



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

MODELO DEFORMABLE DE VOLUMEN FORMADO POR DIFERENTES MATERIALES
UTILIZANDO OCTREE

TESIS
QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS (COMPUTACIÓN)

PRESENTA
ALEJANDRO TORRES JIMÉNEZ

DIRECTOR DE TESIS
DR. FERNANDO ARÁMBULA COSÍO
CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO, UNAM

CODIRECTOR DE TESIS
DR. ALFONSO GASTÉLUM STROZZI
CENTRO DE CIENCIAS APLICADAS Y DESARROLLO TECNOLÓGICO, UNAM

MÉXICO DF, AGOSTO 2015



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Gracias Dr. Fernando Arámbula, Dr. Alfonso Gastélum, Dr. Pablo Pérez,
Dr. Jesús Savage, M.I. Abel Pacheco.

Agradezco al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca otorgada.

Agradezco a la Universidad Nacional Autónoma de México,
al Posgrado en Ciencia e Ingeniería de la Computación, a Lulú, Diana, Amalia y Cecilia.

Gracias Iliana Yáñez, Cecilia Gómez, Verónica Badillo, Viridiana Torres, Andrea Baca,
Brenda Zepeda, Marisol Veloz, Fabián Torres, Zian Fanti, Brandon Pelfrey,
Jamie King, mis hermanos Margarita y Pedro.

Para Mamá y Mariel

Resumen

Este trabajo propone un modelo virtual para deformar un volumen que está constituido por dos o más materiales diferentes, con un enfoque médico, y para hacerlo rápido se utilizó la técnica del Octree.

El Octree es un proceso el cual encierra todos los puntos del volumen en un cubo dividido en ocho octantes, posteriormente si uno de los octantes contiene más de un punto, este se subdivide en otros ocho octantes, dicho proceso iterativo continúa hasta que cada octante contiene un solo punto.

En el presente trabajo, el Octree es usado para mejorar la velocidad de las búsquedas de puntos contenidos dentro una esfera, la cual contiene los puntos que serán deformados, y por tanto, todos los puntos que no están contenidos no sufren deformación, de ahí que haciendo intersecciones esfera-octante se discriminan los octantes que no colisionan con la esfera (con todos sus sub octantes), ahorrando procesamiento computacional.

Con el objetivo de brindar una aplicación funcional al proyecto, se le ha dado un enfoque médico simulando las propiedades básicas de la anatomía de una mama, que en su interior contiene un tumor, de ahí que un mismo volumen contiene dos elementos con características físicas diferentes.

Para graficar el modelo se programó un visualizador en C++ implementado con QT y OpenGL Shading Language utilizando Vertex Buffer Objects, iluminación Phong, navegación virtual con base en teclado y mouse, que además funciona para seleccionar el punto de impacto, que es de donde parte el efecto de la deformación.

Dado que se está simulando tejido humano, se ha trabajado una deformación elástica, es decir, el modelo se deforma de acuerdo al tipo de material y la fuerza aplicada, para posteriormente regresar a su estado original, y debido a que se están considerando dos materiales diferentes, el desplazamiento resultante es diferente para cada uno.

Como resultado se observa que para búsquedas de radios pequeños, el uso de Octrees es más efectivo que la búsqueda por fuerza bruta, además, el visualizador respondió correctamente y la deformación elástica pudo observarse a detalle para los diferentes materiales.

Contenido

RESUMEN.....	3
CAPÍTULO 1 INTRODUCCIÓN.....	9
1.1 DESCRIPCIÓN DEL PROBLEMA.....	9
1.2 OBJETIVO Y ALCANCE.....	13
1.3 TRABAJO PREVIO	13
1.4 ORGANIZACIÓN DE LA TESIS.....	15
CAPÍTULO 2 MARCO TEÓRICO.....	17
2.1 CÁNCER DE MAMA.....	17
2.1.1 <i>Características físicas de la mama</i>	17
2.1.2 <i>Detección temprana del Cáncer y factores de riesgo</i>	18
2.1.3 <i>Definiciones y características físicas de los objetos</i>	19
2.1.4 <i>Módulo de Young para la mama y el tumor</i>	22
2.2 GRAFICACIÓN POR COMPUTADORA	25
2.2.1 <i>Concepto</i>	25
2.2.2 <i>Frustum</i>	25
2.2.3 <i>Transformaciones Geométricas</i>	27
2.2.4 <i>Modelo de Iluminación Phong</i>	28
2.3 HARDWARE.....	31
2.3.1 <i>Graphics Processing Unit (GPU)</i>	31
2.3.2 <i>Ultrasonido</i>	33
CAPÍTULO 3 DISEÑO Y CONSTRUCCIÓN DEL MODELO BASADO EN OCTREE.....	35
3.1 OCTREE	35
3.1.1 <i>Definición y Características</i>	35
3.1.2 <i>Construcción de un Octree</i>	37
3.2 FUNCIONES Y OPERACIONES AUXILIARES.....	38
3.2.1 <i>Cálculo de un punto contenido en una esfera</i>	38
3.2.2 <i>Intersección entre dos cubos que tienen la misma inclinación</i>	39
3.2.3 <i>Búsqueda de puntos delimitados por una esfera usando un Octree</i>	40
3.2.4 <i>Cálculo de frontera para semiesfera y elipsoide</i>	41
3.2.5 <i>Puntos afectados por el impacto</i>	42
CAPÍTULO 4 VISUALIZACIÓN DEL MODELO	43
4.1 OPEN GRAPHICS LIBRARY SHADING LANGUAGE (OPENGLSL).....	43
4.1.1 <i>Concepto</i>	43
4.1.2 <i>Vertex Shader y Fragment Shader</i>	44
4.1.3 <i>Vertex Buffer Objects</i>	45

4.2	VISUALIZACIÓN	46
4.2.1	<i>Volumen de prueba</i>	46
4.2.2	<i>Navegación en la escena</i>	48
CAPÍTULO 5 ANÁLISIS DE RESULTADOS		49
5.1	VISUALIZACIÓN	49
5.1.1	<i>Preparación de Visualización</i>	49
5.1.2	<i>Entrenamiento de Visualización</i>	50
5.2	BÚSQUEDA UTILIZANDO OCTREE.....	51
5.2.1	<i>Preparación de búsqueda del Octree</i>	51
5.2.2	<i>Tiempo de búsqueda e inserción usando Octree</i>	51
5.3	DEFORMACIÓN	52
5.3.1	<i>Entrenamiento usando Octree</i>	52
5.3.2	<i>Preparación de la deformación de un objeto real</i>	54
5.3.3	<i>Resultados de la deformación elástica</i>	59
5.4	CONCLUSIONES	65
5.5	TRABAJO A FUTURO	66
BIBLIOGRAFÍA.....		67

Índice de Figuras

Figura 1 Clasificación de los Métodos Meshfree [8]	12
Figura 2 Grid de celdas, en rojo un obstáculo, abajo en azul una burbuja que subirá a la superficie, también en azul. El resto son frames de la animación de la burbuja [13].	14
Figura 3 Se muestran colisiones tanto al exterior como al interior de una malla, las esferas rojas fuera, las verdes dentro, donde también caen por espacios en la base del objeto [17].	15
Figura 4 Ejemplo de mama con elementos principales: conductos, tejido adiposo (grasa), glándulas mamarias, costillas y músculos pectorales.	17
Figura 5 Frustum. Espacio que delimita lo que se observará en pantalla.	26
Figura 6 Ejemplo de Luz Difusa.	29
Figura 7 Ejemplo de Luz Especular.	29
Figura 8 Rayo de Luz Incidente y los vectores relacionados.	30
Figura 9 Izquierda. The Evans & Sutherland Picture System. The Interactive, dynamic, 3-D line-drawing system. 1974. Derecha. Silicon Graphics Fuel Visual Workstation, información general. 2004 [39].	31
Figura 10 Comparación básica entre la estructura de una CPU y una GPU.	32
Figura 11 Izquierda. General Electric Vscan Pocket Ultrasound Version 1.2, Still frame: JPEG, Movie: MPEG-4, Voice notation: WAV, costo aproximado \$7,900.00 dólares [42]. Derecha. Sonoscape S9 Portable Color Ultrasound Doppler ISO CE FDA [43]	34
Figura 12 Muestra cómo se subdivide un Octree en octantes dentro de octantes.	35
Figura 13 Ejemplo sobre cómo se subdivide en octantes un Octree. Las esferas muestran hojas ocupadas por datos, por tanto, si hay más de un dato en un octante, ese octante se subdivide en 8, y así sucesivamente.	36
Figura 14 Cálculo simple de un punto contenido en una esfera, en azul la distancia d del punto p al centro de la esfera c , en rosa el radio r de la esfera. La comparación de estas distancias dictamina la pertenencia del punto.	39
Figura 15 Representa dos cubos con la misma inclinación, así como sus esquinas máximas (c_{max} , b_{max}) y mínimas (c_{min} , b_{min}), es decir, los puntos más orientados a dirección positiva y negativa (xyz) respectivamente.	39
Figura 17 Describe otra forma para calcular una zona de frontera para una esfera, esto es, dividiendo en segmentos la esfera y calculando esa distancia frontera por segmento. Este método puede aplicarse a otros objetos como una semiesfera donde se puede conocer su origen.	41
<i>Figura 16 Muestra una forma para calcular una zona de frontera para una semiesfera, pero también que no es posible replicarla usando el radio para una elipsoide.</i>	<i>42</i>
Figura 18 Describe cómo, para este trabajo, es necesario calcular sólo los puntos que se encuentran contenidos en la frontera, quedando discriminados los que están fuera, y además, considera sólo los que se encuentran dentro de la esfera de impacto (marcada en rojo). De ahí que los puntos que se modificarán son sólo los verdes.	42

Figura 19 Escena de entrenamiento usando triángulos predefinidos con índices, añadiendo un plano e iluminación ambiental, especular y difusa.....	43
Figura 20 Ejemplifica que el vertex shader es usado en los tres vértices del triángulo pero el fragment shader es llamado para todos los demás pixeles.....	45
Figura 21 Semiesfera con esfera contenida dentro de la primera. Ambas a base de puntos buscando simular una mama con un tumor adentro.....	46
Figura 22 Ejemplo de visualización del modelo de prueba 1 graficado en OpenGLSL con QT.	47
Figura 23 Segmentación de un volumen de mama creado con base en imágenes de ultrasonido. El software utilizado es 3D Slicer.	50
Figura 24 Modelo de volumen de mama segmentado y visualizado con el motor de visualización programado.....	50
Figura 25 Arriba, semiesfera con esfera contenida en su interior en estado inicial, sin deformación aplicada. Abajo, el mismo modelo después de aplicar una deformación simple, sólo a una sección de la parte izquierda de la semiesfera; puede observarse que del lado derecho no hay cambio, y del lado izquierdo sí.	53
Figura 26 Volumen a base de alcohol polivinílico para simular una mama, que por dentro contiene una uva para simular un tumor.	54
Figura 27 Segmentación para separar la uva, que simula un tumor, del volumen que simula una mama utilizando 3D Slicer.	55
Figura 28 A la izquierda se observa la eliminación de pixeles para refinar el detalle del modelo volumétrico segmentado, a la derecha el resultado final posterior a los procesos de segmentación..	56
Figura 29 Izquierda, volumen de la uva asemejando un tumor segmentado en 3D Slicer. Derecha, volumen con malla.	57
Figura 30 Partes utilizadas del modelo para simular la elasticidad.	59
Figura 31 Arriba, visualización del modelo sin deformación. Abajo, visualización del modelo con deformación aplicada y colores para verificar las áreas de mayor impacto.	62
Figura 32 Arriba, visualización sin deformación del interior de la mama, donde también puede observarse el tumor. Abajo, visualización del modelo con deformación aplicada donde se aprecia mayor deformación en la mama.	63
Figura 33 Arriba, visualización del modelo sin deformación con vista al interior de la mama. Abajo, visualización del modelo con deformación aplicada y colores para verificar las áreas de mayor impacto.....	64

Índice de Gráficas

Gráfica 1 Compara el desplazamiento con una fuerza aplicada a un cuerpo.....	21
Gráfica 2 Es el resultado de comparar el tiempo que tarda una búsqueda entre los algoritmos de Fuerza Bruta y el que usa el Octree al aumentar el radio de búsqueda de puntos.....	60

Índice de Tablas

Tabla 1 Organización de la tesis por conceptos.....	16
Tabla 2 Resultados del estudio Elastic Moduli of Breast Prostate Tissues Under Compression.	23
Tabla 3 Resultados del estudio Shear-wave Elastography Improves the Specificity of Breast US: The BE1 Multinational Study of 939 Masses.	24
Tabla 4 Resultados del estudio Shear-wave elastography for breast masses: local shear wave speed (m/sec) versus Young modulus.....	24
Tabla 5 Resultados del estudio Factors influencing breast elasticity measured by the ultrasound Shear Wave elastography – preliminary results.	25
Tabla 6 Muestra los resultados para pruebas de tiempo de inserción de puntos, búsqueda por fuerza bruta y búsqueda usando el Octree, para diferente cantidad de puntos y diferente radio de búsqueda.....	52
Tabla 7 Comparativa de estudios que obtienen la medida de elasticidad del tejido mamario.	58
Tabla 8 Resultados al implementar el modelo basado en Octree en comparación con el de Fuerza Bruta.	61

CAPÍTULO 1

Introducción

1.1 Descripción del problema

Un modelo deformable es una representación gráfica de un objeto que mantiene sus principales características físicas, incluyendo elasticidad, forma, densidad, entre otras, y que además podrán cambiar con respecto al tiempo y contexto ambiental. Son ampliamente utilizados en diversas áreas, principalmente en pruebas para materiales de construcción, reacciones químicas y simulaciones médicas. [\[1\]](#) [\[2\]](#) [\[3\]](#)

La simulación en tiempo real es un área que se encuentra en constante desarrollo, y uno de los motivos por el cual había sido lento es por el costo computacional que representa, debido a la gran cantidad de operaciones que se deben resolver para cada cambio de estado del objeto en estudio. Por otro lado, si se toma por ejemplo una simulación médica en tiempo real, ésta debe incluir el ambiente en el cual se realiza, así como interacción usuario-objetos, de ahí que el costo computacional es mayor, agregando además que se requiere interacción con el usuario. [\[4\]](#)

Se han trabajado constantemente técnicas para acelerar el proceso de deformación con el objetivo de lograr una simulación en tiempo real [\[5\]](#), pero hasta hoy no existe forma definitiva para hacerlo. Se han intentado diversos métodos para realizarlo, siendo dos de los más comunes el Mass-spring Model (MSM), que de manera general, consta de n puntos de masa enlazadas con sus vecinos a través de muelles libres de masa de longitud natural mayor que cero [\[6\]](#), y Finite Element Method (FEM), el cual es un método que a grandes rasgos plantea la división de un volumen en pequeños elementos interconectados por una serie de nodos, de forma tal que se pasa de un sistema de ecuaciones diferenciales a un sistema cuyo comportamiento se modela con un sistema de ecuaciones, lineales o no [\[7\]](#).

De los sistemas mencionados, el MSM suele ser desestimado debido principalmente a que, aunque es de implementación sencilla y costo computacional relativamente bajo, no se obtiene un resultado suficientemente realista en términos físicos para cuando se busca una simulación

con alto detalle; en cambio, con FEM se obtienen mejores resultados, sobre todo con geometrías complejas y propiedades materiales con soluciones complejas, pero es cierto que FEM depende del número de elementos que contenga la estructura, ya que entre más grande sea este número, mejor será el resultado, de ahí que requiere un enorme número de elementos.

En cuanto a FEM, el costo computacional es elevado y difícil de alcanzar para máquinas convencionales actuales cuando se trata de obtener resultados lo más realista posibles, incluso usando una GPU. Otro método utilizado en los últimos años es el Meshfree Method (MM, método libre de malla) [8], el cual tiene una respuesta mucho más rápida para deformaciones, ya que como su nombre lo indica, no requiere construir una malla para cada actualización ejercida sobre él, por tanto será éste el principio del cual se basa el presente estudio.

Propiedades de los Métodos Meshles [9]:

- a) Ausencia de malla. En el MM la conectividad de los nodos es determinada en tiempo de ejecución. La adaptabilidad es simple dado que sólo se requieren añadir nodos y su conectividad es prácticamente automática, realizada en tiempo de ejecución. No se requiere alineación sensible de la malla, tampoco se necesita una creación inicial. Así como, y quizá el punto más importante, no se requiere crear nuevamente una malla durante el cálculo, esto ayuda principalmente en rupturas o grietas.
- b) Continuidad de funciones de forma. Las funciones de forma para el MM pueden ser fácilmente construidas para tener cualquier orden deseado de continuidad de acuerdo al tipo de problema. Casos especiales donde la continuidad de las funciones de forma libre de malla no son convenientes en, por ejemplo, donde existen diferentes propiedades de los materiales, discontinuidades, grietas, que pueden ser manejados con ciertas técnicas.
- c) Convergencia. Para problemas del mismo orden de consistencia numérica los resultados sugieren que los MM son frecuentemente considerados mejores que los basados en malla.
- d) Costo computacional. El tiempo que toma resolver el cálculo para los MM suele ser mayor al tiempo que toma para métodos basados en malla, pero este tiempo suele ser compensado al tomar en cuenta la creación de malla de este último método, además, los MM suelen dar como resultado mejor precisión y estabilidad.
- e) Condiciones de contorno esenciales. La mayoría de los MM no presentan la propiedad delta de Kronecker, contrario a los métodos basados en malla, que frecuentemente tienen esta propiedad.

- f) Bloqueo (Locking). Los MM podrían también sufrir del fenómeno de bloqueo tal como el FEM. Este se produce debido a una imposibilidad de las funciones de forma de representar correctamente el comportamiento de todas las variables con el consiguiente aumento de la rigidez asociado a un incremento espurio de la energía de deformación asociada al corte transversal [\[10\]](#).

Beneficios al aplicar MM:

- a) Todo el tiempo invertido en el mallado del dominio del problema se reduce a la disposición, en principio arbitraria, de un conjunto de nodos que no requieren de una relación entre ellos. Permite reducir ampliamente el tiempo de ejecución.
- b) Al eliminar los elementos, la información es ahora referida a los nodos, de modo que no hay discontinuidad de las variables derivadas, como los esfuerzos, por ejemplo.
- c) Al no estar relacionados unos nodos con otros, las deformaciones pueden ser del grado que requiera el problema, sin perder precisión.
- d) Al representar el dominio del problema a través de nodos, la eliminación o creación de nodos es más flexible, permitiendo representar efectos como el crecimiento de una grieta, añadiendo puntos en el extremo de la misma, o bien, transformaciones.
- e) El remallado se convierte en una tarea relativamente sencilla en problemas 2D y 3D tanto en problemas estáticos, como dinámicos, lineales y no lineales. También pasa a ser una tarea simple y fácilmente automatizable que reduce el tiempo computacional sin empeorar drásticamente la precisión, tanto de problemas 2D como de problemas 3D.

La clasificación de los Métodos Meshless puede resumirse en el siguiente listado, y una descripción de cada uno de ellos puede verse en [\[11\]](#):

1. Smoothed Partiche Hydrodynamics (SPH)
2. Método de los Elementos Difusos (DEM)
3. Método de Galerkin Libre de Elementos (EFGM)
4. Método Meshless de Mínimos Cuadrados (LSMM)
5. Método Meshless Petrov-Galerkin Local (MLPG)
6. Ecuación de la Integral en los Contornos Locales (LBIE)
7. Métodos de Partición de Unidad
8. High Order Approximation of Scattered Data (hp Clouds)
9. Método de los Elementos Naturales (NEM)

- 10. Meshles Finite Element Method (MFEM)
- 11. Reproducing Kernel Element Method (RKEM)

En el artículo del 2004, *Classification and Overview of Meshfree Methods*, se presenta el siguiente diagrama (Figura 1) para especificar la clasificación de los Métodos Meshfree [9].

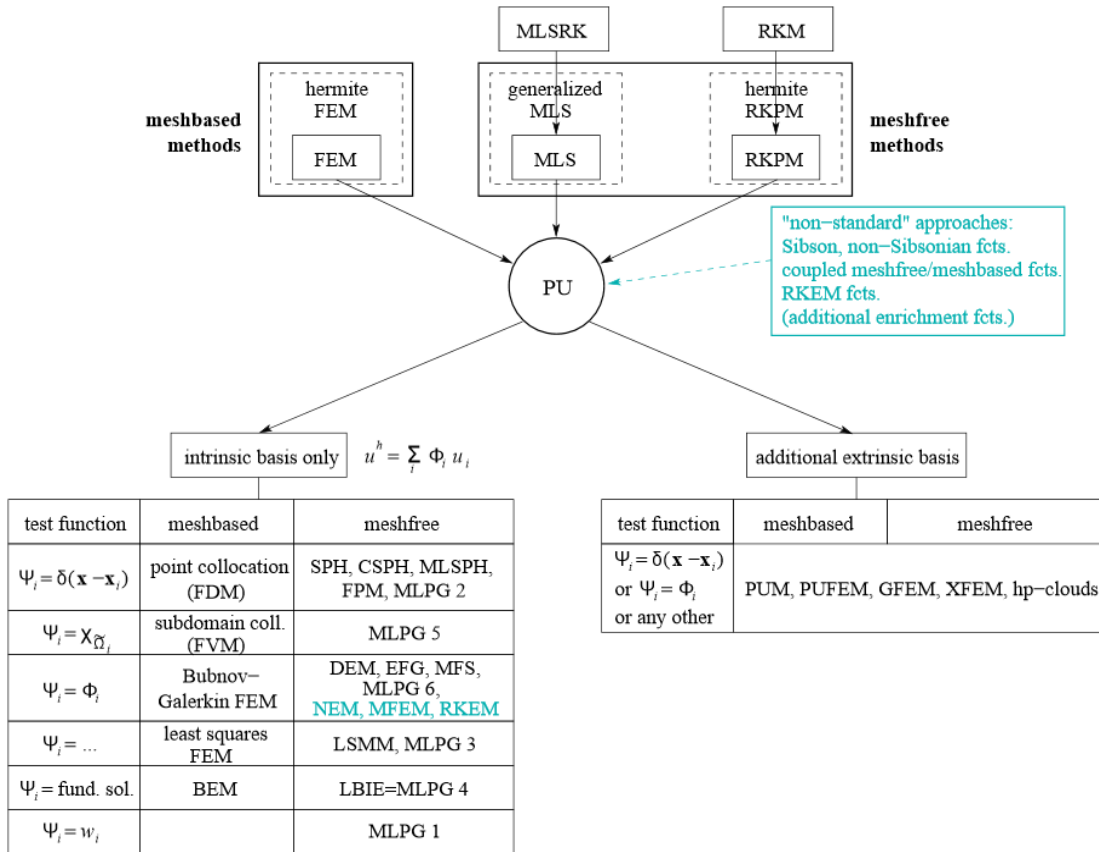


Figura 1 Clasificación de los Métodos Meshfree [9].

Los métodos que no usan malla, por su naturaleza, permiten simular con mayor facilidad problemas de geometría compleja, deformaciones grandes y detalladas, e incluso roturas irreversibles sin realizar un remallado, discontinuidades en un flujo de fluido, o singularidades especiales en ciertos materiales.

Para resolver el problema de un modelo deformable con respuesta en tiempo real, es necesario utilizar por lo menos una GPU, además de buscar métodos de cálculo de operaciones que den una respuesta lo más rápida posible, dado que en una deformación física interviene una gran cantidad de cálculos que se deben ejecutar al mismo tiempo, y esperando una respuesta prácticamente inmediata. El problema aumenta cuando se trabaja con más de un material de ciertas características físicas diferentes entre ellos.

1.2 Objetivo y alcance

El objetivo es proponer un modelo virtual de deformación de volúmenes de respuesta rápida que se pueda aplicar a reconstrucciones de objetos reales formados por materiales diferentes, conservando sus características físicas, y dando un enfoque en su aplicación a la detección de cáncer de mama.

Para cumplir el objetivo se trabajará una deformación sobre la superficie de un volumen utilizando un modelo de búsqueda de puntos usando un Octree aplicado a dicho volumen, la deformación también estará limitada por una profundidad determinada, así como por un radio de impacto. El volumen contendrá dos materiales de diferentes características físicas.

El uso del Octree tiene como función principal agilizar la búsqueda de puntos, mientras que la GPU permitirá para dar velocidad a la visualización, utilizando Shaders de OpenGLSL 4.0.0 para mejorar el desempeño, y Vertex Buffer Objects, todo programado con el lenguaje C++ versión 11.

Además, se creará un escenario virtual de simulación que incluye dos volúmenes: el primero es una semiesfera, simulando ser una mama, y el otro volumen simula ser un tumor dentro de la mama, ambos con características físicas diferentes. Posteriormente, el mismo escenario se usará para graficar un volumen real obtenido a base de imágenes de ultrasonido.

Se espera que en el futuro este modelo sirva como base para ser implementado en otros métodos de deformación meshless, o bien para otros proyectos de visualización.

Este trabajo propone un sistema básico de simulación para deformaciones plásticas y elásticas simples, con respuesta lo más cercana posible al tiempo real, que posteriormente pueda re utilizarse para simular deformaciones reales.

1.3 Trabajo Previo

Existe una gran cantidad de trabajo utilizando el método de Octree como herramienta para resolver problemas de diferentes ámbitos, a continuación se enlistan sólo algunos ejemplos:

- Elseberg et. al. [\[12\]](#) trabajaron un modelo de Octree que guarda y comprime decenas de millones de datos en 3D sin pérdida de precisión. Dicho modelo fue aplicado en la visualización de nubes de puntos, detección de formas, y exploración de coincidencias en 3D.

- Guo y Qin [13] muestran una deformación en tiempo real con un volumen meshless utilizando también una estructura de Octree, y simulación física basada en funciones tipo Meshless Moving Least Squares (MLS).
- Ďurikovič y Kupka [14] presentan una simulación de fluido basada en estructura de octree, que usa técnicas de malla de refinamiento para lograr detalle más alto y resolver las ecuaciones de Navier Stokes para fluidos múltiples, además muestra una solución para objetos flotantes y obstáculos inmóviles.

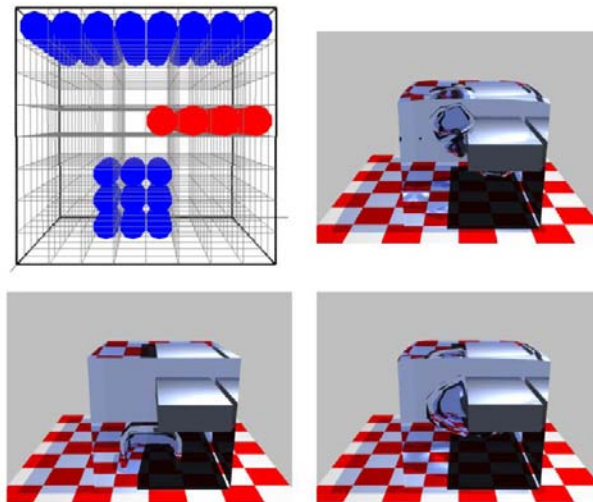


Figura 2 Grid de celdas, en rojo un obstáculo, abajo en azul una burbuja que subirá a la superficie, también en azul. El resto son frames de la animación de la burbuja [14].

- Wu y Hori [15] trabajaron un modelo de planeación de trayectoria para manipular un robot con detección de colisiones, basado en un modelo de Octree, usado para expresar el entorno, y la misma detección de colisiones, su rapidez le permite actuar como planeador de trayectoria en tiempo real.
- Song et. al. [16] trabajaron un método de representación visual de un enorme conjunto de datos médicos en una PC estándar, utilizando una compresión jerárquica basada en una estructura de Octree con el objetivo de lograr su fácil manejo.
- Yau et. al. [17] proponen la base para un sistema virtual de entrenamiento basado en Octree, navegación y detección de colisiones, el cual ha sido utilizado en prácticas de intervención dental, integrando un dispositivo médico háptico.

- WenShan et. al. [18] proponen un método de sustitución basado en Octree para colisión de partículas con superficies (mallas) utilizando una GPU. En esencia, cada nodo hoja del Octree se asocia con una normal correspondiente a las normales de los triángulos que intersecan el nodo. El modelo permite la colisión con ambos lados de la malla y está preparado para trabajar con millones de partículas.

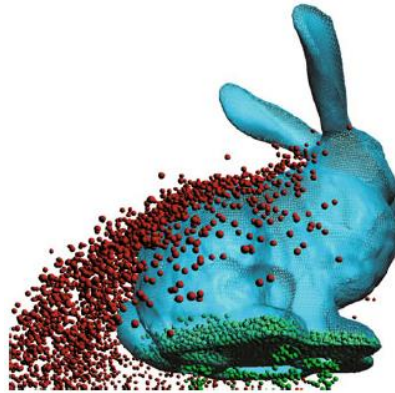


Figura 3 Se muestran colisiones tanto al exterior como al interior de una malla, las esferas rojas fuera, las verdes dentro, donde también caen por espacios en la base del objeto [18].

1.4 Organización de la tesis

La organización de la tesis sigue el orden en que se ha desarrollado el objetivo, intentando explicar paso a paso la realización de cada tema. Se inicia abordando la teoría, posteriormente se explica la construcción del método, así como su visualización, y termina mostrando y analizando los resultados obtenidos. Se incluyen además conclusiones y bibliografía.

En el Capítulo 2 se revisa a detalle el marco teórico, iniciando con el tema del hardware y software actual, comentándose su relación con los gráficos por computadora; se analiza el escenario tecnológico en que ahora se desarrolla y, por tanto, la influencia e impacto de las tarjetas GPU; se revisa el escenario de la paralelización de operaciones y procesos, que permite trabajar con enormes cantidades de datos, y se dan las aplicaciones que tiene esta tecnología en la ciencia.

El Capítulo 3 contiene detalles sobre la creación del modelo de Octree, comentándose el diseño específico del algoritmo así como su programación, se argumenta la aplicación del Octree al objetivo de la tesis, que está relacionado con la búsqueda de puntos que forman objetos volumétricos, esto en áreas delimitadas por fronteras específicas y diferentes características, para así poderlo aplicar a un ejemplo real, que en este caso es un ejercicio simple de deformación, que además funciona para validar que el modelo cumple con su tarea.

En el Capítulo 4 se aborda de lleno el tema de la visualización del modelo utilizando GPU, se explica la implementación de los Shaders con el fin de paralelizar procesos de iluminación, se explica el uso de Vertex Buffer Objects buscando trabajar con facilidad grandes cantidades de vértices, se detalla el tema de la navegación creada para recorrer el modelo volumétrico, y se explica qué tipo de datos puede graficar esta herramienta de visualización.

El Capítulo 5 contiene los resultados generados al aplicar el modelo a un caso real, incluyendo su análisis y validación; también, se aborda el tema de cómo se utilizó un volumen de mama humana extraído de tomas de ultrasonido, así como su segmentación, visualización y su resultado final al aplicar el modelo.

Capítulo 2	Teoría General	<ol style="list-style-type: none"> 1. Software 2. Hardware 3. Graficación por Computadora 4. Iluminación Phong
	Enfoque	<ol style="list-style-type: none"> 1. Características de la mama y del cáncer de mama 2. Módulo de Young de la mama y del tumor mamario 3. Paralelización de procesos y uso de la GPU
Capítulo 3	Modelo Deformable	<ol style="list-style-type: none"> 1. Características generales del Octree 2. Búsqueda de puntos a deformar usando Octree 3. Delimitación del área deformable 4. Especificación de características del modelo
Capítulo 4	Visualización del Modelo	<ol style="list-style-type: none"> 1. Graficación con Shaders de OpenGL 2. Manejo de datos con Vertex Buffer Objects 3. Escena virtual y navegación con teclado y mouse
Capítulo 5	Resultados	<ol style="list-style-type: none"> 1. Entrenamiento de visualización y del modelo deformable 2. Creación de un volumen a base de alcohol polivinílico que asemeja una mama con un tumor 3. Escaneo del volumen con ultrasonido y obtención de imágenes para crear un volumen 4. Segmentación de la mama y del tumor del volumen 5. Obtención de las propiedades físicas de mama y tumor 6. Graficación en el visualizador del volumen creado 7. Aplicación de la deformación al volumen
	Análisis de Resultados Conclusiones	

Tabla 1 Organización de la tesis por conceptos.

CAPÍTULO 2

Marco Teórico

Se detallan los elementos conceptuales que se utilizan para la elaboración del modelo, se abordan temas técnicos de graficación, y del enfoque al que se dirige el modelo propuesto en este trabajo. Se estudian las características físicas de la mama y su relación con el cáncer, se definen conceptos como elasticidad y Módulo de Young.

El motivo de estudiar estos conceptos es porque posteriormente se aplicarán para simular una deformación con un enfoque médico, partiendo de la idea principal de que el modelo deformable puede aplicarse en el futuro para prácticas médicas, como por ejemplo, entrenamiento en toma de biopsias de mama.

2.1 Cáncer de Mama

2.1.1 Características físicas de la mama

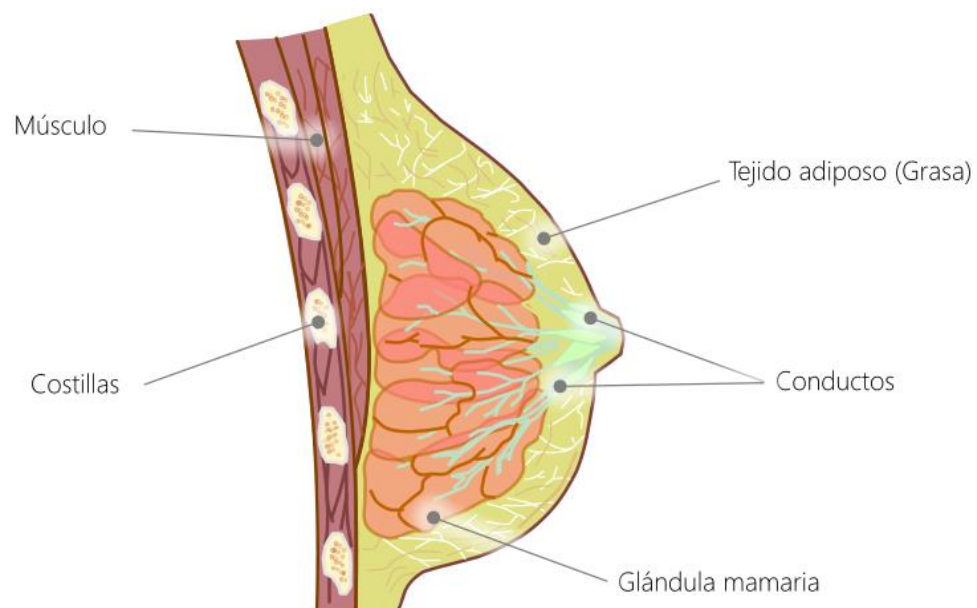


Figura 4 Ejemplo de mama con elementos principales: conductos, tejido adiposo (grasa), glándulas mamarias, costillas y músculos pectorales.

La mama físicamente podría dividirse en tejido adiposo (tejido graso) y glándulas mamarias, que a su vez contienen lóbulos, y estos también contienen lobulillos, que están formados de células productoras de leche, la cual es guiada a través de los conductos que convergen en el pezón. Otros órganos mostrados en la Figura 4 son los músculos pectorales y las costillas.

En la mama también se encuentran vasos sanguíneos y vasos linfáticos, los cuales transportan un líquido conocido como linfa, estos conductos a su vez se conectan a ganglios linfáticos, los cuales se presentan en todo el cuerpo, y su función principal es ayudar a combatir infecciones y enfermedades.

El cáncer de mama es un crecimiento de células malignas en el tejido mamario [\[19\]](#), es el cáncer más frecuente en las mujeres. Se presenta ante un crecimiento descontrolado de células anormales (malignas, cancerosas) en los conductos (cáncer ductal), o bien, donde residen las células productoras de leche dentro de los lobulillos (carcinoma lobulillar). Estos lóbulos están a su vez conectados al sistema linfático, el cual, en una etapa avanzada de cáncer, participa para diseminar las células malignas a otras regiones del cuerpo.

2.1.2 Detección temprana del Cáncer y factores de riesgo

La detección temprana de un tumor maligno se encuentra en constante desarrollo y por tanto ofrece más y mejores opciones de tratamiento, lo cual, dependiendo del avance de la enfermedad (etapa), abre la posibilidad de curación total, o bien, ofrece mejor calidad de vida para un enfermo.

Existen varios factores de riesgo que se enlistan a continuación:

- Menstruación a edad temprana [\[20\]](#).
- Edad avanzada en el momento del primer parto o no haber dado a luz nunca [\[21\]](#).
- Antecedentes personales de cáncer de mama invasivo, carcinoma ductal in situ (CDIS), carcinoma lobulillar in situ (CLIS) o de enfermedad benigna (no cancerosa) de mama [\[22\]](#).
- Antecedentes familiares (familiar de primer grado, como la madre, una hija o una hermana) con cáncer de mama [\[23\]](#).
- Cambios hereditarios en los genes BRCA1 o BRCA2 [\[24\]](#).
- Tratamiento con radioterapia dirigida a la mama o el pecho [\[25\]](#).
- Tejido de la mama que se ve denso en un mamograma [\[26\]](#).

- Uso de hormonas, como estrógeno y progesterona para tratar los síntomas de la menopausia [\[27\]](#).
- Uso previo de la hormona dietilstilbestrol (DES) durante el embarazo o ser la hija de una mujer que tomó DES durante el embarazo [\[28\]](#).
- Obesidad [\[29\]](#).
- Actividad física insuficiente [\[30\]](#).
- Consumo de bebidas alcohólicas [\[31\]](#).
- Raza blanca [\[32\]](#).

Además, los signos del cáncer de mama pueden ser los siguientes [\[19\]](#):

- Masa o engrosamiento en la mama o cerca de ella, o en el área debajo del brazo.
- Cambio en el tamaño o la forma de la mama.
- Hoyuelo o arruga en la piel de la mama.
- Pezón que se vuelve hacia adentro de la mama.
- Líquido que sale del pezón, que no es leche materna; especialmente si es sanguinolento.
- Piel con escamas, roja o hinchada en la mama, el pezón o la aréola (área oscura de piel que rodea el pezón).
- Hoyuelos en la mama parecidos a la piel de naranja.

De todo lo anterior, es comprensible y justificable la importancia de continuar investigando y desarrollando mejores técnicas que ayuden a una detección del cáncer lo más precisa posible, o bien, apoyar al desarrollo de exámenes para detección de esta enfermedad con base en la tecnología actual para obtener cada vez mejores resultados, e incluso aplicar dicha tecnología para capacitar con mejores herramientas y mayor realismo a quienes practican estos ejercicios y exámenes.

2.1.3 Definiciones y características físicas de los objetos

Esfuerzo o Tensión. Fuerza por unidad de área dentro de los materiales, que surge de las fuerzas aplicadas externamente. Es la fuerza de reacción interna en un material creado por una carga externa. F = carga total en newton (N) o kilo-newton (kN), A = área de sección transversal, generalmente en milímetro cuadrado. La unidad para tensión (stress) es pascal (Pa), y múltiplo de pascal como kPa, MPa.

$$\text{Esfuerzo} = \text{Fuerza}/\text{Área}$$

$$\sigma = F/A$$

$$1Pa = 1 N/m^2$$

$$1MPa = 1 N/mm^2$$

Elasticidad. Es la propiedad por la cual un material tiene la capacidad de regresar exactamente a su forma original con la eliminación de la fuerza aplicada. Al exceder el límite de elasticidad del material, se deformará permanentemente y no volverá a su forma original. Cuando un material tiene una carga aplicada, la carga causa que el material se deforme, la elasticidad es la habilidad del material para regresar a su forma original después de que la carga es removida.

Plasticidad. Un material plástico conservará exactamente la forma que asume bajo carga cuando esta carga es removida. Es la habilidad de un material para que se deforme permanentemente sin fracturarse o romperse. Es el opuesto de fuerza (*strength*). Por ejemplo, si un miembro de una estructura de un puente se sobrecarga, la plasticidad permite que el miembro sobrecargado circule (*flow*) permitiendo la distribución de la carga a otras partes de la estructura del puente [33]. Otro ejemplo de material plástico es la arcilla cuando está mojada.

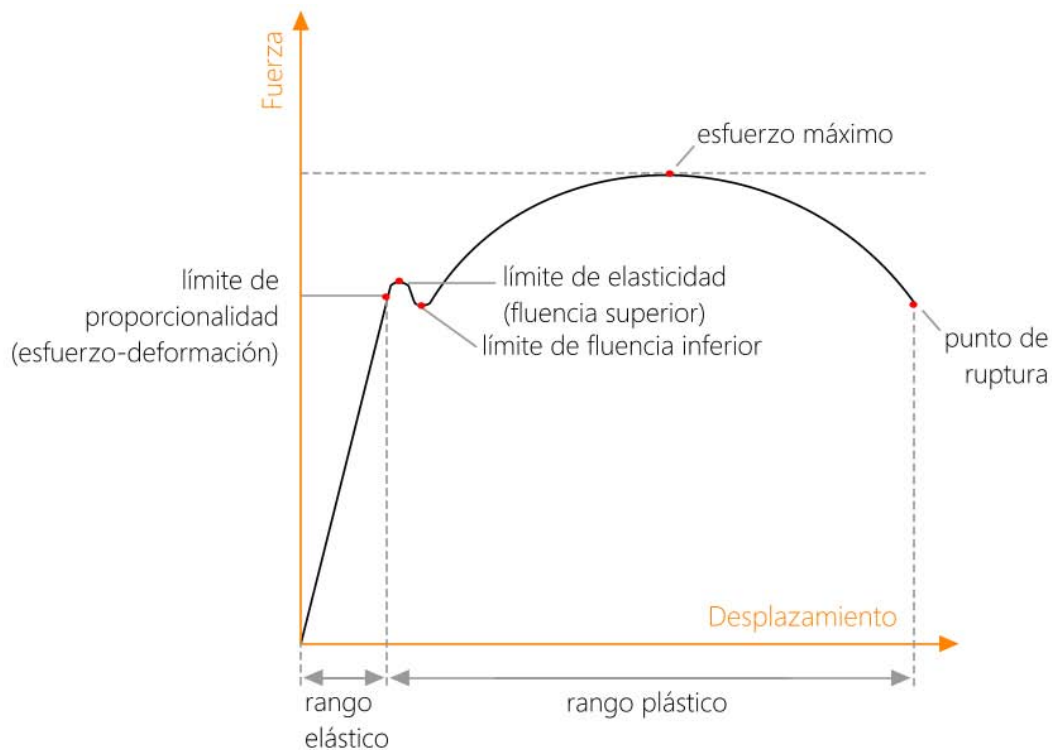
Densidad. La masa por unidad de volumen de un material es llamada densidad. Para obtener la densidad de algo, se divide la masa (cuánto pesa), por el volumen (cuánto espacio ocupa). La densidad del agua es $100\text{kg}/\text{m}^3$, otras sustancias tienen diferentes densidades, siendo más pesadas o más ligeras que el agua. Las densidades son frecuentemente expresadas como peso específico (*specific gravity*), el cual quiere decir que está expresado en proporción del peso del agua. Por lo tanto, el agua contiene 1 unidad de peso específico, mientras que las densidades (expresadas en peso específico) de otras sustancias son: Concreto 2.4; Acero 7.85; Plomo 11.35; Cobre 8.92; Aluminio 2.7; etc.

Viscosidad. La viscosidad es la oposición de un fluido a las deformaciones tangenciales, y es debida a las fuerzas de cohesión moleculares. Todos los fluidos conocidos presentan algo de viscosidad, siendo el modelo de viscosidad nula una aproximación bastante buena para ciertas aplicaciones. Un fluido que no tiene viscosidad se llama fluido ideal. Sólo se manifiesta en líquidos en movimiento, y se ha definido la viscosidad como la relación existente entre el esfuerzo cortante y el gradiente de velocidad. Esta viscosidad recibe el nombre de viscosidad absoluta o viscosidad dinámica. Generalmente se representa por la letra griega μ .

Deformación. La deformación ϵ se define como el cociente entre la variación de la longitud producida por la tensión en una dirección y su longitud antes de ser sometido a la tensión:

$$\varepsilon = \frac{\Delta l}{l_0}$$

De donde Δl es el incremento de longitud y l_0 es la longitud inicial.



Gráfica 1 Compara el desplazamiento con una fuerza aplicada a un cuerpo.

En el rango elástico que se muestra en la Gráfica 1, la tensión es lineal con la deformación y termina justo al límite de proporcionalidad; es en este rango cuando al desaparecer la tensión sobre el cuerpo, regresa a su forma original y no presenta deformación alguna. Posterior al límite de proporcionalidad y hasta el límite de elasticidad, el comportamiento ya no es lineal, el límite elástico se conoce como la máxima tensión que no produce deformación permanente al retirar la tensión. Posterior al límite de fluencia la deformación es permanente, no cede al terminar la tensión, es ahí donde se presenta el Rango plástico. Si se sigue aumentando la tensión, el cuerpo se rompe, a ese instante se le llama deformación de rotura.

Este comportamiento de la gráfica cambia en magnitud para diferentes tipos de materiales. Para materiales frágiles la rotura se produce con poca deformación plástica, por el contrario,

para materiales dúctiles la zona de deformación plástica es más amplia, por tanto se obtienen deformaciones mayores.

En la zona de rango elástico lineal se cumple la ley de Hooke, por lo que se tiene lo siguiente:

$$\sigma = E\varepsilon$$

en donde E es el módulo de Young. El módulo de Young, en el sistema internacional, se mide en N/m^2 , es decir, Pascales.

2.1.4 Módulo de Young para la mama y el tumor

El principio básico de la Elastosonografía es que la compresión del tejido produce una deformación en él, dicha deformación es menor en los tejidos duros que en los suaves, y su objetivo es medir el grado de distorsión de haz de ultrasonido en virtud de la aplicación de una fuerza externa; básicamente se hace mediante la adquisición de dos imágenes ultrasonográficas (antes y después de la compresión de tejido), y se da seguimiento al desplazamiento del tejido evaluando la propagación del haz en las imágenes. El software propio del equipo que genera las imágenes ultrasonográficas provee una medida precisa de la distorsión del tejido. Las imágenes generadas pueden mostrarse en el monitor del ecógrafo.

Fuente de vibración externa sinusoidal. El ultrasonido *Doppler* mide la amplitud de vibraciones externas mediante el principio: algo más duro vibra menos. Al realizar esta técnica, la imagen que se aprecia generalmente muestra un espectro de colores que van desde el color rojo hasta el azul, siendo el rojo el color que proyecta los tejidos blandos, el amarillo los tejidos intermedios, el verde y azul los tejidos más duros. Se obtiene resultado in vivo.

Detección de Cáncer de Próstata utilizando Sonoelastografía por ondas de arrastre. Usando dos fuentes de vibración se mide la amplitud de vibración a diferentes frecuencias. La elasticidad se mide en metros por segundo, hay una relación entre la velocidad de corte con el valor de la elasticidad en kPa. Utilizando fuerza de radiación acústica, si se utiliza el mismo ultrasonido con mayor intensidad crea una fuerza.

La información obtenida puede ser representada de tres formas:

1. Cualitativa (Imagen de Elasticidad): El ecógrafo adquiere los datos correspondientes a la anatomía tisular en condiciones basales en modo-B, luego se aplica una pequeña

presión y se adquiere otro mapa de la anatomía tisular (post-compresión o post-deformación), el resultado de la correlación computarizada de los mapas anatómicos se muestra en una escala de color, proyectado de modo semitransparente sobre la imagen en modo-B

2. Semicuantitativa: En la información cualitativa obtenida se calcula el gradiente del desplazamiento en dos zonas distintas sometidas a la misma presión, (la patológica y la sana), y se obtiene el ratio entre ellas.
3. Cuantitativa: Se cuantifica en números absolutos la dureza del tejido, proporcionando un valor en Kilopascales o en cm/seg.

A continuación se muestra un resumen de varios estudios enfocados en el tejido mamario y sus características.

Elastic Moduli of Breast and Prostate Tissues Under Compression (1999) [34]. Calcula por medio de imágenes de elastografía el módulo de elasticidad con compresión de 5% y 20% de acuerdo a la siguiente ecuación. En la Tabla 2 se muestran los resultados en kPa. Es el estudio más citado en posteriores estudios.

$$E = \frac{2(1 - \nu^2)qa}{w}$$

Tipo de tejido mamario	Masas	5% precompresión			20% precompresión		
		Frecuencia de carga (Hz)			Frecuencia de carga (Hz)		
		0.1	1	4	0.1	1	4
Graso	8	18 ± 7	19 ± 7	22 ± 12	20 ± 8	20 ± 6	24 ± 6
Glandular	31	28 ± 14	33 ± 11	35 ± 14	48 ± 15	57 ± 19	66 ± 17
Fibroso	18	96 ± 34	107 ± 31	116 ± 28	218 ± 87	232 ± 60	244 ± 85
Carcinoma ductal in situ	23	22 ± 8	25 ± 4	26 ± 5	291 ± 67	301 ± 58	307 ± 78
Carcinoma ductal invasivo e infiltrante	32	106 ± 32	93 ± 33	112 ± 43	558 ± 180	490 ± 112	460 ± 178

Tabla 2 Resultados del estudio Elastic Moduli of Breast Prostate Tissues Under Compression.

Shear-wave Elastography Improves the Specificity of Breast US: The BE1 Multinational Study of 939 Masses (2012) [35]. Calcula por medio de imágenes de elastografía shear-wave diferentes valores para distintos tipos de tejido. Usa software para calcular valores y no indica modelo. Es el paper más completo con 939 masas analizadas. En la Tabla 3 se observan los resultados principales en kPa.

Tipo de tejido mamario	Masas	Mediana Emax (kPa)	IQR	Pvalue
Quiste	104	23	Jul-53	<.001, vs fibroadenoma
Fibroadenoma	179	42	28-73	<.001, vs cualquier maligno
Otro benigno	351	44	25-90	<.001, vs cualquier maligno
Lesiones de riesgo	17	74	30-165	<.001, vs cualquier maligno
Absceso	2	133	87-180	.72, vs cualquier maligno
Carcinoma ductal in situ	12	133	137-180	.053, vs carcinoma invasivo de mama
Otro maligno	6	175	137-180	ND
Carcinoma invasivo de mama	271	179	143-180	ND
Suma	939	65	30-168	...

Tabla 3 Resultados del estudio Shear-wave Elastography Improves the Specificity of Breast US: The BE1 Multinational Study of 939 Masses.

Shear-wave elastography for breast masses: local shear wave speed (m/sec) versus Young modulus (kPa) (2014) [36]. Calcula por medio de imágenes de elastografía shear-wave el valor de elasticidad máxima para un tipo de lesión benigna y maligna. Utiliza software estadístico y no muestra modelo usado. En la Tabla 4 se muestra el detalle en kPa.

Lesiones	Emax	Emean	SD	wholeSD
Benigno	40.6 ± 19.5	34.8 ± 17.7	3.6 ± 2.0	6.7 ± 3.6
Maligno	155.1 ± 67.2	140.7 ± 58.5	14.1 ± 9.6	30.8 ± 16.1
P-value	<0.001	<0.001	<0.001	<0.001

Tabla 4 Resultados del estudio Shear-wave elastography for breast masses: local shear wave speed (m/sec) versus Young modulus.

Factors influencing breast elasticity measured by the ultrasound Shear Wave elastography – preliminary results (2010) [37]. Calcula por medio de imágenes de elastografía de onda cortante elasticidad y elasticidad heterogénea para tejido glandular y graso. Divide a la mama en cuadrantes. En la Tabla 5 se muestran los principales resultados en kPa. No indica el modelo utilizado, se realizó por medio de un software estadístico.

Tejido	Elasticidad	Elasticidad heterogénea
Glandular	11.28 ± 5.79	6.22 ± 3.48
Graso	9.24 ± 4.48	5.52 ± 3.04

Tabla 5 Resultados del estudio Factors influencing breast elasticity measured by the ultrasound Shear Wave elastography – preliminary results.

2.2 Graficación por Computadora

2.2.1 Concepto

La Graficación por Computadora tiene como fin principal mostrar datos en forma de imagen por medio de una computadora y es una rama de las Ciencias de la Computación, por tanto se realiza con base en diversos procesos, técnicas y algoritmos determinados.

Abarca la visualización de datos como puntos, imágenes de dos dimensiones, volúmenes en tres dimensiones, e incluso creando escenarios involucrando el tiempo como variable adicional.

A continuación se revisarán conceptos de Graficación por Computadora que serán usados en el transcurso de este trabajo, tanto de teoría como programación, abarcando hardware y software.

2.2.2 Frustum

En una escena de Graficación por Computadora, donde intervienen varios elementos, el frustum puede interpretarse como el contenido que se observa desde el punto de vista de la cámara, excluyendo el contenido no visible; por tanto, partiendo de este principio, el contenido visible es el único que debiera *renderizarse*, aunque la escena completa sea mucho más extensa. El beneficio principal es el costo computacional ahorrado, es decir, la cantidad de operaciones que no se realizan al visualizar la escena.

Conocer qué es lo que observa la cámara no necesariamente es tan sencillo como el concepto de frustum, existen diferentes algoritmos para saber a detalle la posición de los objetos en una escena. Una vez identificando un objeto visible en el frustum, es posible que los objetos cercanos también estén incluidos, por ejemplo, por una sombra o emisión de luz, por tanto,

se requiere de un algoritmo preciso para incluir correctamente todos los elementos involucrados en tal escena. Es ahí donde entra la participación del Octree, el cual descarta áreas que no se desea calcular. El principio básico del Octree se explicará a detalle más adelante.

Los elementos principales que integran al frustum se muestran en la Figura 5, así como la información que éste proporciona.

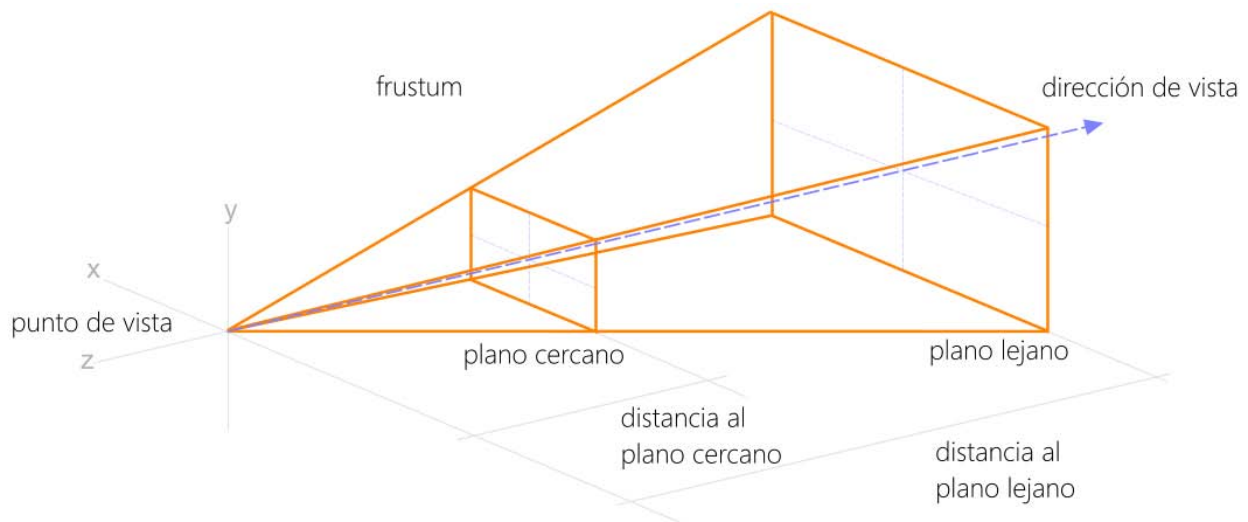


Figura 5 Frustum. Espacio que delimita lo que se observará en pantalla.

En la Figura 5 se observa cómo lo incluido dentro del frustum es lo que se encuentra entre el plano cercano y el plano lejano siguiendo la dirección de vista; una vez calculado, este espacio se utiliza para diferentes fines, por ejemplo, en el manejo de una escena, el espacio definido es lo que aparecerá en pantalla aunque haya otros objetos fuera del frustum; sólo se mostrará el campo de visión desde ese punto de vista, es decir, esta es la única información que será calculada para aparecer en pantalla, y cualquier información fuera de ella no debiera procesarse (*view frustum culling*).

Este principio general, que es enfocar el esfuerzo en realizar únicamente los cálculos de los objetos o de la zona afectada, y no de la escena completa, es el mismo principio que se ha tomado como guía para trabajar las operaciones del área a deformar en el volumen propuesto.

2.2.3 Transformaciones Geométricas

Son operaciones geométricas utilizadas para crear una nueva figura a partir de otra.

Formalmente las transformaciones geométricas en \mathbb{R}^3 son las aplicaciones lineales tal que $\varphi: \mathbb{R}^3 \rightarrow \mathbb{R}^3$.

Básicamente se dividen así:

- Isométricas: El homólogo (figura resultado) conserva dimensiones y ángulos.
- Isomórficas: El homólogo conserva la forma y los ángulos. Existe proporcionalidad entre las dimensiones del homólogo con el original.
- Anamórficas: Cambia la forma de la figura original.

A continuación se presentan las principales, así como la ecuación asociada.

Traslación del origen a un punto $\mathbf{p}_0 = (x_0, y_0, z_0)$

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -x_0 \\ 0 & 1 & 0 & -y_0 \\ 0 & 0 & 1 & -z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Escala por factores s_x, s_y, s_z

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotación alrededor del eje Z , de acuerdo al ángulo θ

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \text{sen}\theta & 0 & 0 \\ -\text{sen}\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotación alrededor del eje Y , de acuerdo al ángulo ϕ

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 & \text{sen}\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotación alrededor del eje X , de acuerdo al ángulo ψ

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\psi & \text{sen}\psi & 0 \\ 0 & -\text{sen}\psi & \cos\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

2.2.4 Modelo de Iluminación Phong

Antes de abordar el Modelo, se definen los siguientes conceptos:

- Resplandor. Cantidad de energía que fluye de la fuente de luz, se mide en Watts.
- Luminosidad o Flujo Luminoso. Cantidad de energía de la fuente percibida por el observador, medida en lumens.
- Brillantez. Descriptor subjetivo que refiere a la noción de la intensidad.

Phong es el modelo de iluminación más simple y popular en la Graficación por Computadora. En esencia es un modelo que parte de la idea de cómo la luz se refleja en una superficie. La luz se modela como un conjunto de los tres componentes de color discretos (rojo, verde y azul). Es posible calcular las intensidades de reflexión de luz independiente para cada fuente de luz y para cada uno de los tres componentes de color [\[38\]](#).

El Modelo Phong permite dos tipos de reflexión:

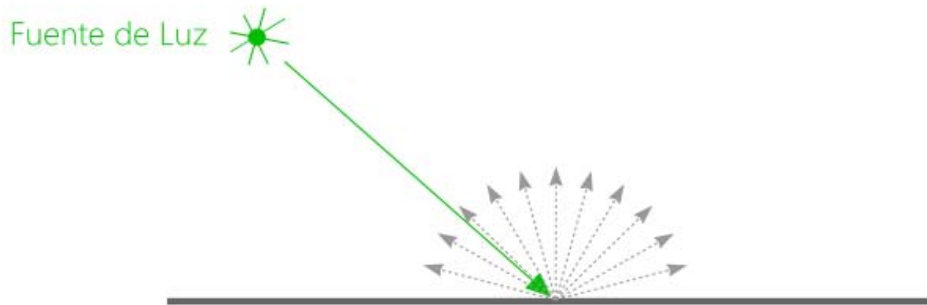


Figura 6 Ejemplo de Luz Difusa.

- Reflexión Difusa. La luz difusamente reflejada es una luz que se refleja uniforme en todas las direcciones de la superficie. La línea verde en la Figura 6 es un haz de la fuente de luz. Las líneas punteadas indican luz reflejada.

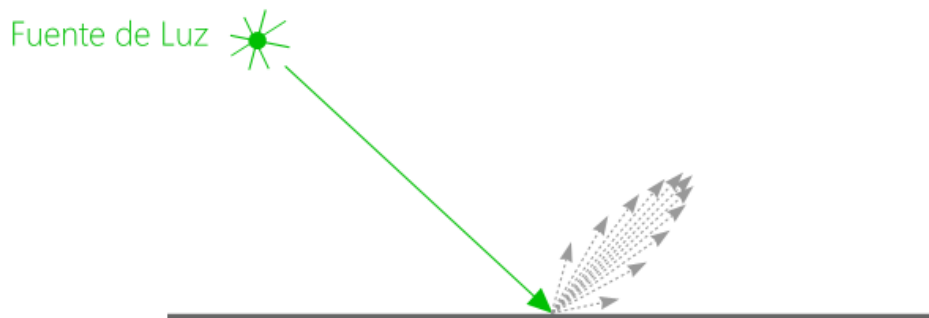


Figura 7 Ejemplo de Luz Especular.

- Reflexión Especular. La luz especularmente reflejada es una luz que es reflejada en una forma similar al espejo, en cuyo caso el ángulo de reflexión es aproximadamente igual a su ángulo de incidencia. Es la parte principal de la luz reflejada desde una superficie pulida o brillante. Suelen aparecer como manchas brillantes en superficies curvas. La línea verde en la Figura 7 es un haz de la fuente de luz. Las líneas punteadas indican luz reflejada.

Adicionalmente se considera una Luz Ambiental, la cual es una luz que se distribuye por igual en todas direcciones, de ahí que este tipo de luz se encuentra propagada en toda la escena.

El modelo de iluminación Phong otorga propiedades materiales a cada superficie; estas propiedades controlan cómo las luces iluminan la superficie. Excepto por el exponente de

reflexión especular, las mencionadas propiedades pueden ser seleccionadas independientemente para cada uno de los tres colores.

A continuación se especifican las propiedades.

- Propiedades de Reflexión Especular. Un coeficiente de reflexión especular, ρ_s , controla la cantidad de reflejo especular.
- Propiedades de Reflexión Difusa. Un coeficiente de reflexión difusa, ρ_d , controla la intensidad relativa de luz reflejada difusamente.
- Propiedades de Reflexión ambiental. Un coeficiente de reflexión ambiental, ρ_a , controla la cantidad de luz ambiental reflejada desde la superficie.

Propiedades de emisión. La emisión de una superficie controla cuánta luz de la superficie se emite en ausencia de cualquier luz incidente. La luz emitida desde una superficie no actúa como una fuente de luz que ilumina otras superficies; en cambio, sólo afecta el color visto por el observador.

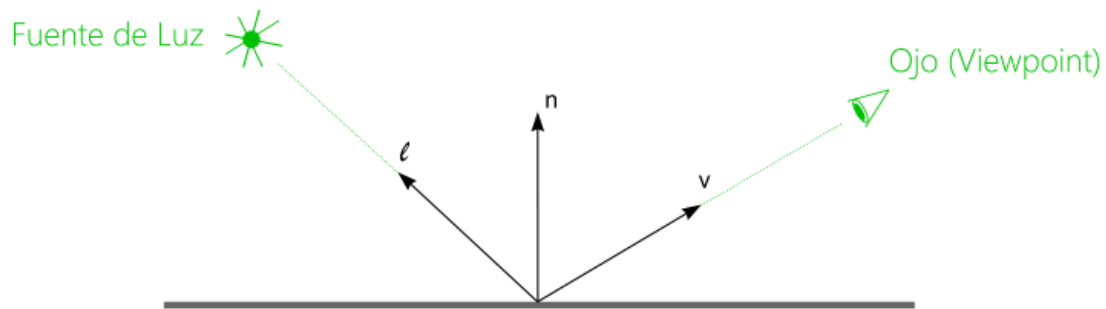


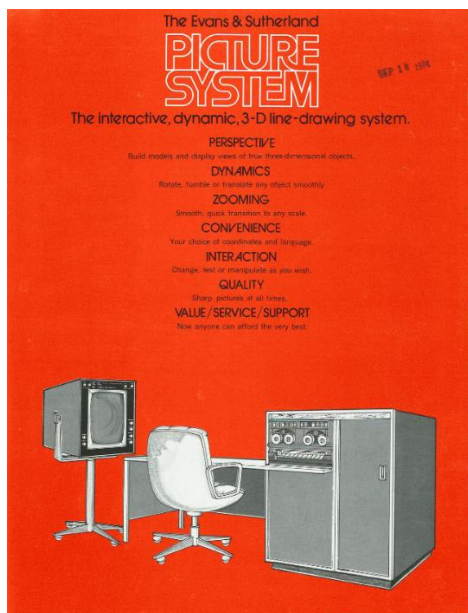
Figura 8 Rayo de Luz Incidente y los vectores relacionados.

Como se muestra en la Figura 8, un punto particular en la superficie es iluminado por una fuente puntual de luz y es visto desde otro punto. La orientación de la superficie es especificada por un vector n apuntando perpendicularmente a la superficie. La dirección de la luz es especificada por un vector unitario l que apunta desde el punto de la superficie hacia la fuente de luz. La dirección del punto de vista es similarmente especificado por un vector unitario v que apunta desde la superficie hacia el *viewpoint*. Estos tres vectores, más las propiedades de la fuente de luz y de la superficie del material, son usados por el modelo Phong para determinar la cantidad de luz que es alcanzada por el ojo.

2.3 Hardware

2.3.1 Graphics Processing Unit (GPU)

La Graficación por Computadora tiene más de 70 años de historia, y se remonta inicialmente a la computadora Whirlwind del Massachusetts Institute of Technology (MIT), la cual fue la primera en desplegar en video información en tiempo real, diseñada en 1947 [39], pasando posteriormente por los sistemas de Evans & Sutherland [40], así como el modelo de iluminación Phong en 1975 [41], también las computadoras de SGI en 1982, hasta las tarjetas de video programables utilizadas actualmente; las cuales pueden resolver una gran cantidad de operaciones al mismo tiempo.



SILICON GRAPHICS | The Source of Innovation and Discovery

Silicon Graphics Fuel® Visual Workstation

Powering a New Level of Performance, Productivity, and Precision on the Desktop

The Silicon Graphics Fuel visual workstation maximizes the performance of your desktop applications while offering you unprecedented productivity gains. A premium blend of industry-leading technology from SGI, Silicon Graphics Fuel features the latest 64-bit R10000 processor and the unparalleled 100% 3D graphics system for R10000 as a new high-classroom architecture. Give your reliability and productivity to a new level with Silicon Graphics Fuel.

Features	Benefits
• High-Performance Processing and Transference High-Resolution Architecture	• Maximized Performance and Throughput
• Single 600 MHz or 700 MHz R10000 processor for each dual L2 cache, 512KB cache for each L2 cache, 100% 3D graphics system architecture based on the SGI R10000 kernel	• Proven 64-bit architecture on a high-bandwidth architecture integrates application performance with CPU-intensive applications
• Industry-leading 3,000 per-second memory bandwidth and 1.6GB per-second system graphics refreshment	• Maximize application performance and throughput for image manipulation and real-time visualization with large models and data sets
• 4 integrated PCI slots, internal CD/DVD-ROM, external DVD-RAM	• Expanded range of available options
• Advanced Wire Graphics	• Accessibility of Highest Quality Desktop Graphics
• Characteristic scalable graphics performance with VP4™ V10 or V12 graphics with non-proprietary graphics hardware supported by the REX OS and SGI IRIX	• Economy, fast geometry and 16-bit performance for high-speed drawing even with very complex, and high-quality image operations, and with fully on-board design, maximum flexibility with cache and lower-resolution size off-screen graphics memory
• Advanced feature management with up to 100MB of feature extension, asynchronous feature download capability	• Efficient rendering of volumetric data sets, high performance processing of textures
• OpenGL on a Chip™	• Hardware acceleration of OpenGL core features, including 3D texture for volume rendering and imaging extensions
• Hardware-accelerated specular shading (Phong effects)	• Improved shading and accuracy for lighting of 3D models without a
• 48-bit (12-bit per component) RGBA with 16-bit Z-buffer capability	• Chromatic quality and highest precision for 2D/3D imaging
• Uncompressed non-compressed serial digital video output with DVI/SDI™	• High-quality video input and output for editing, compositing, and visual effects
• Support for high resolutions, including HDTV, full-frame sensor input and output in a standard Dual Channel Display option	• Capable to display 8-bit data sets at high resolutions, 16-bit video systems, four-channel dual display for double the screen real estate with a single graphics base
• 66-MHz hardware-accelerated non-rotation buffer	• High performance and accuracy with depth of field, soft-focus and anti-aliasing, motion blur, and other effects
• Proven, Robust IRIX OS from SGI	• Proven, Reliability, Leading Real-Time Response and Flexibility
• Built on the 6th-generation 64-bit IRIX operating system	• Industry-leading real-time response, scalability, and reliability, strong compatibility with other IRIX products

Figura 9 Izquierda. The Evans & Sutherland Picture System. The Interactive, dynamic, 3-D line-drawing system. 1974.

Derecha. Silicon Graphics Fuel Visual Workstation, información general. 2004 [40].

Buscando resolver el problema de aplicar una deformación a un modelo volumétrico detallado con respuesta lo más cercana posible al tiempo real, es necesario utilizar la paralelización de operaciones con base en una unidad de procesamiento gráfico (GPU, Graphics Processing Unit), dado que se requiere ejecutar al mismo tiempo gran cantidad de operaciones con respuesta inmediata.

Una GPU es un coprocesador que usualmente está dedicado al procesamiento de gráficos y se ideó para aligerar la carga de operaciones del procesador principal. Es comúnmente usada

para aplicaciones virtuales, ambientes 3D o videojuegos, aunque actualmente también funciona para procesar grandes cantidades de datos que no tienen qué ver con gráficos, sino incluso estadísticos.

A continuación se presenta la estructura general de una GPU y se compara con la de una CPU, dándose además las especificaciones técnicas de la tarjeta mencionada:

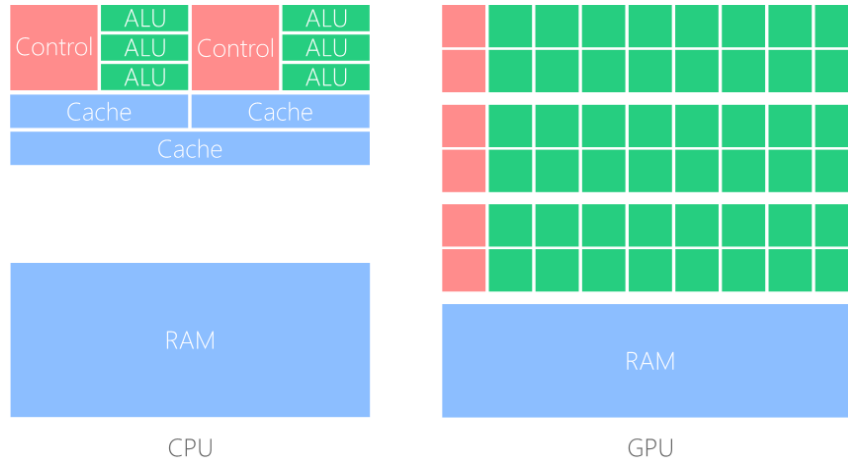


Figura 10 Comparación básica entre la estructura de una CPU y una GPU.

Como se muestra en la Figura 10, la diferencia principal entre una CPU y una GPU es que la GPU incluye cientos de núcleos que usa para realizar operaciones matemáticas al mismo tiempo (paralelización), en comparación con una CPU, la cual puede realizar sólo unas cuantas, dependiendo el modelo.

Es importante mencionar que los núcleos de una CPU son equivalentes a un microprocesador completo, en cambio, los núcleos de una tarjeta GPU son más simples, en esencia son unidades aritmético-lógicas. Otra diferencia es que las operaciones en paralelo que realiza una GPU son idealmente Single Instruction Multiple Data (SIMD), esto es, realizar la misma operación para una gran cantidad de datos, con una salida normalmente de la misma cantidad de datos de entrada; de ahí su gran utilidad en procesamiento de gráficos, matrices, etc. En cambio, para realizar Multiple Instruction Multiple Data (MIMD) es más usual realizarlo por medio de una CPU, dado que cada uno de sus núcleos puede manejar diferentes instrucciones.

La ejecución de las operaciones paralelizables en la GPU se programan por medio de un kernel (porción de código), el cual contiene un grupo breve de instrucciones que se aplicarán a cada uno de los datos a operar. Esto se entiende usualmente como múltiples hilos ejecutando la misma tarea, pero con datos diferentes.

Actualmente la mayoría de las computadoras, tablets y celulares, integran algún tipo de GPU, dado que es mucho más barato y sencillo integrar múltiples núcleos de GPU a múltiples núcleos de CPU.

La programación en paralelo se ejecuta sobre una tarjeta gráfica marca NVIDIA modelo GEFORCE GT 650M con 2GB de memoria, y un procesador Intel Core i7 de tercera generación a 2.40GHz con 16GB de memoria RAM, bajo el sistema Windows 8, desarrollando en Visual Studio 2012 entorno C++.

A continuación el detalle técnico de la tarjeta gráfica Nvidia Modelo GeForce GT 650M

- Especificaciones del motor de la GPU
 - Núcleos CUDA 384
 - Reloj del procesador Up to 900 MHz
 - Tasa de llenado de textura 27.2
- Especificaciones de la memoria
 - Interfaz de memoria 128-bit DDR3/GDDR5
 - Ancho de banda máximo Up to 80.0
- Características de tarjeta
 - 3D Vision
 - PhysX
 - Entorno de programación CUDA
 - DirectX 11
 - OpenGL 4.1
 - Tipo de bus PCI-E 3.0, PCI-E 2.0
 - Blu Ray 3D
- Especificaciones
 - Resolución Digital Máxima 3840x2160
 - Resolución máxima de VGA 2048x1536
 - HDCP
 - HDMI

2.3.2 Ultrasonido

El ultrasonido se define como una serie de ondas, generalmente longitudinales, originadas por la vibración de un cuerpo elástico (cristal piezoeléctrico) y propagadas por un material (tejidos corporales) cuya frecuencia está por arriba de la audición humana, 20,000 ciclos/segundo o 20 kilohertzios (20 KHz). [\[42\]](#)

Su aplicación principal es realizar diagnósticos médicos, con lo cual es posible estudiar diferentes órganos del cuerpo, usualmente los del abdomen, así como el estudio de la evolución de un embarazo. El ultrasonido es una herramienta considerada no invasiva, que a diferencia de los rayos X, no presenta exposición a radiación ionizante.

Un beneficio principal es que el equipo no tiene un costo elevado, de hecho es relativamente barato comparado a otros equipos que usan otros métodos; además, obtiene resultados en tiempo real y, por su tamaño y modelos actuales, el equipo se puede transportar a regiones aisladas donde otro tipo de herramientas difícilmente pueden lograrlo. Un ejemplo de ello se observa en la Figura 11.

Entre las desventajas se encuentra que usualmente se requiere un técnico especializado en este tipo de estudios médicos, además, el detalle que presenta muchas veces no es suficiente para determinar y diagnosticar algunos tipos de enfermedades. Actualmente existen varias técnicas y tipos de ultrasonido que parten del mismo principio, pero aplican tecnología y características cada vez más avanzadas.



Figura 11 Izquierda. General Electric Vscan Pocket Ultrasound Version 1.2, Still frame: JPEG, Movie: MPEG-4, Voice notation: WAV, costo aproximado \$7,900.00 dólares [\[43\]](#). Derecha. Sonoscape S9 Portable Color Ultrasound Doppler ISO CE FDA [\[44\]](#)

CAPÍTULO 3

Diseño y Construcción del Modelo basado en Octree

Se define el diseño y construcción del modelo deformable basado en el método Octree, el cual se usa principalmente para buscar el área afectada del volumen a deformar de forma rápida y efectiva. Se explican otros detalles del modelo que tienen como objetivo la definición y delimitación del área deformada originada en un punto de impacto.

3.1 Octree

3.1.1 Definición y Características

Un octree es una estructura de datos con base en una forma de árbol, en la cual cada nodo es sub dividido en ocho octantes, las cuales son determinados de acuerdo al punto central físico de su nodo padre. La clave del octree es que este proceso puede ser recursivo para cada octante resultado, deteniéndose hasta que llegue a un nivel determinado o hasta que los puntos a insertar terminen.

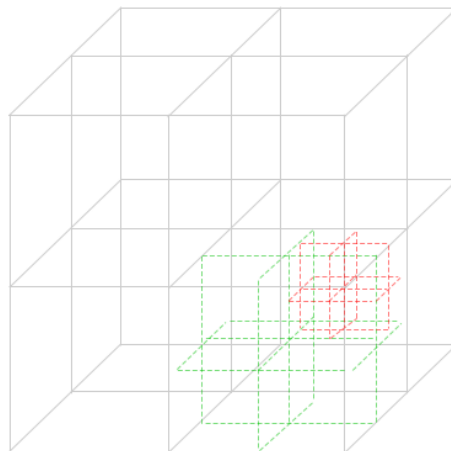


Figura 12 Muestra cómo se subdivide un Octree en octantes dentro de octantes.

Para este trabajo se utilizará un Octree, el cual es el que divide en ocho octantes al volumen, pero también existe una versión similar, el *quadtree*, que como su nombre lo menciona, se divide en cuatro cuadrantes, y así sucesivamente.

Esta técnica es usada comúnmente en el manejo de grandes cantidades de datos, sobre todo en gráficos, más precisamente en objetos y ambientes 3D, por ejemplo, para localizar objetos en un área de visualización y así reducir el número de cálculos que se van a realizar para mostrar la escena.

Otra de las varias ventajas del octree es que suele ser usado en colisiones, enfocándose así en la sección colisionada y no en todo el volumen, evitando un número amplio de comprobaciones de colisiones y posteriores operaciones.

La idea del Octree es partir de un cubo inicial, llamado nodo raíz, que contenga en su interior todo el volumen que debiera ser subdividido y, como lo indica el proceso, subdividirlo en ocho cubos de igual tamaño, continuando recursivamente. Dicha recursividad depende directamente de cuántos niveles se desea que tenga el árbol, o incluso, por el número de inserciones en él.

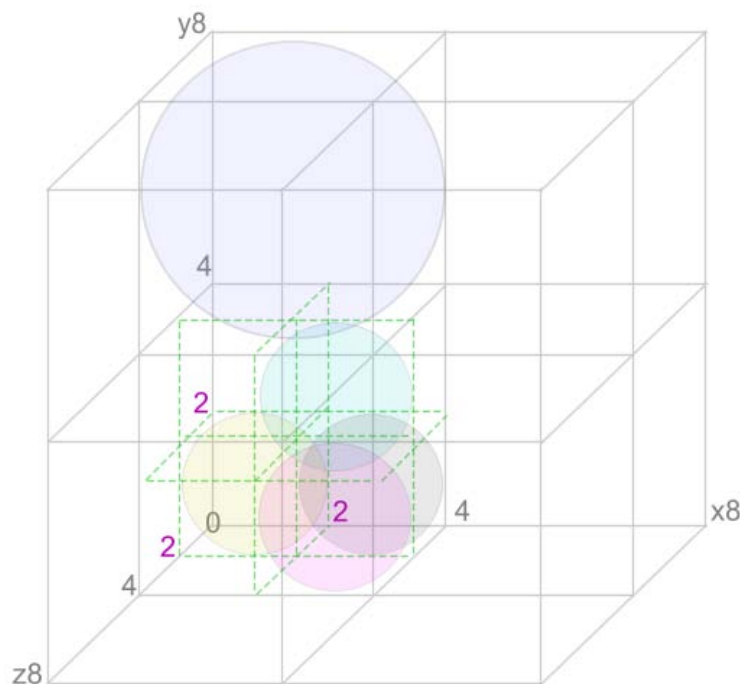


Figura 13 Ejemplo sobre cómo se subdivide en octantes un Octree. Las esferas muestran hojas ocupadas por datos, por tanto, si hay más de un dato en un octante, ese octante se subdivide en 8, y así sucesivamente.

Al construir un Octree, lo ideal es que tenga flexibilidad, es decir, que se pueda reutilizar en otros módulos, y sea ampliable o adaptable a otras funciones con diferente fin. Con esto también se espera que sea dinámico en tamaño, es decir, que no posea un tamaño predefinido y pueda así adaptarse a una escena dependiendo de sus características o de las dimensiones del volumen que se espera contenga.

Como se comentó anteriormente, la importancia de usar un octree radica en acelerar diversos procesos y, por tanto, hacer más eficientes las tareas. Se desea por tanto que el Octree tenga una velocidad de creación rápida, sobre todo en condiciones de movimiento o deformación continua, ya que en esos casos deberá recrearse después de cada cambio de posición.

3.1.2 Construcción de un Octree

Un Octree tiene como base un árbol de datos inicialmente vacío y se construye por medio de n llamadas para insertar n puntos en él. Dicha inserción está basada en un criterio de tres pasos principalmente:

1. El nodo es un nodo interior al árbol.
Se buscará en cuál de los ocho hijos del nodo residirá el punto mediante una llamada recursiva.
2. El nodo es una hoja (sin hijos) y no tiene datos asignados.
Dado que no hay datos asignados en la hoja y tampoco hijos, se asigna el punto al nodo hoja.
3. El nodo es una hoja y tiene un dato asignado.
En este punto nos encontramos en una hoja, pero ya existe un dato asignado ahí. Dado que sólo se guarda un punto por hoja, es necesario recordar qué dato estuvo ahí, además del nuevo dato que se asignará, una vez que los datos están en memoria. El nodo se parte en ocho hijos y posteriormente ambos datos son insertados siguiendo el paso 1.

A continuación se muestra el algoritmo utilizado para el criterio de inserción de un punto al Octree. En esencia se utilizó el criterio expresado en los puntos anteriores.

Algoritmo 2.0 De inserción de un punto al Octree.

```
1: if isLeafNode() // si el punto está en un nodo hoja
2:     if data==NULL // Si el nodo hoja no tiene un punto asignado,
3:         data = point; // guarda el punto
4:     else
5:         for (int i=0; i<8; ++i) // crea nuevo sub octree
6:             children[i] = new Octree
7:         end for
8:         getOctantContainingPoint(oldPoint) // identifica el octante donde debería estar contenido el punto 1
9:         children[octant]->insert(oldPoint) // inserta punto 1
10:        getOctantContainingPoint(point) // identifica el octante donde debería estar contenido el punto 2
11:        children[octant]->insert(point) // inserta punto 2
12:    end if
13: else
14:    getOctantContainingPoint(point) // obtén el octante donde debería estar contenido el punto
15:    insert(point) // reinserta el punto en el octante
16: end if
```

3.2 Funciones y Operaciones Auxiliares

3.2.1 Cálculo de un punto contenido en una esfera

Otra de las operaciones necesarias es saber si un punto está contenido en una esfera. El cálculo utilizado se presenta en la Figura 14.

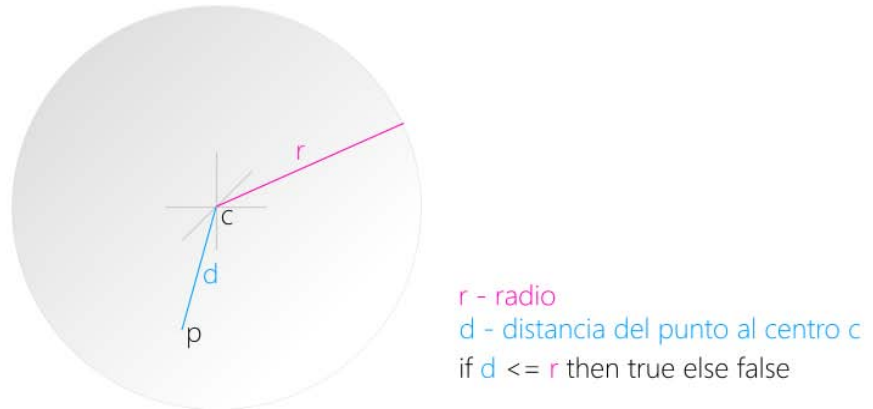


Figura 14 Cálculo simple de un punto contenido en una esfera, en azul la distancia d del punto p al centro de la esfera c , en rosa el radio r de la esfera. La comparación de estas distancias dictamina la pertenencia del punto.

Para saber si un punto se encuentra dentro de una esfera se realiza una diferencia entre el radio r de la esfera y la distancia d del punto hacia el centro de la misma. Si r es mayor que d , entonces el punto está contenido, y si no cumple con lo anterior, el punto está fuera.

3.2.2 Intersección entre dos cubos que tienen la misma inclinación

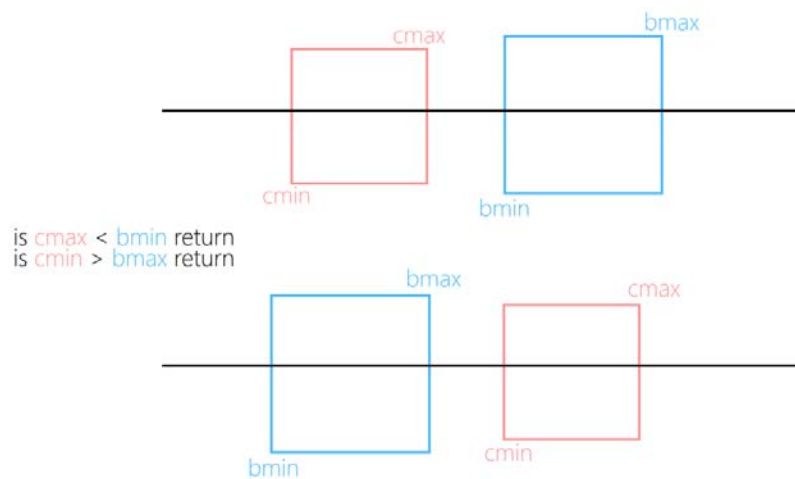


Figura 15 Representa dos cubos con la misma inclinación, así como sus esquinas máximas (c_{max} , b_{max}) y mínimas (c_{min} , b_{min}), es decir, los puntos más orientados a dirección positiva y negativa (xyz) respectivamente.

Un método para detectar si hay colisión de dos cubos que se encuentran con la misma inclinación y diferente tamaño y posición, es evaluando si el punto mínimo b_{min} del cubo b

es menor al punto máximo *cmax* del cubo *c* (Figura 15); o bien, si el punto mínimo *cmin* del cubo *c* es menor que el punto máximo *bmax* del cubo *b*, también existe colisión. En caso contrario, los objetos no colisionan.

3.2.3 Búsqueda de puntos delimitados por una esfera usando un Octree

Una de las tareas principales para lograr los objetivos es obtener los puntos contenidos en una esfera utilizando el Octree.

A continuación se muestra el algoritmo utilizado para la búsqueda de estos puntos. Básicamente se realiza una comparación entre los extremos positivo y negativo del cubo contra la distancia euclidiana con base en el centro y radio de la esfera.

Algoritmo 3.0 Búsqueda de puntos contenidos en la esfera

```
1: if (isLeafNode()) // si el punto se encuentra en un nodo hoja.  
2:     if (data!=NULL) // si existe un dato guardado  
3:         data->getPosition(); // obtiene la posición del punto  
4:         if (dist_squared < ( pow((p.x-s.x),2) + pow((p.y-s.y),2) + pow((p.z-s.z),2) )) // compara radio contra distancia  
           del punto al centro de la esfera.  
5:             results.push_back(data); // guarda en resultado  
6:         end if  
7:     end if  
8: else // se encuentra en un nodo interior  
9:     for (int i=0; i<8; ++i) // se encuentra en un nodo interior  
10:         Vec3 cmax // máxima esquina del cubo  
11:         Vec3 cmin // mínima esquina del cubo (esquina más alejada del anterior)  
12:         if (s.x < cmin.x) dist_squared -= pow(s.x - cmin.x,2) // verifica distancia euclidiana para ambas esquinas.  
13:         children[i]->getPointsInsideSphere() // busca puntos según criterio del paso anterior  
5:     end for  
6: end if
```

3.2.4 Cálculo de frontera para semiesfera y elipsoide

En la Figura 16 se observa que en cada pieza de la semiesfera se mide un límite (color verde), el cual es un espacio que se denomina frontera. El motivo de esta frontera es que cuando se aplica una fuerza sobre una parte del cuerpo en un punto dado, este tiende a deformarse, pero hasta cierto punto, que establece el límite de profundidad de deformación y es un parámetro para el modelo creado.

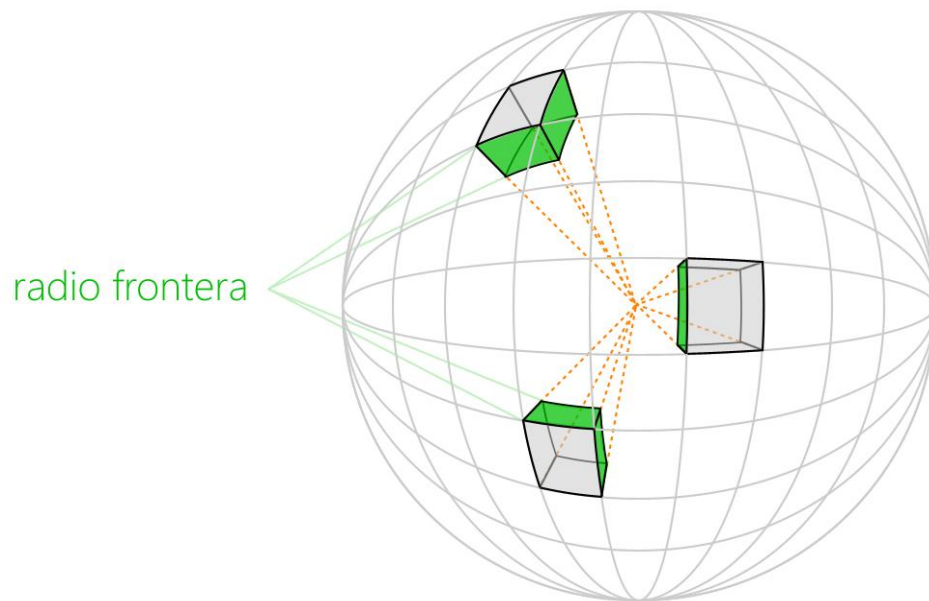


Figura 16 Describe otra forma para calcular una zona de frontera para una esfera, esto es, dividiendo en segmentos la esfera y calculando esa distancia frontera por segmento. Este método puede aplicarse a otros objetos como una semiesfera donde se puede conocer su origen.

Calcular la zona frontera de una semiesfera es sencillo si se trabaja con el radio de la misma, el radio define el límite frontera y todos los puntos que estén más allá estarán fuera; pero el volumen en que se buscan los puntos frontera no necesariamente tiene la forma de una semiesfera, varía según el origen del volumen (en este trabajo se parte de la forma básica de un seno), puede ser, por ejemplo, un elipsoide (Figura 17).

Para una esfera, funciona bien la búsqueda de puntos definiendo una frontera con base en el radio, pero no para el caso de la semiesfera; ahora, si se hace una búsqueda de puntos frontera por sección (Figura 16), se obtendrían mejores resultados. Si la forma es aún más compleja e incluye deformaciones, es más difícil definir una frontera.

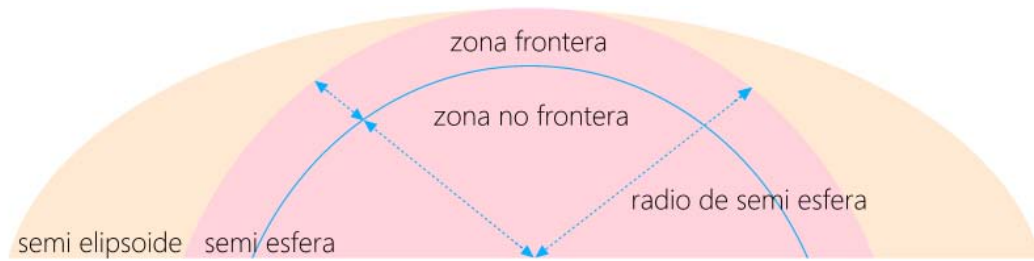


Figura 17 Muestra una forma para calcular una zona de frontera para una semiesfera, pero también que no es posible replicarla usando el radio para una elipsoide.

3.2.5 Puntos afectados por el impacto

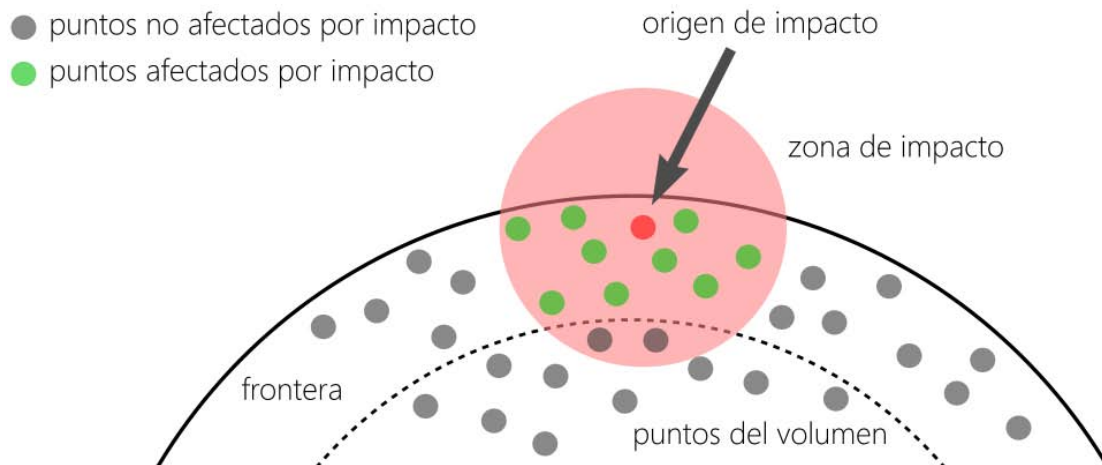


Figura 18 Describe cómo, para este trabajo, es necesario calcular sólo los puntos que se encuentran contenidos en la frontera, quedando discriminados los que están fuera, y además, considera sólo los que se encuentran dentro de la esfera de impacto (marcada en rojo). De ahí que los puntos que se modificarán son sólo los verdes.

Cuando se ejerce una fuerza en el cuerpo humano, sobre un punto determinado de la piel, por ejemplo, en una inyección, esta fuerza no afecta a toda esa parte del cuerpo, sino que sólo se afecta un área determinada. Partiendo de esta idea, se define una frontera, así como un radio de afectación desde el punto de impacto, con el objetivo de delimitar el área que sufrirá un cambio de la que no.

De acuerdo a la Figura 18, los puntos alrededor del origen son los únicos que sufren la deformación, pero además de considerar el radio de impacto, debe considerarse la frontera del volumen que es propensa a deformación (de acuerdo a lo comentado en el párrafo anterior), por tanto sólo se verán afectados los puntos que hagan intersección entre estas dos secciones, de ahí que los puntos que queden afuera no son afectados.

CAPÍTULO 4

Visualización del Modelo

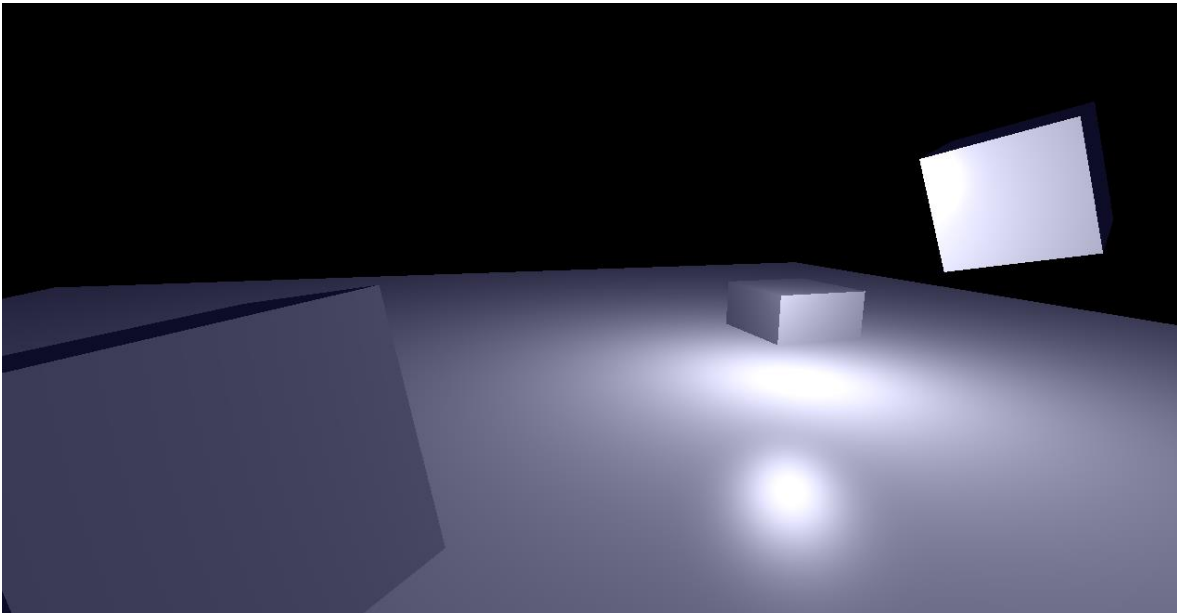


Figura 19 Escena de entrenamiento usando triángulos predefinidos con índices, añadiendo un plano e iluminación ambiental, especular y difusa.

Se programó un motor de visualización para poder observar las deformaciones generadas por el modelo creado; dicho motor utiliza OpenGLSL para paralelizar procesos, Vertex Buffer Objects para un eficiente manejo de los datos, y QT para la navegación con teclado y mouse alrededor del volumen y seleccionar el punto de impacto de la deformación.

4.1 Open Graphics Library Shading Language (OpenGLSL)

4.1.1 Concepto

OpenGL (Open Graphics Library) es una interfaz de programación (API) multiplataforma que

consta de más de 250 librerías independientes compatibles con diversos lenguajes de programación. Gracias a sus librerías específicas para graficar, OpenGL es principalmente usada para representar escenas virtuales de diferente ámbito, que van desde el científico hasta incluso videojuegos.

El OpenGL Shading Language (GLSL) es una parte fundamental de la API de OpenGL. En programas donde se implementa OpenGL, usualmente se utilizan uno o varios sub programas GLSL. Estos mini programas escritos en GLSL son referidos normalmente como *shaders*, los cuales corren a nivel de GPU y típicamente implementan algoritmos de iluminación y efectos para imágenes tridimensionales, motivo por el cual son ampliamente usados en la aceleración de gráficos [\[43\]](#). Los shaders están diseñados para ejecutarse directamente en el GPU y, por tanto, en programación paralela de procesos usualmente repetitivos.

4.1.2 Vertex Shader y Fragment Shader

Los Vertex Shader son subprogramas de OpenGLSL que actúan sobre los vértices y normalmente generan la salida de la posición del vértice en la ventana gráfica; aunque esta no es la única función que realizan, ya que pueden contener otros atributos dependiendo de las necesidades específicas del proyecto.

Los Fragment Shader son otro tipo de subprogramas de OpenGLSL que actúan a nivel fragmento y normalmente suponen la salida del color del pixel, para lo cual realizan una interpolación basadas en atributos de los vértices y la distancia resultante, aunque al igual que en el caso de los Vertex Shaders, pueden dar salida de otros atributos. De ahí que ambos son “subprogramas”.

En la Figura 20, el Vertex Shader es llamado sólo 3 veces, una para cada vértice del triángulo mostrado, en cambio, el Fragment Shader es llamado tantas veces como pixeles cubren el triángulo, ya que en este caso en específico, el resultado va cambiando con respecto a los colores que se presentan en cada vértice.

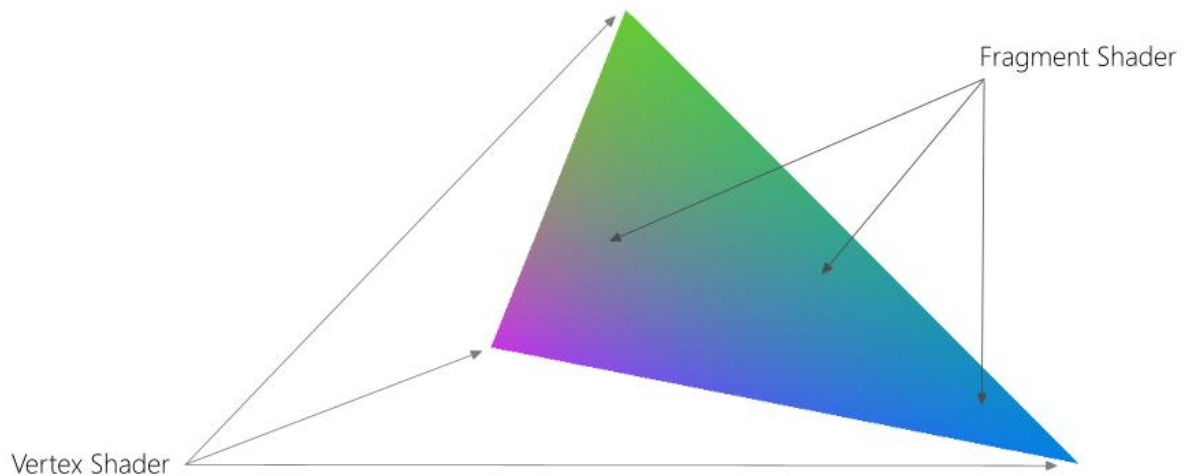


Figura 20 Ejemplifica que el vertex shader es usado en los tres vértices del triángulo pero el fragment shader es llamado para todos los demás pixeles.

4.1.3 Vertex Buffer Objects

Un Vertex Buffer Object (VBO), en programación, es un Objeto tipo Buffer, el cual es usado como fuente para almacenar datos de arreglos de vértices en el VRAM (Video Random Access Memory), por lo que acelera el rendering sobre todo cuando la cantidad de datos que se quiere trabajar es muy grande. Los vértices pueden contener relacionada tanta cantidad de información como así lo requieran para las primitivas que se requieran desplegar posteriormente.

Usar arreglos de vértices puede reducir el número de llamadas a funciones, así como el uso redundante de vértices compartidos, de ahí que son utilizados ampliamente por su alto rendimiento.

Los Vertex Buffer Objects se puede catalogar en 3 tipos: static, dynamic y stream; de los cuales los últimos dos manejan datos dinámicos como animaciones.

- Static. Los datos almacenados serán modificados una vez y usados muchas veces.
- Stream. Los datos almacenados serán modificados una vez y usados al menos pocas veces.
- Dynamic. Los datos almacenados serán modificados repetidas veces y usados muchas veces.

A continuación se presenta, a manera de algoritmo, las funciones utilizadas para trabajar la visualización usando VBO, en el orden que se utilizaron, y se agrega una breve descripción de su funcionamiento.

Algoritmo 3.0 Proceso inicial

```
01: glGenBuffers // genera buffer
02: glBindBuffer // asigna el buffer y tipo de buffer
03: glBufferData // envía datos y especifica el tipo de uso
04: glBufferSubData // actualiza un subconjunto de los datos almacenados
05: glGenVertexArrays // genera nombres para objetos de arreglo de vértices (Vertex Array Object, VAO)
06: glBindVertexArray // asigna un VAO
07: glEnableVertexAttribArray // activa o desactiva el arreglo de atributos de vértices, esto porque se están usando variables de atributo para shaders.
08: glBindBuffer // asigna el buffer y tipo de buffer
09: glVertexAttribPointer // asigna un puntero que le indica a OpenGL dónde buscar los datos en el arreglo de atributos de vértices.
10: glDrawElements // Dibuja los vértices.
```

4.2 Visualización

4.2.1 Volumen de prueba

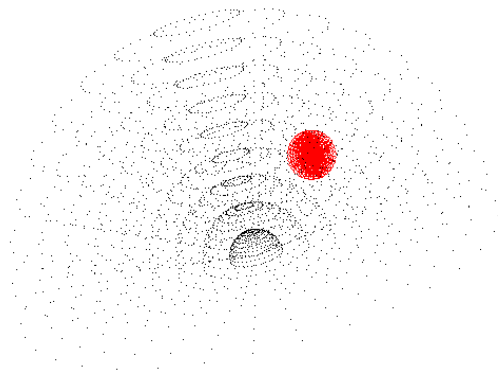


Figura 21 Semiesfera con esfera contenida dentro de la primera. Ambas a base de puntos buscando simular una mama con un tumor adentro.

Con el fin de preparar las pruebas iniciales de reconocimiento y funcionamiento del octree, se creó una semiesfera de 2,906 puntos y otra esfera de 5,525 puntos contenida dentro de la primera, como se observa en la Figura 21.

Además se eliminaron los puntos de la semiesfera que pudieran estar contenidos en la esfera y también se dejó un espacio de separación buscando que los puntos no quedaran pegados entre ambos objetos.

Para realizar otras pruebas, se crearon otros dos volúmenes siguiendo las mismas condiciones que las anteriores, con la diferencia de que la semiesfera contiene 144,991 puntos y la esfera 150,593.

Para ambos casos de los volúmenes anteriores se crearon salidas en formato txt con el siguiente orden: x espacio y espacio z cambio de renglón; los cuales fueron usados posteriormente tanto en el octree como en las pruebas de visualización.

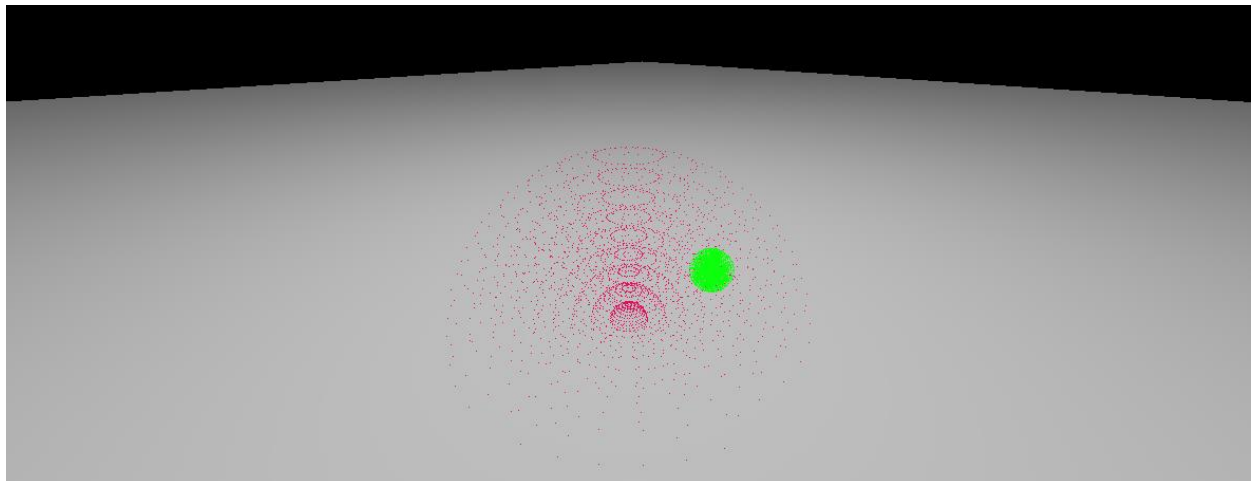


Figura 22 Ejemplo de visualización del modelo de prueba 1 graficado en OpenGLSL con QT.

Una vez cargado el modelo, es posible navegar libremente en la escena y observar a detalle los resultados de la deformación.

El modelo permite ingresar hasta tres objetos diferentes, e independientemente de su forma, los graficará como puntos, o bien, si se tienen los vértices, podrá graficarlos como triángulos. Haciendo algunos ajustes en la programación es posible graficar aún más objetos, en caso de ser necesario. En la Figura 22 se muestran 295,584 puntos graficados, 144,991 de la mama y 150,593 del tumor.

4.2.2 Navegación en la escena

Una vez graficados los objetos, es posible moverse en la escena con fluidez usando el mouse y el teclado. El movimiento del mouse permite desplazarse en la escena cambiando la dirección del punto de vista, mientras que la rueda del mouse permite realizar un zoom a la escena; además, con el teclado es posible realizar otras acciones. Así mismo, por separado es posible usar el teclado para seleccionar con precisión el punto donde se desea que se origine la deformación. A continuación el listado completo:

Posición del Punto de Vista

- Tecla A – Desplaza la cámara hacia la izquierda.
- Tecla B – Desplaza la cámara hacia la derecha.
- Tecla S – Desplaza la cámara hacia abajo.
- Tecla W – Desplaza la cámara hacia arriba.

Dirección del Punto de Vista

- Movimiento del ratón – Mueve hacia donde se desea que apunte la cámara.

Zoom

- Rueda del mouse adelante: Zoom in.
- Rueda del mouse atrás: Zoom out.

Selección del punto de impacto

- Tecla desplazamiento arriba: Incremento eje X.
- Tecla desplazamiento abajo: Decremento eje X.
- Tecla desplazamiento derecha: Incremento eje Z.
- Tecla desplazamiento izquierda: Decremento eje Z.
- Tecla Y: Incremento eje Y.
- Tecla H: Decremento eje Y.
- Botón izquierdo del mouse: Selección del punto de impacto.

CAPÍTULO 5

Análisis de Resultados

Se muestra el resultado de aplicar deformaciones utilizando el modelo deformable y el visualizador sobre un volumen virtual de entrenamiento. Se detalla el procedimiento y resultados al deformar un volumen reconstruido virtualmente con base en un objeto real construido por alcohol polivinílico y una uva en su interior, para simular una mama, y dentro de ella, un tumor.

Se plantean las conclusiones, se comentan los puntos a mejorar y el trabajo a futuro.

5.1 Visualización

5.1.1 Preparación de Visualización

Para verificar que la carga del modelo funciona correctamente en el visualizador, se ha trabajado un volumen que fue originado con tomas de imágenes de ultrasonido. Este volumen fue provisto por Fabián Torres del Laboratorio de Imagenología Biomédica del Centro de Ciencia Aplicada y Desarrollo Tecnológico (CCADET-UNAM) [\[45\]](#), quien hizo tomas de imágenes de ultrasonido, y una vez obtenidas, realizó un proceso de creación del volumen.

Ya con los datos del volumen, se utilizó 3D Slicer 4.3.1 para trabajar una segmentación del mismo utilizando la técnica threshold, la cual consiste en eliminar los puntos que no se encuentran dentro de un umbral. Este umbral, en el caso específico del volumen tratado, fue dictaminado por el valor del pixel que indica el tono del color, donde 0 representa negro, y 255 blanco. El objetivo de realizar la segmentación es eliminar las partes que no son tejido mamario y dejar sólo las partes que sí.

Intentando mantener un algoritmo no complicado y reutilizable, la programación del Octree se ha hecho con vectores y se ha trabajado para que maneje diferentes variables, de las cuales, en este ejemplo de visualización, sólo se utiliza la posición xyz.

5.1.2 Entrenamiento de Visualización

El resultado de la imagen segmentada en 3D Slicer puede observarse en la Figura 23 presentada a continuación.

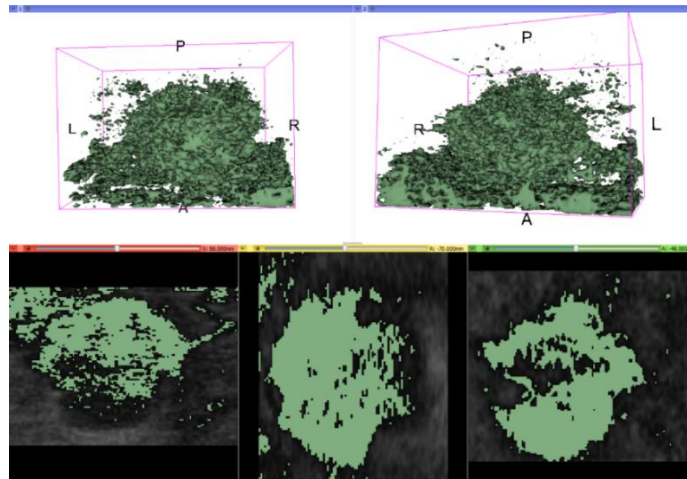


Figura 23 Segmentación de un volumen de mama creado con base en imágenes de ultrasonido. El software utilizado es 3D Slicer.

Una vez realizada la segmentación del volumen, los datos del mismo se exportaron a un archivo de texto, con lo que se crearon 294,755 puntos con coordenadas xyz separadas por tabulador; posteriormente el modelo segmentado se cargó en el visualizador de volúmenes creado, y el resultado se muestra a continuación en la Figura 24:



Figura 24 Modelo de volumen de mama segmentado y visualizado con el motor de visualización programado.

5.2 Búsqueda utilizando Octree

5.2.1 Preparación de búsqueda del Octree

Se realizaron pruebas a un volumen envuelto por un Octree con dimensiones fijas, se ejecutó una búsqueda sobre una esfera de puntos de tamaño variable, sobre la cual se ha comentado en el capítulo anterior.

Con el fin de encontrar las capacidades del modelo, se midieron tiempos de inserción en el Octree, así como de búsqueda por Fuerza Bruta y búsqueda usando el mismo Octree; se trabajó con diversos valores para el diámetro de la esfera, desde .00001 unidades hasta 1 unidad, con tamaño de Octree y volumen de 2 unidades de largo, ancho y alto; y se hicieron pruebas que iban desde mil puntos insertados, hasta 10 millones de puntos. Los resultados obtenidos se presentan en la Tabla 6.

A continuación se muestran los datos con el resultado de ejecución del Octree con 10 millones de puntos insertados.

- Octree creation: 23.406487 seconds
- Total inserted points: 10,000,000
- Using Brute-Force found 653,936 points in 0.31943 seconds
- Using Ocree found 653,939 points in 0.878996 seconds

5.2.2 Tiempo de búsqueda e inserción usando Octree

Como se muestra en los resultados de la Tabla 6, el mejor tiempo de búsqueda de puntos dentro de una esfera, usando el modelo de Octree, es de 0.00002 segundos con un diámetro de búsqueda de 0.0001 unidades (0.005% el largo del Octree) sobre 10 mil puntos, e incluso disminuyendo el radio de búsqueda hasta 0.00001 unidades (0.0005% el largo del Octree) para mil puntos, este resultado (0.00002 segundos) se consideraría, hasta ahora, el mejor resultado para la búsqueda de puntos contenidos por una esfera.

puntos insertados	dimensión de octree [unidades]	diametro de esfera [unidades]	tiempo inserción octree [segundos]	tiempo de búsqueda usando fuerza bruta [segundos]	tiempo de búsqueda usando Octree [segundos]
1k	2x2x2	0.00001	0.00093	0.00002	0.00002
1k	2x2x2	0.00010	0.00098	0.00005	0.00002
10k	2x2x2	0.00010	0.01684	0.00025	0.00002
100k	2x2x2	0.00010	0.16177	0.00245	0.00004
1,000k	2x2x2	0.00010	1.69940	0.02418	0.00004
1,000k	2x2x2	0.00100	1.75740	0.02486	0.00004
1,000k	2x2x2	0.01000	1.71279	0.02499	0.00004
1,000k	2x2x2	0.05000	1.74756	0.02455	0.00006
1,000k	2x2x2	0.10000	1.75000	0.06250	0.00023
10,000k	2x2x2	0.05000	24.89122	0.35404	0.00034
10,000k	2x2x2	0.10000	24.75000	0.30457	0.00117
10,000k	2x2x2	0.20000	23.87500	0.31250	0.00816
10,000k	2x2x2	0.30000	23.06250	0.38333	0.02849
10,000k	2x2x2	0.40000	23.31250	0.31250	0.06252
10,000k	2x2x2	0.50000	23.56250	0.37500	0.09375
10,000k	2x2x2	0.60000	23.25000	0.34375	0.21875
10,000k	2x2x2	0.70000	22.43255	0.37500	0.31256
10,000k	2x2x2	0.80000	23.87656	0.31250	0.37501
10,000k	2x2x2	0.90000	23.43750	0.37500	0.56250
10,000k	2x2x2	1.00000	23.40649	0.31943	0.87899

Tabla 6 Muestra los resultados para pruebas de tiempo de inserción de puntos, búsqueda por fuerza bruta y búsqueda usando el Octree, para diferente cantidad de puntos y diferente radio de búsqueda.

En la Tabla 6 de resultados también se puede observar que la búsqueda de puntos por Octree ya no es funcional cuando el diámetro de búsqueda es de .8 unidades (40% el largo del Octree), ya que el tiempo de búsqueda por Octree es entonces mayor que el tiempo de búsqueda por Fuerza Bruta. Esto se marcó con rojo, por tanto, todos los valores anteriores son factibles.

Así mismo, se obtiene que el mejor tiempo de inserción, cuando se insertan mil puntos al modelo, es entre 0.00093 y 0.00098 segundos, mientras que para 10 mil puntos se obtuvo un tiempo de 0.01684, 10 veces más al insertar 100 mil puntos, y 100 veces más con 1 millón de puntos. Este tiempo es mejor que el obtenido anteriormente, pero aún no es óptimo.

5.3 Deformación

5.3.1 Entrenamiento usando Octree

Una de las pruebas para validar que el modelo de Octree propuesto funciona correctamente es aplicando una deformación; por tanto, en este punto se explicará la deformación que se realizará para los modelos y se expone un ejemplo.

Con base en el modelo de puntos anteriormente trabajado, utilizando el visualizador, se programó el ejercicio de búsqueda de puntos. Para esto se seleccionó un punto específico en la semiesfera y se aplicó una deformación simple que radica en restar a la coordenada z un valor constante k , y posteriormente recargar escena (para que se note el cambio en la visualización), el resultado puede observarse en la Figura 25.

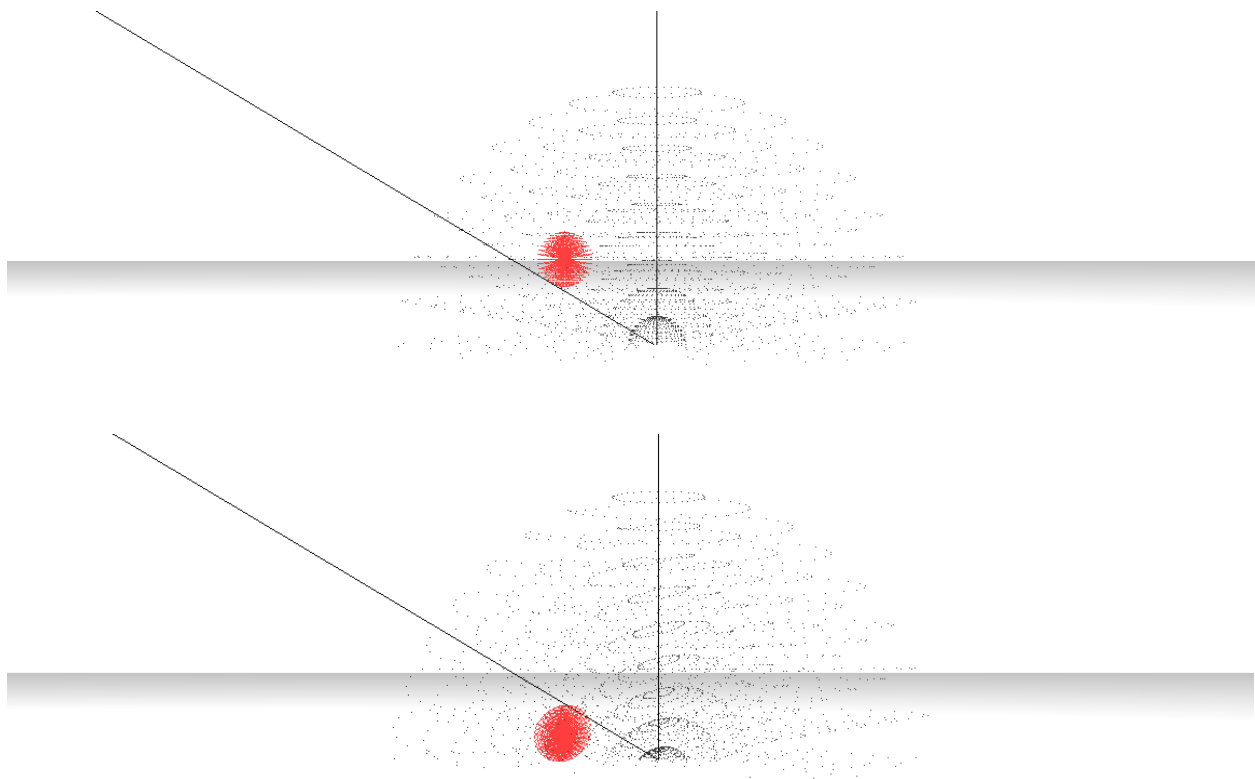


Figura 25 Arriba, semiesfera con esfera contenida en su interior en estado inicial, sin deformación aplicada. Abajo, el mismo modelo después de aplicar una deformación simple, sólo a una sección de la parte izquierda de la semiesfera; puede observarse que del lado derecho no hay cambio, y del lado izquierdo sí.

Esta prueba simple cumple con el objetivo de validar que las funciones realizan su trabajo principal (en este caso aplicando modificación a los valores en el eje z): por un lado, la correcta localización de los puntos en el volumen, por otro, verificar que el modelo acepta una fórmula para ejercer una deformación en el volumen propuesto.

5.3.2 Preparación de la deformación de un objeto real

Con el objetivo de obtener mejores resultados y probar el modelo de Octree en un entorno más realista, el Laboratorio de Imagenología Biomédica del Centro de Ciencia Aplicada y Desarrollo Tecnológico (CCAD-UNAM) creó maniqués hechos a base de alcohol polivinílico [45], intentando simular la forma y propiedades físicas básicas de una mama humana (Figura 26). También, durante la creación de los maniqués, se les introdujo una uva para simular un tumor; el resultado se puede observar en la imagen.

El diámetro de los maniqués obtenidos en promedio es de 10 cm por 4 cm de altura, la uva tenía un aproximado de 1.5 cm de altura por 1 cm de diámetro al momento de ser insertada en el volumen.



Figura 26 Volumen a base de alcohol polivinílico para simular una mama, que por dentro contiene una uva para simular un tumor.

Posteriormente, intentando simular un escenario de estudio médico, el maniquí se escaneó por medio de un equipo de ultrasonido ordinario y durante el proceso se obtuvieron los datos que se van generando al mover el transductor, que a su vez son imágenes en serie, a las que se les aplicó un proceso de generación de volumen tridimensional por computadora, obteniendo como resultado final dos archivos, uno extensión mhd, el otro extensión zraw (tipos de archivo para manejo de imágenes médicas). Estos archivos contienen la información suficiente para su visualización y edición, para lo cual se utilizó nuevamente el software 3D Slicer 4.4.0.

Una vez cargados ambos archivos en el software, el siguiente paso fue realizar una segmentación detallada del volumen obtenido con el objetivo de separar la uva, la mama, y el exterior, para así posteriormente asignarles valores físicos diferentes y aplicar más tarde una deformación en los dos primeros.

La primera parte de la segmentación, que es aislar la uva que asemeja el tumor, se hizo utilizando la herramienta *Label Tracing* de 3D Slicer, la cual funciona básicamente como segmentación por crecimiento regiones (region growing) a partir de inserción de semillas. Esto es, se selecciona manualmente un punto en la imagen (inserción de semilla), para después comparar en automático si los puntos vecinos cumplen con un umbral α determinado a priori, y en caso de que sí suceda, el proceso absorbe los pixeles coincidentes (los anexa a la segmentación) y realiza el mismo proceso iterativo para los puntos absorbidos, lo cual se ejecuta hasta que los puntos vecinos ya no coinciden con el umbral α , y una vez que el proceso ya no encuentra coincidentes en ninguno de los frentes por los cuales se ha extendido, la segmentación se detiene. Así también, es posible sembrar el número de semillas que se considere necesario.

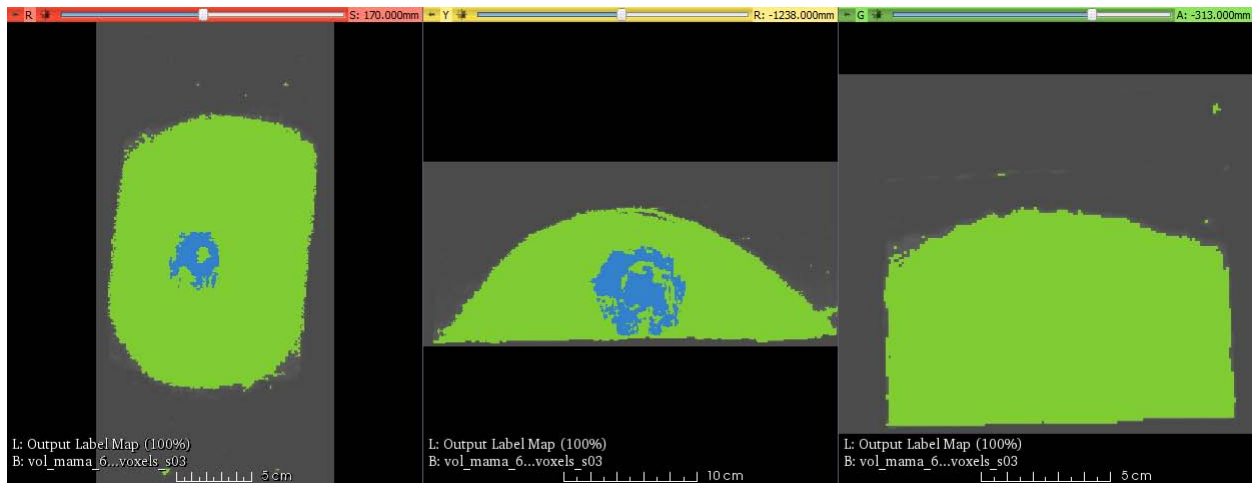


Figura 27 Segmentación para separar la uva, que simula un tumor, del volumen que simula una mama utilizando 3D Slicer.

El criterio utilizado para aislar la uva en las imágenes obtenidas con el ultrasonido fue ubicar las partes más oscuras en la serie de imágenes, apoyado también en la forma de dicho objeto; posteriormente, la segunda etapa fue la segmentación de la mama, que de acuerdo a las imágenes obtenidas, no se mostraba tan oscura como la uva y la parte vacía, esto quiere decir que cuenta con características físicas diferentes.

Para separar la mama del tumor se utilizó un método simple de segmentación por umbral, en el cual, como se ha comentado anteriormente, es un método que ubica los píxeles de una imagen que se encuentran entre ciertos valores previamente delimitados (umbral), lo cual facilitó el trabajo, aprovechando además que la serie de imágenes obtenidas se encontraban definidas entre valores que van del 0 a 255, de donde 0 representaba nuevamente el color negro y 255 el blanco.

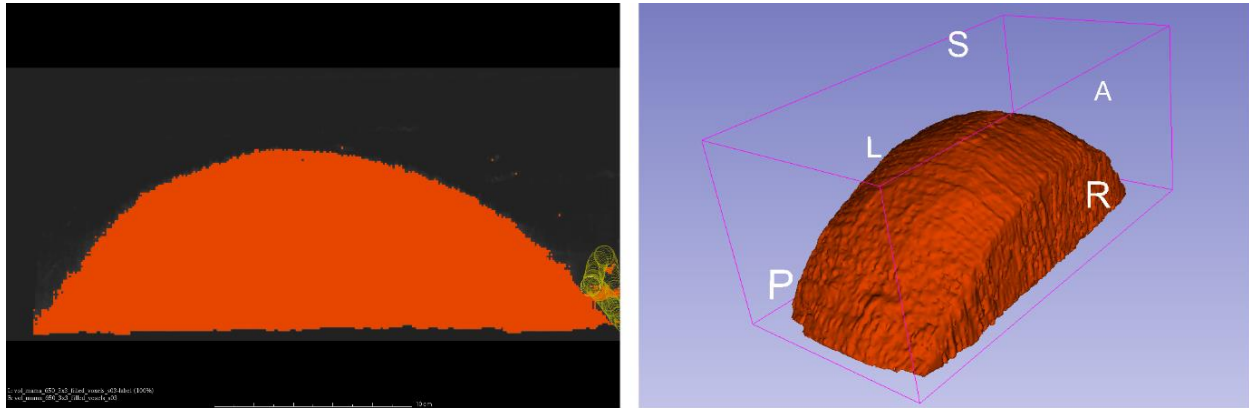


Figura 28 A la izquierda se observa la eliminación de píxeles para refinar el detalle del modelo volumétrico segmentado, a la derecha el resultado final posterior a los procesos de segmentación.

Posteriormente se realizaron correcciones menores eliminando algunos píxeles que se encontraban fuera de los volúmenes segmentados, lo cual se puede observar en la Figura 28.

Para cumplir con esa tarea, se utilizó la herramienta *Paint* y *Eraser* de 3D Slicer en cada imagen. El objetivo de la herramienta *Paint* es seleccionar píxeles específicos para pintarlos de un color. Esta herramienta funciona básicamente seleccionando un radio constante para la brocha (en milímetros), y seleccionando posteriormente lo que se quiere *pintar*; pero si además se selecciona la herramienta *Eraser*, cuyo objetivo básico es borrar píxeles, la función conjunta elimina los píxeles seleccionados con la herramienta *Paint*.

Con lo anterior se logró exitosamente tener una segmentación más limpia en cuanto a detalles se refiere.

Por último, se realizó una unión de la segmentación de la mama con la uva, utilizando el mismo software, y en la Figura 28, a la derecha, se puede observar el resultado final de la segmentación; así también, en la Figura 29 se observa a la izquierda el resultado final de la uva segmentada, y a la derecha, la misma uva pero con un mallado creado para su visualización posterior.

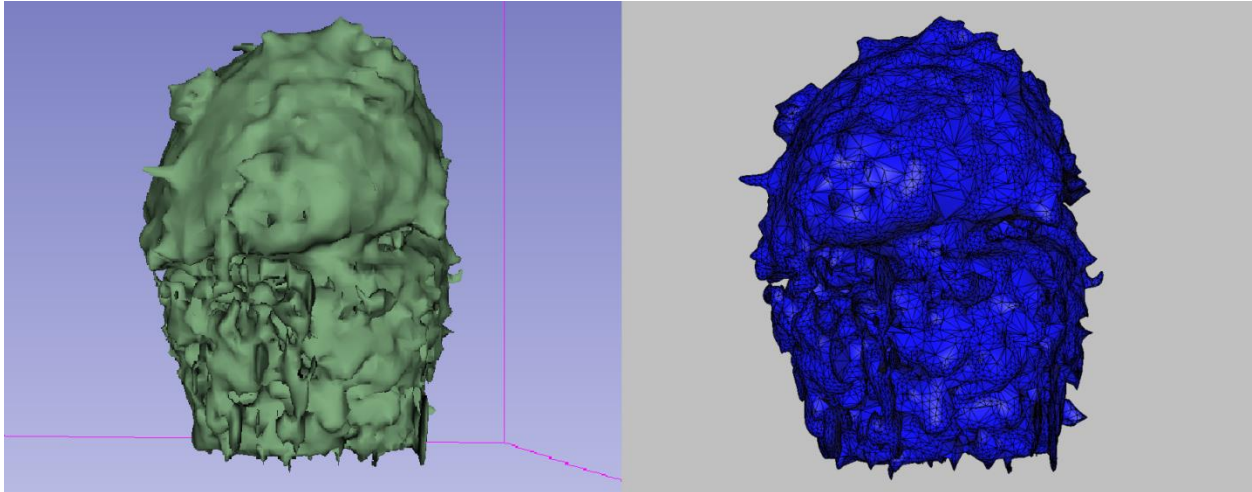


Figura 29 Izquierda, volumen de la uva asemejando un tumor segmentado en 3D Slicer. Derecha, volumen con malla.

Una vez terminado todo el procedimiento de segmentación y unión en 3D Slicer, se exportaron los datos a un solo archivo de texto con el objetivo de darle el formato requerido para su lectura en el modelo de deformación por Octree.

Con los volúmenes segmentados, utilizando 3D Slicer, para cada uno, seno y tumor, se creó una malla volumétrica a base de tetraedros (Figura 29), en la cual, el software también permite extraer las coordenadas de los tres vértices de cada triángulo de la malla, con lo que se pudo obtener las normales utilizando el software 3D Object Converter 4.8. Para mejorar la iluminación, se seleccionó también un color de prueba con base en tonos que reflejaran contraste y cierto parecido al de la piel humana, y otro diferente para el tumor.

Por último, la obtención del módulo de Young que se usó para cada elemento (seno y tumor) se explica a continuación. Con base en los estudios revisados en el Capítulo 3 sobre el Módulo de Young, se obtuvieron los resultados mostrados en la Tabla 7 originados de los siguientes trabajos de investigación:

- Trabajo 1. Elastic Moduli of Breast and Prostate Tissues Under Compression (1999) [\[34\]](#).
- Trabajo 2. Shear-wave Elastography Improves the Specificity of Breast US: The BE1 Multinational Study of 939 Masses (2012) [\[35\]](#).
- Estudio 3. Shear-wave elastography for breast masses: local shear wave speed (m/sec) versus Young modulus (kPa) (2014) [\[36\]](#).
- Estudio 4. Factors influencing breast elasticity measured by the ultrasound Shear Wave elastography – preliminary results (2010) [\[37\]](#).

Tipo de tejido mamario	Estudio 1			Estudio 2 Elasticidad	Estudio 3		Estudio 4 E_mean
	Masas	5% comp 1Hz	20% comp 1Hz		Masas	E_IQR	
Graso	8	19 ± 7	20 ± 6	9.24 ± 4.48			
Glandular	31	33 ± 11	57 ± 19	11.28 ± 5.79			
Fibroso	18	107 ± 31	232 ± 60				
Carcinoma ductal in situ	23	25 ± 4	301 ± 58		12	137-180	
Carcinoma ductal Invasivo e infiltrant	32	93 ± 33	490 ± 112		271	143-180	140.7 ± 58.5
Otro maligno					6	137-180	
Fibroadenoma (benigno)					179	28-73	34.8 ± 17.7
Otro benigno					351	25-90	
Quiste					104	06-53	
Lesiones de riesgo					17	30-165	
Absceso					2	87-180	

Tabla 7 Comparativa de estudios que obtienen la medida de elasticidad del tejido mamario.

De ahí que, con el objetivo de probar el modelo de Octree para crear una deformación diferenciando las características físicas entre el tumor y la mama, se utilizaron los Módulos de Young 11.28kPa para la mama (tejido graso y glandular), y 140.7kPa para el tumor, esto de acuerdo a la fórmula básica de la Ley de Hooke:

$$\delta = -\frac{F}{E}$$

De donde F es la fuerza aplicada en el punto de impacto, E es el Módulo de Young y δ es el desplazamiento que experimenta el material, por tanto, el desplazamiento para el punto de impacto es medido en función de la fuerza ejercida sobre el Módulo de Young específico de cada material.

Ahora, para los puntos contenidos dentro del radio de impacto, el desplazamiento no es el mismo, sino que se pondera de acuerdo a la distancia existente entre cada punto contenido y el punto de impacto. El detalle de los elementos para calcular el desplazamiento resultado puede observarse en la Figura 30.

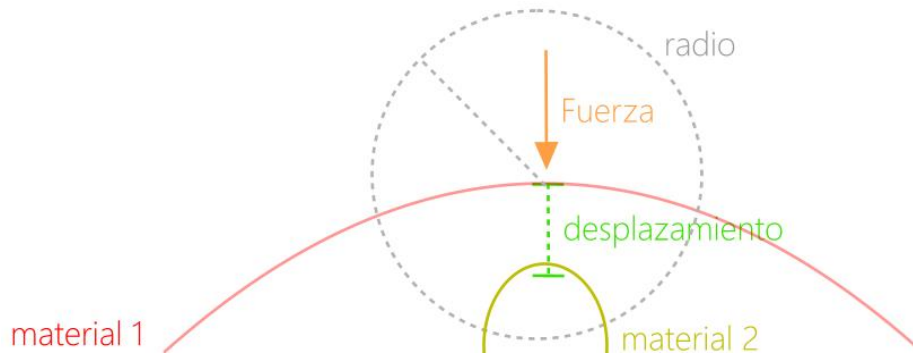


Figura 30 Partes utilizadas del modelo para simular la elasticidad.

La fórmula utilizada para la deformación de cada punto es la siguiente:

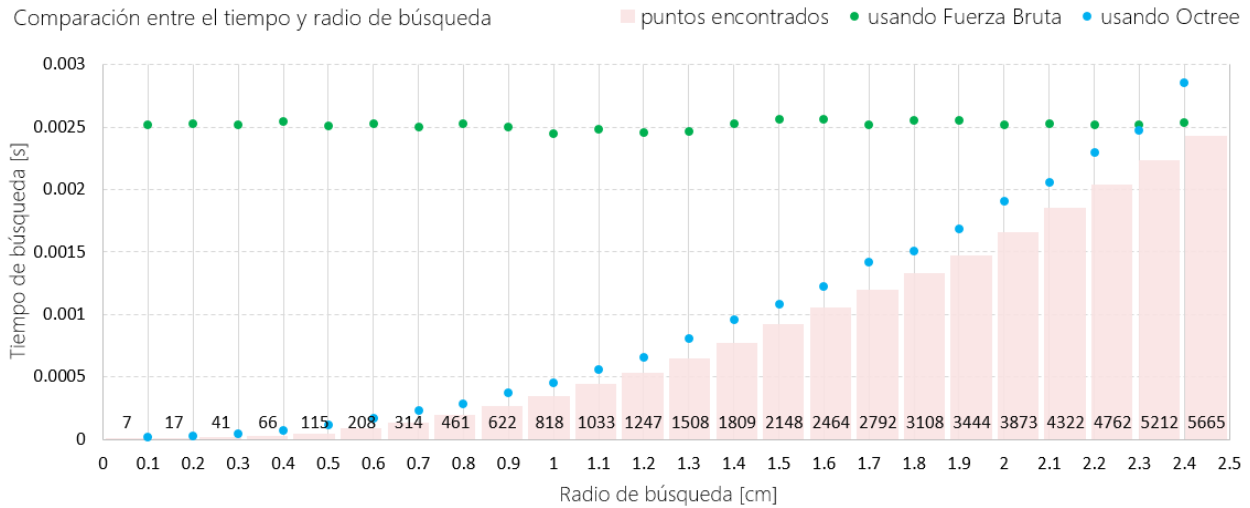
$$D = \left(1 - \frac{d}{r}\right) * \delta$$

En donde d es la distancia entre cada punto evaluado y el punto de impacto, r es el radio de impacto con origen en el punto de impacto, δ es el desplazamiento que sufre el punto de impacto (resultado de dividir la Fuerza aplicada entre el Módulo de Young del material), y D es el desplazamiento para el punto evaluado.

Hasta este instante se ha trabajado con un comportamiento plástico, es decir, se mantiene la deformación aplicada al objeto y no regresa a su forma original; ahora, para obtener la propiedad de elasticidad, es decir, que el objeto regrese a su forma original, se aplica el mismo principio físico pero en sentido contrario, es decir, se aplica la misma fuerza en el punto de impacto pero cambiando la dirección.

5.3.3 Resultados de la deformación elástica

Para validar que es posible lograr una deformación utilizando el modelo de Octree sobre el volumen basado en el maniquí y utilizando la fórmula comentada en el punto anterior, se hizo una comparación contra la búsqueda por Fuerza Bruta, los datos pueden verse en la Gráfica 2.



Gráfica 2 Resultado de comparar el tiempo que tarda una búsqueda entre los algoritmos de Fuerza Bruta y el que usa el Octree al aumentar el radio de búsqueda de puntos.

El método de Fuerza Bruta toma uno por uno los elementos del conjunto de datos y los compara con todo el resto; es un método trivial pero comúnmente usado, además de funcional, para comparar la eficiencia de la búsqueda por Octree que se está proponiendo en este trabajo.

En la Gráfica 2 se puede observar a detalle la comparación entre el método de búsqueda por Fuerza Bruta (verde) y el método propuesto basado en Octree (azul), donde para el primero es predecible que tenga un comportamiento prácticamente constante en cuanto a tiempo de ejecución refiere, en cambio, la búsqueda basada en Octree demuestra claramente cómo para radios de búsqueda pequeños, su desempeño en tiempo de ejecución es mucho mejor, aunque conforme va en aumento el radio, también lo hace el tiempo de búsqueda, tanto que para tiempos mayores a 2.3 segundos el algoritmo que implementa Octree ya no es funcional para este caso.

El método basado en Octree demuestra que, para radios de búsqueda relativamente pequeños, es varias veces más funcional, dado que para un radio de **0.01cm** el tiempo es apenas 0.0077 veces el tiempo de búsqueda por Fuerza Bruta. Por otro lado, la mitad del tiempo consumido por el algoritmo de búsqueda por Fuerza Bruta se presenta cuando se alcanza un radio de **0.16cm**; y justo a los **0.23cm** se cumple el 98% de relación (ver Tabla 8), siendo este el umbral máximo de utilidad de la búsqueda por Octree para el objeto utilizado.

radio [cm]	tiempo lectura [s]	niveles del octree	puntos insertados	tiempo creación octree [s]	puntos encontrados	tiempo usando Fuerza Bruta [s]	tiempo usando Octree [s]	octree/fuerza bruta
0.01	5.88687	40	261,255	0.2519854	7	0.002518	0.0000194	0.77%
0.02	5.89367	40	261,255	0.2479528	17	0.002526	0.0000326	1.29%
0.03	5.88843	40	261,255	0.2493542	41	0.002522	0.0000456	1.81%
0.04	5.920482	40	261,255	0.246825	66	0.002546	0.000073	2.87%
0.05	5.922844	40	261,255	0.2474758	115	0.00251	0.000113	4.50%
0.06	5.928128	40	261,255	0.2497388	208	0.002528	0.000168	6.65%
0.07	5.902268	40	261,255	0.2462312	314	0.002498	0.0002288	9.16%
0.08	5.92115	40	261,255	0.2478574	461	0.00253	0.0002886	11.41%
0.09	5.902808	40	261,255	0.2487534	622	0.002502	0.0003722	14.88%
0.10	5.911668	40	261,255	0.25816	818	0.002444	0.0004528	18.53%
0.11	5.918088	40	261,255	0.248393	1033	0.00248	0.0005622	22.67%
0.12	5.91621	40	261,255	0.248851	1247	0.002458	0.0006622	26.94%
0.13	5.921004	40	261,255	0.24434	1508	0.002462	0.0008046	32.68%
0.14	5.90167	40	261,255	0.263306	1809	0.002532	0.00096234	38.01%
0.15	5.86951	40	261,255	0.2519532	2148	0.002562	0.0010858	42.38%
0.16	5.88922	40	261,255	0.2498428	2464	0.002562	0.001221	47.66%
0.17	5.92554	40	261,255	0.20450652	2792	0.002522	0.0014172	56.19%
0.18	5.895038	40	261,255	0.2504068	3108	0.002554	0.0015088	59.08%
0.19	5.89887	40	261,255	0.2477598	3444	0.002552	0.0016854	66.04%
0.20	5.878826	40	261,255	0.2492216	3873	0.002522	0.0019056	75.56%
0.21	5.887334	40	261,255	0.251031	4322	0.002532	0.0020608	81.39%
0.22	5.87791	40	261,255	0.2477802	4762	0.002522	0.002298	91.12%
0.23	5.92326	40	261,255	0.247104	5212	0.002516	0.0024708	98.20%
0.24	5.924264	40	261,255	0.249929	5665	0.002534	0.0028528	112.58%
Promedio	5.9043776			0.247864955		0.002518083		

Tabla 8 Resultados al implementar el modelo basado en Octree en comparación con el de Fuerza Bruta.

Otro punto a notar es que tomó en promedio 5.9 segundos leer los 261,255 puntos, además se han creado 40 niveles para albergar a todos los datos; por otro lado, el tiempo promedio de creación del Octree es de 0.2478 segundos, independientemente del radio de búsqueda, como se muestra en la Tabla 8.

Para mejor apreciación del resultado, se programó una vista por colores para ver la deformación del área impactada, de donde, como se ha explicado anteriormente, entre más cerca estén los puntos involucrados dentro del área de impacto, mayor es la deformación:

- Rojo – 30% de distancia entre el punto de impacto y el radio total.
- Azul – De 30% a 60% de distancia entre el punto de impacto y el radio total.
- Amarillo – De 60% a 80% de distancia entre el punto de impacto y el radio total.
- Verde Claro – De 80% a 100% de distancia entre el punto de impacto y el radio total.
- Verde Oscuro – Sin impacto ni deformación.

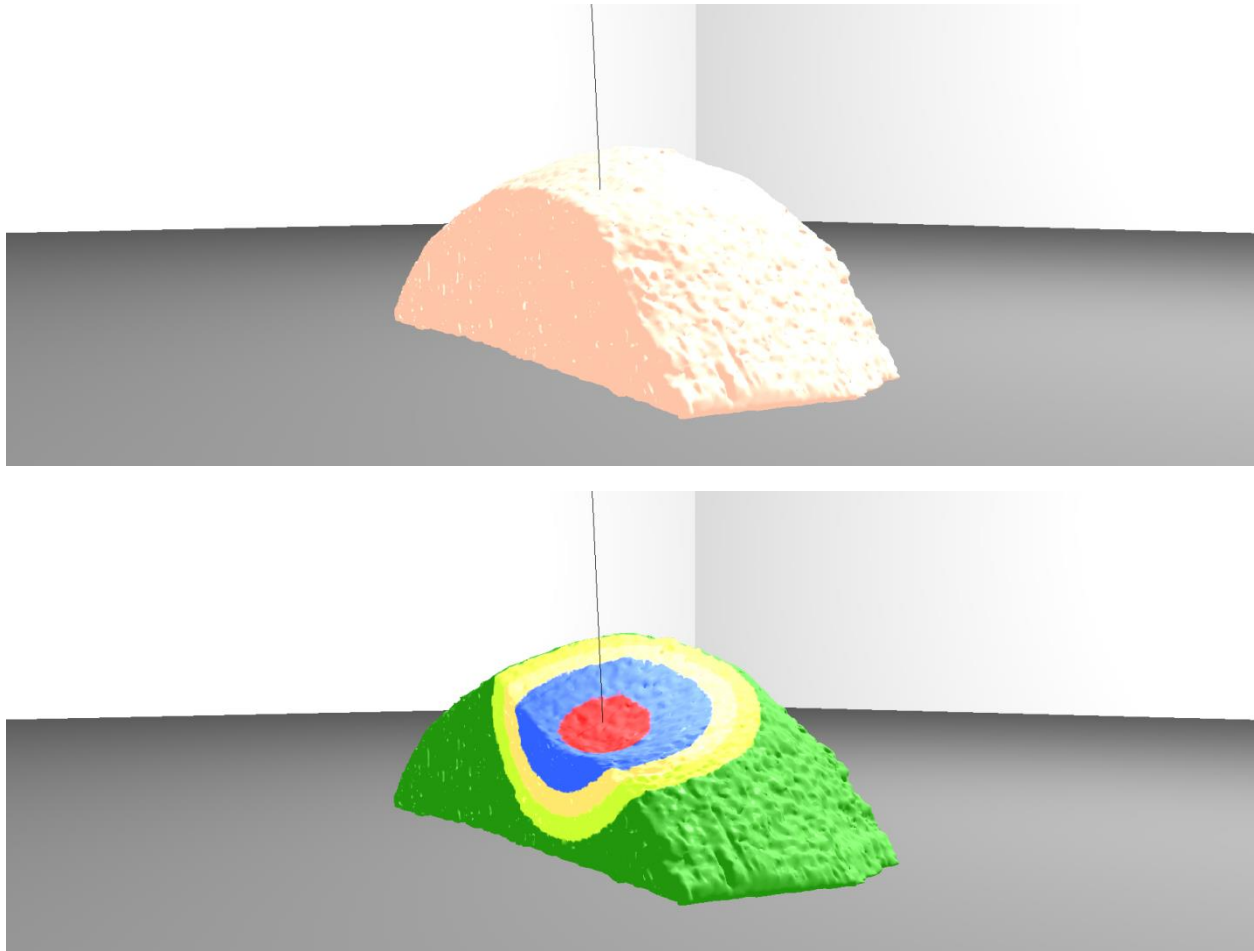


Figura 31 Arriba, visualización del modelo sin deformación. Abajo, visualización del modelo con deformación aplicada y colores para verificar las áreas de mayor impacto.

En la Figura 31, arriba, se observa el volumen en estado inicial, antes de ser deformado; posteriormente se selecciona el punto de impacto y al presionar el botón izquierdo del mouse, se observa el resultado de la deformación en la imagen inferior.

Otro detalle que no se observa en la Figura 31 es la visualización del tumor simulado por una uva, el cual se encuentra al interior de la mama y para verificar su deformación, el modelo de visualización permite entrar al volumen y observar su interior.

Ahora, de acuerdo a la escala de color comentada anteriormente, la mayor afectación se presenta en el área de color rojo y azul (de 0 a 60%), en cambio, en color amarillo y verde claro (de 60% a 100%), es mucho menor. El área que no sufre impacto y por tanto no es modificada es la verde fuerte.

Además, se realizó un ejercicio para verificar que la deformación en el tumor es menor a la obtenida de la mama, lo cual es debido a que, como se comentó anteriormente, el módulo del tumor es mayor al del tejido mamario, lo anterior puede observarse en la Figura 32.

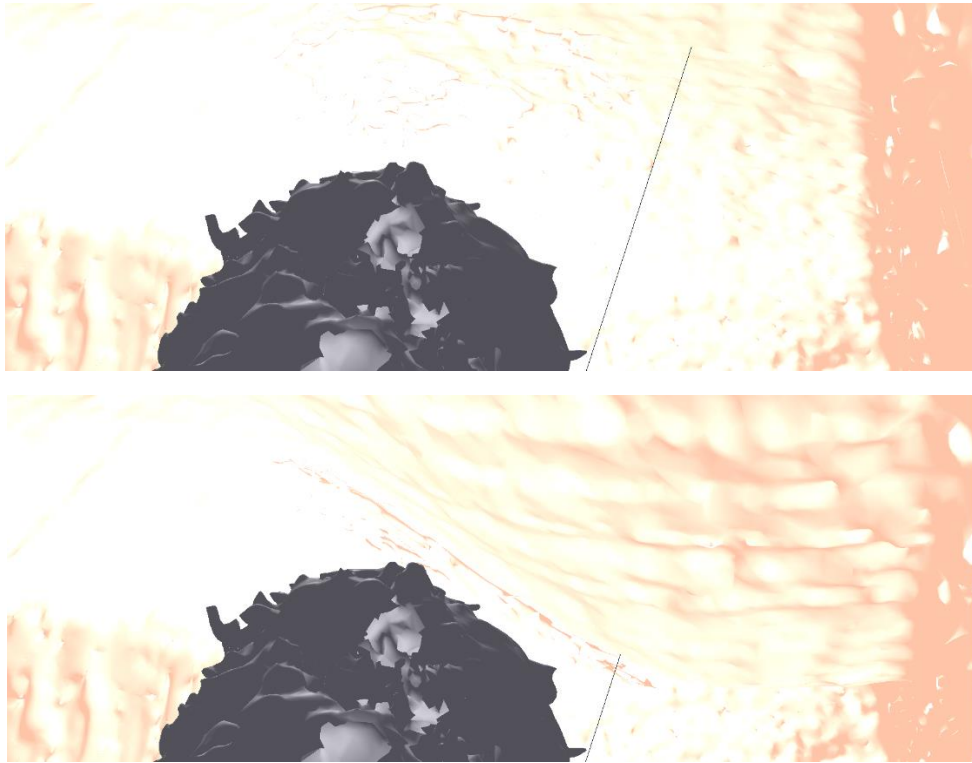


Figura 32 Arriba, visualización sin deformación del interior de la mama, donde también puede observarse el tumor. Abajo, visualización del modelo con deformación aplicada donde se aprecia mayor deformación en la mama.

En la simulación capturada en las imágenes de la Figura 32, arriba se observa el tumor (color gris oscuro) dentro del interior de la mama, en estado inicial, sin aplicar ninguna deformación; en cambio, abajo es posible examinar la deformación para ambos órganos. Así mismo, puede apreciarse cómo para el tejido mamario la deformación es mayor que para el tumor, lo cual representa el resultado al que se quería llegar, considerando que las propiedades físicas de los materiales son diferentes.

Así también, en la Figura 33 se hizo el mismo ejercicio, pero aplicando la vista de impacto por color, donde arriba se puede observar el escenario sin aplicar modificaciones, y abajo se detallan los volúmenes deformados, con la diferencia de que en azul se aprecia la zona de mayor impacto en el tumor, el cual es mucho menor comparado al área impactada en la mama, donde incluso se observa el color rojo.

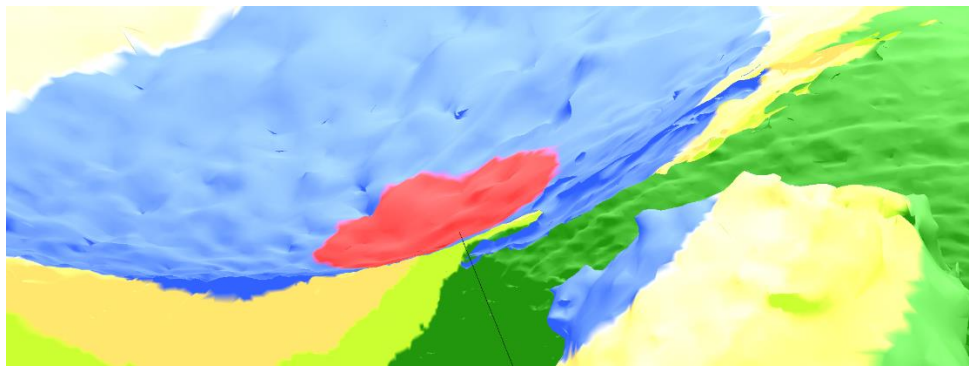
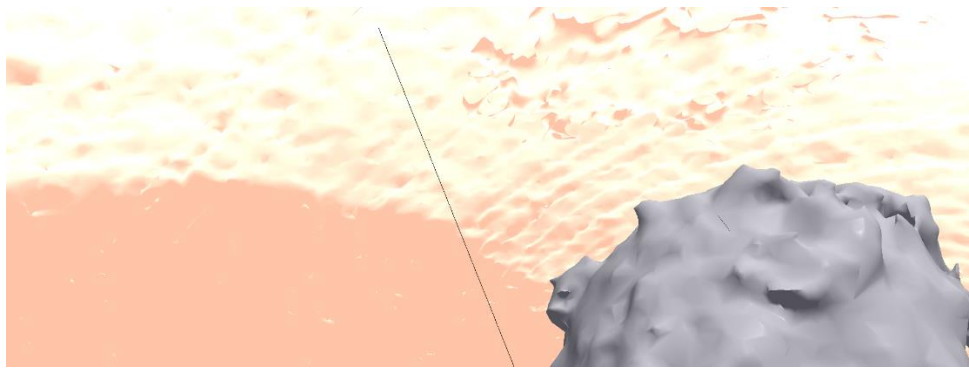


Figura 33 Arriba, visualización del modelo sin deformación con vista al interior de la mama. Abajo, visualización del modelo con deformación aplicada y colores para verificar las áreas de mayor impacto.

En las Figuras 31, 32 y 33 es posible apreciar el resultado final, que consta del volumen trabajado aplicando una deformación y utilizando el método de búsqueda por Octree, trabajando dos diferentes materiales en una misma escena, y aplicando también el método de visualización diseñado.

Además, no es posible apreciarlo en las imágenes, pero se obtuvo la elasticidad del material al regresar el volumen a su posición original. El programa también permite un comportamiento plástico al obtener deformaciones permanentes.

5.4 Conclusiones

Se logró deformar un volumen que simula ser una mama con un tumor en su interior, dicho volumen virtual fue obtenido de un objeto real constituido por dos partes con propiedades físicas diferentes; cada parte se deformó de forma diferente, de acuerdo a su Módulo de Young específico. Se mostró que dicho volumen se deforma y regresa a su forma original, simulando la propiedad de elasticidad.

Aunque se aplicó una fórmula básica para simular la propiedad de la elasticidad del volumen reconstruido, el método está preparado para implementar fórmulas más complejas, dado que acepta el manejo de más variables.

Replicar un escenario real en un ambiente virtual requiere de un amplio estudio de las características de cada objeto; los colores, las formas, las propiedades físicas, la relación que existe entre ellos mismos, etc., cada parte debe tomarse en cuenta para crear un escenario más realista.

El método de búsqueda de puntos para deformar utilizando el Octree demostró ser factible para radios pequeños, y mejor comparado al método de fuerza bruta; aunque conforme se aumenta el radio de búsqueda para delimitar el área deformada, el método ya no es tan efectivo.

La velocidad de navegación en la escena virtual utilizando OpenGLSL es casi inmediata una vez que se leen todos los puntos. El método de iluminación funciona correctamente y fue posible dar a ambos tipos de tejido propiedades físicas diferentes.

El escenario virtual se ejecutó en un equipo laptop similar al que muchas personas tienen en su casa actualmente, lo que demuestra que se requieren pocos recursos para simular una deformación con respuesta rápida y características reales.

El modelo puede simular la deformación de cualquier objeto, si se tienen los vértices (coordenadas xyz) y se definen sus propiedades físicas; tal deformación depende directamente de las características específicas de cada objeto.

5.5 Trabajo a futuro

Como trabajo a futuro, uno de los puntos más interesantes es acelerar los procesos de búsqueda utilizando paralelización mediante GPU, dado que en este trabajo sólo se utilizó para la graficación del volumen.

Para este trabajo, la deformación física se delimitó dentro de la zona elástica (Ver la Gráfica 1) porque el objetivo de la tesis estaba enfocado en aplicar el uso del Octree para la búsqueda de puntos, más que en simular un comportamiento físico totalmente realista (aunque también se obtuvo un comportamiento plástico, que es cuando el volumen no regresa a su forma original y mantiene la deformación), por tanto, como trabajo a futuro se propone aplicar fórmulas más completas para incluir todo el rango del comportamiento físico.

Es posible trabajar en simular la ruptura del volumen agregando propiedades físicas y programando un remallado rápido, dado que en este trabajo sólo se calcula la malla una vez, pero en una ruptura del volumen la malla también se separa, por lo que habría que calcularla de nuevo.

Es posible también integrar este modelo a otros proyectos, sobre todo los que tienen que ver con simulaciones médicas virtuales, pero también para simulación de materiales en general, si es que se tienen sus propiedades físicas.

Bibliografía

- [1] COMSOL Inc. Multiphysics Software Product Suite. Fecha de consulta 6 Junio 2015.
<http://www.comsol.com/>
- [2] ThoroughTec Technologies 2010. ThoroughTec Simulation. Fecha de consulta 6 Junio 2015.
<http://www.thoroughtec.com/>
- [3] SimSurgery AS, surgical simulation. Fecha de consulta 6 Junio 2015.
<http://www.simsurgery.com/>
- [4] Surgical Science 2015 LapSim Virtual Training. Fecha de consulta 6 Junio 2015.
<http://www.surgical-science.com/>
- [5] S. Li, W. K. Liu, "Meshfree and particle methods and their applications", American Society of Mechanical Engineers, *ASME Reprint No AMR319, Appl Mech Rev* Vol. 55, No. 1, January 2002.
- [6] T. Vassilev, B. Spanlang, "A Mass Spring Model for Real Time Deformable Solids", *Proceedings of East-West Vision 2002*, Graz, Austria, September 12-13, pp. 149-154, 2002.
- [7] S. Rau, "The Finite Element Method in Engineering", Elsevier Butterworth-Heinemann, 4th Edition, pp. 3-13, 2005.
- [8] S. A. Viana, D. Rodger, H. C. Lai, "Overview of Meshless Methods". International Compumag Society Newsletter, v. 14, pp. 3-6, 2007.
- [9] T. Fries, H. Matthies, "Classification and Overview of Meshfree Methods". *Technical Report* 2003-3. Institute of Scientific Computing, Technical University Braunschweig, Brunswick, Germany. July, pp. 4-6, 2004.
- [10] F. G. Flores, A. T. Brewer, "Introducción al Método de los Elementos Finitos". *Universidad Nacional de Córdoba, Facultad de Ciencias Exactas, Físicas y Naturales, Departamento de Estructuras*. pp. 80, 2002.
- [11] T. Fries, H. Matthies, "Classification and Overview of Meshfree Methods". *Technical Report* 2003-3. Institute of Scientific Computing, Technical University Braunschweig, Brunswick, Germany. July, pp. 52-77, 2004.
- [12] J. Elseberg, D. Borrmann, A. Nüchter, "One billion points in the cloud – an octree for efficient processing of 3D laser scans". *ISPRS Journal of Photogrammetry & Remote Sensing (JPRS)*, Special issue on terrestrial 3D modelling, Elsevier Science, Volume 76, pp. 76-88, ISSN 0924-2716, February 2013.

- [13] X. Guo, H. Qin, "Real-time meshless deformation", *Computer Animation and Virtual Worlds*, Vol. 16, Num. 3 & 4, pp. 189–200, July, 2005.
- [14] R. Đurikovič, M. Kupka, "VOF Method for Fluids and Solids on Octree Structure", *IEEE VIZ '09 Proceedings of the 2009 Second International Conference in Visualisation*, pp. 76-81, 2009.
- [15] L. Wu, Y. Hori, "Real-time Collision-free Path Planning for Robot Manipulator Based on Octree Model". *Proceedings of the 9th IEEE International Workshop on Advanced Motion Control*, pp. 284-288, 2006.
- [16] W. Song, S. Hua, Z. Ou, H. An, K. Song, "Octree Based Representation and Volume Rendering of Three-Dimensional Medical Data Sets". *Proceedings of the International Conference on BioMedical Engineering and Informatics (BMEI '08)*, vol. 1, pp. 316–320, May 2008.
- [17] H. T. Yau, L. S. Tsou, M. J. Tsai, "Octree-based Virtual Dental Training System with Haptic Device" *Computer-Aided Design & Applications*, 3(1-4), pp. 415-424, 2006.
- [18] F. WenShan, W. Bin, P. Jean-Claude, S. JiaGuang, "An octree-based proxy for collision detection in large-scale particle systems". *SCIENCE CHINA Information Sciences 2013*, Vol. 56 Issue: 012104(10), 2013.
- [19] Cáncer de seno (mama): Tratamiento (PDQ®), Información general sobre el cáncer de seno (mama). National Cancer Institute, National Institutes of Health, USA. Fecha de consulta 6 Junio 2015.
<http://www.cancer.gov/espanol/pdq/tratamiento/seno/Patient/page1>
- [20] J. B. Kampert, A. S. Whittemore, R. S. Jr. Paffenbarger, "Combined effect of childbearing, menstrual events, and body size on age-specific breast cancer risk." *Am J Epidemiol* 128 (5), pp. 962-79, 1988.
- [21] S. F. Altekruse, C. L. Kosary, M. Krapcho, et al.: "SEER Cancer Statistics Review, 1975-2007" Bethesda, Md: National Cancer Institute, 2010.
- [22] J. Kotsopoulos, W. Y. Chen, M. A. Gates, et al., "Risk factors for ductal and lobular breast cancer: results from the nurses' health study." *Breast Cancer Res* 12 (6): R106, 2010.
- [23] G. A. Colditz, B. A. Rosner, F. E. Speizer, "Risk factors for breast cancer according to family history of breast cancer. For the Nurses' Health Study Research Group." *J Natl Cancer Inst* 88 (6), pp. 365-71, 1996.
- [24] P. J. Goodwin, K. A. Phillips, D. W. West, et al., "Breast cancer prognosis in BRCA1 and BRCA2 mutation carriers: an International Prospective Breast Cancer Family Registry population-based cohort study." *J Clin Oncol* 30 (1), pp. 19-26, 2012.

- [25] N. Andrieu, D. F. Easton, J. Chang-Claude, et al., "Effect of chest X-rays on the risk of breast cancer among BRCA1/2 mutation carriers in the international BRCA1/2 carrier cohort study: a report from the EMBRACE, GENEPSO, GEO-HEBON, and IBCCS Collaborators" *Group. J Clin Oncol* 24 (21): 3361-6, 2006.
- [26] N. F. Boyd, L. J. Martin, J. M. Rommens, et al., "Mammographic density: a heritable risk factor for breast cancer. " *Methods Mol Biol* 472: 343-60, 2009.
- [27] T. J. Key, P. N. Appleby, G. K. Reeves, et al., "Circulating sex hormones and breast cancer risk factors in postmenopausal women: reanalysis of 13 studies. " *Br J Cancer* 105 (5): 709-22, 2011.
- [28] R. N. Hoover, M. Hyer, R. M. Pfeiffer, et al. "Adverse health outcomes in women exposed in utero to diethylstilbestrol" *New England Journal of Medicine* 2011; published online October 6, 2011.
- [29] K. Y. Wolin, K. Carson, G. A. Colditz, "Obesity and cancer", *Oncologist* 15 (6): 556-65, 2010.
- [30] L. Bernstein, B. E. Henderson, R. Hanisch, et al., "Physical exercise and reduced risk of breast cancer in young women", *J Natl Cancer Inst* 86 (18): 1403-8, 1994.
- [31] N. Hamajima, K. Hirose, K. Tajima, et al., "Alcohol, tobacco and breast cancer--collaborative reanalysis of individual data from 53 epidemiological studies, including 58,515 women with breast cancer and 95,067 women without the disease", *Br J Cancer* 87 (11): 1234-45, 2002.
- [32] H. Razzaghi, M. A. Troester, G. L. Gierach, et al., "Mammographic density and breast cancer risk in White and African American Women", *Breast Cancer Res Treat* 135 (2): 571-80, 2012.
- [33] "Steelworker", Volume 1 – Navedtra 14250 *Nonresident Training Course*, pp. 4, November 1996.
- [34] Krouskop TA, Wheeler TM, Kallel F, Garra BS, Hall T., "Elastic Moduli of Breast and Prostate Tissues Under Compression", *Ultrason Imaging*, 20(4), 1998.
- [35] W.A. Berg, D.O. Cosgrove, C.J. Doré, et al., "Shear-wave Elastography Improves the Specificity of Breast US: The BE1 Multinational Study of 939 Masses." *Radiology* 262 (2), pp. 435-449, 2012.
- [36] J.H. Youk, E.J. Son, A.Y. Park, J.A. Kim, "Shear-wave elastography for breast masses: local shear wave speed (m/sec) versus Young modulus (kPa)". *Ultrasonography* 2014 Jan, 33(1), pp. 34-39, 2014.

- [37] P. Rzymiski, A. Skorzevska, M. Skibinska-Zielinska, et al. "Factors influencing breast elasticity measured by the ultrasound Shear Wave elastography – preliminary results". *Arch Med Sci* 2011, 7(1), pp. 127-133, 2011.
- [38] S. R. Buss "3-D Computer Graphics A Mathematical Introduction with OpenGL" *Cambridge University Press*, 1st Edition, pp 67-87, 2003.
- [39] R. R. Everett, F. E. Swain, "Whirlwind I Computer Block Diagrams", *Servomechanisms Laboratory*, Massachusetts Institute of Technology, Volume 1 and 2, Report R-127, September 4, 1947.
- [40] The Evans & Sutherland Picture System. "The Interactive, dynamic, 3-D line-drawing system", 1974.
- [41] Bui Tuong Phong, "Illumination for Computer Generated Pictures". *Communications of the ACM*, Vol. 18, no. 6, pp. 311-317. June, 1975.
- [42] A. Vargas, L. Amescua Guerra, M. Bernal, C. Pineda, "Principios físicos básicos del ultrasonido, sonoanatomía del sistema musculoesquelético y artefactos ecográficos". *Acta ortopédica Mexicana*. 22 (6): 2008. 361-373. Instituto Nacional de Cardiología Ignacio Chávez, 2008.
- [43] *Vscan Trademark of General Electric Company* ©2014 General Electric Company. Fecha de consulta 6 Junio 2015.
<https://vscan.gehealthcare.com/introducing-vscan-family>
- [44] *Sonoscape S9 Portable Color Ultrasound Doppler ISO CE FDA* © SonoScape, Co. Fecha de consulta 6 Junio 2015.
<http://www.sonoscape.com/en/index.aspx> [42] David Wolff "OpenGL 4.0 Shading Language Cookbook". Packt Publishing, 1st Edition, pp 6-8, 2011.
- [45] F. Torres, Z. Fanti, E. Lira, C. García Segundo, B. Reyes Ramírez, E.J. Hazan, R. Gerson, F. Arámbula Cosío. "Rastreo de Imágenes y Reconstrucción de Volúmenes de Ultrasonido Médico". *Revista Mexicana de Ingeniería Biomédica*. Vol. 33, Núm. 2, pp. 101-115, Diciembre, 2012.