



Universidad Central de Venezuela

Facultad de Ciencias  
Escuela de Computación

***Colocación Virtual del Marco  
Estereotáctico en la Radiocirugía  
Estereotáctica***

Trabajo Especial de Grado  
presentado ante la Ilustre  
Universidad Central de Venezuela  
por el bachiller

**Christiam A. Mena M.**

para optar al título de  
Licenciado en Computación

**Tutores**

Prof. Ernesto Coto

Prof. Miguel Martín

Caracas, 05 de diciembre de 2012

# Universidad Central de Venezuela

Facultad de Ciencias  
Escuela de Computación



## Acta de Veredicto

---

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación para examinar el Trabajo Especial de Grado, presentado por el Bachiller **Christiam Antonio Mena Mijares**, C.I. V-17.758.087, con el título “**Colocación Virtual del Marco Estereotáctico en la Radiocirugía Estereotáctica**”, a los fines de cumplir con el requisito legal para optar al título de Licenciado en Computación, dejan constancia de lo siguiente:

Leído el trabajo por cada uno de los miembros del jurado, se fijó el día 05 de diciembre de 2012, a las 03:00 p.m., para que su autor lo defendiera en forma pública, en el Centro de Computación Gráfica de la Escuela de Computación, lo cual este realizó mediante una exposición oral de su contenido, y luego respondió satisfactoriamente a las preguntas que les fueron formuladas por el jurado, todo ello conforme con lo dispuesto en la Ley de Universidades y demás normativas vigentes de la Universidad Central de Venezuela. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió **APROBARLO**.

En fe de lo cual se levanta la presente acta, en Caracas el 05 de diciembre de 2012, dejándose también constancia de que actuó como coordinador del jurado el Prof. Ernesto Coto, tutor del trabajo. Igualmente se hace constar que actuó como tutor el Prof. Miguel Martín.

---

Prof. Ernesto Coto  
Escuela de Computación, UCV  
(Tutor Firmante)

---

Prof. Héctor Navarro  
Escuela de Computación, UCV  
(Jurado)

---

Dr. Salvador Somaza  
Hospital de Clínicas Caracas  
(Jurado)

---

Prof. Esmitt Ramírez  
Escuela de Computación, UCV  
(Jurado suplente)

---

Prof. Jesús Dávila  
Escuela de Física, UCV  
(Jurado suplente)

# Universidad Central de Venezuela

Facultad de Ciencias  
Escuela de Computación



## Resumen

---

### ***Colocación Virtual del Marco Estereotáctico en la Radiocirugía Estereotáctica***

**Autor:** Br. Christiam Mena

**Tutores:** Prof. Ernesto Coto  
Prof. Miguel Martin

La Radiocirugía Estereotáctica, es un método extremadamente preciso para aplicar altas dosis de radiación ionizante a un volumen de tejido patológico intracraneano concreto y espacialmente definido, de tal forma que se destruya o se altere su funcionamiento. En la actualidad, el procedimiento que se sigue para la colocación del Marco Estereotáctico (ME) sobre el cráneo y su excentricidad con respecto a la patología, se basa en el nivel de experticia y la percepción visual que tenga el médico que lo realiza. Por ende, el procedimiento puede ser catalogado como cualitativo, subjetivo, engorroso y propenso a errores. Un mal posicionamiento del ME, trae como consecuencia inmediata que el tratamiento no se pueda realizar y deba ser postergado. Adicionalmente, la colocación incorrecta del marco puede ocasionar colisiones del mismo con los colimadores o piezas de la máquina radioquirúrgica utilizada, problemática que puede ser identificada únicamente después de fijar el ME en el cráneo. En este trabajo se desarrolla un sistema que provee al neurocirujano de una guía virtual y cuantitativa para la colocación óptima del ME en el cráneo, haciendo uso de las imágenes diagnósticas del paciente obtenidas antes del tratamiento. El sistema desarrollado genera una serie de marcas fiduciales y estereotácticas en los estudios imageneológicos según el posicionamiento virtual del marco, que servirán de entrada para el sistema de planificación. El sistema fue sometido satisfactoriamente a una serie de pruebas cuantitativas y cualitativas con el objetivo de determinar la precisión y exactitud de las marcas fiduciales generadas, y recopilar las experiencias de los usuarios de la aplicación en términos de utilidad, facilidad de uso y cumplimiento de los objetivos de la investigación. Los resultados obtenidos indican que las marcas generadas pudieron ser procesadas por el sistema de planificación de manera transparente, por lo que realizar una pre-planificación del tratamiento radioquirúrgico usando el sistema desarrollado resulta factible.

**Palabras claves:** Radiocirugía Estereotáctica, Estereotáxia, Marco Estereotáctico, Pre-planificación Virtual.

# Agradecimientos

---



*Agradezco a **Dios**, por darme fuerza en tiempos difíciles; por guiarme y nunca abandonarme.*

*Agradezco a mi **madre**, por su incondicional apoyo, amor y cariño; por estar a mi lado siempre haciéndome sonreír; por estar siempre al tanto de mis estudios y de mi carrera; por toda la educación y valores que me inculcó desde pequeño; por todo eso y mucho más... gracias mamá.*

*Agradezco a mi **padre**, por su apoyo e interés en mi educación.*

*Agradezco a **Diana** por su cariño y apoyo en todos estos años de mi vida.*

*Agradezco a mis **hermanas**, por brindarme soporte durante toda la carrera.*

*Agradezco a mis **tutores**, por guiarme durante la realización de este trabajo; por su paciencia, su tiempo y todas sus enseñanzas.*

*Agradezco a todos los **profesores del Centro de Computación Gráfica**, por las horas invertidas en mi formación en el área y su gran apoyo.*

*Agradezco a todos los **profesores** que tuve la oportunidad de conocer y aprender de ellos durante toda la carrera.*

*Agradezco a mis mejores amigos, **Luiyit, Johan, Dayslin y Rafael**, por sus años de amistad; por las horas de alegrías y por su apoyo incondicional.*

*Agradezco a la **Universidad Central de Venezuela**, la mejor universidad de Venezuela por ser responsable de mi formación universitaria.*

*Un especial agradecimiento a la **Unidad de Radiología Estereotáctica Gamma Knife del Centro Diagnóstico Docente** ubicado en Las Mercedes, por la colaboración prestada en el desarrollo de este trabajo.*

*Agradezco a la **Asociación de Egresados y Amigos de la UCV**, por la ayuda económica brindada durante toda mi formación y por mejorar la calidad de mi educación.*

*Agradezco a la **Asociación Venezolana-Americana de la Amistad (AVAA)**, por permitirme formar parte de tan maravilloso programa y contribuir con mi formación académica, profesional y personal; por exigirme y ayudar a despertar mis talentos y por enseñarme otra cara de mi Venezuela.*

*Finalmente, agradezco a todas aquellas personas que de una u otra manera tocaron mi vida durante estos años universitarios; mis panas, compañeros y colegas; preparadores, profesores y obreros de la Facultad de Ciencias... y demás personas que olvido mencionar pero que no son menos importantes.*

**- A todos, gracias -**

# Tabla de Contenido

---

<b>INTRODUCCIÓN .....</b>	<b>11</b>
<b>CAPÍTULO 1: PROPUESTA DE TRABAJO ESPECIAL DE GRADO .....</b>	<b>14</b>
1.1 DEFINICIÓN DEL PROBLEMA.....	14
1.2 OBJETIVOS DE LA INVESTIGACIÓN .....	15
1.2.1 <i>Objetivo General</i> .....	15
1.2.2 <i>Objetivos Específicos</i> .....	15
1.3 JUSTIFICACIÓN, DELIMITACIÓN Y ALCANCE DE LA INVESTIGACIÓN .....	16
1.3.1 <i>Justificación</i> .....	16
1.3.2 <i>Delimitación y alcance de la investigación</i> .....	16
1.4 METODOLOGÍA.....	17
<b>CAPÍTULO 2: LA RADIOCIRUGÍA ESTEREOTÁCTICA .....</b>	<b>19</b>
2.1 CONCEPTOS Y FUNDAMENTOS DE LA RADIOCIRUGÍA ESTEREOTÁCTICA .....	19
2.2 HISTORIA .....	20
2.3 TÉCNICAS DE RADIOCIRUGÍA ESTEREOTÁCTICA .....	22
2.3.1 <i>El Bisturí de Rayos Gamma (Gamma Knife)</i> .....	22
2.3.2 <i>Aceleradores Lineales (LINAC)</i> .....	24
2.3.3 <i>Radiocirugía usando Protones</i> .....	25
2.4 PERSONAL NECESARIO EN LA RADIOCIRUGÍA ESTEREOTÁCTICA .....	26
2.5 SECUENCIA DEL PROCEDIMIENTO .....	27
2.5.1 <i>Establecimiento del sistema de coordenadas</i> .....	27
2.5.2 <i>Adquisición de las imágenes diagnósticas para la planificación dosimétrica</i> .....	28
2.5.3 <i>Planificación del tratamiento</i> .....	28
2.5.4 <i>Radiocirugía: tratamiento</i> .....	29
<b>CAPÍTULO 3: COLOCACIÓN DEL MARCO ESTEREOTÁCTICO .....</b>	<b>32</b>
3.1 SOBRE LA COLOCACIÓN DEL MARCO ESTEREOTÁCTICO .....	32
3.2 TRABAJOS EN COLOCACIÓN ASISTIDA DEL MARCO ESTEREOTÁCTICO .....	33
<b>CAPÍTULO 4: DISEÑO E IMPLEMENTACIÓN.....</b>	<b>38</b>
4.1 ASPECTOS TÉCNICOS.....	38
4.2 ARQUITECTURA DE LA APLICACIÓN <i>VIRTUAL STEREOTACTIC FRAME</i> .....	39
4.3 LA BIBLIOTECA DINÁMICA <i>DICOMLIB</i> .....	40
4.3.1 <i>Paquete «Util»</i> .....	40
4.3.2 <i>Paquete «Volume»</i> .....	43

4.3.3 Paquetes «Opendicom-sharp» y «ClearCanvas» .....	47
4.4 LA BIBLIOTECA DINÁMICA <i>MODELLIB</i> .....	48
4.5 LA BIBLIOTECA DINÁMICA <i>UTILITYLIB</i> .....	51
4.5.1 Paquete «Glu» .....	53
4.5.2 Paquete «Math» .....	53
4.5.3 Paquete «Graphics» .....	53
4.5.4 Paquete «Transform» .....	57
4.5.5 La clase <i>Shader</i> .....	58
4.5.6 La clase <i>Camera</i> .....	59
4.6 LA APLICACIÓN <i>VIRTUAL STEREOTACTIC FRAME</i> .....	60
4.6.1 La barra de menús principal .....	61
4.6.2 Menús principales a detalle .....	62
4.6.3 La barra de herramientas principal .....	63
4.6.4 Control y configuración de los cuatro Viewports .....	64
4.6.5 El panel de configuración del Marco Estereotáctico .....	68
4.6.6 Barra de herramientas inferior .....	69
4.6.7 Función de transferencia .....	70
4.6.8 Ventana de configuración de las marcas fiduciales .....	72
4.6.9 Visualizador de imágenes DICOM ( <i>DICOM Viewer</i> ) .....	73
4.6.10 Ventana de bienvenida .....	77
<b>CAPÍTULO 5: PRUEBAS Y RESULTADOS .....</b>	<b>79</b>
5.1 PRUEBAS Y RESULTADOS CUANTITATIVOS .....	79
5.1.1 Localización de los centros en las marcas fiduciales .....	79
5.1.2 Distancias entre las marcas fiduciales .....	80
5.1.3 Errores absolutos y errores relativos porcentuales .....	84
5.1.4 Introducción de los datasets generados en el sistema de planificación .....	88
5.2 PRUEBAS Y RESULTADOS CUALITATIVOS .....	90
5.2.1 Análisis de los resultados .....	91
<b>CAPÍTULO 6: CONCLUSIONES Y TRABAJOS FUTUROS .....</b>	<b>94</b>
6.1 CONCLUSIONES .....	94
6.2 TRABAJOS FUTUROS .....	95
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>97</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>101</b>

# Índice de Figuras

---

FIGURA 2.1 EL <i>GAMMA KNIFE</i> DESARROLLADO POR LARS.....	21
FIGURA 2.2 IMAGEN DEL LINAC.....	21
FIGURA 2.3 CORTE ESQUEMÁTICO DEL <i>GAMMA KNIFE</i> EN DONDE SE PUEDE APRECIAR LA DISTRIBUCIÓN DE SUS PARTES.....	22
FIGURA 2.4 COLOCACIÓN DEL MARCO ESTEREOTÁCTICO EN EL PACIENTE Y UBICACIÓN DE LOS CUATRO TORNILLOS.....	23
FIGURA 2.5 CENTRO DE LA LESIÓN COINCIDE CON EL ISOCENTRO DEL <i>GAMMA KNIFE</i> .....	23
FIGURA 2.6 PARTES PRINCIPALES DEL LINAC.....	24
FIGURA 2.7 ROTACIÓN DEL <i>GANTRY</i> Y DE LA MESA DEL LINAC, AMBOS MOVIMIENTOS ROTATORIOS SON ISOCÉNTRICOS.....	25
FIGURA 2.8 MARCO ESTEREOTÁCTICO Y COLOCACIÓN DEL MISMO EN EL PACIENTE HACIENDO USO DE LOS CUATRO TORNILLOS.....	27
FIGURA 2.9 CAJA CON MARCAS RADIOPACAS PARA LOCALIZAR EL BLANCO RADIOQUIRÚRGICO.....	28
FIGURA 2.10 <i>SOFTWARE</i> DE PLANIFICACIÓN IPLAN STEREOTAXY DE BRAINLAB.....	29
FIGURA 3.1 PERSPECTIVA CONVENCIONAL DEL “ <i>SISTEMA PARA CENTRAR EL MARCO ESTEREOTÁCTICO EN EL CRÁNEO</i> ” DE NIETO ET AL. Y VISUALIZACIÓN TRANSVERSAL DEL ACOPLAMIENTO AL ME.....	34
FIGURA 3.2 DESPLIEGUE DEL ME SOBRE LA RM DEL PACIENTE.....	35
FIGURA 3.3 MARCAS FIDUCIALES GENERADAS POR EL SISTEMA SOBRE UNA IMAGEN DE RM.....	35
FIGURA 3.4 IMAGEN DE UNA CABEZA FANTASMA ANTROPOMÓRFICA ( <i>ANTHROPOMORPHIC HEAD PHANTOM</i> ).....	36
FIGURA 4.1 DISEÑO GENERAL E INTERACCIÓN ENTRE LAS BIBLIOTECAS DINÁMICAS EN LA APLICACIÓN <i>VIRTUAL STEREOTACTIC FRAME</i> .....	39
FIGURA 4.2 DIAGRAMA DE CLASES DE LA BIBLIOTECA DINÁMICA <i>DICOMLIB</i> .....	41
FIGURA 4.3 DESPLIEGUE DE VOLÚMENES UTILIZANDO TEXTURAS 2D. SE ALMACENAN 3 COPIAS DEL VOLUMEN, Y SE SELECCIONA LA PILA DE CORTES MÁS PERPENDICULAR A LA DIRECCIÓN DE VISUALIZACIÓN.....	44
FIGURA 4.4 DESPLIEGUE DE POLÍGONOS TEXTURIZADOS ALINEADOS AL OBJETO. LOS POLÍGONOS ALINEADOS (A) SON TEXTURIZADOS UNO POR UNO, ASIGNÁNDOLE COORDENADAS DE TEXTURA A CADA VÉRTICE (B). CADA POLÍGONO TEXTURIZADO (C) ES MEZCLADO MEDIANTE FUNCIÓN DE MEZCLA DEL API DE <i>OPENGL</i> , PARA OBTENER LA IMAGEN RESULTANTE (D). .....	44
FIGURA 4.5 GUÍA PARA LA VISUALIZACIÓN DE LAS MARCAS FIDUCIALES EN EL ESPACIO.....	45
FIGURA 4.6 (A) SE VISUALIZAN LOS FACTORES QUE INTERVIENEN EN EL PROCESO PARA HALLAR LA CORRESPONDENCIA ENTRE UN PUNTO EN EL ESPACIO OBJETO Y UN VÓXEL, (B) IMAGEN EN DONDE SE VISUALIZAN LOS VÓXELES PERTENECIENTES AL MAPEO. ....	46
FIGURA 4.7 DESPLIEGUE DE LA MARCA FIDUCIAL HACIENDO USO DE UNA MÁSCARA CIRCULAR. EN LA IMAGEN SE PUEDE APRECIAR QUE EL VÓXEL CENTRAL TIENE MAYOR INTENSIDAD QUE LOS VÓXELES ADYACENTES.....	47
FIGURA 4.8 PIEZAS DEL MARCO ESTEREOTÁCTICO DISEÑADAS CON <i>GOOGLE SKETCHUP 8</i> ©, .....	49
FIGURA 4.9 DIAGRAMA DE CLASES DE LA BIBLIOTECA DINÁMICA <i>MODELLIB</i> .....	50
FIGURA 4.10 DIAGRAMA DE CLASES DE LA BIBLIOTECA DINÁMICA <i>UTILITYLIB</i> .....	52
FIGURA 4.11 VISUALIZACIÓN DEL COMPONENTE <i>VIEWCUBE</i> CON TODAS SUS CONFIGURACIONES. LAS VISTAS POSIBLES SON: (A) DERECHA, (B) IZQUIERDA, (C) SUPERIOR, (D) INFERIOR, (E) ANTERIOR O FRONTAL, (F) POSTERIOR O TRASERA Y (G) .....	

PERSONALIZABLE. EN LA IMAGEN SE PUEDE APRECIAR EL ICONO DE INICIO (IMAGEN DE UNA CASA) EL CUAL RESTAURA LA CÁMARA A SU POSICIÓN INICIAL. ....	54
FIGURA 4.12 VISUALIZACIÓN DEL COMPONENTE DE ROTACIÓN ( <i>ROTATIONTOOL</i> ). ....	54
FIGURA 4.13 VISUALIZACIÓN DE LA HERRAMIENTA DE TRASLACIÓN ( <i>MOVEMENTTOOL</i> ). ....	55
FIGURA 4.14 VISUALIZACIÓN DEL COMPONENTE DE MEDICIÓN. ....	55
FIGURA 4.15 VISUALIZACIÓN DEL COMPONENTE DE <i>CLIPPING</i> DEL VOLUMEN. EN (A) SE MUESTRAN LOS PLANOS DE <i>CLIPPING</i> Y EN (B) SE MUESTRA EL COMPONENTE QUE PERMITE AJUSTAR LAS POSICIONES DE LOS PLANOS. ....	56
FIGURA 4.16 VISUALIZACIÓN DEL COMPONENTE <i>GRID</i> . ....	57
FIGURA 4.17 ELEMENTOS PRINCIPALES QUE CONFORMAN LA ESTRUCTURA DE UNA CÁMARA. ....	60
FIGURA 4.18 ELEMENTOS QUE CONFORMAN LA INTERFAZ GRÁFICA DE LA APLICACIÓN <i>VIRTUAL STEREOTACTIC FRAME</i> . ....	61
FIGURA 4.19 BARRA DE MENÚS PRINCIPAL. ....	62
FIGURA 4.20 PARTES Y CARACTERÍSTICAS DE LOS MENÚS. ....	62
FIGURA 4.21 LISTA DE CONFIGURACIONES PREDEFINIDAS PARA ALINEAR LOS <i>VIEWPORTS</i> . A) ES LA CONFIGURACIÓN POR DEFECTO Y ALINEA LOS <i>VIEWPORTS</i> EN DOS FILAS Y DOS COLUMNAS; B) EN ESTA CONFIGURACIÓN SE CONSERVA LA PRIMERA COLUMNA Y LA SEGUNDA QUEDA CONFORMADA POR EL <i>VIEWPORT</i> INFERIOR DERECHO; C) SIMILAR A LA ANTERIOR PERO CON EL <i>VIEWPORT</i> SUPERIOR DERECHO; D) LA PRIMERA COLUMNA QUEDA CONFORMADO POR EL <i>VIEWPORT</i> INFERIOR IZQUIERDO Y LA SEGUNDA COLUMNA NO REFLEJA CAMBIOS; E) SIMILAR A LA ANTERIOR PERO CON EL <i>VIEWPORT</i> SUPERIOR IZQUIERDO; F) Y G) DESPLIEGA LOS <i>VIEWPORTS</i> EN COLUMNAS DE UN SOLO <i>VIEWPORT</i> CADA UNA; H) E I) DESPLIEGA LOS <i>VIEWPORTS</i> EN FILAS DE UN SOLO <i>VIEWPORT</i> CADA UNA. ....	63
FIGURA 4.22 DIFERENTES BOTONES EN LAS BARRAS DE HERRAMIENTAS. ....	64
FIGURA 4.23 DIFERENCIAS ENTRE LA VISTA PERSPECTIVA (IZQUIERDA) Y VISTA ORTOGRÁFICA (DERECHA). ....	66
FIGURA 4.24 ELEMENTOS QUE CONFORMAN EL <i>VIEWPORT</i> . ....	67
FIGURA 4.25 MENÚ CONTEXTUAL PARA LA CONFIGURACIÓN DE LOS <i>VIEWPORTS</i> . ....	67
FIGURA 4.26 PANEL DE CONFIGURACIÓN DEL <i>ME</i> . ....	68
FIGURA 4.27 BARRA DE HERRAMIENTAS INFERIOR. ....	70
FIGURA 4.28 VENTANA DE FUNCIÓN DE TRANSFERENCIA. ....	71
FIGURA 4.29 LAS DIFERENTES CONFIGURACIONES POR DEFECTO DE LA FUNCIÓN DE TRANSFERENCIA. ....	72
FIGURA 4.30 VENTANA DE CONFIGURACIÓN DE LAS MARCAS FIDUCIALES. ....	72
FIGURA 4.31 ELEMENTOS QUE CONFORMAN LA INTERFAZ GRÁFICA DEL VISOR <i>DICOM (DICOM VIEWER)</i> . ....	74
FIGURA 4.32 ELEMENTOS QUE CONFORMAN EL <i>VIEWPORT</i> DEL VISOR <i>DICOM</i> . ....	76
FIGURA 4.33 VENTANA DONDE SE MUESTRAN LOS DATOS DE LA CABECERA <i>DICOM</i> . ....	76
FIGURA 4.34 VENTANA DE BIENVENIDA. ....	77
FIGURA 5.1 ALINEACIÓN DE LOS CENTROS DE LA MARCAS FIDUCIALES Y LAS HERRAMIENTAS DE MEDICIÓN UTILIZADAS. ....	80
FIGURA 5.2 MARCA FIDUCIAL EN FORMA DE CRUZ. ....	80
FIGURA 5.3 <i>DATASETS</i> UTILIZADOS PARA EL CÁLCULO DE LOS ERRORES ENTRE LAS MARCAS FIDUCIALES DE LA APLICACIÓN Y LAS GENERADAS POR UNA MÁQUINA DE RESONANCIA MAGNÉTICA. ....	81
FIGURA 5.4 MEDICIONES Y ENUMERACIÓN UTILIZADA PARA IDENTIFICAR LAS MARCAS FIDUCIALES. ....	82
FIGURA 5.5 <i>DATASETS</i> UTILIZADOS PARA EL CÁLCULO DE LOS ERRORES ABSOLUTOS Y RELATIVOS. ....	85
FIGURA 5.6 MEDIDAS REALES DE LOS SEGMENTOS DE LAS MARCAS FIDUCIALES. (A) DISTANCIA DEL SEGMENTO HORIZONTAL, (B) DISTANCIA DEL SEGMENTO VERTICAL Y (C) DISTANCIA ENTRE LOS CONJUNTOS DE MARCAS FIDUCIALES. ....	86
FIGURA 5.7 <i>DATASETS</i> IMPORTADOS EN EL SISTEMA DE PLANIFICACIÓN RADIOQUIRÚRGICO CON LAS MARCAS FIDUCIALES GENERADAS POR LA APLICACIÓN DESARROLLADA. ....	89

# Índice de Algoritmos

---

ALGORITMO 4.1 MÉTODO <i>EXPLOREDIRECTORY()</i> DE LA CLASE <i>DICOMMANAGER</i> . .....	42
ALGORITMO 4.2 PASOS QUE SE SIGUEN EN LA INVOCACIÓN DEL MÉTODO <i>GENERATEMARKS()</i> DE LA CLASE <i>FIDUCIALMARKS</i> . ....	45
ALGORITMO 4.3 CORRESPONDENCIA ENTRE UN PUNTO EN ESPACIO OBJETO Y UN VÓXEL. MÉTODO <i>GENERATEMARKS()</i> .....	46

# Índice de Tablas

---

TABLA 4.1 LISTA DE COMANDOS ENCONTRADOS EN LA BARRA DE HERRAMIENTAS PRINCIPAL DE LA APLICACIÓN VSF. ....	65
TABLA 4.2 LISTA DE COMANDOS ENCONTRADOS EN LA BARRA DE HERRAMIENTAS PRINCIPAL DEL VISOR DICOM.....	75
TABLA 5.1 CARACTERÍSTICAS DE LOS <i>DATASETS</i> UTILIZADOS PARA EL CÁLCULO DE LOS ERRORES ENTRE LA MARCAS FIDUCIALES DE LA APLICACIÓN Y LAS GENERADAS POR UNA MÁQUINA DE RESONANCIA MAGNÉTICA. ....	81
TABLA 5.2 ERRORES DE DISTANCIA PARA LA MUESTRA #1. ....	83
TABLA 5.3 ERRORES DE DISTANCIA PARA LA MUESTRA #2. ....	83
TABLA 5.4 ERRORES DE DISTANCIA PARA LA MUESTRA #3. ....	84
TABLA 5.5 CARACTERÍSTICAS DE LOS <i>DATASETS</i> UTILIZADOS PARA EL CÁLCULO DE LOS ERRORES ABSOLUTOS Y RELATIVOS. ....	85
TABLA 5.6 DETALLES DE LOS ERRORES ABSOLUTOS Y RELATIVOS PORCENTUALES DE LA MUESTRA #4. ....	86
TABLA 5.7 DETALLES DE LOS ERRORES ABSOLUTOS Y RELATIVOS PORCENTUALES DE LA MUESTRA #5. ....	87
TABLA 5.8 DETALLES DE LOS ERRORES ABSOLUTOS Y RELATIVOS PORCENTUALES DE LA MUESTRA #6. ....	88
TABLA 5.9 CARACTERÍSTICAS DE LOS <i>DATASETS</i> IMPORTADOS EN EL SISTEMA DE PLANIFICACIÓN RADIOQUIRÚRGICO. ....	89

# Introducción

---

La Radiocirugía Estereotáctica es un método radioquirúrgico mínimamente invasivo que permite concentrar altas dosis de radiación ionizante en una fracción única, sobre una lesión o estructura anatómica bien definida volumétrica y espacialmente, con el fin de destruirla o alterar su función, respetando los tejidos circundantes. Dicho procedimiento se logra haciendo converger múltiples haces de radiación de baja dosis en un volumen intracraneano definido como “*blanco quirúrgico*” mediante la aplicación de los principios de la estereotáxia.

La estereotáxia es una técnica que consiste en definir un punto o volumen en un espacio determinado, según su posición en un eje de coordenadas cartesianas X, Y y Z, en relación con un sistema de referencia conocido. Como punto de referencia se usa un marco de metal rígido fijado a la cabeza a través de cuatro tornillos. Dicho marco es conocido como Marco Estereotáctico (ME).

El Marco Estereotáctico ofrece una guía tridimensional y establece un sistema de coordenadas reproducibles para localización sub-milimétrica de las anomalías en el cerebro. Sobre éste son colocados unos localizadores que llevan marcas radiopacas que fundamentalmente sirven para localizar la profundidad del “*blanco quirúrgico*” en los estudios imagenológicos.

En la actualidad el procedimiento que se sigue para la colocación del marco sobre el cráneo se basa exclusivamente en la experiencia y la percepción visual del médico que lo realiza, apoyándose únicamente en imágenes médicas del paciente. Por lo tanto, el procedimiento se considera subjetivo y propenso a errores.

Este trabajo tiene como objetivo el desarrollo de un sistema que permita simular en un ambiente virtual y controlado la colocación del Marco Estereotáctico en el cráneo, haciendo uso de las imágenes diagnósticas del paciente obtenidas antes del tratamiento radioquirúrgico. La finalidad del sistema es subsanar el problema de excentricidad existente al momento de la colocación del ME, además de permitir que los datos de salida proporcionados por la aplicación sean útiles para realizar la pre-planificación de la radiocirugía.

Con el fin de lograr los objetivos establecidos, se ha estructurado el trabajo en cinco capítulos que explicarán los diferentes aspectos que se tomarán en cuenta para la creación del sistema. A continuación, se ofrece un breve resumen del contenido de cada uno de estos capítulos:

En el primer capítulo se presenta la propuesta del Trabajo Especial de Grado donde se plantea la problemática existente con respecto a la correcta colocación del Marco Estereotáctico, se muestran en detalle los objetivos de la investigación, el alcance y la metodología que se seguirá para el desarrollo del sistema propuesto.

En el segundo capítulo se describen las bases y conceptos principales de la Radiocirugía Estereotáctica, su historia y evolución. Se describen sus técnicas fundamentales, se mencionan los miembros del equipo de trabajo que interviene en el procedimiento y se aborda la secuencia del procedimiento radioquirúrgico.

En el capítulo tres se describen los problemas asociados a una mala colocación del Marco Estereotáctico y se presentan una serie de publicaciones y trabajos previos que ayudan a solventar dicha problemática haciendo uso de dispositivos mecánicos y/o herramientas de *software*.

En el cuarto capítulo se muestra el diseño general del sistema, incluyendo la aplicación y todas las bibliotecas dinámicas desarrolladas, abarcando los detalles de la implementación de las funcionalidades más importantes.

El quinto capítulo muestra el análisis de los resultados obtenidos tras realizar las pruebas cualitativas del sistema, obtenidas mediante encuestas y entrevistas realizadas a los doctores del área.

Para finalizar, se resumen las conclusiones obtenidas durante la investigación y desarrollo, tomando en cuenta los resultados de las pruebas y se proponen posibles trabajos a futuro para la continuación de la investigación y desarrollo en el área.

# 1

CAPÍTULO 1

# Propuesta de Trabajo Especial de Grado

---

En este capítulo:

***1.1 Definición del Problema***

***1.2 Objetivos de la investigación***

***1.3 Justificación, delimitación y alcance de la investigación***

# Capítulo 1

---

## *Propuesta de Trabajo Especial de Grado*

Se presenta a continuación la propuesta del Trabajo Especial de Grado, donde se define la problemática existente en la actualidad con respecto a la correcta colocación del marco estereotáctico en la Radiocirugía Estereotáctica. Se plantea el desarrollo de un *software* como objetivo general. Dicha aplicación será capaz de proveer al neurocirujano una guía virtual tridimensional y cuantitativa para la colocación del Marco Estereotáctico. Adicionalmente, se especifica la metodología que se seguirá para el desarrollo del sistema propuesto.

### **1.1 Definición del Problema**

Durante los últimos años el uso de sistemas y herramientas computarizadas como parte esencial en los procedimientos neuroquirúrgicos ha cobrado un mayor auge a nivel mundial. En nuestro país, la aplicación y adopción de dicha tecnología por parte de los neurocirujanos es aún muy limitada [1]. Esta limitación se debe, en parte, a los altos costos de adquisición de los sistemas y el escaso desarrollo de aplicaciones médicas orientadas a resolver necesidades específicas del neurocirujano, particularmente en el área de la Radiocirugía Estereotáctica (RE).

La RE es un procedimiento radioquirúrgico no invasivo que consiste en aplicar altas dosis de radiación a una patología intracraneana concreta y espacialmente conocida. La técnica se caracteriza por el uso de la estereotaxia, que no es más que el establecimiento de una correspondencia espacial entre las estructuras anatómicas del paciente y un sistema de coordenadas referenciales ubicadas en un dispositivo auxiliar.

Dentro del ámbito de la radiocirugía, dicho dispositivo auxiliar corresponde a un Marco Estereotáctico (ME) que es fijado al cráneo del paciente durante todo el tratamiento radioquirúrgico. El objeto de este marco es inmovilizar la cabeza del paciente durante la radiocirugía y constituir un punto de referencia para los cálculos de planificación posteriores. El ME se halla fijo al cráneo de tal manera que permite establecer un sistema de coordenadas tridimensionales reproducibles para la localización precisa de las anomalías en el cerebro y la elaboración del plan de tratamiento.

En la actualidad, el procedimiento que se sigue para la colocación del marco sobre el cráneo y su excentricidad con respecto a la patología, se basa en el nivel de experticia y la percepción visual que tenga el médico que lo realiza. Por ende, el procedimiento puede ser catalogado como cualitativo, subjetivo, engorroso y propenso a errores.

Un mal posicionamiento del ME, trae como consecuencia inmediata que el tratamiento no se pueda realizar y deba ser postergado. Adicionalmente, la colocación incorrecta del marco puede ocasionar colisiones del mismo con los colimadores o piezas de la máquina radioquirúrgica utilizada, problemática que puede ser identificada únicamente después de fijar el ME en el cráneo.

Cualquiera sea el caso, si el volumen de interés no puede tratarse con la colocación inicial del ME entonces se necesitará la remoción y el nuevo posicionamiento del mismo, o en caso extremo una nueva planificación del tratamiento. Finalmente, es importante señalar que como secuelas de una mala colocación inicial del marco, el paciente puede sufrir un alto grado de incomodidad durante el procedimiento, se requerirán mayores recursos y la duración del tratamiento será mucho mayor.

La propuesta consiste en desarrollar un *software* que provea al neurocirujano de una guía virtual y cuantitativa para la colocación óptima del ME. El sistema será capaz de desplegar la representación virtual 3D de la anatomía del paciente (específicamente la cabeza), obtenida de las imágenes diagnósticas que se obtienen antes de realizar el tratamiento. Adicionalmente, le permitirá al usuario seleccionar el ME a utilizar, variar el tamaño de los postes y tornillos del mismo, así como la colocación de la caja radiopaca haciendo uso de la interfaz del sistema.

## 1.2 Objetivos de la investigación

### 1.2.1 Objetivo General

Desarrollar un sistema que permita simular en un ambiente virtual la colocación del marco estereotáctico en el cráneo, haciendo uso de las imágenes diagnósticas del paciente obtenidas antes del tratamiento. El sistema deberá generar una serie de marcas fiduciales y estereotácticas en los estudios imageneológicos según el posicionamiento virtual del marco que servirán de entrada para el sistema de planificación.

### 1.2.2 Objetivos Específicos

- Analizar el proceso de colocación del ME sobre el cráneo del paciente para determinar todas las variables que intervienen e identificar las limitaciones del procedimiento.
- Elaborar el diseño de las interfaces gráficas de usuario y definir toda la interacción del usuario con el sistema.
- Cargar en el sistema las imágenes en formato DICOM (*Digital Imaging and Communication in Medicine* – Imágenes Digitales y Comunicación en Medicina) provenientes de los estudios imageneológicos de cada paciente.
- Desplegar las imágenes DICOM usando la técnica de *Volume Rendering* (Despliegue de Volúmenes) denominada planos alineados al objeto con texturas 2D.
- Combinar el despliegue de las imágenes DICOM con un algoritmo de despliegue de superficies para mezclar el despliegue de las imágenes médicas con el de los Marcos Estereotácticos.
- Modelar en 3D y cargar al sistema los Marcos Estereotácticos junto con todos sus componentes basándose en las escalas reales.

- Implementar un módulo para la generación de marcas fiduciales sobre las imágenes DICOM de entrada, en donde se reflejen las guías estereotácticas de la caja fiducial.
- Salvar las imágenes del paciente modificadas con las marcas fiduciales, en formato DICOM.
- Realizar las pruebas del sistema.

## 1.3 Justificación, delimitación y alcance de la investigación

### 1.3.1 Justificación

En la actualidad el procedimiento utilizado para la colocación del ME sobre el cráneo del paciente está basado completamente en la pericia y percepción visual del médico que lo realiza, apoyándose únicamente en el uso de imágenes diagnósticas adquiridas antes de realizar el tratamiento. Dicho procedimiento es subjetivo y propenso a errores humanos.

En este trabajo se propone desarrollar un *software* que permita simular en un ambiente virtual el posicionamiento del ME en el cráneo. Dicho sistema representará una guía tridimensional que permita al neurocirujano la colocación óptima y objetiva del marco. Adicionalmente, el *software* proporcionará todas las características de usabilidad y una interfaz gráfica de usuario que facilite tanto el aprendizaje de la herramienta como el uso de la misma.

Desde el punto de vista metodológico, la investigación y el desarrollo del sistema poseen un carácter innovador, y además pueden servir de modelo para futuras aplicaciones en el campo de la Computación Gráfica y la Radiocirugía Estereotáctica.

El desarrollo del sistema generará beneficios expresados en la optimización del proceso de colocación del ME; minimizando el número de re-planificaciones de tratamientos y/o distintas colocaciones del ME y disminuyendo el grado de incomodidad del paciente una vez colocado el ME y durante toda la radioterapia.

Por último, la investigación pondrá en manifiesto los conocimientos adquiridos durante la carrera y permitirá sentar las bases para otros estudios que surjan partiendo de la problemática aquí especificada.

### 1.3.2 Delimitación y alcance de la investigación

La investigación se enfocará en el desarrollo de un sistema que permita simular en un ambiente virtual la colocación cuantitativa del ME en el cráneo. Dicho sistema dispondrá de una interfaz gráfica de usuario donde se desplegarán los modelos de los MEs diseñados y el volumen que representa el cráneo del paciente haciendo uso de una o más técnicas de *Volume Rendering*. Es importante señalar que el *Volume Rendering* que se utilizará permitirá la visualización de todas las capas de tejidos intracraneales haciendo uso de una función de transferencia. El objetivo de dicha característica es que el usuario pueda visualizar la patología en el cerebro y a la vez pueda detectar otras propiedades y características importantes, como las facciones de la cara del paciente, la visualización del tejido óseo, etc.

La interactividad del sistema permitirá a los usuarios manipular todos los elementos 3D desplegados a través de traslaciones y rotaciones haciendo uso de dispositivos de entrada estándar (teclado y/o ratón). Adicionalmente, los usuarios podrán cambiar, entre los tipos de MEs diseñados, los tamaños y formas de los postes y las longitudes de los tornillos que lo componen.

Aunado a esto, el sistema contará con una serie de paneles y vistas que facilitarán la colocación y el posicionamiento óptimo del ME en el cráneo. Finalmente, el *software* tendrá la capacidad de cargar y desplegar las imágenes DICOM provenientes de los estudios imageneológicos de Resonancia Magnética (RM) del paciente a tratar. Una vez posicionado virtualmente el ME, el *software* generará una salida en formato de imagen DICOM donde se visualice, a través de marcas fiduciales, las guías estereotácticas.

El sistema será diseñado para ser utilizado en ambientes *Microsoft Windows* y hará uso del API de *OpenGL 2.1* para el despliegue de gráficos 2D y 3D.

## 1.4 Metodología

- Se realizarán entrevistas con los médicos especialistas en el área de radiocirugía para estudiar la problemática y definir las variables y requerimientos del sistema. Además, se asistirá a una sesión real de RE para observar el procedimiento de colocación del ME en el cráneo del paciente.
- Recolección de las imágenes médicas de RM que servirán de muestra para la validación del sistema.
- Haciendo uso de un *software* se diseñarán en 3D los distintos tipos de ME, postes, tornillos y cajas radiopacas a escala real que posteriormente se almacenarán en archivos de formato .OBJ.
- Se utilizará el paradigma de Programación Orientada a Objetos (POO). De esta forma todas las funcionalidades del sistema se construirán de manera modular, con una mayor abstracción y facilitará el mantenimiento del código fuente.
- Se utilizará la plataforma *Microsoft .NET* y el lenguaje de programación *C Sharp* para el desarrollo de las funcionalidades del sistema y de las interfaces gráficas del mismo.
- Se utilizará el *framework OpenTK* para la implementación de las funcionalidades de despliegue gráfico 2D y 3D.
- Se estudiarán las distintas bibliotecas que existen para la carga de imágenes DICOM y se seleccionará la que más se adecúe a la solución y a los archivos proporcionados.
- Se realizarán entrevistas con los médicos especialistas en el área de radiocirugía para la evaluación y pruebas del sistema.

# 2

## CAPÍTULO 2

# La Radiocirugía Estereotáctica

---

En este capítulo:

***2.1 Conceptos y fundamentos de la Radiocirugía Estereotáctica***

***2.2 Historia***

***2.3 Técnicas de Radiocirugía Estereotáctica***

***2.4 Personal necesario en la Radiocirugía Estereotáctica***

***2.5 Secuencia del procedimiento***

# Capítulo 2

---

## *La Radiocirugía Estereotáctica*

En el presente capítulo se describen los fundamentos y conceptos principales de la Radiocirugía Estereotáctica. Se aborda su historia y evolución a través de los años. Se presentan las tres técnicas principales para realizar Radiocirugía Estereotáctica, el personal que interviene en el procedimiento y cada una de sus funciones. Finalmente, se detalla la secuencia del procedimiento para hacer Radiocirugía Estereotáctica.

### **2.1 Conceptos y fundamentos de la Radiocirugía Estereotáctica**

La Radiocirugía Estereotáctica (RE), también llamada radioterapia estereotáctica, es un método extremadamente preciso para aplicar altas dosis de radiación ionizante a un volumen de tejido patológico intracraneano concreto y espacialmente definido, de tal forma que se destruya o se altere su funcionamiento.

La RE es considerada una técnica radioterápica. Por una parte utiliza haces de radiaciones que generalmente proceden de equipos comunes a la radioterapia, y por otra, tiene como objetivo conseguir el máximo rendimiento terapéutico posible, entendiendo este como el tratamiento de la lesión evitando complicaciones de los tejidos normales vecinos. Esto último es logrado dirigiendo los haces de radiación de forma selectiva y con precisión sub-milimétrica a las regiones a tratar, evitando irradiar a las estructuras sanas adyacentes.

La RE hace uso de métodos imagenológicos tales como la Tomografía Axial Computarizada (TAC), la Resonancia Magnética (RM) o la Angiografía Digital (AD), para la detección y ubicación previa de la alteración cerebral a tratar.

Según Leksell [2] *“la radiocirugía estereotáctica es una técnica no invasiva para la destrucción de los tejidos o lesiones intracraneales”* que por lo general, por su profunda localización o su cercanía a áreas críticas del cerebro, tienen altas posibilidades de complicaciones o una gran dificultad quirúrgica.

El método se caracteriza por el uso de la estereotáxia, nombre que deriva del griego *stereos*: “tridimensional” y *taxis*: “arreglo metódico”, y que utiliza un sistema de coordenadas tridimensionales independientes del paciente para la localización exacta del tejido patológico en el cerebro (llamado “blanco” o en inglés *“target”*) [3]. Por otra parte, también se caracteriza porque los haces de irradiación son altamente conformados, precisos y convergen sobre la lesión haciendo posible la administración de dosis muy altas de radiación sin incrementar la irradiación de los órganos o estructuras sanas circundantes.

Actualmente la RE se limita a la cabeza y el cuello. Esto se debe a que estas estructuras corporales pueden ser fácilmente inmovilizadas por artefactos de fijación esquelética, restringiendo totalmente su movimiento y permitiendo una alta precisión en el tratamiento [4]. No obstante, existe una variante llamada Radioterapia Estereotáctica Fraccionada (REF) que se administra a intervalos de tiempo constantes y recurrentes, y la cual puede ser aplicada en otras partes del cuerpo con la ayuda de equipos especiales como marcos removibles o máscaras que no requieran fijación esquelética.

Finalmente, como toda forma de radioterapia, la RE no elimina la lesión o el tumor cerebral, pero es capaz de distorsionar el ADN de las células tumorales. De esta manera, las células pierden su habilidad de reproducción y de retención de líquidos. La reducción del tumor ocurre a la misma velocidad que el índice de crecimiento de la propia célula tumoral específica [4].

## 2.2 Historia

Si bien el término radiocirugía fue acuñado por el sueco Lars Leksell en 1951, los principios de la técnica estereotáctica fueron descritos por primera vez en las publicaciones de 1906 y 1908 de Henry Clarke, matemático, y Víctor Horsley, neurofisiólogo y neurocirujano, en donde presentaron los resultados obtenidos mediante el uso de un dispositivo estereotáctico en monos. El propósito de aquel dispositivo era estudiar, desde el exterior, las estructuras profundas del cerebro sin dañar la corteza del mismo. Ellos predijeron que con el uso del dispositivo se "*podría estudiar y registrar cada milímetro cúbico del cerebro*" [5].

Años más tarde, en el año 1949 el neurocirujano del Instituto Karolinska Sjukhuset de Estocolmo, Lars Leksell, diseñó el primer aparato de arco centrado o aparato estereotáctico, basándose en la premisa de que "*era posible dirigir una sonda a cualquier objetivo desde cualquier sitio de entrada sobre la convexidad del cráneo*" [6]. El objetivo del aparato era fijar el cráneo para conseguir inmovilización rígida. En la descripción inicial de este método fue usado un colimador electromagnético de Rayos-X, pero por su gran tamaño, la escasa potencia y precisión espacial; fue reemplazado poco después por cobalto radiactivo como fuente para irradiar lesiones cerebrales tumorales pequeñas y que eran dirigidos estereotácticamente [7]. Aquel cambio fue lo que constituyó la base de la radiocirugía estereotáctica a la que describió como "*una nueva modalidad de tratamiento neuroquirúrgico*" [8], en 1951.

Durante esa misma década los aceleradores lineales fueron desarrollados en países como Inglaterra y Estados Unidos. El mecanismo básico de los mismos consistía en la aceleración de electrones a velocidades muy cercanas a la luz y que en interacción con una aleación metálica generaban Rayos-X que eran redirigidos al área a tratar en el paciente [9].

Para el año de 1961, el mismo Leksell junto a Bjorn Larsson, radiólogo, desarrollaron el bisturí de rayos gamma (*Gamma Knife*). En la **Figura 2.1** se puede observar que el dispositivo consistía de varias fuentes radioactivas de cobalto 60 colocadas en una especie de casco que poseía una serie de canales centrales que convergían en el centro del área a tratar por irradiación y que era exclusivo para radiocirugía [7].

En 1984, Betti y Derechinsky describieron un sistema de radiocirugía utilizando un acelerador lineal (LINAC – *Linear Accelerator*) como fuente de irradiación. Con el tiempo muchas investigaciones subsecuentes modificaron los LINACs, con el fin de asegurar un sistema de radiocirugía adecuado y seguro. Winston y Lutz brindaron avances significativos en la tecnología de los LINACs al incorporar un sistema de posicionamiento estereotáctico [11].

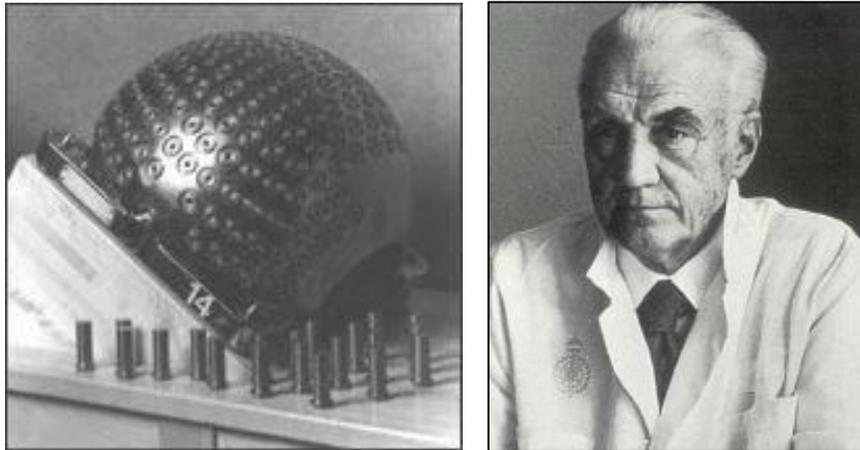


Figura 2.1 El *Gamma Knife* desarrollado por Lars [10]

En la **Figura 2.2** se muestra el sistema radioquirúrgico LINAC. El mismo cuenta con un haz colimado de Rayos-X que es enfocado al paciente mediante la localización estereotáctica del blanco intracraneal. La máquina rota arriba del paciente, produciendo un arco de radiación enfocado a la lesión. La mesa con el paciente es entonces rotada en el plano horizontal y otro arco es formado. De esta manera, se producen múltiples intersecciones de arcos no coplanares de radiación [11].

A finales de esa misma década, para el año de 1987, el profesor de neurocirugía y oncología de la Universidad de Stanford, el Dr. John Adler, desarrolló el sistema de radiocirugía robótica no invasiva llamado *CyberKnife*.

El *CyberKnife* combina dos sistemas: “por un lado utiliza un acelerador lineal compacto y ligero montado sobre un robot que envía radiación hacia las lesiones del paciente, y por otro, un sistema de guía por imágenes radiológicas digitales para seguir la ubicación del tumor en el cuerpo en tiempo real” [12].



Figura 2.2 Imagen del LINAC [13].

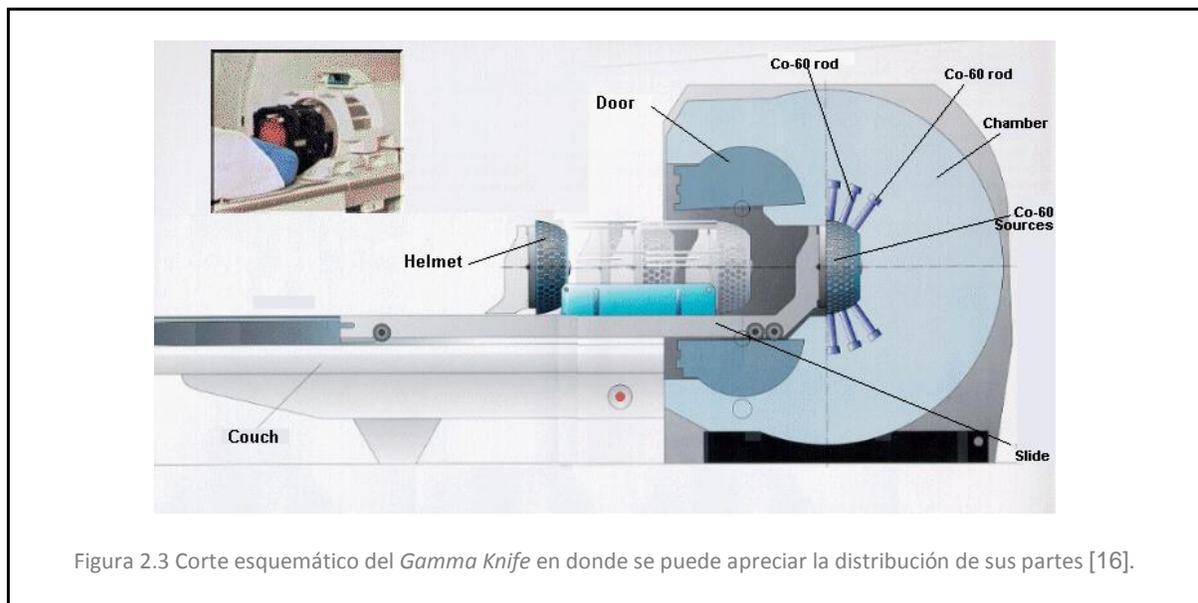
En la actualidad NOVALIS *Shaped Beam Surgery* es el método más avanzado de radiocirugía y radioterapia estereotáctica disponible. NOVALIS permite la configuración dinámica de los haces de irradiación que se adaptan con precisión óptima a las dimensiones y a la forma de la lesión intracraneal en todos los ángulos, asegurando que la misma reciba la dosis completa prescrita y protege a su vez el tejido cerebral sano. La configuración del haz de radiación a la geometría exacta del tumor es el último y más avanzado método para realizar la radiocirugía estereotáctica [14].

## 2.3 Técnicas de Radiocirugía Estereotáctica

Existen tres modalidades básicas de equipo de radiocirugía estereotáctica, cada uno usa distintos instrumentos y fuentes de radiación:

### 2.3.1 El Bisturí de Rayos Gamma (*Gamma Knife*)

El *Gamma Knife* (GK) fue el primer sistema de RE disponible a nivel comercial. Como se aprecia en la **Figura 2.3** las versiones actuales del dispositivo contienen 201 fuentes de radiación de cobalto 60 ( $^{60}\text{Co}$ ) englobadas en un recubrimiento o casco de acero semiesférico. Posee una serie de fuentes de radiación hechas de varias unidades de cobalto integradas en un cilindro. Todo el sistema está acoplado en un recubrimiento, en forma de cámara semiesférica, que tiene una puerta que se abre para admitir al paciente, el cual reposa sobre una mesa deslizante, y una vía de acceso para reemplazar las fuentes de  $^{60}\text{Co}$  y realizar mantenimiento al equipo [15].



Las fuentes de radiación están alineadas de tal manera que sus rayos hagan intersección en un único punto que se denomina isocentro. Como se ve en la **Figura 2.4**, una vez que el paciente se encuentra bajo anestesia local, su cabeza es acoplada a un marco rígido estereotáctico haciendo uso de cuatro tornillos, dos

en la parte frontal y dos en la parte posterior de la cabeza. Dicho marco tiene incorporado un sistema de coordenadas tridimensionales y un isocentro el cual se intenta alinear con el centro de la lesión a tratar [17].



Figura 2.4 Colocación del marco estereotáctico en el paciente y ubicación de los cuatro tornillos [18].

Posteriormente, se obtienen estudios imagenológicos tales como TAC, RM o AD y sus resultados se envían al sistema computarizado de planificación del GK. Juntos, los médicos y radiofísicos médicos delimitan las zonas a tratar y las estructuras anatómicas normales y usan la computadora de planificación para determinar la relación exacta entre éstas y el marco de cabeza, y calculan los parámetros para el tratamiento con el bisturí de rayos gamma [15].

El marco estereotáctico se encuentra rígidamente fijo a la mesa de tratamiento como se puede apreciar en la **Figura 2.5**, de tal manera que cuando dicha mesa avanza dentro del GK, la localización de la lesión dentro de la cabeza coincide con el isocentro del sistema y los rayos gamma convergen a la misma.

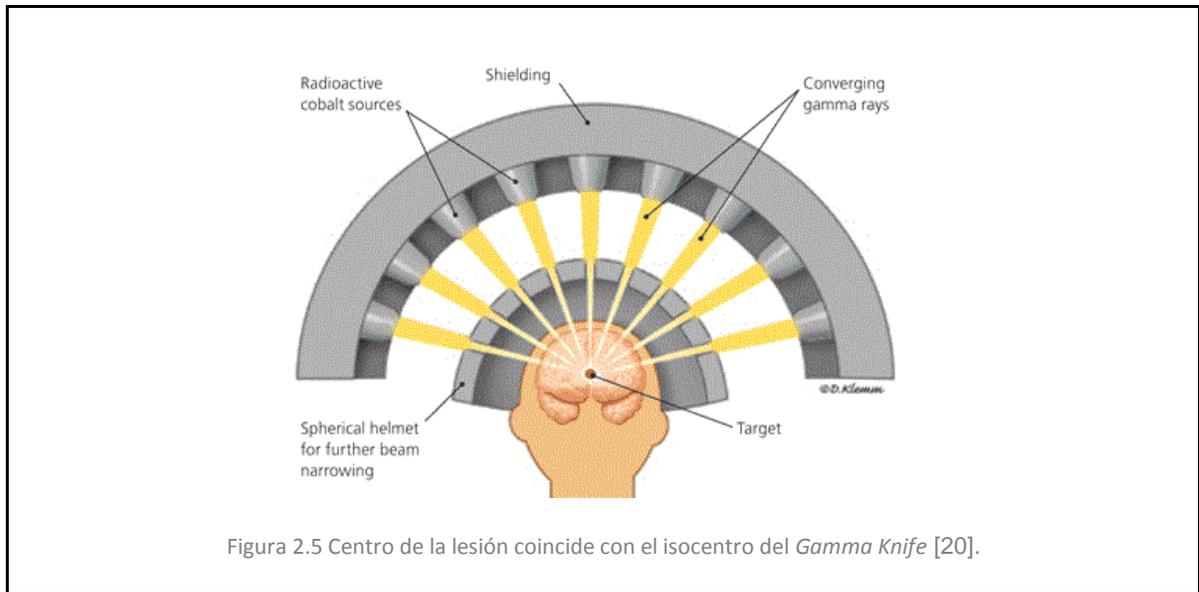


Figura 2.5 Centro de la lesión coincide con el isocentro del *Gamma Knife* [20].

La precisión del GK, en un rango menor de dos décimos de un milímetro, hace que esta tecnología sea la preferida para tratar ciertas enfermedades del cerebro. Especialmente aquellas en las que el tejido cerebral circundante es crítico y crucial para no ocasionar ningún tipo de efecto secundario en el paciente [19].

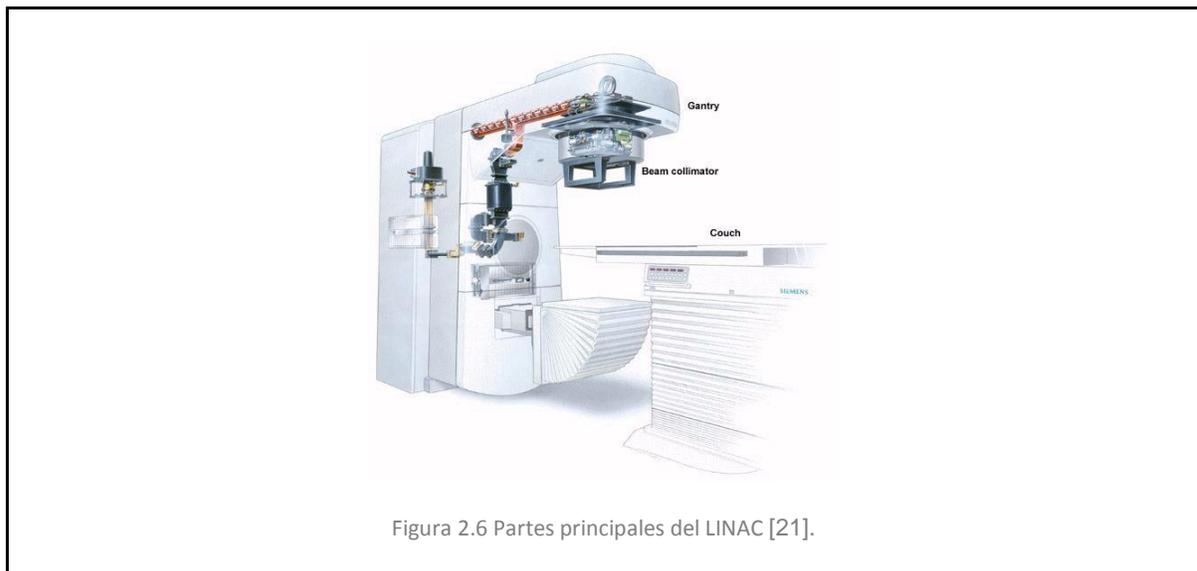
Es importante señalar que el procedimiento radioquirúrgico a través del GK no requiere de múltiples visitas al hospital y el paciente no incurre en gastos por sala de operaciones o cuidados intensivos postoperatorios. Generalmente, este procedimiento se considera la alternativa más económica y es totalmente ambulatorio.

### 2.3.2 Aceleradores Lineales (LINAC)

Los aceleradores lineales (LINAC) son dispositivos usados comúnmente para dar radioterapia de haz externo a enfermos con cáncer. Los resultados al hacer radiocirugía con este sistema son muy similares a los obtenidos con el uso del GK en áreas objetivas del cerebro. La característica principal del LINAC es que suministra una dosis uniforme de radiación a la región del tumor del paciente sin afectar los tejidos circundantes normales.

Como se mencionó anteriormente, los LINAC son sistemas que permiten producir radiación. Esa radiación es generada por un sistema electromagnético en el que se aceleran electrones, que tras el adecuado proceso de frenado pueden emitir dos tipos de radiación: fotones y electrones [15]. Generalmente en RE se usan fotones, con una energía de 6 mv. La energía del haz de radiación producida por un procedimiento electromagnético es constante, en otro sentido, no sufre reducciones con el paso del tiempo. Es importante señalar que en el LINAC no se almacena ningún tipo de producto radiactivo.

Como puede apreciarse en la **Figura 2.6**, el sistema está compuesto por una mesa o camilla móvil donde reposa el paciente, un brazo mecánico llamado “gantry” y un colimador de Rayos-X. Al igual que en el procedimiento anterior, al paciente se le coloca un marco estereotáctico para fijar y restringir el movimiento de la cabeza durante la radiocirugía.



La energía irradiada se produce al acelerar electrones cerca de la velocidad de la luz, luego estos colisionan con un metal pesado colocado en la cabeza del “gantry”. La colisión produce calor, pero un pequeño porcentaje de esa energía es convertida en fotones, también llamados Rayos-X ya que son producidos electrónicamente [22]. La energía producida es entonces enfocada esterotácticamente al isocentro del marco haciendo uso del colimador, entonces el “gantry” del LINAC rota alrededor del paciente produciendo un arco de radiación enfocado al blanco como se aprecia en la **Figura 2.7**. La camilla del paciente también es rotada en el plano horizontal y otro arco es formado. De esta manera, múltiples arcos no coplanarios se intersectan en el volumen objetivo irradiándolo con la dosis del tratamiento.

En cuanto a la planificación se puede hacer no sólo por sumación de esferas de tratamiento, sino también con infinitas combinaciones de los arcos de giro, longitud del arco, dosis de cada arco, combinación de colimadores, etc. La planificación para el LINAC es muy versátil y fácil de adaptar a cualquier volumen [15].

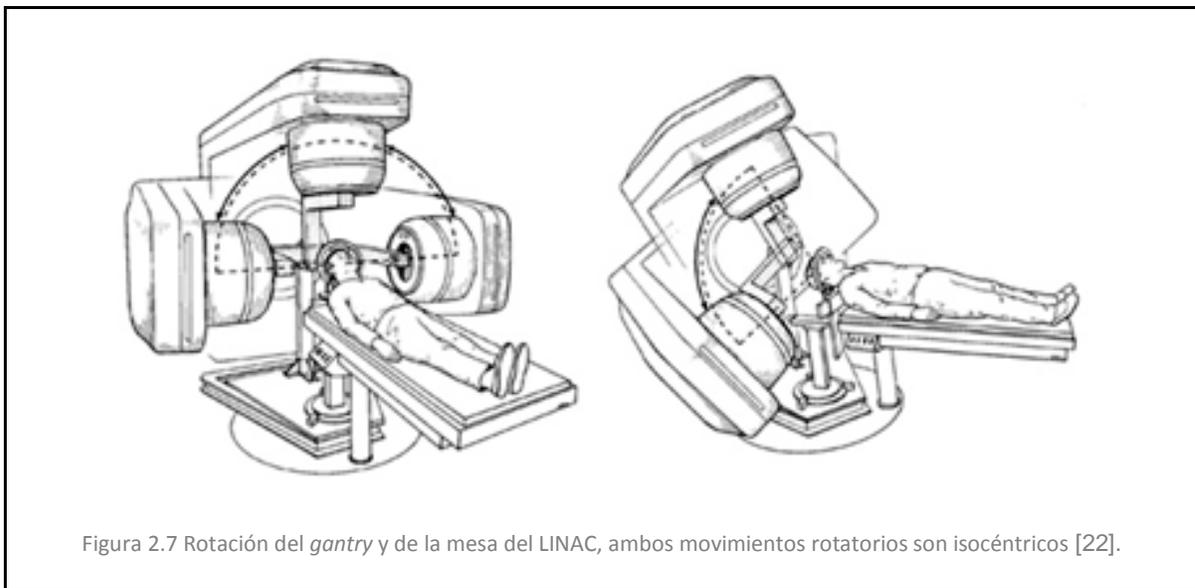


Figura 2.7 Rotación del gantry y de la mesa del LINAC, ambos movimientos rotatorios son isocéntricos [22].

En la actualidad existen ya algunos sistemas de posicionamiento estereotáctico sin guía para el tratamiento usando LINACs. Estos sistemas utilizan determinadas marcas que, localizadas por cámaras de infrarrojos, permiten repetir la misma posición del paciente. Sin embargo, estos sistemas aún están en fase de ensayo [15].

### 2.3.3 Radiocirugía usando Protones

Los protones son partículas que poseen una carga positiva. Similar a como los Rayos-X se utilizan para tratar patologías cerebrales o células tumorales, un haz de protones puede ser utilizado para irradiar lesiones de la misma forma. En general, no existen diferencias entre los efectos biológicos de los protones comparados con los Rayos-X [17]. Físicamente los protones liberan la mayor parte de su energía cuando impactan en el tumor y, a diferencia de los fotones, no aplican una dosis fuera de los límites del tumor. Por lo

tanto, la dosis de radiación proporcionada por este tipo de procedimientos se ajusta mejor al tumor y se produce menos daño al tejido sano.

La terapia con uso de protones utiliza una máquina especial llamada ciclotrón (o sincrotrón) para generar y acelerar los protones. Los protones abandonan la máquina y son dirigidos electromagnéticamente hacia el tumor. Se utilizan otros equipos para modificar el rango de los protones y la forma del haz [17].

Un ciclotrón combina un campo magnético axial producido por los imanes y un campo eléctrico radial de alta frecuencia, que se alterna entre dos estructuras semicirculares llamadas “*dees*”. Los haces son obtenidos ionizando átomos de hidrógeno y acelerando los protones en el campo eléctrico producido entre los dos “*dees*”. Una vez que se alcance la energía deseada, los protones se modulan y se emiten como rayos. Este procedimiento requiere de la colocación del marco estereotáctico en la cabeza para inmovilizarla y hacer que el paciente rote alrededor del punto exacto del blanco [23].

Son aparatos muy costosos y complejos en su manejo, por lo que solamente existen muy pocos equipos en el mundo.

## 2.4 Personal necesario en la Radiocirugía Estereotáctica

El equipo de tratamiento está compuesto por varios profesionales médicos especializados. Típicamente el grupo está conformado por un radioncólogo, un neurocirujano, un neurorradiólogo, un radiofísico médico, un dosimetrista, un radioterapeuta, personal de enfermería de radioterapia, y un neurólogo o neuroncólogo. Indiferentemente de la técnica de radiación utilizada las funciones básicas de cada uno de los especialistas son:

- **El radioncólogo y/o el neurocirujano:** son los que encabezan el equipo de tratamiento y vigilan el tratamiento en su totalidad; bosquejan el área objetiva a tratarse, deciden la dosis apropiada de radiación, aprueban el plan de tratamiento, e interpretan los resultados de los procedimientos radioquirúrgicos.  
En muchos casos el neurocirujano y/o el radioncólogo pueden ayudar a posicionar al paciente en la mesa de tratamiento. Ambos pueden observar y comunicarse con el paciente a través de un sistema de circuito cerrado durante todo el procedimiento.
- **El neurorradiólogo:** es el encargado de la interpretación de las imágenes del cerebro que identifican el(las) área(s) objetiva(s) a tratarse en la RE.
- **El físico médico de radiación:** es el que asegura la entrega de la dosis precisa de radiación. Trabaja en conjunto con los radioncólogos y/o neurocirujanos durante el proceso de planificación del tratamiento. Usa el *software* de planificación para preparar el plan de tratamiento; calcula las exposiciones y configuración del haz para tratar el área conforme con la dosis prescrita.
- **La enfermera de radioterapia:** proporciona al paciente información sobre el tratamiento, lo vigila durante el procedimiento, y ayuda a contestar preguntas después del mismo.
- **El neurólogo o neuroncólogo:** puede participar con el radioncólogo y neurocirujano en el proceso de planificación de la RE; ayuda a decidir en qué casos el procedimiento puede aplicarse y sus posibles beneficios.

## 2.5 Secuencia del procedimiento

Independientemente del tipo de radiación utilizada, la metodología radioquirúrgica es muy similar. El procedimiento consta de cuatro fases: establecimiento del sistema de coordenadas, adquisición de las imágenes diagnósticas para la planificación dosimétrica, planificación computarizada de la dosis, y administración de la radiación.

### 2.5.1 Establecimiento del sistema de coordenadas

La localización espacial de la lesión y de las estructuras cerebrales se consigue ajustando de forma rígida e inamovible durante todo el proceso, un marco o anillo estereotáctico. Como se ve en la **Figura 2.8**, el marco estereotáctico está hecho de metal, tiene un peso aproximado de dos kilogramos y posee un isocentro el cual se trata de alinear, en lo posible, al volumen a tratar. Luego de que el paciente es anestesiado localmente, el marco es fijado a su cabeza a través de cuatro tonillos o pernos atornillados en la tabla externa de la calota; dos en la parte frontal y dos en la parte occipital del cráneo.



Figura 2.8 Marco estereotáctico y colocación del mismo en el paciente haciendo uso de los cuatro tornillos [24] [25].

Seguidamente, acoplado sobre el marco se colocan unos localizadores, generalmente una caja como la que se puede observar en la **Figura 2.9**, que llevan marcas radiopacas y que aparecerán después en los estudios radiológicos. Fundamentalmente, estas sirven para localizar la profundidad del blanco quirúrgico, es decir, definen la posición exacta de la lesión en el espacio. La colocación del marco estereotáctico es realizada por el neurocirujano encargado del caso.

Como se mencionó anteriormente el marco en sí tiene dos funciones importantes. Primeramente, constituye en sí mismo el sistema de coordenadas estereotácticas, en otras palabras, se utiliza para colocar un sistema de puntos de referencia que permiten una localización exacta de la lesión en los tres planos cartesianos. Dicha localización es a la que se referirán los estudios de imagen y es relevante para el posicionamiento del paciente en el tratamiento. En segundo lugar, este marco constituye la base del procedimiento de sujeción e inmovilización durante del tratamiento.

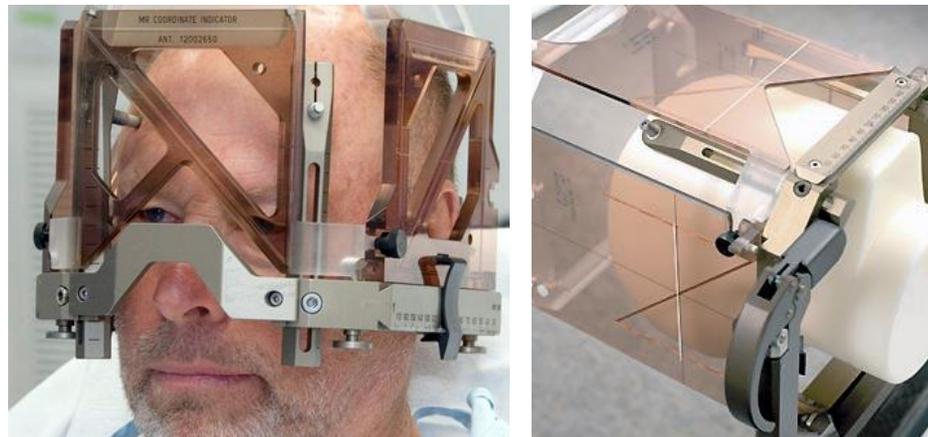


Figura 2.9 Caja con marcas radiopacas para localizar el blanco radioquirúrgico [26].

### 2.5.2 Adquisición de las imágenes diagnósticas para la planificación dosimétrica

La delimitación del volumen a tratar y de los órganos críticos o de riesgo a preservar, se realiza mediante el uso de imágenes diagnósticas de diferentes modalidades. La modalidad fundamental es la TAC, cuyos datos proporcionan la base del cálculo dosimétrico.

Otras técnicas diagnósticas como la AD, la RM, etc.; funcionan como guías o ayudas para la mejor definición y discriminación de las lesiones y los órganos críticos. Durante este procedimiento se puede conocer con exactitud el tamaño, la posición, la forma de la lesión y la relación de ésta con otras estructuras cerebrales. Es importante destacar que el estudio imagenológico depende directamente del tipo de marco estereotáctico utilizado y de la compatibilidad de este con los dispositivos de captura de imágenes.

La correlación entre unas imágenes y otras es realizada mediante técnicas de fusión (en el caso de imágenes tomográficas), o de simple correlación mediante marcas fiduciales, como en el caso de los estudios de angiografía por Rayos-X [27]. Durante la exploración radiológica se marca la posición del tejido patológico al que se quiere acceder en el espacio y en conjunto con los marcadores situados en la guía se calculan las coordenadas de este punto en los tres ejes cartesianos X, Y y Z.

### 2.5.3 Planificación del tratamiento

En esta fase se hace uso de una computadora que dispone de un *software* de planificación que varía según el tipo y modelo del equipo de radiocirugía utilizado. En la **Figura 2.10** se puede ver un ejemplo de la interfaz de un *software* de planificación de RE, en la misma se puede apreciar el contorno de la lesión así como el posible recorrido de los haces a través del tejido.

En el planificador se contornean la(s) lesión(es) y los órganos y/o estructuras sanas adyacentes que son radiosensibles o críticas. Esto permite reconstruir al paciente en tres dimensiones y diseñar de forma virtual el plan de tratamiento más adecuado para el mismo.

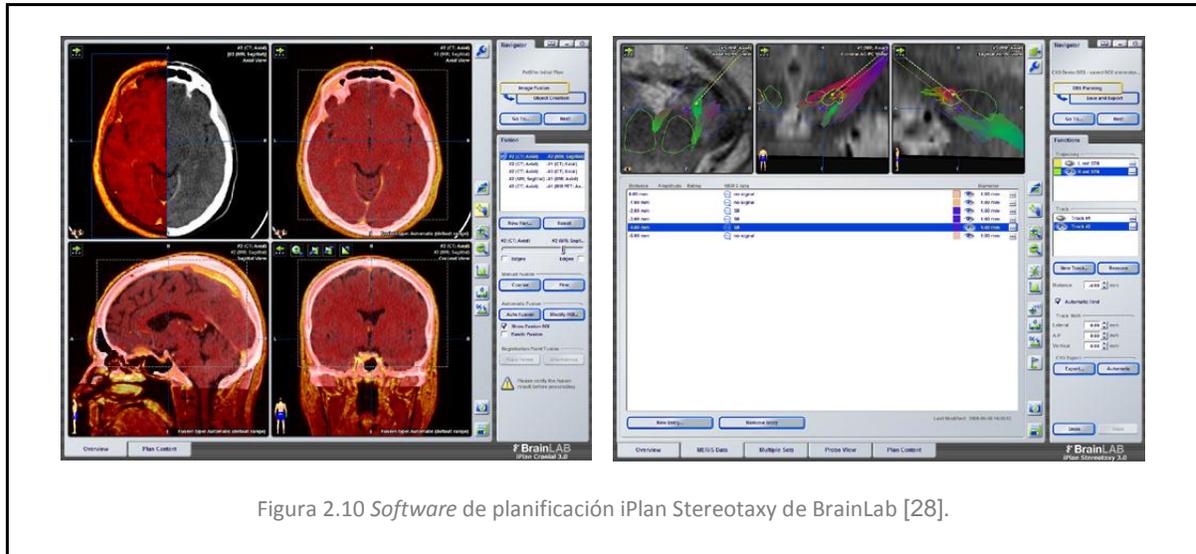


Figura 2.10 Software de planificación iPlan Stereotaxy de BrainLab [28].

El objetivo de la planificación del tratamiento es calcular las distribuciones de dosis para el volumen tridimensional que representa la lesión, y que fue definido por las imágenes diagnósticas procedentes de los estudios descritos en el apartado anterior. El *software* de planificación hace uso de algoritmos complejos que simulan la interacción de la radiación con la materia definida por los métodos imagenológicos, tanto geoméricamente como estructuralmente (tomando en cuenta las diferentes densidades y composiciones atómicas de los tejidos) [27].

De acuerdo con las características de la lesión, el equipo de trabajo establece las condiciones del tratamiento o prescripción dosimétrico-volumétrica, donde se especifican los niveles de las dosis que deben llegar a los volúmenes a irradiar y los correspondientes a los órganos o regiones críticas [27].

Finalmente, la técnica de tratamiento es el resultado del cálculo realizado en computadoras potentes que trabajan con las imágenes y los parámetros de entrada y obtienen los datos necesarios para el cálculo dosimétrico. Es importante destacar que los resultados vienen referidos al marco estereotáctico utilizado, de forma que puedan ser trasladados y ejecutados en el tratamiento propiamente dicho.

## 2.5.4 Radiocirugía: tratamiento

Antes de iniciar el tratamiento debe comprobarse la correcta disposición de ciertos componentes mecánicos y de los haces de irradiación del equipo radioquirúrgico a utilizar. El control de calidad comprende la verificación de las condiciones geométricas, mecánicas y dosimétricas que deben satisfacer el acelerador o equipo generador utilizado.

Una vez que es definida la técnica del tratamiento en la planificación, los datos y parámetros son transferidos a la unidad de tratamiento, con el objeto de ser cargadas y revisadas en la misma. Una vez con-

cluida la fase de verificación del procedimiento, el paciente es colocado en la mesa de la unidad de tratamiento, y se procede a ajustar y fijar su posición respecto al sistema de coordenadas de la máquina. En este paso, las coordenadas numéricas estereotácticas del centro de la lesión, calculadas por el planificador y referenciadas en los tres planos del espacio, son localizadas en una escala calibrada en décimas de milímetro sobre el marco estereotáctico. Estas referencias se hacen coincidir con los haces que apuntan al isocentro de la máquina [29].

El tiempo invertido desde la inserción o diseño del marco estereotáctico hasta la finalización del tratamiento suele oscilar entre 3 y 6 horas dependiendo de la complejidad del proceso. El tiempo real de utilización del dispositivo de radioterapia es de 30 a 90 minutos. El paciente puede ser tratado de forma ambulatoria [29].

# 3

CAPÍTULO 3

## Colocación del Marco Estereotáctico

---

En este capítulo:

***3.1 Sobre la colocación del Marco Estereotáctico***

***3.2 Trabajos en colocación asistida del Marco Estereotáctico***

# Capítulo 3

---

## *Colocación del Marco Estereotáctico*

En el presente capítulo se describen los posibles problemas asociados a la colocación del Marco Estereotáctico (ME) sobre el paciente y sus consecuencias. Adicionalmente, se presentan una serie de trabajos y publicaciones previas que ayudan a solventar los posibles inconvenientes de una mala colocación de marco haciendo uso de dispositivos mecánicos y/o herramientas de *software*.

### **3.1 Sobre la colocación del Marco Estereotáctico**

La Radiocirugía Estereotáctica (RE) es una modalidad radioquirúrgica para aplicar altas dosis de radiación a un volumen de tejido patológico intracraneano concreto y espacialmente definido. Su planificación preoperatoria requiere establecer una correspondencia espacial entre la lesión intracraneal del paciente, contenida en imágenes médicas, y un sistema de coordenadas referencial proporcionado por un Marco Estereotáctico (ME). De esa manera, se puede determinar un conjunto de coordenadas estereotácticas que ubican con precisión sub-milimétrica las lesiones, para las que posteriormente se diseña un plan de tratamiento quirúrgico apropiado.

Dentro del ámbito de la RE, el ME es utilizado para obtener una guía tridimensional que ayuda a localizar objetivos importantes definidos y con precisión en el cerebro [30]. El procedimiento que se sigue en la actualidad en el montaje del marco sobre el cráneo y su excentricidad con respecto a la lesión, se basa en la experticia y la percepción visual del médico que lo realiza [30]. Por consiguiente, el procedimiento es totalmente subjetivo, cualitativo y propenso a errores.

Como consecuencia de una mala colocación del ME pueden ocurrir colisiones entre el marco y los colimadores de la máquina de radioterapia durante la RE o que el tratamiento no pueda ser realizado y deba ser postergado.

De cualquier modo, si el volumen de interés no puede ser tratado con la colocación inicial del marco entonces se necesitarán nuevas colocaciones del mismo o la re-planificación de un nuevo curso para el tratamiento. Todo ello se resume en: incomodidad para el paciente; mayor esfuerzo, uso de recursos y tiempo en el procedimiento.

## 3.2 Trabajos en colocación asistida del Marco Estereotáctico

Como respuesta a los posibles inconvenientes en el posicionamiento del marco descritos en la sección anterior, se han diseñado una serie de aplicaciones y herramientas cuyo objetivo principal es permitir que la colocación del ME se realice con la mayor precisión posible, evitando cualquier indicio de subjetividad en el procedimiento.

En el 2005, Lay *et al.* [31] señalan el desarrollo de un “dispositivo de asistencia para el posicionamiento” y un sistema que optimiza la colocación del ME en las radiocirugías con GK antes del tratamiento. El propósito era evitar colisiones del marco con los colimadores de la máquina durante la RE.

Lay *et al.* [31] describen un procedimiento de dos pasos. Primeramente, un sistema predice las coordenadas óptimas para la colocación del marco haciendo uso de imágenes médicas, como RMs o TACs, obtenidas en la etapa de pre-tratamiento. Seguidamente, se usa un dispositivo para asistir en la colocación del ME utilizando las coordenadas obtenidas por el sistema. Para ello se hace uso de marcadores fiduciales en las imágenes médicas.

El dispositivo fue construido utilizando plástico de alto grado de curvatura para dar cabida a un collarín cervical en su superficie interior y un bloque de acero y un sistema de barras en su superficie externa. El sistema de barras se compone de una barra horizontal (eje Y) y una barra vertical (eje Z) de 90 grados de curvatura en la parte superior (eje X). Las dos barras se unen a través de un bloque de acero con dos orificios oblicuos. Cada orificio está grabado con un tornillo que permite el ajuste a lo largo del eje respectivo. La porción del eje X tiene un bloque de acero que permite el acoplamiento del dispositivo al ME. Según indica Lay *et al.*, el sistema de barras permite la traslación y rotación de cada eje con seis grados de libertad, lo que asegura que la colocación del ME sea precisa.

Finalmente, el sistema permite a los neurocirujanos la habilidad de aplicar el ME a pacientes de manera más precisa y reproducible. Indican los autores que el sistema realmente reduce el tiempo de tratamiento y el estrés que pueda presentar el paciente durante el procedimiento.

Un año más tarde, en el 2006, Nieto *et al.* [32] describen un sistema que “registra y ayuda a corregir la variación lineal horizontal y vertical generada por la posición del marco (ME) con respecto al cráneo, de manera que con la lectura y la corrección de tales variaciones se obtiene la excentricidad deseada”.

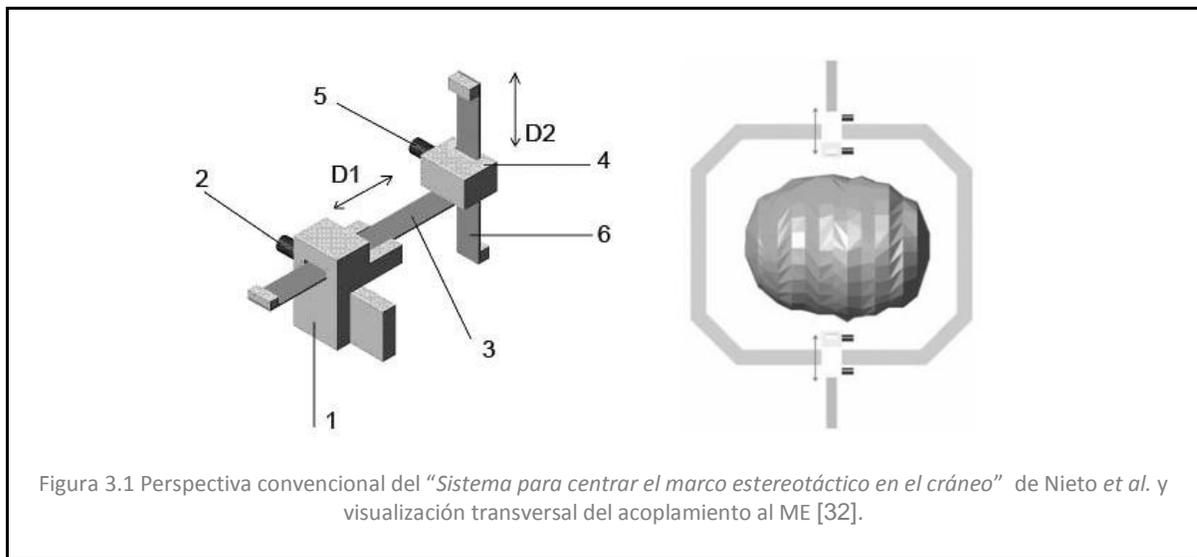
El objetivo del trabajo era proporcionar “un sistema que fuera altamente confiable, cuya principal característica sea el registro y corrección de desplazamientos horizontales y verticales”, así como su fácil acoplamiento al ME. Según describe Nieto *et al.* [32], las piezas fueron fabricadas en PVC (*PolyVinyl Chloride* - PoliCloruro de Vinilo) espumado. Como puede apreciarse en la **Figura 3.1** el sistema está formado por dos contenedores. El primero de ellos (1) está constituido por un perno (2) y una lámina de aluminio graduada en milímetros (3). Dicho perno, junto a la lámina de aluminio, desarrollan movimientos bidireccionales. Cuando se hace girar el perno se genera un desplazamiento (D1) sobre la lámina horizontal con la que se registran y corrigen las variaciones lineales horizontales del marco con respecto al cráneo en el plano transversal.

Por otra parte, el segundo contenedor (6) se encuentra fijado en el extremo de la lámina horizontal. Dicho contenedor, al igual que el primero, está formado por un perno y una lámina graduada en milímetros. El perno y la lámina graduada (4) desarrollan movimientos bidireccionales también. Cuando se hace girar el

perno (5) se generan desplazamientos (D2) sobre la lámina vertical con la cual se registran y corrigen las variaciones lineales verticales del marco con respecto al cráneo en el plano frontal.

Nieto *et al.* detallan el funcionamiento del sistema. Primeramente, dos sistemas son acoplados lateralmente sobre el ME, como se ilustra en la **Figura 3.1**. Las piezas van colocadas una a cada lado del marco. Las escalas laterales graduadas del ME sirven para que los sistemas estén equidistantes entre sí durante el montaje. Seguidamente, en cada sistema se van registrando variaciones lineales y verticales, y se va ajustando la posición del ME sobre el cráneo de forma manual y visual hasta que se logre la ubicación deseada en cada plano. Finalmente, una vez que se logra la posición requerida del marco, se fija al cráneo y los sistemas se desmontan para continuar con el procedimiento clínico normalmente.

Como resultado, Nieto *et al.* obtienen un sistema confiable cuyo propósito es centrar el ME en el cráneo durante su montaje. Su funcionamiento se basa en registrar y corregir desplazamientos verticales y horizontales, los cuales contienen elementos de medición y corrección de variaciones lineales en los planos X e Y.

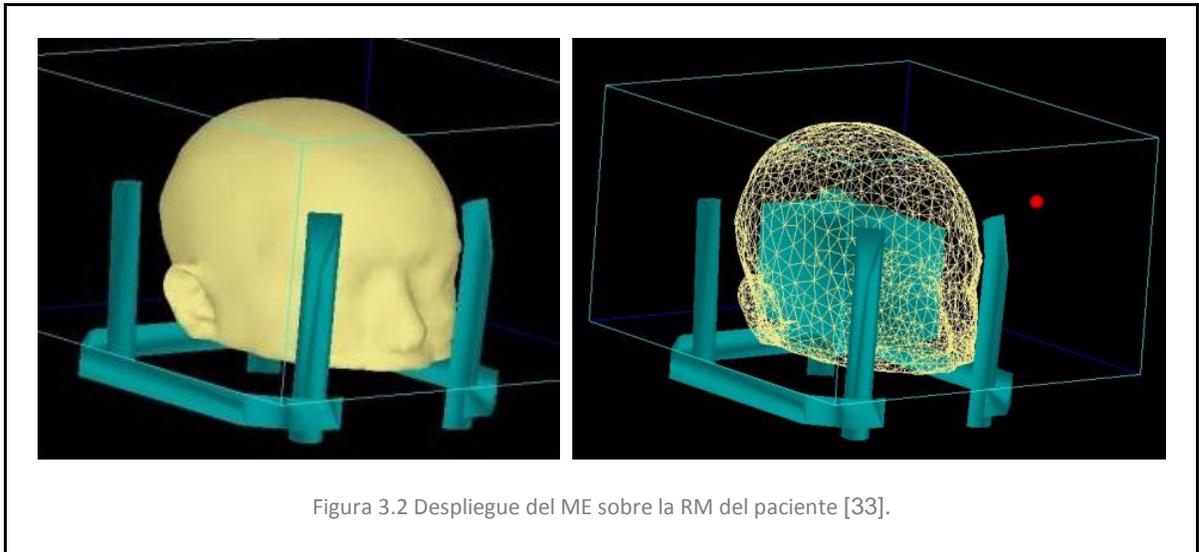


Para el año 2007, Tryggestad *et al.* [33] describen una herramienta virtual para el posicionamiento del marco en radiocirugías con GK. Dicha herramienta provee al neurocirujano una guía visual y cuantitativa para la colocación óptima de ME en el cráneo.

El sistema fue implementado bajo el *framework* de "Pinnacle<sup>3</sup> Treatment Planning System" [34]. Dicho *framework*, provee la capacidad de cargar y manipular mallados de objetos y superficies triangulares creadas externamente bajo el formato VTK y que hagan uso del sistema de coordenadas interno del sistema de planificación.

Tryggestad *et al.* [33] señalan que se diseñaron en 3D varias representaciones del ME Leksell, incluyendo los cuatro postes, tornillos y la caja con las marcas radiopacas. Además, se desarrollaron varios *scripts* en C++ para ayudar a los usuarios en el posicionamiento óptimo del marco.

Los modelos en 3D son cargados en el sistema y son colocados sobre la RM del paciente, como se ilustra en la **Figura 3.2**. Luego, el usuario puede manipular los componentes en 3D hasta lograr la localización óptima del ME.



Como resultados de la investigación, el *software* provee la capacidad de: 1) colocar interactivamente la representación 3D del ME sobre el estudio de RM del paciente para detectar posibles colisiones con el GK y 2) colocar marcas fiduciales sobre una copia de la RM del paciente basadas en la colocación actual del marco, como puede apreciarse en la **Figura 3.3**. Esta última característica puede ser útil para generar un plan de tratamiento inicial estimado, basado exclusivamente en la colocación del ME.

En el año 2009, Wiant *et al.* [35] trataron específicamente el problema de las colisiones entre el ME y los colimadores del GK. El objetivo del trabajo era detectar posibles colisiones entre el ME y los colimadores del casco del GK (para los modelos 4C del GK y versiones anteriores). Esta limitante de tratamiento puede ser detectada únicamente después que el marco ha sido fijado al cráneo del paciente y es colocado en la camilla de la máquina.



Como solución al problema, Wiant *et al.* [35] desarrollaron un *software* que permite la colocación simulada del ME y la detección de colisiones haciendo uso de imágenes médicas del paciente tomadas días antes del pre-tratamiento. La finalidad era minimizar la necesidad de múltiples colocaciones del marco y la incomodidad del paciente.

Las validaciones y pruebas para determinar las capacidades que tenía el sistema en la predicción de una posición clínicamente adecuada del ME se hicieron sobre una cabeza fantasma antropomórfica (*Anthropomorphic head phantom*), similar a las que se pueden apreciar en la **Figura 3.4**.



Figura 3.4 Imagen de una cabeza fantasma antropomórfica (*Anthropomorphic head phantom*) [36].

Por otra parte, también se usó el *software* en veinticinco imágenes médicas de pacientes que no pudieron ser tratados con la colocación inicial de marco. El objetivo era determinar si la herramienta era capaz de localizar una posición optimizada del ME, de tal forma que se hiciera radiocirugía a todos los blancos identificados sin necesidad de remover el marco para una nueva colocación durante el procedimiento. Los resultados obtenidos fueron que cuatro de los casos pudieron ser completados con la colocación inicial y optimizada del marco, mientras que el resto no habría podido ser tratado con una sola colocación del mismo.

# 4

CAPÍTULO 4

# Diseño e Implementación

---

En este capítulo:

***4.1 Aspectos Técnicos***

***4.2 Arquitectura de la aplicación Virtual Stereotactic Frame***

***4.3 La biblioteca dinámica DicomLib***

***4.4 La biblioteca dinámica ModelLib***

***4.5 La biblioteca dinámica UtilityLib***

***4.6 La aplicación Virtual Stereotactic Frame***

# Capítulo 4

---

## *Diseño e Implementación*

Para cumplir con la propuesta realizada se desarrollaron una serie de bibliotecas dinámicas (*DicomLib*, *ModelLib* y *UtilityLib*) en el lenguaje de programación *C-Sharp (C#)*, las cuales contienen los métodos y clases necesarias para el funcionamiento de la aplicación *Virtual Stereotactic Frame*. En la biblioteca dinámica *DicomLib* se encuentran los métodos necesarios para la carga de archivos en formato DICOM, el despliegue de volúmenes y el procesamiento de los mismos entre otras funcionalidades. Por otro lado, en la biblioteca dinámica *ModelLib* se encuentran los métodos y configuraciones relacionadas con el despliegue de los modelos que simulan el Marco Estereotáctico (ME) y sus componentes. Por último, en la biblioteca dinámica *UtilityLib* se encuentran las clases adicionales y *helpers* que dan soporte a la aplicación final.

### 4.1 Aspectos Técnicos

Para la implementación de la aplicación y bibliotecas dinámicas se utilizó el entorno de desarrollo *Microsoft Visual Studio 2010* bajo el marco de trabajo *Microsoft .NET Framework 3.5* y *4.0*. El sistema fue desarrollado para funcionar únicamente en ambientes *Microsoft Windows*<sup>1</sup>.

Como lenguajes de programación se utilizaron: *C-Sharp (C#)* para la programación de las bibliotecas dinámicas e interfaces gráficas, y *GLSL* para la programación en el procesador gráfico o GPU. El API gráfico utilizado fue *OpenGL v2.1*, encapsulado en el *toolkit* de *OpenTK v1.0.2*, el cual soporta la programación de los procesadores de vértices y fragmentos mediante el lenguaje *GLSL*, y está disponible en múltiples plataformas.

Para el desarrollo de las interfaces gráficas de usuario se utilizó el API de *Windows Forms (WinForms)* que proporciona *Microsoft .NET Framework*. Dicho API facilita el diseño e implementación de ventanas, el manejo de eventos, la creación y uso de componentes (como botones, menús, barras de herramientas, cajas de texto, etc.) y la interacción con el ratón y el teclado. En el desarrollo del sistema se utilizó el paradigma de Programación Orientada a Objetos haciendo uso de ciertas convenciones en la nominación de las clases, nombres de bibliotecas dinámicas, estructuras, atributos, métodos y constantes. Los nombres de clases, bibliotecas dinámicas, métodos, constantes y atributos públicos siempre comienzan con la primera

---

<sup>1</sup> Los sistemas operativos soportados son: *Windows 7*, *Windows 7 con Service Pack 1*, *Windows Server 2003 con Service Pack 2*, *Windows Server 2008*, *Windows Server 2008 R2*, *Windows Server 2008 R2 SP1*, *Windows Vista con Service Pack 1* y *Windows XP con Service Pack*.

letra en mayúscula. La nominación de los atributos privados contiene el prefijo “\_” con la primera letra en minúscula. Los nombres de las variables locales y parámetros comienzan en minúsculas sin prefijo alguno. Los nombres de las bibliotecas dinámicas contienen el sufijo “Lib”. En todos los casos si los nombres están formados por dos o más palabras se utiliza la notación *UpperCamelCase* (Mayúsculas/Minúscula Camello). Por ejemplo: *DicomLib*, *Initialize()*, *GenerateCuts()*, *DicomAxialList*, *\_isActive*, *result*, etc.

## 4.2 Arquitectura de la aplicación *Virtual Stereotactic Frame*

El sistema desarrollado consta de un total de seis módulos principales: cinco bibliotecas dinámicas y una aplicación con interfaz gráfica de usuario llamada *Virtual Stereotactic Frame* (VSF). El objetivo de dividir la aplicación en módulos o subprogramas es hacerlo más legible, manejable y de fácil mantenimiento. Cada uno de los módulos del sistema representa una estructura lógica con tareas y opciones específicas. Dichos módulos se pueden comunicar entre ellos a través de interfaces bien definidas, tales como clases, métodos y atributos. La aplicación VSF es la que interactúa directamente con el usuario a través de comandos y acciones predefinidas haciendo uso de la entrada tradicional de teclado y ratón. Adicionalmente, la aplicación VSF se comunica directamente con las bibliotecas dinámicas *UtilityLib*, *ModelLib* y *DicomLib* las cuales proporcionan soporte para la interacción con el usuario y permiten el despliegue volumétrico y de geometrías 2D y 3D.

La biblioteca dinámica *DicomLib* hace uso de varias estructuras y métodos definidos en la biblioteca *UtilityLib* para el despliegue volumétrico y otras funciones que se explicarán más adelante. Adicionalmente, se hace uso de dos bibliotecas dinámicas llamadas *ClearCanvas* y *Opendicom-sharp*, las cuales se comunican directamente con la biblioteca dinámica *DicomLib* para cargar, procesar y salvar imágenes DICOM. En la **Figura 4.1** se puede apreciar un diagrama general del sistema desarrollado y la interacción existente entre las bibliotecas dinámicas y la aplicación VFS.

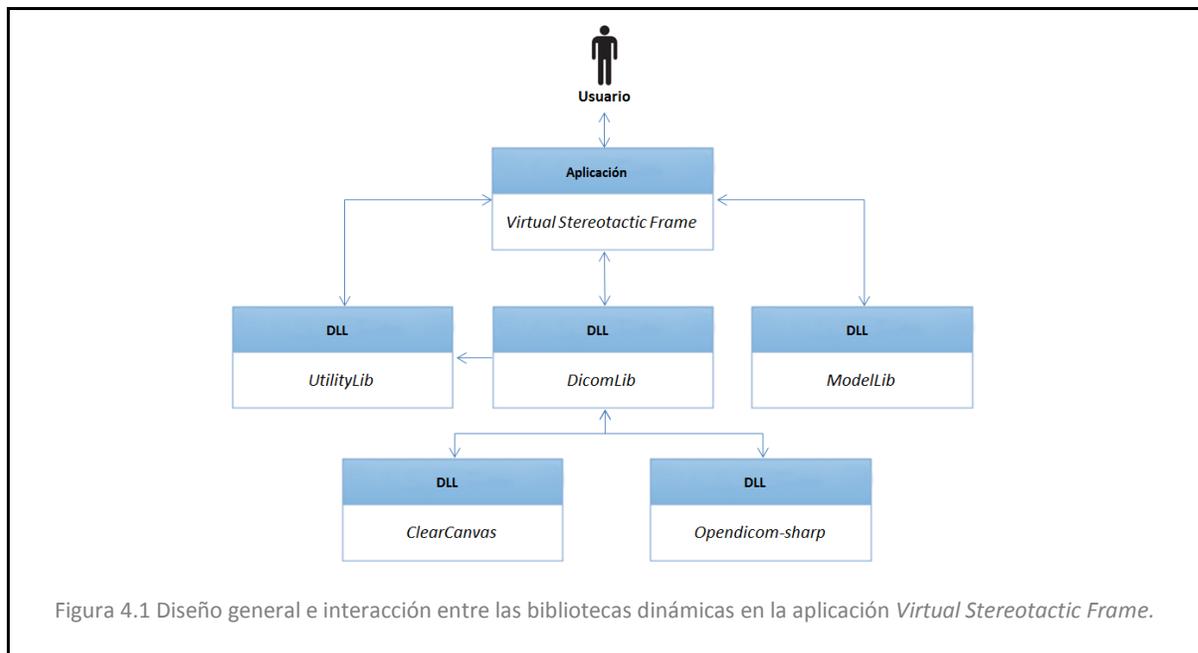


Figura 4.1 Diseño general e interacción entre las bibliotecas dinámicas en la aplicación *Virtual Stereotactic Frame*.

## 4.3 La biblioteca dinámica *DicomLib*

Es el componente más importante de la aplicación. En este se encuentran todos los métodos y clases necesarias para el despliegue de volúmenes, las funciones para cargar y guardar imágenes en formato DICOM y los métodos para el procesamiento del volumen y la generación de las marcas fiduciales. En la **Figura 4.2** se muestra una parte del diagrama de clases de la biblioteca. Es importante destacar que sólo se muestran las clases más importantes y algunos de los métodos más relevantes. Como se puede apreciar la biblioteca está conformada por cuatro paquetes principales: «*Util*», «*Volume*», «*Opendicom-sharp*» [37] y «*ClearCanvas*» [38].

### 4.3.1 Paquete «*Util*»

Como se muestra en la **Figura 4.2**, este paquete importa métodos y atributos públicos de los otros paquetes mencionados. En «*Util*» se encuentran las clases *DicomImage*, *DicomLoader* y *DicomManager*.

La clase *DicomImage* es la estructura donde se almacenan los datos básicos y más relevantes de las imágenes DICOM cargadas. Dentro de los atributos más importantes se encuentran la matriz de bytes (*Bitmap*) que representa la imagen, la ruta y nombre del archivo (*FileName*), la estructura *Header* en donde se almacena la cabecera de la imagen DICOM procesada y el identificador de textura (*TextureID*) que se genera una vez la misma es cargada en la tarjeta gráfica.

Entre los métodos públicos más destacados de la clase *DicomImage* se encuentran:

- *GenerateBitmap()* cuya función es crear un *Bitmap* (estructura del lenguaje de programación C#) recibiendo como entrada un conjunto de bytes almacenados en un arreglo. Dicho *Bitmap* es utilizado posteriormente para la generación de la textura.
- El método *GenerateTexture()* es invocado luego del método anterior y su función es generar una textura en la tarjeta gráfica para su posterior despliegue haciendo uso del *toolkit* de *OpenTK*.
- *UpdateTexture()* tiene una función similar al método anterior, sin embargo este método es usado cuando se quiere actualizar la textura que ya fue generada sin necesidad de destruirla y regenerarla. El método es mayormente utilizado cuando alguna propiedad del volumen o el volumen en sí cambia a lo largo de la ejecución de la aplicación.
- El método *SearchTag()*, cuya función es buscar y retornar el contenido de alguna etiqueta de la cabecera de la imagen DICOM.

La clase *DicomLoader* tiene como finalidad cargar el contenido de las imágenes médicas en formato DICOM y llenar la estructura *DicomImage* con la información necesaria para su funcionamiento. Es una clase estática que dispone de un solo método llamado *Load()*.

Finalmente, la clase *DicomManager* es la encargada de encapsular todo el manejo de las imágenes DICOM comunicándose de forma transparente con las dos clases anteriores para su funcionamiento. Esta clase representa el punto de entrada y salida más importante de toda la aplicación.

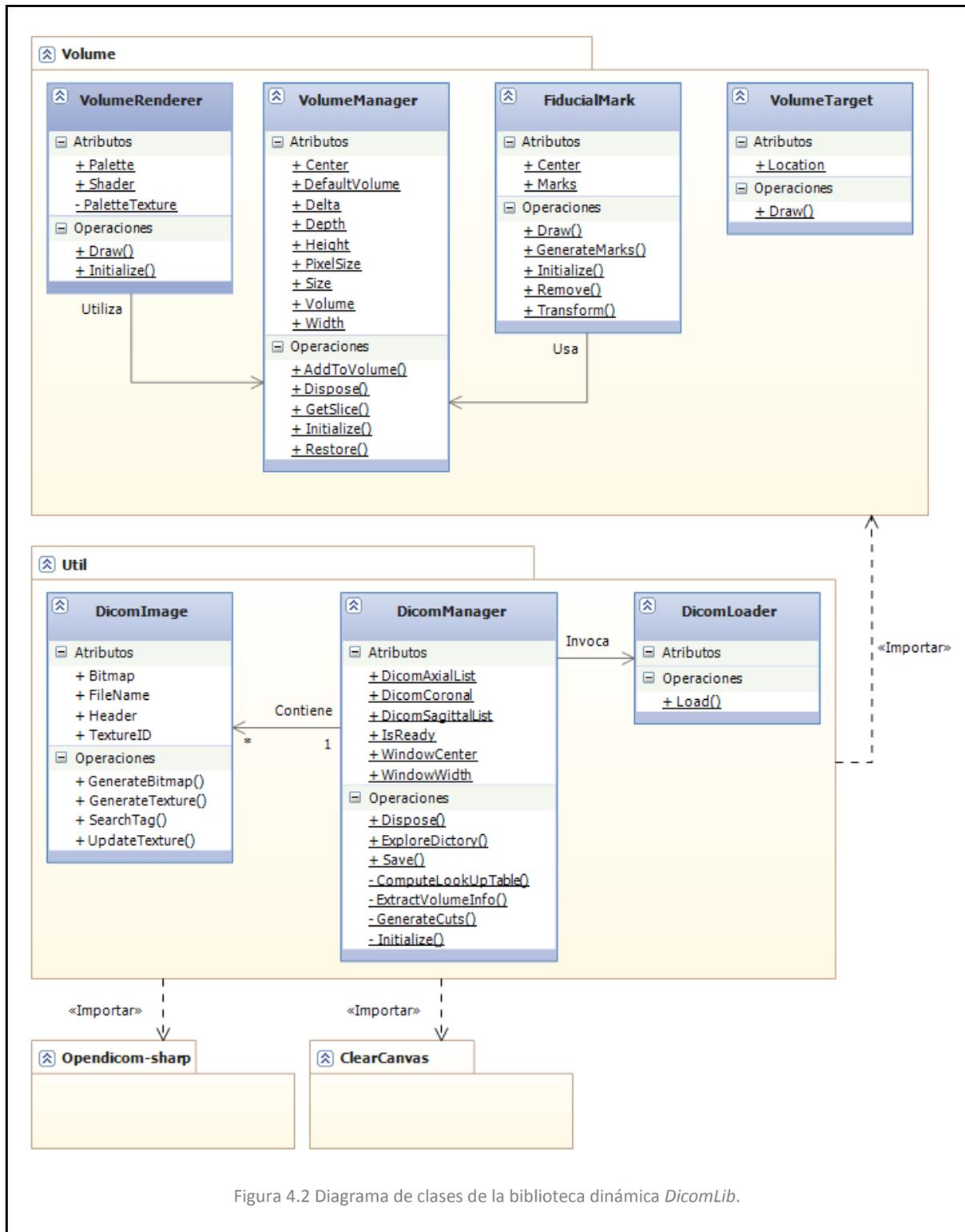


Figura 4.2 Diagrama de clases de la biblioteca dinámica *DicomLib*.

Dentro de los atributos más importantes contenidos en esta clase se encuentran tres listas de tipo *DicomImage* llamadas *DicomAxialList*, *DicomCoronalList* y *DicomSagittalList*. En dichas listas se almacenan de forma ordenada las imágenes DICOM procesadas según su corte (axial, coronal y sagital). También se en-

cuentra una bandera de tipo booleana llamada *IsReady*. Este atributo indica si el volumen fue cargado en su totalidad y está listo para ser desplegado en pantalla. Por último, la clase dispone de dos atributos de tipo entero llamados *WindowWidth* y *WindowCenter*. Dichos valores son usados para calcular la transformación lineal que se le debe aplicar a las imágenes para su correcta visualización.

La clase *DicomManager* se caracteriza por ser una clase estática. La finalidad de esta característica es que todos sus atributos y métodos puedan ser accedidos desde cualquier instancia de la aplicación y que la data que ésta maneja sea consistente en todo momento.

Antes de que el sistema pueda utilizarla, la clase *DicomManager* debe ser iniciada a través del método *Initialize()*. La invocación se hace de forma privada cuando el usuario elige cargar un volumen utilizando el método *ExploreDirectory()*. Una vez iniciada, se puede cargar un volumen e invocar todos sus métodos. Durante el procesamiento de los archivos DICOM de entrada se invoca a los métodos privados *Dispose()*, *ComputeLookUpTable()*, *ExtractVolumeInfo()* y *GenerateCuts()*. El método *Dispose()* descarga de memoria el volumen actualmente cargado (en el caso de que lo haya) y libera todos los recursos utilizados por la clase. El método *ComputeLookUpTable()* genera la transformación lineal que debe ser aplicada a todos los píxeles de las imágenes DICOM. *ExtractVolumeInfo()*, extrae los datos necesarios para iniciar la clase *VolumeManager*. *GenerateCuts()* se encarga de generar los cortes necesarios (axial, coronal y sagital) y llenar las listas de *DicomImage* mencionadas anteriormente. En el **Algoritmo 4.1** se muestra de forma general el algoritmo utilizado al momento de invocar al método *ExploreDirectory()*.

1. Invocar al método *Dispose()*
2. Invocar al método *Initialize()*
3. Por cada archivo DICOM en el directorio a explorar:
  - | | Instanciar un objeto *DicomImage*
  - | | Invocar al método *Load()* de *DicomLoader* con el objeto instanciado
  - | | Agregar la imagen una vez cargada a la lista *DicomAxialList*
  - | | Verificar si es la primera imagen que se lee, de ser así invocar a *ExtractVolumeInfo()* y seguidamente invocar *ComputeLookUpTable()* y el método *Initialize()* de *VolumeManager*
4. Procesar los tres cortes del volumen
5. Invocar al método *GenerateCuts()*
6. Asignar verdadero al atributo *IsReady*

Algoritmo 4.1 Método *ExploreDirectory()* de la clase *DicomManager*.

En este punto es importante señalar que el procesamiento de las imágenes DICOM realizadas en el método *ExploreDirectory()* se realizan de forma paralela o simultánea en la Unidad Central de Procesamiento (UCP o CPU, por sus siglas en inglés) haciendo uso de la Biblioteca de Procesamiento Paralelo basado en Tareas (TPL, por las siglas del término en inglés Task Parallel Library) que proporciona el *framework Microsoft .NET* [39].

Para finalizar, el método *Save()* se encarga de almacenar el volumen procesado en formato DICOM o formato RAW en el directorio indicado por el usuario. Es importante destacar que en el caso de que el usuario seleccione almacenar el volumen en formato DICOM solo se guardarán los cortes axiales del mismo marcando, opcionalmente, los archivos resultantes como imágenes de captura secundaria.

### 4.3.2 Paquete «Volume»

En este paquete se encuentran todas las clases necesarias para el despliegue y procesamiento del volumen, así como la clase que permite posicionar la lesión del paciente en el espacio. En «Volume» se hallan las clases *VolumeRenderer*, *VolumeManager*, *FiducialMark* y *VolumeTarget*.

La clase *VolumeManager* contiene toda la información relevante al volumen tal como su tamaño (ancho, alto y profundidad), el tamaño de los píxeles en cada corte, la ubicación del centro, unidades de escalamiento, la separación entre los cortes, y la matriz tridimensional en donde se almacenan todos los vóxeles. Cabe mencionar que la clase maneja dos matrices de vóxeles, una de ellas llamada *Volume* que contiene la información del volumen que se visualiza actualmente, y la otra llamada *DefaultVolume* donde se almacenan los vóxeles originales del volumen. Esta última es usada en el caso de que el volumen actual sea modificado y necesite ser restaurado a su instancia original.

La clase *VolumeManager* es totalmente estática, de tal forma que se pueda garantizar el acceso a sus métodos y atributos desde cualquier instancia de la aplicación u otros paquetes. Para poder hacer uso de la clase es necesario iniciarla a través del método público *Initialize()*, en donde se asignan los valores de alto, ancho y profundidad del volumen; se instancian las matrices tridimensionales y se calculan los valores para los demás atributos en base con la información suministrada por el método invocador.

Luego de ser iniciada, la clase permite agregar cortes axiales al volumen a través del método *AddToVolume()*. Se pueden obtener cortes axiales, coronales o sagitales del volumen a través del método *GetSlice()*, con tan solo indicar el índice del corte deseado y el tipo de corte. Se puede restaurar el volumen actual a su forma original con el método *Restore()* y se pueden liberar los recursos utilizados por la clase a través del método *Dispose()*.

La clase *VolumeRenderer* tiene como propósito hacer el despliegue del volumen en pantalla utilizando la técnica de texturas bidimensionales llamada Planos Alineados a los Ejes del Objeto.

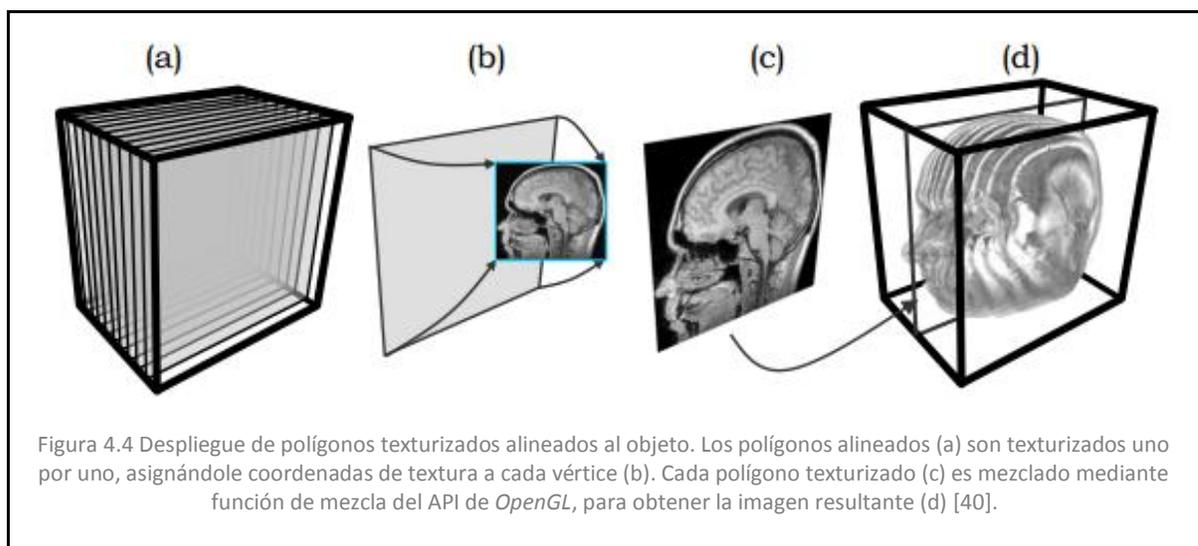
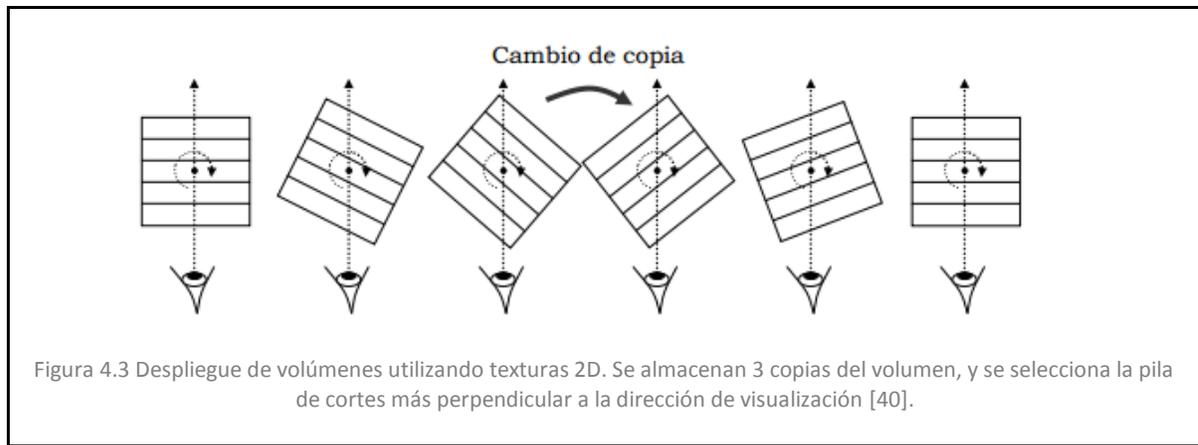
Como toda técnica de despliegue de volúmenes (*Volumen Rendering*, en inglés) basada en el uso de texturas bidimensionales, se requiere la mezcla de geometrías intermedias que se texturizan con cortes planos del volumen [40]. La técnica de texturas bidimensionales con planos alineados a los ejes del objeto consiste en generar tres copias del volumen como pilas de cortes axiales, coronales y sagitales que se almacenen en texturas 2D.

Para realizar el despliegue del volumen es necesario elegir la pila de cortes que sea más perpendicular a la dirección de visualización [40] y seguidamente desplegar una serie de polígonos texturizados alineados a los ejes del objeto usando la función de mezcla (*blending*) del API de *OpenGL* como se muestra en la **Figura 4.3** y la **Figura 4.4**.

Antes de poder hacer uso de la clase *VolumeRenderer* es necesario invocar al método *Initialize()* para asignar los valores iniciales de los atributos de la clase y cargar el *shader*<sup>2</sup> a utilizar en la Unidad Gráfica de Procesamiento (GPU, por sus siglas en inglés).

---

<sup>2</sup> El *shader* fue programado para *OpenGL Shading Language* (GLSL).



Para realizar el despliegue volumétrico se utilizó el *toolkit* de *OpenTK* y solo debe invocarse al método *Draw()* de la clase *VolumeRenderer* indicando la posición de visualización a través de la estructura *Vector3* de *OpenTK*. La clase maneja internamente la función de transferencia que rige la visualización del volumen en el canal alfa (transparencia) a través de una textura 1D que es enviada a la tarjeta gráfica y es procesada en la GPU.

La clase *FiducialMark* es la encargada de desplegar y procesar las marcas fiduciales en el volumen. Dispone de los métodos *Initialize()* para asignar los valores iniciales a los atributos de la clase, el método *Draw()* cuya función es desplegar en pantalla una guía tridimensional que indica la localización de las marcas en el espacio como se muestra en la **Figura 4.5**, el método *GenerateMarks()* el cual procesa los vóxeles del volumen y genera las marcas fiduciales en el mismo y el método *Remove()* el cual invoca al método *Restore()* de la clase *VolumeManager*.

La clase dispone de tres atributos principales los cuales facilitan los cálculos para el posicionamiento de las marcas en el volumen, estos son: el *MarkThickness* que es usado para configurar el grosor de las marcas fiduciales (representado en milímetros), *Center* que indica en todo momento el centro de la caja conte-

nedora formada por las ocho marcas fiduciales o puntos principales y *Marks* que contiene las posiciones en el espacio de las ocho marcas fiduciales principales.

En el **Algoritmo 4.2** se muestra de manera general el conjunto de pasos que se siguen al invocar el método *GenerateMarks()* para generar las marcas fiduciales en el volumen.

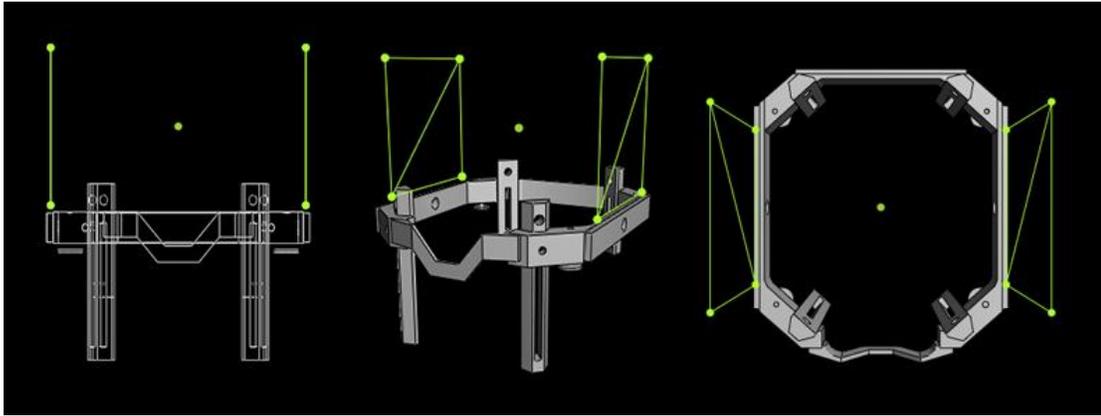


Figura 4.5 Guía para la visualización de las marcas fiduciales en el espacio.

El proceso para hallar los vóxeles que deben ser marcados en el volumen y que representarán las marcas fiduciales se realiza mediante una correspondencia directa entre coordenadas en espacio objeto y las posiciones dentro de la matriz tridimensional que contiene el volumen. En la **Figura 4.6** se muestran los parámetros que intervienen en dicho proceso.

1. **Invocar** al método *Restore()* de *VolumeManager* (asegura que solo exista un conjunto de marcas a la vez)
2. **Generar** una máscara circular con el tamaño en vóxeles correspondiente al grosor de las marcas fiduciales
3. **Calcular** los ocho puntos principales de las marcas fiduciales en espacio objeto y **aplicar** las transformaciones de traslación y rotación necesarias usando el método *Transform()*
4. **Hallar** la correspondencia entre las coordenadas en espacio objeto y los vóxeles del volumen
5. **Marcar** los vóxeles del volumen que correspondan a los ocho puntos iniciales y aplicar la máscara circular generada en el paso 2
6. **Interpolar** los ocho puntos (como correspondan) en espacio objeto y **hallar** su correspondencia con los vóxeles del volumen
7. **Agregar** todos los valores encontrados a un diccionario para asegurar que no se repitan marcas
8. Por cada valor en el diccionario final hacer:
  - | **Marcar** los vóxeles del volumen con el valor de intensidad más alto haciendo uso de los valores del diccionario
9. **Reconstruir** las texturas del volumen

Algoritmo 4.2 Pasos que se siguen en la invocación del método *GenerateMarks()* de la clase *FiducialMarks*.

Para empezar es necesario establecer los valores máximos y mínimos que ocupa la caja contenedora alineada a los ejes (*AABB - Axis-Aligned Bounding Box*) del volumen. Dichos valores pueden obtenerse dividiendo a la mitad cada coordenada del atributo *Size* (representado como un *Vector3*) de la clase *VolumeManager*. Los resultados representan los valores máximos del *Bounding Box*, y negando cada coordenada se pueden obtener los valores mínimos.

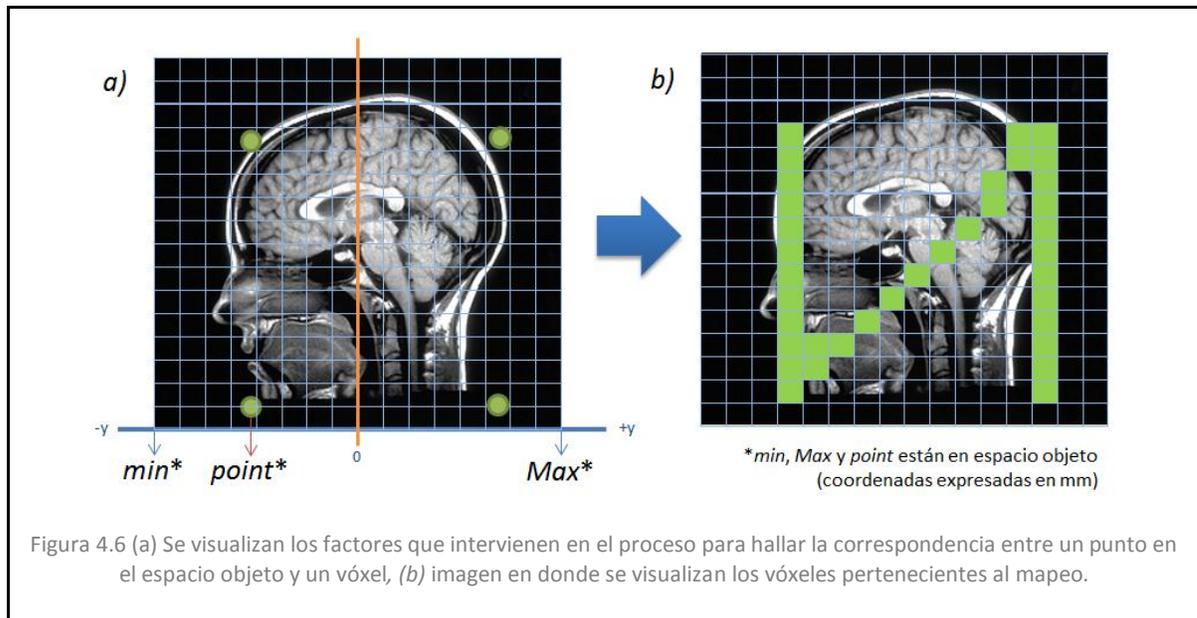


Figura 4.6 (a) Se visualizan los factores que intervienen en el proceso para hallar la correspondencia entre un punto en el espacio objeto y un vóxel, (b) imagen en donde se visualizan los vóxeles pertenecientes al mapeo.

Una vez conocidos los puntos en el espacio objeto donde irán las marcas fiduciales y la caja contenedora del volumen el procedimiento para la asignación de vóxeles resulta trivial y puede verse a grandes rasgos en el **Algoritmo 4.3** y la **Ecuación 4.1** (los cálculos para las otras coordenadas son equivalentes. Sólo hay que hacer uso de cada dimensión del volumen (Width, Height, Depth) según corresponda).

1. **Extraer** la caja contenedora del volumen (*aabb*)
2. Para cada valor de *t* entre 0 y 1 con un salto predefinido de 0.00001 hacer:
  - | | **Interpolan** entre los dos puntos A y B en espacio objeto con el valor *t* (*intp*)
  - | | Si *intp* no está **contenido** en *aabb*, **descartar** el punto y continuar con la siguiente iteración
  - | | **Utilizar** la **Ecuación 4.1**
3. **Agregar** el vóxel hallado a la lista

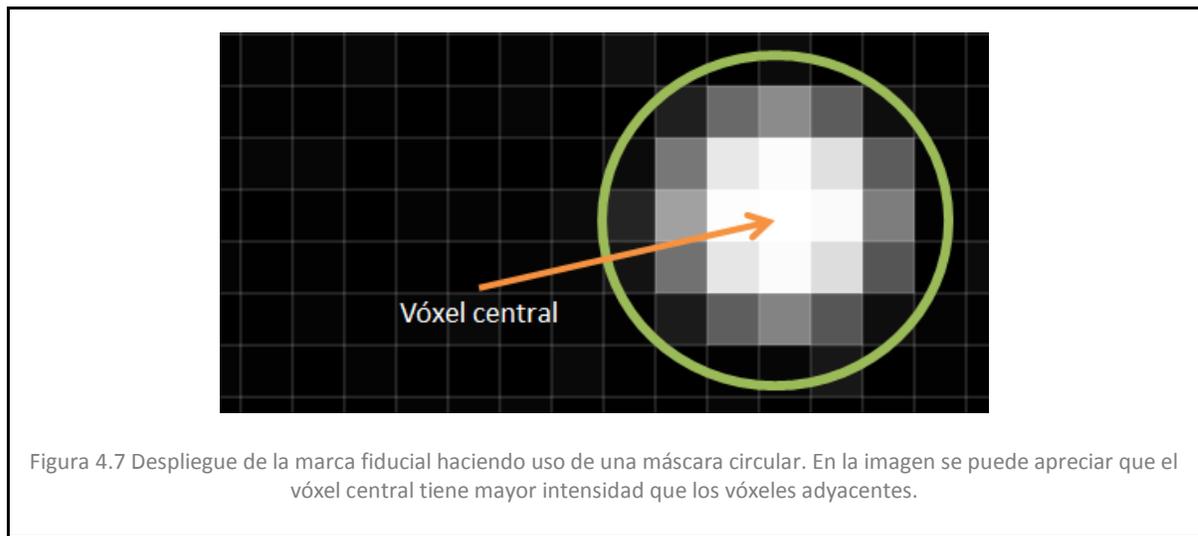
Algoritmo 4.3 Correspondencia entre un punto en espacio objeto y un vóxel. Método *GenerateMarks()*.

$$vóxel_x(intp_x) = \begin{cases} \left( \frac{intp_x - aabb.mín_x}{-aabb.mín_x} \right) \times \left( \frac{Volume_{width} - 1}{2} \right), & intp_x < 0 \\ \left( \frac{intp_x}{aabb.máx_x} + 1 \right) \times \left( \frac{Volume_{width} - 1}{2} \right), & intp_x \geq 0 \end{cases}$$

Ecuación 4.1 Ecuación para realizar el mapeo entre la coordenada X en espacio objeto y los vóxeles en el volumen.

El siguiente paso será marcar estructuralmente los vóxeles hallados en el volumen. En ésta sección se utiliza la máscara circular calculada al inicio del método *GenerateMarks()*. Dicha máscara indica cuales vóxeles se deben marcar y el valor de intensidad que le corresponde a cada uno dependiendo de la posición que ocupe dentro del círculo. Cabe destacar que para hacer uso de la máscara, el centro de ésta debe estar alineado al vóxel central calculado previamente.

Como puede apreciarse en la **Figura 4.7** el vóxel central hallado en el algoritmo **Algoritmo 4.3** tendrá la máxima intensidad de color y mientras más cerca del borde del círculo se encuentre el vóxel su valor de intensidad irá disminuyendo. Dicho procedimiento causa un efecto conocido como *antialiasing* y es usado para suavizar los bordes, disimularlos, y conseguir una apariencia más realista sin perder o deformar la imagen original (en este caso, un círculo).



Para finalizar, la última clase perteneciente a este paquete es *VolumeTarget*. Su propósito es permitir la localización y despliegue del blanco quirúrgico en la aplicación *Virtual Stereotactic Frame*. La clase cuenta con un solo método para realizar el despliegue del blanco quirúrgico en pantalla llamado *Draw()* además de un atributo de tipo *Vector3* llamado *Location*, en donde se almacenan las coordenadas de la lesión en espacio objeto.

### 4.3.3 Paquetes «Opendicom-sharp» y «ClearCanvas»

Los paquetes «*Opendicom-sharp*» y «*ClearCanvas*» fueron utilizados para brindar soporte a la biblioteca dinámica *DicomLib* en lo que respecta a cargar, procesar y salvar imágenes DICOM.

El paquete «*Opendicom-sharp*» es un conjunto de clases que forman parte del proyecto openDICOM.NET [37] el cual provee un API para trabajar con DICOM para el *framework* de *Microsoft .NET* y *Mono*, bajo el lenguaje de programación *C#*. El paquete «*Util*» importa el método *LoadFrom()* para la carga de archivos en formato DICOM y hace uso de la estructura *AcrNemaFile* para procesar y extraer la imagen y su cabecera.

Por otro lado, el paquete «*ClearCanvas*» que forma parte del *SDK* del proyecto de *ClearCanvas* [38] contiene todas las clases y métodos necesarios para guardar imágenes en formato DICOM. El paquete «*Util*» importa los métodos *Load()* para cargar el contenido de una imagen DICOM y *Save()* para crear un nuevo archivo y almacenar una imagen DICOM. Es importante destacar que al momento de salvar los archivos ciertos datos de la cabecera como *MediaStorageSopClassUid*, el *SopClassUid* y el *ImageType* deben ser modificados para indicar que la imagen de primera captura ha sido procesada por un *software* y por lo tanto corresponde ahora a una imagen de captura secundaria.

Si bien ambos proyectos tienen como finalidad el tratamiento del estándar DICOM, por sí solos no disponen de todos los métodos necesarios para brindar soporte a la implementación de la aplicación VSF. Por ello ambos paquetes son usados complementariamente para cubrir las carencias de cada uno de ellos. Mientras que el paquete «*ClearCanvas*» dispone de los métodos para salvar archivos DICOM (que no están presentes en el otro) el paquete «*Opendicom-sharp*» es usado para cargar y procesar las imágenes DICOM. De esta forma la aplicación VSF brinda soporte total para cargar y salvar archivos en formato DICOM.

## 4.4 La biblioteca dinámica *ModelLib*

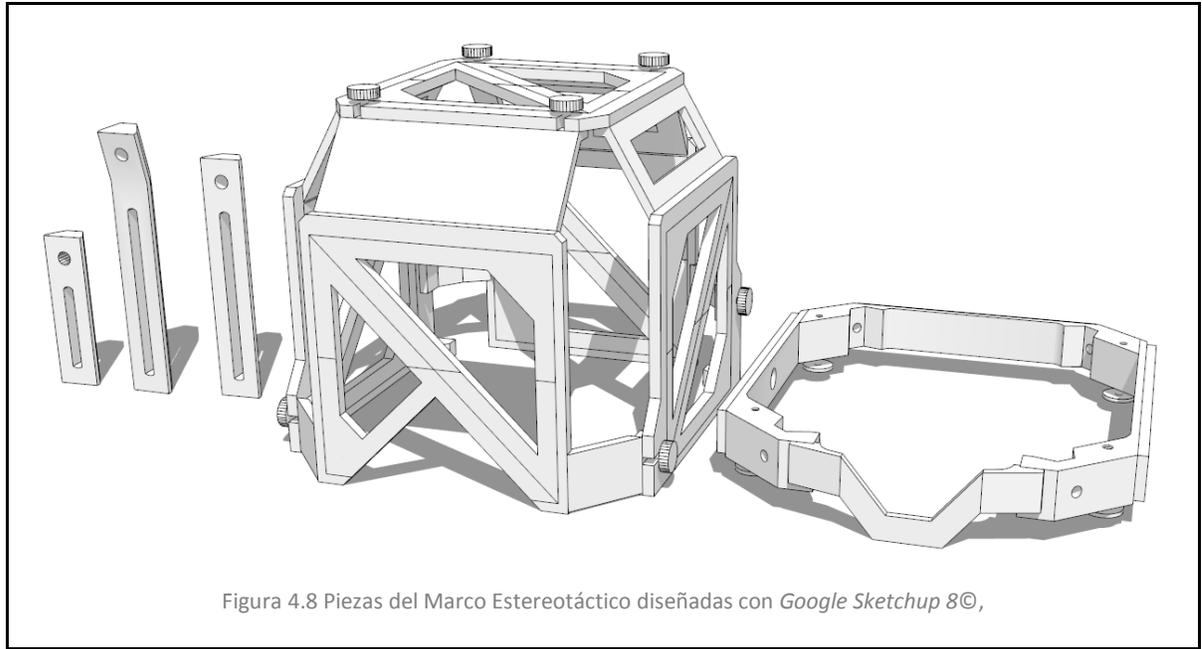
Para realizar el despliegue del Marco Estereotáctico (ME) y sus otros componentes se diseñó e implementó la biblioteca dinámica *ModelLib*. Esta biblioteca es la encargada de cargar, desplegar y encapsular todos los métodos y propiedades de las piezas que conforman el ME (los postes, el marco estereotáctico y la caja radiopaca).

Todas las piezas que por defecto maneja la aplicación VSF fueron diseñadas con *Google Sketchup 8* [41] haciendo uso de aproximaciones a las escalas reales de cada componente y fueron exportadas en formato OBJ por la misma aplicación. En la **Figura 4.8** se pueden apreciar las piezas diseñadas para la aplicación VSF.

La razón principal por la cual los modelos fueron exportados en formato OBJ es porque éste es aceptado universalmente por un gran número de visualizadores de modelos 3D y aplicaciones lo cual permite la edición de las piezas usadas en cualquier momento.

El formato de archivo OBJ es un tipo de archivo donde se almacenan geometrías 3D creado por *Wavefront* para su producto *Advanced Visualizer™* [42]. Estos archivos están basados en texto plano y son compatibles con geometrías de formas libres y poligonales. Además, almacenan datos como la posición de cada vértice, de cada coordenada de textura, las normales, y las caras que representan los polígonos de la geometría almacenada.

En base con lo anteriormente descrito se puede definir la biblioteca dinámica *ModelLib* como un conjunto de clases cuyo propósito es procesar archivos en formato OBJ y desplegarlos en pantalla haciendo uso del *toolkit* de *OpenTK*.



Como se muestra en la **Figura 4.9** la biblioteca dinámica *ModelLib* está constituida por seis clases principales: *ModelManager*, *ModelMesh*, *ModelLoader*, *BoxSetting*, *FrameSetting* y *PostSetting*.

En la clase *ModelMesh* se almacena la geometría que define a cada uno de los modelos 3D o piezas del ME utilizados en la aplicación. Todos los modelos están conformados por una lista de vértices (*Vertices*), una de triángulos (*Triangles*), una de cuadriláteros (*Quads*) y una de aristas (*Edges*). Adicionalmente, disponen de un nombre (*Name*) que los identifica y diferencia de los otros modelos.

Para realizar el despliegue de la geometría en pantalla se utiliza el método *Draw()*. En este punto es importante señalar que dependiendo del soporte y la versión de *OpenGL* que brinde la tarjeta gráfica, la clase es capaz de desplegar los modelos de tres formas diferentes.

La primera forma de despliegue es usando un *Vertex Buffer Object*<sup>3</sup> (*VBO*). Un *VBO* es una característica de *OpenGL* que provee métodos para cargar datos (vértices, vectores normales, colores, etc.) a la tarjeta gráfica para un despliegue no inmediato. Esta característica ofrece ganancias sustanciales en el rendimiento al momento de desplegar geometrías con un gran número de polígonos debido a que los datos residen en la memoria del dispositivo gráfico en lugar de la memoria principal del sistema, por lo que su despliegue puede realizarse directamente por el dispositivo de video [43].

En caso de no tener soporte para el uso de *VBO* la clase *ModelMesh* intentará desplegar el modelo con el uso de *Display List*. El *DisplayList* (*DL*) es un grupo de comandos de *OpenGL* que han sido almacenados (compilados) para una ejecución posterior. Una vez que el *DL* es creado (proceso que se realiza una sola vez), todos los vértices son copiados en la memoria de la tarjeta gráfica. La ventaja en el uso de este tipo de despliegue es que una vez que el *DL* es compilado, éste puede usarse repetidamente sin necesidad de retransmitir la información una y otra vez al dispositivo gráfico en cada cuadro [44].

<sup>3</sup> La especificación de *Vertex Buffer Object* ha sido estandarizada por *OpenGL Architecture Review Board* como parte de *OpenGL* versión 1.5 (en el 2003) [43].

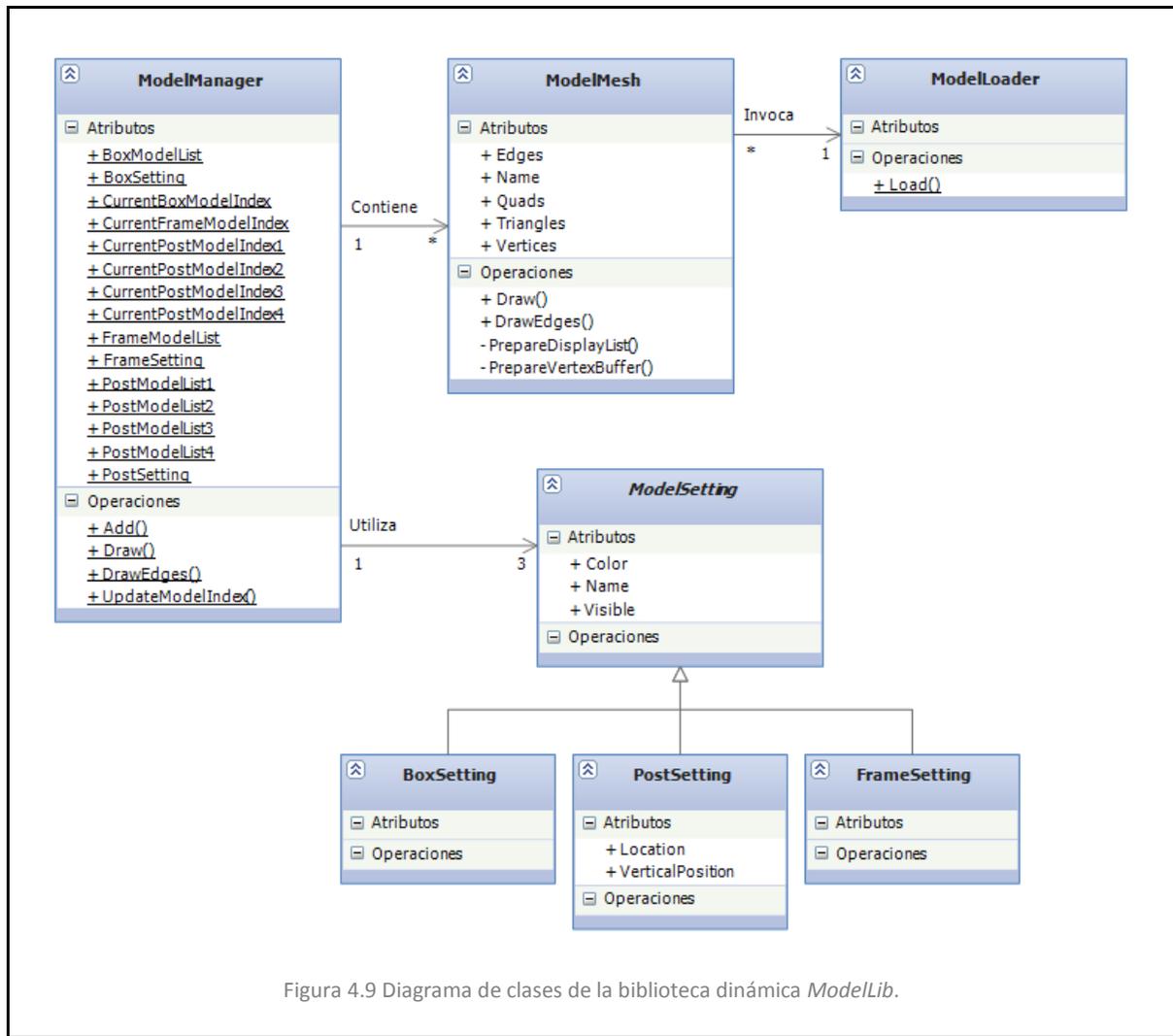


Figura 4.9 Diagrama de clases de la biblioteca dinámica *ModelLib*.

La tercera y última forma de despliegue fue implementada para tarjetas gráficas que no soporten las dos formas anteriores. En esta implementación simplemente se recorren las listas de polígonos del modelo y se despliegan cada uno de ellos usando la forma tradicional. Como consecuencia deben recorrerse todos los polígonos del modelo. Este es el método de menor rendimiento.

Para poder hacer uso de la clase *ModelMesh* es necesario cargar un archivo de tipo OBJ. Para ello la biblioteca dinámica *ModelLib* hace uso de la clase *ModelLoader*, específicamente de su método estático *Load()*, cuya función es obtener y procesar la geometría contenida en un archivo con formato OBJ y asignar a los atributos de la clase *ModelMesh* los valores correspondientes. Dicho método es invocado en el constructor de la clase *ModelMesh*.

La clase *ModelSetting* es la encargada de modificar las propiedades de despliegue de los distintos componentes del ME. Entre los atributos visuales que se pueden modificar están: *Color* que modifica el color con el que se pinta el componente y *Visible* que determina si el componente será mostrado o no. El otro atributo es *Name* el cual indica el nombre del *ModelMesh* que se está desplegando en un momento dado.

Como se muestra en la **Figura 4.9** la clase *ModelSetting* es abstracta y se deriva en las clases *BoxSetting*, *FrameSetting*, *PostSetting*. Cada una de sus clases derivadas tienen como función controlar los valores visuales de cada pieza del ME. Por tanto, la clase *BoxSetting* controla el aspecto visual de la caja radiopaca, la clase *FrameSetting* controla el ME y la clase *PostSetting* controla el aspecto y la configuración de los postes. Ésta última implementa dos atributos adicionales: *Location*, que indica la posición del poste en el ME (anterior derecho, anterior izquierdo, posterior derecho y posterior izquierdo) y *VerticalPosition* que indica la posición vertical (expresada en milímetros) del poste en el ME.

Con el diseño de las clases *BoxSetting*, *FrameSetting* y *PostSetting* se logra separar completamente el despliegue de los modelos con la configuración de visualización de los mismos, de esta forma, la composición y despliegue del ME se realiza de forma transparente para la aplicación VSF, además de otorgar mayor modularidad al proyecto.

Finalmente, está la clase *ModelManager*, la cual encapsula toda la información necesaria para el despliegue de las piezas y todas sus propiedades. La clase cuenta con seis listas (*BoxModelList*, *FrameModelList*, *PostModelList1*, *PostModelList2*, *PostModelList3* y *PostModelList4*) de tipo *ModelMesh* en donde se almacenan todos los modelos a ser desplegados dependiendo de su tipo (ME, postes, o cajas radiopacas). Adicionalmente, la clase contiene seis índices (*CurrentBoxModelIndex*, *CurrentFrameModelIndex*, *CurrentPostModelIndex1*, *CurrentPostModelIndex2*, *CurrentPostModelIndex3* y *CurrentPostModelIndex4*) de tipo entero que indican las posiciones en las listas en donde se encuentran los modelos que están siendo desplegados actualmente o han sido seleccionados por el usuario. La clase también almacena la configuración de todas las piezas haciendo uso de sus atributos *BoxSetting*, *PostSetting* y *FrameSetting*. En este punto es importante destacar que para la configuración y visualización de los cuatro postes se manejan atributos independientes con el fin de facilitar la lógica interna del código.

Entre los métodos más importantes de la clase *ModelManager* se encuentran: *Add()* cuya función es agregar un modelo a una de las listas dependiendo de su tipo; *Draw()* y *DrawEdges()* los cuales se encargan de desplegar los modelos en pantalla; y *UpdateModelIndex()* encargado de actualizar los índices cuando el usuario seleccione un modelo diferente al actual para que sea desplegado en pantalla.

## 4.5 La biblioteca dinámica *UtilityLib*

La biblioteca dinámica *UtilityLib* es la encargada de brindar soporte a la aplicación VSF en lo que respecta a la interacción entre el usuario y la interfaz gráfica. Su objetivo es encapsular en diferentes paquetes el despliegue y la lógica de funcionamiento de los distintos componentes que utiliza la aplicación. Dicha biblioteca puede verse como una capa que separa los componentes que se utilizan en la interfaz gráfica de usuario y la aplicación en sí. De esta forma, todos los métodos y clases contenidas en esta biblioteca dinámica pueden ser reutilizados en otras aplicaciones de manera transparente<sup>4</sup>.

Como se puede apreciar en la **Figura 4.10**, la biblioteca dinámica *UtilityLib* está compuesta por cuatro paquetes principales: «*Glu*», «*Transform*», «*Graphics*» y «*Math*»; adicionalmente, existen dos clases que no pertenecen a ningún paquete llamadas *Camera* y *Shader*.

---

<sup>4</sup> Para poder integrar a la biblioteca dinámica *UtilityLib* con otras aplicaciones es necesario que estén codificadas en el lenguaje de programación *C#* con el *Microsoft .NET Framework 4.0* o equivalente en *Linux* y utilicen el *toolkit* de *OpenTK*.

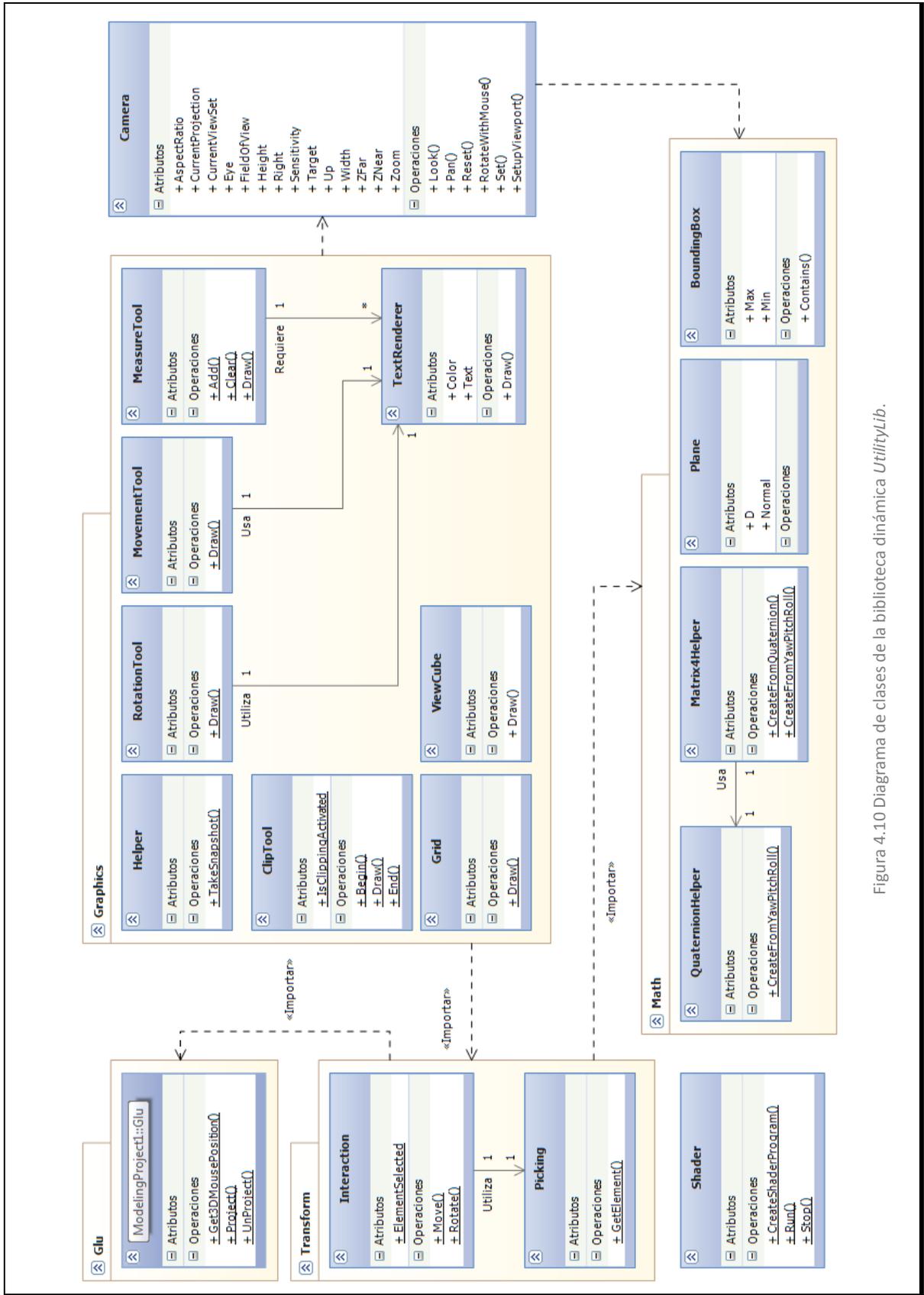


Figura 4.10 Diagrama de clases de la biblioteca dinámica *UtilityLib*.

### 4.5.1 Paquete «Glu»

El propósito de este paquete es brindar un soporte de más alto nivel a los métodos proporcionados por el *toolkit* de *OpenTK* y al API de *OpenGL*. El paquete está constituido solo por la clase *Util*. Dicha clase ofrece tres métodos estáticos: *Project()* y *UnProject()* los cuales permiten el mapeo entre coordenadas en espacio objeto y coordenadas de ventana y el mapeo entre coordenadas de ventana y coordenadas en espacio objeto, respectivamente; y el método *Get3DMouse-Position()* el cual retorna la posición del cursor en coordenadas en espacio objeto.

### 4.5.2 Paquete «Math»

La finalidad de este paquete es ofrecer soporte matemático y agregar algunas estructuras y métodos que no están disponibles en el *toolkit* de *OpenTK* ni en el lenguaje de programación *C#*. El paquete está conformado por cuatro clases principales (*QuaternionHelper*, *Matrix4Helper*, *Plane* y *BoundingBox*) y es importado por la clase *Camera* y el paquete «Glu».

Las clases *QuaternionHelper* y *Matrix4Helper* son utilizadas principalmente para realizar las rotaciones de la cámara (invocadas por el método *RotateWithMouse()* de la clase *Camera*). La clase *Plane* es usada para calcular intersecciones de rayos con planos, primordialmente utilizada en la clase *Interaction* del paquete «Transform». Por último, la clase *BoundingBox* es usada en la biblioteca dinámica *DicomLib*, específicamente en la clase *FiducialMark*, y su función es almacenar la estructura de una caja contenedora alineada a los ejes (*Axis-Aligned Bounding Box*).

### 4.5.3 Paquete «Graphics»

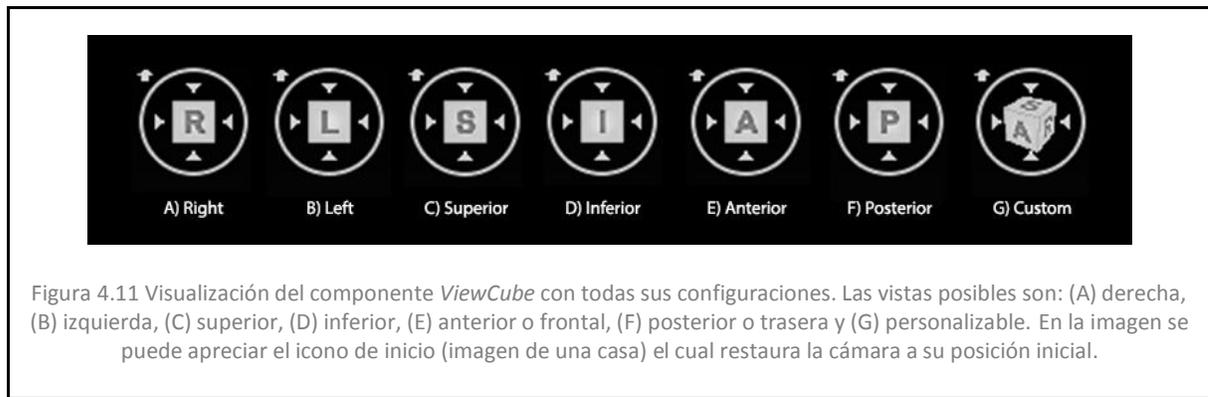
En este paquete se encuentran todos los componentes gráficos que dan soporte a la aplicación VSF, las clases que permiten el despliegue de herramientas de interacción con el usuario y datos como mediciones, posiciones, etiquetas, etc. El paquete hace uso de la clase *Camera* e importa las clases y estructuras del paquete «Transform». Está compuesto por ocho clases principales: *ViewCube*, *Helper*, *RotationTool*, *MovementTool*, *MeasureTool*, *ClipTool*, *Grid* y *TextRenderer*. Cada una de las clases mencionadas representa un componente gráfico que permite la interacción con el usuario a través de los dispositivos de entrada tradicionales, teclado y ratón.

La clase *ViewCube* despliega el componente para la selección de la vista de la cámara (derecha, izquierda, superior, inferior, anterior o frontal, posterior o trasera y personalizable). Dicho componente está formado por un cubo donde se visualizan las iniciales de cada vista en inglés (Left, Right, Superior, Inferior, Anterior y Posterior), cuatro flechas y el icono de inicio con la imagen de una pequeña casa. El componente utiliza la clase *Camera* para su funcionamiento.

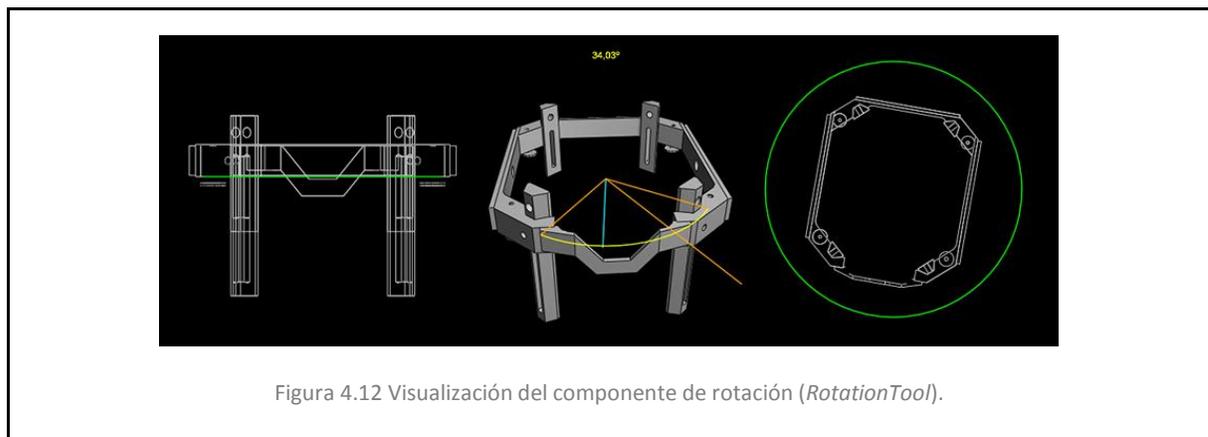
El cubo permite al usuario seleccionar (haciendo clic) la vista que desea tener de la escena, además de ofrecer una guía que indica la orientación de la cámara en todo momento. Es importante destacar que todas las rotaciones que realice la cámara son reflejadas también en el cubo de este componente. Las flechas apuntan a las caras vecinas de la cara actualmente seleccionada y se utilizan para cambiar de una vista

a otra de manera rápida y sencilla. Por último, el icono de inicio (imagen de la casa) permite que la cámara se coloque en su posición inicial. En la **Figura 4.11** se puede apreciar el componente *ViewCube* utilizado en la aplicación VSF mostrando sus diferentes configuraciones.

La clase *Helper* contiene un solo método llamado *TakeSnapshot()* cuya función es capturar en una imagen lo que se muestra en pantalla. Dicho método es estático y una vez que es invocado y después de capturar el contenido de la pantalla, despliega una ventana de diálogo que permite guardar el resultado en la ubicación y formato de imagen especificado por el usuario.



Para poder girar el ME fue necesario diseñar un componente que permitiera al usuario realizar rotaciones usando el ratón. El componente diseñado está contenido en la clase *RotationTool* donde se encapsula toda la lógica necesaria para el despliegue en pantalla del componente de rotación, los indicadores de ángulos y el eje de rotación seleccionado por el usuario. Por cuestiones de requerimientos, la aplicación despliega el componente para que haya rotaciones sólo en el eje Y<sup>5</sup>. Como se puede ver en la **Figura 4.12** el componente de rotación está conformado por un aro o circunferencia centrada en el ME y dos líneas indicadoras principales. La circunferencia actúa como interfaz que permite al usuario hacer clic sobre ella para realizar una rotación específica y las dos líneas funcionan de guías que indican la posición inicial de la rotación (donde se hizo clic) y la posición final de la rotación (donde se encuentra actualmente el cursor del ratón). Adicionalmente, se despliega una etiqueta que indica el ángulo actual expresado en grados.



<sup>5</sup> Expresado en coordenadas de *OpenGL*. Para fines médicos y de visualización la aplicación invierte los ejes Y y Z, en otras palabras, que el usuario sólo verá rotaciones en el eje Z.

Otra herramienta de transformación está encapsulada en la clase *MovementTool*. Esta clase es la encargada de controlar el componente de traslaciones o movimientos del ME. El componente está formado por tres flechas paralelas a los ejes X, Y y Z; una etiqueta para el despliegue de las medidas y tres selectores triangulares, como puede verse en la **Figura 4.13**.

Para interactuar con esta herramienta el usuario debe hacer clic sobre las flechas o los selectores de plano. Cada uno de los elementos tiene comportamientos distintos. Al hacer clic sobre las flecha el usuario solo tendrá permitido trasladar el ME sobre el eje seleccionado, mientras que si selecciona los otros indicadores podrá desplazar el ME por el plano que corresponda (XY, XZ o YZ). Al iniciar el movimiento se mostrará una etiqueta que indicará la distancia desplazada en cada eje con respecto a la posición inicial. Dicha distancia está expresada en milímetros.

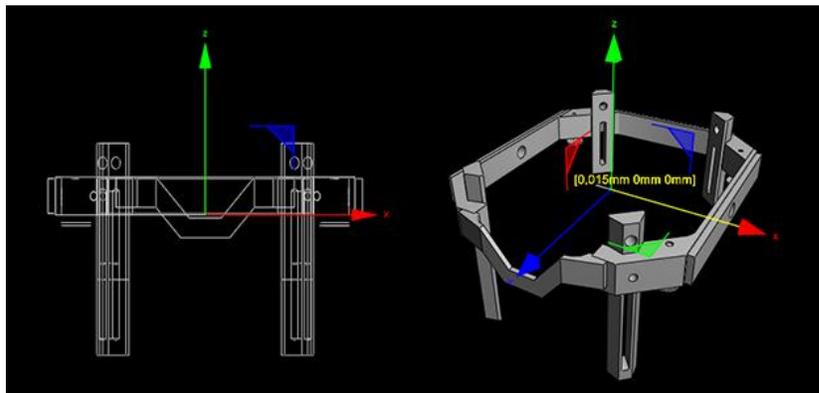


Figura 4.13 Visualización de la herramienta de traslación (*MovementTool*).

Como parte de los requerimientos la aplicación debería proporcionar una herramienta que permitiera realizar mediciones utilizando una especie de regla virtual que señalara la longitud de la distancia medida con líneas y que desplegara en pantalla dicha medida expresada en milímetros como puede verse en la **Figura 4.14**.

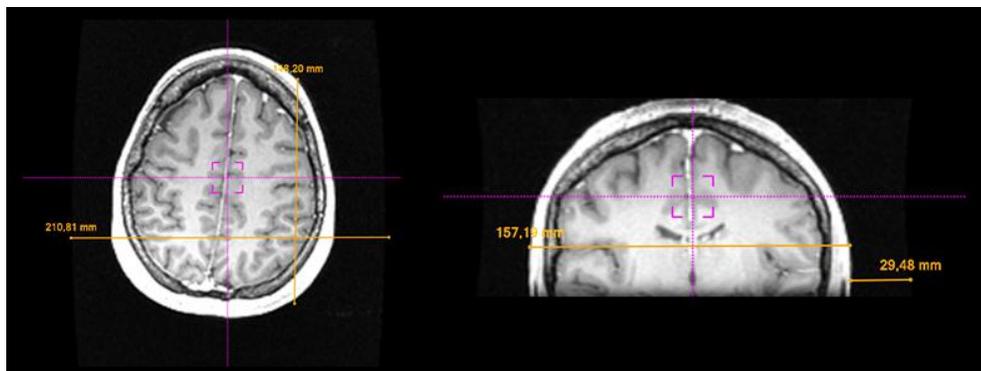


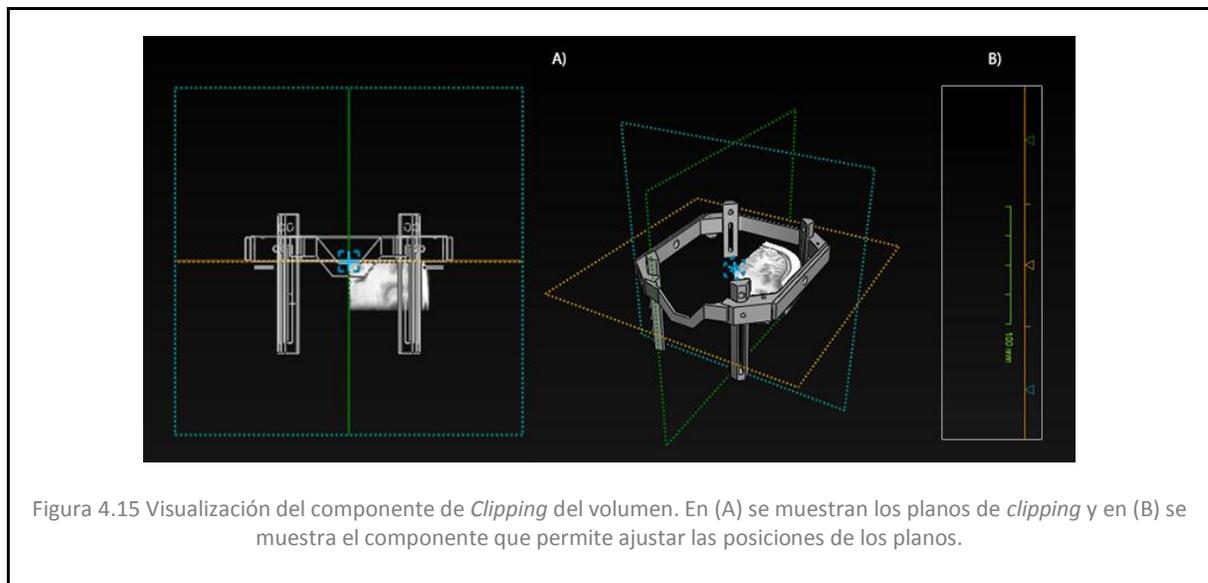
Figura 4.14 Visualización del componente de medición.

Con la finalidad de cumplir con lo anteriormente descrito se diseñó la clase *MeasureTool* la cual posee los siguientes métodos: *Add()* para registrar o agregar una nueva medida al listado de medidas, *Clear()* el cual permite vaciar la lista de medidas registradas y *Draw()* que permite desplegar las medidas en pantalla.

Para registrar una medida el usuario deberá señalar en pantalla la longitud que desea calcular. Para ello, la clase implementa la acción de arrastrar-y-soltar (*drag-and-drop*, en inglés) que consiste en los siguientes tres pasos: primero, presionar y mantener presionado el botón del ratón para empezar el trazo de la regla; segundo, arrastrar el cursor hasta la ubicación deseada y por último dejar de presionar el botón para registrar la medida. A medida que la acción arrastrar-y-soltar se esté ejecutando se desplegará en pantalla una etiqueta con la medida registrada hasta ese momento, de modo que el usuario tenga una mejor referencia de la medición que se está realizando.

El componente también permite registrar medidas en tres listas internas distintas. La finalidad de dicha característica es que se puedan mantener registros separados para los cortes axiales, sagitales y coronales.

La clase *ClipTool* es la encargada de realizar cortes transversales a un elemento 3D haciendo uso del procedimiento de “recorte” (*clipping*, en inglés) definido en el API de *OpenGL*. Esta clase es utilizada en la aplicación VSF como un componente para la visualización interior del volumen desplegado. El procedimiento de *clipping* se encarga de identificar una porción del volumen que se encuentra dentro o fuera de una región tridimensional definida por la intersección de tres planos perpendiculares entre sí, como se aprecia en la **Figura 4.15**. Luego, toda la porción dentro del área definida es mostrada en pantalla y el resto simplemente no es desplegada.

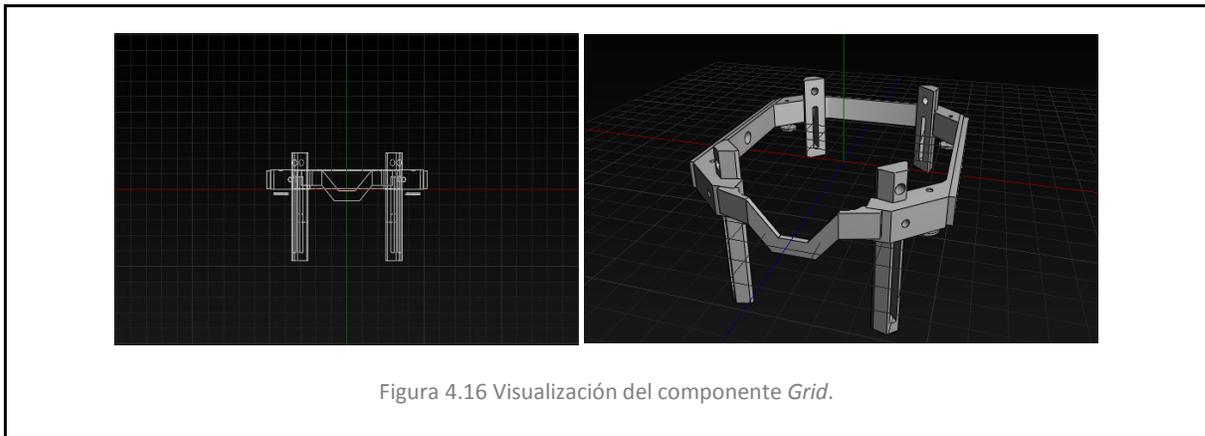


La herramienta está conformada por dos componentes principales. El primer componente corresponde a las tres guías visuales que representan los tres planos perpendiculares entre sí. Estas guías solo indican la posición de los tres planos en el espacio y no permiten la interacción directa con el usuario, en otro sentido, que no es posible interactuar con ellas haciendo uso del ratón o teclado.

El segundo componente está formado por una herramienta que sirve para ajustar las posiciones de los tres planos. Dicha herramienta está compuesta por tres barras de desplazamiento y tres selectores triangulares cuyos colores corresponden a los colores de los planos. Haciendo uso del ratón el usuario puede trasladar hacia arriba o hacia abajo los selectores logrando el desplazamiento de los planos y cambiando las posiciones de corte del volumen.

La clase *ClipTool* cuenta con los siguientes métodos estáticos: *Draw()*, encargado de desplegar los dos componentes principales de la herramienta; *Begin()* y *End()* cuyo propósito es indicarle al componente de *clipping* que todo elemento dibujado entre esas dos instrucciones deben ser “recortadas” transversalmente por los tres planos. Finalmente, la herramienta de *clipping* puede activarse y desactivarse haciendo uso del atributo booleano *IsClippingActivated*.

La clase *Grid* está diseñada para facilitar los desplazamientos del ME. Su función es desplegar una cuadrícula que permita al usuario tener una referencia de las medidas o desplazamientos de los elementos de la escena, ya sea el ME o el volumen desplegado. La cuadrícula utilizada tiene dimensiones de 1.96 mts<sup>2</sup> dividida en sub-cuadrantes de 100 cm<sup>2</sup> que a la vez se subdividen en áreas de 4 cm<sup>2</sup>. En la **Figura 4.16** puede apreciarse el componente *Grid*.



La última clase del paquete «*Graphics*» es *TextRenderer* la cual permite desplegar texto 2D o 3D por pantalla. El diseño de la clase es bastante sencillo, cuenta con un método *Draw()* para realizar el despliegue del texto y los atributos *Color* y *Text* para cambiar el color y el contenido del texto que se despliega. La clase es usada por las clases *RotationTool*, *MovementTool* y *MeasureTool* para desplegar las etiquetas utilizadas por dichos componentes.

#### 4.5.4 Paquete «*Transform*»

En este paquete se encuentran las clases que hacen posible la interacción del usuario con los diferentes componentes contenidos en el paquete «*Graphics*» a través del uso del ratón. El paquete es importado por el paquete «*Graphics*» e importa los paquetes «*Glu*» y «*Math*» para la realización de cálculos matemáticos y el mapeo de coordenadas de un espacio a otro.

La primera clase contenida en este paquete es *Picking*. La función de dicha clase es comprobar si un determinado componente ha sido o no seleccionado con el cursor del ratón. Para lograr tal objetivo se utili-

za el método *GetElement()* que se encarga de leer<sup>6</sup> el valor de color del píxel en el *frame buffer* indicado por la posición del cursor en la ventana.

Luego, la clase determina si el valor de color leído corresponde a alguno de los colores de los elementos de interacción del paquete «*Graphics*» e indica que elemento ha sido seleccionado comunicándose con la clase *Interaction* en donde están contenidos los métodos para el movimiento (*Move()*) y rotación (*Rotate()*) de todos los componentes.

Para poder hacer uso de ambas clases es necesario transformar las coordenadas del cursor expresadas en dos dimensiones a coordenadas 3D (en espacio objeto), y adicionalmente realizar el despliegue de las herramientas de interacción dos veces. El primer despliegue no es mostrado realmente en pantalla y sirve para determinar el elemento seleccionado (clase *Picking*). El segundo despliegue, que si es visible al usuario, sirve para resaltar el elemento seleccionado una vez conocido y permitir la interacción con el mismo (clase *Interaction*).

#### 4.5.5 La clase *Shader*

Los *shaders* son simples programas ejecutados en la Unidad Gráfica de Procesamiento que transforman un vértice (denominado *vertex*) y un píxel (denominado *fragment*) para lograr efectos gráficos de iluminación, sombreado, animaciones, etc.

A grandes rasgos los *shaders* están compuestos por dos secciones principales: los *Vertex Shaders* y los *Fragment Shaders* o *Pixel Shaders*.

Un *Vertex Shader* (*VS*) es una función de procesamiento gráfico que manipula los valores de un vértice en un plano 3D mediante operaciones matemáticas sobre un objeto. Las variaciones que pueden ser diferenciadas en el *VS* son el color, las coordenadas de la textura, la orientación en el espacio o el tamaño del punto. Este programa se ejecuta por cada vértice que conforma la geometría [45].

De manera análoga al *Vertex Shader*, se define un *Fragment Shader* (*FS*) como un pequeño programa que procesa fragmentos y que se ejecuta en la GPU. El nombre de *fragment* viene de que una escena 3D es proyectada en el plano bidimensional donde los puntos son llamados fragmentos. Un fragmento contiene no sólo información de color del punto sino también otros datos como la posición y las coordenadas de la textura por ejemplo. Diversos fragmentos pueden ser unidos para mostrar un píxel en la pantalla [45].

La clase *Shader* permite cargar y ejecutar los *shaders* utilizados en la aplicación VSF. Cuenta con tres métodos importantes: *CreateShaderProgram()* el cual se encarga de crear el *VS* y el *FS* en la tarjeta gráfica; el método *Run()* que se encarga de reemplazar la sección del *hardware* de vídeo que típicamente se denomina *pipeline* de función fija (*Fixed-Function Pipeline* o *FFP*) y hacer uso del *shader* (sección programable del *hardware* gráfico); y *Stop()* que indica que se dejará de utilizar el *shader* y que se hará uso del *pipeline* de función fija.

---

<sup>6</sup> Para obtener el valor de color del píxel en el *frame buffer* se utiliza la función *ReadPixel* del API de *OpenGL* [49].

### 4.5.6 La clase Camera

En *OpenGL* una cámara<sup>7</sup> puede ser definida como un conjunto de puntos y vectores en el espacio tridimensional que determinan la forma como una escena es mostrada en pantalla. Uno de los puntos que conforman la estructura indica la posición de la cámara en sí (*Eye*), otro indica hacia dónde enfoca la cámara (*Target*) y un vector que especifica la dirección hacia arriba de la cámara (*Up*). Esos tres elementos son los necesarios para la implementación de una cámara manipulable y de la cual se pueden obtener las matrices de transformación necesarias para desplegar un mundo 3D en una pantalla 2D. En la **Figura 4.17** pueden apreciarse de manera gráfica los elementos principales que conforman una cámara.

Son necesarias dos matrices: una matriz de visualización (*View Matrix*, en inglés) la cual se encarga de rotar el mundo de acuerdo con la posición 3D de la cámara y una matriz de proyección (*Projection Matrix*, en inglés) la cual permite proyectar en pantalla (2D) la información 3D que se desea desplegar.

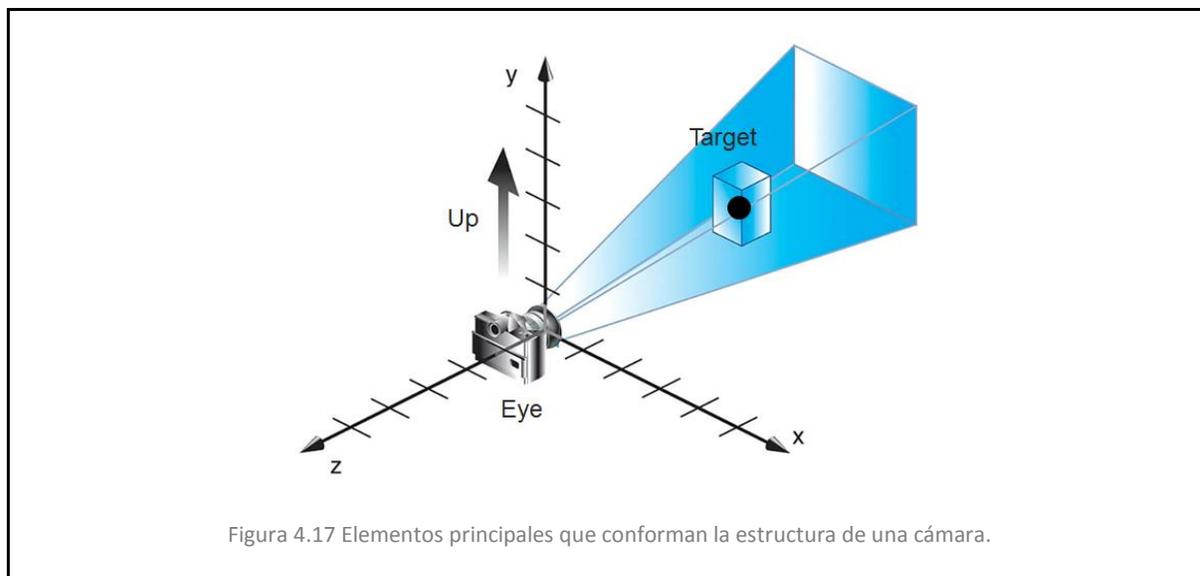
El *toolkit* de *OpenTK* y el API de *OpenGL* cuenta con un conjunto de métodos y operaciones matemáticas que ayudan a realizar lo anteriormente expuesto de manera rápida y sencilla.

La clase *Camera* es una de las clases más importantes de la biblioteca dinámica *UtilityLib*. Sin ella, sería imposible visualizar los diferentes elementos 2D y 3D en pantalla que se despliegan en la aplicación VSF. Cuenta con seis métodos públicos importantes:

- *SetViewport()*: encargado de crear las matrices de transformación necesarias para la visualización y la proyección del mundo en pantalla. Este método es generalmente usado cuando la ventana cambia de tamaño en la aplicación o cuando se desea cambiar de una proyección ortogonal a una perspectiva.
- *Look()*: se encarga de crear la matriz de visualización utilizando los puntos *Eye*, *Target* y el vector *Up*. Este método debe ser invocado antes de realizar el despliegue de los elementos en pantalla con el fin de posicionar la cámara.
- *Pan()*: se encarga de realizar desplazamientos verticales y/u horizontales a la posición de la cámara. Recibe como parámetros de entrada las coordenadas del cursor en pantalla.
- *RotateWithMouse()*: permite que la posición de la cámara rote con un ángulo de 360 grados en todos los ejes, siempre centrada en un punto fijo (*Target*). Este método al igual que el anterior recibe como parámetros de entrada las coordenadas del cursor en pantalla y utiliza el atributo *Sensitivity* para controlar la sensibilidad o velocidad de las rotaciones.
- *Reset()*: es el método encargado de reiniciar los valores con los que por defecto la cámara fue creada.
- *Set()*: dentro de sus varias funcionalidades, este método se encarga de cambiar la proyección de la cámara (ortogonal o perspectiva) y de variar el ángulo que debe rotar la posición de la cámara (*Eye*). Los ángulos de variación están predefinidos para cubrir seis puntos principales: superior, inferior, izquierda, derecha, anterior y posterior. Los cambios entre una posición de visualización predefinida (*View Set*) y otra se realizan interpolando el valor de los ángulos paulatinamente en el tiempo para lograr una animación en donde puede apreciarse la rotación de la cámara.

---

<sup>7</sup> La estructura *Camera* no está definida como parte de las funcionalidades del API de *OpenGL*. La estructura debe ser implementada usando las facilidades que el API de *OpenGL* provee.



Si bien no existe una función para realizar acercamientos o alejamientos de la cámara, el atributo *Zoom* es el encargado de controlar esta funcionalidad. Cada vez que dicho atributo es modificado los valores de la clase son recalculados para acercar o alejar la cámara.

## 4.6 La aplicación *Virtual Stereotactic Frame*

La aplicación interactiva que permite mostrar las funcionalidades implementadas y anteriormente explicadas. Su propósito es permitir al usuario interactuar con el sistema de manera gráfica haciendo uso de la entrada estándar de teclado y ratón.

La aplicación puede ser definida como un sistema que permite simular en un ambiente virtual la colocación del ME en el cráneo, haciendo uso de las imágenes diagnósticas del paciente obtenidas antes del tratamiento.

La interfaz gráfica de la aplicación está dividida en cinco grandes secciones. Cada una de estas secciones está agrupada en sub-elementos según sus características, funcionalidades y similitudes. Por ejemplo, todos los comandos para el tratamiento y manipulación del volumen se encuentran agrupados en el sub-menú *Volume*. Como puede verse en la **Figura 4.18**, las cinco secciones principales son las siguientes:

- **Barra de título y de Menús:** corresponde a la fuente de comandos más común. En la barra de menús se encuentran todos los métodos y funcionalidades de la aplicación. Ambas barras se encuentran en la parte superior de la ventana de la aplicación.
- **Barra de herramientas principal:** es una barra compuesta por una serie de botones con iconos que proveen a la aplicación de acceso en un solo clic a las funciones más importantes del sistema. Se encuentra ubicada debajo de la barra de menús.

- **Panel de vistas:** está compuesta por cuatro vistas<sup>8</sup> (*Viewports*, en inglés) principales que permiten percibir el contenido de la escena desde diferentes ángulos. Al iniciar la aplicación, por defecto se visualiza la escena desde arriba, izquierda, frontal y personalizada. Esta sección ocupa gran parte del centro de la aplicación.
- **Panel de configuración para el Marco Estereotáctico:** localizado a la derecha de los cuatro *Viewports* principales, posee tres pestañas tabuladas y marcadas con iconos que abren o permiten visualizar diferentes paneles. Cada panel incluye una serie de parámetros y configuraciones que cambian dependiendo de la pestaña seleccionada.
- **Barra de herramienta inferior:** posee una colección de controles, mensajes informativos y botones diversos. Está localizada en la parte inferior de la aplicación.

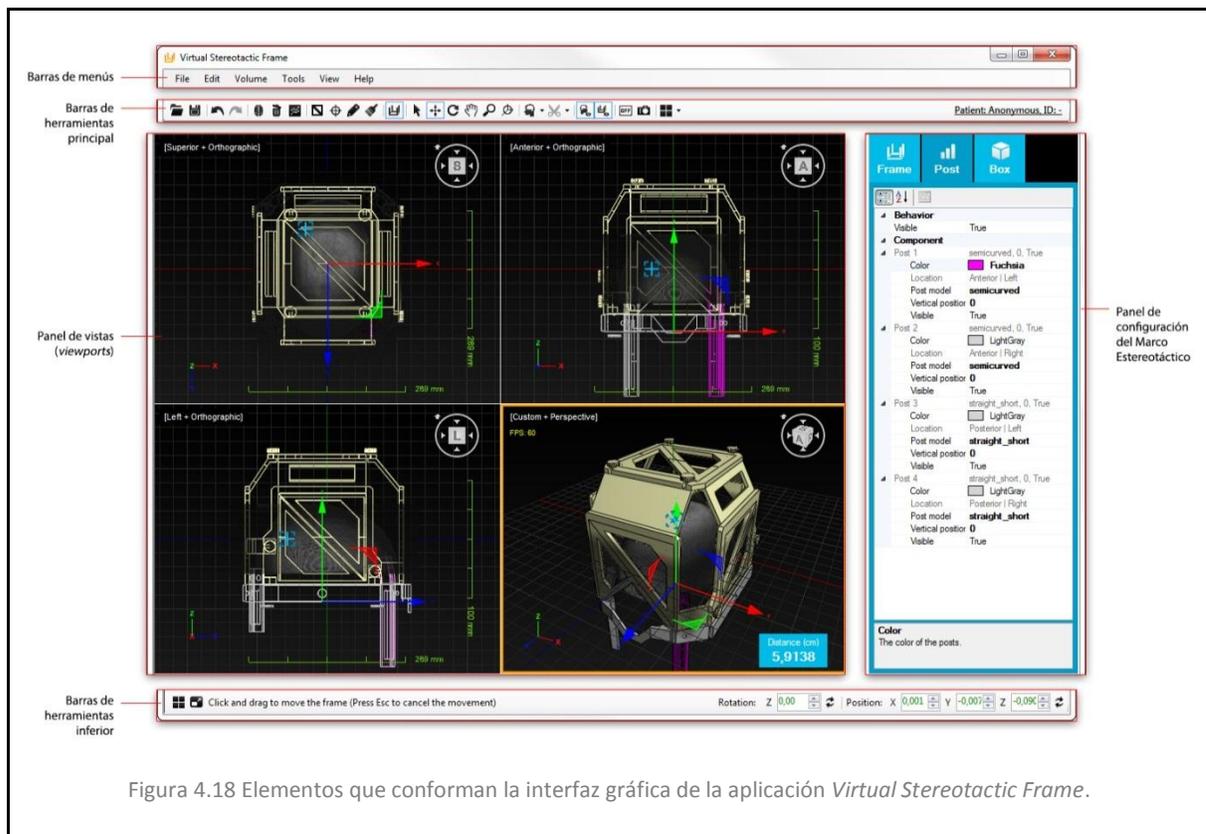


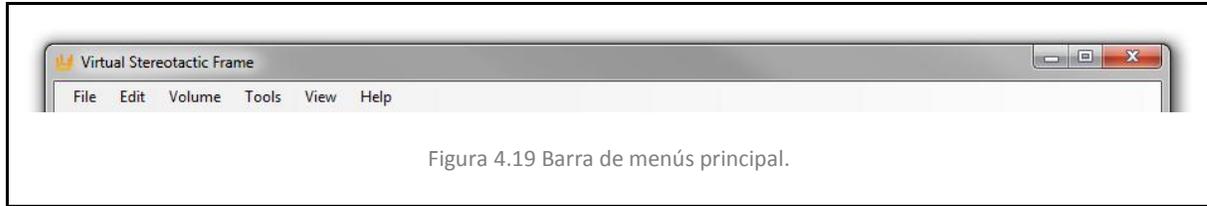
Figura 4.18 Elementos que conforman la interfaz gráfica de la aplicación *Virtual Stereotactic Frame*.

#### 4.6.1 La barra de menú principal

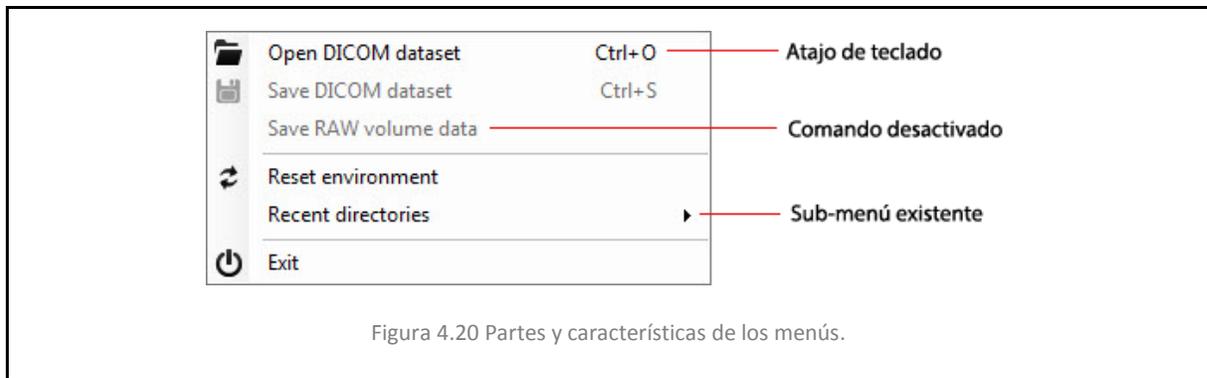
Los menús desplegables ubicados en la parte superior de la ventana principal incluyen muchas de las funcionalidades disponibles en la aplicación VSF. Muchos de los comandos disponibles en los menús tienen su correspondiente botón en la barra de herramientas principal y poseen atajos (*shortcuts*, en inglés) de teclado. Para ejecutar un comando ubicado en alguno de los menús, puede hacerse clic sobre él utilizando el cursor del ratón, puede hacerse clic en su correspondiente botón en la barra de herramientas (sí posee) o utilizarse su atajo de teclado.

<sup>8</sup> Desde este punto en adelante se hará mención de las vistas con su correspondencia en inglés *Viewports* o *Viewport* en singular.

En la **Figura 4.19** puede apreciarse la barra de menús principal la cual contiene las siguientes opciones: Archivo (*File*), Edición (*Edit*), Volumen (*Volume*), Herramientas (*Tools*), Ver (*View*) y Ayuda (*Help*). Las opciones son invariantes durante la ejecución del programa y siempre están visibles y disponibles para el usuario.



No todos los comandos de los menús están siempre disponibles. Si un comando no está disponible en un momento dado, éste se mostrará con un color más claro. Dependiendo del estado del sistema los diferentes comandos se activarán y desactivarán de los menús. Por ejemplo, antes de cargar un volumen los comandos “*Save DICOM dataset*” y “*Save RAW volumen data*” del menú File, se encuentran desactivados, y son habilitados solo después de que un volumen haya sido cargado en el sistema. En la **Figura 4.20** se muestran todas las partes y características de un menú.



## 4.6.2 Menús principales a detalle

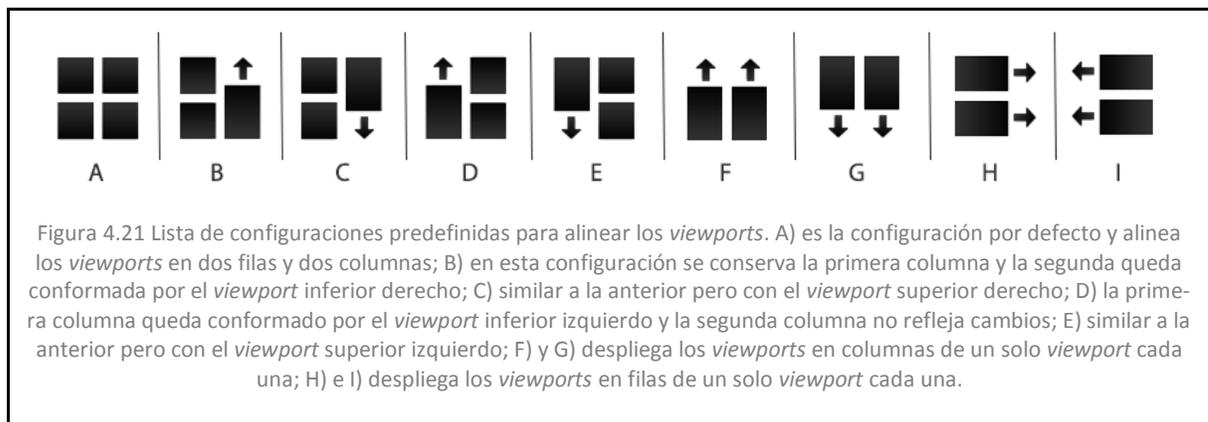
A continuación se lista el contenido de cada menú y se detalla la funcionalidad de cada uno de los comandos.

- **El menú *File*:** en este menú se encuentran los comandos para cargar (*Open DICOM dataset*) y guardar (*Save DICOM dataset* y *Save RAW volumen data*) volúmenes al sistema, reiniciar el ambiente (*Reset environment*), abrir directorios o volúmenes recientes (*Recent directories*) y salir de la aplicación (*Exit*).
- **El menú *Edit*:** contiene los comandos de deshacer y rehacer (*Undo* y *Redo*, respectivamente) una acción en el sistema<sup>9</sup>, mostrar u ocultar la información del paciente en la ventana principal (*Show/Hide Patient*) y cambiar el color de fondo para los *viewports* (*Background color*).
- **El menú *Volume*:** en esta categoría se encuentran los comandos para abrir el visualizador de imágenes DICOM (*DICOM Viewer*), descargar un volumen del sistema (*Unload dataset*), visualizar la

<sup>9</sup> Solo válido para las transformaciones de traslación y rotación del Marco Estereotáctico.

función de transferencia para ajustar la transparencia del volumen (*Transfer Function*), abrir la ventana de configuración para las marcas fiduciales (*Fiducial Marks Settings*), habilitar o deshabilitar la posibilidad de salvar las imágenes DICOM como imágenes de captura secundaria (*Save DICOM as Secondary Capture*), mostrar u ocultar las guías de las marcas fiduciales (*Show/Hide Fiducial Marks*), auto-alinear el centro de las marcas fiduciales y el ME con el centro de la lesión (*Auto-alignment with the target*), generar las marcas fiduciales (*Generate Fiducial Marks*) y remover las marcas fiduciales (*Remove Fiducial Marks*).

- **El menú *Tools*:** en esta opción se agrupan los comandos para la interacción con los *Viewports* y el ME. Dentro de los comandos disponibles están: mostrar u ocultar el panel de configuración del ME (*Stereotactic Frame Settings*), el comando de selección (*Selection*) usado para no permitir rotaciones, traslaciones o desplazamientos de la cámara; movimiento del ME (*Move*) usado para hacer desplazamientos del ME, rotación usada para realizar rotaciones del ME (*Rotate*), desplazamientos verticales y horizontales de la cámara (*Pan*), alejar o acercar los elementos de la escena (*Zoom*), hacer rotaciones de la cámara (*Orbit*), activar o desactivar los planos de “recorte” del volumen (*Toogle Clipping Planes*), los comandos para la visualización de los planos centrales del volumen (*Central Cuts*), encender o apagar las actualizaciones en tiempo real de los *viewports* (*Real time update*), y finalmente el comando para capturar pantalla (*Take snapshot*).
- **El menú *View*:** en este menú se encuentran las diferentes opciones para alinear los cuatro *viewports* de la aplicación. Existen un total de nueve configuraciones predefinidas en las cuales se pueden mostrar cuatro, tres y hasta dos *viewports* en pantalla. En la **Figura 4.21** se pueden apreciar los diferentes iconos de todas las configuraciones de vista posibles.
- **El menú *Help*:** contiene solo dos comandos. El primero de ellos permite abrir la ventana de bienvenida de la aplicación (*Welcome Screen*) y el segundo permite abrir la ventana de “Acerca de la aplicación” (*About VSF*) en donde se aprecian los datos del autor de la aplicación.



### 4.6.3 La barra de herramientas principal

Localizada justo debajo de la barra de menús principal se encuentra la barra de herramientas principal. Como se mencionó anteriormente ésta barra está compuesta por una serie de botones que proveen a la aplicación de acceso rápido a las funciones más importantes y más utilizadas del sistema.

La barra cuenta con tres tipos de botones diferentes. Los botones tradicionales que activan o ejecutan un comando al hacer clic sobre ellos; los botones de dos estados (*Toogle*, en inglés) que además de eje-

cutar una acción en el sistema brinda la posibilidad de deshacer la acción o retornar al estado inicial (dichos botones se identifican desplegando un cuadro alrededor del icono como puede apreciarse en la **Figura 4.22**); y por último se encuentran los botones con sub-menús los cuales despliegan un menú al hacer clic sobre ellos.



En la **Tabla 4.1** se listan todos los comandos disponibles en la barra de herramientas principal de la aplicación VSF y se describen sus funciones o acciones en el sistema.

El último elemento de la barra de herramientas se encuentra alineado a la derecha de la misma, separado de la lista de botones. En este elemento se despliega los datos básicos del paciente<sup>10</sup> y solo está disponible cuando un volumen es cargado. Los datos mostrados son los nombres y apellidos del paciente y su identificación (cédula de identidad para este trabajo). Adicionalmente, el elemento funciona como un botón que al ser pulsado abre la ventana para la visualización de las imágenes DICOM que conforman el volumen.

#### 4.6.4 Control y configuración de los cuatro Viewports

Los cuatro *viewports* conforman la sección más grande de toda la aplicación y proveen diferentes maneras de visualizar todos los elementos de la escena. Cada uno de los *viewports* es perfectamente configurable por el usuario y son independientes entre ellos (en términos de configuración). Esta sección de la aplicación es la más importante de todas.

Un *viewport* puede definirse como una ventana pequeña en donde se despliega la escena vista desde cierta perspectiva. Los cuatro *viewports* poseen numerosas opciones que proveen a la aplicación de diferentes maneras de visualizar la escena.

La primera opción de configuración es la posibilidad de ver la escena con una vista perspectiva u ortográfica (también llamada ortogonal).

La vista ortográfica se caracteriza porque los elementos de la escena son visualizados de forma directa desde un lado o “cara”. Esto revela la vista de la escena desde un único plano, por esa razón los objetos muestran sus medidas reales. Adicionalmente, cuando se observa directamente la cara frontal de un objeto desplegado se distinguen el ancho y altura del mismo en dos dimensiones, pero no la tercera dimensión, la profundidad. Cada vista ortográfica proporciona dos de las tres dimensiones principales.

<sup>10</sup> Por propósitos académicos asociados a este Trabajo Especial de Grado se ha reservado la información de los pacientes con los que se han realizado las pruebas del sistema. Todos los pacientes permanecen anónimos.

Botón	Nombre	Descripción
	<i>Open DICOM dataset</i>	Abre el cuadro de diálogo que permite seleccionar el directorio donde se encuentra el conjunto de imágenes DICOM a cargar.
	<i>Save DICOM dataset</i>	Abre el cuadro de diálogo para seleccionar la ubicación en donde se desea guardar el volumen como un conjunto de imágenes DICOM.
	<i>Undo y Redo</i>	Comandos para deshacer y rehacer los movimientos y rotaciones del ME.
	<i>DICOM Viewer</i>	Abre el visor de imágenes DICOM.
	<i>Unload dataset</i>	Descarga el volumen actual de la aplicación.
	<i>Transfer Function</i>	Abre la ventana para la configuración de la función de transferencia para ajustar la transparencia del volumen.
	<i>Show/Hide Fiducial Marks</i>	Muestra y oculta las guías visuales de las marcas fiduciales.
	<i>Auto-alignment with the target</i>	Auto-alinea el centro de las marcas fiduciales y el ME con el centro de la lesión.
	<i>Generate Fiducial Marks</i>	Genera las marcas fiduciales en el volumen.
	<i>Remove Fiducial Marks</i>	Remueve las marcas fiduciales en el volumen (si las hay).
	<i>Stereotactic Frame Settings</i>	Muestra y oculta el panel de configuración del ME.
	<i>Selection</i>	No permite hacer movimientos ni rotaciones en el ME, además de realizar cambios en la cámara.
	<i>Move</i>	Permite realizar traslaciones al ME.
	<i>Rotate</i>	Permite realizar rotaciones al ME.
	<i>Pan</i>	Permite desplazar la cámara horizontal y verticalmente.
	<i>Zoom</i>	Permite alejar o acercar la cámara.
	<i>Orbit Camera</i>	Permite rotar la cámara.
	<i>Toggle Clipping Planes</i>	Activa y desactiva los planos de “recorte” del volumen.
	<i>Central Cuts</i>	Muestra y oculta los planos centrales del volumen.
	<i>Show/Hide Patient</i>	Muestra y oculta el volumen.
	<i>Show/Hide Stereotactic Frame</i>	Muestra y oculta el ME.
	<i>Indicate whether the updating of the changes in all viewports is in realtime</i>	Activa y desactiva la actualización en tiempo real de los cuatro <i>viewports</i> .
	<i>Take snapshot</i>	Captura en imagen el contenido del <i>viewport</i> seleccionado.
	<i>Layout</i>	Determina la configuración de alineación de los <i>viewports</i> .

Tabla 4.1 Lista de comandos encontrados en la barra de herramientas principal de la aplicación VSF.

En contraparte, la visión perspectiva simula la manera como nuestros ojos realmente funcionan. Todos los puntos convergen a una única posición que converge en la distancia, lo cual recrea la profundidad en la escena y produce el efecto de reducción de tamaño en los objetos. En la **Figura 4.23** se pueden apreciar las diferencias entre estas dos maneras de visualización.

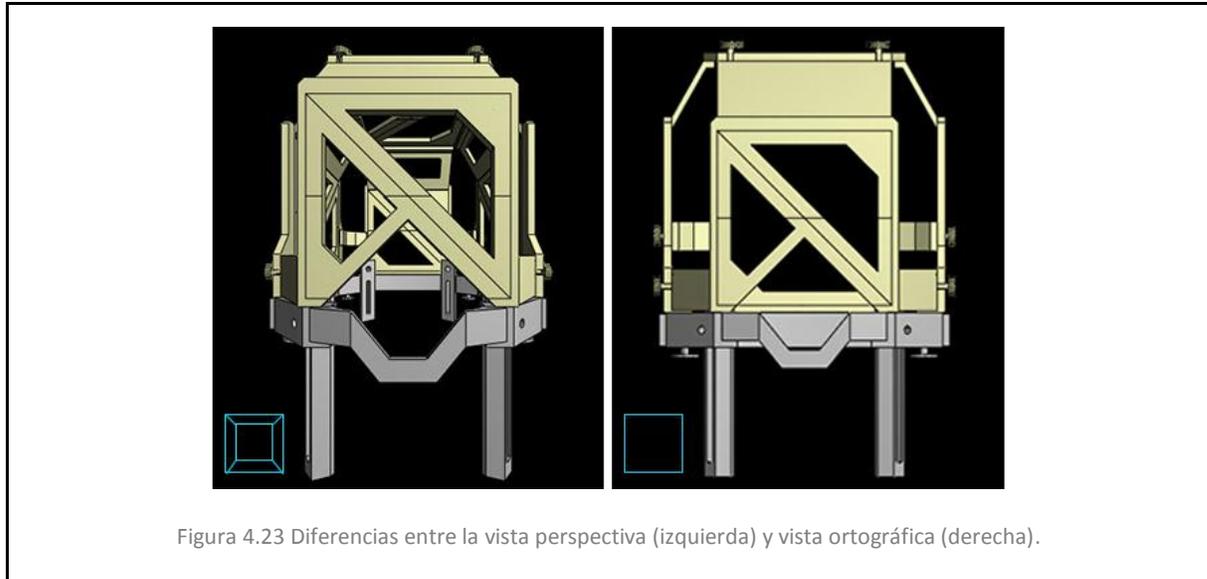


Figura 4.23 Diferencias entre la vista perspectiva (izquierda) y vista ortográfica (derecha).

Al iniciar la aplicación los *viewports* muestran tres vistas ortográficas y una perspectiva. Las tres vistas ortográficas se establecen en los dos *viewports* superiores y en el *viewport* inferior izquierdo, con configuraciones de vista superior, frontal e izquierda de la escena, respectivamente. El último *viewport* posee una vista perspectiva enfocando un lado del ME.

En la esquina superior izquierda de cada *viewport* se muestra su respectiva configuración en todo momento. Adicionalmente, solo un *viewport* puede estar activo a la vez, y es conocido como el *viewport* activo. En esos casos un borde de color naranja se dibujará alrededor del *viewport* indicando que se encuentra activo.

Conjuntamente con el borde de selección del *viewport* activo, son mostradas dos etiquetas informativas. La primera de ellas se ubica en la esquina superior izquierda, debajo de la configuración de *viewport*, e indica la cantidad de cuadros por segundo que son proyectados en pantalla. La segunda etiqueta, ubicada en la esquina inferior derecha, representa la distancia euclidiana entre el centro de la lesión y el centro de la caja radiopaca o el centro de las marcas fiduciales.

Otros elementos visibles en el *viewport* son: el *ViewCube*, el cual permite el cambio de vista de manera rápida y sencilla, ubicado en la esquina superior derecha; las reglas que indican las medidas del volumen, ubicada a la derecha y abajo del *viewport*; y la visualización de los ejes cartesianos tridimensionales, ubicados en la esquina inferior izquierda. En la **Figura 4.24** se pueden apreciar todos los elementos que conforman el *viewport*.

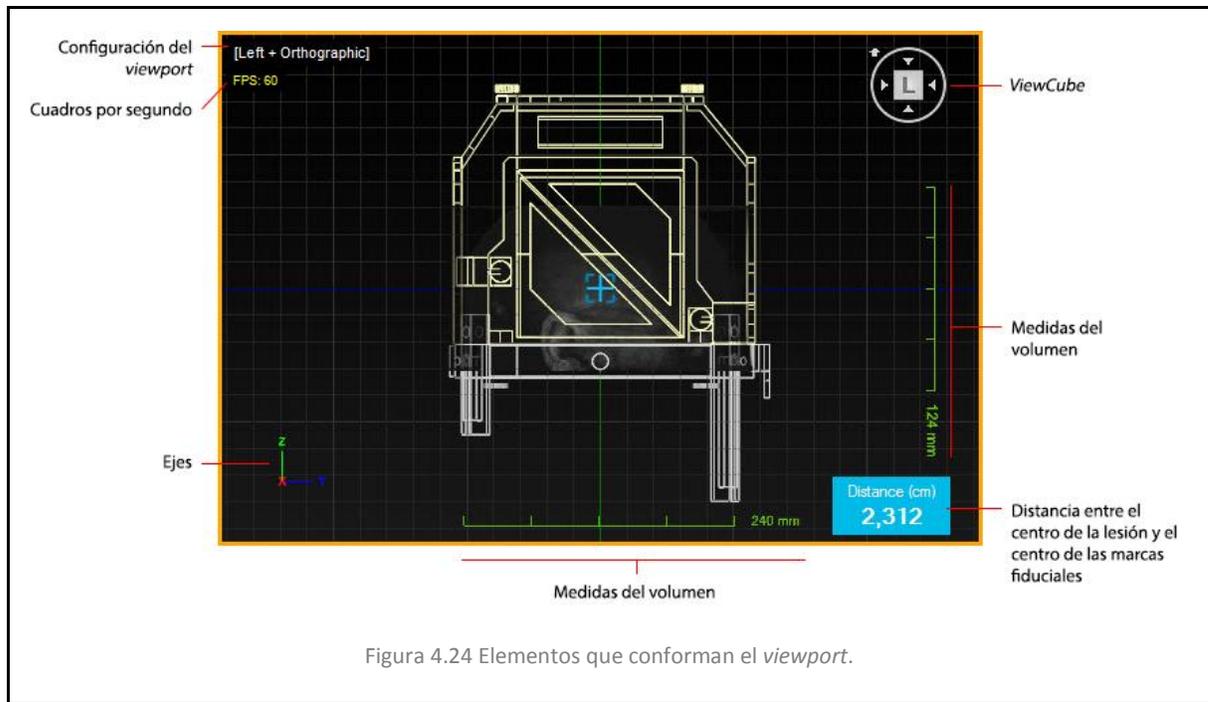


Figura 4.24 Elementos que conforman el viewport.

Otra manera de configurar el viewport es haciendo uso del menú contextual que se despliega al presionar el segundo botón del ratón sobre cualquier área del viewport activo, como puede apreciarse en la **Figura 4.25**. En este menú se despliegan todos los comandos básicos para la configuración del viewport, tales como: opción de vista perspectiva u ortográfica (*Perspective* y *Orthographic*), las vistas predefinidas (*Superior*, *Inferior*, *Left*, *Right*, *Anterior*, *Posterior* y *Custom*), el comando para desplegar los modelos con bordes o solo desplegar los bordes (*Realistic + Edges* y *Edges*), el comando para mostrar y ocultar el rejilla (*Grid*), la opción para maximizar el viewport (*Maximize*) y la opción para reiniciar la cámara a su estado original (*Reset Camera*).

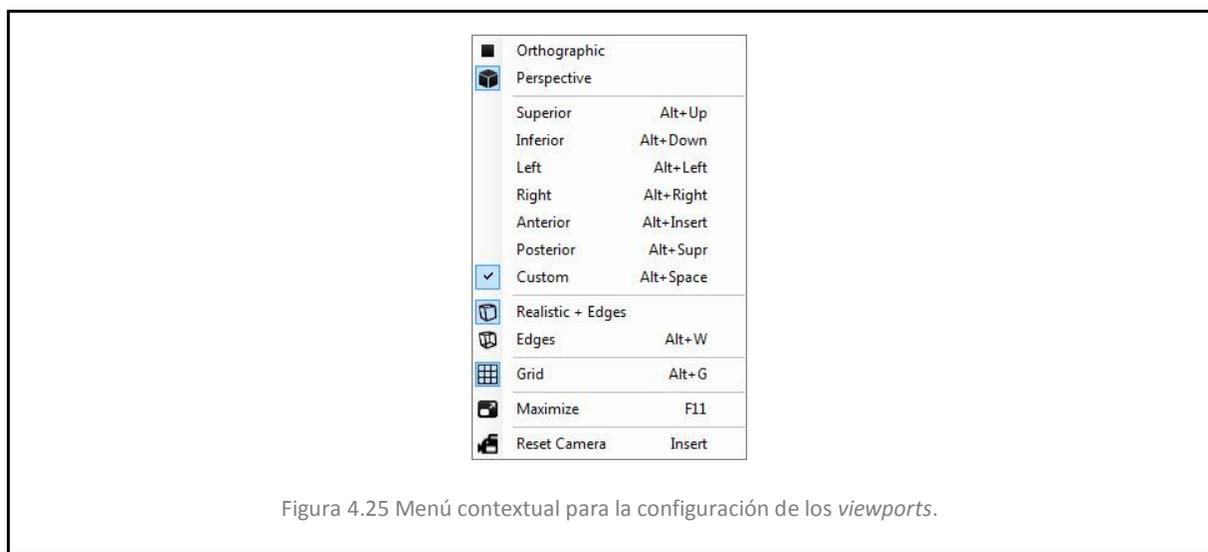


Figura 4.25 Menú contextual para la configuración de los viewports.

Es importante destacar que el tamaño, ancho y alto, de todos los *viewports* es configurable por el usuario. Esto se logra redimensionando cada uno de ellos haciendo clic con el cursor del ratón sobre el borde de separación entre un *viewport* y otro, y luego arrastrando el cursor hasta obtener el tamaño de alto o ancho deseado.

#### 4.6.5 El panel de configuración del Marco Estereotáctico

Además de los *viewports* el panel de configuración del ME es una de las secciones más importantes de todo el sistema. El panel se localiza en la parte derecha de la ventana, justo al lado de la sección de los *viewports* y no se encuentra visible al iniciar la aplicación. Para ocultar o mostrar el panel de configuración se utiliza el comando ubicado en la barra de herramientas o en la barra de menús, como se explica en las secciones anteriores (ver **Tabla 4.1**).

En este panel se encuentran todos los parámetros necesarios para la configuración de la visualización del ME y todas sus piezas. Está dividido en tres sub-paneles, cada uno accedido a través de pestañas identificadas con iconos ubicadas en la parte superior, como puede verse en la **Figura 4.26**. Esos tres sub-paneles son *Frame*, *Post* y *Box*.

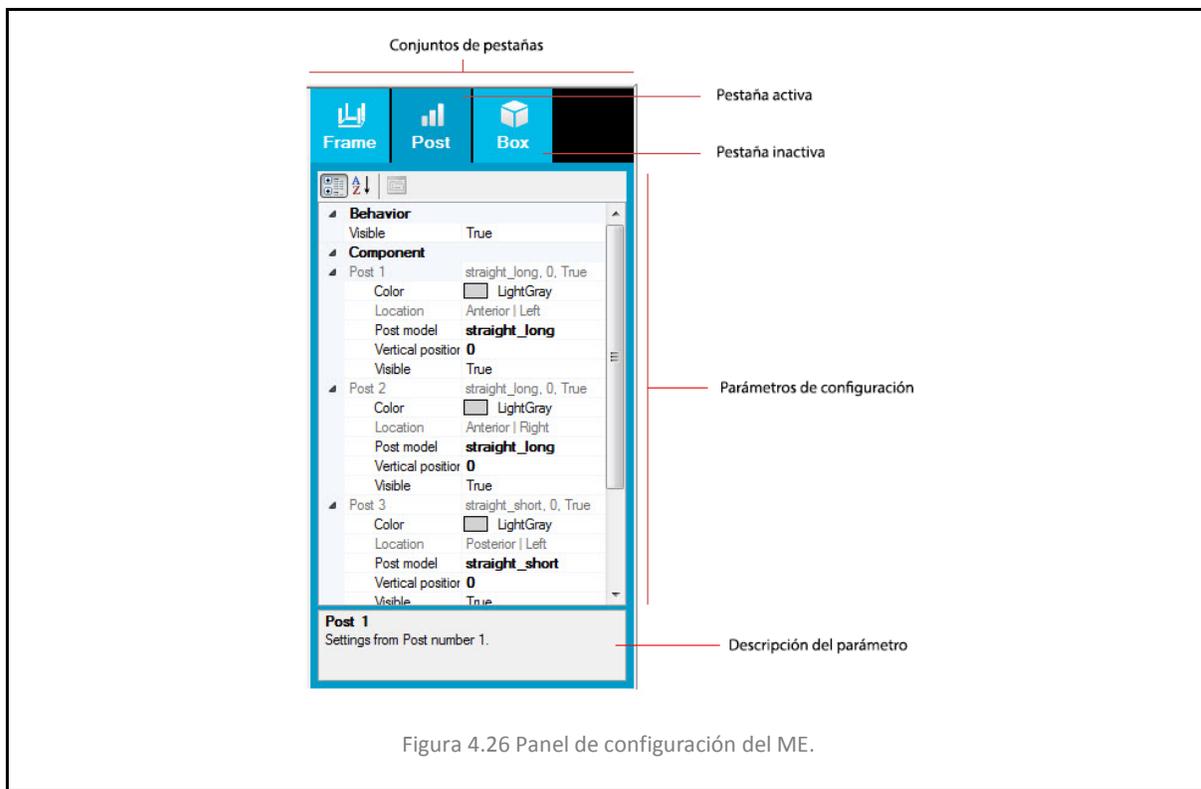


Figura 4.26 Panel de configuración del ME.

Cuando se hace clic sobre alguna de las pestañas de inmediato se aprecia que la misma cambia de color. Esto indica que la pestaña ha sido seleccionada y sirve para diferenciarla de aquellas no seleccionadas. Adicionalmente al cambio de color, en la parte central del panel se despliega una lista con todos los parámetros.

tros configurables para esa sección o elemento. La cantidad y tipos de parámetros varían según la pestaña seleccionada.

Todos los parámetros de configuración se dividen en tres grupos principales: parámetros de comportamiento (*Behavior*), parámetros de apariencia (*Appearance*) y componentes (*Components*). Básicamente, todos los parámetros de comportamiento determinan si un componente del ME debe ser mostrado o no (*Visible*).

Los parámetros de apariencia engloban el color con el que se desea desplegar el elemento y una lista de posibles modelos a ser desplegados. Por ejemplo, en la sección de configuración de los marcos (*Frames*) existe un parámetro llamado *Frame model*, el cual permite seleccionar un modelo dentro de una lista de diferentes marcos estereotácticos disponibles en la aplicación.

Por último, están los parámetros de componentes (*Components*) disponibles en la sección de configuración de los postes (pestaña *Post*). Estos parámetros agrupan la configuración de elementos que forman parte de un conjunto más grande. En este caso existen cuatro postes diferentes que pueden ser configurados independientemente, pero que forman parte del conjunto total de postes. Cada poste está identificado con su nombre y ubicación en el ME, y además de tener los parámetros de comportamiento y apariencia disponibles en todos los elementos, poseen un parámetro adicional llamado *Vertical position*<sup>11</sup> el cual determina la posición vertical del poste con referencia al ME.

Cada vez que el usuario selecciona un parámetro de la lista, en la parte inferior del panel se despliega una breve descripción de la función de dicho parámetro. Adicionalmente, cada vez que se registra un cambio en alguno de los parámetros de configuración, dicho cambio es reflejado automáticamente en el despliegue del ME.

#### 4.6.6 Barra de herramientas inferior

La última sección importante de la ventana principal de la aplicación VSF es la barra de herramientas inferior. Como sugiere su nombre, la barra se encuentra ubicada en la parte inferior de la ventana, debajo de los *viewports* y del panel de configuración del ME. La barra contiene un conjunto de controles fijos, disponibles y visibles en todo momento. La barra de herramientas puede ser apreciada en la **Figura 4.27** e incluye los siguientes controles:

- **Configuración de los *viewports*:** esta sección está ubicada en la parte izquierda de la barra. En ella se encuentran dos botones para configurar los *viewports* de manera rápida y en un solo clic. El primer botón se encarga de restituir la alineación de los cuatro *viewports* a una matriz de dos filas y dos columnas, además de restaurar los valores de ancho y alto para cada *viewport*. El segundo botón se encarga de maximizar el *viewport* activo.
- **La barra de estados:** proporciona información valiosa, tal como mensajes de uso para el comando seleccionado, uso de atajos de teclado, estado del sistema, etc.
- **Valor de rotación del ME:** a diferencia de los elementos anteriores, esta sección está compuesta por una caja de texto en donde se visualiza y edita el valor de rotación del ME expresado en grados.

---

<sup>11</sup> Posición expresada en milímetros.

Aunado a la caja de texto, se encuentra un botón ubicado al lado derecho cuya función es asignar un valor de rotación de cero grados (el valor por defecto).

- **Valores de traslación del ME:** similar a la sección anterior. La diferencia es que esta sección maneja tres cajas de texto que representan las tres coordenadas de traslación del ME. Cuenta también con un botón de reinicio que posiciona al ME en el origen de los planos cartesianos (0, 0, 0) (los valores por defecto).

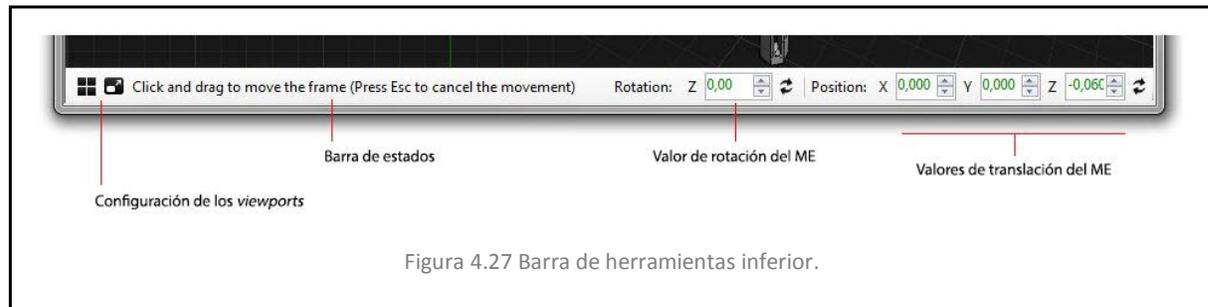


Figura 4.27 Barra de herramientas inferior.

#### 4.6.7 Función de transferencia

La mayoría de las veces cuando se trabaja con despliegue de volúmenes en el área de medicina es necesario visualizar tejidos o zonas específicas del cuerpo humano. Para lograr esto se utiliza una función de transferencia. Una función de transferencia se encarga de asignar propiedades ópticas, como color y transparencia, a las muestras del volumen desplegado. En el caso de funciones de transferencias de una dimensión, las cuales son abarcadas en este trabajo, dicho procedimiento se logra asignando valores de color (rojo, verde, azul y transparencia; o RGBA por sus siglas en inglés) a cada iso-valor del volumen utilizando alguna función de correspondencia.

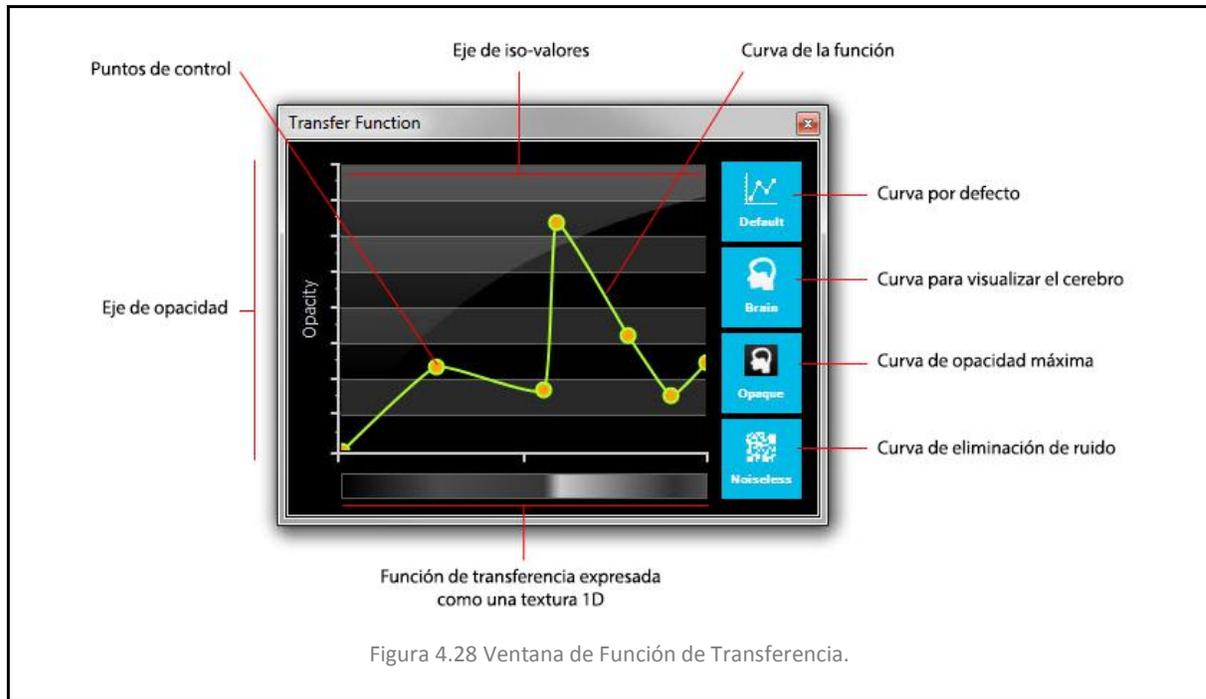
Una función de transferencia puede ser diseñada para visualizar capas de piel, huesos u otros tejidos internos de dos maneras diferentes. La primera es definiendo de forma manual cada valor de correspondencia entre un iso-valor y un color RGBA, lo cual deriva en un proceso tedioso y poco práctico, por lo que no es considerado como una solución factible en este trabajo. La segunda manera es diseñando un control interactivo que permita definir de manera rápida, práctica y lo más importante, en tiempo real, cualquier función de transferencia.

En base con lo anteriormente expuesto, existe un componente para controlar la función de transferencia, el cual se muestra en la **Figura 4.28**, y puede ser accedido a través de la barra de herramientas principal o la barra de menús.

La ventana está compuesta por dos secciones principales: el área de botones de configuración por defecto, ubicada en la parte derecha, y la sección interactiva para controlar la función de transferencia. La función de transferencia está conformada por una serie de puntos de control, representados con círculos de color naranja; la gráfica o curva de la función, representada con líneas verdes; y una textura unidimensional en donde se puede apreciar el resultado de la configuración de la función.

Los puntos de control pueden ser agregados, eliminados y repositionados por el usuario haciendo uso del cursor del ratón. Para agregar un nuevo punto de control basta con hacer clic en cualquier sección del plano cartesiano de la ventana. Para eliminar un punto agregado basta con hacer clic sobre él con el se-

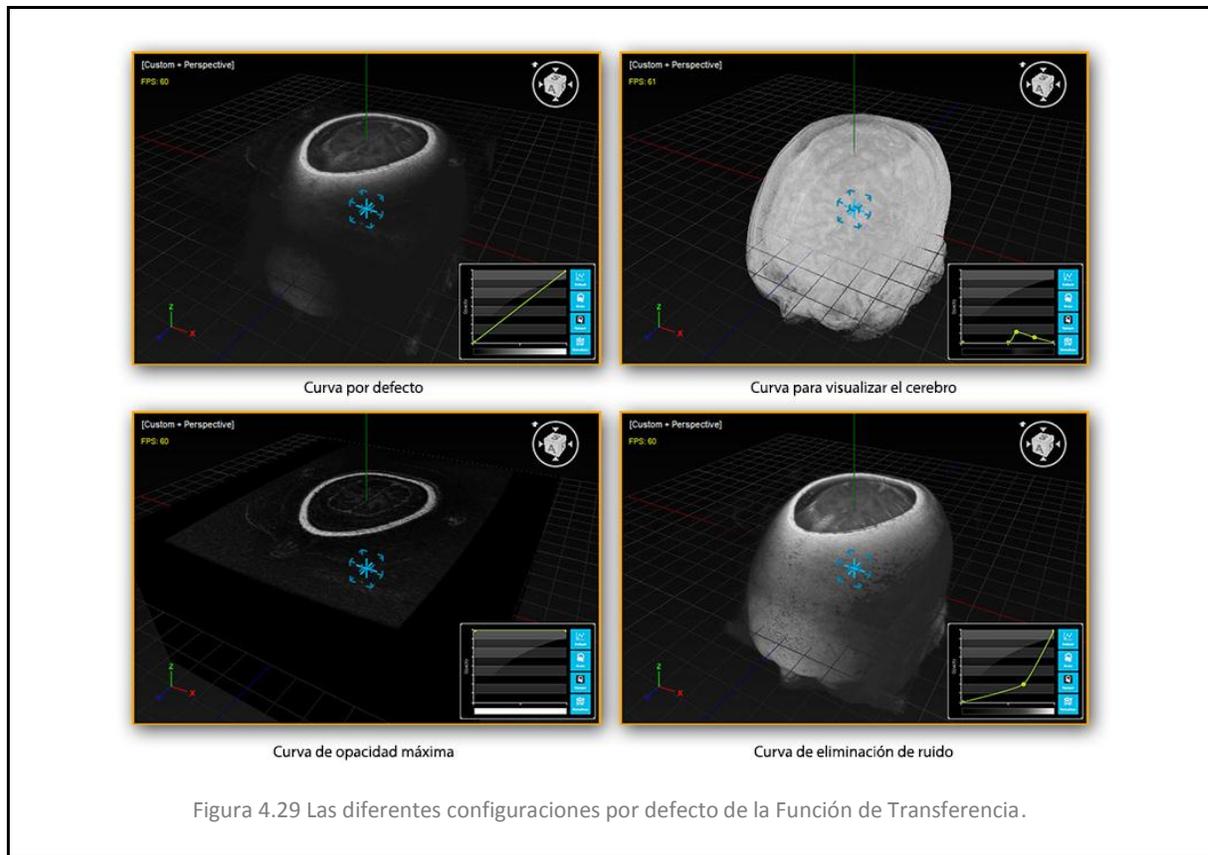
gundo botón del ratón. Para reubicarlo, se debe hacer clic sobre el punto a mover y sin dejar de pulsar el botón del ratón se desplaza el punto hasta la posición deseada. El primer y último punto de control no pueden ser eliminados y solo pueden desplazarse verticalmente.



La función de transferencia manejada en la aplicación no asigna colores a las muestras del volumen, solo controla sus valores de transparencia. Por lo tanto el eje de las ordenadas de la función representa la opacidad de la muestra y el eje de las abscisas representa los iso-valores del volumen.

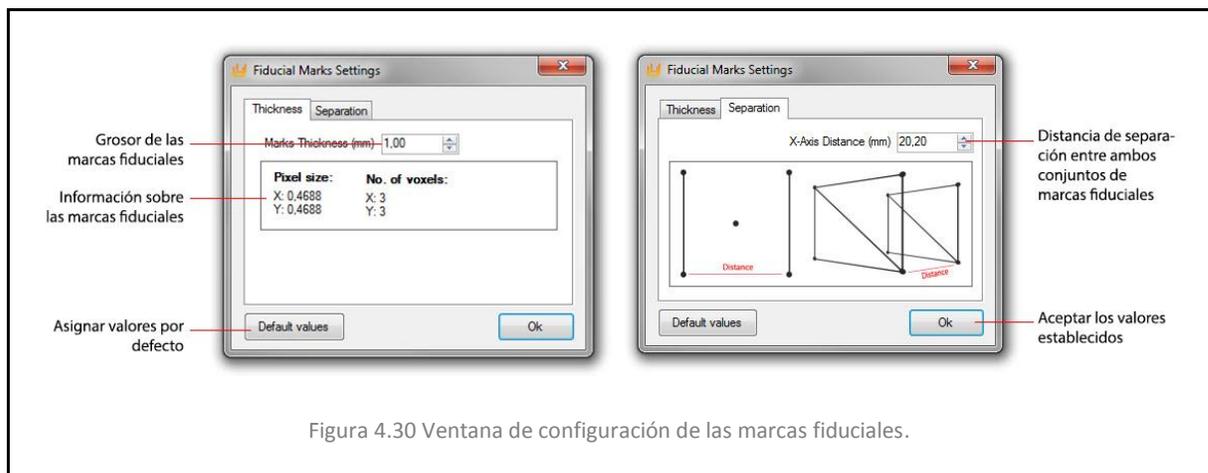
Los cuatro botones ubicados a la derecha, representan pre-configuraciones de diferentes funciones de transferencia. Las pre-configuraciones disponibles se pueden ver en la **Figura 4.29** y son las siguientes:

- **Curva por defecto:** corresponde a una asignación lineal entre los valores de transparencia y los iso-valores. Es la función de transferencia usada por defecto al iniciar la aplicación.
- **Curva para visualizar el cerebro:** las asignaciones de los puntos de control en esta función de transferencia están colocados de tal manera que la capa de hueso del cráneo se hace transparente permitiendo ver a través de él y mostrar el tejido del cerebro.
- **Curva de opacidad máxima:** permite visualizar el volumen con su máxima opacidad, mostrándolo como un conjunto de texturas apiladas formando una caja negra. Esta función resulta útil para realizar cortes del volumen con el comando *Clipping Planes*.
- **Curva de eliminación de ruido:** la función intenta eliminar todo el ruido presente alrededor del cráneo.



#### 4.6.8 Ventana de configuración de las marcas fiduciales

Esta ventana permite configurar las propiedades para la generación de las marcas fiduciales. Está compuesta por dos pestañas, un botón para aceptar los cambios efectuados y otro para asignar los valores por defecto. En la **Figura 4.30** se muestra el contenido de ambas pestañas y todos los elementos que contiene la ventana.



En la primera de las dos pestañas se configura el grosor en milímetros de las marcas fiduciales. Por defecto el valor utilizado es un milímetro pero puede ser cambiado escribiendo el valor deseado en la caja de texto o utilizando los botones de incremento ubicados a la derecha. Al registrar un cambio los valores que indican la cantidad de vóxeles (*No. of voxels*) que ocupará la marca son actualizados automáticamente.

Por último, en la segunda pestaña se puede configurar la separación en centímetros del conjunto de marcas fiduciales en el eje X. Al cambiar los valores de esta propiedad los cambios se reflejarán inmediatamente en los cuatro *viewports* de la ventana principal.

#### 4.6.9 Visualizador de imágenes DICOM (DICOM Viewer)

En esta ventana se pueden apreciar todos los detalles referentes a las imágenes DICOM cargadas en el sistema, así como la información básica del paciente, permite hacer mediciones, localizar y marcar la lesión en el cráneo y cambiar los valores en la función lineal para la visualización de los cortes (ancho y centro de la ventana).

La interfaz gráfica del visor DICOM está dividida en cinco secciones importantes. Cada una de estas secciones está agrupada en sub-elementos según sus características, funcionalidades y similitudes. Como puede verse en la **Figura 4.31**, las cinco secciones principales son las siguientes:

- **Barra de menú:** corresponde a la fuente de comandos más común. En la barra de menú se encuentran todos los métodos y funcionalidades del visualizador. La barra se encuentra ubicada en la parte superior de la ventana justo debajo de la barra de título.
- **Barra de herramientas principal:** es una barra compuesta por una serie de botones con iconos que proveen al usuario de acceso rápido a las funciones más importantes. Se encuentra ubicada debajo de la barra de menú.
- **Viewport principal:** permite visualizar los cortes DICOM desde diferentes ángulos. Al abrir la ventana se visualizan por defecto los cortes axiales del volumen. Esta sección ocupa gran parte del centro de la ventana.
- **Barra de desplazamiento entre cortes:** localizada a la derecha del *Viewport* principal. Su función es permitir desplazarse entre los diferentes cortes del volumen.
- **Barra de estado:** proporciona información valiosa, tal como, mensajes de uso para el comando seleccionado, uso de atajos de teclado, mensajes informativos, etc. Está localizada en la parte inferior de la ventana.

A continuación se lista el contenido de cada uno de los menús y se detalla la funcionalidad de cada uno de los comandos.

- **El menú *File*:** solo contiene el comando para cerrar el visor DICOM (*Exit*).
- **El menú *View*:** en este menú se agrupan los comandos para la interacción con el *viewport* principal, las marcas fiduciales y el ME. Dentro de los comandos disponibles están: cambiar entre las vistas de los cortes axiales, coronales y sagitales (*Axial View*, *Coronal View* y *Sagittal View*), desplazamientos verticales y horizontales de la cámara (*Pan*), alejar o acercar los elementos de la escena (*Zoom*), capturar pantalla (*Take snapshot*), mostrar u ocultar la información del paciente y la información clínica del volumen (*Show/Hide Patient* y *Show/Hide Clinical Data*), y finalmente los comandos para

mostrar u ocultar el ME y las marcas fiduciales (*Show/Hide Stereotactic Frame* y *Show/Hide Fiducial Marks*).

- **El menú *Tools*:** en este menú se encuentran las opciones que permiten interactuar con el *Viewport* principal. Los comandos contenidos en este sub-menú son: el comando que permite visualizar en una nueva ventana toda la información contenida en la cabecera DICOM del corte (*DICOM Header*), modificar la función de visualización de los cortes (*Window/Level*), restaurar la función a su estado original (*Default Window/Level*), reconstrucción del volumen con los nuevos valores en la función de visualización (*Rebuild Volume*), ubicar la lesión en el espacio (*Locate Target*) y por último, el comando para realizar mediciones (*Measuring Tool*).



Figura 4.31 Elementos que conforman la interfaz gráfica del visor DICOM (*DICOM Viewer*).

Localizada justo debajo de la barra de menú principal se encuentra la barra de herramientas principal. Como se mencionó anteriormente ésta barra está compuesta por una serie de botones que proveen al visor de acceso rápido a las funciones más importantes y más utilizadas.

Al igual que la barra de herramientas principal de la aplicación, descrita anteriormente, esta barra cuenta con tres tipos de botones diferentes. Los botones tradicionales que activan o ejecutan un comando al hacer clic sobre ellos; los botones de dos estados y los botones con sub-menús.

En la **Tabla 4.2** se listan todos los comandos disponibles desde la barra de herramientas principal y se describen sus funciones.

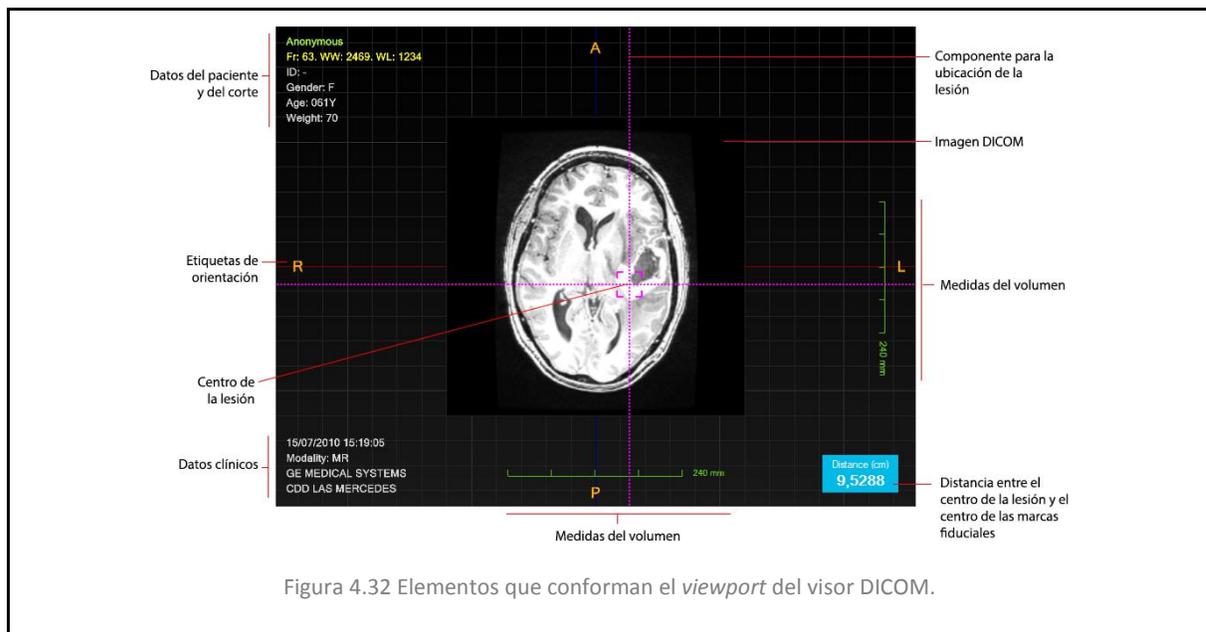
Botón	Nombre	Descripción
	<i>DICOM header</i>	Muestra una ventana con la información contenida en la cabecera DICOM del corte.
	<i>Show/Hide Information</i>	Muestra y oculta la información del paciente y los datos clínicos del volumen.
	<i>Show/Hide Stereotactic Frame</i>	Muestra y oculta el ME.
	<i>Show/Hide Fiducial Marks</i>	Muestra y oculta las guías visuales de las marcas fiduciales.
	<i>Axial View</i>	Permite visualizar los cortes axiales del volumen.
	<i>Coronal View</i>	Permite visualizar los cortes coronales del volumen.
	<i>Sagittal View</i>	Permite visualizar los cortes sagitales del volumen.
	<i>Zoom</i>	Permite alejar o acercar la cámara.
	<i>Pan</i>	Permite desplazar la cámara horizontal y verticalmente.
	<i>Window/Level</i>	Permite ajustar los valores de la función de visualización de los cortes (ancho y centro de la ventana).
	<i>Default Window Level</i>	Restaura los valores por defecto de la función de visualización de los cortes (ancho y centro de la ventana).
	<i>Rebuild Volume</i>	Permite reconstruir el volumen con los valores establecidos para la función de visualización de los cortes (ancho y centro de la ventana).
	<i>Locate Target</i>	Permite ubicar la lesión en el espacio.
	<i>Reset Target Location</i>	Restaura los valores por defecto para la ubicación de la lesión en el espacio (0, 0, 0).
	<i>Measuring Tool</i>	Permite usar la herramienta de medición.
	<i>Clean measures</i>	Elimina todas las mediciones hechas.
	<i>Take snapshot</i>	Captura en imagen el contenido del <i>viewport</i> principal.

Tabla 4.2 Lista de comandos encontrados en la barra de herramientas principal del visor DICOM.

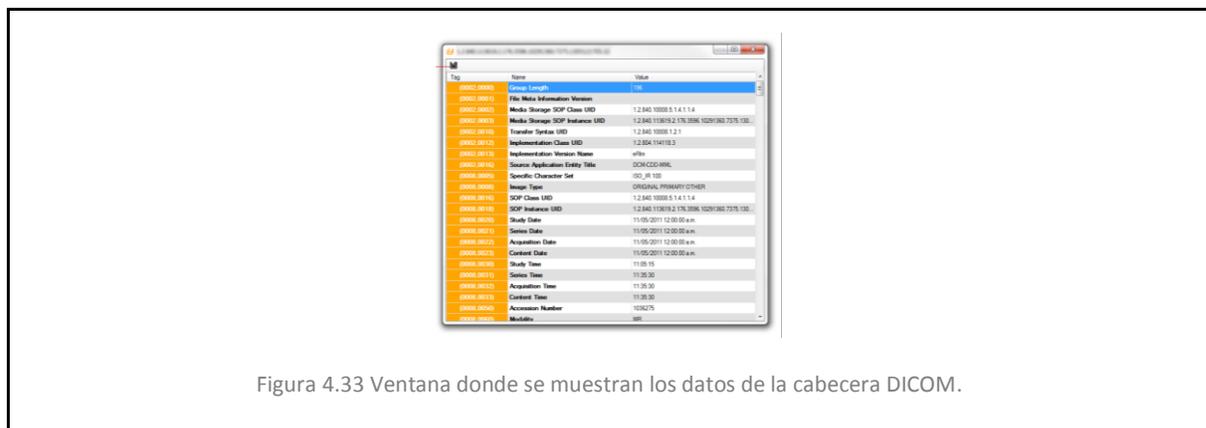
En el *Viewport* principal es donde se visualizan los cortes o las imágenes DICOM que componen el volumen. La sección se puede visualizar en la **Figura 4.32**, y está compuesta por los siguientes elementos:

- En la parte central de esta sección se encuentra la imagen DICOM correspondiente al corte del volumen seleccionado.
- De color violeta y ubicado también en la parte central se despliega el componente para la ubicación de la lesión. Dicho componente está compuesto por dos líneas, una vertical y otra horizontal, y un cuadro que demarcará la zona de la lesión.
- Ubicada en la esquina inferior derecha, se haya la etiqueta que indica la distancia euclidiana entre el centro de la lesión y el centro de la caja radiopaca o el centro de las marcas fiduciales.
- En la esquina superior izquierda se encuentran los datos del paciente (nombre, identificación, género, edad y peso), los valores referentes a la función de visualización de los cortes (ancho y centro de la ventana) y el número del corte actual.

- En la esquina inferior izquierda están los datos clínicos, como modalidad de captura de la imagen, fecha y hora de la captura, y lugar donde se realizó el estudio.
- Ubicadas en las partes inferior y derecha del *viewport* se encuentran dos reglas que indican las dimensiones del corte expresadas en milímetros.
- Finalmente, se encuentran las etiquetas de orientación. Son cuatro etiquetas que varían según el tipo de corte que se está visualizando (axial, coronal o sagital) y que están ubicadas en las partes derecha, izquierda, superior e inferior del *viewport*. Su función principal es indicar al usuario la orientación de cada corte. Por ejemplo, en la **Figura 4.32** se puede ver un corte axial del volumen, la etiqueta de orientación ubicada en la parte superior indica que la zona anterior del corte (marcada con la etiqueta A), o el rostro del paciente, se encuentran en esa dirección.



Por último, en la **Figura 4.33** se puede apreciar la ventana donde se muestra el contenido de la cabecera DICOM de un corte específico. La ventana se compone por una barra de herramientas y una lista. En la barra de herramienta se haya un único botón que sirve para salvar la información de la cabecera en texto plano en la ruta especificada por el usuario. En la lista se despliega todo el contenido de la cabecera DICOM.



#### 4.6.10 Ventana de bienvenida

Cuando la aplicación inicia el usuario es recibido con una ventana de bienvenida, mostrada en la **Figura 4.34**, la cual incluye una lista de los últimos volúmenes cargados en la aplicación (*Recent directories*) y el comando para abrir una nuevo conjunto de imágenes DICOM (*Open directory*). Adicionalmente, se incluye una presentación de la aplicación, su logotipo y la versión de la misma.



# 5

CAPÍTULO 5

# Pruebas y Resultados

---

En este capítulo:

***5.1 Pruebas y resultados cuantitativos***

***5.2 Pruebas y resultados cualitativos***

# Capítulo 5

---

## *Pruebas y Resultados*

En este capítulo se presentan las pruebas realizadas para la validación del sistema desarrollado, así como los resultados cuantitativos y cualitativos obtenidos por el uso de la aplicación. En este trabajo interesa medir la calidad y exactitud de las marcas fiduciales generadas en comparación a las marcas creadas por la caja radiopaca y se desean recopilar las distintas experiencias de los usuarios de la aplicación en términos de utilidad, facilidad de uso y cumplimiento de los objetivos de la investigación.

### **5.1 Pruebas y resultados cuantitativos**

Estas pruebas tienen como objetivo determinar la precisión y exactitud de las marcas fiduciales generadas por la aplicación desarrollada en este trabajo. Entre las pruebas se incluyen: el cálculo del error basado en las distancias entre las marcas fiduciales reales y las generadas por el sistema, el cálculo de los errores absolutos y relativos en las marcas fiduciales generadas en base con las medidas reales de los segmentos, y por último, la introducción de los conjuntos de imágenes DICOM (referidos de ahora en adelante como *datasets*, en inglés) creados, al sistema de planificación radioquirúrgico.

#### *5.1.1 Localización de los centros en las marcas fiduciales*

Antes de realizar las pruebas cuantitativas, es necesario describir el procedimiento que se realizó para la localización de los centros de las marcas fiduciales generadas por el sistema desarrollado y las marcas verdaderas. Los centros de ambos estilos de marcas fiduciales (verdaderas y virtuales) servirán como puntos de referencia (partida y llegada) al momento de hacer las mediciones, ya sea para el cálculo de distancias o de errores absolutos y relativos. En consecuencia, este procedimiento es el punto de inicio para la realización de las pruebas referidas en las siguientes secciones.

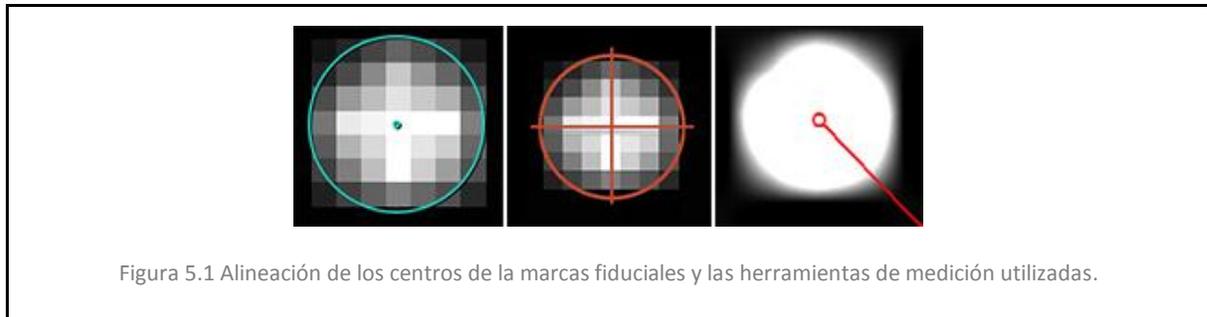
El procedimiento está dividido en dos fases: una manual y otra automatizada. La fase manual corresponde a la localización de los centros de las marcas fiduciales verdaderas por parte del observador. Debido a que las marcas fiduciales presentan, en promedio<sup>12</sup>, una forma circular, se utilizaron herramientas de

---

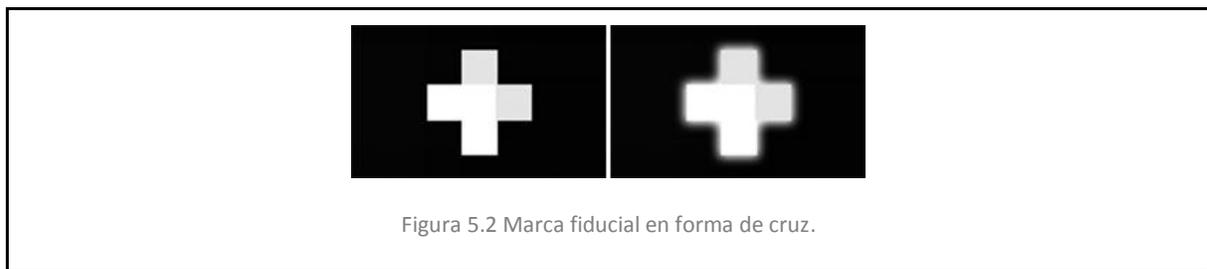
<sup>12</sup> Muchas veces las marcas fiduciales pertenecientes a los segmentos diagonales (los que unen a las marcas fiduciales de los segmentos verticales) de la caja no presentan una forma circular sino recta.

medición cuyos extremos exhibieran la misma forma (ver la **Figura 5.1**) con el propósito de que la alineación de sus centros resultara lo más preciso posible como se aprecia en la **Figura 5.1**.

Como es de esperarse, este proceso lleva siempre implícito una indeterminación asociada, en otras palabras, siempre que se traten de alinear los centros de ambos elementos (la marca fiducial y la herramienta de medición), se corre el riesgo de no “acertar” con el punto exacto. Esto puede ser debido a la imperfección de la herramienta, el hecho de que el movimiento del cursor del ratón no es preciso, la calidad de las marcas fiduciales verdaderas, la experticia del observador, la forma inadecuada de alinear los centros, etc. Toda esa indeterminación se conoce como error humano y está intrínseco en esta primera fase.



La segunda fase concierne al proceso automatizado y sirve exclusivamente para la localización de los centros de las marca fiduciales generadas por la aplicación desarrollada en este trabajo. En este caso el procedimiento para localizar los centros de las marcas reside en el **Algoritmo 4.3** descrito en la sección **4.3.2**. El objetivo del algoritmo es hallar una correspondencia entre un punto en el espacio objeto y el vóxel que constituirá el centro de la marca fiducial. Con fines de realizar las pruebas cuantitativas, el algoritmo fue modificado para que se usara una máscara en forma de cruz. Esto se hizo con la finalidad de marcar con la mayor intensidad posible, el vóxel central y sus cuatro vóxeles vecinos, como puede verse en la **Figura 5.2**.



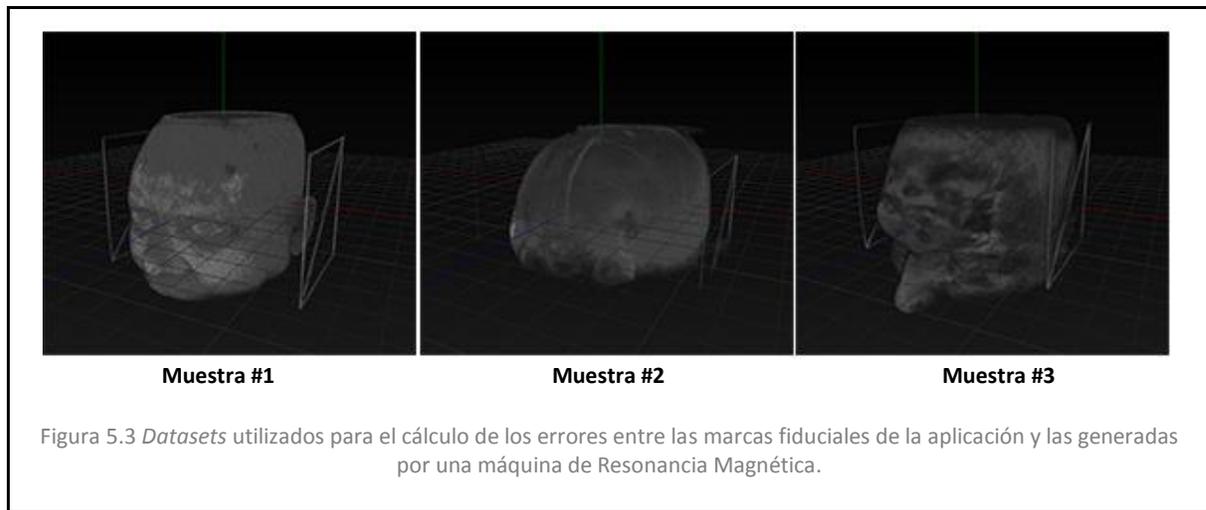
### 5.1.2 Distancias entre las marcas fiduciales

Para las primeras pruebas de cálculo de error se seleccionaron tres *datasets* (detallados en la **Tabla 5.1** y mostrados en la **Figura 5.3**). Todos los *datasets*<sup>13</sup> utilizados poseían marcas fiduciales generadas por una máquina de Resonancia Magnética (RM) y una precisión de 16 bits.

<sup>13</sup> Todos los *datasets* utilizados para la realización de las pruebas del sistema fueron suministrados por la Unidad de Radiocirugía Estereotáctica *Gamma Knife* del Centro Diagnóstico Docente ubicado en Las Mercedes, Caracas, Venezuela.

	Muestra #1	Muestra #2	Muestra #3
<b>Dimensiones del volumen (vóxeles)</b>	512 x 512 x 248	256 x 256 x 200	512 x 512 x 146
<b>Bits por vóxel</b>	16	16	16
<b>Dimensiones del vóxel (mm)</b>	0,5273 x 0,5273 x 0,6	1,1328 x 1,1328 x 0,5	0,6250 x 0,6250 x 1,0

Tabla 5.1 Características de los *datasets* utilizados para el cálculo de los errores entre las marcas fiduciales de la aplicación y las generadas por una máquina de Resonancia Magnética.



Las pruebas consistieron en hacer mediciones sub-milimétricas sobre los cortes axiales de cada *dataset* para determinar la distancia entre las marcas fiduciales generadas por el sistema desarrollado y las marcas verdaderas. Como cada una de las marcas virtuales tiene una correspondencia con una única marca real, las mediciones fueron realizadas en pares como se ve en la **Figura 5.4 (a)**. Adicionalmente, para tener mejor lectura de los resultados se enumeraron los seis pares de marcas fiduciales presentes en el corte axial, como se aprecia en la **Figura 5.4 (b)**.

Todas las medidas fueron realizadas tomando como punto de partida y llegada los centros de cada marca fiducial y utilizando la Distancia Euclidiana (ver **Ecuación 5.1**). La distancia medida sirvió para representar el error en el posicionamiento de las marcas fiduciales generadas, por lo que, mientras mayor fuera esa distancia registrada mayor sería el error cometido.

Para realizar el cálculo de longitud se utilizaron las herramientas de medición de tres aplicaciones diferentes: la aplicación desarrollada en este trabajo llamada *Virtual Stereotactic Frame* ([VSF]), *synedra View* ([SYN]) [46] y *Agnosco DICOM Viewer* ([AGN]) [47].

En los tres casos de estudio el Marco Estereotáctico (ME) fue posicionado virtualmente haciendo translaciones o rotaciones del mismo con el objeto de replicar la colocación original. Estos son los detalles de las transformaciones realizadas:

- **Muestra #1:** rotación en el eje Z de -0,59 grados y translaciones de 0,004 mm, 0,003 mm, y -0,096 mm en X, Y y Z respectivamente.

- **Muestra #2:** rotación en el eje Z de 0,20 grados y translaciones de -0,001 mm, -0,017 mm, -0,121 mm en X, Y y Z respectivamente.
- **Muestra #3:** rotación en el eje Z de -0,92 grados y translaciones de -0,005 mm, -0,006 mm, -0,086 mm en X, Y y Z respectivamente.

El sistema de pruebas fue un PC con procesador *AMD Phenom™ II X6 1055T*, 8GB de memoria principal y una tarjeta gráfica *NVidia GeForce 450 GTS* con 1024MB de memoria dedicada con soporte para la versión 4.2 de *OpenGL*. El sistema operativo del PC es *Windows 7® Ultimate* de 64 bits.

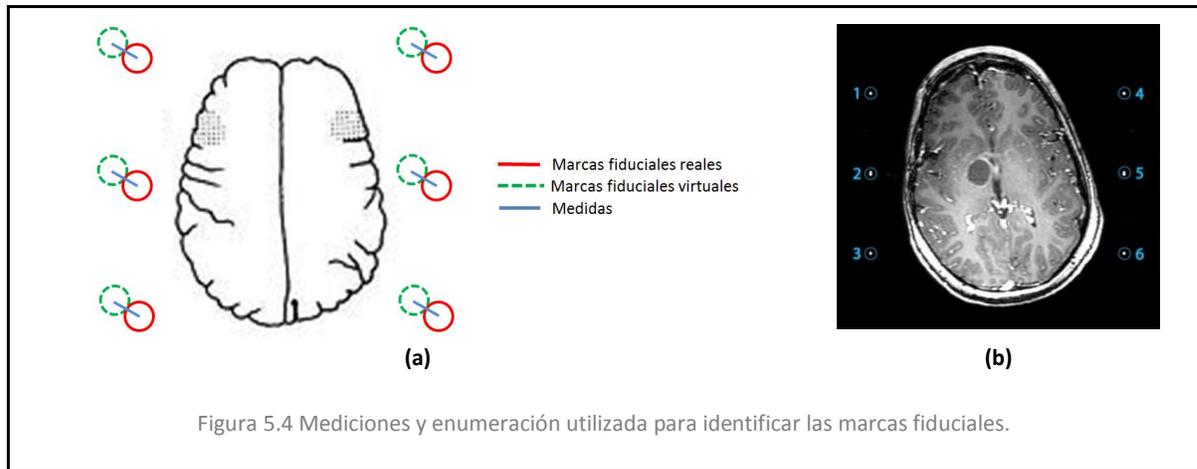


Figura 5.4 Mediciones y enumeración utilizada para identificar las marcas fiduciales.

Sea  $P_1$  y  $P_2$  dos puntos en el Espacio Euclidiano, de coordenadas  $(x_1, y_1)$  y  $(x_2, y_2)$  respectivamente, la Distancia Eclidiana ( $d_E$ ) entre ambos puntos se expresa como:

$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Ecuación 5.1 Distancia Euclidiana.

Los resultados que arrojaron las pruebas sobre la **Muestra #1** son mostrados en **Tabla 5.2** e indican que la distancia promedio existente entre las marcas fue de **0,8432 mm**. Para la mayoría de las marcas la distancia registrada se encuentra por debajo de un milímetro, llegando incluso a ser menor de medio milímetro para el conjunto de marcas número 4. En un solo caso la distancia excedió el milímetro y fue en el conjunto de marcas número 5, registrando una distancia promedio de 1,2167 mm.

Aunque la longitud puede ser considerada elevada, es importante señalar que en este tipo de mediciones intervienen factores como el ruido en la muestra y el hecho de que las marcas fiduciales no estaban alineadas a los ejes cartesianos. En este caso en particular no se pudo reproducir la colocación exacta del ME debido a que originalmente éste fue rotado en el eje X, factor que puede ser visto como un error en la entrada inicial de los datos. Sin embargo, indiferentemente de los problemas anteriormente expuestos, se pudo lograr una buena aproximación a las marcas verdaderas con una distancia promedio de menos de un milímetro de longitud.

Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)
1	[VSF]	0,9000	0,9193	2	[VSF]	0,8500	0,7873	3	[VSF]	0,8000	0,7463
	[SYN]	0,9580			[SYN]	0,9120			[SYN]	0,7390	
	[AGN]	0,9000			[AGN]	0,6000			[AGN]	0,7000	
4	[VSF]	0,4200	0,4397	5	[VSF]	1,1500	1,2167	6	[VSF]	0,9500	0,9500
	[SYN]	0,3990			[SYN]	1,3000			[SYN]	1,0000	
	[AGN]	0,5000			[AGN]	1,2000			[AGN]	0,9000	

Tabla 5.2 Errores de distancia para la Muestra #1.

En cuanto a las pruebas realizadas con la **Muestra #2**, cuyos resultados se detallan en la **Tabla 5.3**, y que a diferencia de la muestra anterior posee un mayor tamaño de vóxel, los resultados mejoraron significativamente. En este caso la distancia promedio en los seis conjuntos de marcas fue de **0,5728 mm**. En tres casos (conjuntos de marcas número 1, 2 y 6) las diferencias entre las marcas fueron menores a medio milímetro. En un solo caso la distancia excedió el milímetro de longitud y fue en el conjunto de marcas número 3, registrando un error promedio de 1,0033 mm. Existen dos factores que explican la reducción de la Distancia Euclidiana en este caso de estudio. El primer factor reside en que las marcas fiduciales verdaderas en esta muestra estaban alineadas a los ejes cartesianos (ligeramente rotada en 0,20 grados en el eje Z, valor despreciable), lo que facilitó la reproducción el posicionamiento del ME. El segundo factor es que esta muestra posee un mayor tamaño de vóxel, lo que se traduce en una mejor calidad de imagen.

Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)
1	[VSF]	0,2800	0,2833	2	[VSF]	0,5000	0,4890	3	[VSF]	1,0000	1,0033
	[SYN]	0,3500			[SYN]	0,4370			[SYN]	0,8100	
	[AGN]	0,2200			[AGN]	0,5300			[AGN]	1,2000	
4	[VSF]	0,7000	0,7163	5	[VSF]	0,5500	0,5707	6	[VSF]	0,4000	0,3743
	[SYN]	0,5490			[SYN]	0,6620			[SYN]	0,3230	
	[AGN]	0,9000			[AGN]	0,5000			[AGN]	0,4000	

Tabla 5.3 Errores de distancia para la Muestra #2.

Para las pruebas realizadas con la **Muestra #3** (ver resultados en la **Tabla 5.4**), la muestra con la menor profundidad de las tres, se notó un incremento en la diferencia entre las marcas verdaderas y las generadas. Antes de desglosar los resultados de este estudio es importante mencionar que, al igual que la **Muestra #1**, en esta comparación intervinieron factores como el ruido en la muestra y el hecho de que las marcas fiduciales verdaderas no estaban alineadas a los ejes cartesianos (rotadas en el eje X). La distancia promedio en los seis conjuntos de marcas fue de **1,2601 mm**. Los conjuntos número 1, 2, 3, 4 y 5 se aproximaron en promedio a un milímetro de diferencia, mientras que el promedio de distancia del conjunto número 6 excedió los dos milímetros.

En relación con lo anterior, si se analizan los resultados independientemente, se puede notar que el promedio de la distancia se eleva por el aporte excesivo del promedio del conjunto de marcas número 6. De

hecho, si se promedian las distancias de los cinco primeros conjuntos de marcas el resultado es de apenas **1,0254 mm**, lo que resulta una buena aproximación considerando los problemas descritos anteriormente.

Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)	Marca	Aplic.	$d_E$ (mm)	$\overline{d_E}$ (mm)
1	[VSF]	1,0000	0,9403	2	[VSF]	1,0000	0,9670	3	[VSF]	1,0000	1,1500
	[SYN]	0,9210			[SYN]	1,0010			[SYN]	1,5000	
	[AGN]	0,9000			[AGN]	0,9000			[AGN]	0,9500	
4	[VSF]	1,0000	1,2333	5	[VSF]	0,8000	0,8367	6	[VSF]	2,5000	2,4333
	[SYN]	1,3000			[SYN]	1,2100			[SYN]	2,7000	
	[AGN]	1,4000			[AGN]	0,5000			[AGN]	2,1000	

Tabla 5.4 Errores de distancia para la Muestra #3.

En general, el promedio de la distancia existente fue de **0,8920 mm**, sin diferenciar entre marcas. Esta cifra muestra que las marcas generadas poseen una gran similitud con respecto a las marcas verdaderas. Adicionalmente, los resultados obtenidos indican que si los datos de entrada no poseen errores (ruido del *dataset* o rotaciones en ejes distintos al eje X) la similitud en términos de localización de las marcas será cada vez mayor.

### 5.1.3 Errores absolutos y errores relativos porcentuales

Para la realización de estas pruebas es necesario el cálculo de los llamados errores de medición. El error de medición es la discrepancia entre la magnitud verdadera y la obtenida, siendo dichas magnitudes los resultados de un proceso de medición.

Estas pruebas tienen como objetivo medir la precisión y exactitud de las marcas fiduciales generadas por la aplicación a través del cálculo de los errores absolutos y relativos. El error absoluto ( $\Delta E$ ) (o desviación absoluta) puede definirse como el módulo de la diferencia entre el resultado de la medida ( $M$ ) y el valor tomado como exacto o verdadero ( $m_0$ ) de la magnitud a medir, su fórmula puede ser apreciada en la **Ecuación 5.2 (a)**. Por otra parte, el error relativo porcentual (o desviación relativa porcentual) se define como el cociente entre el error absoluto ( $\Delta E$ ) y el valor verdadero ( $m_0$ ), su expresión puede verse en la **Ecuación 5.2 (b)**. Éste último tipo de error es adimensional, en otras palabras, carece de unidades de medida, y es utilizado para representar la presión de la medida realizada.

$$\Delta E = |M - m_0| \quad (a) \quad E_r(\%) = \frac{\Delta E \times 100}{m_0} \quad (b)$$

Ecuación 5.2 (a) Ecuación para el cálculo del error absoluto. (b) Ecuación para el cálculo del error relativo porcentual.

Para estas pruebas se seleccionaron tres nuevos *datasets* (detallados en la **Tabla 5.5** y mostrados en la **Figura 5.5**). En contraste con los *datasets* utilizados en las pruebas anteriores, estos *datasets* no poseían las marcas fiduciales generadas por la caja radiopaca.

	Muestra #4	Muestra #5	Muestra #6
<b>Dimensiones del volumen (vóxeles)</b>	512 x 512 x 200	512 x 512 x 124	512 x 512 x 146
<b>Bits por vóxel</b>	16	16	16
<b>Dimensiones del vóxel (mm)</b>	0,5664 x 0,5664 x 0,5	0,4688 x 0,4688 x 1,0	0,625 x 0,625 x 1,0

Tabla 5.5 Características de los *datasets* utilizados para el cálculo de los errores absolutos y relativos.

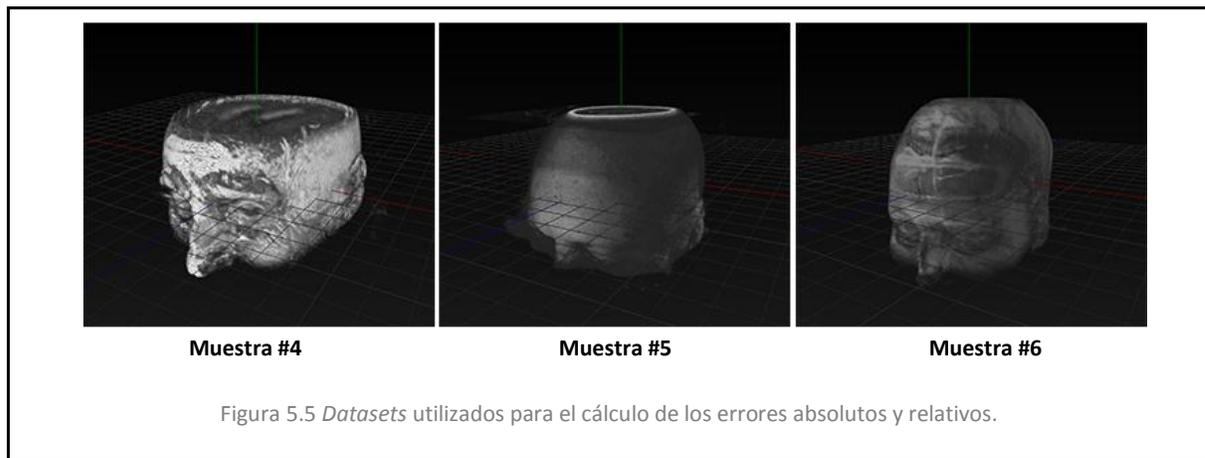


Figura 5.5 *Datasets* utilizados para el cálculo de los errores absolutos y relativos.

Las pruebas consisten en hacer uso del módulo para la generación de marcas fiduciales contenido en el paquete «*Volume*» (ver sección **4.3.2** para mayores detalles). Una vez generadas las marcas, lo sucesivo fue realizar mediciones sub-milimétricas sobre tres segmentos claves (mostrados en la **Figura 5.6**): los segmentos horizontales y verticales sobre los cortes sagitales (con una longitud física exacta de 120 mm cada uno) y el segmento que separa el conjunto de marcas fiduciales derecho del izquierdo (con una longitud física exacta de 190 mm).

Con la obtención de las longitudes de cada segmento y conociendo la longitud verdadera de los mismos en el ME, se aplicó el cálculo de los errores absolutos y relativos porcentuales para cada muestra y segmento, con el objeto de obtener cuantitativamente la exactitud de las marcas generadas.

Al igual que las pruebas anteriores, para realizar el cálculo de longitud se utilizaron las herramientas de medición de tres aplicaciones diferentes: la aplicación desarrollada en este trabajo llamada *Virtual Stereotactic Frame* ([VSF]), *synedra View* ([SYN]) [46] y *Agnosco DICOM Viewer* ([AGN]) [47]. Todas las medidas fueron realizadas tomando como punto de partida y llegada los centros de cada marca fiducial.

Para las pruebas realizadas con la **Muestra #4** (detalladas en la **Tabla 5.6**) se obtuvieron errores absolutos promedio de  $\pm 0,1433$  mm y  $\pm 0,1067$  mm y errores relativos porcentuales promedio de 0,1194% y 0,0889% para los segmentos verticales y horizontales respectivamente.

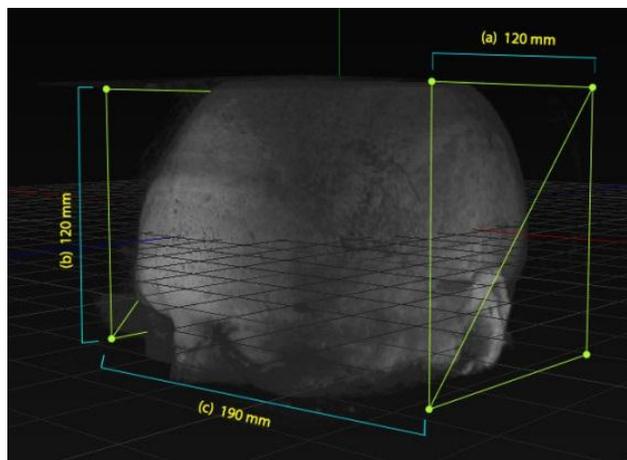


Figura 5.6 Medidas reales de los segmentos de las marcas fiduciales. (a) Distancia del segmento horizontal, (b) distancia del segmento vertical y (c) distancia entre los conjuntos de marcas fiduciales.

Los resultados indican que la magnitud de los errores cometidos en ambos casos no alcanzan los  $\pm 0,15$  mm, lo que puede ser interpretado como una excelente aproximación a las longitudes verdaderas en términos de exactitud. Sin embargo, para la medición del segmento (c) se obtuvo un error absoluto promedio de  $\pm 0,6333$  mm, ligeramente mayor a los anteriores pero que aún puede tomarse como una buena aproximación de la longitud verdadera.

En general para esta muestra se obtuvo un error absoluto promedio de apenas  $\pm 0,2467$  mm y un error relativo porcentual promedio de **0,1805%**, lo que indica la buena precisión y exactitud de las marcas fiduciales generadas.

	Segmento	Aplicación	$M_i$ (mm)	$m_0$ (mm)	$\Delta E$ (mm)	$E_r$ (%)
Muestra #4	(a)	[VSF]	119,8700	120,0000	$\pm 0,1300$	0,1083
		[SYN]	119,6000	120,0000	$\pm 0,4000$	0,3333
		[AGN]	120,1000	120,0000	$\pm 0,1000$	0,0833
	Promedio	-	<b>119,8567</b>	-	<b><math>\pm 0,1433</math></b>	<b>0,1194</b>
	(b)	[VSF]	120,1000	120,0000	$\pm 0,1000$	0,0833
		[SYN]	120,1000	120,0000	$\pm 0,1000$	0,0833
		[AGN]	120,1200	120,0000	$\pm 0,1200$	0,1000
	Promedio	-	<b>120,1067</b>	-	<b><math>\pm 0,1067</math></b>	<b>0,0889</b>
	(c)	[VSF]	189,6000	190,0000	$\pm 0,4000$	0,2105
		[SYN]	189,1000	190,0000	$\pm 0,9000$	0,4737
		[AGN]	189,4000	190,0000	$\pm 0,6000$	0,3158
	Promedio	-	<b>189,3667</b>	-	<b><math>\pm 0,6333</math></b>	<b>0,3333</b>

Tabla 5.6 Detalles de los errores absolutos y relativos porcentuales de la Muestra #4.

En cuanto a las pruebas realizadas con la **Muestra #5**, cuyos resultados se detallan en la **Tabla 5.7**, y que a diferencia de la muestra anterior posee un mayor tamaño de vóxel y mayor profundidad, los resultados mejoraron significativamente. Los errores absolutos promedio obtenidos fueron de  $\pm 0,0200$  mm y  $\pm 0,0733$  mm y errores relativos porcentuales promedio fueron de 0,0167% y 0,0611% para los segmentos verticales y horizontales respectivamente. En ambos casos el error es inferior a una décima de milímetro.

	Segmento	Aplicación	$M_i$ (mm)	$m_0$ (mm)	$\Delta E$ (mm)	$E_r$ (%)
Muestra #5	(a)	[VSF]	120,0100	120,0000	$\pm 0,0100$	0,0083
		[SYN]	119,9000	120,0000	$\pm 0,1000$	0,0833
		[AGN]	120,1500	120,0000	$\pm 0,1500$	0,1250
	Promedio	-	<b>120,0200</b>	-	<b><math>\pm 0,0200</math></b>	<b>0,0167</b>
	(b)	[VSF]	120,0100	120,0000	$\pm 0,0100$	0,0083
		[SYN]	120,0100	120,0000	$\pm 0,0100$	0,0083
		[AGN]	120,2000	120,0000	$\pm 0,2000$	0,1667
	Promedio	-	<b>120,0733</b>	-	<b><math>+0,0733</math></b>	<b>0,0611</b>
	(c)	[VSF]	189,6100	190,0000	$\pm 0,3900$	0,2053
		[SYN]	188,9000	190,0000	$\pm 1,1000$	0,5789
		[AGN]	189,8000	190,0000	$\pm 0,2000$	0,1053
	Promedio	-	<b>189,4367</b>	-	<b><math>\pm 0,5633</math></b>	<b>0,2965</b>

Tabla 5.7 Detalles de los errores absolutos y relativos porcentuales de la Muestra #5.

Los resultados representan un aumento con respecto a la **Muestra #4** en la exactitud de las marcas correspondientes a los segmentos verticales y horizontales por una diferencia de -0,1027% y -0,0278%, respectivamente. En cuanto al segmento (c), se registró un error absoluto promedio de  $\pm 0,5633$  mm y un error relativo porcentual promedio de 0,2965%, que aunque resultan ligeramente elevados en comparación a los otros segmentos, representa una mejora en términos de exactitud con una diferencia de -0,0368% para el error relativo porcentual con respecto a la **Muestra #4**.

En general para esta muestra se obtuvo un error absoluto promedio de apenas  **$\pm 0,2189$  mm** y un error relativo porcentual promedio de **0,1248%**, lo que indica la excelente precisión y exactitud de las marcas fiduciales generadas para esta muestra.

Para las pruebas realizadas con la **Muestra #6** (ver **Tabla 5.8**), la muestra con el mayor tamaño de vóxel de las tres, se notó una mejora en la exactitud de las marcas en los segmentos (b) y (c) mientras que en el segmento (a) presentó la mayor inexactitud de los tres casos de estudio. Para el segmento (a) se obtuvo un error absoluto promedio de  $\pm 0,1467$  mm y un error relativo porcentual promedio de 0,1222%, lo que representa una diferencia de +0,0028% en comparación con el error relativo porcentual promedio de la **Muestra #4** y de +0,1055% con respecto al error relativo porcentual promedio de la **Muestra #5**. Sin embargo, la magnitud del error cometido en promedio no alcanza los  $\pm 0,15$  mm, por lo que la longitud del segmento puede considerarse una buena aproximación a las medidas reales.

	Segmento	Aplicación	$M_i$ (mm)	$m_0$ (mm)	$\Delta E$ (mm)	$E_r$ (%)
Muestra #6	(a)	[VSF]	119,7600	120,0000	$\pm 0,2400$	0,2000
		[SYN]	119,7000	120,0000	$\pm 0,3000$	0,2500
		[AGN]	120,1000	120,0000	$\pm 0,1000$	0,0833
	Promedio	-	<b>119,8533</b>	-	<b><math>\pm 0,1467</math></b>	<b>0,1222</b>
	(b)	[VSF]	119,9000	120,0000	$\pm 0,1000$	0,0833
		[SYN]	119,9000	120,0000	$\pm 0,1000$	0,0833
		[AGN]	120,0000	120,0000	$\pm 0,0000$	0,0000
	Promedio	-	<b>119,9333</b>	-	<b><math>\pm 0,0667</math></b>	<b>0,0556</b>
	(c)	[VSF]	189,9900	190,0000	$\pm 0,0100$	0,0053
		[SYN]	189,7000	190,0000	$\pm 0,3000$	0,1579
		[AGN]	190,6000	190,0000	$\pm 0,6000$	0,3158
	Promedio	-	<b>190,0967</b>	-	<b><math>\pm 0,0967</math></b>	<b>0,0509</b>

Tabla 5.8 Detalles de los errores absolutos y relativos porcentuales de la Muestra #6.

En contraparte, la exactitud de los segmentos (b) y (c) mejoraron en relación con las dos muestras anteriores registrando errores absolutos promedios de  $\pm 0,0667$  mm y  $\pm 0,0967$  mm respectivamente. En ambos casos hubo un decremento del error relativo porcentual promedio cometido de hasta -0,2456% para el segmento (c) donde se venía registrando el mayor error. En general para esta muestra se obtuvo un error absoluto promedio de  **$\pm 0,1034$  mm** y un error relativo porcentual promedio de **0,0762%**, lo que indica la excelente precisión y exactitud de las marcas fiduciales generadas para esta muestra.

Basado en las tres muestras estudiadas, el error absoluto promedio cometido fue de  **$\pm 0,1897$  mm** y el error relativo porcentual fue de **0,1272%**, sin diferenciar el segmento medido. Estas cifras muestran una alta precisión y exactitud en términos de localidad y longitud de las marcas fiduciales generadas por el sistema desarrollado. Adicionalmente, los resultados indican que mientras mayor sea el tamaño del vóxel mayor precisión y exactitud tendrán las marcas generadas, lo que resulta lógico ya que se trabajan con texturas de mayor resolución, lo que se traduce en, no solo marcas fiduciales más exactas, sino de mayor calidad visual.

#### 5.1.4 Introducción de los datasets generados en el sistema de planificación

Para estas pruebas fue utilizado el sistema de planificación de tratamiento radioquirúrgico *Leksell GammaPlan*© (LGP) [48]. El LGP es un potente sistema computarizado para el diseño del plan de tratamiento radioquirúrgico, específicamente diseñado para la simulación y planeación de radiocirugía estereotáctica usando el *Leksell Gamma Knife*® basado en imágenes proyectadas y tomografías.

La base de la planificación del tratamiento es la adquisición y procesamiento de imágenes digitales por parte de una estación de trabajo (*Workstation*, en inglés) en la cual se encuentra el sistema de planificación. El programa es capaz de manejar una amplia gama de modalidades de imágenes médicas, incluyendo

Tomografías Axiales Computarizadas (TAC), Resonancias Magnéticas (RM), Tomografías por Emisión de Positrones (TEP) y Angiografías Digitales (AD).

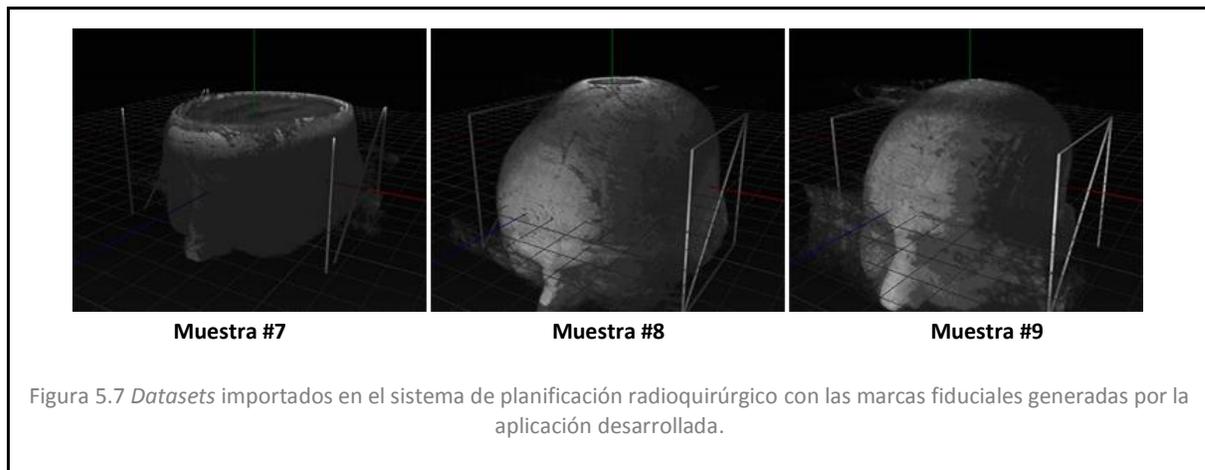
Las imágenes digitales pueden ser cargadas o importadas al sistema a través de la red, por medio de un disco compacto o con una memoria *flash* a través de un puerto USB.

Las pruebas consistieron en importar los *datasets* de salida de la aplicación desarrollada en el sistema de planificación LGP y determinar si las marcas fiduciales generadas podían ser procesadas correctamente. En este punto se estudian dos características principales: la estructura de las marcas generadas (tamaño, forma e intensidad de color) y, al igual que el apartado anterior, las longitudes de los segmentos claves (mostrados en la **Figura 5.6**) para determinar la desviación de los mismos.

Se seleccionaron tres *datasets* que al igual que los utilizados en las pruebas anteriores, no poseían las marcas fiduciales generadas por la caja radiopaca, sino que fueron generadas por la aplicación desarrollada en este trabajo. Los *datasets* utilizados están detallados en la **Tabla 5.9** y son mostrados en la **Figura 5.7**.

	Muestra #7	Muestra #8	Muestra #9
<b>Dimensiones del volumen (vóxeles)</b>	512 x 512 x 200	512 x 512 x 146	512 x 512 x 146
<b>Bits por vóxel</b>	16	16	16
<b>Dimensiones del vóxel (mm)</b>	0,5664 x 0,5664 x 0,5	0,625 x 0,625 x 1,0	0,625 x 0,625 x 1,0
<b>Rotaciones</b>	No	No	No

Tabla 5.9 Características de los *datasets* importados en el sistema de planificación radioquirúrgico.



Los primeros resultados indican que las estructuras de las marcas generadas pudieron ser aceptadas y procesadas por el sistema de planificación, en otras palabras, las marcas poseían el tamaño, la forma y las intensidades de color exactas para ser procesadas. Los siguientes resultados corresponden a las desviaciones medias y máximas de los segmentos claves medidos. La desviación media ( $D_m$ ) se define como el promedio o la media de las desviaciones absolutas (o errores absolutos, ver la **Ecuación 5.2 (a)**), su expresión puede verse en la **Ecuación 5.3 (a)**. Por otra parte, la desviación máxima ( $D_{m\acute{a}x}$ ) corresponde al máximo

valor de las desviaciones absolutas (ver la **Ecuación 5.3 (b)**). Para ambas desviaciones (medias y máximas), mientras menor sea el valor registrado más exactas fueron las marcas generadas.

$$D_m = \frac{1}{N} \sum_{i=1}^N |M_i - m_i|$$

**(a)**

$$D_{m\acute{a}x} = \text{MAX}(|M_i - m_i|), i \in [1..N]$$

**(b)**

Ecuación 5.3 (a) Ecuación de desviación media. (b) Ecuación de desviación máxima.

Los resultados obtenidos fueron los siguientes:

- **Muestra #7:** en esta muestra se registró una desviación media de  $\pm 0,4$  mm con una desviación máxima de  $\pm 0,6$  mm.
- **Muestra #8:** la muestra registró una desviación media de  $\pm 0,3$  mm, menor que la muestra anterior. No obstante se registró una desviación máxima de  $\pm 0,6$  mm, igual al caso de estudio anterior.
- **Muestra #9:** en este caso se registraron las menores desviaciones con  $\pm 0,1$  mm y  $\pm 0,4$  mm, para la desviación media y máxima respectivamente.
- Como puede verse en la **Tabla 5.9** las **Muestras #8 y #9** son iguales a nivel estructural, sin embargo, los resultados obtenidos discrepan por 0,2 mm para ambas desviaciones. Estos resultados sugieren que existen factores externos, como el ruido en la muestra, que pueden intervenir en el procesamiento de las marcas.
- Para las tres muestras en general se registró una desviación media de  **$\pm 0,2667$  mm**. Dicho valor se aproxima al error absoluto promedio registrado en la sección anterior.

## 5.2 Pruebas y resultados cualitativos

Las pruebas cuantitativas se basaron en recopilar las distintas experiencias de los usuarios de la aplicación en términos de facilidad de uso, utilidad y cumplimiento de los objetivos de la investigación.

El instrumento diagnóstico utilizado fue la entrevista y se organizó en torno a doce (12) preguntas contenidas en cuatro tópicos principales:

- 1) Utilidad de la interfaz gráfica de usuario.
- 2) Facilidad en el posicionamiento y configuración del Marco Estereotáctico.
- 3) Evaluación cualitativa de las marcas fiduciales generadas.
- 4) Utilidad de la aplicación en términos de tiempo, esfuerzo y recursos utilizados.

La entrevista fue realizada en la Unidad de Radiocirugía Estereotáctica *Gamma Knife* del Centro Diagnóstico Docente ubicado en Las Mercedes, Caracas, Venezuela. El entrevistado fue un doctor en Física Molecular con más de dos años de experiencia en el área de Radiocirugía Estereotáctica con *Gamma Knife*.

### 5.2.1 Análisis de los resultados

A continuación, se detallan y analizan los resultados obtenidos:

- Utilidad de la interfaz gráfica de usuario: en este rubro se evaluó la interacción humano-computador, la usabilidad y la curva de aprendizaje de la aplicación y de todas sus funcionalidades.
  - Los resultados de la entrevista indican que todas las herramientas interactivas proporcionadas por la interfaz gráfica de usuario facilitan el uso de la aplicación ya que las mismas permiten la visualización completa del ME y del volumen del paciente desde distintos puntos de vista, facilitan el desplazamiento y rotaciones del ME y posibilitan la visualización de los diferentes cortes del volumen (en dirección axial, coronal o sagital).
  - Los resultados revelan que el usuario considera que la información brindada por el sistema en el posicionamiento del ME es de gran utilidad, ya que da una referencia anatómica de la faz del paciente que permite la ubicación del ME en la realidad, aporta las medidas de los postes e indica la distancia entre el centro de la caja radiopaca y el blanco quirúrgico.
  
- Facilidad en el posicionamiento y configuración del Marco Estereotáctico: en este tópico se evaluó el procedimiento entorno al posicionamiento del ME así como la localización de la lesión, la utilidad del despliegue volumétrico del paciente y las opciones para la configuración del ME y sus piezas.
  - El entrevistado señaló que gracias al despliegue volumétrico de las imágenes DICOM, el posicionamiento del ME se hace más sencillo. Esto se logra debido a que es posible visualizar simultáneamente el volumen del cerebro, la ubicación de la lesión y el cráneo del paciente.
  - Los resultados muestran que la localización de la lesión en el espacio se realiza de manera sencilla. Esto se logra gracias a que la aplicación permite el despliegue de las imágenes DICOM del paciente según los planos axial, sagital y coronal; donde de manera independiente se puede ubicar la lesión en el espacio tridimensional. Por último, el sistema posee una herramienta que hace coincidir el centro de la caja de marcas con el centro de la lesión previamente ubicada.
  - Los resultados de la entrevista también indicaron que la mayor parte del procedimiento para la colocación del ME y sus piezas es realizada de manera sencilla e intuitiva. Sin embargo, cuando se trataba la ubicación vertical de los cuatro postes, el sistema dejaba de ser interactivo. La causa que originó tal respuesta apunta a que el desplazamiento de los postes se hace a través del panel de configuración del ME y no con el uso de un componente gráfico (similar a la herramienta de translación del ME) que interactuara directamente con los movimientos del ratón.
  - El entrevistado se manifestó totalmente de acuerdo con las opciones brindadas por la aplicación para la configuración visual del ME y señala que los parámetros de configuración para cada componente son suficientes.
  - El usuario se mostró de acuerdo con el hecho de que la aplicación proporciona toda la información necesaria para el posicionamiento del ME en cuanto al despliegue del volumen del paciente, el modelo del ME, la caja de marcas externas, la distancia entre el centro de la caja radiopaca y la lesión, y las medidas desplegadas.

- Evaluación cualitativa de las marcas fiduciales generadas: en esta sección se evaluaron las marcas fiduciales generadas en términos de calidad visual y efectividad. Las respuestas obtenidas para ambas preguntas fueron neutrales.
  - Las respuestas obtenidas revelan que el hecho de que la aplicación permita la configuración de los parámetros que intervienen en la generación de las marcas fiduciales (ver sección 4.6.8) supuso un inconveniente para el usuario. La problemática reside en que antes de proceder a la generación de las marcas, todos los parámetros deben estar correctamente configurados. Debido lo anteriormente expuesto, existe la posibilidad de que se genere un conjunto de marcas fiduciales inválido y que conlleve a que el *dataset* no pueda ser procesado por el sistema de planificación radioquirúrgico. No obstante, si los parámetros de configuración mantienen sus valores por defecto no existirán problemas al momento de generar las marcas fiduciales.
  - El usuario percibió que las marcas fiduciales generadas no se asemejaban visualmente a las marcas originales. Si bien las marcas generadas presentan la forma redonda, las respuestas indican que no son tan difusas como las marcas de una imagen de resonancia magnética. Sin embargo, es importante señalar que la respuesta a esta pregunta pudo verse afectada por el método y las características (uso de *antialiasing*, pérdida de profundidad y resolución en las imágenes, transformación lineal aplicada a los píxeles, etc.) de despliegue del *software* de visualización de imágenes DICOM utilizado.
  
- Utilidad de la aplicación en términos de tiempo, esfuerzo y recursos utilizados: en este tópico se englobaron las preguntas referentes a la utilidad y posibilidades de uso del sistema desarrollado.
  - Los resultados obtenidos indican que el uso del sistema representa una mejora significativa en referencia al tiempo y recursos invertidos en el proceso de colocación del ME cuando se cuentan con lesiones de complicada ubicación. No obstante, cuando en el procedimiento se tratan lesiones de fácil localización y relativamente centradas en el cerebro, el uso de la aplicación resulta innecesario según los usuarios. Por tales motivos, las respuestas a esta pregunta resultaron neutrales.
  - El entrevistado aseguró que con el uso del sistema, en principio se puede generar un plan de tratamiento inicial basado exclusivamente en los estudios imagenológicos del paciente. Sin embargo, las respuestas obtenidas sugieren que es necesario que el sistema contenga un módulo que permita ubicar y medir el hueso del cráneo, similar al módulo del sistema de planificación.
  - El entrevistado mencionó que el sistema puede ser usado como una herramienta de formación interactiva para el personal médico del área de radiocirugía. Esto se debe a las características del sistema, su facilidad de uso, la posibilidad de trabajar bajo el método heurístico de prueba y error, y el hecho de que los usuarios trabajan con una réplica del ME y el volumen del paciente.

# 6

CAPÍTULO 6

# Conclusiones y Trabajos Futuros

---

En este capítulo:

***6.1 Conclusiones***

***6.2 Trabajos Futuros***

# Capítulo 6

---

## *Conclusiones y Trabajos Futuros*

En este trabajo se ha desarrollado un sistema que permite simular de manera virtual la colocación del Marco Estereotáctico en el cráneo del paciente, haciendo uso de las imágenes diagnósticas obtenidas antes del tratamiento radioquirúrgico. A continuación se listan las conclusiones y los trabajos futuros más importantes producto de esta investigación.

### 6.1 Conclusiones

Durante la investigación se analizaron todas las variables y condiciones que intervienen durante la colocación del ME en el cráneo. Con la información obtenida se estudiaron las limitaciones del procedimiento y se encontró que se debía desarrollar un sistema para el posicionamiento interactivo de la representación tridimensional del ME en un ambiente virtual a través de rotaciones y traslaciones utilizando una interfaz gráfica de usuario.

La interacción humano-computador fue ideada para reducir la curva de aprendizaje de la aplicación y de todas sus funcionalidades. De acuerdo a los resultados obtenidos en la entrevista realizadas, se encontró que el conjunto de componentes e interfaces gráficas que se diseñaron e implementaron permiten al usuario interactuar con el sistema de manera sencilla, facilita el posicionamiento del ME y la ubicación del blanco quirúrgico.

La biblioteca dinámica *DicomLib* permite cargar y procesar las imágenes en formato DICOM provenientes de los estudios imageneológicos del paciente. El conjunto de imágenes DICOM es desplegado en pantalla empleando la técnica de despliegue volumétrico con texturas bidimensionales llamada Planos Alineados a los Ejes del Objeto. El despliegue volumétrico utilizado se introduce como una mejora en comparación con el despliegue de superficies utilizado en el sistema desarrollado por Tryggestad *et al.* [33] descrito en la sección 3.2. Los resultados de la entrevista señalaron que gracias al despliegue volumétrico de las imágenes DICOM, el posicionamiento del ME se hace más sencillo. Esto se logra debido a que es posible visualizar simultáneamente el volumen del cerebro, la ubicación de la lesión y el cráneo del paciente.

La aplicación es capaz de cargar y desplegar modelos 3D haciendo uso de archivos en formato OBJ. Gracias a esto se logró modelar a escala real cada una de las piezas del ME y se logró su despliegue junto con las imágenes del cráneo del paciente. Esto también permitió la configuración interactiva de todas las partes del ME (postes, marco, caja) y desplegar la información necesaria (en escala real) para que el usuario colo-

que correctamente el ME. Los resultados de la entrevista arrojaron que los usuarios se encontraron muy satisfechos con estas dos características de la aplicación desarrollada. Igualmente, esto hace posible que en el futuro se agreguen nuevos modelos de ME o piezas a la aplicación.

El sistema posee un módulo para la generación de marcas fiduciales sobre el conjunto de imágenes (*datasets*, en inglés) DICOM del paciente basadas en el posicionamiento virtual del marco ME. Los resultados de la entrevista indican que ésta característica permite generar un plan de tratamiento radioquirúrgico inicial estimado, basado exclusivamente en los estudios imagenológicos del paciente, lo que se traduce en una mejora significativa en términos de tiempo y recursos en casos donde la lesión sea de complicada ubicación.

Con la finalidad de lograr una alta fidelidad en la simulación del proceso de colocación del ME, todas las marcas fiduciales se generan utilizando una máscara circular, la cual indica cuales vóxeles deben ser marcados y el valor de intensidad que le corresponde a cada uno de ellos dependiendo de su posición dentro del círculo. Dicha característica permitió que el sistema de planificación radioquirúrgico (*Leksell Gamma-Plan*® [48]), donde se realizaron las pruebas, fuera capaz de detectar y procesar las marcas generadas por la aplicación.

Los resultados de la entrevista también indican que la información brindada en todo momento, la simplicidad existente en la interacción del usuario con todos los componentes e interfaces gráficas, los diferentes ángulos de percepción durante el posicionamiento del ME, la posibilidad de trabajar bajo el método heurístico de prueba y error, y el hecho de manipular una representación (en estructura y composición) del cráneo del paciente a tratar, son factores por los cuales se presenta la aplicación como una posible herramienta de entrenamiento o formación interactiva para el personal médico del área de radiocirugía, en especial para los neurocirujanos.

También se realizaron diferentes tipos de pruebas cuantitativas con el objetivo de determinar la precisión y exactitud de las marcas fiduciales generadas por la aplicación desarrollada en este trabajo. Las pruebas incluyeron:

- El cálculo del error basado en las distancias entre las marcas fiduciales reales y las generadas por el sistema: Los resultados indicaron que el promedio de la distancia existente fue de 0,8920 mm, sin diferenciar entre marcas. Esta cifra muestra que las marcas generadas poseen una gran similitud con respecto a las marcas verdaderas.
- El cálculo de los errores absolutos y relativos en las marcas fiduciales generadas en base a las medidas reales de los segmentos: Los resultados reflejan un error absoluto promedio de  $\pm 0,1897$  mm y un error relativo porcentual de 0,1272%, sin diferenciar el segmento medido. Estas cifras sugieren una alta precisión y exactitud en términos de localidad y longitud de las marcas fiduciales generadas por el sistema desarrollado.
- La introducción de los *datasets* creados al sistema de planificación radioquirúrgico: Para las muestras estudiadas se registró una desviación media de  $\pm 0,2667$  mm, similar a los resultados anteriores. Los resultados obtenidos indican que las estructuras de las marcas generadas pudieron ser procesadas por el sistema de planificación sin ningún tipo de problema.

## 6.2 Trabajos Futuros

El sistema puede ser adaptado a futuro para evitar las colisiones entre las piezas del ME y el volumen del paciente; esto es, incluir la implementación de un algoritmo de detección de colisiones entre polígonos y vóxeles. La detección de colisiones puede garantizar que las marcas fiduciales sean generadas correctamente en el espacio y no sobre algún tejido o parte del cráneo del paciente. Adicionalmente, se pudiera refinar el algoritmo para que señale la proximidad del ME con algún tejido cartilaginoso, como por ejemplo los pabellones auriculares, usando un umbral configurable desde la interfaz gráfica de la aplicación. Dicha opción evitaría que el ME incomode al paciente al momento de realizar el tratamiento.

Es importante complementar la aplicación con un algoritmo que sea capaz de detectar colisiones entre el ME y la máquina de radiocirugía a utilizar, en este caso el *Gamma Knife*. Este complemento sería de gran utilidad al momento de realizar una pre-planificación del tratamiento radioquirúrgico, ya que le indicaría al neurocirujano si el procedimiento es factible o si tendrá que suspenderse por colisiones inminentes.

La implementación desarrollada le permite al usuario la ubicación manual del blanco quirúrgico a través de la exploración de los distintos cortes del volumen (axial, coronal y sagital). Sin embargo, no existe ninguna herramienta que permita la localización automática de la lesión. Una solución sería incluir un algoritmo de segmentación de volúmenes, que sea capaz de localizar la lesión de forma transparente al momento de cargar el volumen al sistema.

La investigación realizada podría ser la base para el desarrollo de un sistema más complejo que permita la colocación virtual y física del ME sobre el cráneo del paciente. Para ello sería necesario que el sistema interactuara directamente con algún dispositivo de captura de imágenes, como por ejemplo: cámaras web, sensores de profundidad, etc. Dicho sistema registraría el movimiento del ME a través de los dispositivos de captura de imágenes e indicaría al usuario el error cometido en la colocación del ME en tiempo real.

La aplicación puede ser ajustada para permitir que el usuario guarde el estado actual del sistema; esto es, almacenar la posición, rotación y configuración del ME, mediciones realizadas, ubicación de la lesión, configuración de los *viewports*, etc. La aplicación sería capaz de guardar y cargar estos archivos de configuración y se manejarían como una especie de archivos de proyecto.

Se debe incluir un módulo que realice las mediciones del hueso del cráneo en la misma forma que se hace en la práctica real. Dicho módulo requiere el despliegue de un modelo geométrico que se coloque sobre el ME.

Se recomienda realizar la entrevista a una mayor cantidad de usuarios para un mejor análisis de los resultados cualitativos de la aplicación desarrollada.

Finalmente, para mejorar la percepción visual en el despliegue volumétrico, se puede incorporar un modelo de iluminación.



# **Referencias Bibliográficas**

---

- [1] Villegas R., Jara L., Montilla G., Bosnjak A., Villegas H., "Experiencias en la Planificación Asistida por Computadora de Neurocirugía Estereotáctica", *Bioingeniería en Venezuela: Tendencias, Propuestas y Avances*, pp. 244-247, 2008.
- [2] Leksell L., "Stereotactic radiosurgery", *Journal of Neurology, Neurosurgery & Psychiatry*, num. 46, pp. 797-803, 1993.
- [3] Alonso M., Austria J., "Historia de la cirugía estereotáctica", *Archivos de Neurociencias*, vol. 3, num. 8, pp. 158-165, 2003.
- [4] Madrazo I., Herreo A., "Radiocirugía Estereotáctica", *Cirugía y Cirujanos*, vol. 2, num. 73, pp. 137-141, 2005.
- [5] Forcht D., "Stereotactic Surgery", *Neurosurgery Clinics of North America*, vol. 1, num. 12, pp. 69-90, 2001.
- [6] Wiki de Neurocirugía. (2010) Estereotáxia. [En línea] <http://tinyurl.com/3lxm9y2>.
- [7] Wikipedia. (2012) Radiocirugía. [En línea] <http://tinyurl.com/3nudns2>
- [8] Leksell L., "The stereotactic method and radiosurgery of the brain", *Acta Chirurgica Scandinavica*, vol. 4, num. 102, pp. 316-319, 1951.
- [9] Friedman W., Bova F., Spiegelmann R., "Linear accelerator radiosurgery at the University of Florida", *Neurosurgery Clinics of North America*, num. 3, pp. 140-166, 1992.
- [10] University of Maryland Medical Center. (2011) Gamma Knife Center Patient Success Stories. [En línea] <http://tinyurl.com/5trqu8t>
- [11] Betti O., Derechinsky V., "Hyperselective encephalic irradiation with linear accelerator", *Acta Neurochirurgica Supplementum (Wien)*, num. 33, p. 385-390, 1984.
- [12] CyberKnife Institute at Mercy Hospital. (2011) La Historia del CyberKnife. [En línea] <http://tinyurl.com/5rlw69m>
- [13] Lookfordiagnosis. (2009) Imágenes Radiocirugía. [En línea] <http://tinyurl.com/629om9z>
- [14] ACNefi Informa. (2002) Radioterapia Estereotáctica. [En línea] <http://tinyurl.com/3qtu44q>
- [15] Conde J., González J., Castellote J., "Cirugía Estereotáctica: Indicaciones y situación en España.", *Agencia de Evaluación de Tecnologías Sanitarias (AETS), Ministerio de Sanidad y Consumo - Instituto de Salud Carlos III*, 1997.
- [16] Taringa: Inteligencia colectiva. (2010) Body Gamma Knife ¿Fraude o Realidad?. [En línea] <http://tinyurl.com/5soxcd8>

- [17] RadiologyInfo.org. (2011) Bisturí de rayos gamma (Gamma Knife). [En línea] <http://tinyurl.com/68y492s>
- [18] Gamma Knife Center at Beaumont. (2009) Gamma Knife Overview. [En línea] <http://tinyurl.com/44f3s6u>
- [19] Northridge Hospital. (2011) ¿Qué es el bisturí de rayos gamma?. [En línea] <http://tinyurl.com/6ghcrsg>
- [20] American Academy of Family Physicians. (2008) Recent Advances in Radiation Therapy. [En línea] <http://tinyurl.com/3fqpkmb>
- [21] UNICAMP. (1997) Radiosurgery. [En línea] <http://tinyurl.com/5ttyvah>
- [22] Minnesota Gamma Knife Center. (2011) Stereotactic Radiosurgery. [En línea] <http://tinyurl.com/6cg8n9g>
- [23] Hassen-Khodja, R., "Gamma Knife and Linear Accelerator Stereotactic Radiosurgery", 2004.
- [24] University of Virginia. (2011) Stereotactic Surgery. [En línea] <http://tinyurl.com/3l57gbr>
- [25] Revista Avances. (2011) Cirugía Estereotáctica o Estereotáxica. [En línea] <http://tinyurl.com/3rgze48>
- [26] Mayo Clinic. (2010) Stereotactic Head Frame. [En línea] <http://tinyurl.com/6ers2wd>
- [27] Calcerrada N., Sabés R., "Efectividad, seguridad y estimación de costes del sistema de radio cirugía Cyberknife", *Madrid: Unidad de Evaluación de Tecnologías Sanitarias (UETS)*, 2005.
- [28] BrainLab. (2011) Stereotactic Planning Software. [En línea] <http://tinyurl.com/678vk3f>
- [29] Aristu, J., "Radioterapia estereotáctica", *Anales del Sistema Sanitario de Navarra*, vol. 2, num. 32, pp. 61-71, 2009.
- [30] Lara J., Hernández D., Alonso V., Austria V., "Desarrollo de un aparato estereotáctico con el sistema de arco centrado", *Archivos de Neurociencias*, vol. 3, num. 10, pp. 196-202, 2005.
- [31] Lay M., Movahed A., Ahmad S., Mardirossian G., "Prediction and Optimization of Stereo Tactic Frame Placement for Collision Avoidance in Gamma Knife Radio Surgery", *Medical Physics*, vol. 6, num. 32, p. 2044, 2005.
- [32] Nieto M., Jesús J., "Sistema para centrar el marco estereotáctico en el cráneo", *Archivos de neurociencias*, vol. 4, num. 11, pp. 310-312, 2006.
- [33] Tryggestad E., Christodouleas J., McNutt T., Kleinberg L., Purger D., Daniele R., Wong J., Ford E., "A Virtual Frame System for Stereotactic Radiosurgery Planning", *Medical Physics*, num. 34, p. 2413, 2007.
- [34] Philips. Pinnacle3. [En línea] <http://tinyurl.com/68e6zlh>

- [35] Wiant D., Bourland J., "Simulated gamma knife head frame placement for radiosurgical pre-planning", *Technology in cancer research & treatment*, vol. 4, num. 8, pp. 265-270, 2009.
- [36] Universal Medical Inc. RSD Anthropomorphic Full Angiographic Head Phantom. [En línea] <http://tinyurl.com/6h5xywt>
- [37] openDICOM.NET. (2012) [En línea] <http://opendicom.sourceforge.net/>
- [38] ClearCanvas Inc. (2012) ClearCanvas. [En línea] <http://tinyurl.com/yb4lthp>
- [39] Microsoft. (2012) Biblioteca de procesamiento paralelo basado en tareas (TPL). [En línea] <http://tinyurl.com/97nhnpy>
- [40] Carmona R., Fröhlich B., "Visualización Multi-Resolución de Volúmenes de Gran Tamaño", *Universidad Central de Venezuela*, 2008.
- [41] Google. (2012) Google Sketchup. [En línea] <http://tinyurl.com/yrrdh8>
- [42] Google. (2012) Exportación de modelos 3D. [En línea] <http://tinyurl.com/96qegez>
- [43] Wikipedia. (2012) Vertex Buffer Object. [En línea] <http://tinyurl.com/yf7hgn6>
- [44] Song Ho Ahn. (2005) OpenGL Display List. [En línea] <http://tinyurl.com/7q3zud5>
- [45] Sabia. ¿Qué son los shaders?. [En línea] <http://tinyurl.com/c2jbmta>
- [46] Synedra Information Technology. (2012) synedra View. [En línea] <http://tinyurl.com/8vkkez6>
- [47] Agnosco. (2011) Agnosco Free DICOM Viewer. [En línea] <http://tinyurl.com/8gfzbtk>
- [48] Elektra. (2012) Leksell GammaPlan®. [En línea] <http://tinyurl.com/8tq9cxy>
- [49] OpenGL.org. (1991-2006) glReadPixels. [En línea] <http://tinyurl.com/3638g7a>



# **Glosario de Términos**

---

## - A -

**Acelerador lineal:** dispositivo eléctrico para la aceleración de partículas subatómicas que posean carga eléctrica, tales como electrón, positrón, protón o ion.

**Aliasing:** artefacto en la representación de los gráficos debido a limitaciones en la resolución final de una imagen.

**Anestesia local:** uso de medicaciones llamadas anestésicos para producir una pérdida temporal de sensación en un área específica del cuerpo durante un procedimiento quirúrgico o médico de otra clase. Si bien el área local inyectada con el anestésico se adormece, el paciente sigue despierto y responsivo.

**Angiografía Digital (AD):** examen de diagnóstico por imagen cuya función es el estudio de los vasos circulatorios que no son visibles mediante la radiología convencional.

**Antialiasing:** cualquier técnica que reduce la apariencia accidentada (*aliasing*) de los bordes en imágenes digitales tanto planas como en tres dimensiones. La apariencia accidentada suele deberse al alto contraste entre los píxeles de los bordes y sus alrededores.

**API (Application Programming Interface - Interfaz de Programación de Aplicaciones):** es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizada por otro software como una capa de abstracción. Son usados generalmente en las bibliotecas.

**Axis-Aligned Bounding Box:** es un *bounding box* alineado con los ejes espaciales X, Y y Z.

## - B -

**Biblioteca (Library):** es un conjunto de subprogramas utilizados para desarrollar software. Las bibliotecas contienen código y datos, que proporcionan servicios a programas independientes, es decir, pasan a formar parte de éstos. Esto permite que el código y los datos se compartan y puedan modificarse de forma modular.

**Bisturí de Rayos Gamma:** ver *Gamma Knife*.

**Boolean:** tipo de dato lógico que puede representar valores de lógica binaria, esto es 2 valores, normalmente falso o verdadero.

**Bounding Box:** recuadro de selección. Es el rectángulo más pequeño que encierra completamente a un objeto.

## - C -

**Calota:** bóveda del cráneo que está formada por los huesos parietales y la región superior del frontal.

**Ciclotrón:** tipo de acelerador de partículas en el cual las partículas cargadas son propulsadas por un campo eléctrico alterno entre dos grandes electrodos, en un campo magnético constante creado por dos imanes grandes. Las partículas son inyectadas en el centro del imán y salen en forma de espiral a medida que aumenta la energía. Los protones producidos en un ciclotrón pueden ser usados para tratar el cáncer, y los protones producidos por ciclotrón pueden crear radioisótopos para procedimientos médicos nucleares.

**Cobalto (Co):** Elemento metálico color acero, de número atómico 27 y peso atómico 58,93320 situado en el grupo 9 de la tabla periódica de los elementos.

**Cobalto 60 (<sup>60</sup>Co):** es un isótopo radiactivo del cobalto, con una vida media de 5,27 años. Decae por "desintegración beta" al isótopo estable níquel-60 (<sup>60</sup>Ni). En el proceso de desintegración, <sup>60</sup>Co emite un electrón con una energía de 315 keV y luego dos rayos gammas con energías de 1,17 y 1,33 MeV, respectivamente.

**Colimador:** instrumento de precisión para una tarea especial. El objetivo de la colimación es hacer que el eje óptico de cada lente o espejo coincida con el rayo central del sistema.

**Coordenadas de ventana:** es la posición de los puntos en un formulario o ventana. El origen de las coordenadas de ventana está en la esquina superior izquierda del área de la ventana. Las coordenadas de ventanas aseguran que una aplicación puede usar valores de coordenadas coherentes mientras se dibuja en una ventana o formulario, sin tener en cuenta la posición del formulario o de la ventana en la pantalla.

**Coordenadas Estereotáxicas:** es un sistema de coordenadas cartesianas tridimensionales X, Y y Z, que sirven

para localizar pequeñas estructuras o puntos concretos dentro del cuerpo.

**C-Sharp (C#):** es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET.

## - D -

**DICOM (Digital Imaging and Communication in Medicine – Imágenes Digitales y Comunicación en Medicina):** es el estándar reconocido mundialmente para el intercambio de imágenes médicas, pensado para el manejo, almacenamiento, impresión y transmisión de imágenes médicas.

**Dosimetría:** ciencia y técnica para determinar la dosis de radiación.

## - E -

**Electrón:** partícula subatómica con carga eléctrica negativa y que gira alrededor del núcleo atómico. El flujo de electrones constituye la corriente eléctrica.

**Estereotáxia:** es una técnica que consiste en definir un punto o volumen en un espacio determinado, según su posición en un eje de coordenadas cartesianas X, Y, Z, en relación con un sistema de referencia conocido.

## - F -

**Fotón:** es un "paquete" de energía sin masa, que viaja a la velocidad de la luz y su comportamiento físico se caracteriza por la cantidad de energía que transporta, independiente del origen que tengan.

**Frame Buffer:** es la porción de memoria reservada para mantener temporalmente una imagen ráster a la espera de ser enviada al monitor o a un dispositivo. Es usado para compensar las diferentes tasas de flujo de datos entre los componentes de una computadora.

**Framework:** conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar. Estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de *software* concretos,

con base en la cual otro proyecto de *software* puede ser organizado y desarrollado.

## - G -

**Gamma Knife (GK):** bisturí de rayos gamma. Es un aparato usado para tratar tumores cerebrales mediante la administración de radiación en alta intensidad de una manera que concentra la radiación sobre un pequeño volumen.

**Gantry:** coraza que aloja el tubo de Rayos-X, los colimadores y los detectores de una máquina de TAC o de radioterapia, con una abertura grande en donde se introduce al paciente. Soporte mecánico para montar un dispositivo y moverlo en círculos.

**GLSL:** ver *OpenGL Shading Language*.

**GPU:** ver Unidad de Procesamiento Gráfico.

## - H -

**Helper, Helper Class:** es una técnica de programación en la programación orientada a objetos. Es un término dado a las clases que se utilizan para ayudar a proporcionar alguna funcionalidad, a pesar de que la funcionalidad no es el propósito principal de la aplicación.

## - I -

**Interfaz de Usuario:** es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo, normalmente suelen ser fáciles de entender y fáciles de accionar.

**Interfaz Gráfica de Usuario:** es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una computadora.

## - M -

**Marcas fiduciales:** son objetos visibles durante el tratamiento que permiten determinar la posición del tumor y

cualquier órgano próximo permitiendo dirigir la radiación con precisión.

**Marco Estereotáctico (ME):** dispositivo o guía tridimensional que se ajusta a su cabeza.

**Megavoltio (mv):** un millón de voltios.

**Metodología de desarrollo de software:** es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información.

**Métodos imagenológicos:** conjunto de técnicas y procesos usados para crear imágenes del cuerpo humano, o partes de él, con propósitos clínicos (procedimientos médicos que buscan revelar, diagnosticar o examinar enfermedades) o para la ciencia médica (incluyendo el estudio de la anatomía normal y función).

**Microsoft .NET Framework:** es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

**Mono:** es un proyecto de implementación del Framework .NET de Microsoft utilizando código libre, gestionado por *Ximian* y basado en las especificaciones definidas en ECMA.

## - O -

**OBJ:** es un formato de archivo para la definición de geometrías desarrollado por *Wavefront Technologies* para su paquete de animación *Advanced Visualizer*. Es un formato de archivo abierto y ha sido adoptado por otros proveedores de aplicaciones de gráficos 3D. Es universalmente aceptado y representa una geometría 3D. En este tipo de archivos se almacenan datos como la posición de cada vértice, la posición de cada coordenada de textura, las normales y las caras que representan los polígonos de la geometría almacenada.

**Occipital:** hueso de la parte posterior e inferior del cráneo.

**OpenGL:** es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D. La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos.

**OpenGL Shading Language (GLSL - Lenguaje de Sombreado de OpenGL):** también conocido como *GLSLang*, una tecnología parte del API estándar *OpenGL*, que permite especificar segmentos de programas gráficos que serán ejecutados sobre el *GPU*.

**OpenTK:** Es una biblioteca de C# que permite a aplicaciones *.NET* y *Mono* hacer uso de *OpenGL*, *OpenAL* y *OpenCL*. Es intuitiva, rápida, multiplataforma y usable por todos los lenguajes *.NET* y *Mono*.

## - P -

**Patológico:** enfermo; causado por una enfermedad.

**Programación Orientada a Objetos (POO):** es un paradigma de programación que usa objetos y sus interacciones, para diseñar aplicaciones y programas informáticos. Está basado en varias técnicas, incluyendo herencia, abstracción, polimorfismo y encapsulamiento.

**Protones:** partícula de carga positiva que es un componente fundamental del núcleo de todos los átomos.

**PVC (PoliCloruro de Vinilo - PolyVinyl Chloride):** Se presenta como un material blanco que comienza a reblandecer alrededor de los 80 °C y se descompone sobre 140 °C. Cabe mencionar que es un polímero por adición y además una resina que resulta de la polimerización del cloruro de vinilo o cloroeteno.

## - R -

**Radiación:** dispersión en todas las direcciones desde el centro. Emisión de luz, ondas cortas de radio, rayos ultravioleta o X, o cualquier otro tipo de rayo para tratamiento, diagnóstico u otros fines. Energía radiante de ondas o partículas subatómicas.

**Radiación gamma:** ver rayos gamma.

**Radiación ionizante:** toda aquella emisión que posee la energía suficiente para remover electrones de átomos o moléculas generando iones capaces de interactuar y producir cambios en la materia.

**Radiosensibilidad:** sensibilidad que tienen los diferentes tejidos y células a las radiaciones ionizantes.

**Radioterapia:** especialidad médica que se ocupa del uso de radiación electromagnética o de partículas para el tratamiento de las enfermedades.

**Radioterapia Estereotáctica Fraccionada (REF):** forma de radiocirugía estereotáctica que usa una dosis de radiación fraccionada, es decir, una dosis más pequeña a lo largo de varios días o semanas, en vez de una única dosis grande, para reducir al mínimo el daño al tejido.

**Rayos gamma:** también llamado radiación gamma. Radiación electromagnética de altísima frecuencia que consiste en fotones emitidos por elementos radiactivos. Los rayos gamma pueden lesionar y destruir células y tejidos del cuerpo, especialmente los núcleos celulares.

**Resonancia Magnética (RM):** es una técnica no invasiva que utiliza el fenómeno de la resonancia magnética para obtener información sobre la estructura y composición del cuerpo a analizar. Esta información es procesada por ordenadores y transformada en imágenes del interior de lo que se ha analizado.

## - S -

**SDK:** ver *Software Development Kit*.

**Shader:** es un procedimiento de sombreado e iluminación que permite al artista/programador especificar el renderizado de un vértice o de un píxel. Es cualquier unidad escrita en un lenguaje de sombreado que se puede compilar independientemente. Es una tecnología destinada a proporcionar al programador la interacción con la GPU. Los *shaders* son utilizados para realizar transformaciones y crear efectos especiales, como por ejemplo iluminación, fuego o niebla. Para su programación los *shaders* utilizan lenguajes específicos de alto nivel que permitan la independencia del hardware.

**Software Development Kit (SDK):** un *kit* de desarrollo de *software* o *SDK* es generalmente un conjunto de herramientas de desarrollo de *software* que le permite al programador crear aplicaciones para un sistema concreto, por ejemplo ciertos paquetes de *software*, *frameworks*, plataformas de *hardware*, computadoras, videoconsolas, sistemas operativos, etc.

**Sincrotrón:** ver *ciclotrón*.

## - T -

**Tomografía Axial Computarizada (TAC):** formación de una imagen con información anatómica obtenida de un corte transversal del cuerpo. Cada imagen es producida por la síntesis computarizada de los datos de transmisión radiográfica de muchas direcciones diferentes sobre un plano determinado.

**Toolkit:** es un conjunto coherente de herramientas que, bajo una misma interfaz, provee de funcionalidades extensibles y versátiles.

## - U -

**Unidad de Procesamiento Gráfico (GPU - *Graphics Processing Unit*):** es un coprocesador dedicado al procesamiento de gráficos u operaciones de coma flotante, para aligerar la carga de trabajo del procesador central en aplicaciones como los videojuegos y o aplicaciones 3D interactivas.

## - V -

**Volume Rendering (Despliegue de Volúmenes):** proceso a través del cual se puede realizar la visualización de datos provenientes de un campo escalar discreto, sin necesidad de reconstruir la geometría del objeto en estudio, haciendo una simulación aproximada de la propagación de la luz a través de un medio representado por el volumen.

**Vóxel (*Volumetric Pixel*):** es la unidad cúbica que compone un objeto tridimensional. Constituye la unidad mínima procesable de una matriz tridimensional y es, por tanto, el equivalente del píxel en un objeto 3D.