con la visión (Yasargil, 1984, p. 211). En este campo hay varios instrumentos involucrados:

- Tijeras. Las tijeras de presión y tijeras del tipo cocodrilo junto con sus distintas variedades de tamaños y de formas. Estas tijeras deben tener una acción de cierre suave y cortar de manera limpia.
- Escalpelos. En la sala operatoria se encuentran disponibles diferentes microscalpelos con distintos tipos y formas de hoja para corte.
- Aparato de succión. Las puntas de succión utilizadas en este departamento tienen un borde suavemente redondeado para evitar daños a las estructuras cerebrales durante la disección y tienen un diámetro pequeño. Se prefiere la succión baja, ya que una succión alta puede romper pequeños vasos sanguíneos, así como aneurismas y dañar estructuras neurales. Para ello se han introducido ejes de succión sin agujeros de regulación de presión y una bomba de succión eléctrica cuya presión está estrechamente regulada.
- Taladro de alta velocidad (taladro eléctrico / craneótomo). Un concepto importante en la microcirugía de aneurismas ha sido la craneotomía para crear la brecha necesaria para acceder al cerebro. Para quitar este hueso, se hace uso del taladro eléctrico de alta velocidad. La velocidad se controla

con mayor precisión que con instrumentos accionados por aire y se puede invertir el sentido de giro del taladro.

Además de estos instrumentos se encuentra una variedad de pinzas, bisturís, ciertos aparatos de cauterización, anestésicos, algunos disectores y herramientas de suturación.

### **2.3 Entrenamiento**

El entrenamiento, en general, se trata de un proceso ideado para obtener conocimientos, desarrollar habilidades y reforzar capacidades. En medicina, es un proceso de gran importancia, ya que busca minimizar los riesgos, por ello se han desarrollado tecnologías que auxilien en la adquisición de habilidades y destrezas, requeridas para efectuar un procedimiento quirúrgico. En este sentido, la Neurosimulación es un término nuevo, empleado para describir el estudio y desarrollo de sistemas capaces de recrear condiciones neurológicas, utilizando recursos tecnológicos. Este tipo de sistemas pueden clasificarse en tres tipos:

- HPS (del inglés *Human Patient Simulators*). En este tipo se emplean modelos físicos o maniqués elaborados de materiales especiales que pueden otorgar una mejor sensación realista al tacto y pueden incluir complejos sistemas de control electrónico y/o por computadora (Luengas C., López, & Sánchez, 2017, p. 515). Algunos ejemplos son *OperativeExperience Inc* (OEI, Figura 9. Izquierda), el cual tiene la misión de capacitar al equipo quirúrgico y médico haciendo uso de simuladores médicos con una gran fidelidad anatómica y quirúrgica con el uso de una instrucción experiencial rigurosa. OEI reduce los costos de capacitación al tiempo que aumenta la efectividad y la retención del entrenamiento. OEI hace uso de maniqués sobre los que se pueden realizar importantes operaciones quirúrgicas en el cuerpo utilizando instrumentos quirúrgicos estándar ya que los maniqués están compuestos de materiales que reaccionan quirúrgicamente como el tejido humano y pueden ser cortados, retraídos y suturados. OEI se dedica a aplicar esta tecnología a lo largo de todo el cuidado continuo, desde operaciones tácticas

en el campo hasta procedimientos de emergencia en el quirófano<sup>8</sup>. Otro ejemplo de este tipo es *SimMan 3G* (Figura 9. Derecha) el cual es un simulador de paciente de adultos diseñado para ofrecer entrenamiento médico lo más realista posible, este simulador puede mostrar tanto los síntomas neurológicos como fisiológicos. Es de fácil manejo e incluye tecnología innovadora para aumentar el realismo y crear una simulación más efectiva, algunos ejemplos de esta tecnología lo es el reconocimiento de fármacos, las pupilas sensibles a la luz, la secreción de fluidos corporales y la portabilidad Wi-Fi. Otras características que ofrece el simulador en el maniquí son que puede imitar convulsiones, tiene simulación de sangrado en múltiples zonas, secreciones, sonidos intestinales y sonidos pregrabados que simulan la voz del paciente<sup>9</sup>.



*Figura 9. (Izquierda) OperativeExperience tiene un simulador de craneotomía que incluye el poder realizar cortes de piel, agujeros en el cráneo, limpieza tras un sangrado y suturación en un maniquí haciendo uso de los instrumentos quirúrgicos estándar. Imagen extraída de: <https://operativeexperience.com/craniotomy>. (Derecha) SimMan 3G es un simulador de paciente avanzado que puede mostrar tanto los síntomas neurológicos como fisiológicos. Es de fácil manejo e incluye tecnología innovadora, como reconocimiento de fármacos. Imagen extraída de: <https://www.laerdal.com/la/SimMan3G#/Images>*

- Simuladores Computarizados. Son aquéllos en los cuales se establecen escenarios idealizados en donde el usuario tiene una interacción con el ambiente virtual; el uso de dispositivos automatizados puede incluirse ya que ayuda a crear una sensación de inmersión con el simulador. Un ejemplo es *NeuroTouch*. El simulador neuroquirúrgico en sus componentes incluye un sistema de renderizado gráfico tridimensional (estereoscopio), un sistema de

<sup>8</sup> Fuente: <https://operativeexperience.com/about/>

<sup>9</sup> Fuente: <https://www.laerdal.com/la/SimMan3G#/Info>

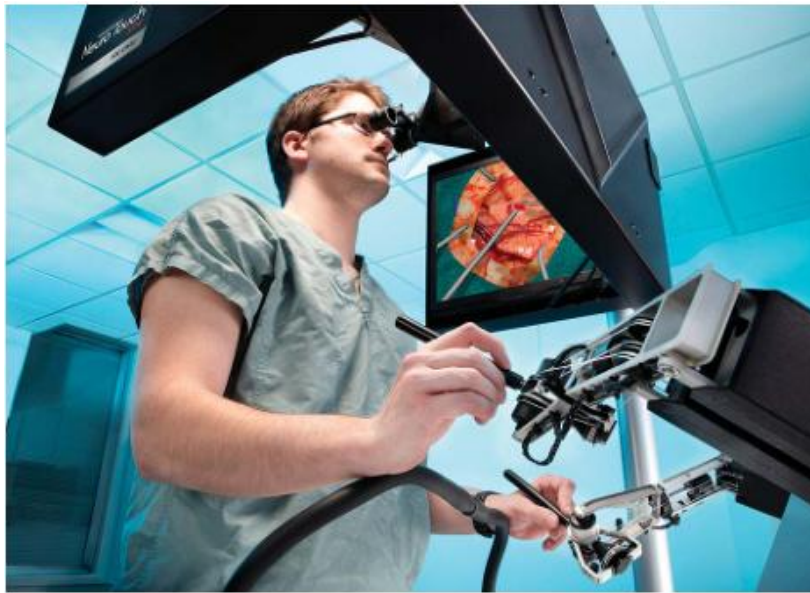


renderizado háptico bimanual, otros controles y una o dos computadoras. Los componentes del simulador se montan en un marco que permite el ajuste de la altura de los dispositivos hápticos, el estereoscopio y permite el ajuste del ángulo de inclinación del estereoscopio. El marco también incluye reposamuñecas ajustables como se ve en la Figura 10 (Delorme, Laroche, DiRaddo, & Del Maestro, 2012, p. 3).



*Figura 10. Componentes de NeuroTouch. Estereoscopio (a), sistemas hápticos (b), fuentes de alimentación y amplificadores para sistemas hápticos (c), computadora (d). Imagen extraída de (Delorme et al., 2012, p. 4)*

*NeuroTouch* (Figura 11) permite la simulación de la extirpación de tumor cerebral mediante un abordaje de craneotomía en 3 tareas de entrenamiento utilizando el aspirador quirúrgico, el aspirador ultrasónico, el electrocauterio bipolar y las microtijeras. El simulador permite ajustar de forma independiente la rigidez tisular cualquier momento durante la simulación usando 2 perillas. Además de los tejidos deformables, el espacio de trabajo quirúrgico incluye representaciones rígidas y fijas de cortinas, piel, cráneo, duramadre y ganchos (Delorme et al., 2012, p. 6).



*Figura 11. Residente de Neurocirugía probando un prototipo de NeuroTouch para proporcionar retroalimentación. Imagen extraída de (Delorme et al., 2012, p. 3).*

- **Simuladores Híbridos**. Este tipo hace uso de técnicas que mezclan la realidad aumentada y la virtualidad aumentada, combinan el uso de maniqués, modelos virtuales y dispositivos electrónicos (Luengas C. et al., 2017, p. 515).

# CAPÍTULO 3

## Reconstrucción de modelos 3D a partir de imágenes de CT

Para dotar al ambiente virtual de elementos de interacción visual, se requirió de la representación de algunas estructuras anatómicas y del modelado de herramientas quirúrgicas necesarias en la simulación. Para ello, se aplicaron algunas técnicas como la generación de mallas usando paquetes de software comercial y libres, que ya cuentan con algoritmos robustos para este tipo de aplicaciones.

### 3.1 Software para la reconstrucción de modelos

Para el proceso de reconstrucción del modelo del cráneo se usaron diversas herramientas, siendo *OsiriX* la principal para obtener un modelo a partir de un conjunto de imágenes extraídas por Tomografía Computarizada; posteriormente el modelo tridimensional fue editado en el software para modelado *Blender*. La edición consistió en la remoción de vértices duplicados e innecesarios en la malla del modelo del cráneo; estos vértices extra son producto de pequeñas imperfecciones generadas por el algoritmo de mallado. Adicionalmente se aplicaron métodos de refinamiento de mallas; este refinamiento es recomendable aplicarlo cuando usamos modelos con una malla poligonal muy compleja como lo fue el modelo del cráneo reconstruido a base de imágenes de CT. El refinamiento del modelo se realizó con ayuda del software MeshLab<sup>10</sup>, ya que ofrece diferentes formas de

---

<sup>10</sup> Enlace al software: <http://www.meshlab.net/>

simplificar superficies triangulares y es capaz de conservar detalles geométricos e información de textura.

### 3.1.1 OsiriX

El software *OsiriX* está diseñado para facilitar la manipulación y permite al usuario navegar a través de series de imágenes multidimensionales mientras se ajusta la combinación de imágenes de diferentes modalidades, contraste e intensidad de la imagen y la velocidad de visualización de imágenes dinámicas. El ajuste simultáneo de los diversos parámetros de visualización es facilitado por herramientas especialmente diseñadas, y las barras de herramientas dentro del programa pueden ser personalizadas por el usuario según sea necesario. *OsiriX* es un proyecto de código abierto, distribuido bajo la licencia LGPL<sup>11</sup> (Rosset, Spadola, Pysher, & Ratib, 2006).

El objetivo de *OsiriX* es ofrecer el mejor visualizador posible para las imágenes producidas por equipos de radiología, tales como Resonancia Magnética (MR<sup>12</sup>), Tomografía Computarizada (CT<sup>13</sup>), Tomografía de Emisión de Positrones (PET<sup>14</sup>), imágenes fusionadas (PET-CT), la unión de una gamma cámara para Tomografía computarizada de emisión de fotones individuales con un CT (SPECT<sup>15</sup>-CT), Ultrasonidos, etc. Sus características clave son: visualización, revisión, interpretación y post-procesamiento de las imágenes. *OsiriX* es un software de procesamiento de imágenes dedicado a imágenes DICOM (extensión *.dcm*) (Osirix, Dicom, Dicom, & Osirix, 2003, p. 1).

DICOM (*Digital Imaging and Communication in Medicine*) es el estándar para el intercambio de imágenes médicas que se utiliza en la mayoría de los hospitales

---

<sup>11</sup> El proyecto GNU tiene dos licencias principales para usarse en bibliotecas. Una es la *Lesser GPL* (o *GPL* reducida) de GNU, la otra es la *GPL* de GNU ordinaria. La elección de una u otra licencia tiene mucha importancia: usar la *LGPL* permite el uso de la biblioteca en programas privativos. El uso de la *GPL* ordinaria para una biblioteca la hace disponible únicamente para programas libres. Fuente: <http://www.gnu.org/licenses/why-not-lgpl.es.html>

<sup>12</sup> Del inglés *Magnetic Resonance*

<sup>13</sup> Del inglés *Computed Tomography*

<sup>14</sup> Del inglés *Positron Emission Tomography*

<sup>15</sup> Del inglés *Single Photon Emission Computed Tomography*

del mundo; su estructura actual está diseñada para garantizar la interoperabilidad de los sistemas utilizados para producir, almacenar, visualizar, enviar, consultar, procesar, recuperar, imprimir imágenes médicas y documentos estructurados derivados, así como para administrar el flujo de trabajo relacionado<sup>16</sup>.

### **3.1.2 Blender**

*Blender*<sup>17</sup> es un programa informático, dedicado especialmente al modelado, iluminación, renderizado, animación y creación de gráficos tridimensionales. También de composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura y pintura digital. En *Blender*, además, se pueden desarrollar videojuegos con simulaciones basadas en física.

El programa fue inicialmente distribuido de forma gratuita, pero sin el código fuente, con un manual disponible para la venta, aunque posteriormente pasó a ser software libre distribuido bajo la licencia GPL. *Blender* es multiplataforma y funciona igualmente bien en ordenadores Linux, Windows y Macintosh. Su interfaz utiliza *OpenGL* para proporcionar una experiencia consistente.

## **3.2 Reconstrucción a partir de imágenes de CT**

### **3.2.1 Escáner CT**

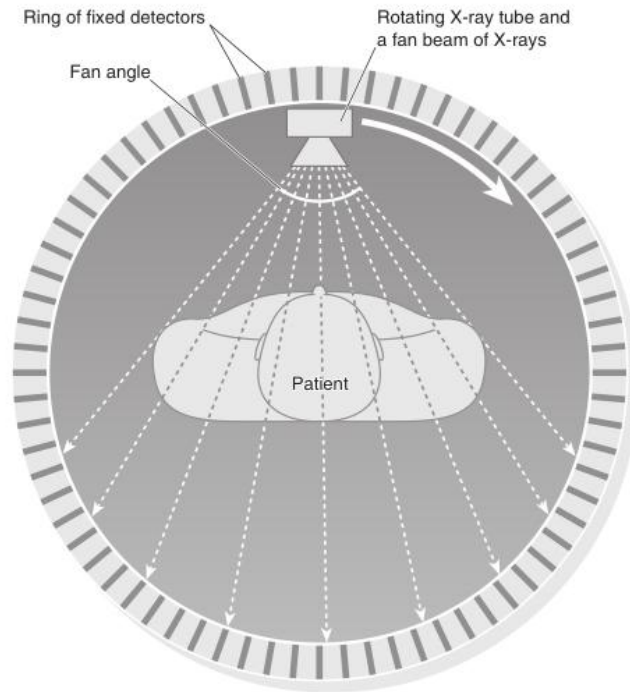
El primer escáner de tomografía computarizada (CT) fue desarrollado en 1972 por Sir Godfrey Hounsfield, desde entonces su uso se ha establecido como una técnica radiológica esencial aplicable en una amplia gama de situaciones clínicas. La CT utiliza rayos X para generar imágenes transversales bidimensionales del cuerpo. Estas imágenes se adquieren mediante una rotación rápida del tubo de rayos X a 360° alrededor del paciente. La radiación transmitida se mide luego por un anillo de varios detectores de radiación sensibles ubicados en el pórtico alrededor del paciente como se ve ilustrado en la Figura 12. La imagen final se

---

<sup>16</sup> Fuente: <http://www.dicomstandard.org/using/>

<sup>17</sup> Fuente: <https://www.blender.org/about/>

genera a partir de esta serie de mediciones, utilizando el principio básico que establece que una estructura interna del cuerpo, se puede reconstruir a partir de múltiples proyecciones de rayos X.



*Figura 12. La generación de imágenes de CT se adquieren mediante una rotación rápida de 360° del tubo de rayos X alrededor del paciente. En la imagen se ven los componentes de un escáner CT, el anillo de dispositivos detectores, el tubo emisor de rayos X y el ángulo del abanico de despliegue y el posicionamiento del paciente dentro del escáner. Imagen extraída de (Payne, 1978, p.4).*

A principios de la década de 1980 se lograron avances significativos en la tecnología del escáner de CT, este progreso ayudó a desarrollar la tecnología de anillo colector, el cual permitía que el tubo de rayos X girara continuamente en una dirección alrededor del paciente; esto evolucionó a lo que hoy se conocen como escáneres CT helicoidales o en espiral, los cuales agregaron el movimiento mecánico axial de la mesa en la que se encuentra el paciente. Con estos tomógrafos, la radiación transmitida toma la forma de una espiral, haciendo que la información se adquiriera como un volumen continuo en lugar de divisiones individuales, esto permite obtener imágenes de regiones anatómicas más grandes sin la necesidad de realizar múltiples procedimientos con el escáner (Payne, 1978, p. 3).

La siguiente generación de tomógrafos fueron los llamados multicorte o multidetectores, esta generación usa los principios del escáner helicoidal, pero incorpora múltiples anillos detectores acomodados en fila con el propósito de aumentar el área de interés en el paciente.

Las imágenes producidas o planos de corte se subdividen individualmente en una matriz con dimensión de hasta 1024x1024 elementos de volumen (conocidos como vóxeles). Cada vóxel es recorrido durante el examen por una gran cantidad de fotones de rayos X, cuya intensidad de radiación transmitida es medida por los detectores. A partir de estas lecturas de intensidad se puede calcular la densidad o el valor de atenuación del tejido en cada punto del corte. Los valores específicos de atenuación se asignan a cada vóxel individual. La imagen visualizada se reconstruye luego como una matriz correspondiente de elementos de imagen (píxeles) (Payne, 1978, p. 7).

### **3.2.2 Unidades y escala de Hounsfield**

A cada píxel se le asigna un valor numérico (número CT), que es el promedio de todos los valores de atenuación contenidos en el vóxel correspondiente. Este número se compara con el valor de atenuación del agua y se muestra en una escala de unidades arbitrarias llamadas unidades Hounsfield (HU) en honor a Sir Godfrey Hounsfield. La escala de Hounsfield asigna al agua como un valor de atenuación (HU) de cero. El rango de los números CT es de 2000 HU, aunque algunos escáneres modernos tienen un rango mayor de hasta 4000 HU. Cada número de la escala representa un tono de gris siendo +1000 el blanco y -1000 el negro como podemos ver en la Figura 13.

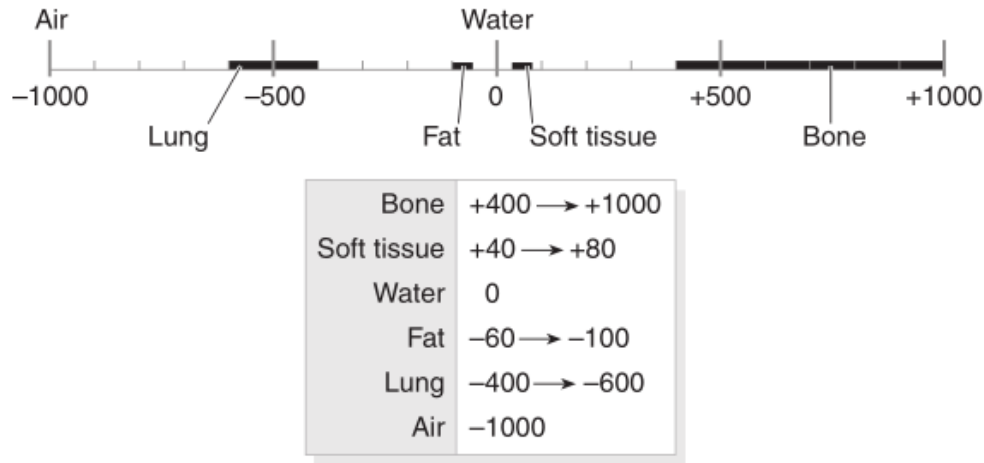


Figura 13. Escala de Hounsfield de números CT, algunos ejemplos son la escala que tiene el Hueso, los tejidos blandos, el agua, la grasa, los pulmones y el aire. Extraído de (Payne, 1978, p.7).

Mientras que el rango de números de CT reconocidos por la computadora es 2000, el ojo humano no puede distinguir con precisión entre 2000 diferentes tonos de gris. Por lo tanto, para permitir que el observador interprete la imagen, sólo se muestra un número limitado de HU. Se logra una escala de grises clínicamente útil al establecer el nivel de ventana (WL<sup>18</sup>) y el ancho de ventana (WW<sup>19</sup>) en la consola de la computadora a un rango adecuado de unidades Hounsfield, dependiendo del tejido que se estudia (Payne, 1978, p. 8).

### 3.2.3 Reconstrucción de imágenes

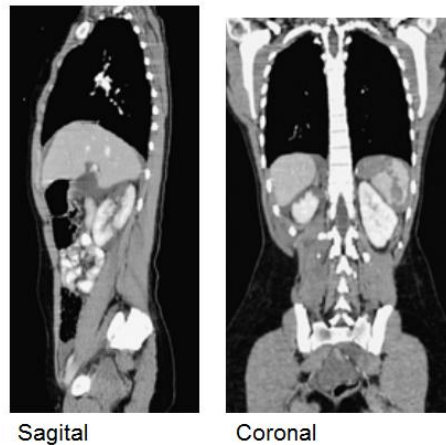
La adquisición de datos volumétricos mediante el uso de escáner CT espiral significa que las imágenes pueden procesarse posteriormente de forma apropiada a la situación clínica. A continuación, se presentan algunos ejemplos de reconstrucción de imágenes mediante el uso de escáneres CT.

<sup>18</sup> Del inglés *Window Level*, representa la HU central de todos los números dentro del ancho de la ventana.

<sup>19</sup> Del inglés *Window Width*, cubre la HU de todos los tejidos de interés y estos se muestran en varios tonos de gris.



- Reformateo multiplanar (MPR<sup>20</sup>). Al tomar una sección a través del conjunto tridimensional de los números de CT adquiridos con una serie de cortes contiguos, se pueden ver los planos sagitales, coronal (Figura 14) y oblicuo junto con el plano transaxial estándar (Figura 15).



*Figura 14. Plano Sagital y plano Coronal obtenidas del MPR. Imagen extraída de (Payne, 1978, p.11).*



*Figura 15. Plano Axial obtenido de MPR. Imagen extraída de (Payne, 1978, p.11).*

- Imágenes tridimensionales. El uso de datos reconstruidos de la computadora permite ver la estructura externa e interna de los órganos. Los datos se pueden proyectar como un modelo tridimensional para mostrar la información espacial o las características de la superficie (como el volumen y la representación de la superficie).

<sup>20</sup> Del inglés *Multiplanar reformatting*.

- Angiografía por tomografía computarizada (CTA<sup>21</sup>). Con la mejora del procedimiento usando contraste intravenoso, las imágenes se adquieren en la fase arterial y luego se reconstruyen y se muestran en formato 2D o 3D. Esta técnica se usa comúnmente para obtener imágenes de la aorta, las arterias renales y cerebrales. Además, existe un interés creciente en el uso de CTA para obtener imágenes de los vasos coronarios y periféricos.

### **3.3 Técnicas de mallado a partir de volúmenes de imágenes**

#### **3.3.1 Volume rendering**

En el área médica, la visualización de conjuntos de datos volumétricos adquiridos mediante CT ayuda a comprender las condiciones patológicas del paciente, mejora la planificación quirúrgica y tiene un papel importante en la educación (Xie, Yang, & Zhu, 2008, p. 117). El renderizado volumétrico (*volume rendering*, en inglés) es un conjunto de métodos empleados para visualizar una proyección en dos dimensiones de un conjunto de datos tridimensionales discretamente muestreados, pertenecientes a un campo escalar en el espacio de tres dimensiones. En aplicaciones médicas, estos datos son factores de opacidad que van de un rango de entre 0 y 1, donde los huesos están representados por el valor más alto por lo que se muestran completamente opacos, mientras que los tejidos se representan con el valor más bajo, por lo que al visualizarlos se muestran transparentes. Técnicamente las representaciones de volumen se convierten en proyecciones cuando se visualizan en una pantalla bidimensional, por ende, la diferenciación entre proyección y *volume render* se dificulta; sin embargo, algunas técnicas facilitan su distinción haciendo uso de mapas de color y sombreado para crear representaciones más realistas (Figura 16).

---

<sup>21</sup> Del inglés *CT angiography*.



Figura 16. Reconstrucción tridimensional de arterias cerebrales empleando la técnica de Volume render a partir de imágenes de angiotomografía. Cortesía del Hospital General de México.

### 3.3.2 Datos volumétricos

En comparación con los datos de superficie que sólo determina la capa exterior de un objeto, los datos de volumen se utilizan para describir las estructuras internas de un objeto sólido. Anteriormente se mencionó, que un volumen es una matriz tridimensional de valores de datos (vóxeles), la matriz tridimensional también se puede ver como una pila de matrices bidimensionales de vóxeles y cada una de estas matrices bidimensionales como una imagen (o corte), donde cada uno de los vóxeles representa un píxel.

Los datos volumétricos generalmente son un conjunto de muestras  $S(x, y, z, v)$ , en una ubicación tridimensional donde el valor  $v$  (vóxel) incluso puede ser un vector, que representa, por ejemplo, alguna propiedad mensurable de los datos, como pueden ser una estructura anatómica, su densidad, temperatura o presión.

Los datos de volumen médico obtenidos de MRI y escáneres CT típicamente son anisotrópicos<sup>22</sup> con una densidad de muestreo igual en la dirección  $x$  e  $y$ , pero con una densidad más gruesa a lo largo de la dirección  $z$ . El tamaño es típicamente de más de 100 cortes o más con  $512 \times 512$  vóxeles cada uno, lo que requiere un

<sup>22</sup> Cuando el espaciado entre muestras a lo largo de cada eje es una constante, entonces se llama isotrópico, pero puede haber tres constantes de espaciado diferentes para los tres ejes. En ese caso, el conjunto es anisotrópico

cuidado especial con respecto a la eficiencia de los algoritmos desarrollados. Estos conjuntos de datos  $v$  son la base para la evaluación de algoritmos de representación de volúmenes (Zhou & Tonnies, 2003, p. 2).

### 3.3.3 Clasificación de algoritmos de *Volume Render*

En general, las técnicas de *volume render* se dividen en dos categorías: técnicas indirectas (IVR<sup>23</sup>), empleadas para representar superficies; y las directas (DVR<sup>24</sup>), que no extraen explícitamente estructuras geométricas de los datos de volumen, sino a través de segmentación difusa usando una función de transferencia. El enfoque de *ray-casting* es el típico método de renderizado de volumen directo (Zhou & Tonnies, 2003, p. 3).

En la técnica de *ray-casting*, los rayos se convierten en un volumen de datos y se toman muestras a lo largo de cada rayo mediante la interpolación de los vóxeles circundantes (Li & Zhang, 2010, p. 157). El objetivo básico de la proyección de rayos es permitir el mejor uso de los datos tridimensionales y no intentar imponer ninguna estructura geométrica en éste. Resuelve una de las limitaciones más importantes de las técnicas de extracción de superficie, la cual es la forma en que muestran una proyección de una capa delgada en el espacio. Las técnicas de extracción de superficie no toman en cuenta que, especialmente en imágenes médicas, los datos pueden originarse a partir de fluidos y otros materiales que pueden ser parcialmente transparentes y deben modelarse como tales. La técnica de *ray-casting* no sufre de esta limitación (Zhou & Tonnies, 2003, p. 4).

El método clásico de representación volumétrica de *ray casting* es el algoritmo de Levoy, el cual se compone de cuatro pasos. El primer paso se preparan los datos para que los vóxeles estén alineados correctamente con la interpolación de cuadrículas y aumentar el número de vóxeles en un eje. El segundo paso incluye sombreado y clasificación. El sombreado se realiza para producir colores por cada vóxel, usando el modelo básico de sombreado de Phong (interpola las normales de

---

<sup>23</sup> Del inglés *Indirect Volume Rendering*.

<sup>24</sup> Del inglés *Direct Volume Rendering*.

superficie entre polígonos rasterizados y calcula los colores de los píxeles en base a las normales interpoladas y un modelo de reflexión), mientras que la clasificación es para determinar las opacidades de vóxel con dos métodos diferentes: iso-valor de superficies de contorno, que es para representación de volumen puro, con la posibilidad de mostrar varias superficies. En el tercer paso, el volumen está configurado para la etapa de *ray casting*. Los rayos se proyectan en el volumen de cada píxel en la pantalla, a intervalos equidistantes usando interpolación tri-lineal de los ocho véxeles vecinos. El último paso es la composición, donde se calcula el color final del píxel en coordenadas de pantalla (Li & Zhang, 2010, p. 157, 158).

Los algoritmos de IVR (o algoritmos orientados a la superficie) son un estándar para casi todos los problemas de visualización en 3D. Se definen como la visualización de un conjunto de datos volumétricos transfiriendo primero los datos a un dominio diferente, para eso, estos algoritmos primero ajustan las primitivas geométricas a valores en los datos y posteriormente es representando directamente desde el nuevo dominio. Los valores de los datos generalmente se eligen de una iso-superficie, que es el conjunto de ubicaciones en los datos donde el campo escalar es igual a algún valor. Los problemas de visualización típicos a resolver en el contexto médico son, por ejemplo, tejidos que no tienen superficie definida o vasos sanguíneos cuyo tamaño está cerca del límite de la resolución de la imagen. Sin embargo, los métodos orientados a la superficie dejan estos problemas sin resolver.

Los algoritmos comunes de iso-superficie son cubos opacos (*Opaque Cubes*, *Cuberille*), cubos marchantes (*Marching Cubes*), tetraedros marchantes (*Marching Tetrahedra*) y cubos divisibles (*Dividing Cubes*) (“Volume Rendering,” 2007). A continuación, se describen los primeros dos, los cuales fueron de importancia para este trabajo:

- *Opaque Cubes* o *Cuberille* (Elvins, 1992, p. 7): Este algoritmo fue uno de los primeros métodos ampliamente utilizados para visualizar datos de volumen en forma de contornos. Implica decidir si un elemento de volumen particular es una parte de la superficie o no. Simplemente visualiza las celdas que son parte de la

iso-superficie. El algoritmo consiste en dos etapas: en la primera, el usuario selecciona un valor de umbral y se realiza un recorrido del volumen, buscando celdas con valores contenidos a lo largo del valor de umbral. Se genera un hexaedro por cada celda (Elvins, 1992, p. 7) (Figura 17 izquierda). La segunda etapa del algoritmo implica clasificar qué vóxeles pertenecen al objeto y cuáles no. Si se eligen varios umbrales, cada conjunto correspondiente de polígonos de celda se puede diferenciar por su color y opacidad. El algoritmo es fundamentalmente deficiente al mostrar pequeños detalles en imágenes de baja resolución, creando un efecto de “voxeleado”. La apariencia de bloque de la iso-superficie aproximada se puede reducir mediante el uso de alguna técnica de suavizado en el paso de sombreado (Figura 17 derecha).

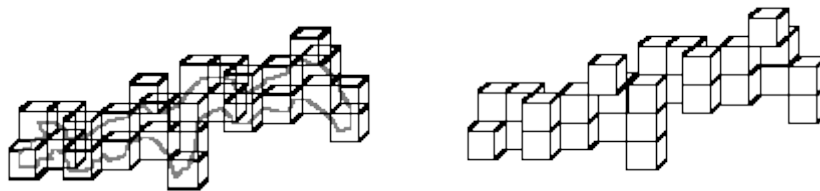


Figura 17. Representación del funcionamiento del algoritmo de Opaque Cubes. Extraída de Siggraph: <https://www.siggraph.org/education/materials/HyperVis/vistech/volume/surface3.htm>

- **Marching Cubes:** Este algoritmo es una variación del algoritmo *Cuberille*. La noción básica consiste en definir un vóxel por los valores de píxel en las ocho esquinas de un cubo. Si uno o más píxeles de un cubo tienen valores inferiores a un umbral especificado por el usuario, y uno o más tienen valores superiores a este valor, concluimos que el vóxel debe contribuir con algún componente de la iso-superficie. Al determinar qué bordes del cubo se cruzan con la iso superficie, podemos crear parches triangulares que dividen el cubo entre las regiones dentro de la iso superficie y las regiones exteriores. Cada uno de los 8 vértices de un vóxel puede estar dentro o fuera del valor iso-superficie. Los puntos exactos de intersección de los bordes se determinan utilizando 15 conjuntos de polígonos predefinidos, cómo se ven en la Figura 18. Al conectar los parches de todos los cubos en el límite de la iso-superficie, obtenemos una representación de la malla objetivo.

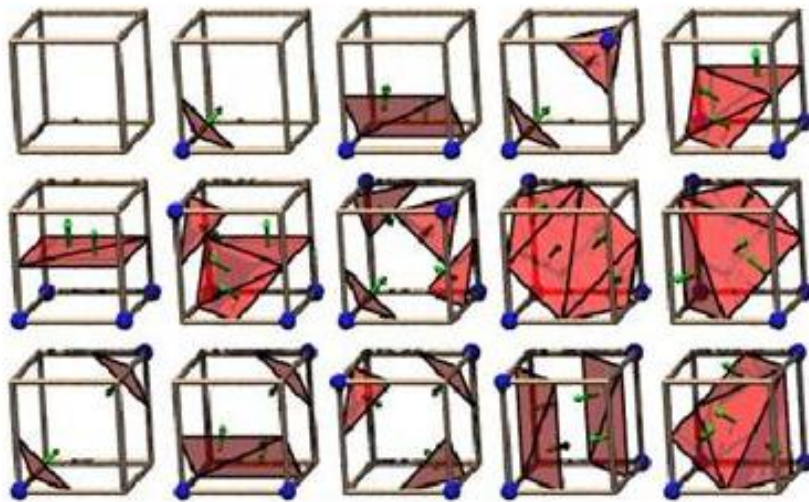


Figura 18. Hay exactamente 256 formas en que cuatro o menos triángulos pueden ajustarse a una celda, y el número de casos puede reducirse a 15 por reflexión y rotación (Elvins, 1992, p. 7).  
 Imagen extraída de ("Volume Rendering," 2007)

Tanto los algoritmos de *Marching Cubes* como el de *Cuberille* implican una aproximación intermedia de las superficies en el conjunto de datos utilizando primitivas geométricas. Esta técnica tiene una serie de desventajas:

1. Se pierde precisión cuando se visualizan detalles pequeños o borrosos.
2. Las suposiciones que se hacen sobre los datos pueden no ser necesariamente válidas. Esto se aplica particularmente a la suposición de que las superficies existen dentro de los datos para mapear las primitivas geométricas.
3. A menos que la información original se almacene junto con la representación geométrica, la información en el interior de las superficies se pierde.

Las dos primeras desventajas enumeradas aquí están directamente relacionadas con el requisito de mapear primitivas geométricas en los datos. Si bien este método funciona bien para algunos conjuntos de datos, se descompone cuando hay pequeños detalles en una escala similar al tamaño de la cuadrícula en los datos, y cuando no existen superficies bien definidas.

Debido a esto se han propuesto varias extensiones al algoritmo de *Marching Cubes* para evitar los casos ambiguos. Estos incluyen *Marching Tetrahedra* y *Dividing Cubes*. Adicionalmente, el algoritmo de cubos marchantes tiene la desventaja de generar un gran número de triángulos, lo que complica el procesamiento de polígonos a visualizar.

### **3.4 Visualización y refinamiento**

Debido al amplio avance tecnológico en el campo de los gráficos por computadora durante los últimos años, ha habido una expansión de las aplicaciones relacionadas con el modelado de objetos. Para la representación de tales modelos se usan comúnmente mallas poligonales. Con las crecientes demandas de calidad, la complejidad de los cálculos que tenemos que manejar se representan en modelos que tienen cientos de miles o incluso millones de triángulos que lo conforman.

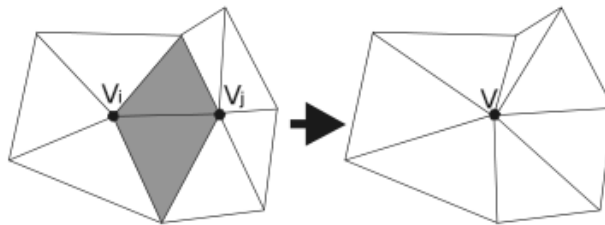
En todas las áreas que emplean modelos complejos, existe una relación entre la precisión con la que se modela la superficie y el tiempo necesario para procesarla. En un intento por reducir el tiempo, a menudo sustituimos el modelo original con una aproximación. Por lo tanto, se han desarrollado técnicas para la simplificación de mallas poligonales grandes y altamente detalladas. El objetivo de tales técnicas es reducir la complejidad del modelo conservando sus detalles importantes (Franc & Skala, 2002, p. 42).

Una de las técnicas de simplificación de mallas es conocida como decimación; en general, estas técnicas se pueden dividir de acuerdo con las primitivas que componen a la malla. Algunos de los casos son la decimación por vértices y la decimación por borde (contracción o colapso) ilustrada en la Figura 19.

El principio de estos métodos es similar, parten de una evaluación de la importancia de la primitiva (vértices, bordes, parches) en la malla y las de menor importancia se eliminan, rellenando el agujero resultante con algún método de triangulación. La malla resultante contiene una menor cantidad de triángulos.



- El método de decimación de vértices consiste en que cada vértice es evaluado según su distancia desde un plano (o línea) promedio dado por sus vértices vecinos. El segundo paso es eliminar el vértice de menor importancia y hacer una nueva triangulación. Como estamos en tres dimensiones, la triangulación es bastante compleja y lleva mucho tiempo.
- El método de la decimación de borde consiste en evaluar todos los bordes que salen del vértice y el borde menos importante se contrae a longitud cero. Esto significa que el vértice se mueve al punto final del borde elegido, que debe eliminarse de la malla junto con dos triángulos adyacentes.



*Figura 19. Decimación por borde: el borde menos importante se reduce a cero, esto significa que el vértice se mueve al punto final del borde elegido, que debe eliminarse de la malla junto con dos triángulos adyacentes. Imagen extraída de (Franc & Skala, 2002, p. 43).*

En la Figura 20 se muestra un ejemplo del proceso de reconstrucción de una superficie usando las técnicas descritas en este capítulo, el cual se empleó para representar los modelos geométricos del cráneo para los fines de esta tesis (Figura 21). Los modelos finales, incluidos en el simulador de craneotomía se describen en los siguientes capítulos.

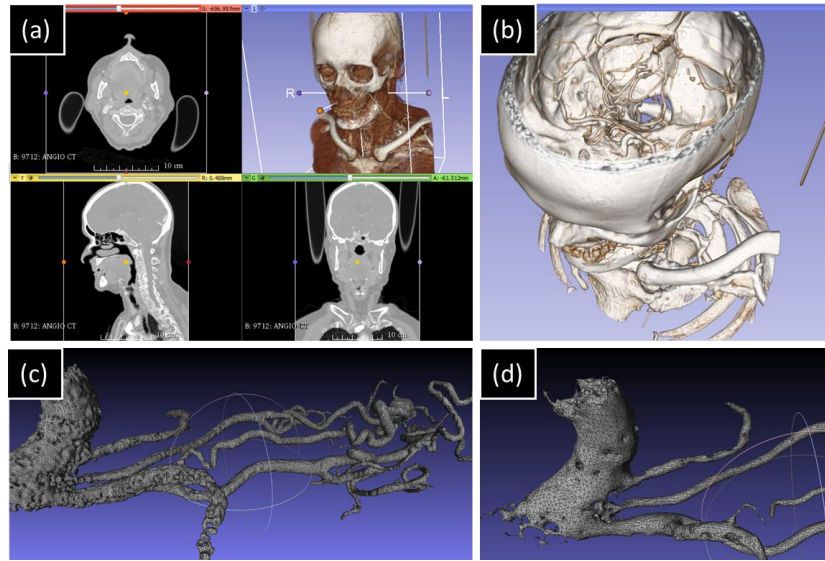


Figura 20. Proceso de reconstrucción de vascularidades a partir de imágenes de CTA. (a) visualización con 3DSlicer de cortes axiales, sagitales y coronales de un estudio de CTA, y su correspondiente representación (b) usando un método de volumen render basado en ray casting. (c) Visualización de la malla resultante en MeshLab, tras aplicar un método basado en marching cubes con 3DSlicer. (d) Aplicación del algoritmo de decimación por borde en MeshLab.

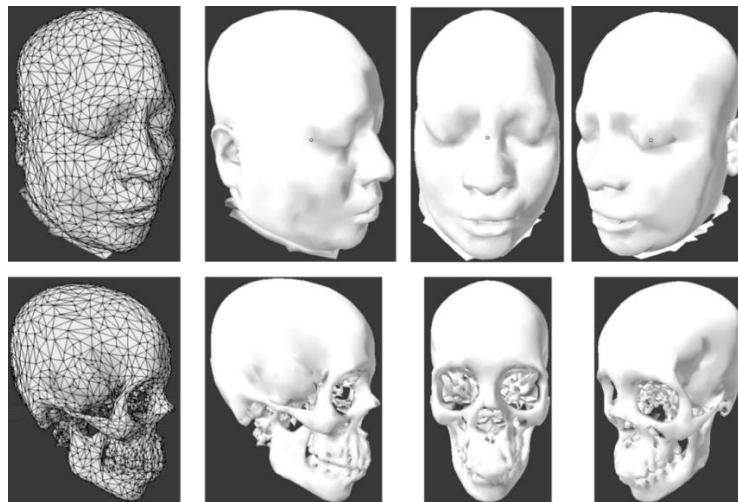


Figura 21. Los modelos de piel y de cráneo se someten a una reducción manual de triángulos mediante el uso de software para modelado como Blender y un refinamiento de la malla con el uso de MeshLab.

# CAPÍTULO 4

## Simulador de entrenamiento para craneotomía

El simulador para entrenamiento en el procedimiento de craneotomía está compuesto por un conjunto de bloques de software cuyas funcionalidades están dirigidas a retroalimentar al usuario con estímulos aurales, táctiles y visuales. Se trata de una plataforma de software que presenta un escenario idealizado, en el que un aprendiz del procedimiento se ve envuelto en un conjunto de acciones controladas por el flujo del programa. Los dispositivos de interacción, que censan los movimientos del usuario, proveen información útil al sistema para realizar cálculos de fuerza al contacto con un modelo geométrico. En este capítulo, se presenta el diseño e implementación del método de cortes en objetos rígidos para la simulación de corte de hueso en un modelo virtual.

### 4.1 Metodología

El sistema se desarrolló en lenguaje C++ con *Visual Studio 2015* para Windows 10 de 64 bits, usando el SDK (Software Development Kit) *OpenHaptics* junto con el dispositivo háptico (*Haptics* en inglés) *Geomagic Phantom Omni* de la compañía 3DSYSTEMS<sup>25</sup>.

El kit de desarrollo *OpenHaptics* proporciona la capacidad de integrar un dispositivo háptico en una aplicación 3D para configurar propiedades físicas y simular superficies de algún material, así como aplicar algoritmos de interacción con

---

<sup>25</sup> Página oficial: <https://es.3dsystems.com/>

mallas poligonales. El kit está disponible de forma gratuita para desarrollo y su uso no comercial. *OpenHaptics* intenta simplificar la programación al encapsular las características comunes presentes en la mayoría de las aplicaciones hápticas. Estas características incluyen:

- Carga y almacenamiento de archivos en formatos estándar (\*.obj)
- Configuración de ventanas gráficas e inicialización del entorno con *OpenGL*
- Inicialización de dispositivos hápticos
- Configuración de escena y cámara virtual
- Asignación de parámetros de fuerzas (fricción estática y dinámica) y rigidez a objetos en la escena
- Configuración de *callback's* para manejo de eventos

El diagrama de la Figura 22, muestra el diagrama de bloques del sistema de simulación, en el cual se identifican tres módulos principales:

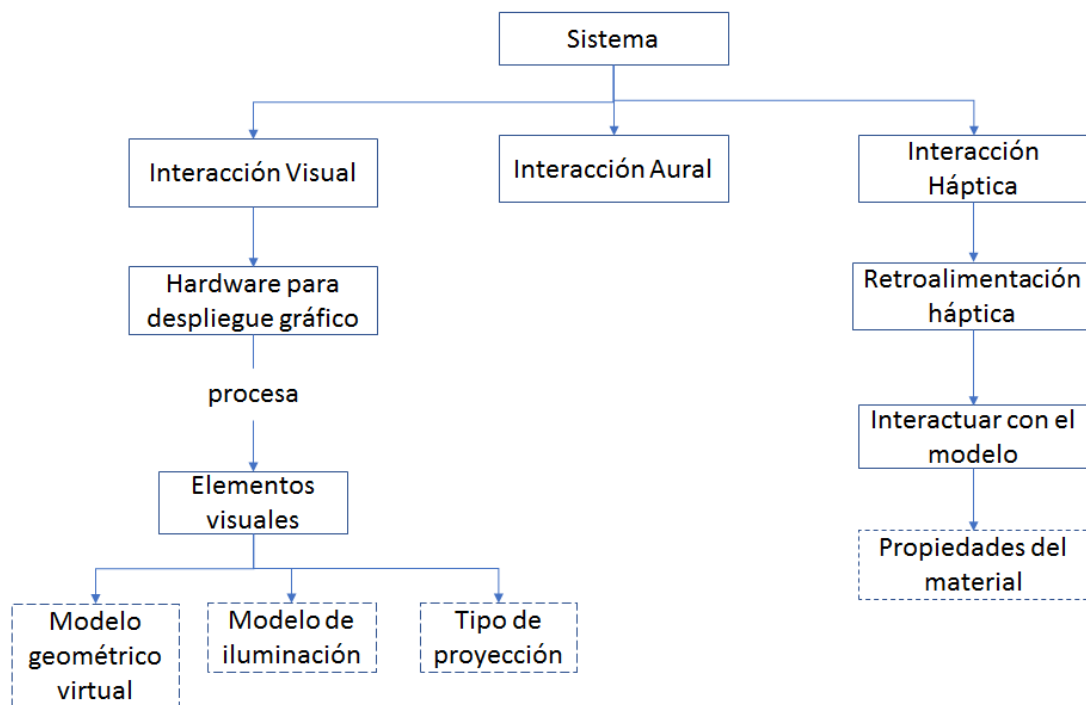


Figura 22. Arquitectura del sistema del simulador.

- La interacción visual: Consiste en la presentación de contenido gráfico en un monitor; en esta sección se procesan los elementos visuales como lo son los modelos geométricos virtuales, el modelo de iluminación y el tipo de proyección.
- La interacción aural: Consiste en el estímulo auditivo que proporciona el programa al usuario.
- La interacción háptica: Consiste en el estímulo táctil que el sistema provee al usuario; aquí ocurre la interacción con el modelo virtual a partir del intercambio de información de la posición del dispositivo, se definen las propiedades de los materiales que se reflejan como una fuerza.

Se emplea un modelo de iluminación simple, el cual define la naturaleza de la luz que emana una fuente luminosa, así como su distribución de intensidad sobre una superficie. En general, para una escena gráfica, se pueden definir cinco tipos de luces:

- Luz omnidireccional. Es una fuente de luz donde los objetos son afectados por igual en la escena produciendo un efecto de iluminado uniforme con intensidad y color fijos.
- Luz direccional o de foco. Este tipo de fuente ilumina a todos los objetos por igual desde una dirección dada, su posición dentro de la escena no importa ya que tamaño y distancia que abarca es infinita.
- Luz puntual. En este tipo luz, los rayos provenientes de la fuente no se dispersan, por lo que se percibe como un punto iluminado sobre la superficie de un objeto.
- Luz de área. En este tipo de fuente, la luz se origina de un solo punto o de un plano.
- Luz volumétrica. En este tipo de fuente, la luz se origina en un pequeño volumen, un espacio cerrado que ilumina a los objetos dentro de éste.

Además de la iluminación, en el programa se hace uso de una proyección en perspectiva, la cual proyecta una imagen de un objeto en tres dimensiones sobre una superficie plana (en dos dimensiones). La Figura 23 ilustra la forma de un

volumen finito de vista en proyección de perspectiva con un plano de vista que se coloca entre el plano de corte cercano y el punto de referencia de proyección.

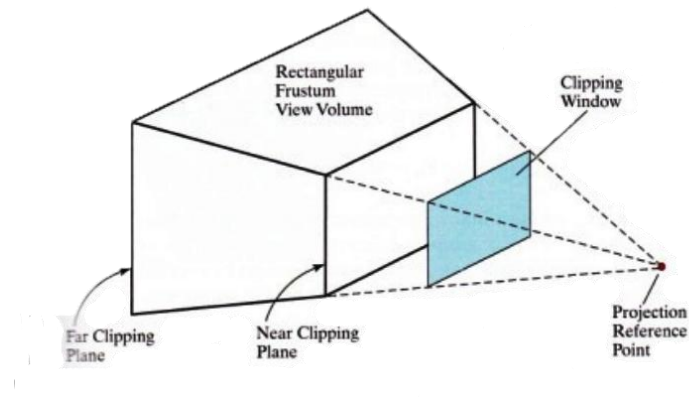


Figura 23. Un volumen de vista en proyección en perspectiva (frustum) con el plano de vista "enfrente" del plano de corte cercano. Imagen extraída de (Hearn et al., 2004, p. 373).

El flujo del programa, representado en la Figura 24, muestra las acciones realizadas por un usuario desde la inicialización del programa hasta la obtención de resultados. En las secciones siguientes, analizaremos el funcionamiento de cada uno de estos bloques.

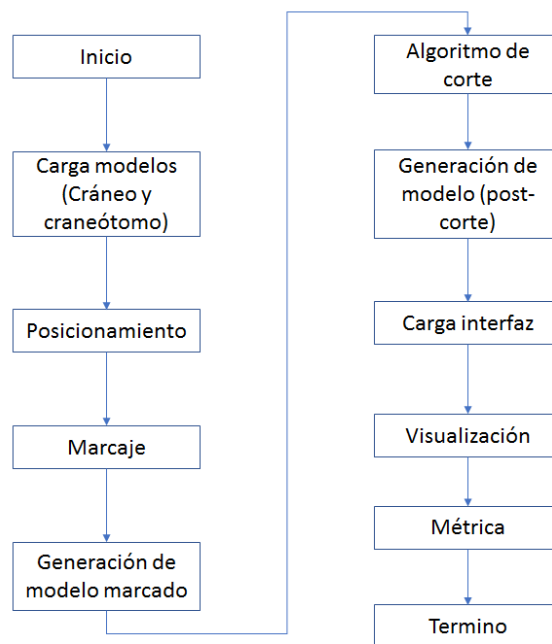
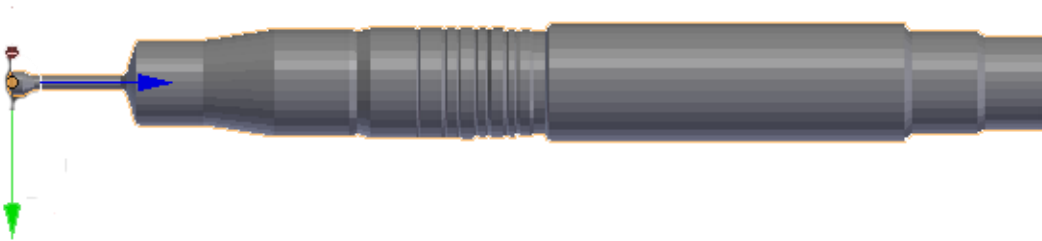


Figura 24. Diagrama de flujo del programa.

## 4.2 Carga de modelos

Una vez que el programa es ejecutado se despliega una ventana con un menú principal cuya función principal es la carga del modelo tridimensional en formato OBJ<sup>26</sup>. Para los fines de este trabajo, se emplearon modelos de cráneo que fueron reconstruidos usando la metodología descrita en el capítulo 3.

A la par que el modelo del cráneo, un modelo del craneótomo virtual es cargado en la interfaz por medio de una función de inicialización de geometrías. Este modelo se generó a partir de una técnica de modelado por curvas de revolución y *blueprints*<sup>27</sup>. La Figura 25 muestra el resultado del proceso de modelado del instrumento virtual.



*Figura 25. Modelo en 3D del craneótomo. Modelado en Maya y limpiado en Blender, el craneótomo fue modelado con la ayuda de imágenes reales de la herramienta.*

El modelo del craneótomo virtual es dibujado dentro de la función de renderizado, siempre y cuando cumpla que el cursor (definido por la posición de la pluma del dispositivo háptico) este activo. Las transformaciones del cursor háptico (proxy) se almacenan en un arreglo de 16 valores de tipo *double*, que representa la matriz homogénea de posición, rotación y escalamiento del dispositivo. Esta matriz se emplea unir el espacio visual con el espacio háptico. Multiplicando la matriz del

---

<sup>26</sup> El formato de archivo OBJ es un formato de datos simple que representa solamente la geometría 3D e incluye solo la posición de cada vértice, la posición UV de cada vértice de coordenadas de texturas, las normales y las caras que componen cada polígono definido como lista de vértices, y los vértices de texturas. Fuente: [http://docs.mcneel.com/rhino/5/help/es-es/fileio/wavefront\\_obj\\_import\\_export.htm](http://docs.mcneel.com/rhino/5/help/es-es/fileio/wavefront_obj_import_export.htm)

<sup>27</sup> Los *blueprints* son guías que sirven para el proceso de modelado de un objeto 3D; consisten en imágenes que se colocan en el fondo de una proyección ortogonal para posteriormente marcar el contorno de un objeto.

modelo virtual con la transformación del dispositivo, se puede lograr el desplazamiento del craneótomo en la escena e interactuar con la geometría del cráneo, como se muestra en la Figura 26.



*Figura 26. Modelos del cráneo y craneótomo virtuales visualizados. El craneótomo sigue los movimientos de la pluma del dispositivo háptico.*

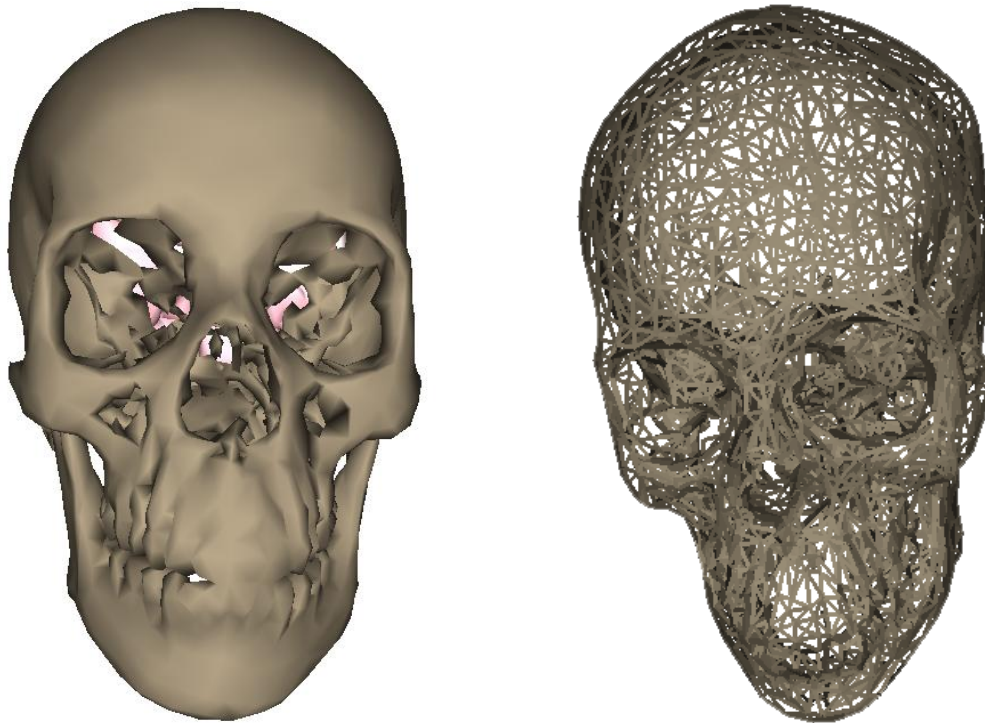
Algunas opciones de visualización del programa incluyen: representación sólida (modelo sólido) y la representación alámbrica (conocida como *wireframe* en inglés). El modelo sólido es un volumen que representa un objeto 3D, y que tiene además de propiedades visuales, propiedades como masa, centro de gravedad y momentos de inercia. Se pueden configurar sólidos a partir de primitivas de sólido como conos, prismas rectangulares, cilindros y pirámides, o mediante la extrusión, el barrido o por la revolución sobre un eje. Los modelos sólidos se pueden combinar usando operaciones booleanas como la unión, la diferencia y la intersección<sup>28</sup>. Un modelo de estructura alámbrica (*wireframe*) en 3D es una representación del esqueleto o las aristas de un objeto modelado mediante el uso de líneas y curvas. Estos modelos constan de un conjunto de puntos, líneas y otras curvas que definen

---

<sup>28</sup> Referencia: <https://knowledge.autodesk.com/es/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2015/ESP/AutoCAD-Core/files/GUID-9DACE807-BC9D-4357-B47E-C6199F6AF1A2-htm.html>



las aristas del objeto. En la Figura 27 podemos ver el modelo 3D del cráneo en su representación como modelo sólido y malla de alambre.



*Figura 27. Modelo en 3D del cráneo. A la izquierda, representación en modelo sólido. A la derecha, representación de un modelo en malla de alambre.*

### **4.3 Posicionamiento**

Una vez cargado los modelos tridimensionales en la escena son colocados en una posición por defecto y para poder comenzar a interactuar con ellos. Una opción que tenemos es el reposicionamiento de la cámara que visualiza el modelo; esta acción se colocó debido a que en el proceso quirúrgico de la craneotomía la cabeza del paciente necesita estar con una posición en específico y esta posición depende de la aproximación que usarán para llegar al área de interés. Este reposicionamiento en el programa se hace mediante el uso del mouse y nos permite rotar, escalar y trasladar el modelo del cráneo. Internamente, se define una función de evento `OnMouseMove()`, que implementa banderas de estado. Estas banderas están declaradas en pequeñas funciones y cada función está relacionada al estado de los botones del mouse:

- `OnLeftButtonDown`: pone en verdadero una bandera cuando detecta que el botón izquierdo del mouse está presionado; esta bandera permite la rotación de la cámara, mientras que una segunda bandera la desactiva al detectar que el botón es dejado de presionar. Dentro de la función `OnMouseMove()` existe una condicional que se activa cuando se detecta que la bandera está en verdadero, en este condicional se realiza la rotación de la cámara. Se hace uso de vectores, uno de ellos almacena la posición actual del puntero y otro la posición anterior, ambos vectores son normalizados y después se realiza un producto cruz entre ellos, el resultado es almacenado en un nuevo vector que contiene los valores de la rotación del modelo. El ángulo de rotación es calculado usando el vector obtenido previamente, se saca su magnitud y posteriormente el arco seno para tenerlo en radianes. Finalmente se construye una matriz de rotación que se aplica a toda la escena virtual.
- `OnRightButtonDown`: pone en verdadero una bandera cuando se detecta que el botón derecho del mouse se está presionando; esta bandera junto con el movimiento del mouse hacia adelante y atrás permite la escala (zoom) de la cámara que visualiza al modelo del cráneo, la bandera se pone en falso cuando detecta que el botón derecho es dejado de presionar.
- `OnMiddleButtonDown`: pone en verdadero una bandera cuando detecta que el botón medio del mouse está siendo presionado; esta bandera permite que se traslade la cámara (paneo). La bandera se pone en falso cuando se deja de presionar el botón.

#### **4.4 Marcaje (Interacción háptica)**

El marcaje consiste en describir la interacción háptica con el modelo geométrico virtual. En la literatura, la interacción háptica es la ciencia de incorporar el sentido del tacto y el control a las aplicaciones informáticas a través de la fuerza o la respuesta kinestésica (3DSYSTEMS, 2015, p. 5). Esta interacción es una parte fundamental para lograr un mejor sentido de inmersión en el programa, ya que agrega el sentido del tacto en la simulación. En el caso del corte de hueso, el

especialista requiere tener retroalimentación de este sentido, dado que es una guía que le sirve como referencia sobre la presión adecuada en la superficie del cráneo.

La interacción háptica se apoya con dispositivos especiales que envían señales de entrada y salida a la aplicación receptora, para que de esta forma el usuario pueda interactuar y manipular los modelos virtuales tridimensionales. El tipo de respuesta utilizada para transmitir el sentido del tacto está determinado por el tipo de sistema que se utiliza (3DSYSTEMS, 2015, p. 5). Los dispositivos hápticos utilizan motores para crear las fuerzas de retorno al usuario para simular el tacto y la interacción con objetos virtuales. En función del modelo, los dispositivos proporcionan una retroalimentación de fuerza de 3 ó 6 grados de libertad (DOF<sup>29</sup>); los 6 grados de libertad están directamente relacionados con la posición y la rotación de los manipuladores del dispositivo. Adicionalmente el dispositivo presenta una fricción interna residual que se puede traducir como un ruido e inercia inherente al movimiento.

Los dispositivos Geomagic Phantom Omni (Figura 28 y Figura 29), empleados en el programa de simulación, miden la posición espacial 3D (a lo largo de los ejes X, Y y Z) y la orientación (giro, inclinación y dirección) del lápiz de mano, éste último es un elemento muy importante en estos dispositivos ya que para este programa es necesario brindar la sensación de inmersión mediante el sentido del tacto.

---

<sup>29</sup> Los grados de libertad también son conocidos como DOF que son las siglas de su traducción en inglés, *Degrees Of Freedom*.



Figura 28. Movimientos del Geomagic Phantom Omni®. La cabeza principal, de izquierda a derecha, la pluma, movimiento hacia adentro y afuera, el brazo conectado a la pluma, movimiento hacia arriba y abajo. Imagen extraída de (Touch, X, Openhaptics, & Desktop, 2013, p. 22).

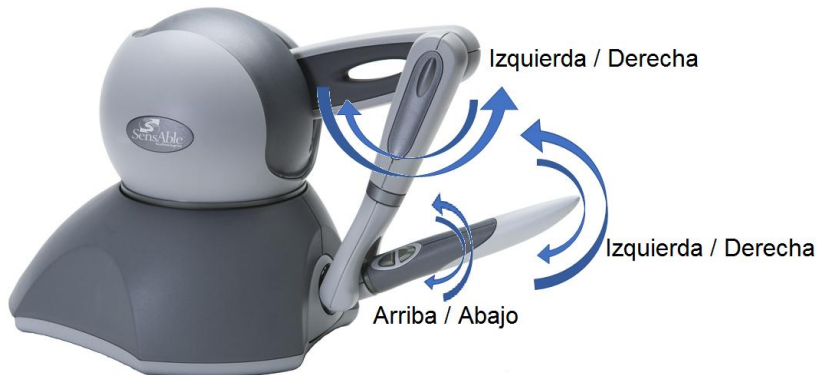


Figura 29. Movimientos del Geomagic Phantom Omni®. Del brazo conectado a la pluma, movimiento de izquierda a derecha, de la pluma, movimiento de arriba hacia abajo y de izquierda a derecha. Imagen extraída de (Touch et al., 2013, p. 23).

En el SDK de *OpenHaptics*, las propiedades los materiales y superficies, se definen por medio de la función *hMaterialfv*, los cuales se pueden configurar de forma independiente para los triángulos con los que se tiene contacto (frontal y/o posterior) perteneciente a los modelos. Esta función se usa antes de definir las formas o modelos con los que se trabaja.

La definición de materiales se realiza durante la inicialización del programa, se configura la cara que va a ser la que reciba las propiedades del material (en

nuestro caso será la cara frontal, *HL\_FRONT*), y esta cara se pasa como primer parámetro para la configuración del material. Un segundo parámetro que recibe la función *hlMaterialfv* es la propiedad que modificamos, en este caso fueron 4:

- *HL\_STIFFNESS*

Controla qué tan dura es la superficie de contacto.

- *HL\_DAMPING*

Reduce la propiedad de amortiguamiento de la superficie.

- *HL\_STATIC\_FRICTION*

Controla la resistencia de la superficie, conocida como fricción estática.

- *HL\_DYNAMIC\_FRICTION*

Controla la resistencia de la superficie respecto a un movimiento tangencial con respecto al cambio de la posición del dispositivo háptico.

Los valores empleados en esta etapa fueron 1.0, 0.2, 0.5 y 0.5 respectivamente con el propósito de intentar replicar la sensación de rigidez y fricción que tiene el cráneo. El poder sentir la rigidez simulada en el modelo tridimensional nos ayuda para dos cosas, la primera es aumentar el nivel de realismo a la simulación y la segunda es ayudar a ubicar el modelo tridimensional intangible en nuestra realidad.

En el proceso quirúrgico de la craneotomía es necesario marcar la zona en la cabeza donde se tiene planeado hacer el corte para acceder al cerebro. Esta acción se realiza con el uso del dispositivo háptico, ubicando el puntero y tocando el modelo tridimensional del cráneo, para definir una trayectoria sobre la superficie.

OpenHaptics maneja funciones denominadas *callback* las cuales son llamadas de función que ocurren como resultado de un evento. El movimiento, un "toque" háptico o una pulsación de botón son ejemplos de eventos que pueden desencadenar una devolución de llamada (3DSYSTEMS, 2015, p. 9). En el programa se hacen uso de tres funciones callback, las cuales se agregan mediante

el uso de la función *hlAddEventCallback* y posteriormente se remueven con el uso de *hlRemoveEventCallback*. Para agregar un evento requiere de:

- El evento que lo activa. En nuestro caso es el botón que se presiona, que el deje de ser presionado o que haya movimiento.
- El identificado de la forma con la que se activa. Para el evento de presionar y de movimiento es necesario pasar el identificador del modelo tridimensional mientras que al evento para soltar la presión del botón la devolución de llamada se llamará independiente de cualquier objeto
- El subproceso para llamar a la función de devolución de llamada. La función de devolución de llamada se invocará desde el hilo del cliente, cuando el programa cliente se active.
- Un puntero a la función de devolución de llamada. El nombre de la función *callback*.
- Un puntero a los datos específicos del cliente que se pasarán a la función de devolución de llamada.

Para remover un evento requiere de los mismos argumentos con excepción del último listado anteriormente.

Una vez que el puntero háptico es detectado ejerciendo un toque en el modelo tridimensional, al presionar el primer botón de la pluma del dispositivo háptico (ver Figura 30), se dispara una bandera *setButtonPressed* como verdadera dentro del *callback buttonDownClientThreadCallback()* el cual corresponde a la acción de presionar el botón. Esta bandera sirve cuando se llama al *callback* relacionado con el evento de movimiento, la denominada como *MotionMoving()*, que al detectar que la bandera está en verdadero (significa que el puntero háptico está en movimiento) agrega puntos de la ruta que está realizando el puntero háptico sobre el modelo tridimensional; posteriormente los puntos son utilizados en una función llamada *drawLines()* que se encarga de unir los puntos mediante líneas con un color rojo para su visualización. Una vez que soltamos el botón podemos ver el resultado del marcaje, esto se realiza en una función de asignación de color, que se genera de manera aleatoria a la ruta marcada, como podemos ver en la Figura 31.

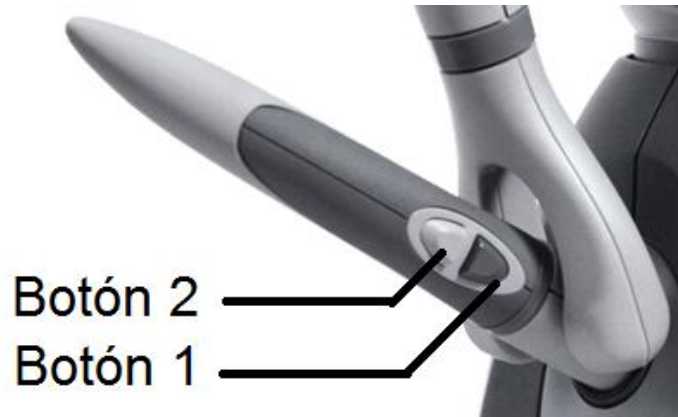


Figura 30. Configuración de botones del dispositivo háptico.

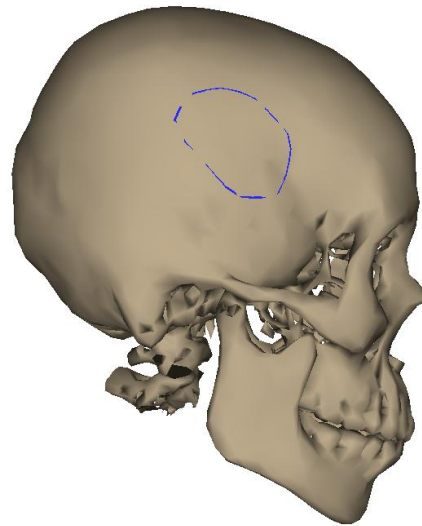


Figura 31. Visualización de la ruta marcada con ayuda del dispositivo háptico sobre el modelo tridimensional.

#### 4.5 Generación del modelo marcado

Una vez que el botón de la pluma del dispositivo háptico se deja de presionar, se manda a llamar a una función de almacenamiento *storeRope()*. Esta función sirve para generar el archivo OBJ que contendrá la lista de puntos que conforman la trayectoria marcada con el dispositivo háptico. El archivo de salida está compuesto por un encabezado que contiene la información de la trayectoria como su longitud y la escala a la que están los puntos en coordenadas de la escena. Al no formar parte del estándar del formato OBJ, se emplea la directiva “#” para que el algoritmo posteriormente ignore esta sección. Las siguientes líneas definen los puntos de la

trayectoria, que se imprimen por un ciclo *for*, donde se le asigna como primer carácter de la cadena la letra “v”, debido a que la estructura del archivo OBJ considera un vértice la cadena iniciada con esa letra. Previo a este último paso, los puntos se multiplican por la escala, de tal forma que al pasarse a Blender las coordenadas de los puntos coincidan con el modelo tridimensional del cráneo. Por último, se imprime la secuencia con la cual se van a unir los puntos de la trayectoria con el identificador “l”, agregando al final de la lista (que define un lazo) el primer valor para cerrar la trayectoria, terminando así la edición del archivo. La Figura 32 muestra un ejemplo del contenido de un archivo OBJ con las características previamente mencionadas.

```
# Longitud de la línea = 1.35052, Escala = 10.7923
o rope
v -7.99046 8.7743 1.95948
v -8.01768 8.80201 1.84275
v -8.07832 8.85435 1.59772
v -8.13489 8.9113 1.08768
v -8.044 9.10233 0.258555
...
v -8.06699 8.1259 2.56392
v -8.08304 8.13264 2.52858
v -8.1669 8.21306 2.304
v -8.18126 8.34761 2.10538
v -8.04007 8.53348 2.21345
l 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 1
```

Figura 32. Ejemplo de archivo OBJ con información de la trayectoria.

#### 4.6 Algoritmo de corte

Después de crearse el archivo OBJ con la ruta marcada sobre el modelo tridimensional del cráneo, se procede a ejecutarse el algoritmo de corte. El algoritmo consta de un script realizado en lenguaje de programación Python que se ejecuta vía Blender, usando su biblioteca *bpy*.



Python es un lenguaje de programación que cuenta con estructuras de datos eficientes y de alto nivel con un enfoque a la programación orientada a objetos. La sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada, lo hacen un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en diversas áreas y para la mayoría de las plataformas. El intérprete de Python y la extensa biblioteca estándar están a libre disposición para las principales plataformas a través de su sitio Web<sup>30</sup>.

Los scripts de Python son una forma poderosa y versátil de extender la funcionalidad de Blender. La mayoría de las funciones de Blender se pueden programar vía Python, incluidas animación, procesamiento, importación, exportación de modelos, creación de objetos y creación de scripts personalizados. Para interactuar con Blender, los scripts pueden hacer uso de la API de Python estrechamente integrada en su interfaz<sup>31</sup>.

Para los fines de este trabajo, se integró la ejecución de comandos de la terminal de Blender desde el programa principal haciendo uso de la instrucción `WinExec()`. Esta instrucción permite la ejecución de programas en Windows desarrollados bajo la plataforma Win32 sin necesidad de abrir la interfaz de Blender, ejecutando el script por línea de comandos<sup>32</sup>. La instrucción necesita dos parámetros, el primero es la línea de comando para la aplicación que se ejecutará y el segundo la opción de visualización; con la opción `SW_SHOWMINIMIZED` hacemos que el algoritmo de Python para Blender se ejecute en segundo plano. Por ejemplo, la siguiente instrucción manda a ejecutar un archivo por lotes `skull_exe_v2.bat`:

```
WinExec("skull_exe_v2.bat", SW_SHOWMINIMIZED);
```

Internamente, el archivo por lotes contiene la instrucción de ejecución del script de Blender como se muestra en la siguiente línea:

---

<sup>30</sup> Página del sitio: <https://www.python.org/>

<sup>31</sup> Referencia: <https://docs.blender.org/manual/en/dev/advanced/scripting/introduction.html>

<sup>32</sup> Referencia: [https://docs.blender.org/api/blender\\_python\\_api\\_2\\_62\\_release/info\\_tips\\_and\\_tricks.html](https://docs.blender.org/api/blender_python_api_2_62_release/info_tips_and_tricks.html)

```
blender --background --python skull_v4.py
```

Esta instrucción, manda a llamar al núcleo de Blender para ejecutarse en segundo plano para procesar un archivo “skull\_v4.py”.

El script de Python inicia con encabezados necesarios para importar una secuencia de comandos y otorgar acceso a datos, clases y funciones de Blender.

```
import bpy
import os
```

La escena de Blender se limpia para poder cargar los modelos OBJ del cráneo y de la ruta marcada, declaramos una ruta a un nuevo archivo el cual contendrá el modelo resultante tras aplicarse el corte. Una vez cargados los modelos los renombramos para facilitar el manejo. Para poder hacer una selección de un elemento dentro de la escena primero es necesario deseleccionar todos los elementos de la escena y con esto procedemos a seleccionar el elemento deseado. Es necesarios que en el script trabajamos con la orientación normal de los objetos, por defecto la orientación es global, este cambio se realiza con las siguientes líneas:

```
for area in bpy.context.screen.areas:
    if area.type == 'VIEW_3D':
        bpy.ops.transform.select_orientation({'area':area},
            orientation="NORMAL")
        break
```

Al terminar estos cambios nos pasamos al modo edición de Blender y el siguiente paso dentro del código es la conversión de la ruta marcada en un modelo tridimensional; para esto, extruimos la región del marcaje en el eje Y y en el Z respecto a la orientación normal del modelo, agregamos una cara, invertimos la selección y agregamos la segunda cara como podemos ver en la siguiente sección de código:

```
bpy.ops.mesh.edge_face_add()
bpy.ops.mesh.select_all(action='INVERT')
bpy.ops.mesh.edge_face_add()
```

Una vez que el modelo de la ruta está completo con sus nuevas caras es trasladado en el eje Z respecto a la orientación normal para posicionarlo dentro del modelo del cráneo. Lo siguiente consiste en agregar el modificador de tipo booleano en el modelo de la ruta marcada para que se aplique la operación diferencial sobre el modelo del cráneo:

```
bool_diff = skull.modifiers.new(type="BOOLEAN", name="bool diff")
bool_diff.object = rope
bool_diff.operation = 'DIFFERENCE'
rope.hide = True
objects['skull'].select = True
bpy.context.scene.objects.active = skull
bpy.ops.object.modifier_apply(apply_as='DATA', modifier="bool diff")
```

#### 4.7 Generación de modelo resultante tras aplicarse el corte

Una vez que la operación booleana es aplicada en el modelo del cráneo se obtiene un nuevo modelo con el corte del marcaje y se procede a exportarlo. La ruta destino (definida al inicio del script) guardará al modelo como un nuevo archivo OBJ.

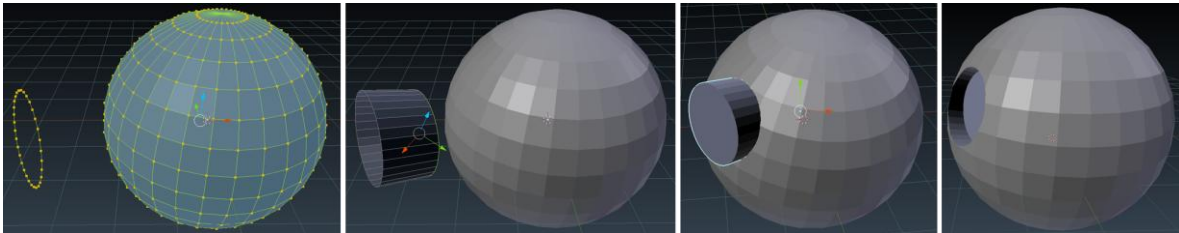
El proceso para exportar dentro del script consta de una breve secuencia de líneas, primero seleccionamos el modelo de la ruta marcada y la borramos, después seleccionamos el modelo del cráneo y procedemos a exportar como se ve en la siguiente sección de código:

```
objects['rope'].select = True
objects['skull'].select = False
bpy.ops.object.delete(use_global=False)
objects['skull'].select = True

bpy.ops.export_scene.obj(filepath=file_loc_exit,
check_existing=True, axis_forward='-Z', axis_up='Y',
filter_glob="*.obj;*.mtl", use_selection=True, use_animation=False,
use_mesh_modifiers=True, use_edges=True, use_smooth_groups=False,
use_smooth_groups_bitflags=False, use_normals=True, use_uv=True,
use_materials=True, use_triangles=True, use_nurbs=False,
use_vertex_groups=False, use_blen_objects=True,
group_by_object=False, group_by_material=False,
keep_vertex_order=False, global_scale=1, path_mode='AUTO')
```

Dentro de la línea encargada de exportar el modelo podemos observar que tiene muchos parámetros, siendo los más importantes la ruta destino (*filepath*) y el tipo de archivo (*filter\_glob*), con esta última línea se termina el script que realiza el proceso de corte con los modelos dentro del programa.

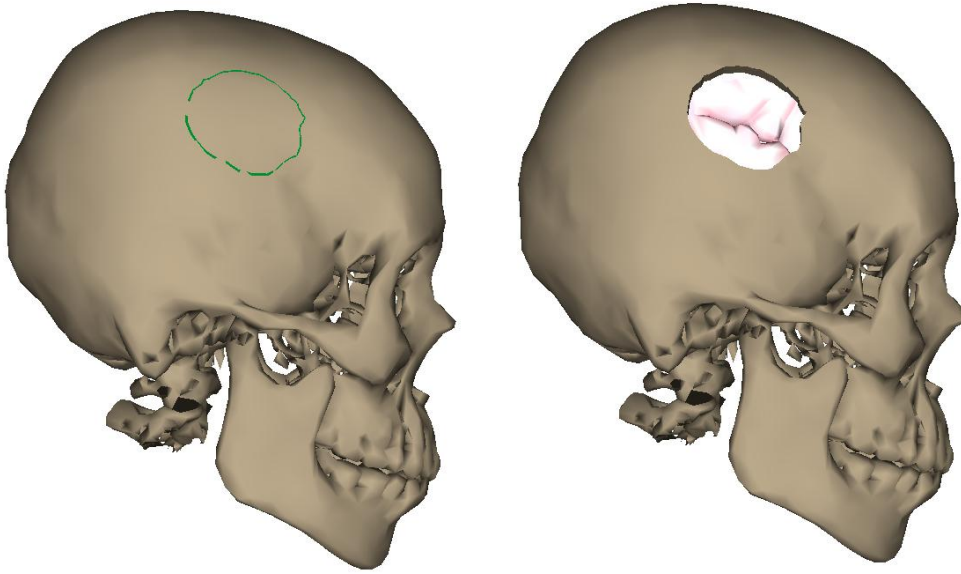
La Figura 33 muestra un resumen del principio de funcionamiento del algoritmo de corte.



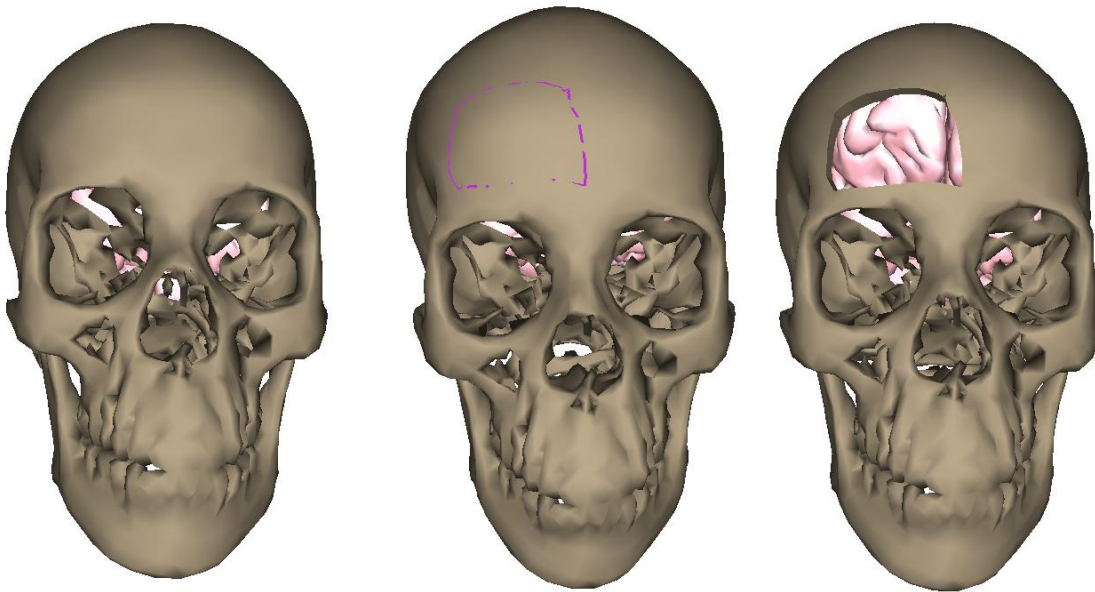
*Figura 33. Resumen del algoritmo de corte usando operaciones booleanas entre una superficie y una trayectoria cerrada*

#### **4.8 Carga de interfaz, visualización y métrica**

El proceso de corte usando el script de Python en Blender se realiza en segundo plano como mencionamos previamente, una vez que se termina de ejecutar procedemos a cargar el archivo OBJ que tiene el modelo resultante tras la aplicación del script. A continuación se presentan imágenes (Figura 34 y Figura 35) de distintos marcajes en diferentes lados del modelo del cráneo así como los resultados tras realizado y ejecutarse el script de corte en el programa.



*Figura 34. Marcaje realizado en la zona derecha del modelo del cráneo (izquierda). Modelo del cráneo resultante de la aplicación del script de corte (derecha).*



*Figura 35. Marcaje realizado en la parte frontal del modelo del cráneo (izquierda). Modelo del cráneo resultante de la aplicación del script de corte (centro). Otra vista del modelo resultante (derecha).*

## 4.9 Audio

En el sistema hay una respuesta auditiva ante la interacción con el programa, para la implementación de audio se utilizó SFML (API portátil y escrita en C++) como biblioteca multimedia ya que posee características para desarrollar software multimedia<sup>33</sup>. Durante la ejecución del programa podemos escuchar dos pistas de audio, la primera es un audio que representan los signos cardiacos monitoreados del paciente, esta pista permanece en constante reproducción hasta terminar el programa y se activa en cuanto los modelos son cargados en la escena; la segunda pista es el sonido del craneótomo, esta se activa cuando se cumplen con dos condiciones, la primera es que el puntero háptico se encuentre tocando al modelo del cráneo y la segunda es cuando se presiona el primer botón del dispositivo háptico.

---

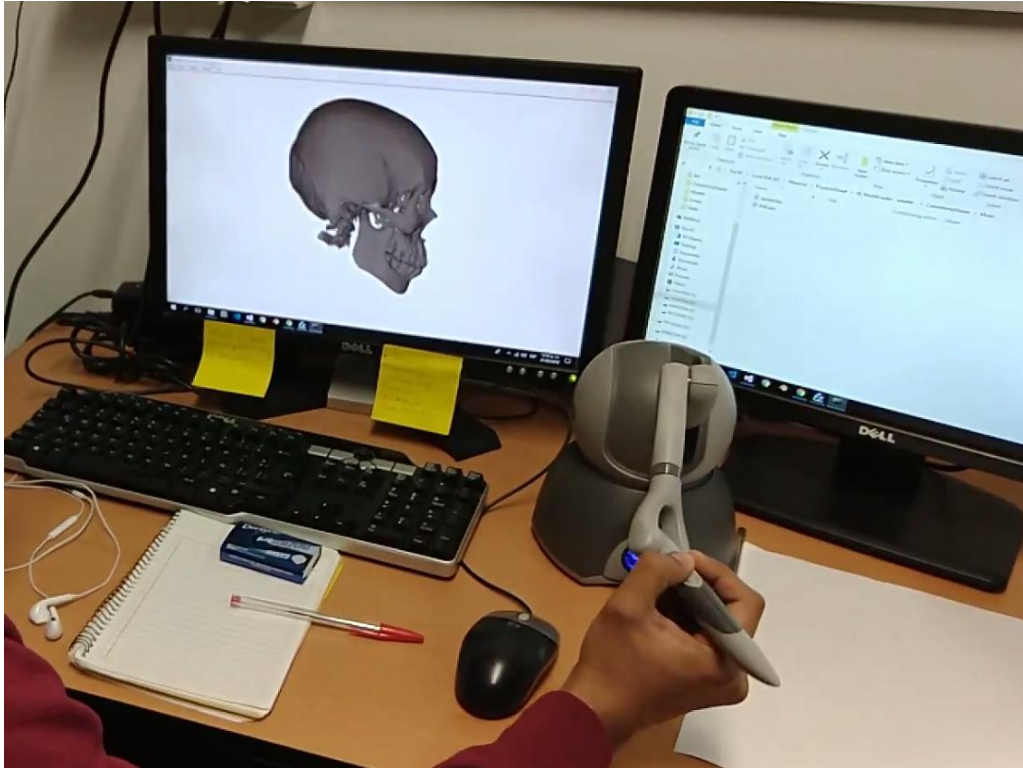
<sup>33</sup> Referencia: <https://www.sfml-dev.org/faq.php#grl-what-is>

# CAPÍTULO 5

## Pruebas y resultados

El resultado final fue el desarrollo de un programa de simulación de corte craneal (craneotomía) para la ayuda en el entrenamiento médico. Este programa desarrollado en C++ y con la ayuda de la herramienta *OpenHaptics* ayudó a brindar una sensación de inmersión debido a la retroalimentación háptica que brinda al programa y al usuario. El funcionamiento del sistema se probó con un modelo tridimensional del cráneo, reconstruido a partir de imágenes de CTA, correspondiente a un caso de un paciente con aneurisma cerebral, cuyo tratamiento requirió de intervención quirúrgica por craneotomía abierta y clipaje.

Para su validación, se presentó el sistema a especialistas en neurocirugía con experiencia en el proceso quirúrgico de la craneotomía, con el propósito de evaluar dicha funcionalidad a nivel de usuario (Figura 36). Por esta razón, se solicitó la ayuda de médicos pertenecientes al Hospital General de México, principalmente al Dr. Aldo Francisco Hernández Valencia, especialista en Neurocirugía del Hospital General de México “Dr. Eduardo Liceaga” y parte de la Unidad de Investigación y Desarrollo Tecnológico de la UNAM. El Dr. Hernández es egresado de la Facultad de Medicina de la Universidad Autónoma de Querétaro, además de contar con certificaciones relacionados a la neurocirugía y un gran número de especialidades, destacando su alta especialidad en cirugía cerebrovascular.



*Figura 36. Resultado final del sistema funcionando con la interfaz háptica.*

Como instrumento de evaluación se utilizó un cuestionario estructurado que consta de 6 preguntas con una posible calificación que van de:

- Totalmente de acuerdo
- A menudo de acuerdo
- Indeciso
- A veces no estoy de acuerdo y
- Totalmente en desacuerdo

A continuación, mostramos las respuestas proporcionadas por médicos del Hospital General de México respecto a la experiencia como usuarios del sistema:



Por favor marque con una (X) la respuesta que concuerde más con su experiencia al usar el simulador

Cuestionario					
Nombre completo: Aldo Francisco Hernández Valencia					
Número de cirugías: _____					
Años de experiencia: 20 años					
Cuestión	Totalmente de acuerdo	A menudo de acuerdo	Indeciso	A veces no estoy de acuerdo	Totalmente en desacuerdo
El movimiento de la herramienta es consistente con respecto a mi mano		X			
La sensación de rigidez y fricción del modelo tridimensional del cráneo se asemeja a la sensación original de un cráneo	X				
Considero adecuada la anatomía del modelo tridimensional del cráneo	X				
La implementación de sonidos ayudo a sentir una mayor sensación de inmersión en el simulador	X				
Creo que el simulador es un buen primer acercamiento para el entrenamiento de la craneotomía	X				
El simulador es fácil de usar		X			

Comentarios:

[Respecto al modelo] Es mucha la resistencia que ofrece y es menos preciso que craneótomo real (Midas Rex) de la marca MEDTRONIC.

[Refiriéndose al entrenamiento] Es un buen primer acercamiento. Con algunas modificaciones será muy útil para enseñarle a mis alumnos.

Por favor marque con una (X) la respuesta que concuerde más con su experiencia al usar el simulador

Cuestionario					
Nombre completo: Jenner Laredo Gómez					
Número de cirugías: _____					
Años de experiencia: 6 años					
Cuestión	Totalmente de acuerdo	A menudo de acuerdo	Indeciso	A veces no estoy de acuerdo	Totalmente en desacuerdo
El movimiento de la herramienta es consistente con respecto a mi mano		X			
La sensación de rigidez y fricción del modelo tridimensional del cráneo se asemeja a la sensación original de un cráneo			X		
Considero adecuada la anatomía del modelo tridimensional del cráneo		X			
La implementación de sonidos ayudo a sentir una mayor sensación de inmersión en el simulador	X				
Creo que el simulador es un un buen primer acercamiento para el entrenamiento de la craneotomía			X		
El simulador es fácil de usar			X		

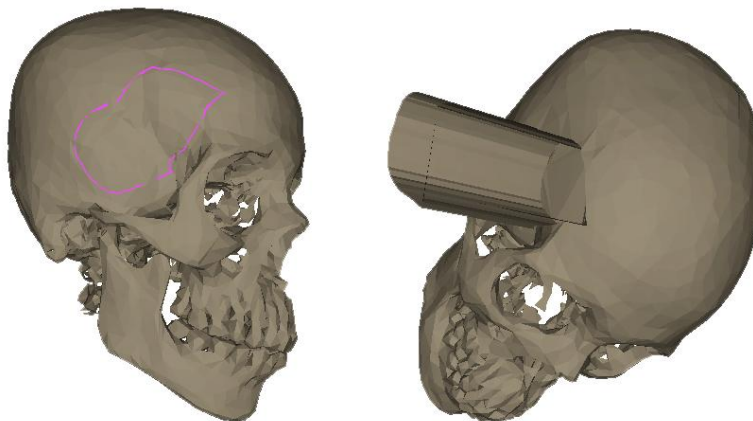
Comentarios:

Creo que puede mejorar mucho al aislar el movimiento (real) de la visión del usuario, usar lentes para limitar la visión periférica, esto mejoraría la experiencia.

El simulador recibió una serie de cambios durante su desarrollo hasta llegar a ser la versión que se presentó ante los médicos del Hospital General, estos cambios fueron entre estéticos como funcionales. En una versión previa, los modelos tridimensionales se veían muy poligonales, se notaba aún la malla que lo formaba, como podemos visualizar en la Figura 37, el script de corte también recibió varias modificaciones, esto fue debido a un error en particular que hacía que el área marcada se desplazara y no realizara la extracción sobre el modelo del cráneo, Figura 38.

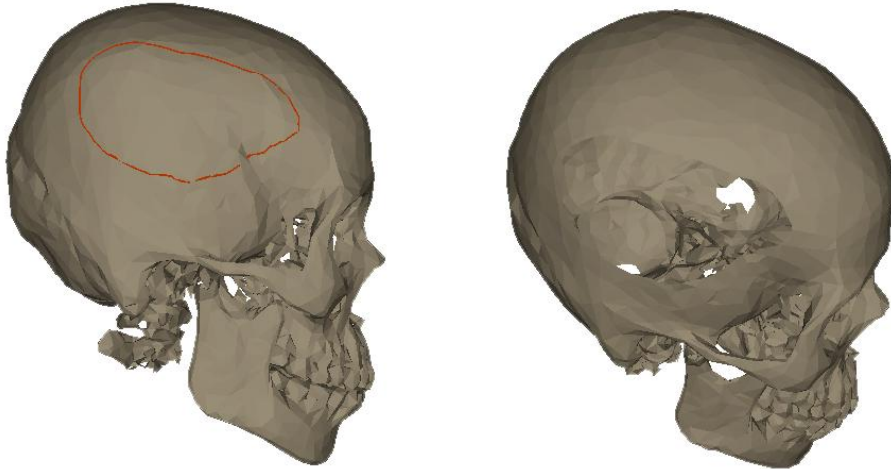


*Figura 37. Modelo tridimensional del cráneo de una versión previa del simulador.*



*Figura 38. La ruta trazada se desplazaba y no realizaba el corte de manera adecuada.*

También podemos mencionar que hay un pequeño desplazamiento del corte en el modelo del cráneo respecto a la ruta marcada sobre él, además, existe una particularidad que se presenta cuando se corta entre la región temporal y parietal del modelo tridimensional, el resultado tras aplicarse el corte provoca que esa sección se visualice más delgada, como si fuera solo un plano, el ejemplo lo podemos ver en la Figura 39.



*Figura 39. Ruta trazada que atraviesa la sección temporal y parietal del modelo del cráneo (Izquierda). Resultado tras aplicarse el script de corte, se nota un delgado corte en la zona marcada, no presenta algún volumen (Derecha).*

# CAPÍTULO 6

## Conclusiones y trabajo a futuro

El entrenamiento en los médicos es el pilar principal para mejorar su técnica, como se mencionó el estudio presentado en el Capítulo 1, la integridad del paciente no puede ser perjudicada en su proceso, por esta razón ellos necesitan hacer uso de dispositivos e instrumentos que logren reforzar su habilidad sin provocar daños, es en este punto donde la medicina puede hacer uso de la ingeniería. La conexión entre la medicina y la ingeniería radica en los avances tecnológicos que poco a poco los médicos han incorporado en su profesión; la robótica y los sistemas computacionales son los principales temas de captan gran interés en la comunidad médica y es la ingeniería en computación la que logra mezclar estos temas para poder ofrecer las herramientas que le ayuden a mejorar su entrenamiento y comprensión del procedimiento quirúrgico.

Como pudo constatarse en el instrumento de evaluación, los usuarios obtuvieron una reacción positiva respecto a la experiencia con el simulador; esta retroalimentación fue de gran utilidad durante todo el proceso de desarrollo. Con las debidas correcciones y la constante evolución de las versiones, se pudo lograr que el sistema fuera mejorando para tener un primer prototipo funcional de una herramienta para el entrenamiento.

El especialista en neurocirugía mencionó una serie de problemáticas que se observan con el residente de nuevo ingreso, el cual, al no contar con una técnica adecuada refleja errores importantes. Asimismo, los novatos no tienen la habilidad necesaria para el control de la instrumentación quirúrgica en sus primeras experiencias; un ejemplo de esto es en el uso del craneótomo, al no tener el manejo

adecuado provocan que la sierra que tiene en la punta se llegue a romper debido a la mala posición en la trayectoria. En este sentido, el simulador desarrollado representa una primera aproximación de un sistema más completo que ayude a la formación académica de los médicos. Así como los pilotos de aviones los cuales se les exige cierto número de horas en el simulador para poder tener un acercamiento a la maquinaria final, a un médico se le exigiría un total de horas previas en un simulador para poder acercarse a un quirófano a practicar el conocimiento aprendido.

El algoritmo de corte presentado es una aproximación al corte que un médico podría realizar sobre un cráneo y trabaja sobre operaciones booleanas; sin embargo, existen otros métodos más sofisticados que involucran cambios topológicos sobre una malla triangular o de volumen. La principal desventaja de estos métodos actualmente es que requieren tiempo de procesamiento y difícilmente trabajan en tiempo real.

Como trabajo a futuro, se espera mejorar la parte de la sensación del modelo del cráneo debido a que la sensación que se implementó es un poco más rígida de lo que debe de ser en comparación con un hueso real, también se busca automatizar ciertas acciones como la del cargado de modelos tras realizarse el corte con el script. A sugerencia del Dr. Hernández, se desea implementar un tipo de actividad dentro del simulador, donde se planteé cierto caso que requiera de una aproximación y corte determinado donde al usuario se le calificarían ciertos rubros y se reflejaría en un marcador final.

# Referencias bibliográficas

3DSYSTEMS. (2015). OpenHaptics Toolkit version 3.4.0: Programmer's Guide, 151. Retrieved from [www.3dsystems.com](http://www.3dsystems.com)

Alaraj, A., Luciano, C. J., Bailey, D. P., Elsenousi, A., Roitberg, B. Z., Bernardo, A., ... Charbel, F. T. (2015). Virtual reality cerebral aneurysm clipping simulation with real-time haptic feedback. *Neurosurgery*, *11 Suppl 2*, 52–58. <https://doi.org/10.1227/NEU.0000000000000583>

Alcañiz, M., Botella, C., Perpiñá, C., Baños, R., & Montesa, J. (2000). *A New Realistic 3D Body Representation in Virtual Environments for the Treatment of Disturbed Body Image in Eating Disorders*. *CyberPsychology & Behavior* (Vol. 3). <https://doi.org/10.1089/10949310050078896>

Delorme, S., Laroche, D., DiRaddo, R., & Del Maestro, R. F. (2012). NeuroTouch: a physics-based virtual simulator for cranial microneurosurgery training. *Neurosurgery*, *71*(1 Suppl Operative), 32–42. <https://doi.org/10.1227/NEU.0b013e318249c744>

Elvins, T. T. (1992). A survey of algorithms for volume visualization. *ACM SIGGRAPH Computer Graphics*, *26*(3), 194–201. <https://doi.org/10.1145/142413.142427>

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

Franc, M., & Skala, V. (2002). Fast algorithm for triangular mesh simplification based on vertex decimation. *Computational Science—ICCS 2002*, 42–51. Retrieved from [http://link.springer.com/chapter/10.1007/3-540-46080-2\\_5](http://link.springer.com/chapter/10.1007/3-540-46080-2_5)

Hayward, V., Astley, O. R., Cruz Hernandez, M., Grant, D., & Robles De La Torre, G. (2004). Haptic interfaces and devices. *Sensor Review VO - 24*, (1), 16. <https://doi.org/10.1108/02602280410515770>

- Hearn, D., Baker, M., & Stroustrup, B. (2004). *Computer Graphics with OpenGL 3/E. Book*. Retrieved from [http://syllabus.cs.vt.edu/cache/www\\_cs\\_rit\\_edu/49-syllabus.pdf](http://syllabus.cs.vt.edu/cache/www_cs_rit_edu/49-syllabus.pdf)
- Lawton, M. T. (2011). *Seven Aneurysms: Tenets and Techniques for Clipping*. Thieme. Retrieved from <https://books.google.com.mx/books?id=wI5kEN8TFpEC>
- Li, Z., & Zhang, J. (2010). Study on volume rendering of CT slices based on ray casting. *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010*, 7(Figure 2), 157–160. <https://doi.org/10.1109/ICCSIT.2010.5564768>
- Luengas C., L. A., López, J., & Sánchez, G. (2017). *Identificación Del Comportamiento De Rangos Articulares Con La Alineación En Amputados Transtibiales. Libro de memorias XII CIECT*.
- Osirix, U., Dicom, A. D., Dicom, T., & Osirix, H. T. (2003). Introduction 1.1, 1–15. <https://doi.org/10.1016/B978-1-85617-745-0.00001-2>
- Padilla, M. A., Teodoro, S., Lira, E., Soriano, D., Altamirano, F., & Arambula, F. (2009). Virtual reality simulator of transurethral resection of the prostate. In *2009 Pan American Health Care Exchanges* (pp. 116–119). <https://doi.org/10.1109/PAHCE.2009.5158378>
- Payne, J. T. (1978). Introduction To Ct. *Journal of Computer Assisted Tomography*, 2(2), 236. <https://doi.org/10.1097/00004728-197804000-00024>
- Peddie, J. (2017). *Augmented Reality: Where we all live. Ercim News* (Vol. October 20). <https://doi.org/10.1007/978-3-319-54502-8>
- Pereira Riveron, R. (2007). Craneotomías, 1–15. <https://doi.org/10.1016/B978-84-8086-373-5.50001-9>
- Riva, G. (2000). *From Telehealth to E-Health: Internet and Distributed Virtual Reality in Health Care. CyberPsychology & Behavior* (Vol. 3).



<https://doi.org/10.1089/109493100452255>

- Riva, G. (2003). Applications of virtual environments in medicine. *Methods Of Information In Medicine*, 42(5), 524–534. Retrieved from <http://pbidi.unam.mx:8080/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=cmedm&AN=14654887&lang=es&site=eds-live>
- Rosset, A., Spadola, L., Pysher, L., & Ratib, O. (2006). Navigating the Fifth Dimension: Innovative Interface for Multidimensional Multimodality Image Navigation1. *RadioGraphics*, 26(1), 299–308. <https://doi.org/10.1148/rg.261055066>
- Shannon, R., & Johannes, J. D. (1976). Systems Simulation: The Art and Science. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(10), 723–724. <https://doi.org/10.1109/TSMC.1976.4309432>
- Teodoro, S., Soriano, D. A., Gastélum, A., Hernández, A., & Padilla, M. Á. (2017). Simuladores virtuales y el entrenamiento médico en neurocirugía.
- Touch, G., X, G. T., Openhaptics, G., & Desktop, P. (2013). Questions or Comments. Volume Rendering. (2007). Retrieved January 1, 2017, from [http://www.byclb.com/TR/Tutorials/volume\\_rendering/Index.aspx](http://www.byclb.com/TR/Tutorials/volume_rendering/Index.aspx)
- Xie, K., Yang, J., & Zhu, Y. M. (2008). Real-time visualization of large volume datasets on standard PC hardware. *Computer Methods and Programs in Biomedicine*, 90(2), 117–123. <https://doi.org/10.1016/j.cmpb.2007.12.006>
- Yasargil, M. G. (1984). Subarachnoid Cisterns. *Microneurosurgery*, 5–52.
- Zhou, J., & Tonnie, K. D. (2003). State of The Art for Volume Rendering Technical Report. *Simulation*, (September), 1–29.

# Tabla de Figuras

Figura 1. Continuo de la Realidad-Virtualidad de Milgram (1994). Imagen extraída de (Peddie, 2017, p. 13). Pág. 1

Figura 2. Ejemplo de aplicaciones del metaverso de la realidad-virtualidad. (a) SimMan3D, Laerdan, Norway, USA, Canada. (b) Microsoft Hololens. (c) Poster: A Virtual Body for Augmented Virtuality by Chroma-Keying of Egocentric Videos. Frank Steinicke. (d) Second Life con el uso de Oculus Rift. Pág. 3

Figura 3. Posiciones del paciente para un procedimiento de craneotomía: decúbito dorsal, supino, decúbito lateral y decúbito prono. Poses realizadas en Blender con ayuda de modelos generados con MakeHuman. Pág. 11

Figura 4. Craneotomía Frontotemporal de Yasargil. En la craneotomía se hacen 4 agujeros de trépano para obtener la pieza del cráneo que se removerá. Cráneo generado con el software ORGANON. Pág. 12

Figura 5. Osteotomías Orbitocigomáticas. La unidad orbitocigomática es liberada por una serie de 6 osteotomías en: (1) Raíz cigomática, (2) Cuerpo cigomático, (3) Fisura orbital inferior, (4) Techo orbital medio, (5) Techo orbital posterior y (6) Pared orbital lateral. Imagen extraída de (Lawton, 2011, p. 35). Osteotomías Orbitocigomáticas. Diferente ángulo de las 6 osteotomías. Imagen extraída de (Lawton, 2011, p. 35). Pág. 13

Figura 6. Existe una modificación en el proceso quirúrgico anterior el cual se usa para abordar los aneurismas de la arteria pericallosal en una región diferente, el enfoque interhemisférico anterior modificado utiliza la gravedad para retraer el hemisferio derecho. Imagen extraída de (Lawton, 2011, p. 38). Pág. 14

Figura 7. Posición del paciente recostado en decúbito supino, se hace uso de refuerzos bajo el hombro izquierdo para poder levantarlo 45 grados y la cabeza es rotada 90 grados a la derecha. Imagen extraída de (Lawton, 2011, p. 38). Pág. 15

Figura 8. Tres maniobras posicionan la cabeza óptimamente: flexión en el plano anteroposterior hasta que el mentón esté a un dedo del esternón; rotación de 45 grados lejos del lado de la lesión, trayendo la nariz abajo hacia el piso, y la flexión lateral 30 grados hacia abajo hacia el suelo. Imagen extraída de (Lawton, 2011, p. 41). Pág. 16

Figura 9. (Izquierda) OperativeExperience tiene un simulador de craneotomía que incluye el poder realizar cortes de piel, agujeros en el cráneo, limpieza tras un sangrado y suturación en un maniquí haciendo uso de los instrumentos quirúrgicos estándar. Imagen extraída de: <https://operativeexperience.com/craniotomy>. (Derecha) SimMan 3G es un simulador de paciente avanzado que puede mostrar tanto los síntomas neurológicos como fisiológicos. Es de fácil manejo e incluye tecnología innovadora, como reconocimiento de fármacos. Imagen extraída de: <https://www.laerdal.com/la/SimMan3G#/Images>. Pág. 20

Figura 10. Componentes de NeuroTouch. Estereoscopio (a), sistemas hápticos (b), fuentes de alimentación y amplificadores para sistemas hápticos (c), computadora (d). Imagen extraída de (Delorme et al., 2012, p. 4). Pág. 21

Figura 11. Residente de Neurocirugía probando un prototipo de NeuroTouch para proporcionar retroalimentación. Imagen extraída de (Delorme et al., 2012, p. 3). Pág. 22

Figura 12. La generación de imágenes de CT se adquieren mediante una rotación rápida de 360° del tubo de rayos X alrededor del paciente. En la imagen se ven los componentes de un escáner CT, el anillo de dispositivos detectores, el tubo emisor de rayos X y el ángulo del abanico de despliegue

y el posicionamiento del paciente dentro del escáner. Imagen extraída de (Payne, 1978, p.4). Pág. 26

Figura 13. Escala de Hounsfield de números CT, algunos ejemplos son la escala que tiene el Hueso, los tejidos blandos, el agua, la grasa, los pulmones y el aire. Extraído de (Payne, 1978, p.7). Pág. 28

Figura 14. Plano Sagital y plano Coronal obtenidas del MPR. Imagen extraída de (Payne, 1978, p.11). Pág. 29

Figura 15. Plano Axial obtenido de MPR. Imagen extraída de (Payne, 1978, p.11). Pág. 29

Figura 16. Reconstrucción tridimensional de arterias cerebrales empleando la técnica de Volume render a partir de imágenes de angiotomografía. Cortesía del Hospital General de México. Pág. 31

Figura 17. Representación del funcionamiento del algoritmo de Opaque Cubes. Extraída de Siggraph: <https://www.siggraph.org/education/materials/HyperVis/vistech/volume/surface3.htm>. Pág. 34

Figura 18. Hay exactamente 256 formas en que cuatro o menos triángulos pueden ajustarse a una celda, y el número de casos puede reducirse a 15 por reflexión y rotación (Elvins, 1992, p. 7). Imagen extraída de ("Volume Rendering," 2007). Pág. 35

Figura 19. Decimación por borde: el borde menos importante se reduce a cero, esto significa que el vértice se mueve al punto final del borde elegido, que debe eliminarse de la malla junto con dos triángulos adyacentes. Imagen extraída de (Franc & Skala, 2002, p. 43). Pág. 37

Figura 20. Proceso de reconstrucción de vascularidades a partir de imágenes de CTA. (a) visualización con 3DSlicer de cortes axiales, sagitales y coronales de un estudio de CTA, y su correspondiente representación (b) usando un método de volumen render basado en ray casting. (c) Visualización de la malla resultante en MeshLab, tras aplicar un método basado en marching cubes con 3DSlicer. (d) Aplicación del algoritmo de decimación por borde en MeshLab. Pág. 38

Figura 21. Los modelos de piel y de cráneo se someten a una reducción manual de triángulos mediante el uso de software para modelado como Blender y un refinamiento de la malla con el uso de MeshLab. Pág. 38

Figura 22. Arquitectura del sistema del simulador. Pág. 40

Figura 23. Un volumen de vista en proyección en perspectiva (frustum) con el plano de vista "enfrente" del plano de corte cercano. Imagen extraída de (Hearn et al., 2004, p. 373). Pág. 42

Figura 24. Diagrama de flujo del programa. Pág. 42

Figura 25. Modelo en 3D del craneótomo. Modelado en Maya y limpiado en Blender, el craneótomo fue modelado con la ayuda de imágenes reales de la herramienta. Pág. 43

Figura 26. Modelos del cráneo y craneótomo virtuales visualizados. El craneótomo sigue los movimientos de la pluma del dispositivo háptico. Pág. 44

Figura 27. Modelo en 3D del cráneo. A la izquierda, representación en modelo sólido. A la derecha, representación de un modelo en malla de alambre. Pág. 45

Figura 28. Movimientos del Geomagic Phantom Omni®. La cabeza principal, de izquierda a derecha, la pluma, movimiento hacia adentro y afuera, el brazo conectado a la pluma, movimiento hacia arriba y abajo. Imagen extraída de (Touch, X, Openhaptics, & Desktop, 2013, p. 22). Pág. 48

*Figura 29. Movimientos del Geomagic Phantom Omni®. Del brazo conectado a la pluma, movimiento de izquierda a derecha, de la pluma, movimiento de arriba hacia abajo y de izquierda a derecha. Imagen extraída de (Touch et al., 2013, p. 23). Pág. 48*

*Figura 30. Configuración de botones del dispositivo háptico. Pág. 51*

*Figura 31. Visualización de la ruta marcada con ayuda del dispositivo háptico sobre el modelo tridimensional. Pág. 51*

*Figura 32. Ejemplo de archivo OBJ con información de la trayectoria. Pág. 52*

*Figura 33. Resumen del algoritmo de corte usando operaciones booleanas entre una superficie y una trayectoria cerrada. Pág. 56*

*Figura 34. Marcaje realizado en la zona derecha del modelo del cráneo (izquierda). Modelo del cráneo resultante de la aplicación del script de corte (derecha). Pág. 57*

*Figura 35. Marcaje realizado en la parte frontal del modelo del cráneo (izquierda). Modelo del cráneo resultante de la aplicación del script de corte (centro). Otra vista del modelo resultante (derecha). 57*

*Figura 36. Resultado final del sistema funcionando con la interfaz háptica. Pág. 60*

*Figura 37. Modelo tridimensional del cráneo de una versión previa del simulador. Pág. 63*  
*Figura 37. Modelo tridimensional del cráneo de una versión previa del simulador.*

*Figura 38. La ruta trazada se desplazaba y no realizaba el corte de manera adecuada. Pág. 63*

*Figura 39. Ruta trazada que atraviesa la sección temporal y parietal del modelo del cráneo (Izquierda). Resultado tras aplicarse el script de corte, se nota un delgado corte en la zona marcada, no presenta algún volumen (Derecha). Pág. 64*