

**RECONSTRUCCIÓN TRIDIMENSIONAL A PARTIR DE IMÁGENES
MÉDICAS APLICANDO LAS TÉCNICAS DE *SURFACE RENDERING* Y
*VOLUME RENDERING***

JOSÉ ENRIQUE GÓMEZ MIER

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA
2011**

**RECONSTRUCCIÓN TRIDIMENSIONAL A PARTIR DE IMÁGENES
MÉDICAS APLICANDO LAS TÉCNICAS DE *SURFACE RENDERING* Y
*VOLUME RENDERING***

JOSÉ ENRIQUE GÓMEZ MIER

**Trabajo de grado para optar al título de
Ingeniero Electrónico**

Director

Msc. EDWIN ALBERTO SILVA CRUZ

Codirector

Msc. CÉSAR DUARTE GUALDRÓN

Codirector

Phd. JAIME MENESES

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA
2011**

Los obstáculos son esas cosas que las personas
ven cuando dejan de mirar sus metas.

E. JOSEPH COSSMAN.

No hay nada imposible
para quién sabe esperar

ANÓNIMO.

En este mundo solo hay dos cosas infinitas: el universo y la estupidez
humanay del universo aún no estamos muy seguros.

ALBERT EINSTEIN.

DEDICATORIA

Ya hace unos cuantos años que entré a la Universidad (más de los que me gustaría), con el propósito de llegar a ser ingeniero. Para cuando empecé con mis estudios parecía una meta muy lejana, pero por fin ha llegado el momento de graduarme como Ingeniero. Tras todos estos años de estudios, este proyecto es el encargado de poner fin a esta etapa de mi vida.

Quiero dedicar este proyecto a Dios por ser la luz que ilumina y guía mi camino, a mi madre Ana por su esfuerzo y dedicación, a la memoria de mi padre Enrique, a mi esposa Liliana por su motivación, por ser mi pilar y darle dirección a mi vida, a mi hijo Daniel por ser el motivo de seguir sin desfallecer, a mis hermanas Tulia y Carolina por su ayuda y colaboración incondicional, y a todas las personas que me han apoyado durante estos años.

GRACIAS A TODOS

José Enrique

AGRADECIMIENTOS

La realización de este proyecto hubiese sido imposible sin la ayuda directa o indirecta de otras personas, especialmente agradezco a mi director de proyecto Msc. Edwin Alberto Silva Cruz quién me ha guiado durante todo el desarrollo del mismo con paciencia. Gracias por su dedicación y apoyo ya que sin su dirección y supervisión este proyecto no hubiera sido posible.

También agradezco a Jorge Humberto Echeverri Perico, M.D., por todo su tiempo y colaboración. A mi calificador, PhD. Arturo Plata, por la paciencia e invaluable aportes para la presentación y terminación del mismo.

CONTENIDO

	Pág.
INTRODUCCIÓN	17
1 GENERALIDADES DEL PROYECTO	19
1.1 RESUMEN DEL PROYECTO -----	19
1.2 DESCRIPCIÓN DEL PROYECTO -----	19
2. OBJETIVOS	21
2.1 OBJETIVO GENERAL -----	21
2.2 OBJETIVOS ESPECÍFICOS -----	21
3. PROCESAMIENTO DE LAS IMÁGENES	22
3.1 CORTE DE TEJIDO BIOLÓGICO -----	22
3.1.1 OBTENCIÓN DE IMÁGENES POR MEDIO DE CORTÉS DE TEJIDO BIOLÓGICO -----	22
3.2 PRE-SEGMENTACIÓN -----	24
3.2.1 ECUALIZACIÓN DEL HISTOGRAMA -----	24
3.2.2 ESPECIFICACIÓN DEL HISTOGRAMA -----	24
3.2.3 TÉCNICAS DE REDUCCIÓN DEL RUIDO -----	25
3.3 SEGMENTACIÓN DE IMÁGENES -----	28
3.3.1 SEGMENTACIÓN BASADA EN EL USO DE UN UMBRAL -----	28
3.3.2 SEGMENTACIÓN POR DETECCIÓN DE BORDES -----	31
3.3.3 ALGORITMOS DE CLÚSTERING -----	35
3.3.4 SEGMENTACIÓN POR WATERSHED -----	37
3.4 POST-SEGMENTACIÓN -----	40
4. VISUALIZACIÓN TRIDIMENSIONAL A PARTIR DE IMÁGENES	41
4.1 REDIMENSIONAMIENTO DE LA SECUENCIA DE DATOS -----	41
4.2 MÉTODOS DE SURFACE RENDERING -----	41
4.2.1 MARCHING CUBES -----	42
4.2.2 POST-PROCESAMIENTO DE SUPERFICIES -----	47
4.3 MÉTODOS DE VOLUME RENDERING -----	47

4.3.1 VOLUME RENDERING BASADO EN TEXTURAS-----	48
4.4 MANIPULACIÓN DE LA VISUALIZACIÓN-----	52
4.4.1 NAVEGACIÓN SOBRE LA VISUALIZACIÓN-----	52
4.4.2 CORTE SOBRE LA VISUALIZACIÓN-----	54
5. METODOLOGÍA Y RESULTADOS	58
5.1 DESCRIPCIÓN DEL EQUIPO USADO-----	58
5.2 ESQUEMA DE BASICO DEL FUNCIONAMIENTO DE LA APLICACIÓN-----	58
5.2.1 ETAPA INICIAL-----	59
5.2.2 ETAPA DE VISUALIZACION TRIDIMENSIONAL-----	60
5.2.3 ETAPA DE MANIPULACIÓN-----	61
5.3 PRUEBAS REALIZADAS-----	63
6. CONCLUSIONES Y RECOMENDACIONES	82
6.1 FUTURAS MEJORAS-----	84
6.2 FUTUROS TRABAJOS-----	85
BIBLIOGRAFÍA	86
ANEXOS	92

LISTADO DE FIGURAS

	Pág.
Figura 1 Imágenes pertenecientes al Proyecto Humano Visible.....	22
Figura 2 Imágenes de corte realizados al corazón de un ratón infectado con el virus de Chagas.....	23
Figura 3 Imágenes de un corazón humano introducido en un bloque de parafina.	23
Figura 4 Ejemplo de realce de la imagen por medio de la ecualización del histograma.....	25
Figura 5 Aplicación de un filtro promedio.....	26
Figura 6 Aplicación del filtro de la mediana.....	27
Figura 7 Aplicación del filtro Wiener.....	27
Figura 8 Resultados de la umbralización con los distintos métodos de umbral. ...	30
Figura 9 Aplicación del gradiente de Robert.....	32
Figura 10 Aplicación del gradiente de Prewitt.....	32
Figura 11 Aplicación del gradiente de Sobel.....	33
Figura 12 Aplicación del Laplaciano.....	34
Figura 13 Aplicación del gradiente del Laplaciano.....	34
Figura 14 Aplicación del filtro Canny.....	35
Figura 15 segmentacion por el algoritmo de Kmeans.....	36
Figura 16 Segmentacion por algoritmo de Fcm.....	37
Figura 17 Ejemplo de la forma como es interpretada una imagen por medio de la transformada Watershed.....	37
Figura 18 Etapas en el algoritmo de Watershed.....	38
Figura 19 Ejemplo de la segmentacion por el método de Watershed con sobresegmentación.....	39
Figura 20 Formación de un cubo lógico a partir de dos capas.....	42

Figura 21 Cálculo del índice que define la colocación del cubo con respecto al volumen.	43
Figura 22 Índice de colocación del cubo.	43
Figura 23 Casos de intersección de un cubo con la superficie.	44
Figura 24 Filtro de suavizado.	45
Figura 25 Casos complementarios del algoritmo de <i>Marching Cubes</i>	46
Figura 26 Etapa de Post-Procesamiento de Superficies.	47
Figura 27 Mapeo de los cortes a los polígonos (planos).	48
Figura 28 Esquema de los tres métodos de texturas.	49
Figura 29 Visualización del volumen de datos por el método de texturas 2D.	50
Figura 30 Visualización del volumen de datos	52
Figura 31 Ejemplo de rotación de la visualización tridimensional.	53
Figura 32 Ejemplo de zoom en la visualización tridimensional.	53
Figura 33 Ejemplo de desplazamiento de la visualización tridimensional.	54
Figura 34 Plano de corte.	54
Figura 35 Imágenes obtenidas al realizar un corte sobre la visualización tridimensional.	57
Figura 36 Esquema Utilizado Para El Funcionamiento De La Aplicación.	59
Figura 37 Secuencia De Imágenes De Cortes Corонаles De Una Cabeza.	63
Figura 38 Visualización tridimensional por el método de <i>surface rendering</i> de la secuencia de imágenes de la Figura 36.	65
Figura 39 Cortes arbitrarios sobre la visualización de la figura 37.	66
Figura 40 Visualización tridimensional por el método de <i>Volume Rendering</i> de la secuencia de imágenes de la Figura 36.	67
Figura 41 Secuencia De Imágenes De Cortes axiales De Un Cuerpo Humano. ..	67
Figura 42 Visualización tridimensional por el método de <i>surface rendering</i> de la secuencia de imágenes de la Figura 40.	69
Figura 43 Cortes arbitrarios sobre la visualización de la figura 41.	70
Figura 44 Visualización tridimensional por el método de <i>Volume Rendering</i> de la secuencia de imágenes de la Figura 40:	71

Figura 45	Secuencia De Imágenes De Cortes Axiales De Una Cabeza.....	71
Figura 46	Visualización tridimensional por el método de <i>surface rendering</i> de la secuencia de imágenes de la Figura 44.	72
Figura 47	Cortes arbitrarios sobre la visualización de la figura 45.....	73
Figura 48	Visualización tridimensional por el método de <i>Volume Rendering</i> de la <i>secuencia de imágenes de la Figura 44:</i>	74
Figura 49	Secuencia De Imágenes De Cortes Coronales De Una Parte De La Cabeza.	74
Figura 50	Visualización tridimensional por el método de <i>surface rendering</i> de la secuencia de imágenes de la Figura 48.	75
Figura 51	Cortes arbitrarios sobre la visualización de la figura 49.....	76
Figura 52	Visualización tridimensional por el método de <i>Volume Rendering</i> de la <i>secuencia de imágenes de la Figura 48:</i>	77
Figura 53	Secuencia De Imágenes De Cortes Axiales De Una Cabeza.....	77
Figura 54	Visualización tridimensional por el método de <i>surface rendering</i> de la secuencia de imágenes de la Figura 52	78
Figura 55	Cortes arbitrarios sobre la visualización de la figura 53.....	79
Figura 56	Visualización tridimensional por el método de <i>Volume Rendering</i> de la <i>secuencia de imágenes de la Figura 52</i>	80
Figura 57	Imagen de la aplicación desarrollada en la interfaz de visualización tridimensional.....	80
Figura 58	Imagen de la aplicación desarrollada en la interfaz de corte	81

LISTA DE ANEXOS

	Pág.
ANEXO A. MANUAL DE USUARIO	92

RESUMEN

TITULO: VISUALIZACIÓN TRIDIMENSIONAL A PARTIR DE IMÁGENES MÉDICAS APLICANDO LAS TÉCNICAS DE *SURFACE RENDERING* Y *VOLUME RENDERING**.

AUTOR: JOSÉ ENRIQUE GÓMEZ MIER**

PALABRAS CLAVES: Isosuperficie, Mri, Pet, *Rendering*, Segmentación, *Surface Rendering*, Visualización, *Volume Rendering*.

DESCRIPCION:

El avance en las últimas décadas en el área de generación, procesamiento y análisis de las imágenes médicas ha tenido un gran impacto en el ejercicio de la medicina. En efecto, la posibilidad de acceder visualmente al interior del cuerpo humano y observar aspectos estructurales y funcionales de los distintos órganos y sistemas, sin requerir de una intervención quirúrgica, es una realidad lograda.

El avance obtenido en el manejo de imágenes medicas a nivel mundial ha generado importantes beneficios para el paciente y para la sociedad, disminuyendo el tiempo real del diagnostico y mejorando la eficiencia y los costos operativos en la ejecución de tratamientos.

Por otra parte, es lamentable que en nuestro país no existan las condiciones adecuadas para el desarrollo de este tipo de procesamiento. Además, los programas existentes que manejan nuevas tecnologías de hardware y permiten mejores análisis en las regiones a estudiar son muy costosos, ya que son programas desarrollados en el exterior cuyas licencias requieren de una gran inversión y los hospitales no cuentan con los recursos para su adquisición.

Como consecuencia a lo inmediatamente anterior, se busca por medio del manejo de imágenes tridimensionales, desde nuestras propias investigaciones, ayudar a los médicos en la realización de diagnósticos y determinación de terapias más precisas al poder acceder visualmente al interior del cuerpo humano, sin requerir de una intervención quirúrgica.

Para lograr un apoyo efectivo al personal médico, este proyecto propone investigar y desarrollar tecnología matemático-computacional que guie a través de imágenes el diagnóstico, la escogencia de tratamientos terapéuticos y el seguimiento en la mejoría de los pacientes. Se elaborará un algoritmo que permita generar imágenes tridimensionales a partir de cortes bidimensionales o imágenes médicas por medio de los métodos de visualización tridimensional más usados (*surface rendering* y *volume rendering*).

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería Eléctrica, Electrónica y telecomunicaciones. Director Msc. Edwin Alberto Silva Cruz.

SUMMARY

TITLE: TRIDIMENSIONAL VISUALIZATION FROM MEDICAL IMAGES APPLYING SOME TECHNIQS AS *SURFACE RENDERING* AND *VOLUME RENDERING**.

AUTHOR: JOSÉ ENRIQUE GÓMEZ MIER**

MAIN WORDS: Isosurface, Mri, Pet, *Rendering*, Segmentation, *Surface Rendering*, Visualization, *Volume Rendering*.

DESCRIPTION:

The advance in the last decades in the generation area, processing and analysis of medical images has had a great impact in the medicine. In fact, the possibility to access visually inside the human body and can observe structural and functional aspects of the different organs and systems, without using a surgical intervention, it is a reality achieved.

The advance gave in the medical images management around the world has created relevant benefits to the patient and the society, decreasing the real time for the diagnostic and improving the efficiency and the operative costs in treatments.

In other hand, it is a pity that our country doesn't have the adequate conditions for the development of this kind of process. Besides, the programs that manage new technologies of hardware and permit best analysis in the regions are expensive, So they are developed programs abroad which licenses needs investing and the hospitals don't have the resources to adquire them.

Consequently, through the management of tridimensional images, from our own researches, to help the doctors on doing diagnostics and resolution of specified therapies to access visually inside the human body, without using a surgical intervention.

To ensure effective support to medical staff, this project proposes to research and develop math-computation technology that directs the diagnostic through images, the therapeutic treatments and the following on the patients. It will make an algorithm that permits create tridimensional images from bidimensional cuttings or medical images through visualization tridimensional second hand methods (surface rendering and volume rendering).

* Degree work

** School of Mechanical and Physical Engineering, School of Electrical Engineering, Electronics and Telecommunications. Director Msc. Edwin Alberto Silva Cruz.

GLOSARIO

Isosuperficie: Superficie que representa un valor constante de una función escalar.

Mri: Imagen de Resonancia Magnética, es una técnica de extracción de imágenes de organismos vivos y su fundamento funcional reside en la excitación de los núcleos de hidrógeno en las moléculas de agua, cambiando la orientación de su campo magnético. Al hacerlo, las moléculas son detectadas obteniéndose el valor del campo para su posterior procesamiento y generación de las imágenes correspondientes.

Pet: Tomografía de emisión de positrones. Es una técnica de medicina nuclear que genera una imagen tridimensional de tejidos, usando como principio, la detección de los fotones de rayos gamma que resultan de la colisión de los positrones contra los electrones.

Rendering: Rendición visual de un modelo a partir de un conjunto de datos de entrada.

Segmentación: Procesado digital que consiste en reconocer de forma automática los objetos de una escena sin ninguna intervención previa por parte del usuario.

Surface Rendering: Consiste en la visualización de superficies de un volumen de Vector Normal: Es el vector que es perpendicular al plano tangente de una superficie sobre un punto.

Visualización: Es el proceso de explorar, transformar y ver los datos como imágenes (u otras formas sensoriales) para ganar conocimiento acerca de estos datos.

Volume Rendering: Proceso a través del cual, se puede hacer la visualización de datos provenientes de un campo escalar discreto sin necesidad de reconstruir la geometría del objeto en estudio.

INTRODUCCIÓN

La aplicación de la ingeniería electrónica en cada una de las ramas de la ciencia hoy en día es más frecuente, ejemplo de ello lo demuestra la Bioingeniería. Sin embargo, es común encontrar en la medicina procesos obsoletos que se realizan de forma manual y en jornadas extensas que conllevan a la posibilidad de errores humanos.

En la búsqueda por reducir los errores humanos y contribuir al desarrollo tecnológico de nuestro país, se ha abordado este trabajo que consiste en el Diseño de un software que permite hacer una visualización tridimensional a partir de imágenes médicas, aplicando las técnicas de *Surface Rendering* y *Volume Rendering*. La finalidad perseguida se presenta en el disponer de sistemas computacionales capaces de realizar esta operación de forma más eficiente y presentar un buen resultado haciendo uso de las capacidades gráficas avanzadas que proporciona hoy la informática.

En la actualidad, las imágenes médicas obtenidas por medio de métodos tales como tomografía Computarizada (TC), Resonancia Magnética (RM) y Tomografía Computarizada de Emisión de Fotón Único (SPECT), sirven de gran ayuda a los médicos en los diagnósticos de sus pacientes, pero tienen un gran inconveniente debido a que no se ven los órganos en su forma original.

Sin embargo, como acercamiento a ver órganos en forma original existe la posibilidad de crear una imagen bidimensional con un punto de vista tridimensional arbitrario utilizando múltiples imágenes de cortes transversales. Para lo anterior, primero se deben crear las imágenes de los cortes transversales: estas imágenes se pueden crear con las técnicas anteriormente citadas (TC; RM; SPECT) luego, utilizando alguna de las técnicas de visualización tridimensional, obtenemos la visualización volumétrica del objeto tridimensional.

En general puede decirse, que el tratamiento de imágenes ayuda a los médicos a comprender la compleja anatomía humana. Por lo cual la ingeniería electrónica concede un apoyo tecnológico a la rama de la salud a través de las superficies tridimensionales de las imágenes médicas, lo cual ofrece una valiosa herramienta directa para:

- Realizar Diagnósticos

- Realizar Planificaciones
- simulaciones quirúrgicas

1 GENERALIDADES DEL PROYECTO

1.1 RESUMEN DEL PROYECTO

Durante el ciclo del diagnóstico médico del paciente es preciso modelar los procesos del organismo a nivel fisiológico y geométrico, es por ello que la visualización tridimensional de estructuras anatómicas aporta información muy estimable en medicina. Aunque los especialistas en medicina tienen la habilidad de generar mentalmente imágenes tridimensionales a partir de las imágenes bidimensionales de los cortes axiales, esta visualización mental resulta demasiado tediosa, subjetiva y lenta. La solución ideal es disponer de sistemas computacionales capaces de realizar esta operación y presentar el resultado haciendo uso de las gráficas avanzadas que proporciona hoy la informática.

Desde el punto de vista médico, durante el diagnóstico de pacientes con problemas resulta imprescindible conocer la morfología de tumores, traumas, trombosis, traumas craneo-encefálicos, entre otros, sus dimensiones, su localización y el área afectada. Por ello resulta de gran interés una aplicación informática que permita la segmentación automática de estas estructuras y su visualización tridimensional haciendo uso de gráficas avanzadas.

En este proyecto se propone desarrollar un software que permita realizar una segmentación de una secuencia de imágenes para luego generar una visualización tridimensional. Esta visualización sirve para manipular distintas perspectivas, realizar un corte arbitrario y la manipulación del seudocolor y el nivel de opacidad de la zona a examinar, permitiendo al médico realizar un diagnóstico más exacto.

1.2 DESCRIPCIÓN DEL PROYECTO

La disciplina de la visualización tridimensional surge como una necesidad para aprovechar y asimilar la enorme cantidad de datos producidos por los modernos sistemas de imágenes médicas. El avance experimentado durante las últimas

décadas en la generación, procesamiento y análisis de las imágenes médicas ha tenido un gran impacto en el ejercicio de la medicina. En efecto, la posibilidad de acceder visualmente al interior del cuerpo humano, sin requerir una intervención quirúrgica, y de observar aspectos estructurales y funcionales de los distintos órganos y sistemas es una realidad lograda en países desarrollados. Tal realidad mantiene gran incidencia en el diagnóstico, tratamiento y seguimiento de las enfermedades e importantes beneficios para el paciente y para la sociedad en su conjunto, disminuyendo el costo final del diagnóstico y tratamiento de las enfermedades.

Lamentablemente, en nuestro país no existen las condiciones adecuadas para el desarrollo de este tipo de procesamiento y los programas existentes que manejan nuevas tecnologías de hardware son muy costosos, ya que son programas desarrollados en el exterior cuyas licencias requieren de una gran inversión y los hospitales de nuestro país no cuentan con los recursos para su adquisición.

No obstante, con el uso de la tecnología a nuestro alcance a través de la ingeniería electrónica se busca ayudar a los médicos en la realización de diagnósticos e indicación de terapias más precisas al acceder visualmente al interior del cuerpo humano, sin requerir de una intervención quirúrgica.

Es por lo tanto que este proyecto se propone investigar y desarrollar tecnología matemático-computacional para apoyar el diagnóstico, la escogencia de tratamientos terapéuticos y el seguimiento de la recuperación de los pacientes, guiados por imágenes. Para ello se elaborará un algoritmo que permita generar imágenes tridimensionales a partir de cortes bidimensionales o representaciones producidas por el tomógrafo, visualizar las imágenes teniendo en cuenta los intereses y necesidades de las instituciones hospitalarias, la originalidad de la investigación y su potencial expansión a otras áreas, aumentando así el impacto en el sector salud del país, contribuyendo de esta manera en el avance de la tecnología dentro de nuestro territorio y permitiendo que más personas tengan acceso a este tipo de exámenes al disminuir los costos y mejorar la calidad de vida.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Diseñar un software que permita hacer una visualización tridimensional a partir de imágenes médicas aplicando las técnicas de *Surface Rendering* y *Volume Rendering*.

2.2 OBJETIVOS ESPECÍFICOS

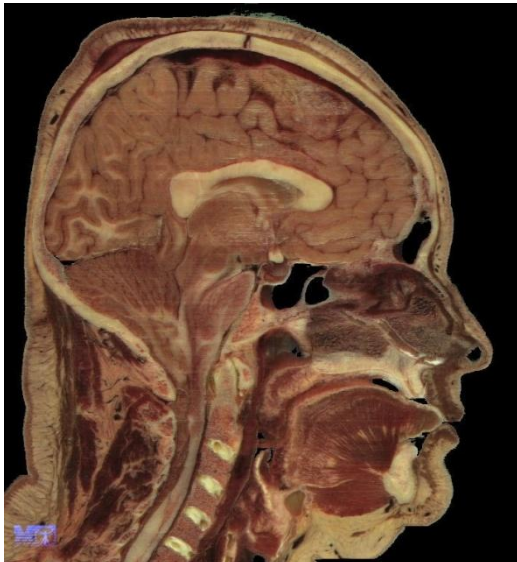
- Desarrollar algoritmos de visualización tridimensional, aplicando las técnicas de *Volume Rendering* y *surface rendering*.
- implementar los algoritmos más comunes de segmentación para una secuencia de imágenes.
- Crear herramientas que permitan la manipulación de diferentes perspectivas del volumen visualizado.
- Crear una herramienta de corte arbitrario que permita seccionar el volumen visualizado y extraer la imagen del área donde fue realizado.
- Crear una herramienta de manipulación del seudocolor y el nivel de opacidad para la técnica de *Volume Rendering*.

3. PROCESAMIENTO DE LAS IMÁGENES

3.1 CORTE DE TEJIDO BIOLÓGICO

Se pueden elaborar cortes finos de muestras de tejidos de organismos vivos en el contexto de la histología. Sin embargo, habitualmente tales muestras de tejidos suelen ser tejidos con patologías, removidos previamente por biopsia con el fin de saber de cual patología se trata, con fines médicos. Pero en algunos casos se realizan secuencias de cortes sobre órganos de humanos o de animales con fines científicos, un ejemplo de esto es el proyecto humano visible (The Visible Human Project); proyecto generado como iniciativa de la Biblioteca Nacional de Medicina (National Library of Medicine) de EE. UU., para crear una completa representación anatómica detallada en 3 dimensiones del cuerpo humano de un hombre y una mujer (ver figura 1).

Figura 1 Imágenes pertenecientes al Proyecto Humano Visible.



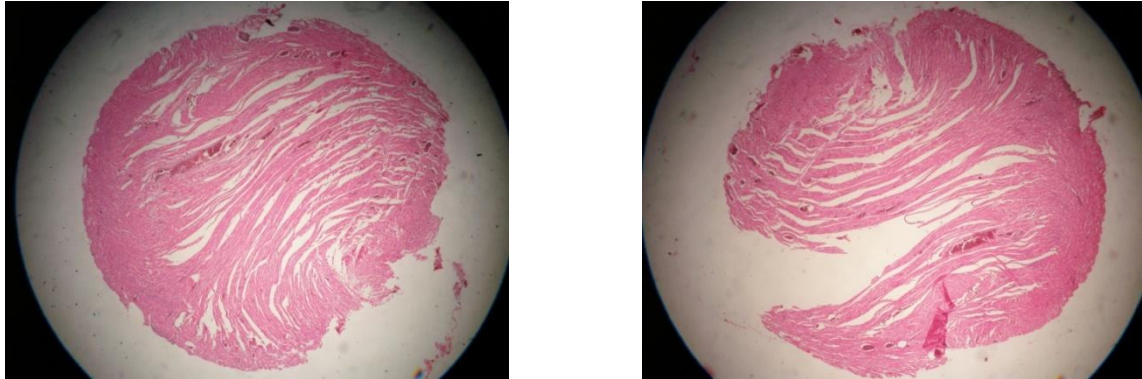
Fuente: <http://www.fundabiomed.fcs.uc.edu.ve/project.html>

3.1.1 Obtención De Imágenes Por Medio De Cortés De Tejido Biológico

Para el presente proyecto se realizaron dos intentos de elaboración de cortes, en los cuales desafortunadamente no se obtuvieron resultados positivos. En el

primero se utilizó el corazón de un ratón infectado con el virus de Chagas (ver figura 2).

Figura 2 Imágenes de corte realizados al corazón de un ratón infectado con el virus de Chagas.



Fuente: suministrada por el autor.

Se trató de elaborar cortes finos de muestra de tejido con el corazón de ratón, utilizando la técnica tradicional en la cual los tejidos son endurecidos, substituyendo el agua por parafina (técnica de infiltración y teñidos) para aumentar la visibilidad de las estructuras celulares. Como se observa en las figura 2 debido al corte se deformaron los tejidos, lo cual nos dio como resultado cortes no aptos para a la visualización tridimensional.

Figura 3 Imágenes de un corazón humano introducido en un bloque de parafina: (A) Resultado del intentar cortar una tajada del bloque. (B) Bloque de parafina con el corazón humano en su interior dentro de una plataforma diseñada para su corte.



Fuente: suministrada por el autor.

Para el segundo caso en el cual se utilizó un corazón humano se aplicó la técnica tradicional, sin resultados favorables porque no se encontró la herramienta de corte indicada debido a las dimensiones del bloque.

3.2PRE-SEGMENTACIÓN

Consiste en que la imagen de entrada siempre debe pasar por una etapa de pre-segmentación. El objetivo fundamental es obtener una buena óptica de la imagen que se quiere segmentar. Las técnicas más importantes y las más utilizadas son especificación del histograma y reducción de ruido.

3.2.1Ecuación del histograma

El principal propósito de la ecualización del histograma es obtener un histograma donde todos los niveles de gris tengan la misma probabilidad, para lo cual el ecualizado redistribuye las distribuciones de intensidad. Si el histograma de una imagen tiene algunos picos y valles, seguirá teniendo picos y valles después del ecualizado.

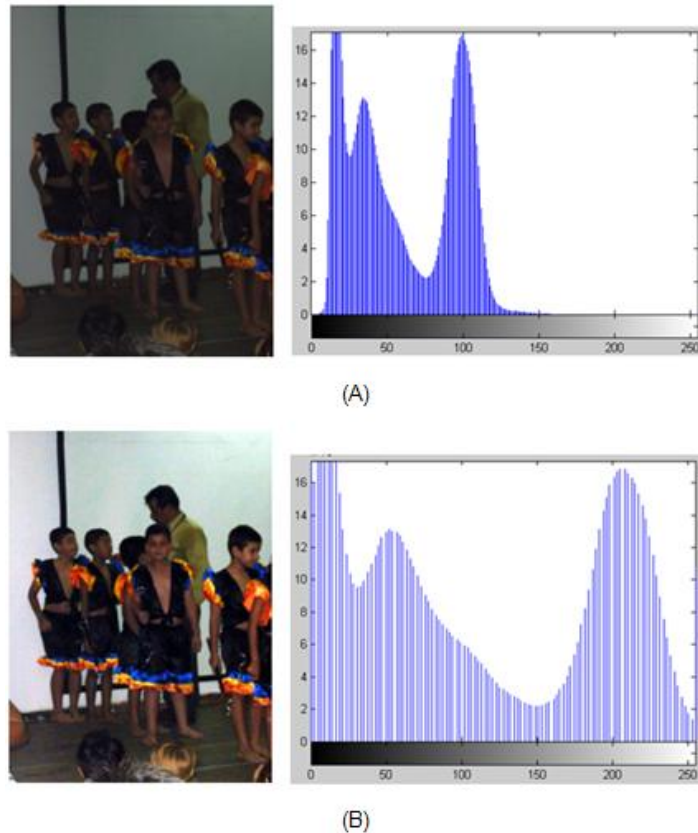
3.2.2Especificación del histograma

La especificación del histograma es un proceso simple que requiere como entrada un histograma deseado y una imagen, este proceso se realiza en dos sencillos pasos:

- Ecualizar el histograma de la imagen original.
- Realizar una ecualización inversa sobre el histograma de la imagen ecualizada.

El resultado de llevar a cabo los dos pasos descritos será una imagen que utiliza más apropiadamente todo el rango disponible de niveles de gris y como consecuencia de ello, tendrá una apariencia más balanceada. En la figura 4-a se muestra una imagen así como su correspondiente histograma. En la figura 4-b se pueden observar los efectos de las funciones explicadas anteriormente, sobre las imágenes y sus histogramas, y observar su representación gráfica; se puede observar que el histograma ha sido expandido para ocupar todo el rango disponible. También se aprecia que la imagen de salida presenta mayor contraste y en consecuencia resulta fácil percibir todas las estructuras que la componen.

Figura 4 Ejemplo de realce de la imagen por medio de la ecualización del histograma: (A) Imagen original. (B) realce del contraste.



Fuente: suministrada por el autor.

3.2.3 Técnicas de reducción del ruido

El objetivo de esta fase consiste en eliminar la mayor cantidad de ruido que sea posible, con el fin de mejorar el proceso de segmentación. No obstante, existen distintos tipos de ruido en una imagen, tales como: ruido Gaussiano, impulsivo, frecuencias y multiplicativo [Escalera, 2001], por lo cual se dificulta la reducción.

Aunque existe una gran cantidad de filtros de eliminación del ruido, en el presente proyecto se analizan tres técnicas: el filtro promedio, Gaussiano y el filtro de la mediana. Cada filtro tiene por objetivo eliminar de mejor forma algún tipo de ruido.

a. FILTRO PROMEDIO

El filtro promedio pertenece a la clase de los filtros lineales. La aplicación de este filtro obtiene el promedio de los píxeles que se encuentren contenidos en la

máscara de este filtro. La salida es el promedio de todos los píxeles vecinos a la posición, incluyendo el píxel central. De este modo, la variación de los niveles de gris disminuye debido al promedio de los píxeles, causando un efecto de suavizado en la imagen. La figura 5 muestra la aplicación del filtro promedio. Se observa que no se ha eliminado el ruido, sólo se ha suavizado, además los bordes de los objetos han perdido su definición y contraste, dificultando el posterior proceso de segmentación.

Figura 5 Aplicación de un filtro promedio: (A) Imagen con ruido Gaussiano. (B) Imagen después de Aplicar el filtro promedio con máscara 5x5.



Fuente: Suministrada por el autor.

b. FILTRO DE LA MEDIANA

Este filtro reemplaza cada píxel de la imagen por el valor de la mediana, calculada a partir de los valores de la máscara que lo rodea, incluyendo el píxel central. El tamaño de la máscara en general es un número impar. La mediana es calculada, ordenando en forma creciente todos los valores de los píxeles incluidos en la máscara y luego se selecciona como la mediana al píxel que es considerado como el valor del medio de la lista.

Ahora bien, en la figura 6-b se presenta la imagen resultante de la aplicación del filtro de la mediana sobre una imagen con ruido Gaussiano (ver figura 6-a); en este caso, no existe una pérdida de los bordes. Las estructuras se han mantenido y gran parte el ruido se ha reducido, generando un conjunto de niveles de gris más uniforme.

Figura 6 Aplicación del filtro de la mediana: (A) Imagen con ruido Sal y Pimienta. (B) Imagen después de Aplicar el filtro de la mediana.



Fuente: Suministrada por el autor.

c. EL FILTRO WIENER

El filtro de Wiener es un filtro lineal de mínimos cuadrados que se utiliza en casos donde el ruido ha llegado a ser significativo. El objetivo del filtro de Wiener es determinar la respuesta al impulso del sistema; de forma que el error sea lo más pequeño posible a través de la minimización del error cuadrático medio. En la figura 7 se presenta la aplicación del filtro Wiener.

Figura 7 Aplicación del filtro Wiener: (A) Imagen con ruido Gaussiano. (B) Imagen después de Aplicar el filtro Wiener.



Fuente: Suministrada por el autor.

3.3 SEGMENTACIÓN DE IMÁGENES

La segmentación de la imagen se define como una partición de una imagen en un conjunto de regiones constituyentes no solapadas, las cuales son homogéneas con respecto a alguna característica como la textura, nivel de gris, color, etc., con el objeto de separar las regiones de interés para su posterior reconocimiento [González, 1992],

El objetivo de la segmentación es simplificar y/o cambiar la representación de una imagen en otra más significativa y más fácil de analizar. La segmentación se usa tanto para localizar objetos como para encontrar los límites de estos dentro de una imagen, más precisamente, la segmentación de la imagen es el proceso de asignación de una etiqueta a cada pixel de la imagen, de forma que los pixeles que compartan la misma etiqueta también tendrán ciertas características visuales similares.

3.3.1 Segmentación basada en el uso de un umbral

La umbralización es una de las técnicas más efectivas, sencillas y tal vez por ello una de las más utilizadas en la segmentación de imágenes en general [Sahoo87], [Gonza93]. La umbralización consiste en la determinación de un umbral que delimita el rango de intensidades que caracterizan y discriminan los distintos objetos de la imagen.

La incógnita central en el proceso de umbralización es saber cómo elegimos el umbral. “No hay un procedimiento general para saber el umbral adecuado; existen muchas formas de averiguarlo, pero ninguna ofrece un resultado igual y en ocasiones se desvía bastante del estimado como óptimo”[Young, 1998]. La umbralización manual suele ser una herramienta fundamental en la segmentación de imágenes, si bien es preferible usar técnicas automáticas que reduzcan la variabilidad introducida por el usuario. Existen técnicas de detección automática del umbral, las cuales veremos a continuación.

a. MÉTODO DE OTSU:

El objetivo de este método es encontrar el valor del umbral, para el cual la varianza entre las regiones a segmentar sea máxima. El método de Otsu calcula el valor umbral de forma que la varianza de cada región sea lo más pequeña posible, pero al mismo tiempo la varianza sea lo más alta posible entre las regiones.

b. MÉTODO DE ENTROPÍA:

En este método el umbral se obtiene aplicando la teoría de la información. Dicho método utiliza la entropía del histograma de los niveles de gris de la imagen como una medida de homogeneidad entre las clases, donde es fundamental determinar el umbral maximizando el nivel superior de la entropía.

c. MÉTODO DE ISODATA:

Es un proceso en el que la conjetura inicial sobre un umbral es refinada a través de consecutivos pasos por la imagen donde la imagen se divide en dos regiones usando un valor de umbral inicial, que es igual al nivel medio ponderado de gris. Para hallar un nuevo umbral, se calcula promediando los niveles medios ponderados de gris de cada una de las regiones obtenidas, este proceso se repite hasta que el valor umbral no cambie o su cambio no sea significativo.

d. MÉTODO DE LA CONSERVACIÓN DEL MOMENTO:

En este método el valor del umbral se calcula determinísticamente de manera que los momentos de la imagen a ser segmentada se conservan en la imagen de salida. Específicamente, antes de segmentar, se calcula los momentos de los niveles de gris de la imagen de entrada. Los umbrales son entonces seleccionados de manera que los momentos de la imagen de salida se mantengan sin cambios [Tsai, 1985].

e. MÉTODO DE ANÁLISIS DE CONCAVIDAD DEL HISTOGRAMA:

Se puede definir un buen umbral analizando la concavidad del mismo [Duda, 1972]. Al analizarlo se encuentra que entre dos valles y su elevación correspondiente en el histograma se puede determinar el umbral característico.

Supongamos un histograma definido por un vector representativo de los niveles de gris con mayor ocupación en una imagen, todos ellos distintos de cero; la concavidad del segmento de histograma que denota dicho vector, determina el umbral óptimo.

f. MÉTODO DE SELECCIÓN ITERATIVA:

Esta técnica iterativa para elegir un umbral fue desarrollada por Ridler y Calvard [Ridler, 1978]. En esta técnica el histograma se divide en dos porciones de segmentos usando un valor de umbral de comienzo, tal como la mitad del máximo del rango dinámico. Luego, se computa la media de los valores asociados con cada una de las partes en que ha quedado segmentado el histograma. Ahora el nuevo valor umbral se calcula mediante el promedio de las medias. Posteriormente el proceso continúa hasta que el valor umbral calculado no cambia o su cambio no se considera significativo.

Figura 8 Resultados de la umbralización con los distintos métodos de umbral.



Fuente: Suministrada por el autor.

Una vez expuestos los métodos utilizados en la umbralización, hay que decir que los resultados obtenidos por dichos métodos son muy dispares tal y como se puede comprobar en la Tabla presentada a continuación.

La siguiente tabla expresa todos los valores de umbral aplicados sobre la imagen original.

UMBRAL	VALOR DE UMBRAL
MÉTODO DE OTSU	120
MÉTODO DE ENTROPIA	93
MÉTODO DE ISODATA	156
MÉTODO DE LA CONSERVACIÓN DEL MOMENTO	19
MÉTODO DE ANÁLISIS DE CONCAVIDAD DEL HISTOGRAMA	239
MÉTODO DE SELECCIÓN ITERATIVA	171

3.3.2 Segmentación por detección de bordes

La detección de bordes es el procedimiento empleado mas habitualmente para la detección de discontinuidades. Un borde se define como la frontera entre dos regiones con nivel de gris relativamente diferente. Vamos a suponer a partir de ahora, que las regiones de interés son suficientemente homogéneas de modo que la transición entre dichas regiones se puede determinar empleando el cálculo de un operador local de derivación.

a. DETECTORES BASADOS EN GRADIENTE

A continuación se enuncian los detectores de bordes basados en gradiente más utilizados.

1. ROBERTS

El operador gradiente de Roberts es el método no lineal más simple utilizado para la detección de bordes. Presenta la desventaja que, dependiendo de la dirección, ciertos bordes son más realzados que otros, inclusive teniendo igual magnitud. También es muy sensible al ruido, ya que usa muy pocos pixeles para aproximar el gradiente.

En la figura 9-bse muestra el resultado del filtro de Roberts, se observa que los bordes encontrados han sido en las zonas en donde la variación de tonalidad ha

sido mayor; al contrario, en las zonas donde existe una diferencia menor en la tonalidad, el filtro no ha generado dichos bordes.

Figura 9 Aplicación del gradiente de Robert: (A) Imagen original. (B) Imagen después de Aplicar el gradiente de Robert.



Fuente: Suministrada por el autor.

2. PREWITT

Este filtro encuentra una mayor cantidad de bordes, en comparación con el filtro de Roberts. No obstante, se observa en la figura 10-b que no se han generado todos los bordes de la imagen, además es deficiente en generar bordes cerrados.

Figura 10 Aplicación del gradiente de Prewitt: (A) Imagen original. (B) Imagen después de Aplicar el gradiente de Prewitt.



Fuente: Suministrada por el autor.

3. SOBEL

Este Modelo es el resultado de aplicar la teoría del cuadrado del binomio. En este caso, se pondera la fila y la columna en la cual se calcula el gradiente. El operador de Sobel tiene la ventaja de proporcionar un suavizado además del efecto de derivación. Ya que la derivación acentúa el ruido, el efecto de suavizado es particularmente interesante, puesto que elimina parte del ruido.

A continuación, en la figura 11-bse muestra el resultado del filtro de Sobel. Se distingue que no existe una mayor diferencia con respecto al resultado generado por el gradiente de Prewitt. Por este motivo, el uso del cuadrado del binomio no ha producido una mejor definición de los bordes, además no se han generado bordes cerrados.

Figura 11 Aplicación del gradiente de Sobel: (A) Imagen original. (B) Imagen después de Aplicar el gradiente de Sobel.

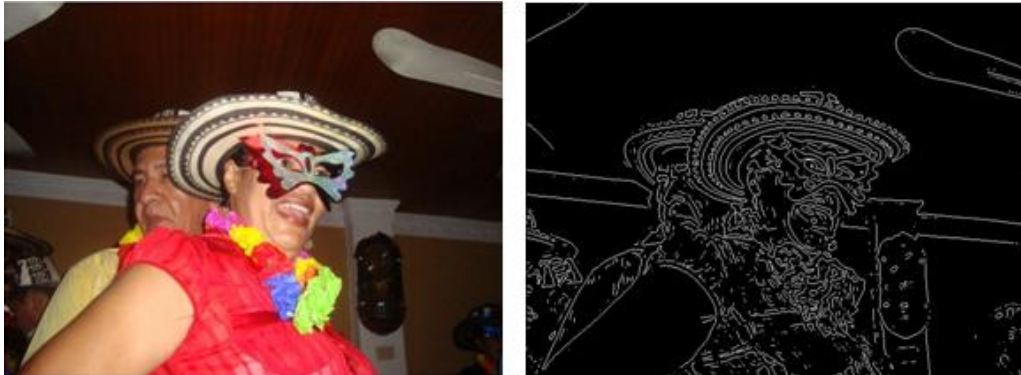


Fuente: Suministrada por el autor.

b. DETECTORES BASADOS EN LAPLACIANO

Otra posibilidad para la acentuación de bordes consiste en usar la segunda derivada o Laplaciano. Por su parte aunque el Laplaciano responde a transiciones en la intensidad de la imagen, se emplea en pocas ocasiones en la práctica. Su poco empleo es debido a que es un operador de segunda derivada sensible en exceso a la presencia de ruido y además, el Laplaciano da lugar a bordes dobles y no permite determinar direcciones (ver figura 12-b).

Figura 122 Aplicación del Laplaciano: (A) Image original. (B) Imagen después de Aplicar el Laplaciano.



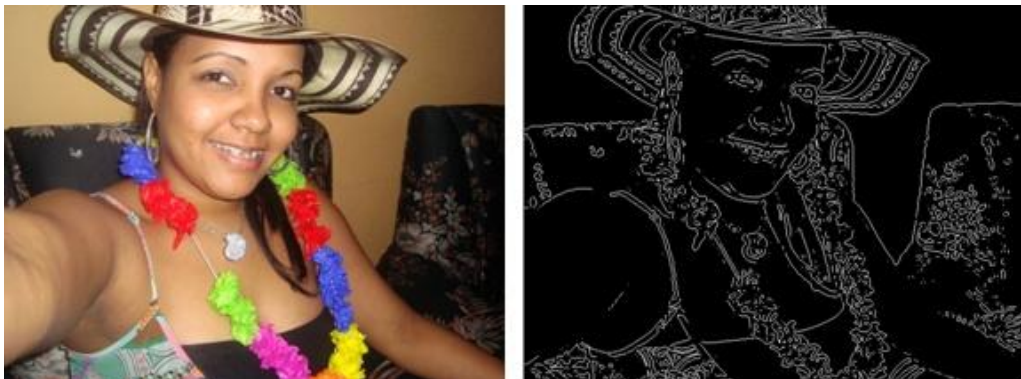
Fuente: Suministrada por el autor.

c. LAPLACIANA DE LA GAUSSIANA

El Laplaciano del Gaussiano combina el filtrado Gaussiano con el Laplaciano para la detección de bordes. Al hacer el filtrado se eliminan las pequeñas estructuras debidas al ruido, pero los bordes son ensanchados, por lo que son considerados como bordes aquellos que producen un máximo local; lo anterior es logrado al detectar el cruce por cero en la segunda derivada.

A diferencia de los filtros anteriores, el filtro Laplaciano de la Gaussiana produce una mejor definición de bordes, generando una mayor cantidad y capturando incluso zonas con menor tamaño (ver figura 13-b), sin embargo, aún persiste el problema de las regiones abiertas.

Figura 113 Aplicación del gradiente del Laplaciano: (A) Image original. (B) Imagen después de Aplicar el gradiente del Laplaciano.



Fuente: Suministrada por el autor.

d. DETECTOR DE BORDES DE CANNY

El detector de Borde Canny es el algoritmo más utilizado [Canny, 1986], ya que es el detector de bordes más poderoso que existe actualmente. Es la primera derivada de la Gaussiana y se aproxima al operador que optimiza el producto de señal a ruido y localización (ver figura 14-b).

Figura 14 Aplicación del filtro Canny: (A) Image original. (B) Aplicación del filtro Canny.



Fuente: Suministrada por el autor.

3.3.3 Algoritmos de clústering

Entendemos por Algoritmo de Clústering aquel que dado una serie de datos de entrada con una serie de características agrupa dichos datos en una serie de conjuntos denominados clústeres. Para realizar esta agrupación, los algoritmos de clústering deben determinar el criterio de homogeneidad que suele basarse en el uso de distancias (generalmente la Euclidiana), mediante el cual se guiarán a la hora de determinar a qué conjunto (clúster) pertenece cada dato.

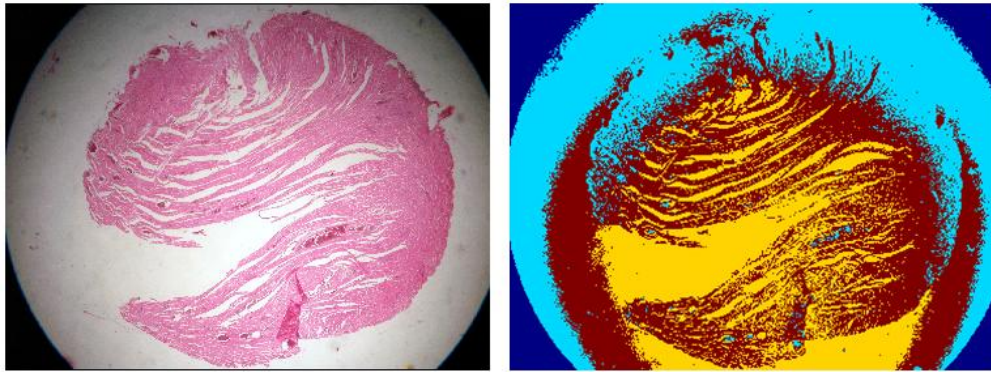
a. K-MEANS

El algoritmo K-medias [MacQueen, 1967] es una de las técnicas de agrupamiento más populares debido probablemente a su sencillez. También es llamado el algoritmo de las medias móviles porque en cada iteración se recalculan los centros de las clases. Este algoritmo requiere de un único parámetro “K” definido como el número de agrupamientos que debe encontrar.

Luego de definir el parámetro k, se clasifican todos los datos en la clase del centroides demás cercano de acuerdo a la distancia Euclidiana, después se

recalculan los centroides. Si esos centroides no difieren de los centros anteriores, entonces el algoritmo termina; caso contrario, se repite el proceso de asociación con los nuevos centroides hasta que no haya variación en los centroides.

Figura 15 segmentación por el algoritmo de Kmeans: (A) Image original. (B) Imagen segmentada por el método de Kmeans.



Fuente: Suministrada por el autor.

Para los objetivos buscados en este proyecto se implementaron dos métodos para la inicialización del algoritmo K-Means:

1. METODO ALEATORIO:

Para este método se escogen píxeles aleatorios de la imagen teniendo en cuenta que los valores de intensidad de estos píxeles no se repitan y que no sean iguales a cero (ya se definió a cero como el fondo de la imagen).

2. METODO MANUAL:

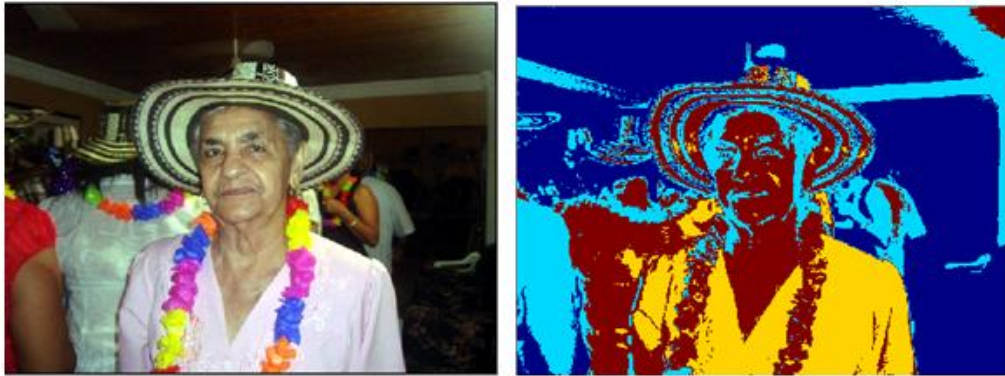
En este método se despliega la imagen y el usuario puede escoger el píxel deseado por medio del mouse. En este método se utilizan las mismas restricciones de intensidad del píxel que en el método anterior.

b. FUZZY C-MEDIAS

El algoritmo Fuzzy C-medias o k-medias borroso [Dunn, 1973] es una variante del k-medias tradicional que introduce los principios de la lógica difusa. La principal diferencia es que en el k-medias cada dato solo puede pertenecer a una única clase mientras que en el Fuzzy C-medias puede pertenecer a varias clases a la vez, con una probabilidad determinada por las distancias a cada uno de los

centroides. Es un método más robusto que el k-medias pero mucho más costoso computacionalmente.

Figura 16 Segmentación por algoritmo de Fcm: (A) Image original. (B) Imagen segmentada por el método de Fcm.

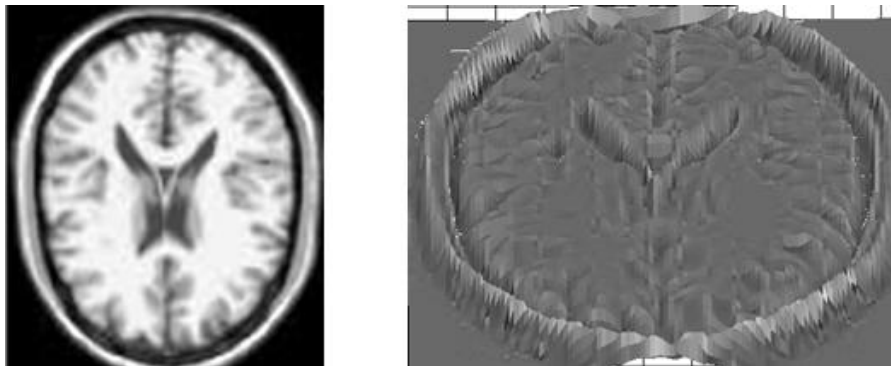


Fuente: Suministrada por el autor.

3.3.4 Segmentación por Watershed

La transformada conocida como Watershed es una técnica bien conocida en área de la topografía y es fácilmente adaptable a los diferentes tipos de imágenes, es capaz de distinguir objetos sumamente complejos que no pueden ser analizados correctamente mediante algoritmos convencionales [Beucher, 1993], [Couprie, 1997].

Figura 17 Ejemplo de la forma como es interpretada una imagen por medio de la transformada Watershed: (A) Imagen de medica. (B) una representación en forma de superficie topografica.

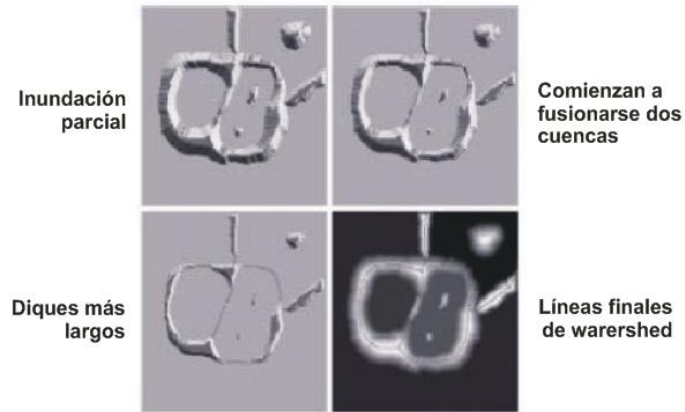


Fuente: Suministrada por el autor.

Esta técnica consiste en interpretar la imagen como una superficie topográfica donde el nivel de gris de un punto indica la altitud en dicho punto (ver figura 17). En

este sentido, podemos decir que las intensidades de gris de mayor amplitud se corresponden con llanuras o montañas mientras que las intensidades de menor valor se corresponden con valles y ríos. Por tanto, se intenta dividir la imagen de grises en regiones. Generalmente una de ellas se corresponde con el fondo de la imagen y el resto con los objetos o regiones que se pretende extraer. El objetivo último de esta técnica es determinar los contornos que definen dichos objetos.

Figura18Etapas en el algoritmo de Watershed.



Fuente:<http://bibing.us.es/proyectos/abreproy/11619/fichero/PFC%252FAPENDICES%252FApendice+B.pdf>

Aludiendo a lo anterior, imagínese que se hace un pequeño agujero en cada mínimo local del relieve topográfico y que hundimos este relieve en un recipiente con agua con una velocidad vertical constante [Gonzales, 2000]. El agua va subiendo e inundando las cuencas. Cuando el agua de dos cuencas está a punto de juntarse se construye un estancamiento para evitar la fusión de las dos cuencas. La inundación continua y llega a un punto en que solo se ve la parte de arriba de las presas por encima de la línea de agua. Las líneas de Watershed forman un camino conexo, dando por lo tanto bordes continuos entre las regiones.

a. SOBRESegmentación

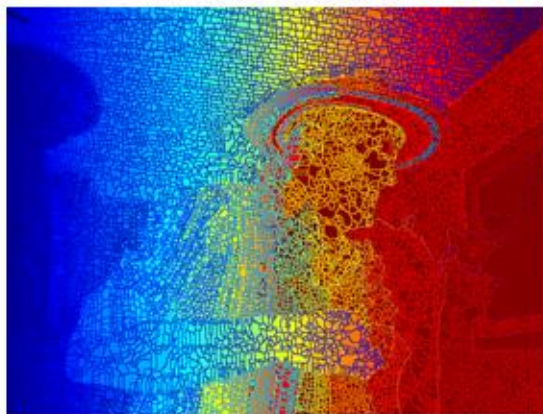
El resultado en imágenes reales suele conducir al fenómeno de sobresegmentación, porque la presencia de mínimos locales en las imágenes son bastantes altas. Para evitar este inconveniente se suele eliminar todos los mínimos parasitarios y reemplazarlos por marcadores. El nuevo algoritmo resultante se denomina Watershed controlado por marcadores. Es por lo anterior

que la elección de unos buenos marcadores será clave para obtener una exitosa segmentación.

En este caso los marcadores son seleccionados empleando la capacidad de la reconstrucción geodésica o morfológica [Sternberg, 1986] que se realiza mediante una serie de dilataciones o erosiones condicionadas, teniendo en este último caso la llamada reconstrucción dual, reduciendo así el número de mínimos regionales y evitando la sobresegmentación.

Es por lo tanto que, el Watershed controlado por marcadores reduce el efecto de sobre identificar zonas o regiones en la imagen (Ejemplo de lo anterior, ocurre en la figura 19). La transformación de Watershed de la imagen expuesta provoca una sobresegmentación. En la figura 19 se tiene el resultado del Watershed controlado por marcadores, por lo cual, la mejora del proceso de segmentación es evidente.

Figura 19 Ejemplo de la segmentación por el método de Watershed con sobresegmentación: (A) Imagen original. (B) Imagen segmentada por el método de Watershed con sobresegmentación. (C) Imagen segmentada por el método de Watershed sin sobresegmentación.



Fuente: Suministrada por el autor.

3.4 POST-SEGMENTACIÓN

Cuando aplicamos métodos de segmentación como umbralización o alguno de los detectores de bordes, se pueden provocar frecuentemente errores. En aquellos errores, aparecen falsas uniones de caracteres, o bien caracteres separados en trozos no conexos donde no debieran. Para evitar estos defectos se pueden realizar, posteriormente, algunas técnicas para cerrar contornos o adelgazarlos que tienden a mejorar la binarización. Este problema es solucionado empleando el tratamiento morfológico, el cual es una técnica de procesamiento no lineal, caracterizado en realzar la geometría y forma de los objetos. Su fundamento matemático se basa en la teoría de conjunto.

4. VISUALIZACIÓN TRIDIMENSIONAL A PARTIR DE IMÁGENES

Visualizar un objeto tridimensional, se refiere a obtener su representación en la memoria de un computador, manteniendo sus características y forma. Primero, se deben crear las imágenes de los cortes transversales. Luego, cada imagen de la secuencia es filtrada para eliminar ruido y segmentada de acuerdo con la estructura anatómica que se quiere reconstruir. Después, son apilados visualmente los cortes y se puede visualizar la apariencia del objeto tridimensional.

Seguidamente, se realiza un redimensionamiento de la secuencia de datos en la cual se elimina secciones del cubo donde no exista información, obteniendo un cubo de datos de menor tamaño que el original y finalmente se utiliza alguno de los métodos de visualización tridimensional. Además se pueden realizar manipulaciones de la visualización tridimensional tales como realizar un corte sobre esta, generando dos mitades, modificaciones de los niveles de color y transparencia.

4.1 REDIMENSIONAMIENTO DE LA SECUENCIA DE DATOS

Por lo general, cuando se apilan las secuencias de datos en un cubo, hay partes de este cubo que no aportan ninguna información del volumen, lo que termina generando un costo computacional innecesario a la hora de realizar la visualización tridimensional. Para solucionar este problema se debe determinar la dimensión real del cubo que contiene información correspondiente al volumen, lo cual se logra realizando un redimensionamiento del cubo.

4.2 MÉTODOS DE *SURFACE RENDERING*

Estadística se basa en los contornos del objeto para obtener una malla de polígonos. Esta malla poligonal se ajusta entre los isocontornos de dos imágenes consecutivas obteniendo una isosuperficie del volumen.

Los métodos de *surface rendering* más utilizados son la descrita en el algoritmo del escultor y la descrita en el algoritmo de *Marching Cubes*. Valga decir de una vez que este último método enunciado fue el finalmente usado para la realización de

este proyecto de grado. Sin embargo cabe explicar que el primero consiste en generar isosuperficie partiendo de una nube de puntos y el segundo genera de manera eficaz una isosuperficie basada en polígonos triangulares teniendo en cuenta la pertenencia de los voxels al contorno del objeto.

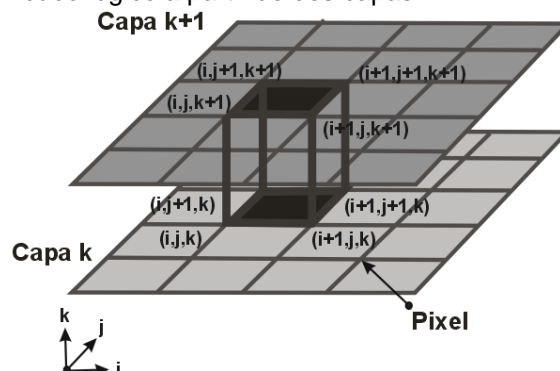
4.2.1 *Marching cubes*

Un caso especial de visualización Tridimensional lo constituye el método llamado *Marching Cubes*[Cline, 1987]. El cual es un método simple y bastante popular para obtener una malla triangular de la superficie (isosuperficie) externa del objeto contenido en los datos volumétricos. El algoritmo desarrollado está basado en la descripción realizada por William E. Lorensen y Harvey E. Cline [Cline, 1987].

a. DESCRIPCIÓN DEL ALGORITMO

El algoritmo consiste básicamente en determinar que partes del volumen formarán la isosuperficie, para ello se crea un cubo lógico a partir de dos capas contiguas que contenga en sus 8 vértices (cada vértice corresponde a un voxel) la información de la intensidad que corresponde a ese punto. Posteriormente, se comparan los ocho vértices del cubo lógico formado con el valor umbral introducido.

Figura 20 Formación de un cubo lógico a partir de dos capas.

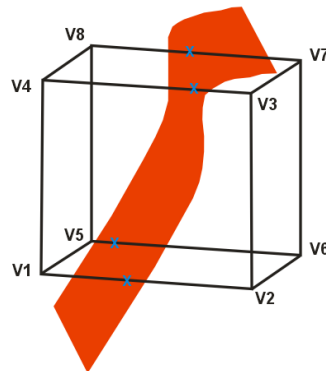


Fuente: <http://eciencia.urjc.es/dspace/bitstream/10115/765/1/PFC%20CASTA%C3%91O%20CABR%20ALES.pdf>.

En consecuencia, el cubo es cortado por la isosuperficie si al menos hay un vértice que está por debajo y otro que está por encima del umbral. Si cumple lo anterior indica que el cubo formará parte de la isosuperficie final. Cuando todos los vértices del cubo están por debajo o por encima del umbral no se procesan ya que no

formarán parte de la isosuperficie final. La figura 20 muestra como se forma el cubo lógico a partir de dos capas adyacentes del volumen, el cual se utiliza para marchar a través de todo el volumen de datos.

Figura 21 Cálculo del índice que define la colocación del cubo con respecto al volumen: Los vértices 7, 3, 2 y 6 se encuentran en el interior de la superficie ($V \geq \text{umbral}$) y se marcan como 1, mientras que los vértices 1, 4, 5 y 8 se encuentran en el exterior de esta ($V < \text{umbral}$) y se marcan como 0.



Fuente: <http://www.dccia.ua.es/~jmsaez/Publicaciones/OtrosTrabajos/PFC%20-%20Sistemas%20de%20Representaci%F3n%20de%20Blobs%20-%20JMSaez.pdf>.

De lo anterior se crea un índice de 8 bits que nos defina la colocación de cada cubo con respecto al volumen (ver figura 22). Dicho índice se formará con la información de la pertenencia de los distintos vértices del cubo al volumen. Los vértices que estén dentro de la superficie (que sean igual o mayor al valor de umbral) los marcaremos con un 1; los vértices que estén fuera de la superficie (por debajo del valor de umbral), con un 0. En la figura 22 se puede observar cómo se realiza este procedimiento.

Figura 22 Índice de colocación del cubo.

V1	V2	V3	V4	V5	V6	V7	V8
0	1	1	0	0	1	1	0

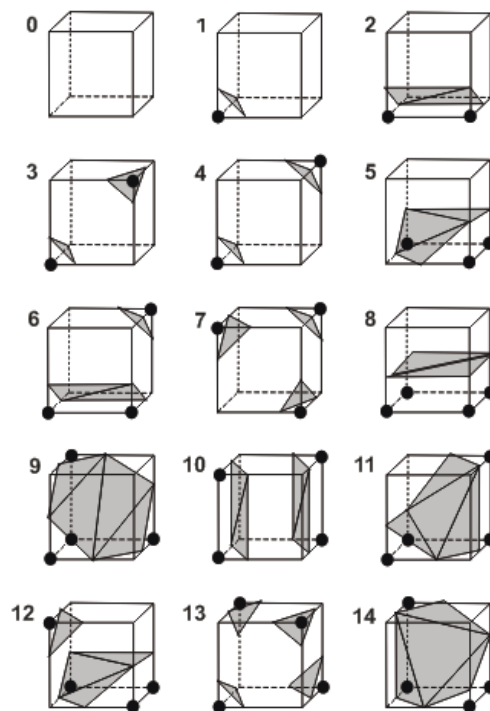
Fuente: <http://www.dccia.ua.es/~jmsaez/Publicaciones/OtrosTrabajos/PFC%20-%20Sistemas%20de%20Representaci%F3n%20de%20Blobs%20-%20JMSaez.pdf>.

Con la información sobre la colocación del cubo con respecto al volumen, procederemos a estudiar qué forma tendrá la porción de superficie asociada a dicho cubo. Para ello, Lorensen y Cline [Cline, 1987] estudiaron de cuántas maneras puede un cubo interceptar con una superficie y llegaron a la conclusión de que hay 256 formas. Esto se debe a que tenemos ocho vértices en cada cubo y dos posibles estados ($2^8=256$). Si quisiéramos generar los 256 casos es posible,

pero sería tedioso. Sin embargo, aplicando dos operaciones de simetría sobre el cubo (Complemento y Rotación), reducimos el problema de 256 casos a tan solo 15.

En la Figura 23 se pueden ver los 15 casos de intersección del cubo con la superficie, así como los triángulos para cada caso. El caso más simple (primer caso) ocurre si todos los vértices están por encima (o por debajo) del valor seleccionado y no produce ningún triángulo. El siguiente caso (segundo caso) ocurre si la superficie separa un vértice de los otros siete, produciendo un triángulo, definido por la intersección con las tres aristas que se juntan en ese vértice. Los otros casos producen múltiples triángulos y se detallan en [Cline, 1987].

Figura 23 Casos de intersección de un cubo con la superficie: Los vértices con un punto simbolizan que dicho vértice se encuentra en el interior de la superficie.



Fuente: <http://users.polytech.unice.fr/~lingrand/MarchingCubes/algo.html>.

Una vez se selecciona el caso al que pertenece cada cubo, podemos formar los triángulos, estudiando las intersecciones de sus aristas con la superficie. Con la información de posición del cubo con respecto al volumen, calcularemos la posición exacta de la intersección de la superficie con las aristas, mediante interpolación. Se usa interpolación lineal, por ser la más sencilla. Lorensen y Cline

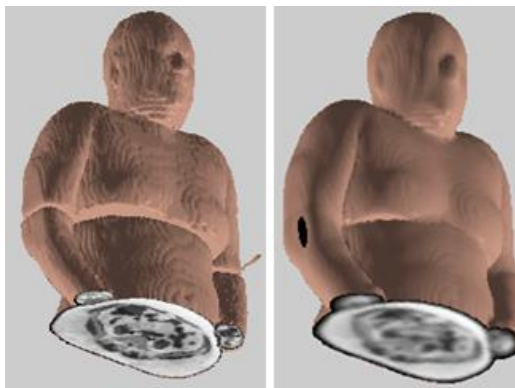
[Cline, 1987] han experimentado con interpolaciones de orden mayor pero obtuvieron pocas mejoras.

Ahora bien, al determinar los lugares del cubo por donde pasa la superficie se pueden generar triángulos que unan los puntos de intersección. A continuación se mueve (marcha) al siguiente cubo y así sucesivamente. Una vez generados todos los triángulos de los cubos pertenecientes a la frontera, podemos asegurar que el volumen está generado completamente, puesto que los triángulos asociadas a cada volumen parcial han sido calculados y la unión de los mismos nos dará la isosuperficie (superficie buscada).

b. INCONVENIENTES DEL ALGORITMO

Marching cube suele producir triangulaciones que tienen gran cantidad de triángulos muy pequeños o con ángulos muy agudos. En la Visualización, los triángulos con ángulos muy agudos y poca superficie pueden aparecer como bordes agudos en la superficie y triángulos cuya superficie es muy pequeña aparecen como puntas agudas. Además, se obtienen superficies muy cuadradas y escalonadas que no representan la curvatura natural de la superficie, dando como resultado imágenes de baja calidad. El modelo también contiene demasiados nodos y superficies triangulares, que dificultan fuertemente la eficacia computacional si se utilizara directamente para la generación de la malla volumétrica (ver figura 24-a).

Figura 24 Filtro de suavizado: (A) Visualización tridimensional sin filtro de suavizado. (B) Visualización tridimensional con filtro de suavizado.



Fuente: Suministrada por el autor.

Sin embargo, una de las alternativas para mejorar la calidad de la superficie es aplicar un filtro de suavizado. Este filtro permite relajar la malla y que

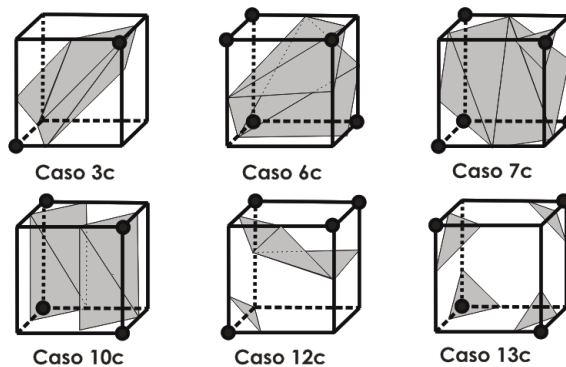
desaparezcan las irregularidades. No obstante, es muy importante que la malla conserve la topología, ya que si esta se deforma se perderán detalles. Teniendo en cuenta el origen de esta malla (una imagen médica), estos detalles representan partes de la anatomía y no pueden perderse.

Cabe aclarar, que para el presente trabajo de grado el filtro de suavizado que se decidió utilizar fue el filtro Gaussiano, el cual se aplica sobre la secuencia de imágenes antes de utilizar el método de *Marching cubes*. Al aplicar este filtro se “relajan” los bordes de las imágenes. Si se utiliza *Marching cubes* una vez aplicado el filtro sobre la imagen los resultados mejoran considerablemente, la superficie es más suave, se mejora la apariencia de la malla y la forma de los triángulos de la superficie, logrando un mejor ajuste a la región real conservando la topología del modelo (ver figura24-b).

Ahora bien, otro problema muy común se debe a que el algoritmo se aplica a cada cubo de forma independiente y esto produce problemas de ambigüedad. Si se observan cuidadosamente los casos 3, 6, 7, 10, 12 y 13, se puede observar que la superficie puede pasar por distintas aristas (caras ambiguas). El resultado de estas ambigüedades es que se pueden producir agujeros en la superficie.

El error mencionado en la parte inmediatamente anterior, ha sido solucionado por varios autores extendiendo los 15 casos vistos con otros complementarios, para los casos que pueden tener ambigüedad. Estos casos se diseñan para que sean compatibles con los casos vecinos y eviten agujeros en la superficie. Los seis casos complementarios que se necesitan, para los casos 3, 6, 7, 10, 12 y 13 se muestran en la figura25.

Figura2525Casos complementarios del algoritmo de *Marching Cubes*.



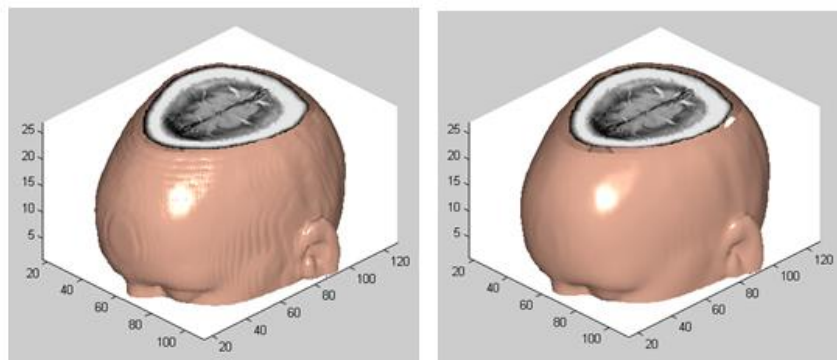
Fuente: <http://users.polytech.unice.fr/~lingrand/MarchingCubes/algo.html>.

4.2.2 Post-procesamiento de superficies

El objetivo del post-procesamiento será resolver los problemas de precisión de *Marching Cubes*. La idea es obtener una malla tridimensional más precisa a partir de la que nos proporciona el algoritmo. Para ellos se utiliza el crecimiento por gradiente en los vértices de cada triángulo de la isosuperficie. Este procedimiento asegurará la normal entre triángulos contiguos, aplicando criterios basados en información local. Por tanto, resolveremos problemas relacionados con la pérdida de exactitud en zonas que requieran alto nivel de detalle.

Así mismo, los puntos son desplazados de manera inversamente proporcional a la magnitud de su gradiente. El desplazamiento es en sentido de la normal. Algunos puntos no son desplazados ya que la magnitud del gradiente es muy grande.

Figura 26 Etapa de Post-Procesamiento de Superficies: (A) Visualización tridimensional antes de la etapa de Post-Procesamiento de Superficies. (B) Visualización tridimensional después de la etapa de Post-Procesamiento de Superficies.



Fuente: suministrada por el autor.

4.3 MÉTODOS DE VOLUME RENDERING

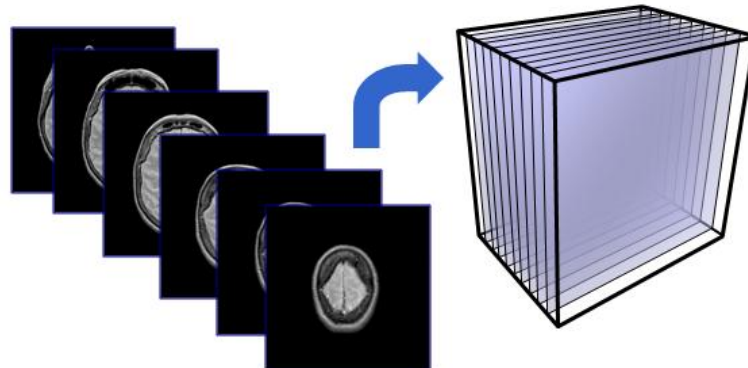
Volume Rendering se define como el proceso mediante el cual datos escalares provenientes de un espacio tridimensional son procesados para mostrar sobre una proyección bidimensional una posible representación gráfica de éstos de acuerdo a Levoy [Levoy, 88], Lacroute [Lacroute, 1994] y McReynolds & Blythe [McReynolds, 2005]. Mediante *Volume Rendering* es posible hacer esta visualización sin la necesidad de construir modelos geométricos intermedios. Las bases que constituyen el proceso de visualización son los modelos físicos de propagación de la luz sobre materiales que tienen índices de color y opacidad variables.

Los métodos de *VolumeRendering* más utilizados son Ray Casting y el basado en texturas. El primero consiste en lanzar rayos (voxel) y el segundo es muy parecido al anterior solo que en vez de lanzar rayos se lanzan planos. Hay que aclarar que este trabajo se centra en el segundo método. En este tipo de método es muy común utilizar tarjetas graficas debido a la cantidad de información que se despliega. Sin embargo, debido al costo de algunas de estas tarjetas se decidió hacer una implementación sin hacer uso de ella.

4.3.1 *Volume rendering* basado en texturas

Esta técnica fue implementada a partir del trabajo de Westermann y Sevenich en [Westermann 2001] y consiste en separar el volumen de datos en una serie de texturas 2D, las cuales se montan sobre una serie de polígonos (llamados proxy geometry, ver figura27) y se lanza los planos de forma perpendicular a alguno de los ejes y orientada al objeto como se puede observar en la figura 28. En este caso el objeto es el volumen a visualizar. Al dibujar esta geometría comenzando por la textura más lejana a la cámara (de atrás hacia adelante), cada textura se superpone a la siguiente para componer la imagen del volumen de datos. La superposición de texturas se ilustra en la figura27.

Figura 27 Mapeo de los cortes a los polígonos (planos).

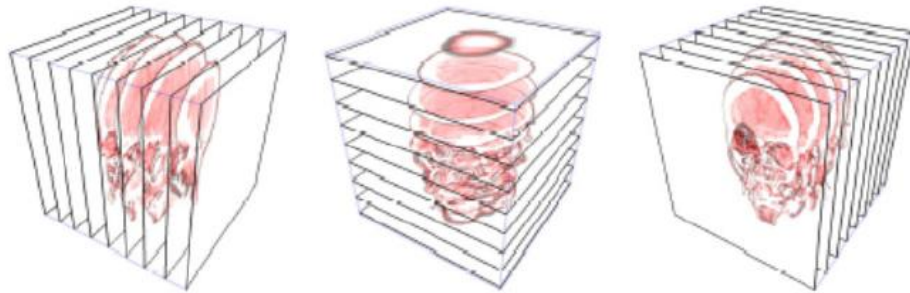


Fuente: <http://lcg.ciens.ucv.ve/~ernesto/ppts/volumeRendering.ppt>

Lo mostrado anteriormente puede ser pensado como un apilamiento de imágenes bidimensionales; se colocan las imágenes una encima de la otra. Los pixeles de cada imagen están alineados con sus correspondientes pixeles de la misma posición arriba y abajo. Así pues, se pueden etiquetar los pixeles con coordenadas tridimensionales. Ellos mantienen sus coordenadas originales y toman una nueva,

la cual es simplemente el número del corte con el que están asociados los píxeles. De esta forma se elimina el concepto de imágenes de cortes. Se representan todos los píxeles de todas las imágenes de cortes del objeto en una imagen tridimensional única.

Figura 28 Esquema de los tres métodos de texturas, donde cada uno es perpendicular a un eje del plano cartesiano.

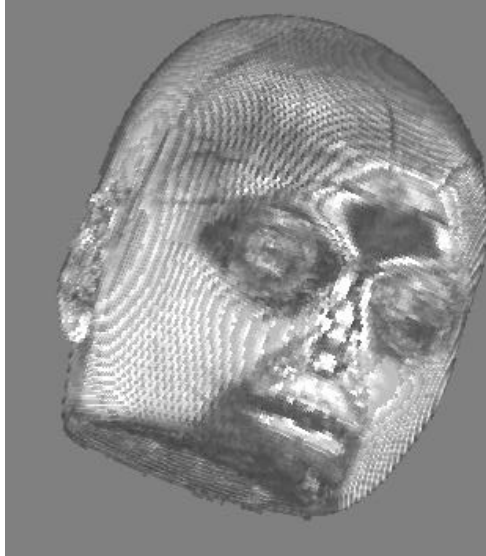


Fuente: http://www.oocities.com/the_break_heart/tesisLopez.pdf

En la figura 28 se puede notar que en la técnica existen 3 maneras de visualizar el volumen por el método de texturas dado el número de ejes de visualización a partir del volumen (cada una perpendicular a cada eje del volumen).

Las imágenes obtenidas para cada una de las tres formas de representación de texturas, presentan bandas alrededor del volumen (*banding*) debidas a los espacios entre cada una de las texturas superpuestas. Lo anterior produce una visualización errónea del volumen con respecto a lo capturado inicialmente debido a que esta técnica considera cada plano del volumen bidimensionalmente. Una solución parcial a este problema es combinar los tres conjuntos de texturas explicados anteriormente. De esta manera se logra obtener una mejor visualización del volumen (ver figura 29).

Figura 29 Visualización del volumen de datos por el método de texturas 2D: En el cual se combinando los 3 tipos de conjunto de texturas. También se puede observar los artefactos denominado *popping* causados por ausencia de interpolación espacial.



Fuente: Suministrada por el autor.

Ahora bien, la representación del volumen obtenidas con el uso de los 3 conjuntos de texturas, presentan bandas alrededor del volumen o artefactos denominado *popping* causados por ausencia de interpolación espacial. El cambio entre copias de los conjuntos de texturas es notable, el ancho de estas bandas depende de la cantidad de planos que interceptan el volumen y la orientación de las mismas depende de la orientación del volumen. Un ejemplo de esto puede ser apreciado en la figura 29. Una solución a este problema es realizar la visualización por medio del método de texturas 3D la cual obtiene una visualización del volumen en la cual no aparecen los artefactos llamados *popping*.

Tenemos además que en el método de texturas 2D se utiliza un polígono (el cual es un cuadrado o un rectángulo) compuesto sólo por cuatro puntos. En cambio para el caso de texturas 3D se utiliza un polígono compuesto por $n*m$ puntos donde n y m son las dimensiones de la imagen, por lo que podríamos considerarlo más bien una malla. El procedimiento del método de texturas 3D es exactamente igual que el de textura 2D, la única diferencia es que al tener un punto en el espacio por cada voxel de la imagen podemos realizar una interpolación. Cabe aclarar, que debido a que la representación por texturas 3D es considerablemente más pesada que la de texturas 2D, decidimos no incluirla en la aplicación final de este proyecto de grado.

Por su parte, la visualización directa de volúmenes es una tarea que supone gran carga computacional, a medida que crecen los volúmenes de datos o los requisitos de calidad en las imágenes. Sin embargo, con una máquina lo suficientemente potente, se puede navegar interactivamente. Igualmente se puede considerar, si se intenta reducir aún más el tiempo necesario para proyectar el volumen de datos sobre el plano de imagen, la posibilidad de paralelizar dicho algoritmo, esto es, de dividir la tarea entre varios procesadores, que trabajaran simultáneamente. Las prestaciones dependerán del número de procesadores utilizados [Crockett, 1996], [Crockett, 1997], [Yagel, 1995], [Neumann, 1994].

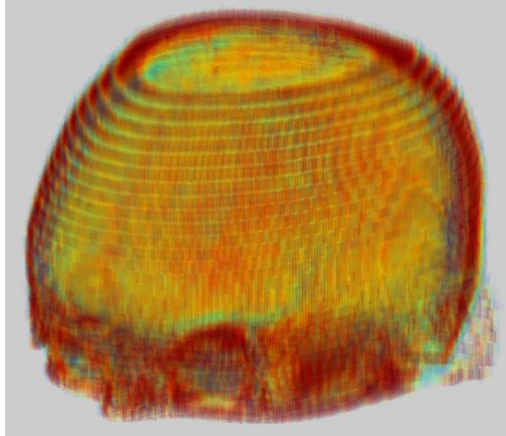
Otra opción al mejorar el rendimiento es recurrir a hardware específico de renderización, el cual puede aplicarse a muchos problemas de visualización. Sin embargo, este enfoque no posee la flexibilidad de la renderización basada en software.

a. SOMBREADO

En la fase de sombreado se pueden aplicar varias técnicas [Udupa, 1991], [Lichtenbelt, 1998] entre las que se incluyen la de sombreado “original”, sombreado en profundidad y sombreado de superficie. La técnica que utiliza en este proyecto es la de sombreado original consistente en asignar a cada voxel un nivel de opacidad y un nivel de color proporcional a la Intensidad registrada en ese punto (función de transferencia).

Creando una característica de opacidad se puede generar una imagen cuyo interior sea visible. Las capas exteriores de un objeto pueden ser traslúcidas, mostrando los datos que existen debajo de ellas. Esto hace posible que sean visibles las estructuras internas de un objeto. Además, los puntos de vista del *volume rendering* se pueden situar dentro del objeto. Esto permite explorar detalles en el interior del objeto tales como tumores y otras anomalías. Además manipulado los valores de transparencia y color se obtiene un alto grado de interactividad con el volumen desplegado.

Figura 30 Visualización del volumen de datos: Donde han sido manipulado los niveles de color y transparencia.



Fuente: Suministrada por el autor.

En la figura 30 se observa una visualización volumétrica por texturas con un nivel de transparencia, haciéndose visible el interior del volumen de datos.

4.4 MANIPULACIÓN DE LA VISUALIZACIÓN

El sistema de manipulación está diseñado para permitir al usuario realizar el proceso de visualización de una manera amigable. Las herramientas desarrolladas para la manipulación de la visualización permiten al usuario realizar un corte sobre el volumen visualizado, generando dos mitades de este volumen. También permite la navegación sobre el volumen, ya que brinda la posibilidad de cambiar el punto de vista o perspectiva de la visualización.

4.4.1 Navegación Sobre La Visualización

La navegación del mundo virtual permite una interacción básica que consiste en que el usuario cambie su punto de vista de la visualización. En este tipo de aplicación, el usuario es un espectador del escenario, pero no completamente pasivo, puede desplazarse por el mismo para contemplarlo desde otro ángulo, o desde otra posición. Este sistema de navegación está compuesto por 3 Movimientos, los cuales son:

- Rotación.

- Zoom.
- Desplazamiento.

La rotación permite girar la visualización tridimensional para obtener diferentes vistas del volumen visualizado. Para lo cual, se implementaron dos rotaciones una alrededor del eje Z y la otra alrededor de un eje intermedio entre el eje X y el eje y. En la figura 31 se puede observar un ejemplo de rotación de la visualización tridimensional.

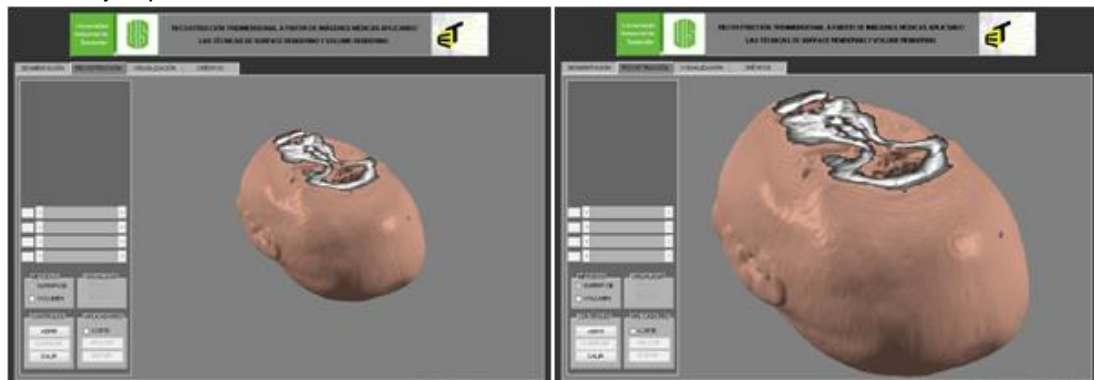
Figura 31 Ejemplo de rotación de la visualización tridimensional.



Fuente: Suministrada por el autor.

El zoom permite acercar y alejar de la visualización tridimensional para tener una mejor observación de un lugar específico de la visualización. En la figura 32 se puede observar un ejemplo de acercamiento de la visualización.

Figura 32 Ejemplo de zoom en la visualización tridimensional.



Fuente: Suministrada por el autor.

El desplazamiento permite mover la visualización tridimensional para desplazarla, esto sirve cuando se ha acercado de forma considerable a la visualización y de

este modo poder navegar sobre ella. En la figura 33 se puede observar un ejemplo de desplazamiento de la visualización tridimensional.

Figura 33 Ejemplo de desplazamiento de la visualización tridimensional.

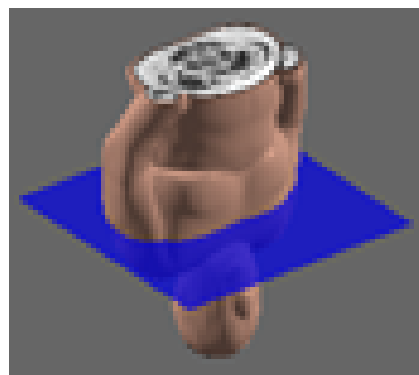


Fuente: Suministrada por el autor.

4.4.2 Corte Sobre La Visualización

Se pueden realizar un corte arbitrario sobre la visualización de manera sencilla y precisa, el cual permite dividir la visualización en dos y a la vez obtener la imagen bidimensional de este corte, para de este modo poder acceder a partes del interior de la visualización. Esta herramienta nos permite realizar cortes sobre objetos en zonas donde no es posible realizarlos físicamente por razones de manipulación de la muestra. Esta herramienta es muy utilizada para el diseño de entrenadores quirúrgicos.

Figura 34 Plano de corte.



Fuente: Suministrada por el autor.

Dicho corte es realizado por medio de un plano de corte, el cual es visualizado en primera instancia en la posición del medio de la geometría con respecto a la dirección de apilamiento, que para el caso de este proyecto es el plano z, esto se

puede ver en la figura 34. Luego el usuario define una orientación arbitraria para la geometría del corte. Para lo cual se utilizan unos parámetros para geometría del plano de corte, los cuales son 3 rotaciones, una para cada uno de los ejes del plano cartesiano y un desplazamiento por el vector normal al plano de corte. Se desarrolló una interfaz para permitir la definición de estos parámetros en una forma más amigable para el usuario (ver apéndice A).

A partir de lo anterior se procede colocando un límite a los parámetros, para el caso de las 3 rotaciones ese límite va de una rotación de 0 a 180 grados. Lo anterior se debe a que el corte se realiza con un plano con forma rectangular, por lo cual, un ángulo mayor a 180 produce un corte equivalente a un corte realizado con un ángulo dentro del rango mencionado anteriormente.

Para el caso del desplazamiento por el vector normal es un poco más complejo ya que con cualquier cambio en el alguno de los ángulos de rotación haría cambiar el vector normal y por lo tanto cambiarían también los límites para este desplazamiento, además se desea que el límite se encuentre en el punto exacto donde se termina el volumen visualizado.

De acuerdo a lo anterior, implementamos un algoritmo semejante al algoritmo de punto medio en el cálculo de raíces, sólo que en este caso se utiliza la matriz de apilamiento donde se busca en qué nivel de desplazamiento del plano de corte por el vector normal se encuentra el final del volumen. Se asume como inicio de este algoritmo al plano equivale al menor límite en la dirección de desplazamiento de la matriz y al plano equivalente al mayor límite en la dirección de desplazamiento de la matriz, los cuales dependen de los límites de la matriz.

Como continuación, se promedian los valores de posición de estos planos sobre el vector normal. El valor resultante de este promedio nos da la posición de la ubicación del nuevo plano y posteriormente se promedian los valores de intensidad de los voxels que pertenecen al plano. Del resultado final de los promedios anteriores se tiene que: si el resultado valor es mayor a cero se asume este plano como el plano de límite inferior y si el valor es igual a cero se asume el plano como el plano de límite superior, esto se repite hasta que la ubicación del nuevo plano no varíe. Lo anterior, se debe realizar para cada uno de los 2 límites (el superior y el inferior), cada uno de ellos pertenecientes a la dirección positiva y la posición negativa del vector normal.

Por lo general, se realiza el corte sobre el volumen visualizado como tal, pero este método se apoya en el manejo de tarjetas graficas, por lo cual, para el presente trabajo, no se puede utilizar este método. Para este proyecto se realiza el corte sobre la matriz de apilamiento generando dos matrices, cada una para una de las mitades del corte. Posteriormente se realiza una visualización tridimensional para cada una de las mitades en la que queda dividido el volumen.

De acuerdo a lo anterior, para realizar el corte sobre matriz de apilamiento se crea una matriz con un tamaño igual a la distancia máxima para el peor caso. Esta distancia sería una de las esquinas de la matriz de apilamiento, la distancia se calcula desde el centro de la matriz de apilamiento hasta alguna de las esquinas. Luego se extraen los índices de la matriz, los cuales son rotados y trasladados por el vector normal. La rotación de estos índices produce un gran costo computacional, por lo cual se decide utilizar un plano en lugar de una matriz reduciendo significativamente el peso computacional generado en la rotación de los índices. En consecuencia, luego de rotar los índices se desplazan en dirección del vector normal al plano.

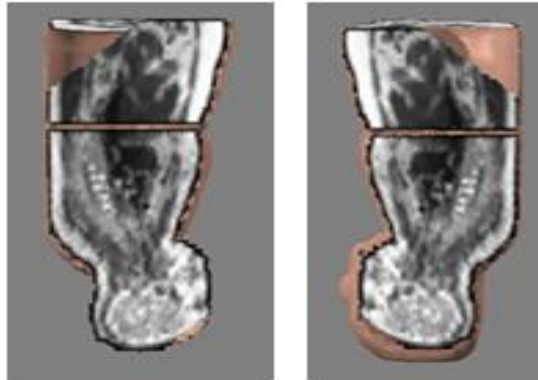
Teniendo los índices del plano ya rotados y desplazados podemos generar los demás índices de la matriz. Esto se logra calculando los índices de los planos paralelos y continuos al plano obtenido. En este punto se debe tener en cuenta el desplazamiento, ya que si el desplazamiento es mayor o igual a cero se generan los demás planos en dirección positiva del vector normal y si el desplazamiento es negativo se generan los demás planos en dirección negativa al vector normal. Como el plano tiene algún nivel de desplazamiento sólo se generara cierta cantidad de planos igual a la diferencia entre la distancia ya calculada para el peor caso y el valor del desplazamiento.

Algunos de los índices pueden estar por fuera de la matriz de apilamiento, por lo cual son eliminados ya que producirían errores al trabajar con el indexado de esta matriz. En consecuencia, se crea otra matriz de ceros de igual tamaño a la matriz de apilamiento, luego, se copian los valores de de los índices de la matriz de apilamiento a la nueva matriz y en la matriz de apilamiento estos valores los convertimos en cero; de esta manera obtenemos dos matrices de apilamiento una para cada mitad del corte.

Con las dos matrices de apilamiento se generan la visualización tridimensional para cada mitad del corte, por los métodos mencionados con anterioridad. Las reconstrucciones se notan escalonadas debido al corte. Para solucionar lo anterior

se realiza un suavizado de las dos matrices de apilamiento antes de realizar la visualización. El suavizado que se aplica es igual al utilizado en el proceso de visualización tridimensional por el método de *surface rendering*. En la figura 35 se observa las visualizaciones finales de los dos cortes.

Figura35 Imágenes obtenidas al realizar un corte sobre la visualización tridimensional.



Fuente: Suministrada por el autor.

Otra de las cosas que se realizan es la obtención de la imagen bidimensional del corte, la cual obtenemos extrayendo los valores de la matriz de apilamiento con los índices del primer plano de la matriz y realizando una proyección al plano de vista. En la figura 35 podemos observar la imagen obtenida del corte sobre la visualización tridimensional. En la aplicación diseñada se incluye una aplicación perteneciente al Toolbox de imágenes de Matlab, con el cual podemos calcular distancias sobre la imagen obtenida del corte sobre la visualización tridimensional (ver apéndice A).

5. METODOLOGÍA Y RESULTADOS

En este capítulo se mostraran los resultados obtenidos por la aplicación diseñada para este proyecto. Este estudio de resultados se ha dividido en tres partes principales:

- Surface Rendering.
- Corte arbitrario.
- Volume Rendering.

5.1 DESCRIPCIÓN DEL EQUIPO USADO

Para la realización de las pruebas, se dispuso de un equipo con las siguientes características:

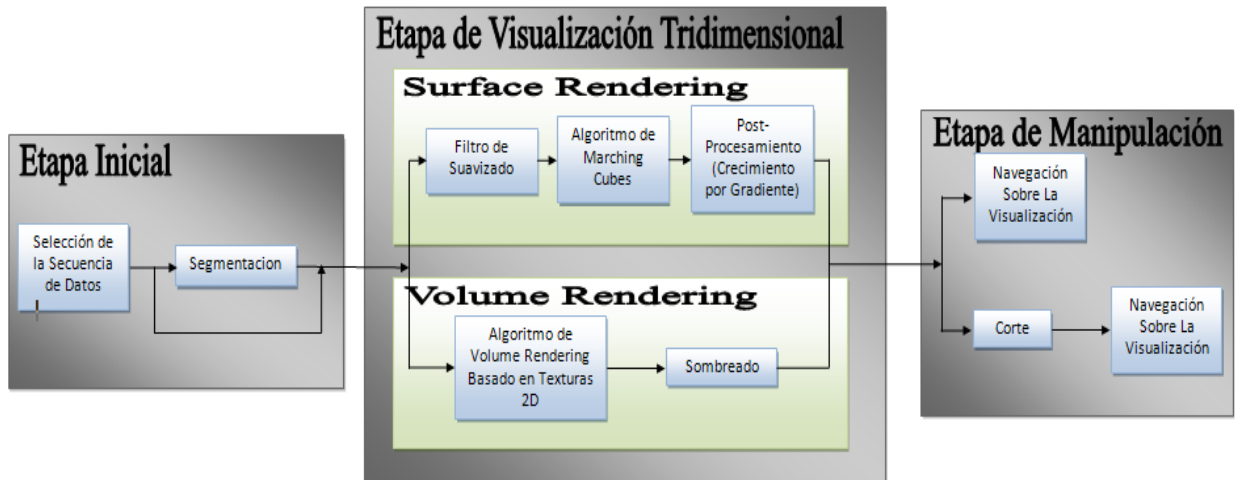
- **Procesador:** Intel Pentium IV a 1800 MHz
- **Memoria de Sistema:** 512 MB.
- **Tarjeta aceleradora de gráficos:** Integrada.
- **Sistema Operativo:** Microsoft Windows XP Professional Service Pack 2.

5.2 ESQUEMA DE BASICO DEL FUNCIONAMIENTO DE LA APLICACIÓN

El esquema utilizado para el funcionamiento de la aplicación, se puede observar en la figura 36. El cual se divide en tres etapas basicas. Las cuales son:

- Etapa Inicial.
- Etapa De Visualizacion Tridimensional.
- Etapa De Manipulacion.

Figura 366 Esquema Utilizado Para El Funcionamiento De La Aplicación.



Fuente: Suministrada por el autor.

5.2.1 Etapa inicial

En esta etapa, lo primero que se hace es seleccionar la secuencia de imágenes a procesar. Esto se realiza oprimiendo el botón de abrir de la interfaz de visualización tridimensional, la cual se puede visualizar en la figura 57. Luego, se puede segmentar o no la secuencia, ya que la segmentación no es un requisito para realizar la visualización tridimensional. En el caso de segmentación se implementaron varios tipos, tales como umbralización (otsu, entropía, isodata, conservación del momento, análisis de concavidad del histograma, selección interactiva y manual), detección de borde (roberts, prewitt, sobel, laplaciano, laplaciano de gauss y canny), algoritmos de clustering (k-means y fuzzy c-means) y watershed. Lo anterior se puede ver con más detalle en la sección 3.3.

En el caso de la segmentación, se puede realizar etapas de pre-segmentación y post-segmentación. Las cuales, son opcionales. En la primera, se obtiene la mejor óptica posible de la imagen a segmentar. Para ello, se implementaron tres algoritmos, los cuales son especificación del histograma y reducción de ruido. La especificación del histograma, obtiene como resultado una imagen que utiliza más apropiadamente todo el rango disponible de niveles de gris y como consecuencia de ello, tendrá una apariencia más balanceada, tal como se puede observar en la sección 3.2.2.

Continuando con lo anterior. En la reducción de ruido, se elimina la mayor cantidad de ruido que sea posible, con el fin de mejorar el proceso de segmentación. Para la aplicación se implementaron tres filtros de reducción de ruido los cuales son, el filtro promedio, el filtro de la mediana y el filtro de Wiener. Lo anterior se puede ver con más detalle en la sección 3.2.3.

Para el caso de la pre-segmentación, es utilizada en segmentaciones que dan como resultado imágenes binarias. tal es el caso, de umbralización y detectores de bordes. debido a que, muchas veces aparecen falsas uniones de caracteres, o bien caracteres separados en trozos no conexos donde no debieran. La pre-segmentación trata de mejorar la binarización, utilizando tratamiento morfológico para cerrar contornos o adelgazarlos.

5.2.2 Etapa de visualización tridimensional

En esta etapa se realiza la visualización tridimensional de la secuencia de imágenes, para ello se escoge el método a utilizar. Estos son surface rendering y volume rendering.

a. SURFACE RENDERING

En este caso, la secuencia de imágenes pasa, primero por un filtro de suavizado, con el cual se evita que la malla resultante de esta reconstrucción se vea cuadrada y escalonada, como se puede ver en la figura 24. Luego, se utiliza el algoritmo de Marching Cubes para generar la malla. El cual, consiste básicamente en determinar que partes del volumen formarán la isosuperficie, para ello se crea un cubo lógico a partir de dos capas contiguas que contenga en sus ocho vértices la información de la intensidad que corresponde a ese punto. Posteriormente, se comparan los vértices con el valor umbral introducido y se crea un índice con la información de la pertenencia de los distintos vértices del cubo al volumen. Para este caso, se coloca un cero si el valor de vértice es menor que el umbral o un uno si el valor del vértice es mayor o igual al umbral, este índice se puede observar en la figura 22.

Una vez creado el índice, se procede a determinar qué forma tendrá la porción de superficie asociada a dicho cubo, para ello existen 15 casos predeterminados, los cuales se pueden observar en la figura 23. Ahora, se determina los lugares del cubo por donde pasa la superficie para generar los triángulos que unan los puntos de intersección. A continuación se mueve (marcha) al siguiente cubo y así

sucesivamente. Hasta generar todos los triángulos de los cubos pertenecientes a la frontera.

Para finalizar, se realiza un post procesamiento. En el cual se utiliza el crecimiento por gradiente para eliminar pequeñas imperfecciones debidas a la falta de precision, como se puede observar en la figura 26. el crecimiento por gradiente, consiste en desplazar cada vertice de un triangulo un valor igual ala inversa de la magnitud del gradiente en este punto. el desplazamiento se realiza en direccion al vector normal del vertice.El procedimiento de Marching cubes se puede ver con más detalle en la sección 4.2.1.

b. VOLUME RENDERING

En el caso, de utilizar el metodo de volume rendering. Se utiliza el algoritmo de volume rendering basado en texturas 2D,Consiste en separar el volumen de datos en una serie de texturas 2D, las cuales se montan sobre una serie de polígonos y se lanza los planos de forma perpendicular a alguno de los ejes, esto se puede observar en la figura 28.Lo anterior se realiza para cada uno de los tres ejes y se superponen. Esto se hace para evitar una visualización errónea del volumen debida a los espacios entre cada una de las texturas superpuestas.

A continuacion, se realiza el sombreado. Lo cual es asignara cada voxel un nivel de opacidad y un nivel de color proporcional a la Intensidad registrada en ese punto, como se puede observar en la figura 30.Esto hace posible que sean visibles las estructuras internas de la visualización. Además, los puntos de vista del *volume rendering* se pueden situar dentro del objeto. Esto permite explorar detalles en el interior del objeto tales como tumores y otras anomalías.El procedimiento de *Volume Rendering* basado en texturas 2D se puede ver con más detalle en la sección 4.3.1.

5.2.3Etapa de manipulación

Esta etapa se utiliza cuando el usuario requiere hacer cambios sobre la visualizacion tridimensional. en este caso hay dos posibilidades. Las cuales son navegacion sobre la visualizacion o corte. La primera, permite que el usuario cambie el punto de vista de la visualización, por medio de tres movimientos. Los cuales son rotación, desplazamiento y zoom. En el caso del movimiento de rotación,permite girar la visualización tridimensional para obtener diferentes vistas del volumenvisualizado. Para lo cual, se implementaron dos rotaciones una

alrededor del eje Z y la otra alrededor de un eje intermedio entre el eje X y el eje y. un ejemplo de esto se puede observar en la figura 31.

Continuando con lo anterior, El movimiento de zoom permite acercar y alejar de la visualización tridimensional para tener una mejor observación de un lugar específico de la visualización. En la figura 32 se puede observar un ejemplo de acercamiento de la visualización. Por último, El movimiento de desplazamiento permite mover la visualización tridimensional para desplazarla, esto sirve cuando se ha acercado de forma considerable a la visualización y de este modo poder navegar sobre ella. En la figura 33 se puede observar un ejemplo de desplazamiento de la visualización tridimensional. La navegación sobre la visualización se puede observar con más detalle en la sección 4.4.1.

Para el caso de corte, el usuario puede realizar un corte arbitrario sobre la visualización. Por medio del cual, se divide la visualización en dos y a la vez obtener la imagen bidimensional de este corte, para de este modo poder acceder a partes del interior de la visualización. Dicho corte es realizado por medio de un plano de corte, el cual es visualizado en primera instancia en la posición del medio de la geometría con respecto a la dirección de apilamiento, que para el caso de este proyecto es el plano z, esto se puede ver en la figura 34.

Luego, el usuario define una orientación arbitraria para la geometría del corte. Para lo cual se utilizan unos parámetros para el plano de corte, los cuales son 3 rotaciones, una para cada uno de los ejes del plano cartesiano y un desplazamiento por el vector normal al plano de corte. Para lo anterior, hay que colocando un límite a los parámetros, para el caso de las 3 rotaciones ese límite va de una rotación de 0 a 180 grados. En el caso del límite del desplazamiento por el vector normal es un poco más complejo ya que con cualquier cambio en el alguno de los ángulos de rotación haría cambiar los límites para este desplazamiento.

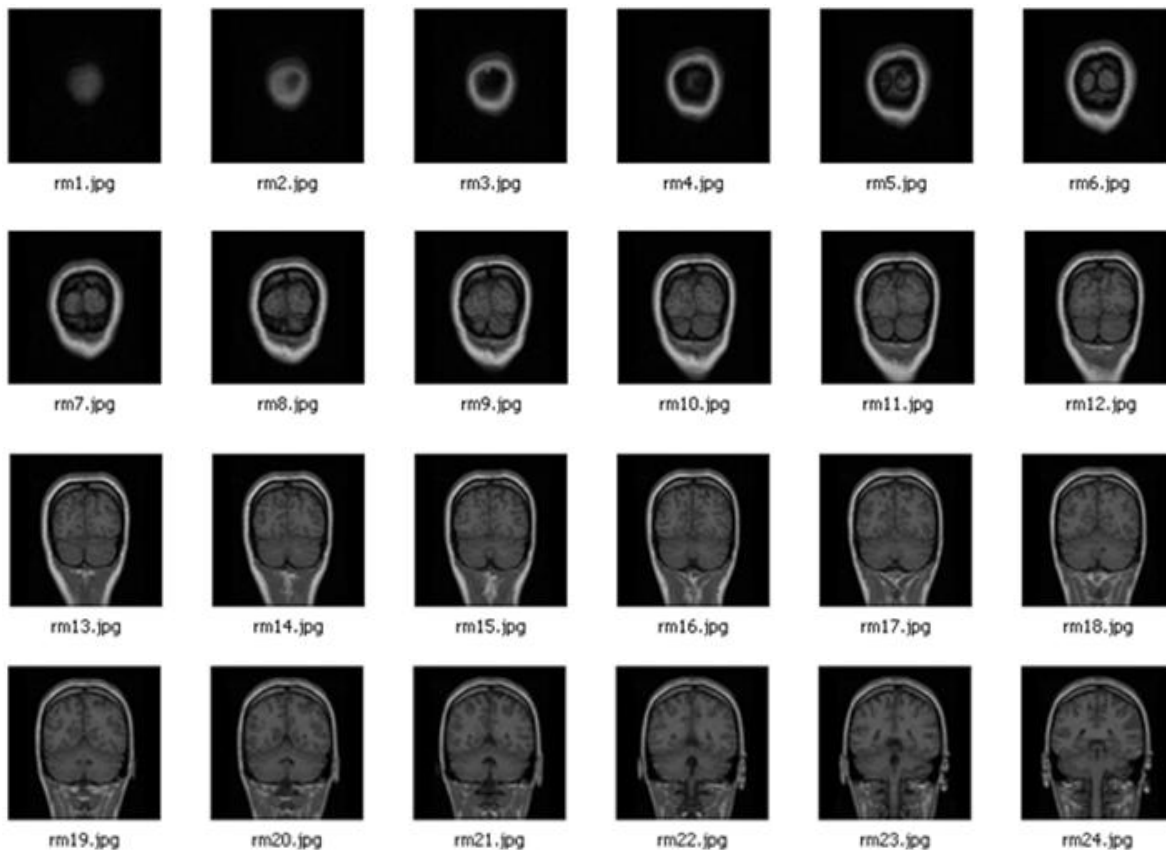
Para lo anterior, se implementó un algoritmo semejante al algoritmo de punto medio en el cálculo de raíces. Por medio del cual, se busca en qué nivel de desplazamiento del plano de corte por el vector normal se encuentra el final del volumen, esto se realiza para las dos direcciones del vector normal. El procedimiento de corte y el cálculo de los límites del desplazamiento por el vector normal se pueden ver con más detalladamente en la sección 4.4.2.

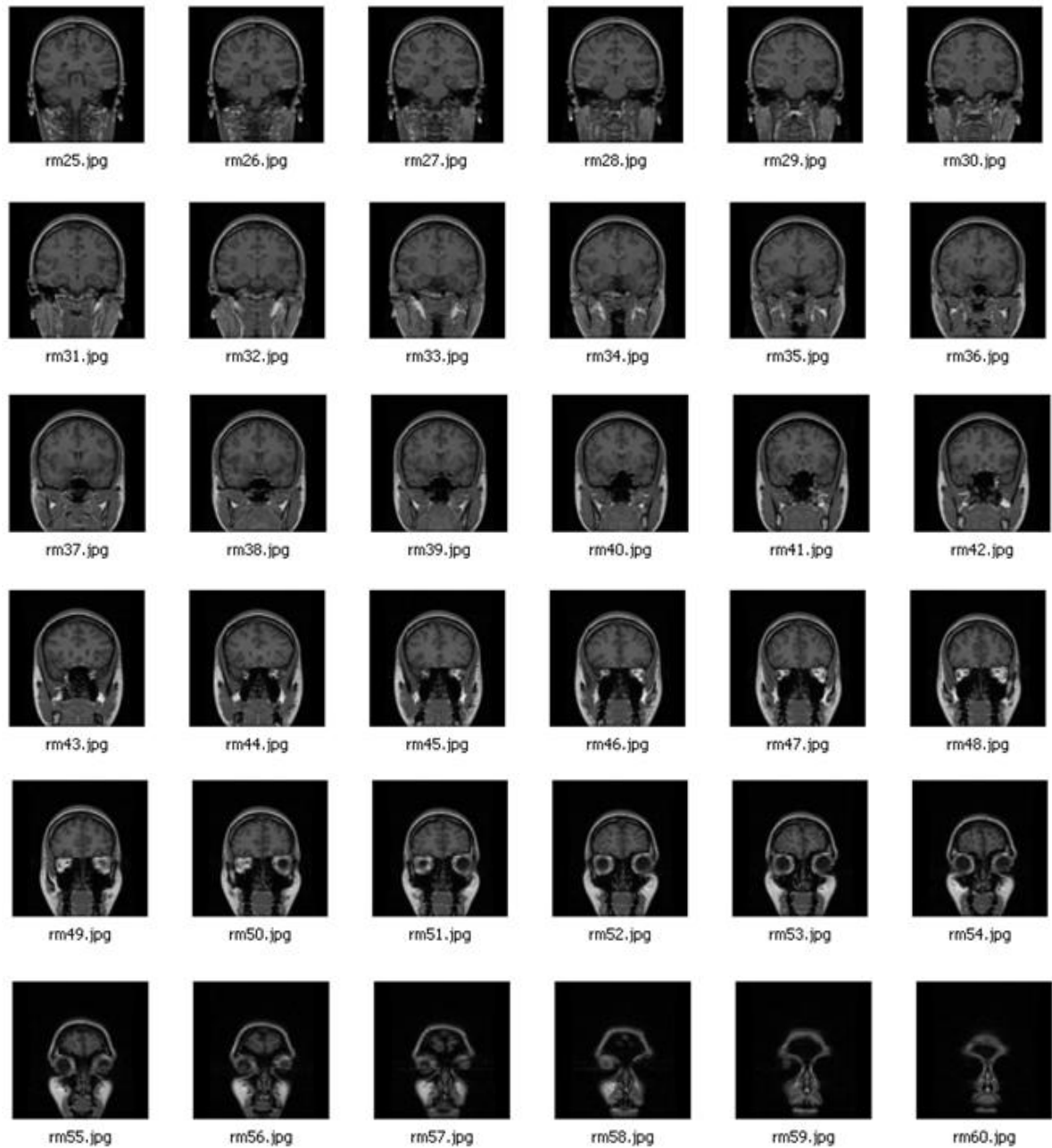
Para concluir, se genera una visualización tridimensional para cada mitad del corte, por los métodos mencionados con anterioridad. Las reconstrucciones se notan escalonadas debido al corte. Para solucionar lo anterior se realiza un suavizado igual al utilizado en el proceso de visualización tridimensional por el método de *surface rendering*. En la figura 35 se observa las visualizaciones finales de los dos cortes. Para cada una de estas mitades, se puede realizar la navegación sobre la visualización.

5.3 PRUEBAS REALIZADAS

Se realización 5 pruebas con igual número de bases de datos. Estas pruebas son mostradas a continuación. La primera base de datos, se muestra a continuación. La cual está compuesta por 60 imágenes, con un tamaño de 256 x 256.

Figura 377 Secuencia De Imágenes De Cortes Coronales De Una Cabeza.



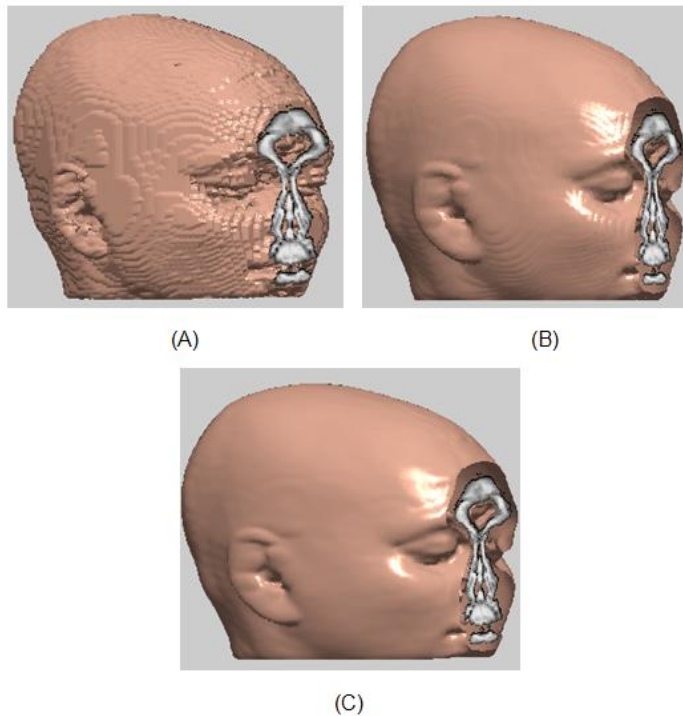


Fuente: <http://es.dreamstime.com/fotos-de-archivo-libres-de-regal-iacuteas-mri-la-radiograf-iacutea-del-cerebro-image7521608>.

A continuación, se muestran las imágenes obtenidas al realizar la visualización por el método de *Surface rendering* a las secuencia de imágenes de la figura 36. En la figura 37-a se observa una visualización bastante rudimentaria debido a que la malla generada es muy cuadrada y escalonada. Esto se soluciona fácilmente utilizando un filtro de suavizado Gaussiano como se puede observar en la figura

37-b, dando como resultado una malla más adecuada. Pero aun se observan zonas donde se alcanzan a visualizar pequeñas imperfecciones debidas a la falta de precisión. Para solucionar esto se aplica el método de crecimiento por gradiente como se puede observar en la figura 37-c.

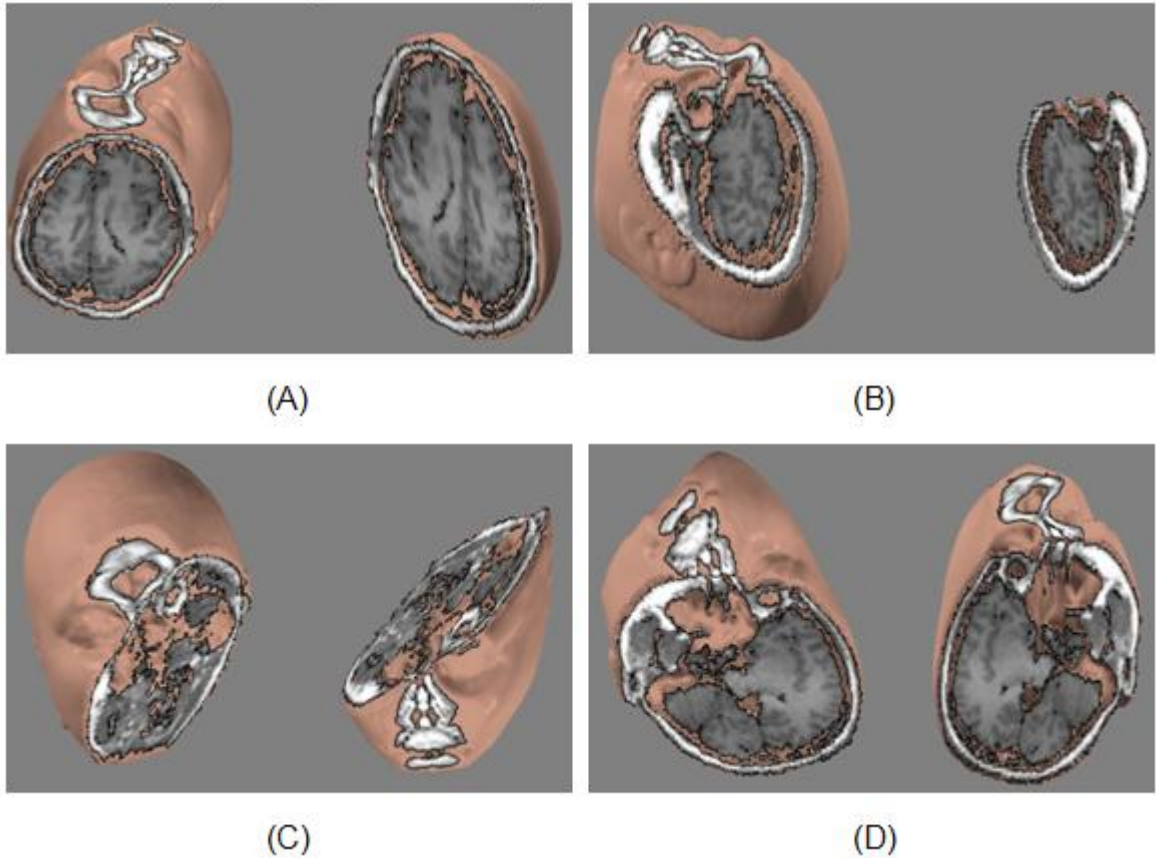
Figura 388 Visualización tridimensional por el método de *surface rendering* de la secuencia de imágenes de la Figura 36: (A) Visualización sin filtro de suavizado. (B) Visualización con filtro de suavizado (C) Visualización después de la etapa de post-procesamiento (crecimiento por gradiente).



Fuente: Suministrada Por El Autor.

La segunda etapa de prueba, es la de corte arbitrario sobre la visualización, para la visualización se realizaron cuatro pruebas de corte, las cuales se pueden observar a continuación.

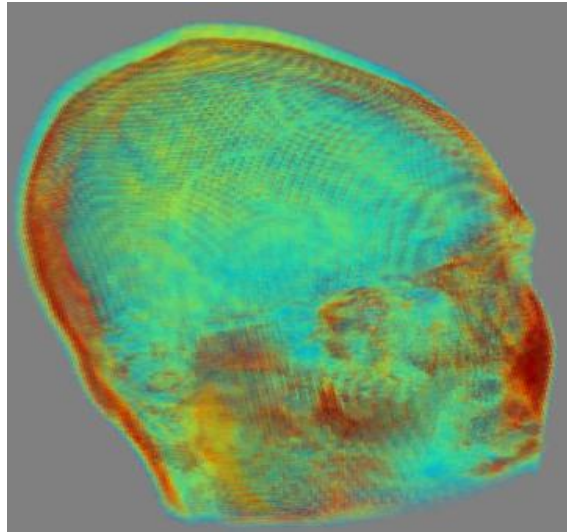
Figura 39 Cortes arbitrarios sobre la visualización de la figura 37: (A) Corte realizado con un ángulo 141 grados en el eje X, 85 grados en el eje Y, 98 grados en el eje Z y un desplazamiento de 19 por el vector normal. (B) Corte realizado con un ángulo 38 grados en el eje X, 172 grados en el eje Y, 77 grados en el eje Z y un desplazamiento de -26 por el vector normal. (C) Corte realizado con un ángulo 15 grados en el eje X, 154 grados en el eje Y, 113 grados en el eje Z y un desplazamiento de -9 por el vector normal. (D) Corte realizado con un ángulo 29 grados en el eje X, 131 grados en el eje Y, 91 grados en el eje Z y un desplazamiento de 0 por el vector normal.



Fuente: Suministrada Por El Autor.

Por último, en la tercera etapa de prueba se muestran las imágenes obtenidas al realizar la visualización por el método de *Volume rendering* a las secuencia de imágenes. En la figura 39 se puede observar el interior del volumen, como son la piel, el cráneo y en menor medida el cerebro. Esto es debido a la a las propiedades de transparencia y seudocolor asignadas a la visualización. También se observa los artefactos producidos por la falta de interpolación espacial.

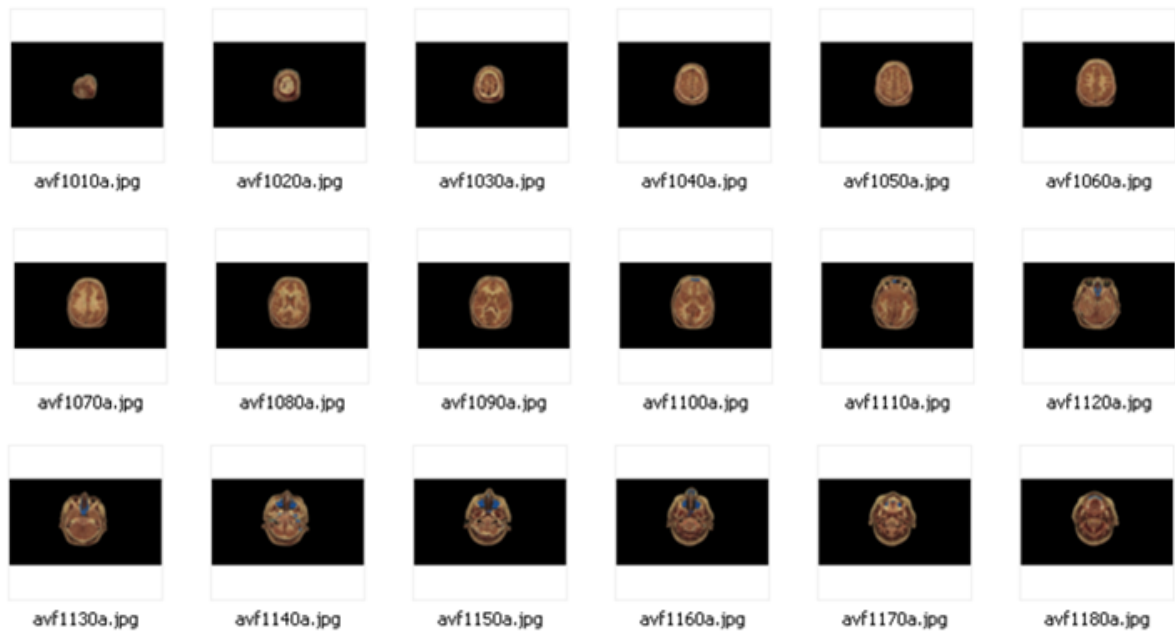
Figura 400 Visualización tridimensional por el método de *Volume Rendering* de la secuencia de imágenes de la Figura 36.

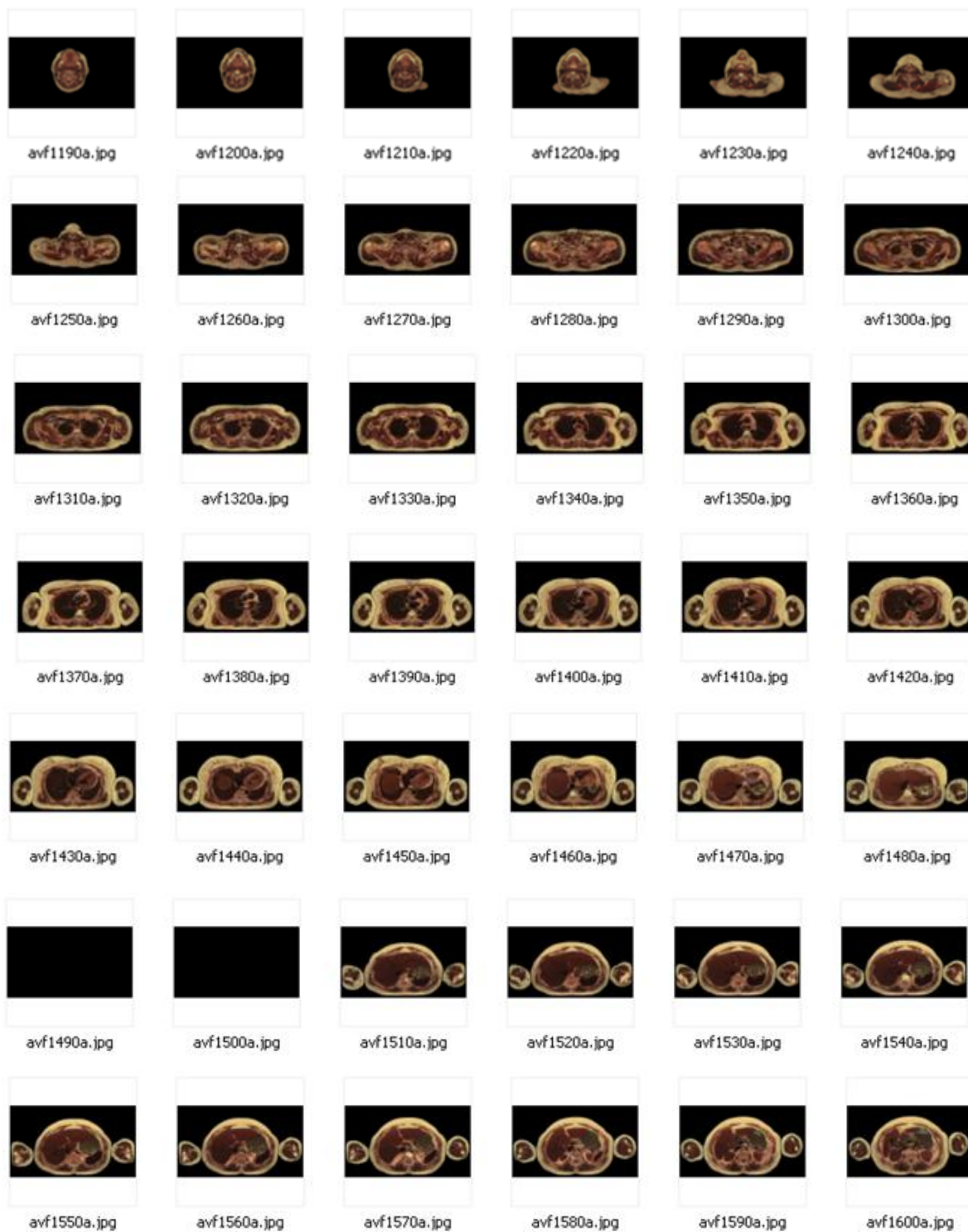


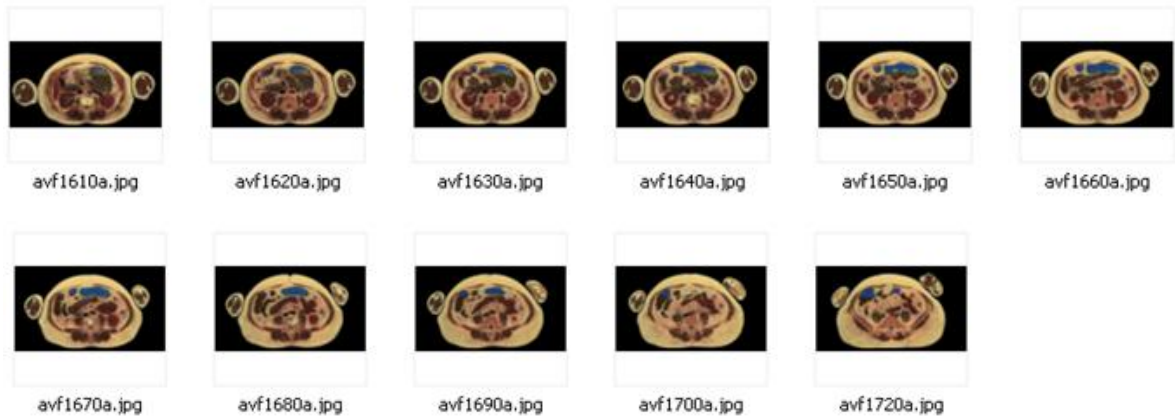
Fuente: Suministrada Por El Autor.

La segunda base de datos, se muestra a continuación. La cual está compuesta por 71 imágenes, con un tamaño de 555 x 309.

Figura 411 Secuencia De Imágenes De Cortes axiales De Un Cuerpo Humano.



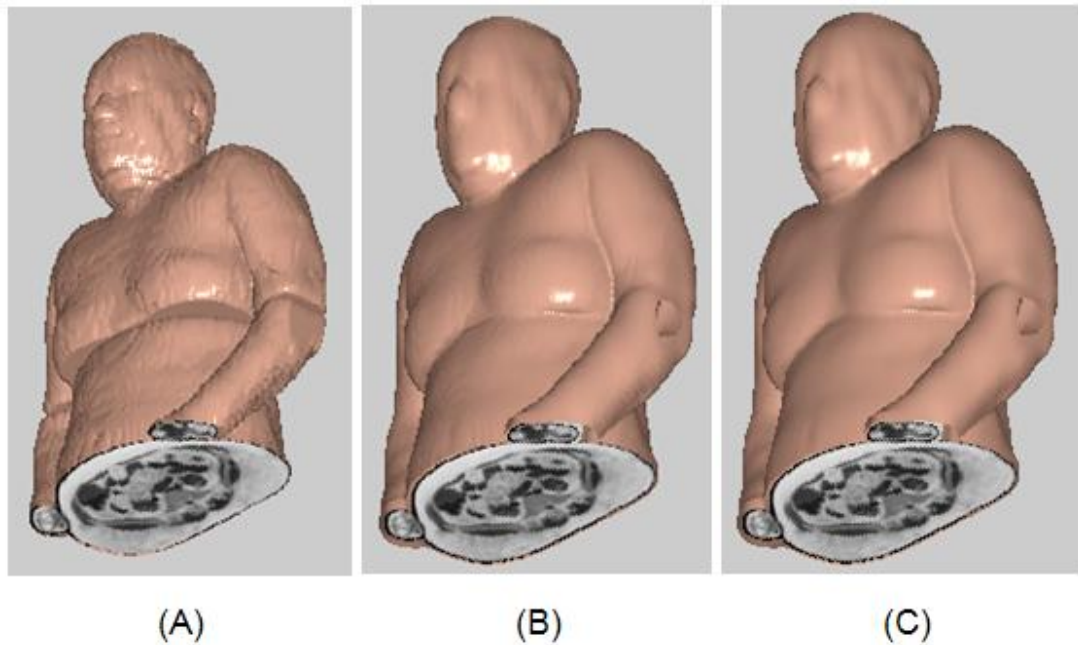




Fuente: http://www.nlm.nih.gov/research/visible/visible_human.html.

Al realizar la visualización por el método de *Surface rendering* a las secuencia de imágenes anterior, se obtienen los resultados mostrados en la figura 41.

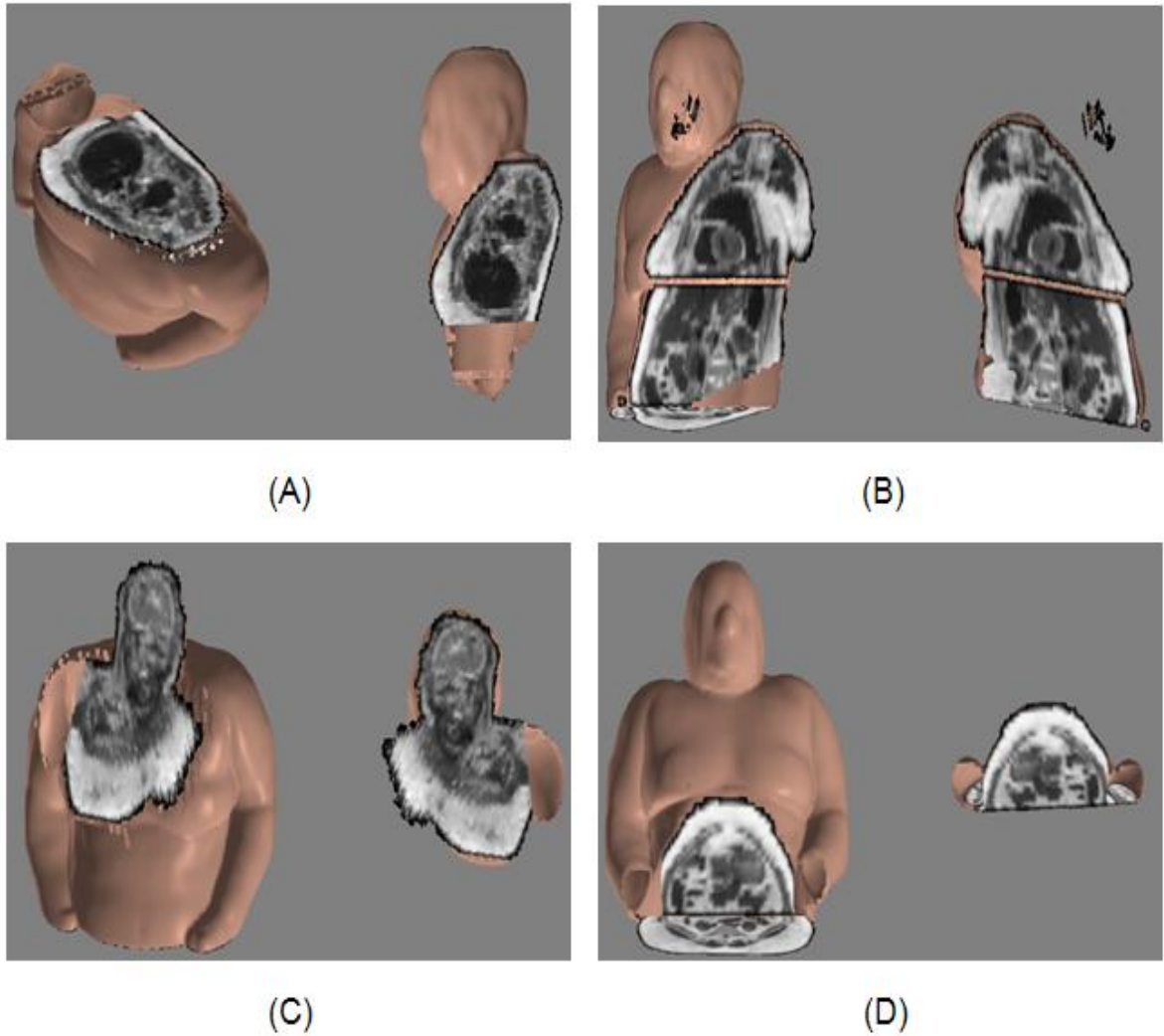
Figura 42 Visualización tridimensional por el método de *surface rendering* de la secuencia de imágenes de la Figura 40: (A) Visualización sin filtro de suavizado. (B) Visualización con filtro de suavizado (C) Visualización después de la etapa de post-procesamiento (crecimiento por gradiente).



Fuente: Suministrada Por El Autor.

En la etapa de corte arbitrario sobre la visualización, para la visualización se realizaron cuatro pruebas de corte, las cuales se pueden observar a continuación.

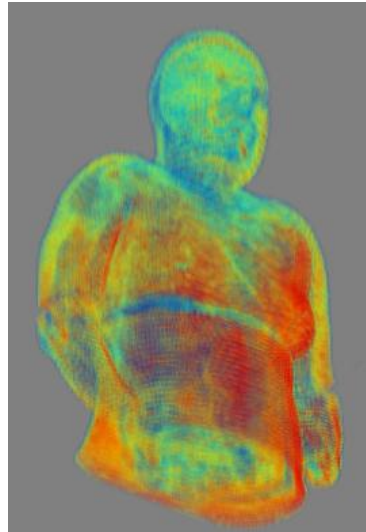
Figura 433 Cortes arbitrarios sobre la visualización de la figura 41: (A) Corte realizado con un ángulo 18 grados en el eje X, 0 grados en el eje Y, 116 grados en el eje Z y un desplazamiento de -4 por el vector normal. (B) Corte realizado con un ángulo 58 grados en el eje X, 10 grados en el eje Y, 146 grados en el eje Z y un desplazamiento de 10 por el vector normal. (C) Corte realizado con un ángulo 163 grados en el eje X, 46 grados en el eje Y, 127 grados en el eje Z y un desplazamiento de 9 por el vector normal. (D) Corte realizado con un ángulo 0 grados en el eje X, 130 grados en el eje Y, 92 grados en el eje Z y un desplazamiento de -27 por el vector normal.



Fuente: Suministrada Por El Autor.

En la tercera y última etapa de prueba se muestran las imágenes obtenidas al realizar la visualización por el método de *Volume rendering* a las secuencia de imágenes de la figura 40. La cual se observa a continuación.

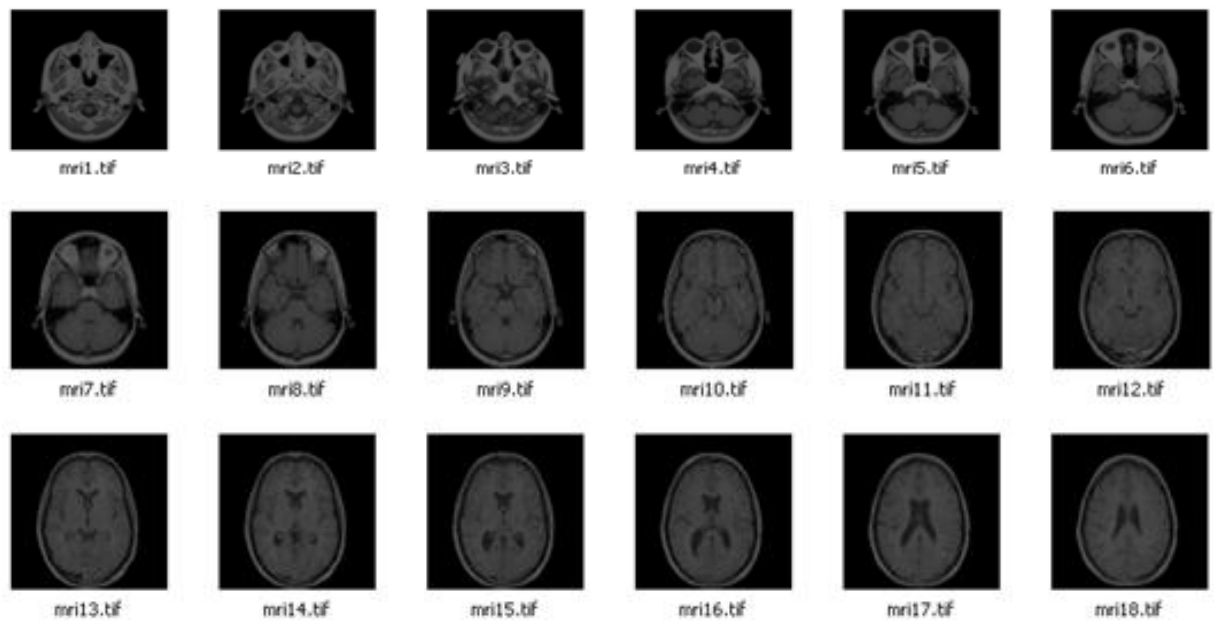
Figura 444 Visualización tridimensional por el método de *Volume Rendering* de la secuencia de imágenes de la Figura 40:

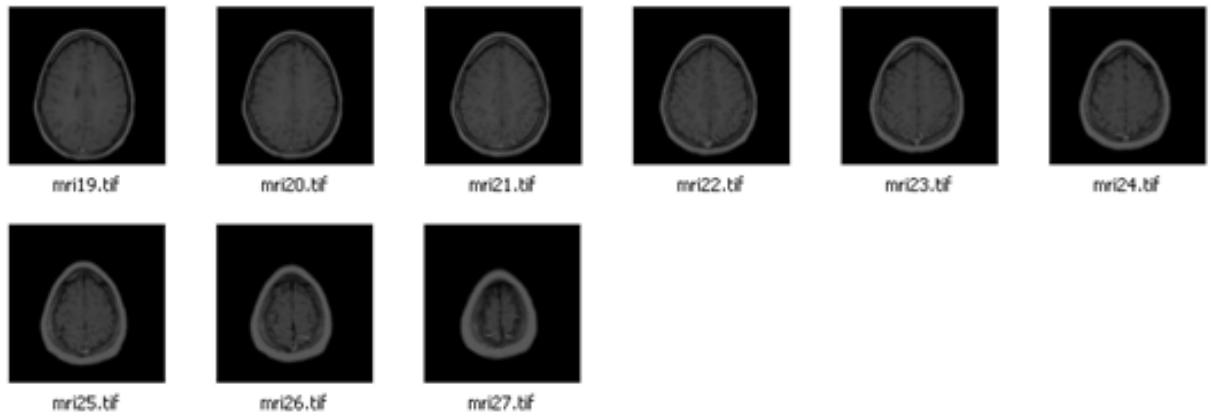


Fuente: Suministrada Por El Autor.

La tercera base de datos, se muestra a continuación. La cual está compuesta por 27 imágenes, con un tamaño de 128 x 128.

Figura 455 Secuencia De Imágenes De Cortes Axiales De Una Cabeza.

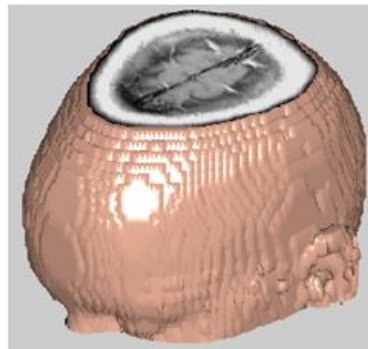




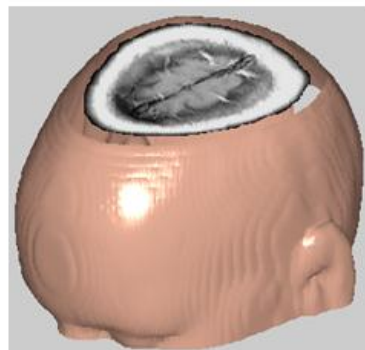
Fuente: Imágenes Pertenecientes a Matlab.

Al realizar la visualización por el método de *Surface rendering* a las secuencia de imágenes anterior, se obtienen los resultados mostrados en la figura 45.

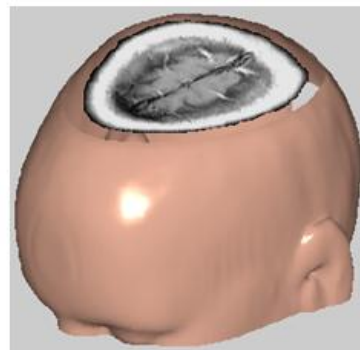
Figura 466 Visualización tridimensional por el método de *surface rendering* de la secuencia de imágenes de la Figura 44: (A) Visualización sin filtro de suavizado. (B) Visualización con filtro de suavizado (C) Visualización después de la etapa de post-procesamiento (crecimiento por gradiente).



(A)



(B)

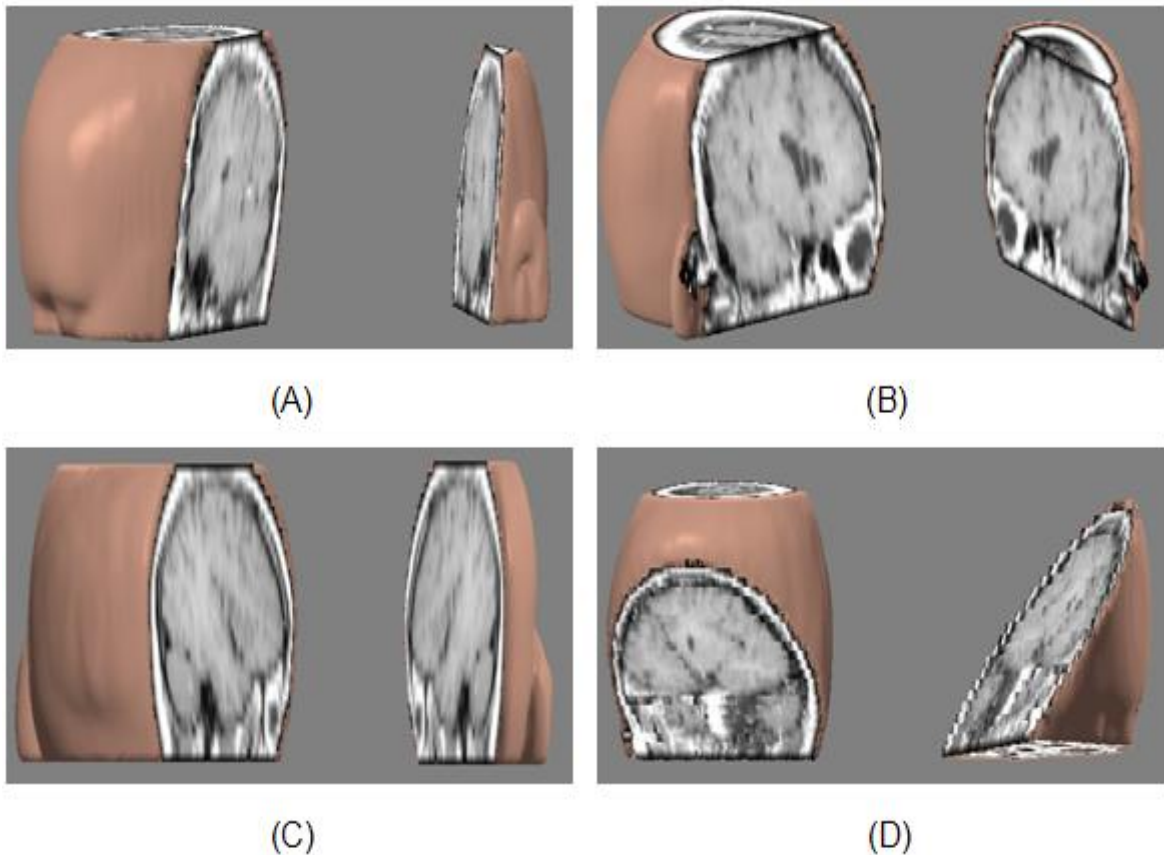


(C)

Fuente: Suministrada Por El Autor.

En la etapa de corte arbitrario sobre la visualización, para la visualización se realizaron cuatro pruebas de corte, las cuales se pueden observar a continuación.

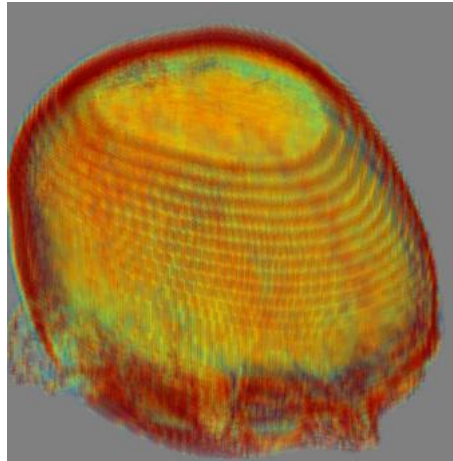
Figura 477 Cortes arbitrarios sobre la visualización de la figura 45: (A) Corte realizado con un ángulo 133 grados en el eje X, 61 grados en el eje Y, 0 grados en el eje Z y un desplazamiento de 23 por el vector normal. (B) Corte realizado con un ángulo 38 grados en el eje X, 105 grados en el eje Y, 93 grados en el eje Z y un desplazamiento de -14 por el vector normal. (C) Corte realizado con un ángulo 90 grados en el eje X, 90 grados en el eje Y, 90 grados en el eje Z y un desplazamiento de -20 por el vector normal. (D) Corte realizado con un ángulo 0 grados en el eje X, 22 grados en el eje Y, 41 grados en el eje Z y un desplazamiento de -9 por el vector normal.



Fuente: Suministrada Por El Autor.

En la tercera etapa de prueba se muestran las imágenes obtenidas al realizar la visualización por el método de *Volume rendering* a las secuencia de imágenes. La cual se observa a continuación.

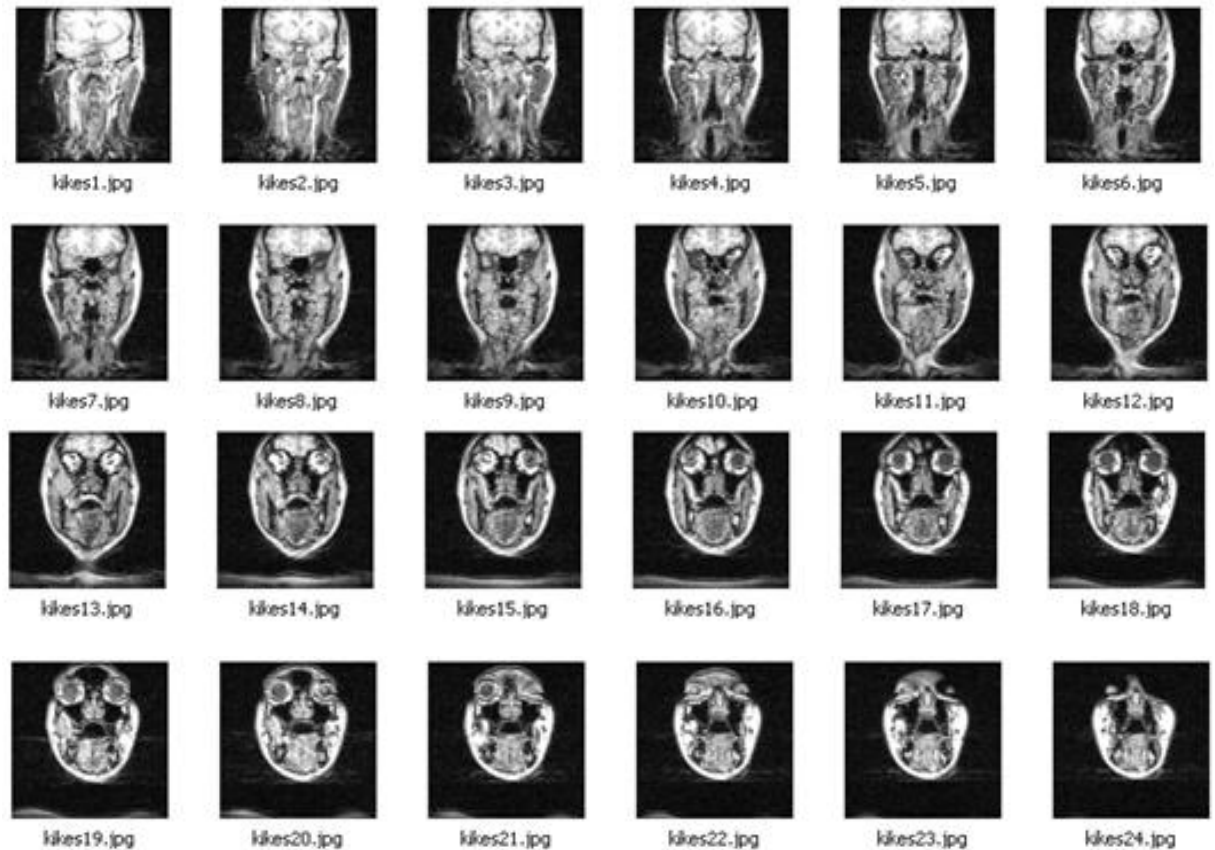
Figura 488 Visualización tridimensional por el método de *Volume Rendering* de la secuencia de imágenes de la Figura 44:

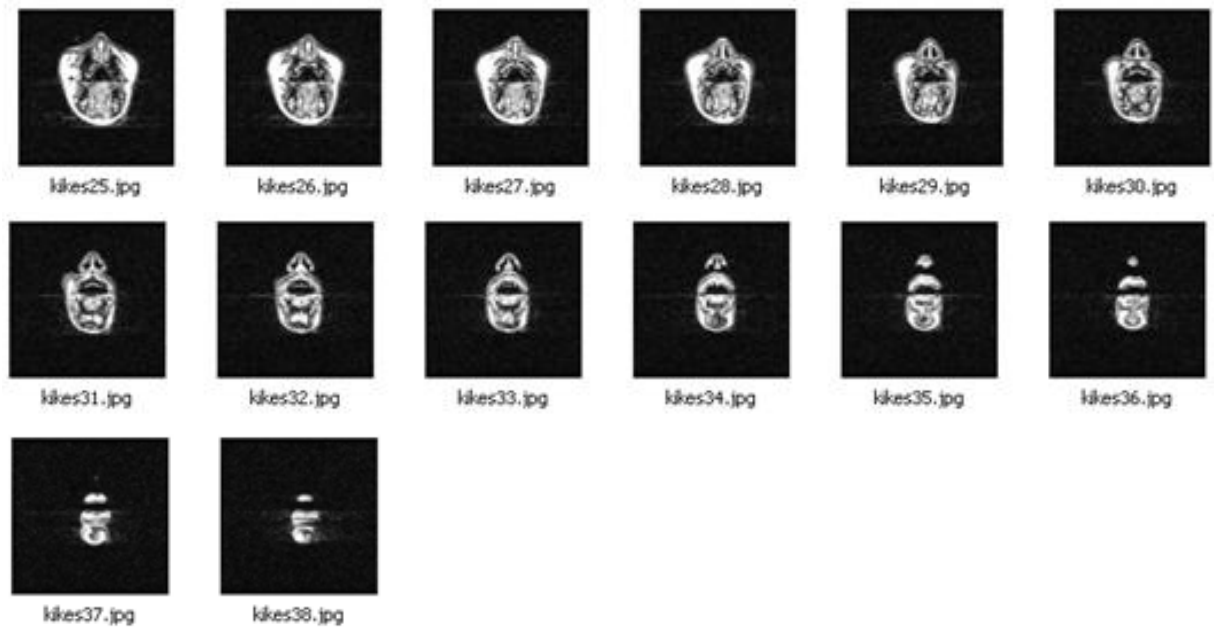


Fuente: Suministrada Por El Autor.

La segunda base de datos, se muestra a continuación. La cual está compuesta por 38 imágenes, con un tamaño de 256 x 256.

Figura 499 Secuencia De Imágenes De Cortes Coroneales De Una Parte De La Cabeza.

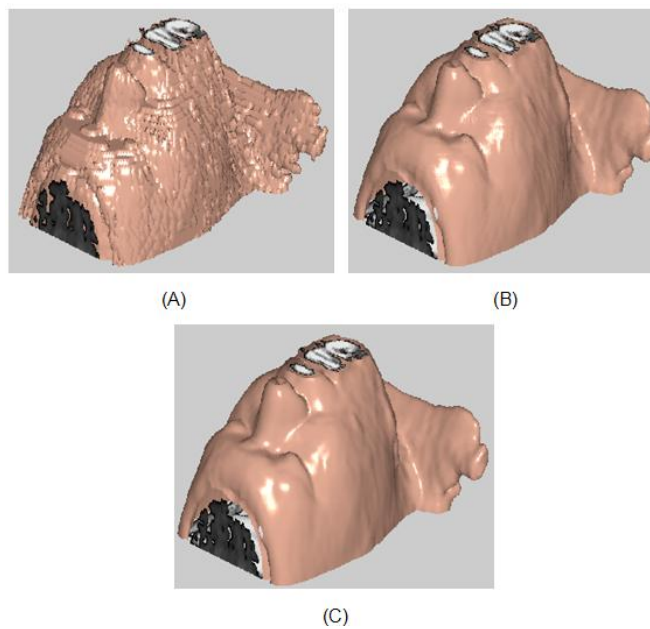




Fuente: http://www.canstockphoto.es/imagenes-fotos/mri_4.html.

Al realizar la visualización por el método de *Surface rendering* a las secuencia de imágenes anterior, se obtienen los resultados mostrados en la figura 49.

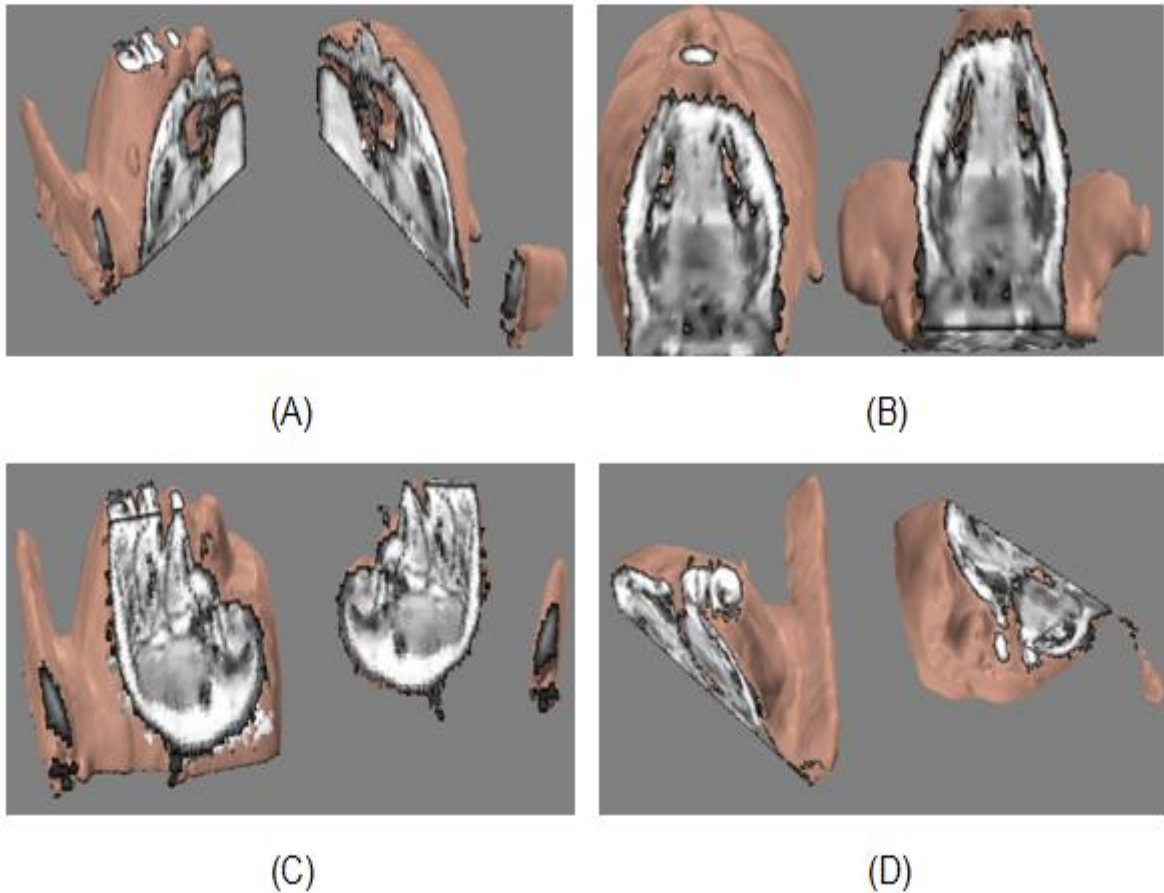
Figura 500 Visualización tridimensional por el método de *surface rendering* de la secuencia de imágenes de la Figura 48: (A) Visualización sin filtro de suavizado. (B) Visualización con filtro de suavizado (C) Visualización después de la etapa de post-procesamiento (crecimiento por gradiente).



Fuente: suministrada por el autor.

En la etapa de corte arbitrario sobre la visualización, para la visualización se realizaron cuatro pruebas de corte, las cuales se pueden observar a continuación.

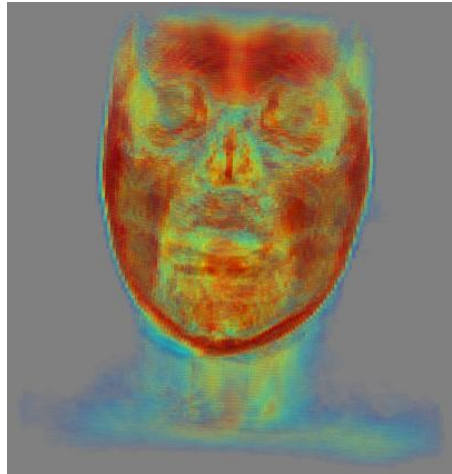
Figura 511 Cortes arbitrarios sobre la visualización de la figura 49: (A) Corte realizado con un ángulo 96 grados en el eje X, 27 grados en el eje Y, 64 grados en el eje Z y un desplazamiento de -15 por el vector normal. (B) Corte realizado con un ángulo 3 grados en el eje X, 45 grados en el eje Y, 83 grados en el eje Z y un desplazamiento de 8 por el vector normal. (C) Corte realizado con un ángulo 36 grados en el eje X, 160 grados en el eje Y, 116 grados en el eje Z y un desplazamiento de -14 por el vector normal. (D) Corte realizado con un ángulo 25 grados en el eje X, 70 grados en el eje Y, 18 grados en el eje Z y un desplazamiento de 0 por el vector normal.



Fuente: Suministrada Por El Autor.

En la tercera etapa de prueba se muestran las imágenes obtenidas al realizar la visualización por el método de *Volume rendering* a las secuencia de imágenes. La cual se observa a continuación.

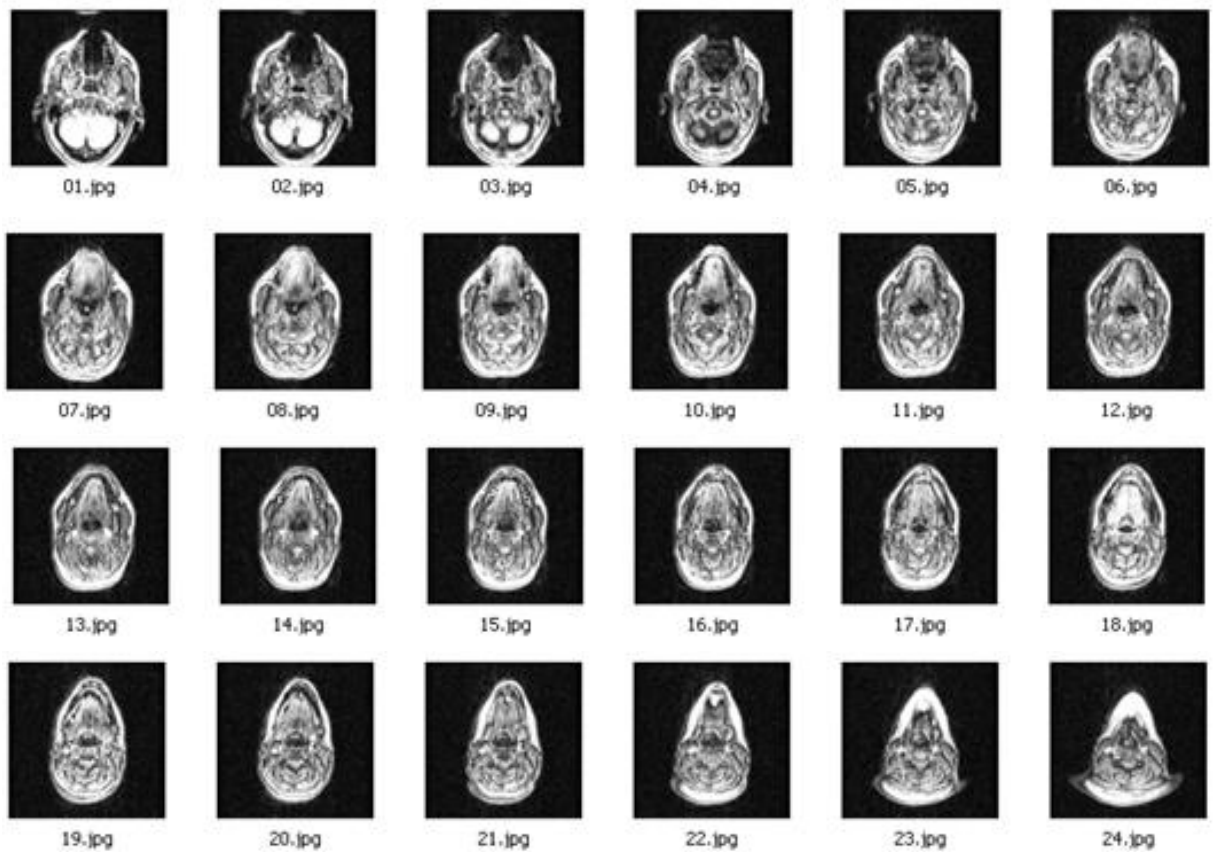
Figura 522 Visualización tridimensional por el método de *Volume Rendering* de la secuencia de imágenes de la Figura 48:

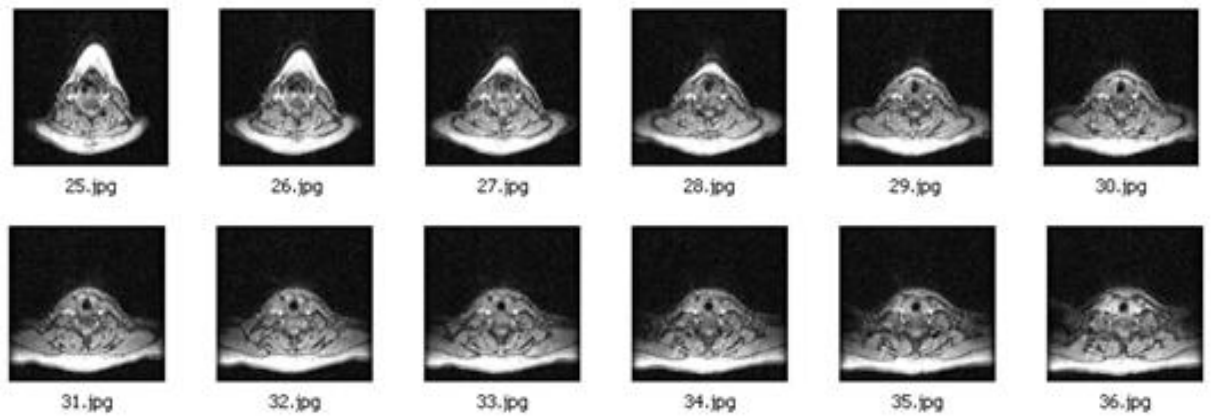


Fuente: Suministrada Por El Autor.

La segunda base de datos, se muestra a continuación. La cual está compuesta por 36 imágenes, con un tamaño de 256 x 256.

Figura 533 Secuencia De Imágenes De Cortes Axiales De Una Cabeza.

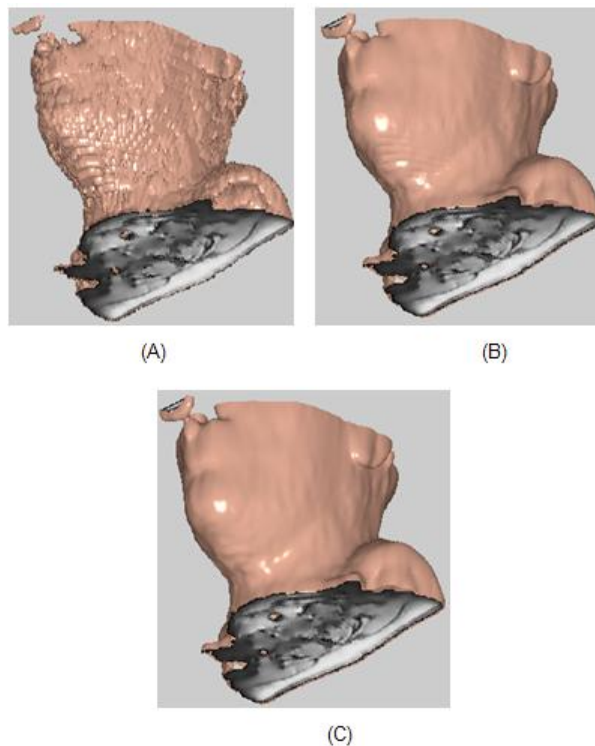




Fuente: http://www.canstockphoto.es/imagenes-fotos/mri_4.html.

Al realizar la visualización por el método de *Surface rendering* a las secuencia de imágenes anterior, se obtienen los resultados mostrados en la figura 53.

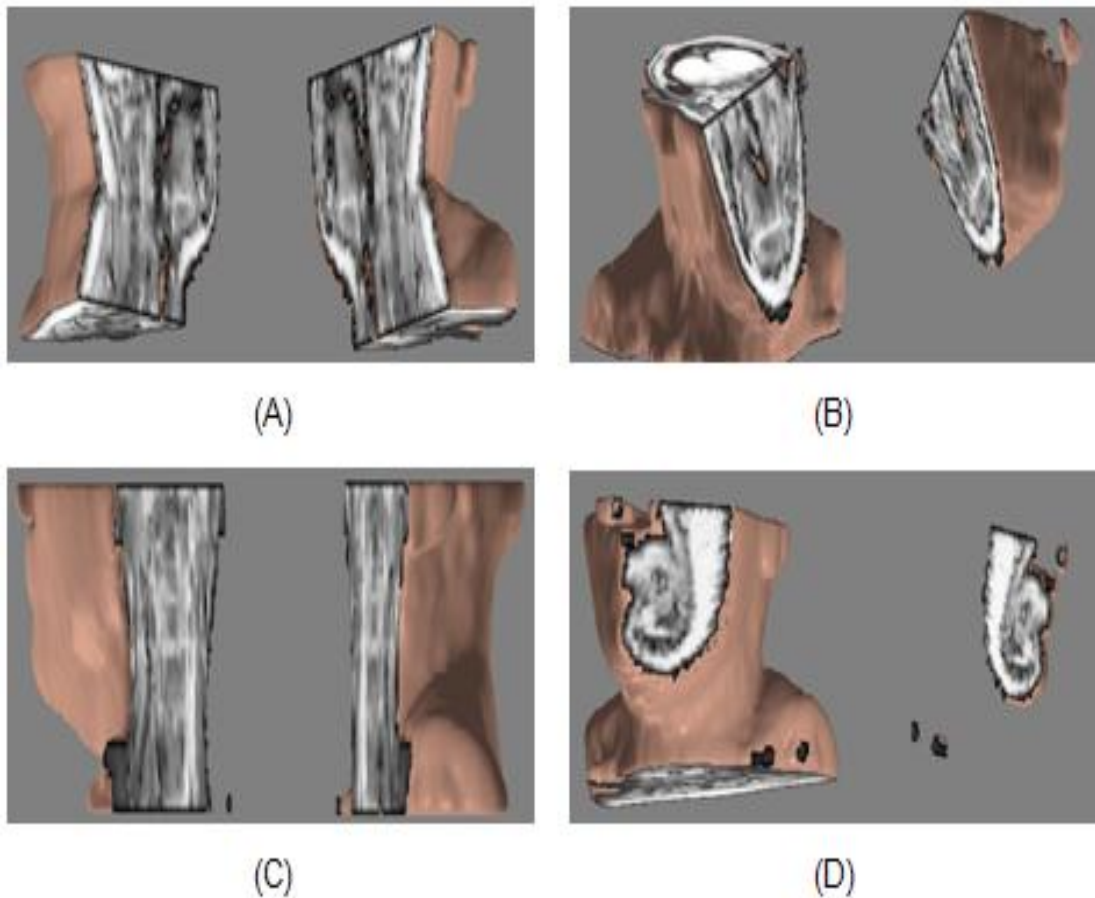
Figura 54 Visualización tridimensional por el método de *surface rendering* de la secuencia de imágenes de la Figura 52: (A) Visualización sin filtro de suavizado. (B) Visualización con filtro de suavizado (C) Visualización después de la etapa de post-procesamiento (crecimiento por gradiente).



Fuente: Suministrada Por El Autor.

En la etapa de corte arbitrario sobre la visualización, para la visualización se realizaron cuatro pruebas de corte, las cuales se pueden observar a continuación.

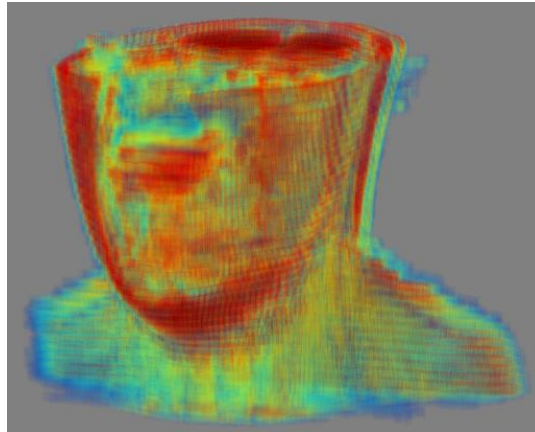
Figura 555 Cortes arbitrarios sobre la visualización de la figura 53: (A) Corte realizado con un ángulo 81 grados en el eje X, 36 grados en el eje Y, 64 grados en el eje Z y un desplazamiento de 0 por el vector normal. (B) Corte realizado con un ángulo 130 grados en el eje X, 0 grados en el eje Y, 146 grados en el eje Z y un desplazamiento de 10 por el vector normal. (C) Corte realizado con un ángulo 82 grados en el eje X, 85 grados en el eje Y, 0 grados en el eje Z y un desplazamiento de 8 por el vector normal. (D) Corte realizado con un ángulo 118 grados en el eje X, 148 grados en el eje Y, 118 grados en el eje Z y un desplazamiento de 32 por el vector normal.



Fuente: Suministrada Por El Autor.

En la tercera etapa de prueba se muestran las imágenes obtenidas al realizar la visualización por el método de *Volume rendering* a las secuencia de imágenes. La cual se observa a continuación.

Figura 566 Visualización tridimensional por el método de *Volume Rendering* de la secuencia de imágenes de la Figura 52:

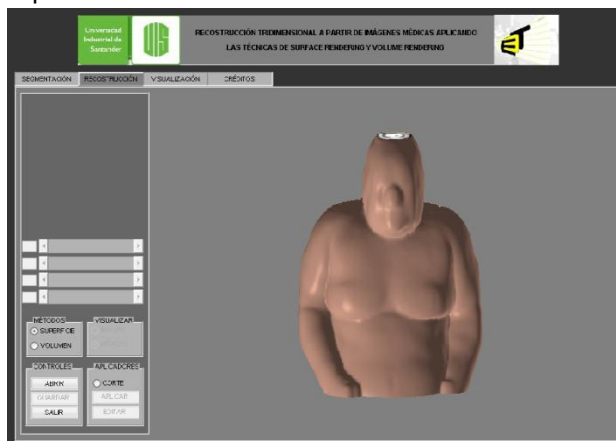


Fuente: Suministrada Por El Autor.

En general, los resultados de estas pruebas son satisfactorios. Se obtuvieron visualizaciones tridimensionales por el método se *Surface Rendering* en la cual se obtuvo una malla tridimensional muy precisa acercándose a la realidad de los datos. También en el caso de los cortes, se obtuvieron visualizaciones tridimensionales de las dos mitades muy exactas y con muy buena definición del corte arbitrario.

En el caso de *Volume Rendering*, se obtuvo una visualización tridimensional bastante buena con algunos problemas de bandas sobre el volumen debido a la falta de interpolación espacial, además se logro asignar propiedades de opacidad y mapa de color (seudocolor). Los cuales permiten observar el interior del volumen y diferenciar parte del volumen como la piel, el hueso y etc.

Figura 577 Imagen de la aplicación desarrollada en la interfaz de visualización tridimensional:

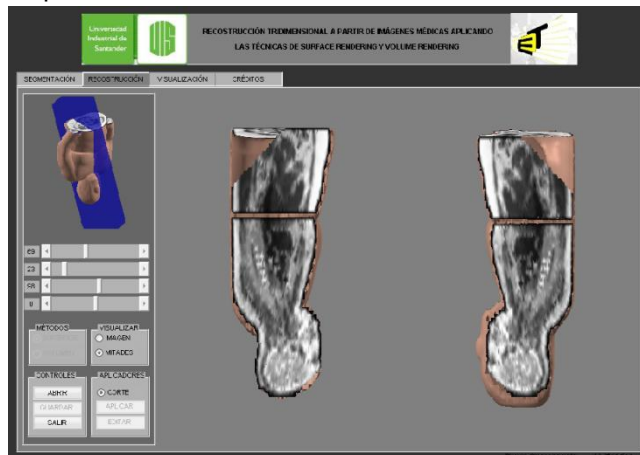


Fuente: Suministrada Por El Autor.

Los resultados mostrados en este capítulo se obtuvieron con la aplicación desarrollada para este proyecto. Para el caso de visualización tridimensional la interfaz es como se observa en la figura 56.

En esta interfaz podemos escoger que tipo de reconstrucción que vamos a realizar, por *Volume Rendering* o por *Surface Rendering*. Si queremos realizar un corte seleccionamos la opción corte. Cuando esto sucede, la interfaz de visualización tridimensional sufre algunos cambios como se puede ver en la figura 57.

Figura 58 Imagen de la aplicación desarrollada en la interfaz de corte:



Fuente: Suministrada Por El Autor.

Obsérvese que la imagen de la derecha en la que estaba la visualización tridimensional desaparece y esta visualización aparece en una imagen, ubicada arriba a la izquierda. También aparece un plano en color azul en la mitad de la visualización tridimensional con respecto al eje z, este plano es el plano de corte. Además, el control de rotación es activado, este control de rotación es el que nos permite rotar y trasladar el plano.

6. CONCLUSIONES Y RECOMENDACIONES

El proyecto se ha encaminado en dos líneas diferentes pero estrechamente ligadas, la segmentación y la visualización tridimensional. En cuanto a la segmentación se han tratado temas como Umbralización, Extracción De Contornos, Clustering Y Watershed. Además se implementaron etapas de presegmentación y postsegmentación.

Los aportes más importantes en torno a la umbralización han sido principalmente dos: la primera, es la implementación de los métodos de umbralización más comunes como Otsu, Entropía, Isodata, Conservación Del Momento, Análisis De Concavidad Del Histograma Y Selección Interactiva. La segunda, es el desarrollo de una aplicación que permite al usuario cambiar o elegir un umbral a su gusto para cada imagen, basándose en la visualización y análisis del histograma. Además le permite visualizar de forma inmediata el resultado de la umbralización al usar dicho umbral.

Por su parte, la etapa de pre segmentación se centró en la reducción del ruido y la modificación del histograma. La sección de reducción de ruido permite al usuario escoger entre el filtro Promedio, el filtro de la mediana y el filtro de Wiener, los cuales son los filtros más eficaces en reducción de los tipos de ruido más comunes en las imágenes digitales como lo son: el ruido Gaussiano, impulsivo, frecuencias y multiplicativo. De otra manera, la sección de modificación del histograma permite realizar ecualización del histograma y realce del contraste de una imagen. Lo concreto está en que con la etapa de presegmentación se logra mejorar la calidad y una mejor interpretación de la imagen para luego obtener una visualización tridimensional más exacta.

Al seguir avanzando, encontramos que en la etapa de postsegmentación se incluyeron dos opciones de operaciones morfológicas compuestas como son la operación de open y la operación de close. Además se agregaron operaciones básicas como invertir una imagen, llenado de huecos y eliminar bordes de una imagen. Todas estas operaciones permiten al usuario realizar un proceso de postsegmentación más versátil. Para obtener un mejor resultado de la segmentación de acuerdo al tipo de imagen segmentada.

Otra opción incluida en la postsegmentación es la posibilidad de escoger qué regiones de la imagen segmentada serán incluidas en el resultado final de la segmentación. Esta posibilidad sólo se incluye para métodos de segmentación por regiones como son Watershed, Fcm Y Kmeans. Las regiones que componen la imagen segmentada para los casos de segmentación por regiones son representadas por colores distintos para cada región.

También se incluyo la posibilidad de escoger que tipo de visualización de la segmentación se puede observar. Para lo cual se incluyo dos posibilidades, mascara y aplicada. La máscara permite ver el resultado de la segmentación como tal y la aplicada permite ver el resultado de aplicar la máscara a la imagen original.

Para el caso de la visualización tridimensional se desarrollo por las técnicas de *volume rendering* y *surface rendering*. Para el caso de *volume rendering* se usó el método basado en texturas y para el caso de *surface rendering* se uso el método de maching cube. También se desarrollo una herramienta de corte arbitrario; este corte permite al usuario dividir en dos mitades la visualización tridimensional. El usuario genera el corte por medio de un plano de corte, el cual pose 3 movimientos de rotaciones y uno de traslación, las rotaciones se realizan sobre cada uno de los ejes del plano de coordenadas y la translación se realiza sobre el vector normal al plano de corte.

En este punto, se da la opción de escoger la forma de visualización del corte realizado sobre la visualización tridimensional. Se puede escoger entre Imagen y mitades. La opción imagen permite visualizar solamente la imagen bidimensional de corte y la opción mitades permite visualizar cada de las dos mitades del corte.

Se desarrolló un sistema de navegación sobre el volumen de la visualización el cual permite al usuario variar el punto de vista de la visualización y también permite acercarse y alejarse del volumen para observarlo con mayor precisión. También se desarrolló una aplicación la cual permite modificar los mapas de color y de transparencia dando al usuario una gran variedad de posibilidades de visualización del volumen. Permitiéndole ocultar y resaltar algunos rangos de intensidades.

Así mismo, se creó una interfaz que permite visualizar de forma individual un grupo de imágenes de corte guardas o un grupo de imágenes almacenadas y se incluyo en ella, una pequeña aplicación del Toolbox de imágenes de Matlab llamada *Imgtools* la cual permite calcular distancias sobre la imagen.

En general se desarrolló una aplicación muy completa para la segmentación, visualización y corte. Con un funcionamiento completamente independiente una de otra. Se logró independizar la aplicación de la plataforma de Matlab, permitiendo de esta forma que el usuario pueda ejecutar la aplicación en cualquier equipo sin necesidad de tener instalado Matlab. Se creó un instalador de la aplicación el cual nos permite integrar todos los archivos necesarios para el funcionamiento de la aplicación en uno solo, lo cual le permite al usuario la facilidad de transporte de la aplicación de un equipo a otro.

Los resultados obtenidos han sido satisfactorios en general. Todas las metas trazadas han sido alcanzadas, aunque también muchos aspectos son mejorables. Las directrices se han marcado poniendo los pilares y sentando las bases para posteriores trabajos.

6.1 FUTURAS MEJORAS

Si muchas son las aportaciones realizadas, no menos son las posibles mejoras a este proyecto. Comenzando por la segmentación, se podría implementar otros métodos de segmentación como son segmentación a color, por Crecimiento De Regiones, por División Y Fusión (“Split & Merge”) y por Redes Neuronales. También se podría implementar otros métodos de cálculo de umbral como podría ser Algoritmo Fondo-Simetría (Background-Simetry), Método del Triángulo, Método de P-tile, Método de Error Mínimo, Método de la Matriz de Coocurrencia, Método De Conjuntos Difusos, Método De Relajación, Método De Medias Móviles, Método de Deravi y Pad y Método de Scatter Plot.

Sin embargo, la principal mejora aplicable en la visualización tridimensional es la reducción del tiempo para el procesamiento y manipulación de la visualización tridimensional, donde aparece la posibilidad de paralelizar el algoritmo. Esto es dividir la tarea entre varios equipos que trabajaran simultáneamente. Otra opción a la hora de mejorar el rendimiento es recurrir a hardware específico de renderización, el cual puede aplicarse a muchos problemas de visualización. También es común ver en el mercado tarjetas que contienen un chip gráfico que cuenta con capacidades aceleradoras avanzadas y una elevada capacidad de almacenamiento en memoria.

Otra mejora es la funcionalidad que proveen los chips gráficos recientes en la capacidad de manejar simultáneamente múltiples elementos de textura. Pues una

aproximación a la visualización de un volumen de datos con varias clases al mismo tiempo puede ser alcanzada utilizando una textura para cada clase. Sin embargo, el límite en la capacidad de unidades de manejo de texturas lo pone el hardware que se emplea.

6.2 FUTUROS TRABAJOS

Las líneas de desarrollo de futuros trabajos que se pueden seguir son muy amplias. Se considera que los próximos trabajos que se pueden desarrollar son simuladores quirúrgicos y sistema de cirugía guiada por imagen.

Los simuladores quirúrgicos de entrenamiento basados en realidad virtual incorporan gráficos de gran realismo y en algunos casos el sentido del tacto. Esto permite al cirujano practicar diversos tipos de procedimientos bajo el control del computador y con ello reducir riesgos en los pacientes.

Es de destacar cómo empresas y universidades están desarrollando sistemas que permitan simular procedimientos más complejos, como suturar tejido, insertar un catéter en una vena o manipular los instrumentos quirúrgicos usados en laparoscopia.

En el caso de cirugía guiada por imagen permite una navegación quirúrgica en tiempo real. Se basa en determinar las coordenadas de los instrumentos de cirugía en la mesa de operaciones, la profundidad de la lesión y la dirección del abordaje quirúrgico a través de la prolongación virtual de la punta de los instrumentos. Además, esta técnica muestra sus posiciones en las imágenes de estudios previos desde diversas perspectivas (axial, sagital, coronal, vistas trayectoria, vista de sonda), brindando una realimentación visual de la posición de los instrumentos y la trayectoria que se está siguiendo durante el acto quirúrgico, lo que favorece y simplifica la toma de decisiones. Además permite guiar al cirujano en zonas de difícil visibilidad o acceso.

BIBLIOGRAFÍA

- S. W. Bartz D., "Parallel construction and isosurface extraction of recursive tree structures," in WSCG, Vol III, Plzen, 1998.
- C. R. Bartz D., Mueller K., "A practical evaluation of four popular *volume rendering* algorithms," in ACM Symposium on Volume Visualization, 2000.
- S. Beucher and F. Meyer, "The morphological Approach to Segmentation: The Watershed Transformation: Mathematical Morphology and Its application to image processing", E. R. Dougherty, Ed. New York: Marcel Decker, ch. 12, pp.433-481, 1993.
- Brice, C., Fennema. C, "Scene analysis using regions", Artificial Intelligence., vol. 1. pp. 205 – 226, 1975.
- Bueno G, Musse O, Heitz F, Armspach JP, "3D Segmentation of Anatomical Structures in MR Images on Large Data bases", Magnetic Resonance Imaging, ch. 19, pp. 73-88, 2001.
- J. Canny A, "Computational Approach to Edge Detection", IEEE Trans. on Pattern Analysis and Machine Intelligence, pp. 679-698, 1986.
- Carlton, C.G., Mitchell. O.R., "image segmentation using texture and gray level", Proc. IEEE conf. Pattern Recognition and Image Processing. Troy. New York, no. 6 – 8, pp. 387 – 391, 1977.
- Chien, Y.P, Fu, K.S., "Preprocessing and feature extraction of picture patterns", TR – EE 74 – 20, Purdue University, West Lafayette, Indiana, 1974.
- Cline, H.E, Lorensen, W.E., "*Marching Cubes*: A High Resolution 3D Surface Construction Algorithms", Computer Graphics, v.21, pp. 163-169, Julio 1987.

- Coatrieux J.L., Roux C., Collorec R., " Fusion d'informations en imagerie medicale tridimensionnelle", Bulletin de liaison de la recherche en informatique et automatique, no. 132, pp. 12-16, 1991.
- M. Couprie and G. Bertrant, "Topological grayscale Watershed Transform", SPIE Vision Geometry Proceedings, vol. 3168, pp. 136- 146, 1997.
- T. W. Crockett, "Beyond the renderer: software architecture for parallel graphics and visualization". Technical Report TR-96-75, Institute for Computer Applications in Science and Engineering, 1996.
- T. W. Crockett, "AnIntroduction to parallel *rendering*". Parallel Computing, vol. 23, num. 7, pp. 819 -- 843, 1997.
- Davis, L.S., "A Survey of Edge Detection Techniques", Computer Graphics and Image Processing, vol. 4, pp. 248 – 270, 1975.
- Duda. R.O., Hart, P.E., "Use of the Hough Transformation to detect line and curves in pictures", Comm. ACM., vol. 15, pp. 11 – 15, 1972.
- Duda R. y Hart, P., "Pattern Classification and Scene Analysis", Ed. Wiley,1973.
- Dunn J. C. 1973. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, Journal of Cybernetics 3: 32-57.
- De La Escalera H. A., "Visión por Computador, fundamentos y métodos". Prentice Hall, Madrid, España, 2001.
- Fu. K.S, Mui, J. K, "A Survey on Image Segmentacion". Parttern Recognition, Vol. 13, pp. 3 – 16, 1981.
- Gerig G, Kübler R., Kikinis R, Jolesz, FA. "Nonlinear Anisotropic Filtering of MRI Data". IEEE Trans. Medical Imaging, 2:221-232,1992.
- C. A. Glasbey and G. W. Horgan, "Image analysis for the biological science", Statistics in Practice, Series Editor Vic Barnett., Jonh Wiley and Sons, 1994.

- Gonzalez R. Woods R., "Digital Image Processing", Addison Wesley, U.S.A.,1992.
- Gonzalez R. Woods R. "Digital Image Processing" Second Edition, Prentice Hall 2000.
- González y Woods, "Digital Image Processing", Ed. Prentice Hall, 2002.
- Haralick R. A. y Shapiro L., "Computer and Robot Vision", Volume I, Addison-wesley Publishing Company, USA, 1992.
- Haris K, Efstratiadis S, Maglaveras N, Katsaggelos A., "Hybrid Image Segmentation Using Watersheds and Fast Region Merging, IEEE Trans. Image Processing,7:1684-1699, 1998.
- Herman G., "Image Reconstruction from Projections: The fundamentals of Computerized Tomography", Editions Academic Press, New York, 1980.
- Honrowitz, S.L., Pavlidis, T., "picture segmentacion by Directed Split _ and _ Merge Procedure", Proc. Second Int. Joint Conf. Pattern Recognition. pp. 424 – 433, 1974.
- Kelly, M., "edge detection by computer using planning", In Machine Intelligence., vol. VI, pp. 397 – 409, Edinburgh University Press, Edinburgh, 1971.
- Philippe Lacroute, "Fast*volume rendering* using a shear-warp factorization of the viewing transformation", Ph.d. dissertation, Stanford University, 1994.
- Lecomte J., "Reconstruction d'images P.E.T. par Inversion Directe", Rapport de DEA en Signal, Télécommunications, Image Radar, Université de Rennes I, 1995.
- Mark Levoy, "Display of surfaces from volume data", IEEE Computer Graphics and Applications, 8(3):29–37, May, 1988.

- Lichtenbelt, B, Crane, R, Naqvi, S., "Introduction to *Volume Rendering*", Hewlett-Packard Professional Books, 1998.
- Lindley C. A, "Practical Image Processing in C"; John Wiley & Sons, Inc., United States of America, 1991.
- W. E. Lorensen, H. E. Cline, "*Marching cubes: a high resolution 3D surface construction algorithm*". Computer Graphics, vol. 21, num. 4, pp. 163 -- 169, 1987.
- MacQueen. J., "Some methods for classification and analysis of multivariate observations". Proceedings of the Fifth Berkeley Symp. on Math., Statistics and Probability, pp. 281, 1967.
- Martelli. A., "Edge detection using heuristic search methods", Computer Graphics and Image Processing., Vol. 1. pp. 182, 1972.
- Martelli. A., "An application of heuristic search methods to edge and contour detection", Comm. ACM., vol. 14, pp. 73 – 83, 1976.
- Tom McReynolds and David Blythe. "Advanced Graphics Programming using OpenGL". Morgan Kauffman Publishers, first edition, 2005.
- Mery, D., "Inspección Visual Automática". Primer Congreso Internacional de Ingeniería Mecatrónica. Lima, Perú, 8-12 Abril, 2002.
- U. Neumann., "Communication cost for parallel *volume-rendering* algorithms", Technical Report TR93-038, 3, 1994.
- Ollinger J., Fessler J., "Positron-Emission Tomography", IEEE Signal Processing Magazine, vol. 14, n° 1, pp. 43-55, 1997.
- Otsu, Nobuyuki, "A Threshold Selection Method from Gray – Level Histograms", IEEE Transactions on Systems, man and Cybernetics, vol. SMC – 9. no. 1, pp. 62 – 66, 1979.
- Pratt W. K., "Digital Image Processing", John Wiley & Sons, USA, 1978.

- Pun, T., "A new method for gray-level picture thresholding using the entropy of the histogram", signal process. vol. 2, pp. 223-237, 1980.
- Quistgaard J., " Signal Acquisition and Processing in Medical Diagnostic Ultrasound", IEEE Signal Processing Magazine, vol. 14, n° 1, pp. 67-74, 1997.
- Ridler, T., Calvard, S., "Picture thresholding using an iterative selection method". IEEE Trans. Systems Man Cybernet. 8, 629–632, 1978.
- Rosenfield, A., Kak. A.C., "Digital Picture Processing", Academic Press. New York, 1976.
- Rosenfeld, A., "Interactive methods in image analysis", Proc. IEEE conf. Pattern Recognition and Image Processing, Troy, New York, no. 6-8, pp. 14 – 20, 1977.
- Sahoo. P.K., Soltani, S., Wong, A.K., Chen, Y.C., "A survey of Thresholding Techniques", Computer Vision, Graphics and Image Processing vol. 1, pp. 233 – 260, 1988.
- Siedband M., " Medical Imaging Systems", in Medical Instrumentation Application and Design, Ed. J. Webster, Éditions Houghton Mifflin Company, Boston, pp. 630-693, 1992.
- Sijbers J, Scheunders P, Verhoye M, Linden AVD, Dyck DV, Raman E., "Watershed Segmentation of 3D MR Data for Volume Quantization", Magnetic Resonance Imaging, 15:679-688, 1997.
- Sternberg SR, "Grayscale Morphology", Computer Vision, Graphics and Image Processing, 35:335:355 1986.
- U. Tiede et al, "Investigation of medical 3D-rendering algorithms. IEEE Computer Graphics and Applications", vol. 10, num. 2, pp. 41 -- 53, 1990.
- Tsai, W., "Moment preserving thresholding: A new approach", Computer Vision, Graphics and Image Processing vol. 29, pp. 377 – 393, 1985.
- Udupa, J.K.; Herman, G.T., "3D Imaging in Medicine", CRD-Press ,1991.

- Vincent L, Soille P., "Watersheds in Digital Spaces: An Efficient Algorithms Based on Inmersion Simulations". IEEE Trans. Pattern Anal. Machine Intell, 13:583-598,1991.
- J. A. Waterworth, "Survey of medical applications of virtual reality," Dept of Informatics Research Report, Web report 9802, 1998.
- Rudiger Westermann and Bernd Sevenich, "Accelerated volume ray-casting using texture mapping", IEEE Visualization 2001, 2001.
- Wu, A.Y., Tosenfels, A., "Threshold selection using quatree", IEEE trans. Pattern anal. Mach. Intell. Vol. Pami – 4, pp. 90 – 94, 1982.
- R. Yagel, R. Machiraju, "Data-parallel *volume rendering* algorithms". Visual Computer, vol. 11, num. 6, pp. 319 -- 338, 1995.
- Young, I.T.; J.J. Gerbrands y L.J. Van Vliet, "Fundamentals of image processing. Delf University of Technology". Netherlands,1998.
- Zucker. S.W., "Survey. Region Growing: Childhood and Adolescence. ", Computer Graphics and Image Processing. vol. 5, pp. 382 – 399, 1976.

ANEXOS

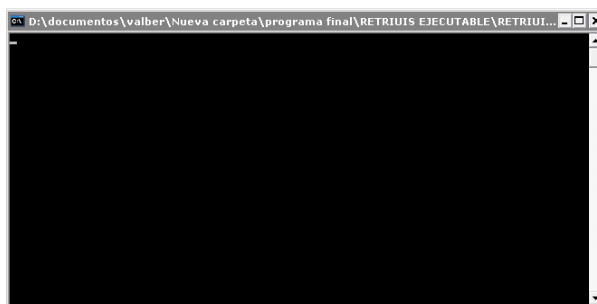
ANEXO A. MANUAL DE USUARIO

A continuación se incluye un manual de usuario, con el fin de facilitar el rápido aprendizaje y manejo de la aplicación. Se irán mostrando las diferentes pantallas obtenidas en la ejecución de la aplicación.

A.1. ABRIR LA APLICACIÓN

Haga clic en el programa Vitriuis a través del menú inicio en todos los programas seleccionas la opción Vitriuis y haces clic en Vitriuis.exe.

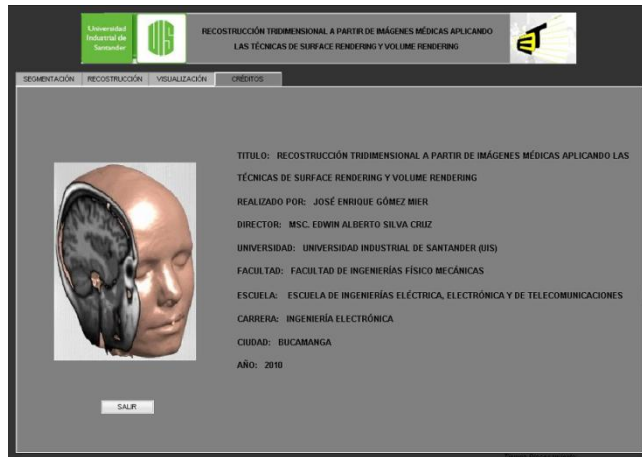
Aparecerá la siguiente imagen en la cual se comienza a ejecutar la aplicación, esta ventana es un defecto producido al independizar la aplicación de la plataforma de Matlab y no fue posible eliminarla.



La ventana desaparece y espere un momento y aparecerá la siguiente ventana la cual es una presentación de inicio de la aplicación, mientras se carga la interfaz de la aplicación.



Espere un momento y desaparecerá la ventana y aparecerá la siguiente ventana en la cual se visualiza una interfaz, la cual podemos observar a continuación.



En esta interfaz podemos observar cuatro pestañas las cuales son:

- Pestaña Segmentación.
- Pestaña Visualización Tridimensional.
- Pestaña Visualización Bidimensional.
- Pestaña Créditos.

Cada una de estas pestañas es una interfaz diferente, ya que cada una de estas interfaces son herramientas que funcionan de forma independiente. Cada vez que cambia de una pestaña a otra la interfaz de la pestaña anterior es reseteada. Todo lo visualizado o generado en esa interfaz es borrado por lo cual es recomendable guardar las imágenes antes de dar clic en otra pestaña diferente. Esta aplicación solo se puede cerrar desde el botón salir el cual se encuentra en cada una de la interfaces.

A.2. PESTAÑA DE SEGMENTACIÓN

En esta interfaz sólo hay dos botones habilitados el de salir y el de abrir los cuales nos permite cargar la imagen o imágenes a segmentar. A continuación los mencionamos.

A.2.1. CONTROLES

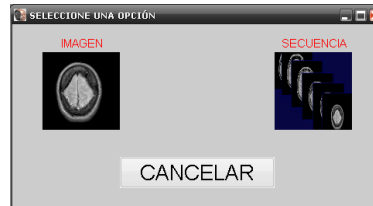


Esta sección de controles se encuentran ubicados en la parte inferior izquierda. Está dividido en 3 secciones las cuales son:

- Abrir.
- Guardar.
- Salir

b. ABRIR

Al pulsar el botón abrir nos aparecerá la siguiente ventana.

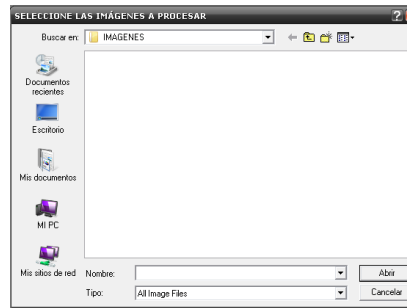


En la cual nos aparece tres opciones para escoger, las cuales son:

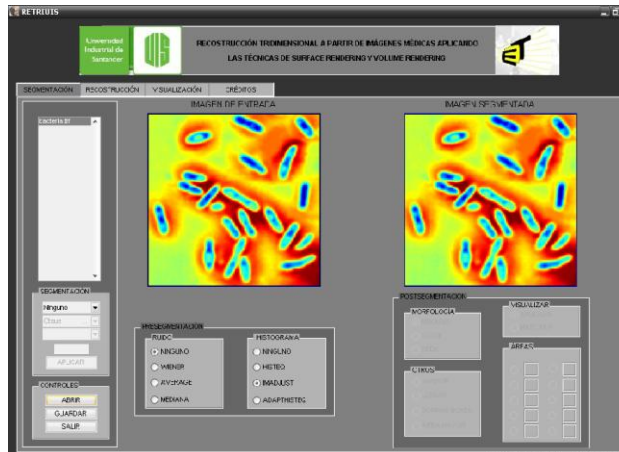
- Imagen.
- Secuencia.
- Cancelar.

1. IMÁGEN

Si da clic en Imagen se abrirá la siguiente ventana. También se resetea la interfaz de segmentación, borrándose todas las imágenes visualizadas.



En esta ventana tiene que seleccionar una imagen para cargar en la interfaz, para luego segmentar por alguno de los métodos, puede cambiar de directorio para buscar la imagen, se debe dar clic sobre la imagen y luego dar clic en abrir. Espere a que se llene la barra y volverá a la interfaz de segmentación, la cual se visualizará de la siguiente manera.



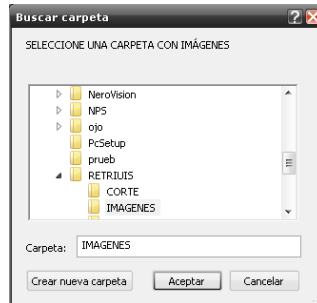
Obsérvese que en la parte izquierda y arriba de la interfaz de segmentación aparece un cuadro el cual es una lista donde aparecerán el nombre de todas la imágenes, en este caso como sólo es una imagen, el cuadro está inactivo y solo aparece el nombre de una imagen, el cual es el de la imagen que estamos trabajando. En esta interfaz se pueden observar 2 imágenes que son desplegadas, la imagen de la izquierda es la imagen de entrada y la imagen de la derecha es la imagen segmentada.

La interfaz de segmentación está dividida en 3 partes:

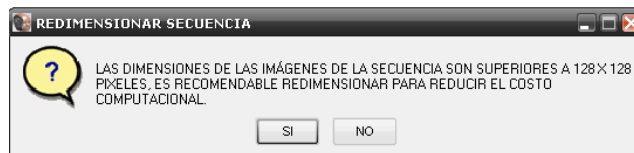
- Presegmentación.
- Segmentación.
- Postsegmentación.

2. SECUENCIA

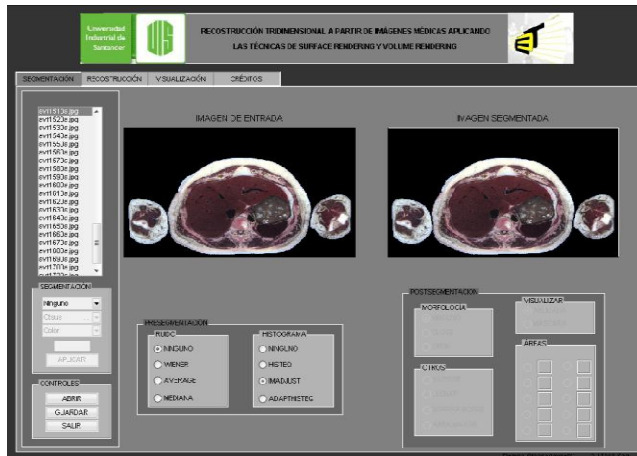
Si da clic en Imagen se abrirá la siguiente ventana. También se resetea la interfaz de segmentación, borrándose todas las imágenes visualizadas.



En esta ventana se tiene que seleccionar una carpeta que contenga una secuencia de imágenes para cargar en la interfaz, para luego segmentar por alguno de los métodos de segmentación. Puede cambiar de directorio para buscar la carpeta; se debe dar clic sobre ella y luego dar en aceptar. Si la carpeta no contiene imágenes aparecerá una ventana de error. También aparecerá alguna ventana de error si la carpeta contiene imágenes que no tienen la misma extensión, Si la carpeta contiene imágenes que no tienen el mismo nombre y Si la carpeta contiene imágenes que no tienen el mismo tamaño. Además Si el tamaño de las imágenes es mayor a 128 x128 pixeles. Saldrá la siguiente ventana.



En la ventana vista se informa que las imágenes son superiores a 128 x 128 pixels y se le recomienda redimensionar las imágenes para reducir costo computacional, existe dos opciones: sí y no. Si pulsas si, se redimensionan las imágenes a un tamaño de 128 x 128 pixels. Si se pulsa no, no se redimensionan. Espere a que se llene la barra y volverá a la interfaz de segmentación, la cual se visualizara de la siguiente manera.



Obsérvese que en la parte izquierda y arriba de la interfaz de segmentación aparece un cuadro el cual es una lista donde aparecerá el nombre de todas las imágenes. En este caso el cuadro está activo y aparece el nombre de todas las imágenes que han sido cargadas. Seleccionando con el mouse alguno de los nombres o moviendo las teclas de flecha hacia arriba y hacia abajo, podemos cambiar la imagen que estemos visualizando.

En esta interfaz se pueden observar 2 imágenes que son desplegadas, la imagen de la izquierda es la imagen de entrada y la imagen de la derecha es la imagen segmentada.

La interfaz de segmentación está dividida en 3 partes:

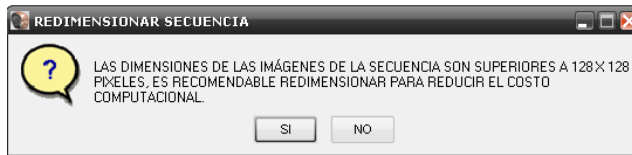
- Presegmentación.
- Segmentación.
- Postsegmentación.

3. CANCELAR

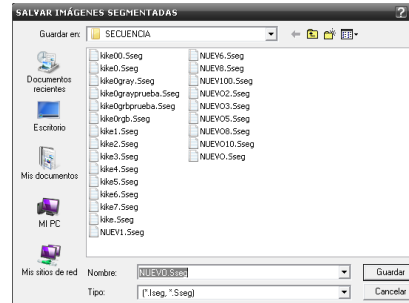
Si pulsamos el botón cancelar no se cargará ninguna imagen y la interfaz quedará como estaba antes de pulsar el botón abrir.

c. GUARDAR

Si se pulsa el botón guarda, si las imágenes son mayores a 128 x 128 píxeles y se decidió no redimensionarlas, saldrá la siguiente ventana. Si la imagen no es mayor a 128 x 128, aparecerá la ventana siguiente a esta.



Después aparece la siguiente ventana.

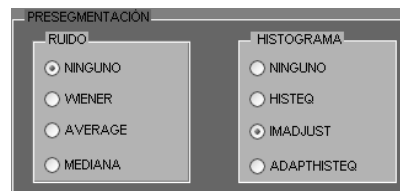


En esta ventana se puede escoger el directorio y el nombre con el cual será almacenada la imagen o secuencia de imágenes segmentada, para el caso de una imagen, será almacenada con la extensión .Iseg y para el caso de una secuencia será almacenada con la extensión .Sseg.

d. SALIR

Si se pulsa el botón salir se cerrara la aplicación.

A.2.2. PRESEGMENTACIÓN



Los controles para la presegmentación se encuentran ubicados en la parte inferior izquierda debajo de la imagen de entrada. Cualquier cambio en estos controles será ejecutado inmediatamente en la imagen de entrada. Para que surtan efecto en la imagen final se deberá oprimir el botón aplicar. Está dividido en 2 secciones las cuales son:

- Ruido.
- Histograma.

a. RUIDO

Esta sección nos permite utilizar algunos filtros para reducir el ruido en las imágenes de entrada, sólo se permite seleccionar uno a la vez. Los filtros que se pueden utilizar son:

- Ninguno.
- Wiener.
- Promedio.
- Mediana.

b. HISTOGRAMA

Esta sección nos permite utilizar algunos métodos de modificación del histograma, para mejorar el contraste y ecualizar el Histograma en las imágenes de entrada. Como en el caso anterior solo se permite seleccionar uno a la vez. Los métodos de modificación del histograma, que se implementaron son:

- Ninguno.
- Histeq.
- Imadjust.
- Adaphisteq.

A.2.3. SEGMENTACIÓN



Los controles de la segmentación se encuentran ubicados en la parte izquierda debajo del cuadro de lista de imágenes. Está compuesto por 6 parámetros:

- Métodos de Segmentación.
- Métodos de Umbralización.
- Tipo de Imagen.
- Tipo de Puntos de Inicio.
- Numero de Aéreas.
- Aplicar.

La variación de la mayoría de los parámetros no será ejecutada de manera inmediata, solo será ejecutada cuando el usuario pulse el botón aplicar.

a. MÉTODOS DE SEGMENTACIÓN

En este parámetro se selecciona el tipo de segmentación a utilizar. Este parámetro influye sobre los demás ya que según el tipo de segmentación hay parámetros que cambian o no son utilizados; por lo tanto este parámetro queda inactivo. Los tipos de segmentación disponibles son:

- Ninguno (no se utiliza ningún otro parámetro).
- Umbralización (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Sobel (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Prewitt (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Roberts (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Log (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- zerocross (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Canny (se utiliza los parámetros de métodos de umbralización, de tipo de imágenes y aplicar).
- Butterworth (se utiliza el parámetro aplicar).
- Gaussianhigh (se utiliza el parámetro aplicar).
- Idealhigh (se utiliza el parámetro aplicar).
- Watershed (se utiliza los parámetro número de aéreas y aplicar).
- Fcm (se utiliza los parámetro número de aéreas y aplicar).

- Kmeans (se utiliza los parámetro número de aéreas, aplicar y tipo de puntos de inicio, este ultimo cambia de posición con el parámetro tipo de imágenes).

b. MÉTODOS DE UMBRALIZACIÓN

En este parámetro se selecciona el tipo de umbralización a utilizar. Este parámetro solo se utiliza para el método de segmentación por umbralización y para los métodos de detención de bordes. Los tipos de umbralización disponibles son:

- Otsu.
- Entropía.
- Isodata.
- Conservación del momento.
- Análisis de concavidad del histograma.
- Selección iterativa.
- Método manual.

1. MÉTODO DE UMBRALIZACIÓN MANUAL

Cuando se selecciona esta opción aparecerá la siguiente ventana. Todos los cambios que se realicen en los métodos de segmentación que utilicen umbral serán aplicados automáticamente y por consiguiente también aparecerá esta imagen, mientras esté el umbral en modo manual.

Espere a que se llene la barra y se desplegará la siguiente ventana. Esta pequeña ventana permite al usuario cambiar el valor de umbral para la imagen seleccionada, la cual se visualiza a mano izquierda en imagen de entrada.



En la ventana anterior se puede visualizar el histograma de la imagen seleccionada y una línea vertical, la cual representa al valor de umbral de esta imagen. Este valor se puede modificar haciendo clic sostenido sobre la línea vertical y moviéndola de forma horizontal hasta el valor deseado y

automáticamente se puede observar el cambio en la imagen segmentada. La única forma de cerrar la ventana de umbral es cambiando el método de umbralización por un método distinto a umbral manual.

c. TIPO DE IMÁGEN

Este parámetro sólo se utiliza para la segmentación de umbralización y las segmentaciones de detención de bordes, además solo es aplicable a imágenes de color. Este parámetro tiene los siguientes valores:

- Color.
- Gris.

Este parámetro nos permite cambiar la forma como se calcula los umbrales, si está en color los umbrales se calculan sobre la imagen RGB y si esta en gris el umbral se calcula sobre la imagen en escala de grises.

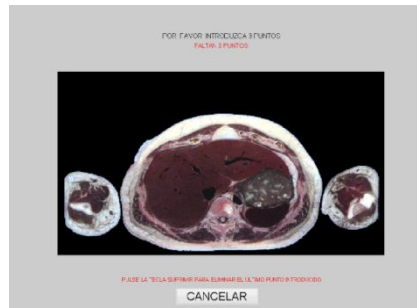
d. TIPO DE PUNTOS DE INICIO

Este parámetro se utiliza solamente para la segmentación de Kmeans, además utiliza la misma ubicación que el parámetro de tipo de imagen. Este parámetro tiene los valores siguientes:

- Automático.
- Manual.

Este parámetro nos permite cambiar la forma como se eligen los punto iniciales para la segmentación de Kmeans. Si este parámetro está en Automático, los parámetros se eligen de forma aleatoria teniendo en cuenta que no se repitan los valores de intensidad y que este valor sea diferente a cero. Si el parámetro es igual a manual el usuario tiene la posibilidad de introducir los puntos por medio de una interfaz gráfica.

Si es pulsado el botón aplicar, cuando este parámetro está en manual, aparecerá la siguiente ventana.



En esta interfaz se introducirán el número de puntos que estén en el parámetro de numero de regiones, cuando se dé clic sobre la imagen aparecerá un asterisco en ese punto indicado que ya se seleccionó este punto. En la parte superior de la imagen aparece en rojo la cantidad de puntos que restan por ingresar. Si se desea quitar el punto anterior elegido, se pulsa la tecla suprimir y si no desea continuar simplemente se pulsa el botón cancelar, lo cual cerrara la ventana y la interfaz de segmentación quedará igual que antes de pulsar el botón aplicar.

e. NUMERO DE AÉREAS

Este parámetro se utiliza para las siguientes segmentaciones: Watershed, Fcm y Kmeans. Este parámetro indica la cantidad de regiones en la que será dividida la imagen. Este parámetro se introduce por medio del teclado.

f. APLICAR

Los cambios serán ejecutados. Después de pulsar aplicar, este botón quedará inactivo hasta que se haga un cambio en los parámetros.

A.2.4. POSTSEGMENTACIÓN



Los controles para la postsegmentación se encuentran ubicados en la parte inferior derecha debajo de la imagen segmentada. Cualquier cambio en estos controles será ejecutado inmediatamente. Está dividido en 4 secciones las cuales son:

- Morfología.
- Visualizar.
- Otros.
- Áreas.

a. MORFOLOGÍA

Esta sección nos permite utilizar algunas transformaciones de morfología para mejorar las imágenes resultantes de la segmentación. Las transformaciones morfológicas que se utilizan son las siguientes:

- Ninguno.
- Close.
- Open.

b. VISUALIZAR

Esta sección nos permite cambiar la forma como se visualiza la imagen segmentada. Las formas de visualizar las imágenes segmentadas son las siguientes:

- Aplicada.
- Mascara.

Si la opción seleccionada es aplicada se visualizará la imagen final, la cual es el resultado de aplicar la máscara a la imagen de entrada. Si es seleccionada la opción máscara se visualizará la imagen resultante de la segmentación.

OTROS

Esta sección nos permite utilizar algunas operaciones básicas para eliminar algunos imperfeitos en las imágenes resultantes de la segmentación. Las operaciones básicas que se utilizan son las siguientes:

- Invertir.
- Llenar.
- Borrar bordes.
- Área mayor.

c. ÁREAS

Esta sección nos permite escoger cuales regiones de la segmentación se pueden visualizar, según su color representativo. Al principio todas las regiones son visualizadas, el usuario puede desmarcar en el color representativo de la región que se quiere quitar y de inmediato desaparecerá de la imagen segmentada. Si el usuario quiere volver a visualizar la región que eliminó simplemente tiene que volver a marcar en el color representativo de la región

A.3. PESTAÑA DE VISUALIZACIÓN TRIDIMENSIONAL

En esta interfaz solo hay dos botones habilitados el de salir y el de abrir el cual nos permite cargar la secuencia de imágenes. A continuación se habla de ellos.

A.3.1. CONTROLES

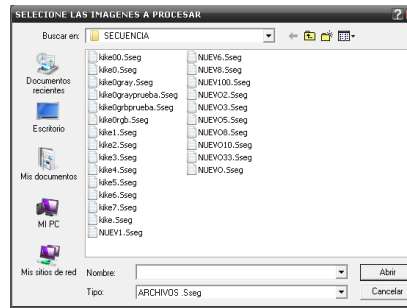


Esta sección de controles básicos se encuentra ubicada en la parte inferior izquierda. Está dividido en 3 secciones las cuales son:

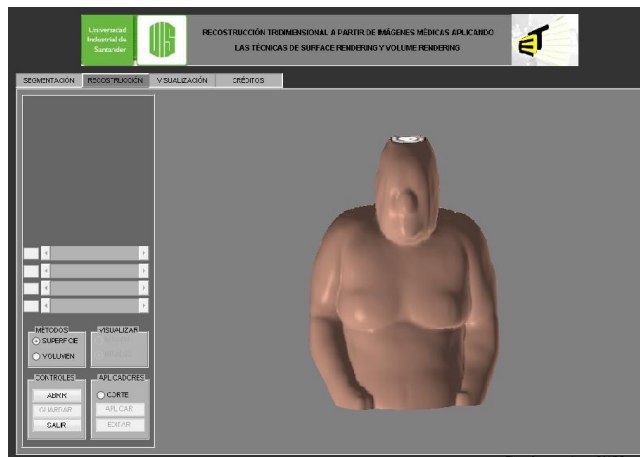
- Abrir.
- Guardar.
- Salir

a. ABRIR

Al pulsar el botón abrir nos aparecerá la siguiente ventana.



En esta ventana se puede escoger el directorio y la secuencia de imágenes con extensión “Sseg” a la cual se le genera su Visualización tridimensional. Cuando se escoge una de las secuencias la interfaz es restreada. volverá a la interfaz de Visualización tridimensional, la cual se visualizara de la siguiente manera.



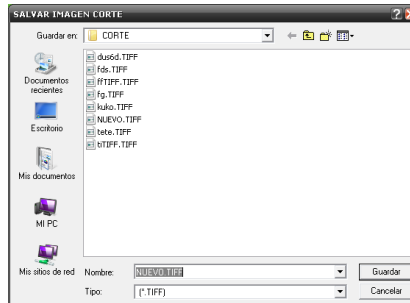
Obsérvese que en la parte derecha aparece la visualización tridimensional, esta reconstrucción es generada por el método de *surface rendering*, ya que este es el método por defecto de esta interfaz.

En la parte izquierda aparecen todos los controles de esta interfaz, los cuales están compuestos por 4 secciones, las cuales son:

- Métodos.
- Aplicadores.
- Rotación.
- Visualizar.

b. GUARDAR

Este botón permanece inactivo, solo es activado si se ha generado un corte y el control de visualizar se encuentra en la opción imagen. Si se pulsa este botón se abrirá la siguiente ventana.

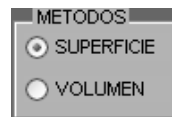


En esta ventana se puede escoger el directorio y el nombre con el cual será almacenada la imagen del corte; esta imagen solo se puede guardar en formato TIFF. Para guardar esta imagen hay que oprimir el botón guardar.

c. SALIR

Si se pulsa el botón salir se cerrara la aplicación.

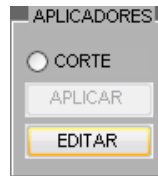
A.3.2. MÉTODOS



El control de método se encuentra ubicado en la parte izquierda debajo del control de rotación y permiten escoger por cual método se realizara la visualización tridimensional. Cualquier cambio en estos controles no serán ejecutados hasta que se oprima el botón aplicar, el cual se encuentra en el control de aplicadores. Está dividido en 2 secciones las cuales son:

- Superficie (*surface rendering*).
- Volumen (*volume rendering*).

A.3.3. APLICADORES

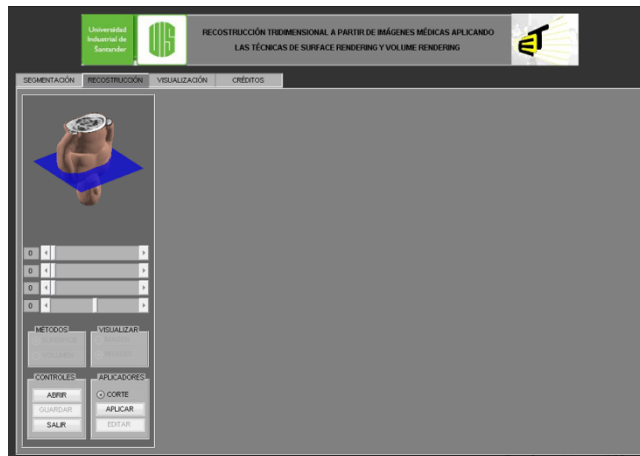


El control de aplicadores se encuentra ubicado a la izquierda de controles y debajo de visualización. Este control está dividido en 3 secciones las cuales son:

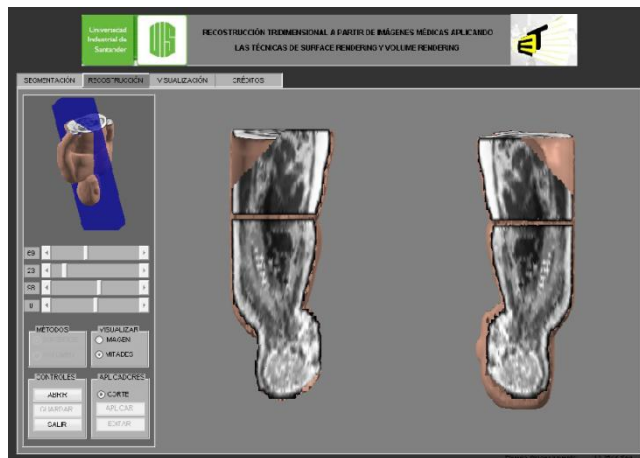
- Corte.
- Aplicar.
- Editar.

a. CORTE

Cuando se marca la opción corte, la interfaz de visualización tridimensional sufre algunos cambios como se puede ver en la siguiente imagen.



Obsérvese que la imagen de la derecha en la que estaba la visualización tridimensional desaparece y esta visualización aparece en una imagen, ubicada encima del control de rotación. También aparece un plano en color azul en la mitad de la visualización tridimensional con respecto al eje z, este plano es el plano de. Además, el control de rotación es activado, este control de rotación es el que nos permite rotar y trasladar el plano, pero es necesario oprimir el botón aplicar para visualizar la dos mitades generadas, como se ve en la siguiente imagen.



Si la opción corte es desmarcada volvemos la interfaz a su forma original en la cual se observa la visualización tridimensional completa en una imagen a la derecha, los controles se vuelven inactivos y la imagen que contiene la visualización con el plano de corte que se encontraba encima de el control de rotación desaparece.

b. APLICAR

En esta interfaz, el botón aplicar realiza cambio de dos tipos distinto según el estado que tenga la opción corte. Si la opción corte no está marcada, o sea, está inactiva, si pulsamos el botón aplicar en este caso produciremos una visualización dependiendo del tipo de visualización tridimensional que este seleccionada y siempre y cuando esta sea diferente a la que este visualizada en ese momento.

La otra posibilidad es, si la opción corte está marcada, o sea, está activa, si pulsamos el botón aplicar en este caso produciremos dos visualización tridimensional una para cada mitad del corte. Este corte se realiza según la posición que tenga el plano de corte en ese momento. Si la posición del plano de corte varia será posible realizar otra vez el corte.

c. EDITAR

Para este caso existe dos interfaces diferente, una para cada método de visualización.

- **Surface Rendering**

Si se pulsa el botón editar y esta seleccionada la opción superficie del control de métodos. Se abrirá la siguiente ventana.

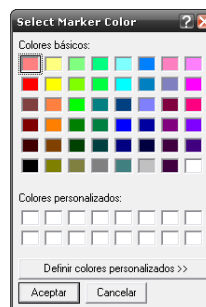


Esta pequeña interfaz de herramientas nos permite modificar el color de la superficie y el mapa de color de las tapas de visualización tridimensional para el método de *surface rendering*. Esta interfaz está compuesta por 2 controles los cuales son:

- Color.
- Mapa de color.

1. COLOR

El control de color se encuentra ubicado en la parte superior de la interfaz, si se pulsa el botón de selección (el cual está ubicado a la derecha) se abrirá la siguiente ventana.



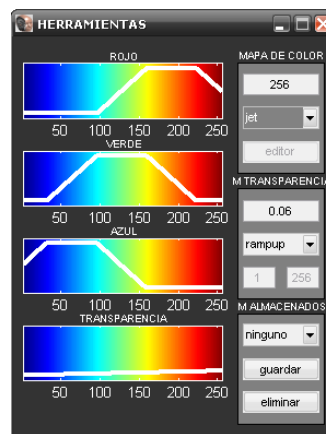
Si se selecciona alguno de los colores de esta ventana y luego se oprime el botón aceptar, a la superficie visualizada se le asignará el color seleccionado.

2. MAPA DE COLOR

Esta variable nos permite cambiar el mapa de color, en este caso afectaría las tapas de la superficie de visualización. Estos mapas son predeterminados que incluye Matlab, el mapa de color se puede visualizar en la parte izquierda del control.

- **Surface Rendering**

Si se pulsa el botón editar se abre la siguiente ventana.



Esta pequeña interfaz de herramientas nos permite modificar el mapa de color y el mapa de transparencia de la visualización tridimensional para el método de *volume rendering*. Esta interfaz está compuesta por 4 graficas y tres controles.

Las tres primeras graficas pertenecen al mapa de color, una para cada componente del formato de RGB, distribuidas de la siguiente manera: la primera grafica corresponde a los componentes del color rojo, la segunda a los componentes del color verde y la tercera corresponde a los componentes de color azul y la cuarta grafica corresponde a los componentes del mapa de transparencia.

Los tres controles de esta interfaz son:

- Mapa de color.
- Mapa de transparencia.
- Mapas Almacenados.

3. MAPA DE COLOR



El control de mapa de color se encuentra ubicado arriba a la derecha. Este control está dividido en 3 secciones las cuales son:

- Magnitud.
- Mapa.
- Editor.

a) MAGNITUD

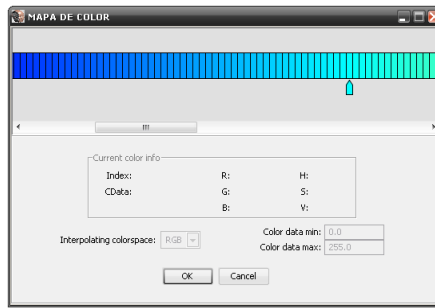
Esta variable nos permite cambiar el número de elementos que componen los mapas. Este valor por defecto es de 256. El cambio de este valor afectará la visualización tridimensional y también afectará el rango de las gráficas.

b) MAPA

La opción de mapa manual, permite al usuario manipular el mapa de color a su gusto. Para esto hay dos opciones, la primera es por medio del mouse colocándolo sobre la línea de alguna de las graficas de los planos de color y dando un clic sostenido; se podrá mover el mouse de forma vertical para modificar el valor de ese pixel en el plano. Esto afectará el color en ese valor de pixel, lo que podremos observar de forma inmediata sobre el fondo de la grafica, ya que el color en ese punto cambiará. La segunda opción es por medio de otra interfaz que incluye Matlab la cual podremos abrir al pulsar el botón editor el cual es activado al seleccionar la opción de mapa manual y será explicado a continuación.

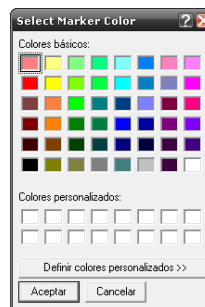
c) EDITOR

Cuando se pulsa el botón aplicar se cierra la ventana de herramientas y aparece la siguiente imagen.



Esta interfaz la incluye Matlab, la cual permite al usuario modificar o crear un nuevo mapa de color. Esto se logra agregando un punto color al dar clic debajo de la barra para agregar un nuevo punto de color y para quitando un punto de color dando un clic sobre él y luego oprimiendo la tecla suprimir.

Los colores de los espacios entre dos puntos de color son llenados por medio de la interpolación entre los colores de estos dos puntos. El color de un punto de color se puede cambiar dando doble clic sobre él, aparecerá la siguiente imagen.



Si se selecciona alguno de los colores de esta ventana y luego se oprime el botón aceptar, el punto de color se le asignara el color seleccionado. Todas las modificaciones realizadas sobre el mapa de color serán aplicadas si se oprime el botón aceptar.

4. MAPA DE TRANSPARENCIA



El control de mapa de transparencia se encuentra ubicado debajo del control del mapa de color; cada elemento de este control es independiente de los demás. Este control está dividido en 3 secciones las cuales son:

- Magnitud.
- Mapa.
- Límites del escalón.

a) MAGNITUD

Esta variable nos permite cambiar la amplitud máxima del mapa de transparencia. Este valor por defecto es de 0.06 y posee un rango de 0 a 1. El cambio de este valor afectará la visualización tridimensional.

b) MAPA

Esta variable nos permite cambiar el mapa de transparencia. Estos mapas son mapas predeterminados que incluye Matlab.

La opción de mapa escalón permite crear un mapa de transparencia con forma de escalón, el cual tiene dos límites, uno inferior y otro superior que son modificados por el usuario. La opción de mapa manual, permite al usuario manipular el mapa de transparencia a su gusto. Esto se realiza por medio del mouse colocándolo sobre la línea de la gráfica del mapa de transparencia y dando un clic sostenido se podrá mover el mouse de forma vertical para modificar el valor de ese pixel en el mapa. Esto afectará la transparencia en ese valor de pixel.

c) LÍMITES DEL ESCALÓN

Este parámetro está compuesto por dos parámetros los cuales son los límites del mapa de transparencia de escalón, uno inferior y otro superior. Los valores de límites por defecto son uno y el valor de magnitud del mapa del color, lo cual incluye todos los pixeles. Los pixeles que estén por fuera de este rango tendrán un valor igual a cero y los que estén dentro del rango tendrán un valor igual a la magnitud.

5. MAPAS ALMACENADOS



El control de mapas almacenados se encuentra ubicado debajo del control del mapa de transparencia. Cada elemento de este control es independiente de los demás. Este control está dividido en 3 secciones las cuales son:

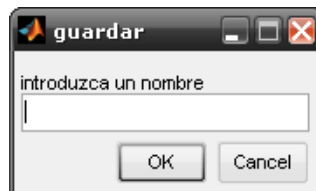
- Mapa.
- Guardar.
- Eliminar.

a) MAPA

Esta variable nos permite cambiar los mapas de color y de transparencia, estos mapas son mapas almacenados por el usuario.

b) GUARDAR

Al pulsar el botón guardar aparecerá la siguiente ventana.



Se escribe un nombre y se oprime el botón ok para almacenar los mapas que tenemos visualizados en la ventana de herramientas.

c) ELIMINAR

Al pulsar el botón eliminar aparecerá una ventana, selecciona el nombre con el cual están almacenados los mapas que queremos eliminar y pulsamos ok para eliminarlos.

A.3.4. ROTACIÓN

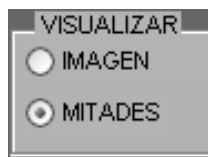


El control de rotaciones se encuentra ubicado arriba de los demás controles. Este control está dividido en 4 secciones las cuales son:

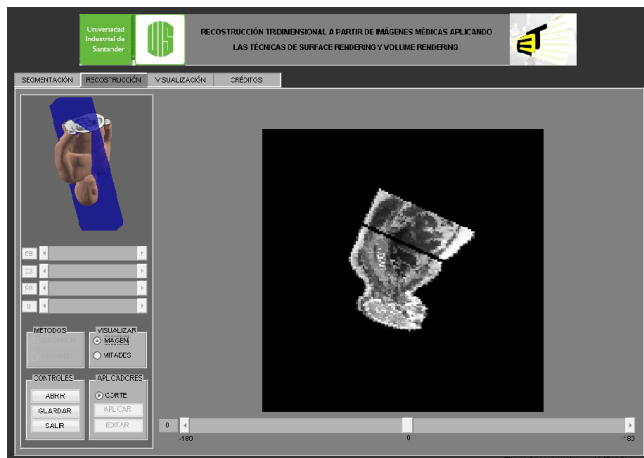
- Rotación en el eje X.
- Rotación en el eje Y.
- Rotación en el eje Z.
- Traslación por el vector normal.

La manipulación de estos parámetros permite mover el plano de corte a la posición que el usuario desee para realizar el corte de la visualización tridimensional. Los valores por defecto son cero.

A.3.5. VISUALIZAR



El control de visualizar se encuentra ubicado a la derecha del control de métodos y permiten escoger de qué forma se puede visualizar el corte. Este control permite cambiar la forma como se visualiza el corte, si está seleccionada la opción Mitades que es la opción por defecto. En esta opción se visualizará las dos mitades de la visualización tridimensional, lo cual ya se mostró en la parte de corte. Si se selecciona la opción Imagen, se visualizara la imagen bidimensional del corte, lo cual, se puede ver en la siguiente imagen.



Se puede observar en la imagen anterior la imagen bidimensional del corte. Debajo de ella hay control que no permiten rotar la imagen de 180 a -180 grados, para lograr una mejor visualización de la imagen de corte. También se puede observar, que al seleccionar la opción de Imagen se activa de inmediato el botón guardar de la parte de controles básicos.

A.3.6. NAVEGACIÓN SOBRE LA VISUALIZACIÓN TRIDIMENSIONAL

Se implementó un sistema de navegación para la interfaz de visualización tridimensional, el cual permite al usuario manipular el punto de vista de la visualización. Este sistema de navegación se puede utilizar solo para dos casos: el primero es en la visualización tridimensional inicial en la cual no se ha aplicado el corte y la segunda, es cuando se ha aplicado el corte pero se visualiza las dos mitades(se puede utilizar en cualquiera de las dos mitades, pero una a la vez). Este sistema de navegación está compuesto por 4 Movimientos los cuales son:

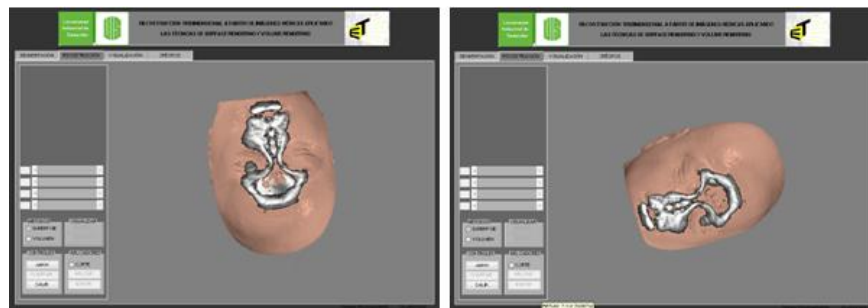
- Rotar visualización tridimensional.
- Acercar y alejarla la visualización tridimensional.
- Mover la visualización tridimensional.
- Reiniciar.

a. ROTAR LA VISUALIZACIÓN TRIDIMENSIONAL

Se puede rotar la visualización tridimensional para variar el punto de vista. Para lo cual, se implementaron dos rotaciones una alrededor del eje Z y la otra alrededor de un eje intermedio entre el eje X y el eje y. Estas rotaciones se puede realizar de dos formas distintas: una forma es por medio del mouse, dando un clic sostenido

sobre la visualización rotar; si se mueve el mouse de forma horizontal se realizara una rotación sobre el eje z y si se mueve de forma vertical se realizara una rotación sobre el eje intermedio.

La otra manera de realizar la rotación es por medio del teclado. Primero hay que seleccionar la visualización a rotar. Luego se puede rotar por medio de las flechas del teclado; la flecha hacia arriba y la de abajo realizaran una rotación sobre el eje intermedio y con las flechas hacia la derecha y la izquierda realizaran una rotación sobre el eje Z. En la siguiente imagen se puede observa un ejemplo de rotación de la visualización tridimensional.



b. ACERCAR Y ALEJARLA LA VISUALIZACIÓN TRIDIMENSIONAL

Se puede acercar y alejar de la visualización tridimensional para tener una mejor observación de un lugar específico de la visualización. Lo anterior se puede realizar de dos formas distintas: una forma es por medio del scroll del mouse; si se mueve el scroll hacia adelante se acerca a la visualización y si se mueve el scroll hacia atrás se aleja de la visualización.

La otra manera de acercar y alejar la visualizaciones por medio del teclado. Primero hay que seleccionar la visualización tridimensional. Luego se puede acercar por medio de la tecla insertar del teclado y se puede alejar con la tecla suprimir de teclado. En la siguiente imagen se puede observa un ejemplo de acercamiento de la visualización.



c. MOVER LA RECONSTRUCCIÓN

Se puede mover la visualización tridimensional para desplazarla, esto sirve cuando se ha acercado de forma considerable a la visualización y de este modo poder navegar sobre ella. Se puede mover de dos maneras: de forma vertical y de forma horizontal. Lo anterior se puede realizar por medio del teclado. Primero hay que seleccionar la visualización. Hay dos formas de mover la visualización horizontalmente: Una es a la derecha con la tecla Inicio y la otra a la izquierda con la tecla Fin. También hay dos formas de mover la visualización verticalmente: la primera es hacia arriba con la tecla Repág y la otra es hacia abajo con la tecla Avpag. En la siguiente imagen se puede observar un ejemplo de desplazamiento de la visualización tridimensional.



d. REINICIAR

Se implementó la posibilidad de reiniciar el punto de vista de la visualización tridimensional, esto se realiza pulsando la tecla Esc.

A.4. PESTAÑA DE VISUALIZACIÓN BIDIMENSIONAL

Esta interfaz está diseñada para visualizar un grupo de imágenes; en especial las imágenes generadas del corte de las visualizaciones tridimensionales.

A.4.1. CONTROLES



Esta sección de controles básicos se encuentra ubicada en la parte inferior izquierda. Está dividido en 2 secciones las cuales son:

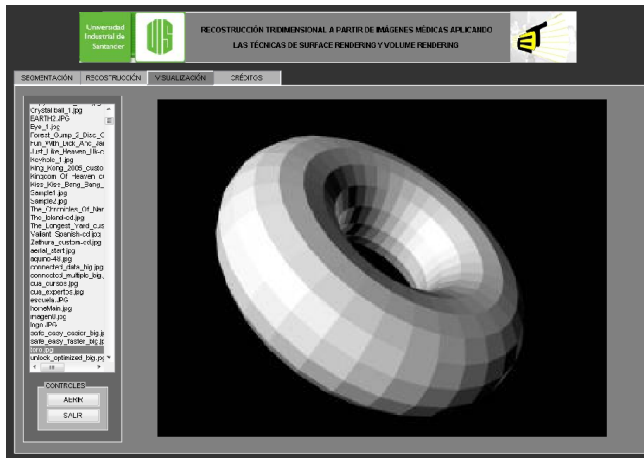
- Abrir.
- Salir

a. ABRIR

Al pulsar el botón abrir nos aparecerá la siguiente ventana.



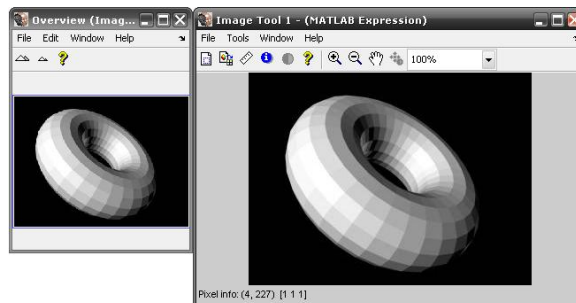
En esta ventana se tiene que seleccionar una carpeta que contenga imágenes para cargar en la interfaz. Si la carpeta no contiene imágenes aparecerá una ventana de error. La interfaz de visualización bidimensional se visualizara de la siguiente manera.



Obsérvese que en la parte superior izquierda de la interfaz de visualización bidimensional aparece un cuadro el cual es una lista donde aparecerá el nombre de todas las imágenes que han sido cargadas. Seleccionando con el mouse alguno de los nombres o moviendo las teclas de flecha hacia arriba y fecha hacia abajo, podemos cambiar la imagen que estemos visualizado.

INTERFAZ DE MEDICIÓN

Es posible abrir una interfaz que incluye Matlab la cual se llama imagetool, la cual permite calcular distancias, realizar zoom sobre la imagen y ver el componente de cada pixel de la imagen. Para abrir esta interfaz se debe hacer doble clic sobre la imagen visualizada en la interfaz de visualización.



b. SALIR

Si se pulsa el botón salir se cerrara la aplicación.