
UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO
FACULTAD DE INGENIERÍA



VISUALIZACIÓN CIENTÍFICA Y REALIDAD VIRTUAL
COMO APOYO A LA DOCENCIA DE LA FACULTAD DE
MEDICINA.

Director: Ing. José Julián Argil Torres

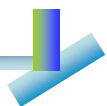
TESIS

Que para obtener el Título de
INGENIERO EN COMPUTACIÓN

Presentan:

José Antonio Terroba Vázquez
Víctor Hugo Villanueva Martínez

México, DF, Septiembre 2008.



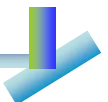
AGRADECIMIENTOS

Un sincero agradecimiento a la Universidad Nacional Autónoma de México, nuestra Alma Mater, que nos proporcionó todas las herramientas necesarias para desarrollarnos como seres humanos.

A la Facultad de Ingeniería, a todos aquellos profesores que fueron de gran influencia en nuestra vida académica.

Gracias a nuestro asesor de tesis Julián Argil por su amistad y confianza.

A la Facultad de Medicina en especial al Dr. Víctor Hugo Ibarra, y a todos aquellos que hicieron posible la elaboración y desarrollo de este trabajo de investigación.



Quisiera agradecer a mi familia, que siempre tuvo fe en mí y cuyo apoyo en las buenas y en las malas me permitió llevar a buen término esta etapa de mi vida.

A mis maestros a través de un largo recorrido, que me dotaron de los conocimientos necesarios para llegar a donde estoy ahora.

Y finalmente, a mis amigos, quienes estuvieron a mi lado durante todo el camino, e hicieron más liviana la carga. Ustedes saben a quiénes me refiero.

José A. Terroba V.

Un agradecimiento profundo a mis padres por darme la vida. Gracias de todo corazón porque sin su educación, sus palabras de aliento, su guía y ayuda incondicional no hubiera logrado ser la persona que soy.

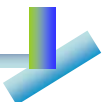
A mis hermanos por ser mis compañeros de peleas y juegos durante toda su vida. Gracias por soportarme y brindarme su amistad.

Gracias mis abuelos por sus sabios consejos, a mis tíos y primos por su apoyo.

No pueden faltar los grandes amigos que me han acompañado en aquellos momentos de adversidad y diversión. Me gustaría poder nombrar a todos y cada uno, pero sencillamente son demasiados.

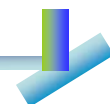
Gracias a todos por permitirme ser parte de su vida.

Víctor Hugo Villanueva M.

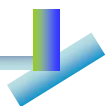


ÍNDICE

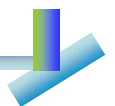
<i>CAPÍTULO 1. Introducción.</i>	1
<i>CAPÍTULO 2. Adquisición y procesamiento de imágenes médicas.</i>	4
2.1 Adquisición de Imágenes Médicas	5
2.1.1 Rayos x	5
2.1.2 TAC	6
2.1.3 Resonancia Magnética	9
2.2 Procesamiento Digital de Imágenes	10
2.2.1 Reducción de Ruido	13
2.2.2 Filtrado	13
2.2.3 Segmentación de imágenes	15
<i>CAPÍTULO 3. Algoritmos de Reconstrucción 3D.</i>	17
3.1 Realidad Virtual	18
3.2.1 Mundo Virtual	19
3.1.2 Realidad Virtual y Visualización	21
3.2 Reconstrucción 3D	22
3.3 Volume Rendering	23
3.3.1 Volume Rendering Indirecto	23
3.3.1.1 Volume Rendering Basado en Planos	25
3.3.1.2 Volume Rendering Basado en Superficies	26
3.3.2 Algoritmos de Volume Rendering Indirecto	27
3.3.2.1 Voxelización	27
3.3.2.2 Marching Cubes	29
3.3.3 Modelo Teórico de Volume Rendering Directo	32
3.3.4 Algoritmo de Volume Rendering Directo: Ray Casting	32
3.4 Amira Visualizador 3D	33
3.4.1 Requisitos del sistema	35
3.5 Optimización de modelos 3D	36



<i>CAPÍTULO 4. Animación.</i>	37
4.1 Animación	38
4.1.1 Timing	39
4.2 Animación mediante cinemática	40
4.2.1 Cinemática Directa	41
4.2.2 Cinemática Inversa	42
4.2.2.1 Aproximación Analítica	43
4.2.2.2 Aproximación iterativa	44
4.2.3 Curvas Cinemáticas	45
4.3 Animación mediante Keyframes (cuadros clave)	45
4.4 Animación Esqueletal en Maya	47
<i>CAPÍTULO 5. Biomecánica de la Columna Vertebral y Miembro Inferior Humanos: Análisis</i>	49
5.1 Caso Práctico: Biomecánica de la Columna Vertebral y Miembro Inferior	50
5.2 Investigación Preliminar	51
5.3 Análisis	57
<i>CAPÍTULO 6. Biomecánica de la Columna Vertebral y Miembro Inferior Humanos.: Desarrollo.</i>	62
6.1 Desarrollo Modelado	65
6.1.1 Adquisición de imágenes	65
6.1.2 Procesamiento de imágenes con Matlab	66
6.1.3 Generación de Modelos 3D con Amira	68
6.1.4 Segmentación De huesos	68
6.1.5 Exportación de formato Amira - Maya	69
6.1.6 Reconstrucción Anatómica	70
6.1.7 Animación Esqueletal en Maya	70
6.2 Desarrollo Animación	71
6.2.1 Joints	71
6.2.2 Ik-handle	74
6.2.3 Script MEL	76
6.3 Pruebas	80
6.4 Aportación a la Docencia	80
6.5 Resultados	83



<i>CONCLUSIONES.</i>	84
<i>GLOSARIO.</i>	88
<i>APÉNDICES.</i>	
I. Formato DICOM	92
II. Formato VRML	96
III. Formato OBJ	100
<i>BIBLIOGRAFÍA.</i>	107
<i>MESOGRAFÍA.</i>	109



CAPÍTULO 1

INTRODUCCIÓN

El campo de la visualización está enfocado en crear imágenes que transportan información sobresaliente acerca de datos y procesos. La visualización juega un papel importante al facilitarnos comprender datos complejos, datos en dos, tres o más dimensiones llevadas a diversas aplicaciones como procesos médicos, ciencias terrestres y espaciales, fluidos, procesos biológicos, entre muchas otras áreas.

La visualización médica es un área especial de la visualización científica, establecida como un área independiente a finales de la década de 1980. Sus raíces son varias, desde la ilustración anatómica comenzada con Leonardo Da Vinci, hasta la graficación por computadora. Ésta última ofrece representaciones para almacenar geometría tridimensional y algoritmos para mostrar dichas representaciones. Otra influencia viene del procesamiento de imágenes, el cual básicamente definió el campo del Análisis de Imágenes Médicas. Esto, sin embargo es aplicable más al procesamiento de imágenes bidimensionales, mientras que su extensión tridimensional normalmente se acredita a la visualización médica. [1]

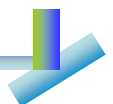
Este trabajo de tesis es referente al uso de la visualización científica y médica como un apoyo a la docencia. Su uso como una herramienta educativa no es algo nuevo. *VoxelMan* es un sistema educativo avanzado que combina superficies de alta calidad y Volume Rendering con instalaciones de interacción 3D, y una base de datos para apoyar la educación en la materia de anatomía. [1].

En nuestro país, ya existen herramientas utilizadas hoy en día para apoyar el trabajo docente. En la Facultad de Medicina de la UNAM, en conjunto con el Laboratorio de Visualización Ixtli, el proyecto antecesor al concerniente a este trabajo de tesis, de una representación tridimensional de un Cráneo Desarticulado Humano, es parte de las clases regulares de algunos grupos que toman la asignatura de Anatomía.

A lo largo de este trabajo, se cubrirán los antecedentes teóricos necesarios para entender el proceso de la visualización científica, y la aplicación de los mismos para la elaboración de un demo interactivo capaz de ser ejecutado en la sala Ixtli utilizando las capacidades estereoscópicas del mismo.

Se verán a fondo los principales métodos de adquisición de imágenes médicas, como son los rayos X, las imágenes de resonancia magnética, y de tomografía asistida por computadora. Analizando los procedimientos y los principios físicos que explican el funcionamiento de dichas máquinas. A su vez, se explorarán diversos algoritmos y técnicas de procesamiento digital de imágenes, que permitirán mejorar la calidad de las imágenes adquiridas.

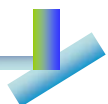
Posteriormente, se explicarán diversos algoritmos que nos habilitan a elaborar modelos tridimensionales a partir de imágenes adquiridas anteriormente. Se toman en cuenta los principales algoritmos para este



propósito. Siendo el más importante el de Volume Rendering, en sus dos variantes, directo, e indirecto. Examinaremos los modelos teóricos detrás de los algoritmos, y detallaremos los más representativos, por ejemplo, el algoritmo de Volume Rendering indirecto llamado Marching Cubes, de uso muy común en esta área.

Después, se analizarán técnicas de animación, poniendo especial atención a aquellas que, debido a su utilidad, son ampliamente ocupadas en el desarrollo de animaciones por computadora. Tal es el caso de la animación por keyframes, o por cinemática, ya sea directa o inversa. También tomamos el cuidado de explicar un caso muy particular, de la animación esquelética en el software de modelado y animación tridimensional de Autodesk, Maya. Debido a la utilización de modelado jerárquico, la robustez y la facilidad que este software brinda al usuario para poder crear animaciones de alta calidad, sin ser del todo complicadas.

En los siguientes dos capítulos, se profundiza en el proyecto realizado. Se muestra el análisis del problema a resolver, se explica la aplicación de los conceptos teóricos ampliados en los capítulos anteriores, y la manera de resolver dicho problema mediante el uso de nuestros conocimientos y las herramientas a nuestro alcance.



CAPÍTULO 2

ADQUISICIÓN Y PROCESAMIENTO DE IMÁGENES MÉDICAS

2.1 Adquisición de Imágenes Médicas

Las imágenes médicas sirven varios propósitos, diagnósticos, planeación de tratamientos, navegación intraoperatoria, y monitoreo postoperatorio. Principalmente, se utilizan para confirmar o descartar una teoría hecha por el doctor antes de hacer un diagnóstico.

2.1.1 Rayos X

Los rayos X fueron descubiertos en 1895 por Wilhelm Konrad Röntgen. Las imágenes de rayos X se basan en la atenuación de cuantos de rayos X que viajan a través del objeto a analizar. La atenuación se basa en dos procesos: Los cuantos se *absorben* por el objeto alcanzado por los rayos X, y son *dispersados* por el llamado efecto Compton. El efecto Compton ocurre si los fotones llegan a un átomo con alta energía, y causan que electrones del exterior del átomo se desprendan y se muevan a cierto ángulo de la dirección del fotón. Este efecto limita el contraste y calidad de las imágenes de rayos X. [1]

Mientras más denso sea el material del objeto, como hueso o metal, más cuantos son absorbidos. La absorción también depende del grosor del objeto.

Los cuantos de rayos X se generan por electrones acelerados a través de un campo eléctrico. Cuando estos electrones golpean el material objetivo (ánodo), su energía cinética se transforma en calor (alrededor de 99%) y cuantos de rayos X. El material objetivo suele ser tungsteno, para huesos, o molibdeno para mamografías u otros tejidos suaves. [1]

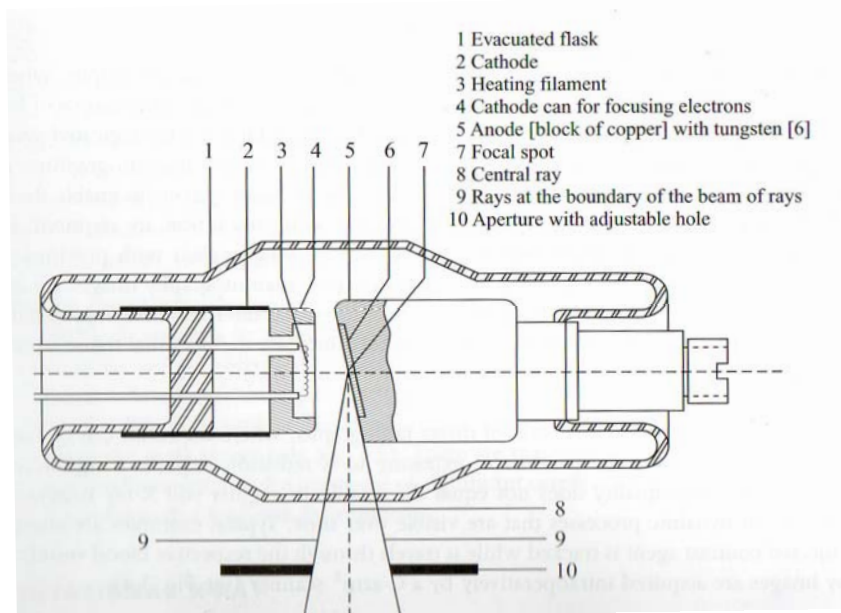


Figura 2.1: Máquina de Rayos X.

En general hay dos tipos de rayos X, los llamados *suaves* o *duros*. Los primeros se crean cuando se usa un voltaje menor a 100 kV, los segundos cuando es mayor. Son llamados de este modo porque con los rayos X duros, el efecto dispersor predomina sobre la absorción, por lo que solo los objetos de mayor densidad absorben, como los huesos. Al contrario, en los rayos X suaves, predomina la absorción, lo que habilita una mejor representación de tejidos suaves, sin embargo, esto, a su vez, causa mayor absorción de energía a los tejidos, por lo que los rayos suaves son más dañinos que los duros. [1]

Un gran problema de los rayos X, es que tienen la capacidad de dañar el material genético, lo cual es considerado una causa de cáncer. El problema radica en la cantidad de radiación a la que se exponga el paciente. En general, existe un compromiso entre calidad de la imagen y la dosis de radiación, aun cuando los avances tecnológicos han mejorado la calidad de la imagen reduciendo la cantidad de radiación. [1]

2.1.2 TAC

Las Tomografías Computarizadas (CT, CAT, o TAC, por sus siglas en español) fueron introducidas en 1968 por Godfrey Hounsfield. Esencialmente, las TAC son una serie de imágenes de rayos X agrupadas en un conjunto de datos volumétricos. Estas imágenes son adquiridas por un sistema emisor/detector que rota alrededor del objeto a escanear. Cada imagen de rayos X representa un perfil de intensidad medido por los detectores. De estos perfiles obtenidos por una rotación completa, se computa la imagen del corte del objeto a través de la transformada de Radon [RADON 1917, CHO et al. 1993], a su vez basada en el teorema de Fourier.

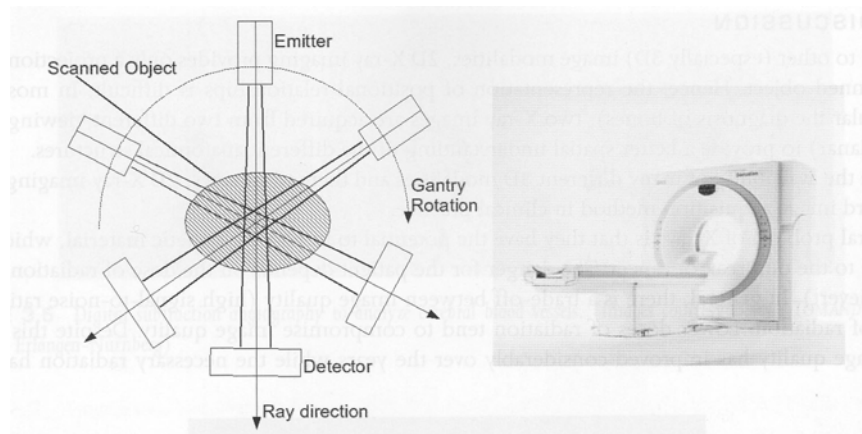


Figura 2.2: Máquina de TAC.

Comparando con los rayos X, las TAC son 2 órdenes de magnitud más sensibles. Los rayos X no pueden distinguir entre tejidos blandos como el hígado o el páncreas, mientras que las TAC pueden hacerlo. Sin embargo, el

contraste entre tejidos blandos es muy pequeño con las TAC, para un mejor contraste, es mejor realizar una resonancia magnética.

Principio de generación de datos TAC

La principal diferencia entre las etapas de desarrollo de las TAC son los diferentes algoritmos de reconstrucción y la arquitectura del emisor y el receptor. La primera generación utilizaba un emisor de rayos X en forma de pluma y un único receptor al lado opuesto del objeto. Para obtener los datos, el emisor se trasladaba a lo largo del objeto y después se rotaba para la siguiente serie de rayos. Esta arquitectura era poco eficiente en el uso de la radiación, y el movimiento mecánico del emisor y el receptor lo hacían lento y costoso.

La siguiente generación utilizó rayos en forma de abanico, transformados computacionalmente en rayos paralelos, y receptores múltiples para escanear dos proyecciones rotacionales circundantes al mismo tiempo. Esto redujo el número de rotaciones necesarias, a su vez, reduciendo el tiempo necesario para escanear un objeto. También mejoraba la utilización de radiación.

La siguiente mejora aumentó el ángulo del abanico de rayos y el número de receptores para cubrir el objeto completo, de este modo, la traslación se hizo innecesaria, y se redujo el tiempo de escaneo drásticamente (de entre 10-60 segundos por corte a 5 segundos). Al igual que en las generaciones anteriores, la radiación se emitía en intervalos fijos y era medida por los receptores.

En la cuarta generación, el detector rotatorio se sustituyó por un detector fijo en forma de anillo circular, lo cual redujo el esfuerzo técnico de mover la mayor masa del emisor y receptor. En este modelo, la radiación se emitía permanentemente, y los detectores sólo se habilitaban en intervalos arbitrarios. Sin embargo, los altos costos y la baja calidad de imagen causada por algunos efectos de dispersión de los rayos X, causaron que se invirtiera más en el desarrollo de la tercera generación.

Hoy en día. Los sistemas utilizan una configuración en espiral o helicoidal, donde el sistema emisor/receptor rota permanentemente alrededor del objeto, mientras el objeto se mueve continuamente en dirección perpendicular para obtener el volumen de datos completo. Esta técnica permite un escaneo más rápido, debido al movimiento continuo del sistema emisor/receptor. Desde 1999, diferentes capas de emisores y receptores se han combinado para crear scanners TAC multicorte, logrando escaneos rápidos y casi isotrópicos. Hasta 2007 los scanners más avanzados pueden adquirir hasta 64 cortes (espirales, en realidad) en un tiempo.

Aparte del desarrollo en las arquitecturas, diferentes algoritmos de reconstrucción de volumen/corte se utilizan en los diversos sistemas y generaciones. El enfoque básico es la transformación inversa de las

proyecciones de los cortes sobre los cortes del volumen por la transformada de Radon. En los primeros días de las TAC, se utilizaban *técnicas de reconstrucción algebraicas* (ART, por sus siglas en inglés), para resolver el problema. Sin embargo, el alto costo computacional de resolver matrices de gran tamaño¹ iterativamente hacía que este método no fuera factible en aplicaciones regulares. El método estándar hoy en día es el de proyección trasera filtrada, de rayos paralelos y de cono, donde cada proyección se compone de acuerdo a la dirección en que se mida. Aun cuando hay métodos de rayos paralelos y de abanico, los métodos para rayos abanico son más complejos y menos eficientes que los algoritmos de rayos paralelos. Es por esto que las proyecciones de los scanners de abanico se reclasifican como rayos paralelos antes de la reconstrucción. Otra modificación se requiere para el problema de la plancha de los scanners espirales de la actualidad, que está en constante movimiento. Una interpolación-z corrige las proyecciones medidas de acuerdo con el movimiento de la plancha.

Los perfiles de intensidad de las imágenes de proyección son pasadas por un proceso de filtrado en la etapa de reconstrucción, lo cual puede aumentar o suavizar los bordes.

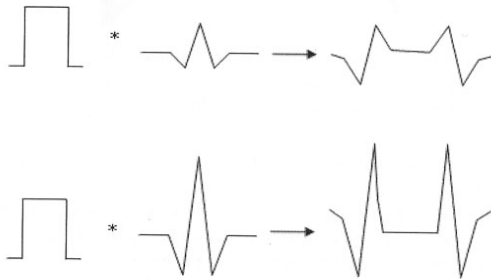


Figura 2.3: Filtros de Reconstrucción TAC

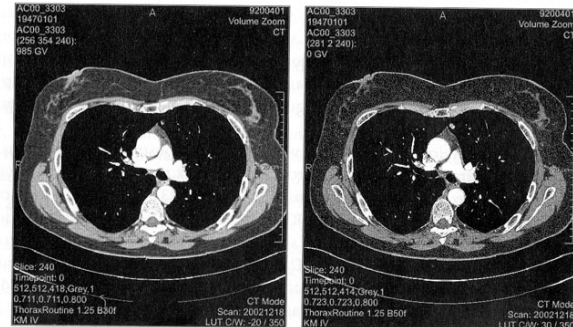


Figura 2.4: Reconstrucción del Tórax pasada a través de diferentes filtros

Parámetros TAC.

Los conjuntos de datos adquiridos por un scanner TAC se pueden describir por su resolución espacial, compuesta por el número de cortes, el número de pixels por corte, y la distancia entre voxels. El número de pixels en un corte también se conoce como *matriz de imagen*. Un conjunto de datos con una distancia entre voxels diferente a la distancia entre pixels se describe como un conjunto anisotrópico, mientras que si son iguales, se tiene un conjunto isotrópico.

¹ El tamaño de una matriz de reconstrucción es igual a la resolución del corte.

La resolución de los datos influye en el nivel de ruido: datos con mayor resolución contiene más ruido si la dosis de radiación se mantiene constante. La dosis de radiación también influye. Con una mayor dosis, se logra una mejor relación señal/ruido. Una vez más, existe un compromiso entre la calidad de imagen deseada y cantidades indeseables de radiación. Es por este compromiso que la visualización basada en datos clínicos no puede alcanzar la calidad que se puede alcanzar con datos obtenidos de cadáveres.

2.1.3 Resonancia Magnética

Las Imágenes de Resonancia Magnética (MRI por sus siglas en inglés) se basan en diferentes propiedades del tejido humano bajo fuertes campos magnéticos. En particular de la existencia de núcleos de hidrógeno en el tejido humano es lo que se aprovecha para la generación de imágenes. Estos núcleos se pueden considerar como pequeños magnetos dipolos que se alinean de forma paralela o anti-paralela a lo largo de un campo magnético intenso. Cuando están alineados al campo, los núcleos giran arbitrariamente alrededor del eje del campo. Hay una pequeña diferencia en el número de protones² alineados de forma paralela y anti-paralela. Esto resulta en una magnetización neta. Para medir dicha magnetización, se debe generar una componente perpendicular al campo, esto se logra con una señal de pulso en radiofrecuencia a la frecuencia *Larmor*³. Como resultado de este pulso RF, los protones hacen un movimiento de precesión con la frecuencia *Larmor*. La resonancia nuclear fuerza a que los protones reciban energía del pulso RF. Este pulso también obliga a todos los protones a girar de manera síncrona, o en fase, y de girar a la orientación anti-paralela de mayor energía, hasta que el número de protones paralelos es igual al número de protones anti-paralelos.

Para reconstruir información espacial de la señal medida, se aplican dos campos magnéticos gradientes adicionales. El primero elige la dirección z del volumen, ya que una capa de protones es suficiente para la frecuencia Larmor para los campos principales y gradientes. Un campo adicional en la dirección de x elige el plano de y, lo cual permite medir diferentes direcciones de proyección, cosa imposible en las TAC, que siempre son axiales.

² Los núcleos de hidrógeno se conforman de un protón.

³ La frecuencia Larmor se puede describir como la frecuencia de resonancia de los protones en un campo magnético. Esto se determina por la relación giromagnética entre los protones (la cual es constante para protones del mismo tipo, como protones de hidrógeno) y la intensidad del campo magnético.

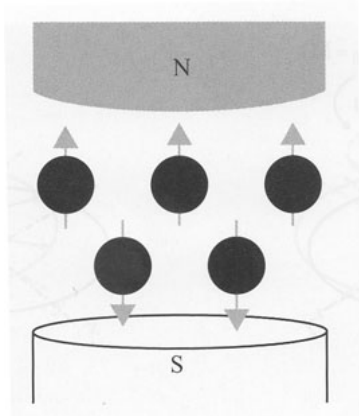


Figura 2.5: Alineación Magnética de los Protones

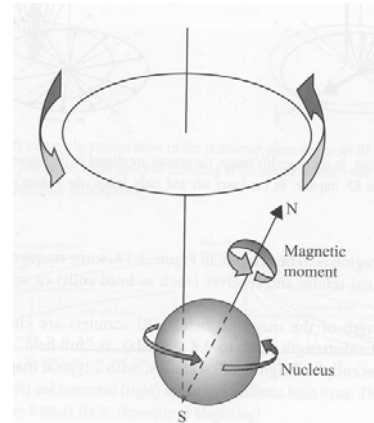


Figura 2.6: Giro y Precesión de Protones

Las imágenes de resonancia magnética se realizan a través de un magneto anular de suficiente tamaño para abarcar al paciente completo. Dentro de este magneto, bobinas gradientes están integradas de tal manera que los campos magnéticos gradientes ortogonales (x, y, z) puedan ser generados. Las bobinas utilizadas para la adquisición de imágenes RM son muy diferentes, dependiendo de la parte del cuerpo de donde se quieran adquirir.

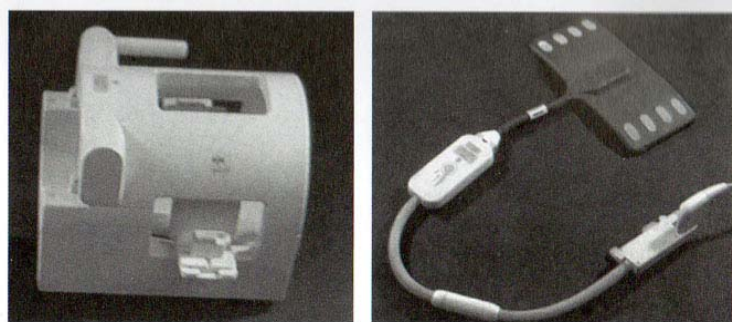


Figura 2.7 Bobinas para MRI.

Dependiendo de la intensidad del campo magnético, los scanners MRI se caracterizan como de "campo bajo" (hasta 0.8 T⁴), de "campo completo" (1.5 T), o recientemente, de "campo alto" (3 T o más).

2.2 Procesamiento Digital de Imágenes

El desarrollo del área de procesamiento digital de imágenes se encuentra ligado directamente al desarrollo de la tecnología computacional. Las primeras computadoras capaces de realizar procesamiento de imágenes aparecieron alrededor de los años 60 con el programa espacial de los estados unidos de Norteamérica.

⁴ Tesla.

A lo largo de miles de años la evolución del ser humano ha dado como resultado la aparición de diversos sentidos que nos han permitido desarrollar ciertas habilidades que nos permiten tener una percepción clara del mundo que nos rodea, entre estas habilidades sin duda la vista es la de mayor desarrollo.

En el caso de las imágenes es fundamental contar con la percepción visual de varios observadores ya que, dependiendo de cómo se maneje la imagen existirán diversas opiniones de si dicha imagen es buena o no.

No podemos pensar en el procesamiento de imágenes sin considerar el sistema visual humano, compuesto de ciertos órganos que intervienen en la percepción y en la manipulación de las imágenes que adquirimos por medio de los ojos, que son la entrada principal para la información que será procesada en nuestro cerebro.

El ojo básicamente es una cámara oscura que tiene como misión enfocar una imagen en la retina. Para enfocarla nitidamente y conseguir una buena calidad visual, el globo ocular posee un sistema de lentes para conseguirlo. Son básicamente 5 lentes puestas una delante de otra, pero recurriremos a un modelo simplificado, las consideraremos todas como una sola lente.

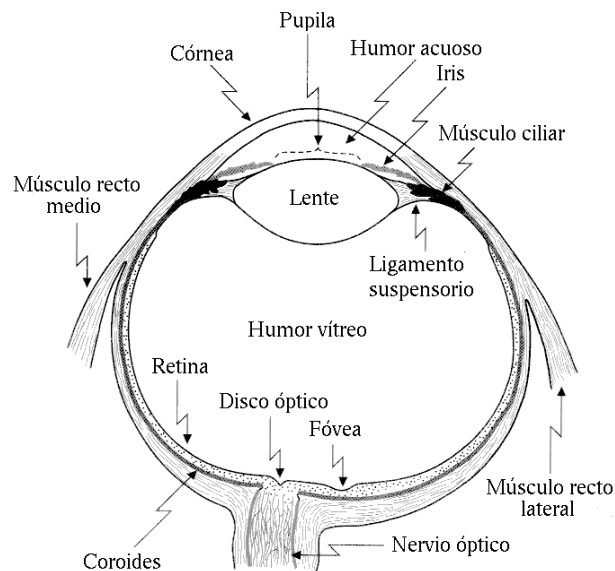


Figura 2.8: Esquema del ojo humano.

Una imagen puede ser definida como una función bidimensional, $f(x, y)$, donde x, y son coordenadas espaciales (en un plano), y la amplitud, o luminosidad, de f en cualquier par de coordenadas (x, y) es llamado intensidad, o nivel de gris de la imagen en ese punto. Cuando los valores de x, y y la amplitud de la función, son todos finitos, en cantidades discretas, es llamada imagen digital. [2]

El 40% de la actividad del cerebro se dedica a la visión, lo que demuestra que una de las funciones más importantes de este órgano es la visión humana. De las 30 áreas del cerebro implicadas en la visión, sólo se conocen bien el funcionamiento de tres de ellas: la retina, el núcleo geniculado lateral y la corteza visual primaria. [M1]

Aunque la experiencia visual se origina en la retina, ésta no es una parte determinante de la percepción, ya que ésta se consolida después en el cerebro, donde se le da sentido.

En cada nervio óptico hay un millón de cables o conexiones, lo que se puede equiparar a los píxeles de una pantalla. Sin embargo, la percepción que se tiene es mucho más completa que únicamente lo que se vería en una pantalla. El cerebro ayuda a completar la visión mediante la abstracción, de modo que el ojo, en realidad, lo que transmite son los datos importantes de lo que vemos, no muestrea toda la escena visual por igual, sino que pone atención en la información más relevante, por ejemplo, los bordes de los objetos y las esquinas constituyen la información que realmente registra y transmite el ojo cuando vemos; el resto, su contenido, lo completa ya el cerebro. [M1]

El análisis de imágenes médicas tiene algunos requerimientos esenciales:

- **Robustez:** Para que una solución de imagen médica sea útil, se requiere que sea robusta, es decir que funcione para una gran cantidad de casos. Soluciones automáticas que solo funcionen en ciertas circunstancias pueden ser parte de un sistema de software, pero deben ser complementadas por otra solución para otros casos.
- **Exactitud:** La segmentación de imagen y las operaciones subsecuentes deben ofrecer resultados exactos. Estudios de validación se requieren para determinar si la exactitud se logra. Dichos estudios son esenciales, aunque normalmente tiene un rango limitado, típicamente para algún protocolo de imagen especializado.
- **Reproducibilidad:** Esto significa que un usuario tendría un resultado muy similar si aplicara un algoritmo varias veces (reproducibilidad intra-observador) y que diferentes usuarios también producirán resultados similares (reproducibilidad inter-observador). Este requerimiento es más fácil de probar porque no es necesaria ninguna correlación con una verdad básica. La reproducibilidad también se conoce como *precisión*, pero se utiliza el anterior para no confundirse con exactitud.
- **Velocidad:** Puede haber soluciones excelentes que sean computacionalmente muy demandantes, como puede ser un

filtrado muy sofisticado. Si no es posible acelerar estos métodos a un nivel para el cual un uso interactivo sea factible, no deben ser utilizados. El aumento del poder de cómputo alivia un poco el problema, aunque los tamaños crecientes de los datos de imagen a menudo compensan el poder computacional adicional.

2.2.1 Reducción de Ruido

Las imágenes médicas presentan ruido aleatorio debido a los procesos estocásticos en la adquisición de las mismas. El ruido se caracteriza por una cierta amplitud y distribución. El nivel de ruido regularmente se mide con la relación señal/ruido de toda la imagen. El nivel de ruido depende de la resolución espacial, entre otros factores; datos de alta resolución exhiben más ruido. En los rayos X y las TAC, el nivel de ruido depende de la cantidad de radiación. Por lo tanto, se mantiene el nivel de radiación ionizante a un mínimo, a expensas de una relación señal/ruido baja.

Se utilizan filtros de reducción de ruido para mejorar los datos. En general, se considera que el ruido ocurre en altas frecuencias, es por esto que se utilizan filtros paso bajas para eliminar el ruido. El diseño de estos filtros se basa en suposiciones concernientes a la distribución y amplitud del ruido. Varios filtros se han diseñado para reducir el ruido con una distribución Gaussiana, cuyo kernel se representa con la siguiente ecuación.

$$G(x, \sigma, \mu) = \frac{1}{\sigma \cdot \sqrt{2 \cdot \pi}} \cdot e^{-\frac{x-\mu}{\sigma^2}}$$

Donde:

μ es el centroide de la función

σ es la desviación estándar de la función

2.2.2 Filtrado

El filtro Gaussiano es un ejemplo del concepto general del filtrado de una función g con un filtro F , a menudo representado con una convolución.

$$g'(x) = g(x) \otimes F(x) = \int_{-\infty}^{\infty} g(x) \cdot F(x - x_1) \cdot dx$$

Donde \otimes es el operador convolución.

Aplicando este concepto a datos de imagen discretos, la convolución se expresa como la suma ponderada de la señal g sobre el filtro F con $2N+1$ elementos.

$$g'(u) = \sum_{i=-N}^N g(u) \cdot F(u - i)$$

Los filtros simples son filtros locales que modifican la intensidad de imagen en cada voxel por una combinación de las intensidades de imagen de los voxels circundantes. Los filtros discretos locales los caracteriza un kernel, una matriz de elementos con el mismo tamaño que el vecindario considerado. Normalmente la matriz tiene un tamaño de $(2N+1) \times (2N+1) \times (2M+1)$, con N siendo 1, 2 o 3, y M siendo 0, 1 o 2. En casos isotrópicos, M y N tienen el mismo valor, mientras que en el caso más típico, el anisotrópico, una M más pequeña se elige para compensar la extensión mayor en la dirección de z . Números pares para el tamaño de la matriz aseguran que exista un valor central.

Cuando $M=0$, el filtro se aplica a cada corte por separado y no afecta al resto de los cortes.

Si los elementos de la matriz son constantes, el filtro correspondiente es estático. Los filtros que adaptan su contenido a las características de imagen locales se llaman dinámicos.

Filtros de Reducción de Ruido Estáticos

Estos filtros están escalados de manera que la suma de sus elementos sea 1, esto resulta que el valor de la media de los valores de gris de la imagen permanece igual después de que se aplica el filtro. Esta normalización es importante para mantener la intensidad de la imagen en conjunto.

El filtro se aplica iterando sobre todos los pixels y reemplazando la intensidad con un promedio ponderado de los pixels circundantes. Los pesos de dichos pixels se caracterizan por la matriz del filtro. El filtro más simple es el filtro promedio, donde cada pixel del vecindario tiene la misma influencia. Se consiguen mejores resultados con un filtro que tome en cuenta la distancia al voxel central. Un ejemplo de este tipo de filtros es el binomial, que es una versión discretizada de la función Gaussiana. Los elementos de estos filtros son coeficientes binomiales.

Un filtro binomial de 5x5

1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

El problema inherente a los filtros estáticos es que no se adaptan. Características como los bordes en una imagen no se preservan y se ven borrosos.

Filtros de Reducción de Ruido Estáticos

En general, se pueden obtener mejores resultados con filtros dinámicos, que analizan y consideran intensidades de imagen en el vecindario local. Un ejemplo importante es el filtro de mediana. Con este filtro, los voxels en el vecindario se clasifican en contenedores de acuerdo a su intensidad de imagen, y se determina la mediana (intensidad del voxel a la mitad de la secuencia). El voxel central es reemplazado por el valor de la mediana. Sin embargo, la etapa de clasificación toma un tiempo considerable; por lo tanto, los kernels no deben ser de gran tamaño. En general, los filtros dinámicos son más demandantes de recursos de cómputo, pero permiten adaptarse a las características de los datos.

2.2.3 Segmentación de imágenes

La segmentación se refiere al proceso de descomponer datos de imagen en estructuras significativas que son relevantes para una tarea específica. La segmentación tiene dos aspectos:

- Los objetos relevantes deben ser identificados, lo que significa que deben ser reconocibles como una estructura anatómica particular.
- Deben ser delineadas, es decir, que sus bordes deben ser precisamente especificados.

El reconocimiento es una tarea de alto nivel que los humanos usualmente hacen mejor que las computadoras. La delineación, por otra parte, es una tarea donde la exactitud es esencial. El reto de la segmentación es combinar la fuerza del usuario con el potencial del apoyo computarizado.

Segmentación Manual

La manera más general y fácil de lograr de la segmentación de imagen es el dibujo manual en los cortes de datos radiológicos: el usuario dibuja el contorno de las estructuras relevantes con un dispositivo señalador. Para modificar el contorno, es a menudo posible volver a dibujar una porción particular. Que reemplaza la porción previamente dibujada del contorno.

Este enfoque es robusto (siempre aplicable); sin embargo, ocupa mucho tiempo, no es reproducible, y no es preciso, porque el usuario suele desviarse ligeramente del contorno deseado. A pesar de estos problemas, su uso es bastante común, particularmente si los objetos son muy difíciles de delinear debido a bajo contraste o una forma inesperada. Por ejemplo, la segmentación de tumores suele hacerse a mano.

Segmentación Basada en Umbrales

Con un umbral global o un intervalo de umbrales superiores e inferiores aplicados a la intensidad de una imagen, una imagen binaria se genera.

$$\varepsilon = |I(v) - a|$$

La anterior ecuación ejemplifica la situación típica, donde un intervalo de umbrales se define por una media a y una tolerancia ε . Todos los valores de voxel $I(v)$ que caigan dentro del intervalo $([a - \varepsilon, a + \varepsilon])$ son seleccionados.

La aplicación más común de la segmentación basada en umbrales es la identificación de huesos en los datos TAC. Sin embargo, aún en esta aplicación, la exactitud de este enfoque es limitada, ya que usualmente los huesos muy delgados no son correctamente identificados. Esto se debe al efecto de volumen parcial, que hace una media de las intensidades de imagen lo cual conlleva menores valores de intensidad para voxels que solamente representan estructuras óseas parcialmente.

Selección de Umbral

La segmentación basada en umbrales se puede hacer de manera más veloz y más reproducible si se soporta el proceso de selección de umbral. Hay varios métodos para "sugerir" valores de umbral. Muchos de estos métodos se basan en el histograma de intensidades de imagen.

Un mínimo local en el histograma a menudo representa el umbral, el cual es óptimo para distinguir entre dos tipos de tejido. El mínimo local es una sugerencia razonable si la frecuencia en esta posición es baja.

Análisis de Componentes Conectados

Este análisis considera la imagen binaria e inicializa un primer componente con el primer pixel. El algoritmo busca pixels adyacentes recursivamente en la imagen binaria y los suma a este componente. Si no encuentra más pixels conectados, y aún hay pixels sin visitar, se inicializa un nuevo componente. El procedimiento termina cuando se han procesado todos los pixels y se han asignado a una región. Esto debe ser mostrado al usuario, por ejemplo, utilizando diferentes colores.

Este procedimiento se usa a menudo con la segmentación basada en umbrales.

CAPÍTULO 3

ALGORITMOS DE RECONSTRUCCIÓN 3D

3.1 Realidad Virtual

El término de Realidad virtual se refiere al uso que se le da la tecnología computacional para poder crear el efecto de un mundo tridimensional interactivo en el cual los objetos tienen un sentido de presencia espacial.

(Steve Bryson - NASA Ames) [M2]

La realidad virtual pretende la manipulación de los sentidos humanos a través de medios electrónicos o artificiales. [3]

En la actualidad el término de realidad virtual es más común que en décadas pasadas, debido principalmente al auge de nuevos usos de los avances computacionales, los cuales permiten a un usuario sumergirse en un mundo tridimensional generado por computadora, con el que es posible interactuar, experimentando sensaciones similares a las que se tienen en un espacio físico real. Las tecnologías basadas en realidad virtual pueden ser aplicadas en áreas tan diversas como la industria, la investigación, el entretenimiento y en el área de nuestro interés, las ciencias médicas.

La realidad virtual tiene la capacidad de proveer información sensorial de suficiente fidelidad, por lo que el usuario puede, en algunos casos, suspender la realidad y aceptar que se encuentra en algún otro lugar. [4]

Existen dos tipos de realidad virtual, la inmersiva y la no inmersiva, ambas generan ambientes o mundos virtuales en tiempo real.

La realidad virtual inmersiva se refiere a que en el mundo virtual generado por la computadora nosotros podemos tener la sensación de pertenecer directamente a él, ya que podemos interactuar a través de diversos dispositivos tales como cascos, lentes estereoscópicos, cámaras, guantes u otros dispositivos capaces de capturar la posición y rotación de nuestro cuerpo. Una desventaja es que se requiere una inversión monetaria considerable por el hecho de que son necesarios medios electrónicos que enfatizan el sentido de profundidad, además de que se convierte en una interfaz que no es accesible para la gran mayoría de los usuarios.

En el caso de la realidad virtual no inmersiva interactuamos de forma externa con el mundo virtual creado por la computadora de esta manera no somos capaces de sentirnos parte del mismo debido a que no tenemos otro dispositivo con el cual interactuar más que algunos periféricos de la computadora como lo son el teclado, el monitor y el ratón, algunos ejemplos claros son los juegos MMORPG¹ o Second Life en los cuales se puede interactuar en tiempo real con otras personas en espacios que físicamente no existen. Este tipo de realidad virtual es más accesible a un mayor número de usuarios por el hecho de que no se necesita una gran inversión y la interfaz

¹ Por sus siglas en inglés, Juego de rol masivamente multijugador en línea (Massively Multiplayer Online Role-Playing Game).

necesaria es una simple computadora con periféricos básicos, esto aumenta su facilidad de uso.

Para lograr gráficos 3D reales, una imagen estereoscópica es producida para proveer dos diferentes vistas del mismo objeto para cada ojo del usuario. Al proveerse dos imágenes separadas se permite a nuestra vista tener un campo de visión independiente, de forma que cada ojo sólo está viendo su imagen, de esta manera se “engaña” a nuestro cerebro, haciéndole creer que cada ojo está viendo una imagen desde una perspectiva ligeramente distinta, lo cual genera el sentido de profundidad e inmersión. [M3]



Figura 3.1: Un casco de realidad virtual (Head Mounted Display) permite un mayor de sentido inmersión al usuario. [M4]

La tecnología de hardware ha tenido que avanzar para generar mayor interacción del usuario en tiempo real, algunos de los dispositivos de hardware principales son: tarjetas gráficas, dispositivos que permiten el despliegue de imágenes estereoscópicas (monitores, cascos (figura 3.1), lentes estereoscopios, etc.), tarjetas de sonido que proporcionan profundidad de sonido.

3.1.1 Mundo virtual (ambiente virtual)

Gracias al avance tecnológico el cine se vio beneficiado con la producción de películas de ciencia ficción sumamente interesantes. Una de ellas, The Matrix, muestra cómo historia principal el desarrollo de mundos generados totalmente por computadora (mundo virtual), que logran controlar todo, incluso las mentes humanas, Figura 3.2.



Figura 3.2: "The Matrix" (Warner Brothers, 1999). [M5]

El desarrollo de aplicaciones de realidad virtual efectivas requiere de complejos ambientes virtuales.

Un mundo virtual es un ambiente similar a la realidad que se crea utilizando un medio artificial; es una simulación de un ambiente real o imaginario que provee una experiencia en tiempo real proporcionando estímulos parecidos a los de la realidad. [3]

Con la ayuda de la realidad virtual se puede generar ambientes o mundos virtuales en los cuales podemos interactuar, navegar alrededor de ellos e incluso sumergirnos dentro de otros ambientes virtuales. Existen diversas características importantes de estos ambientes virtuales, pero podemos considerar a 3 de ellas como las más destacadas, que son: inmersión, presencia e interacción:

Inmersión. Hace referencia a lo que es, en principio, una descripción cuantificable de la tecnología, es decir, dispositivos electrónicos auxiliares, de esta manera entre mayor información de sensores electrónicos se proporcionen el usuario experimentará una gran cantidad de sensaciones que implica que no deben existir distracciones y que la mayor parte o incluso toda la atención del usuario se enfoca al ambiente virtual por lo que tendrá la sensación de encontrarse dentro.

Presencia. La presencia puede contribuir a dar una mayor naturalidad al ambiente virtual en el cual un usuario se encuentra interactuando, fidelidad de imagen de despliegue, riqueza sensorial, y el comportamiento del usuario en tiempo real son algunas de las características importantes para dar el efecto de estar presente en el mundo virtual.

Despliegue Multimodal. El ser humano cuenta con 5 sentidos: visión, olfato, audición, gusto y tacto. El ingreso de información para manipular uno o más de nuestros sentidos pueden incrementar nuestra percepción del ambiente virtual, al combinar dispositivos electrónicos con el sentido visual, éste último,

es el que más aporta para que el usuario tenga la sensación de inmersión en el mundo virtual.

3.1.2 Realidad virtual y Visualización

El hombre ha desarrollado distintos sentidos que le han permitido ayudar a sobrevivir desde hace muchos años, la visualización toma ventaja de las habilidades naturales del sistema de visión humano.

"La visualización es un método de cómputo. Transforma lo simbólico en geométrico, permitiendo a los investigadores observar sus simulaciones y cálculos. La visualización ofrece un método para ver lo que no se ha visto. Enriquece el proceso de descubrimiento científico e impulsa el surgimiento de intuiciones inesperadas. En muchos campos está revolucionando el modo en que los científicos hacen ciencia." [5]

La visualización es una herramienta computacional que permite estudiar y analizar grandes conjuntos de datos, lo cual puede combinarse perfectamente con la realidad virtual para dar a los usuarios una interacción más accesible e intuitiva.



Figura 3.3: Usuario interactuando con datos geofísicos que describen un domo de sal en el golfo de México. [4]

Existen distintos tipos de terminología que hacen referencia a la visualización:

- Visualización científica. Trata principalmente de la visualización, exploración, y análisis de conjuntos de datos provenientes de mediciones o simulaciones de fenómenos del mundo real.
- Visualización de datos. Es generalmente interpretada para ser más general que la visualización científica, ya que implica el tratamiento de fuentes de datos que pueden ser obtenidos por parte de la ciencia

y la ingeniería, tales como datos financieros, datos de mercadotecnia, la aplicación de datos estadísticos y algunas otras técnicas de análisis.

- Visualización de información. Este campo tiene como propósito la visualización de información abstracta, tales como hipertextos de Internet (www), estructuras de directorios o archivos en un sistema operativo, lo cual es interesante debido al reto que presenta de poder generar sistemas de coordenadas, métodos de transformación o estructuras de datos que permitan el acceso a la información.

3.2 Reconstrucción 3D

En la actualidad la tecnología ha evolucionado de tal manera que ha permitido el desarrollo de nuevas herramientas auxiliares en diversas áreas de investigación. El área de nuestro interés es la visualización científica aplicada a la medicina, particularmente a la visualización de datos tridimensionales que permitirán el análisis de diversas partes del cuerpo humano a través de la computadora.

Como su nombre nos indica la reconstrucción 3D nos permite "reconstruir" algo que está roto o ha sido dividido en partes. A partir de las múltiples imágenes médicas adquiridas se pueden construir los modelos de cada uno de los huesos que componen el cuerpo humano.

Para comprender mejor el manejo de las imágenes podemos auxiliarnos de una de las definiciones de imagen digital que nos dice que las imágenes digitales de n dimensiones pueden ser pensadas como un arreglo de n dimensiones de valores de grises tales que cada elemento del arreglo representa exactamente un punto en la imagen. [6]

Los elementos de un arreglo de dimensión 2 (bidimensional), son llamados pixels (picture elements) y los elementos de un arreglo de dimensión 3 (tridimensional) son llamados voxels (volume elements). En general un arreglo de dimensión n son llamados raster elements o spatial elements (elementos espaciales). [6]

La reconstrucción de superficies 3D a partir de imágenes médicas es uno de los problemas en el área de la visualización científica por computadora debido a que el escáner 3D es capaz de producir información parcial de los objetos los cuales serán reconstruidos mediante algoritmos que permitan completar la información faltante y así poder crear los modelos completos. También es importante realizar una segmentación de las imágenes para separar aquellas zonas del volumen que no se deseen, de modo que sea posible visualizar la superficie 3D representada por esta segmentación.

En general las aplicaciones médicas constan de 4 pasos [M6]:

1. Adquisición de datos: Normalmente imágenes 2D. Estos datos dependen de la técnica de adquisición utilizada.
2. Procesamiento de la imagen: Algunos algoritmos usan técnicas de procesamiento de imágenes para analizar la superficie o filtrar datos de la imagen original.
3. Construcción de la superficie: Lo usual es utilizar para ello elementos volumétricos en 3D, como voxels o polígonos.
4. Despliegue del modelo tridimensional: Mediante algún software que permita la visualización y/o análisis de conjuntos de datos volumétricos.

Diversos métodos de extracción y despliegue de volúmenes han sido desarrollados, algunos de ellos serán descritos brevemente, tal es el caso de Voxelización y Marching Cubes.

3.3 Volume Rendering

Desde el desarrollo de diversas aplicaciones médicas la visualización de datos a través de primitivas tales como puntos, líneas, polígonos y geometrías han facilitado la generación de volúmenes a través de la computadora.

Como resultado de popularidad y usabilidad de los datos volumétricos en las últimas décadas, han surgido una amplia clase de técnicas conocidas como volume rendering, las cuales tienen como propósito principal el transformar efectivamente la información en datos volumétricos (arreglos de datos 3D).

Estas técnicas son usadas para describir un método de despliegue de datos volumétricos en una imagen o plano bidimensional, como lo es el monitor de la computadora, la cual permitirá dar un efecto de profundidad necesaria para poder ver el volumen tridimensional, es decir, la idea básica es que el observador sea capaz de percibir los datos volumétricos desde la proyección en un plano. [8]

Desde que el volume rendering es usado para generar imágenes que representan un conjunto de datos 3D en una imagen 2D, se han introducidos varios retos. La clasificación debe ser hecha para asignar color y opacidad a las regiones dentro del volumen y los modelos volumétricos deben ser definidos para soportar sombreado. Además la eficiencia y compactibilidad son de gran importancia debido a la complejidad de los métodos de volume rendering y el tamaño de los conjuntos de datos volumétricos generados. Un modelo geométrico que consiste de millones de primitivas generalmente es considerado grande, mientras un conjunto de datos volumétricos con un millón de voxels es un poco menor.

Algunas ventajas de trabajar con conjuntos de datos volumétricos es que se puede modelar el interior de algún objeto, como lo pueden ser el interior de los huesos o tejidos del cuerpo humano que permitan detectar enfermedades o malformaciones, además permite modelar fenómenos naturales que no pueden ser manejados físicamente por el hombre, con lo que se pueden hacer simulaciones sin exponer vidas humanas. La uniformidad de cómo se representa el objeto es independiente del proceso que se realiza para su visualización.

La representación de volúmenes también realiza un prefiltrado de imágenes, así que el despliegue del volumen se trata de procesar eficientemente dejando de lado el problema de aliasing, de esta forma el desarrollo de aplicaciones relacionadas con datos volumétricos tiene diversas opciones disponibles para desplegar los datos, la naturaleza de los datos determina el algoritmo que es usado. Existen tres opciones generales [7] para desplegar datos volumétricos en una superficie o plano bidimensional:

1. Dividir el conjunto de datos en cortes con una sección de un plano. Es una opción fácil si el plano está paralelo a uno de los planos coordenados, de esta manera se podrá desplegar la información en dos dimensiones.

2. Extraer un objeto que es conocido dentro de un conjunto de datos y desplegarlo de una forma normal, por ejemplo, si fuera el caso de contar con imágenes médicas se podría visualizar por separado el órgano interno de un cuerpo si es que ha sido diseccionado previamente. Esto implica que el objeto puede ser segmentado del resto de los datos y el segmento es convertido en elementos gráficos.

3. Asignar transparencias y color a los voxels que se encuentran dentro del conjunto de datos para poder ver el volumen completo desde cualquier ángulo, lo cual es una ventaja ya que dependiendo del ángulo de visión se puede dar transparencia u opacidad a ciertos elementos, una aplicación médica es un buen ejemplo ya que se podría simular los rayos X reales, si se le otorga a algunos voxels de los músculos la propiedad de ser semitransparente y los huesos tenerlos en la escala de colores de grises.

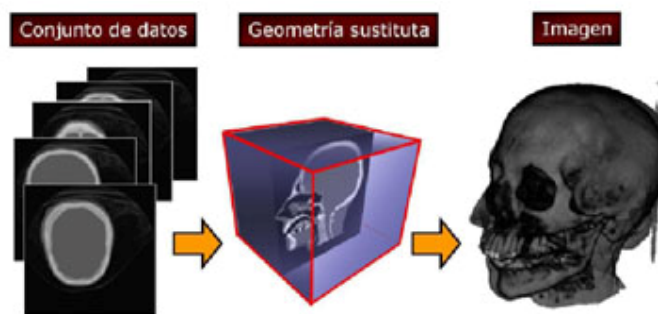


Figura 3.4: Esquema de Volume Rendering.

El volume rendering no solo se limita a aplicaciones de software, también es utilizado a nivel hardware por medio de las tarjetas de video las cuales realizan el procesamiento de los datos para poder ser cargados como texturas así como las propiedades físicas que el objeto debe tener, y mediante una serie de polígonos que se dibujan de atrás hacia adelante se logra reconstruir el volumen a partir de los datos adquiridos, esto permite aprovechar la capacidad de la tarjeta.

Algunas de las operaciones que permiten las tarjetas graficas son:

- Operaciones de rasterización.
- Descomposición de fragmentos.
- Interpolación de colores.
- Texturización (Interpolación y combinación).
- Operaciones sobre fragmentos.
- Prueba de profundidad.
- Alpha blending.

Existen dos tipos de Volume Rendering, directo e indirecto. La visualización indirecta genera una representación intermedia del conjunto de datos que es rendereada más tarde. En el caso del algoritmo *Marching Cubes*, explicado más adelante, es una representación poligonal de una isosuperficie. Como esto no es el conjunto de datos original, se le llama indirecta. La visualización directa, en cambio, representa visualmente el conjunto de datos sin generar una representación intermedia.

3.3.1 Volume Rendering Indirecto

3.3.1.1 Volume Rendering Basado en Planos [1]

Una técnica ampliamente usada en la medicina es el *modo cine*, donde los cortes de un conjunto de datos volumétrico se examinan subsecuentemente. Debido a la naturaleza anisotrópica de dichos conjuntos, el modo cine es utilizado a lo largo de la orientación de los cortes DICOM, especificada por el protocolo de adquisición (axial para las TC), ya que así la resolución del corte y el espaciado del voxel son idénticos².

En general, la mayoría de las estructuras anatómicas no están alineadas con el conjunto de datos ni con cualquier estructura geométrica regular. Por lo tanto, una orientación estándar de cortes nos proveerá una representación insuficiente de la estructura objetivo.

² Por ejemplo, 512x512 y 0.3mm x 0.3mm

Para estas situaciones, cortes oblicuos del volumen nos darán una mejor representación, ya que pueden ser orientados arbitrariamente dentro del conjunto de datos. Los voxels en la intersección del conjunto de datos volumétrico y ese corte serán la representación visual del corte oblicuo.

Para computar los voxels intersectados por el plano oblicuo se puede usar una versión tridimensional del algoritmo de Bresenham. Sin embargo, hoy en día, se aprovechan las capacidades del hardware gráfico, donde el conjunto de datos se interpreta como una textura tridimensional, mapeada a un polígono. Si el polígono representa un corte oblicuo a través del conjunto de datos, las capacidades de mapeado de texturas del hardware lo llenarán con los voxels respectivos. Debido a que este proceso se hace en hardware, es veloz. Además, utiliza interpolación trilineal o cuadrilineal³ para lograr una reconstrucción de muy alta calidad.

3.3.1.2 Volume Rendering Basado en Superficies [1]

Las estructuras de interés en los datos volumétricos se diferencian de los datos de imagen circundantes por un límite o una interfaz material. El volume rendering indirecto basado en superficies busca una representación visual de dicho límite, que necesita ser especificada. Típicamente, este límite está en voxels que tienen el mismo o un similar valor de intensidad. Por ende, la superficie resultante en estos voxels se llama isosuperficie.

Una isosuperficie puede ser especificada como una superficie implícita, donde la función implícita es igual al isovalor (también llamado umbral).

$$i(x) = V(x) - \tau = 0$$

Donde x es la posición del voxel y τ es el isovalor o umbral.

Una isosuperficie separa el conjunto de datos volumétrico en un *interior* y un *exterior*. En el exterior se encuentran todos los voxels con una intensidad menor al umbral τ ($i(x) < 0$). Mientras que todos los voxels con una intensidad mayor a τ son parte del interior ($i(x) > 0$), y todos los voxels con una intensidad igual a τ forman parte de la isosuperficie ($i(x) = 0$).

³ La interpolación cuadrilineal combina 2 interpolaciones trilineales en diferente nivel de resolución con una interpolación lineal

3.3.2 Algoritmos de Volume Rendering Indirecto

3.3.2.1 Voxelización [9]

Un término importante dentro de la reconstrucción 3D es el de voxel, el cual proviene de la contracción del término en inglés volume element (elemento de volumen). Un voxel no es más que otra primitiva que puede ser usada para poder desplegar imágenes tridimensionales, es el equivalente del píxel en un objeto 2D.

Si realizamos una analogía entre una imagen con un volumen, podemos observar que se puede representar un objeto bidimensional en un arreglo de pixels, de igual manera cualquier forma tridimensional se puede representar mediante voxels.

En la literatura podemos encontrar dos definiciones principales, una de ellas nos dice que un voxel es un cubo de pequeño tamaño, o dimensión, en un espacio tridimensional; la otra considera a un voxel como un simple punto sin tamaño en un espacio tridimensional. [8]

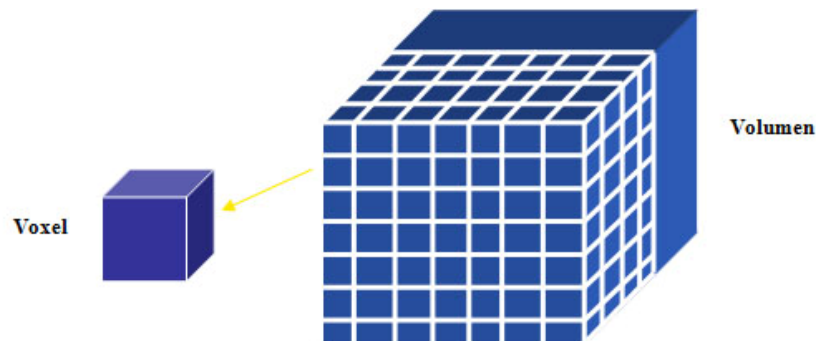


Figura 3.5: Un voxel permite generar volúmenes.

Se han inventado diversas técnicas de adquisición de imágenes que permiten el desarrollo de aplicaciones médicas para su visualización y almacenamiento, indicando la presencia de algún tejido, su densidad o su tipo, y desplegándose ya sea a través de cortes transversales o agregando una cierta cantidad de color a cada voxel para definir cada elemento del cuerpo humano.

Entendemos por voxelización al proceso de convertir geometrías o superficies poligonales en un conjunto de voxels que permiten visualizar de mejor manera un modelo tridimensional, es decir, que el modelo u objeto se dividirá en cubos del mismo tamaño en un espacio tridimensional, auxiliándose de múltiples planos de corte o rebanadas que permitirán segmentar el volumen a lo largo del método.

La voxelización requiere calcular las intersecciones de un modelo poligonal con cada plano o rebanada de un volumen. Un método general para voxelizar un objeto 3D comprende ciertos pasos, algunos de ellos son:

- Generar rebanadas del objeto 3D y almacenarlas en memoria RAM (frame buffer) teniendo funciones para combinarlas,
- Cada píxel del frame buffer representará un voxel del objeto.

Para generar las rebanadas del objeto podemos realizar algunos pasos:

1. Determinar el tipo de modelo 3D del objeto, es decir, qué forma es la que se requiere tener, por ejemplo, representar una manzana.
2. Definir un volumen visible compuesto de al menos una parte del objeto en el espacio tridimensional.
3. Desplegar en el frame buffer la superficie de cualquier parte del objeto 3D encontrado con el volumen visible
4. Realizar cortes al objeto 3D para que los voxels de cada rebanada se almacenen en memoria; la rebanada almacenada es representada en el frame buffer, una parte de la memoria es matriz 3D que contiene la ubicación del plano escogido con lo que se mantiene la posición de la rebanada en el objeto 3D, este paso se debe repetir hasta que todo el objeto este voxelizado.

El método de la voxelización se refiere a la conversión de superficies poligonales en un conjunto de datos que representan volúmenes tridimensionales, se requiere el cálculo de la intersección del modelo poligonal con cada rebanada del volumen. Para poder realizar este algoritmo se deben cumplir con ciertos requisitos, algunos de los requerimientos para las superficies son las siguientes:

1. La superficie debe ser un polígono de malla cerrada sin frontera, esto nos permitirá distinguir entre la parte interna y externa del objeto para su voxelización
2. La superficie debe consistir de polígonos planos. Es un requisito indispensable para poder convertir los modelos de polígonos a triángulos.
3. El orden de los vértices y la orientación de los vectores normales deben ser consistentes. Significa que las normales de los vectores no cambien de dirección para caras adyacentes las cuales deben tener una orientación definida en sentido horario u antihorario, esto es crucial para poder identificar cual será la cara frontal y trasera.

El proceso para calcular las intersecciones entre el objeto poligonal y el plano se puede dividir en 4 pasos.

1. Se tienen un plano de corte con la misma posición y orientación que el de la imagen. Este plano de corte remueve la porción de las caras de la geometría, así que el interior del polígono se vuelve visible
2. Solo las caras traseras del objeto son rendereadas usando el color de fondo y el buffer de profundidad se actualiza simultáneamente. La sección. El resultado aun no es correcto porque las caras traseras son ocultadas por las caras frontales con respecto a al posición de la cámara de cruce.
3. Se renderiza por segunda vez, las porciones erróneas del paso dos son removidas dibujando las caras frontales otra vez con el color de fondo. La prueba de profundidad es requerida aquí para asegurar que solamente aquellas partes de las caras traseras son actualmente ocultas por las caras frontales
4. Hasta este paso ya se tienen los puntos de cruce correctos entre el objeto y el plano de corte. La siguiente rebanada del volumen puede ser creada cambiando el plano de corte a la siguiente posición y se repite nuevamente el paso dos hasta que todo el volumen es creado.

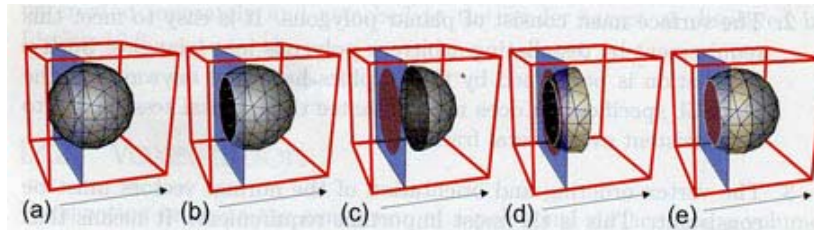


Figura 3.6: Algoritmo de Voxelización. Proceso para calcular la intersección entre superficies y un plano de corte.

3.3.2.2 Marching Cubes. [6]

El algoritmo de marching cubes fue desarrollado para aproximar superficies 3D en una maya de triángulos, por lo que permite la generación de modelos triangulares de superficies de densidad constantes a partir de un conjunto de datos volumétricos, como las tomografías, que nos permiten analizar las superficies tridimensionales a partir de las imágenes medicas, que permiten realizar diagnósticos, planificaciones o simulaciones quirúrgicas, esto permite a los médicos a entender mejor la anatomía que se puede presentar en las tomografías adquiridas.

Las superficies que se usan dentro del algoritmo de marching cubes también cuentan con ciertas características, como lo son:

- Son útiles para la representación de formas "suaves".
- Permiten el manejo de formas orgánicas y moleculares
- A cada punto del espacio le corresponde un valor
- La superficie queda definida por el conjunto de puntos con un determinado valor del campo.

La idea básica es la siguiente:

- Para cada celda a través de la cual pasa la superficie se crean pequeños polígonos aproximando a la superficie dentro de la celda
- La superficie intersecta los bordes donde los vértices cortan umbral
- Se tiene máximo una intersección

Al determinar los lugares del cubo por donde pasa la superficie se pueden generar triángulos que unan los puntos de intersección. Al unir todos los triángulos generados se obtiene la superficie buscada.

Los datos 3D obtenidos son procesados por capas y así se realiza el cálculo de los vértices de los triángulos usando interpolación. Se halla el gradiente a los datos originales, se normaliza, y se usa como base para dar iluminación a los modelos.

Marching Cubes es un algoritmo de alta resolución con un bajo costo computacional, es el algoritmo más conocido de creación de modelos triangulares. La idea básica es la siguiente. Se considera un cubo cuyos vértices están definidos por 8 voxels de los cuales son 6 mutuamente adyacentes, como se muestra en la siguiente figura:

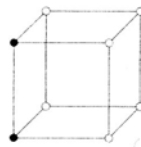


Figura 3.7: Cubo con vértices definidos por voxels.

El algoritmo procede inspeccionando cada cubo dentro de la imagen delimitada por una malla mientras trata de determinar en donde la superficie del objeto se intersecta con el cubo.

El algoritmo inspecciona cada cubo y atributos de la superficie. Teóricamente existen $256 = 2^8$ configuraciones diferentes de voxels blancos y negros concebibles dentro de un cubo. Como el cubo tiene simetría solo nos quedan 22 configuraciones de esas 22, 8 mas pueden ser omitidas debido a que son similares a alguna otra configuración, de esta manera solo se tienen 14 posibles configuraciones.

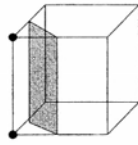


Figura 3.8: Intersección de una superficie y un voxel.

Estos casos se muestran en la siguiente imagen:

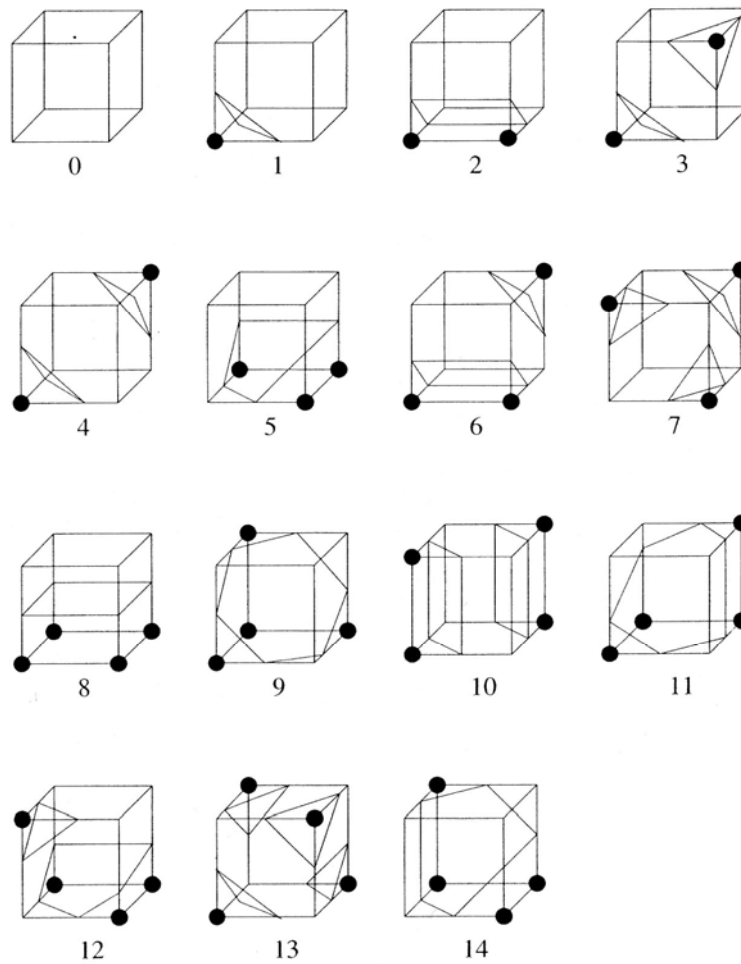


Figura 3. Posibles configuraciones del algoritmo Marching Cubes.

3.3.3 Modelo Teórico de Volume Rendering Directo [1]

El render en general es la interacción de la luz, objetos, y el medio entre ellos. Se describe típicamente con la absorción, emisión y dispersión, y tiene sus fundamentos físicos en la *Teoría de transporte de la Luz* [Krüger 1990a].

En general, se ignora la influencia de medios cambiantes, y no se consideran diferentes longitudes de onda o la difracción, igual que la dispersión, refracción y reflexión, para llegar al modelo estándar del Volume Rendering, el modelo *emisor de densidad* introducido por Sabella [1988].

Este modelo solamente considera emisión y absorción. Modela cada partícula de nuestro volumen como una minúscula fuente de luz atenuada por la absorción mientras viaja por el conjunto de datos volumétrico.

3.3.4 Algoritmo de Volume Rendering Directo: Ray Casting [1]

Este algoritmo es una variación del *ray tracing*. En ambos casos, rayos son lanzados desde el observador a través del plano de imagen y el conjunto de datos. A diferencia del *ray tracing*, el rayo no atraviesa una escena que contiene diversos objetos, sino un conjunto de datos volumétricos. Consecuentemente, no computa intersecciones con objetos, pues todos los voxels del conjunto de datos podrían estar contribuyendo. En cambio, el rayo muestrea el conjunto de datos a lo largo de su trayectoria. Dependiendo de propiedades especificadas por la función de transferencia, las muestras pueden estar contribuyendo, (es decir, opacidad mayor a cero) o no contribuyendo (opacidad siendo cero). Las posiciones de las muestras en el rayo dentro del volumen de datos dependen de la dirección del rayo y la tasa de muestreo, que dicta la distancia entre muestreos adyacentes en ese rayo. Debido a que en muchos casos dichas posiciones no se localizarán directamente en la posición de un voxel, es necesario aplicar una interpolación para obtener los valores de muestreo. El método típico es interpolación trilineal.

Otra gran diferencia entre este algoritmo y el *ray tracing* es la falta de rayos secundarios, pues los rayos no se refractan ni reflejan en los puntos de muestreo.

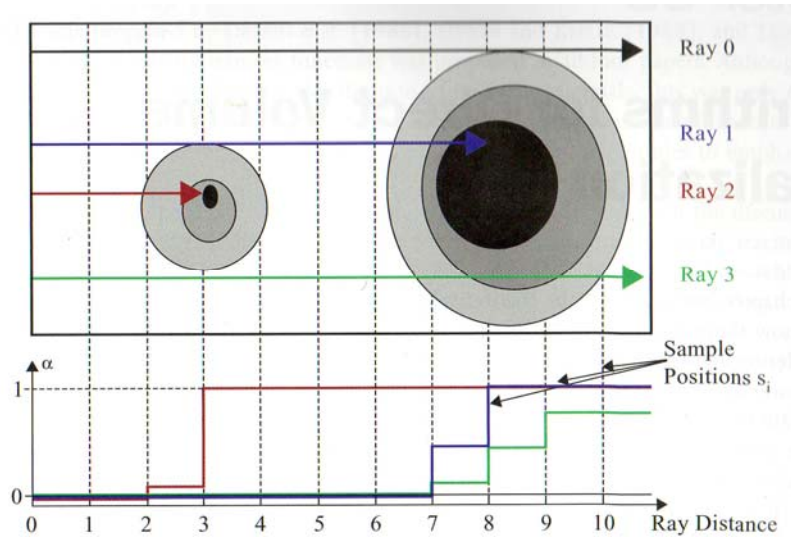


Figura 3.10: Ray Casting. Muestreo de acuerdo a la opacidad de un conjunto de datos.

Una similitud con el *ray tracing* es que se espera que al menos un rayo se lance por cada píxel del plano de imagen. Sin embargo, el teorema del muestreo nos dice que esto no es suficiente para evitar artefactos de muestreo. Por lo tanto, se deberían lanzar 2x2 rayos por píxel. Como esto incrementa la carga computacional cuatro veces, rara vez es llevado a cabo.

3.4 Amira Visualizador 3D

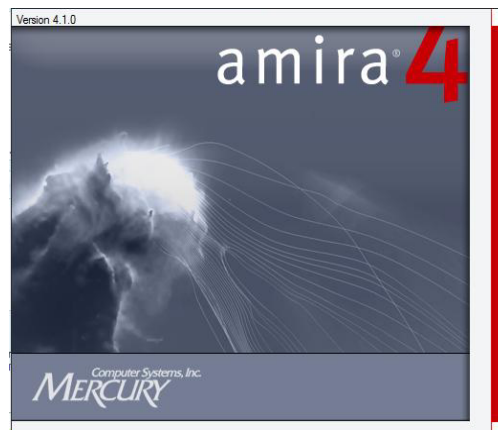


Figura 3.11: Amira. Visualizador 3D.

En la actualidad uno de los objetivos principales del desarrollo del software es el de poder visualizar y analizar datos volumétricos, los cuales son comunes en medicina, biología y microscopía

Para poder visualizar superficies tridimensionales es necesario contar con un programa de cómputo para la visualización y modelado de volúmenes, este programa se llama AMIRA, diseñado por un grupo de visualización científica en el instituto Zuse de Berlín, Amira se encuentra disponible como un producto comercial el cual cuenta con diversas extensiones

Con la ayuda de AMIRA los volúmenes pueden ser visualizados y segmentados, los modelos poligonales 3D pueden ser reconstruidos, y estos modelos posteriormente pueden ser procesados y convertidos en mallas formadas por tetraedros y con esto se logra una representación 3D muy completa.

Amira generalmente puede reconstruir volúmenes obtenidas de imágenes de cortes transversales de algún volumen, como pueden ser partes del cuerpo humano. Estos cortes transversales se caracterizan por estar compuestos de arreglos bidimensionales que proporcionan ciertos valores que pueden considerarse como coordenadas (x, y) , de esta forma si se apilan las imágenes se puede construir el volumen si se agrega la coordenada faltante de profundidad, z , de esta manera se logra tener coordenadas tridimensionales confinadas en un paralelepípedo, cada coordenada será representada por un voxel. [M7]

Algunas de las características de Amira son:

- Es especializado en la visualización médica y biológica.
- Fácil de usar por el hecho de que simplemente se necesitan un par de clicks para visualizar un elemento, esto ayuda a las personas que no cuentan con entrenamiento previo a familiarizarse con el software.
- Es un programa flexible y capaz de trabajar con distintos tipos de datos al mismo tiempo, además de realizar operaciones con módulos.
- Es interactivo debido a que los componentes en 2D y 3D pueden manipularse fácilmente por el usuario, a pesar de no estar familiarizado con el software.
- El usuario puede agregar nuevas entradas y salidas.
- Permite el manejo de programación por medio de scripts para simplificar tareas.
- Es multiplataforma, de esta manera puede trabajar indistintamente con hardware y sistema operativo utilizado.
- Usa técnicas de visualización tales como el direct volume rendering y basado en texturas, lo que permite optimizar la calidad de la imagen desplegada.

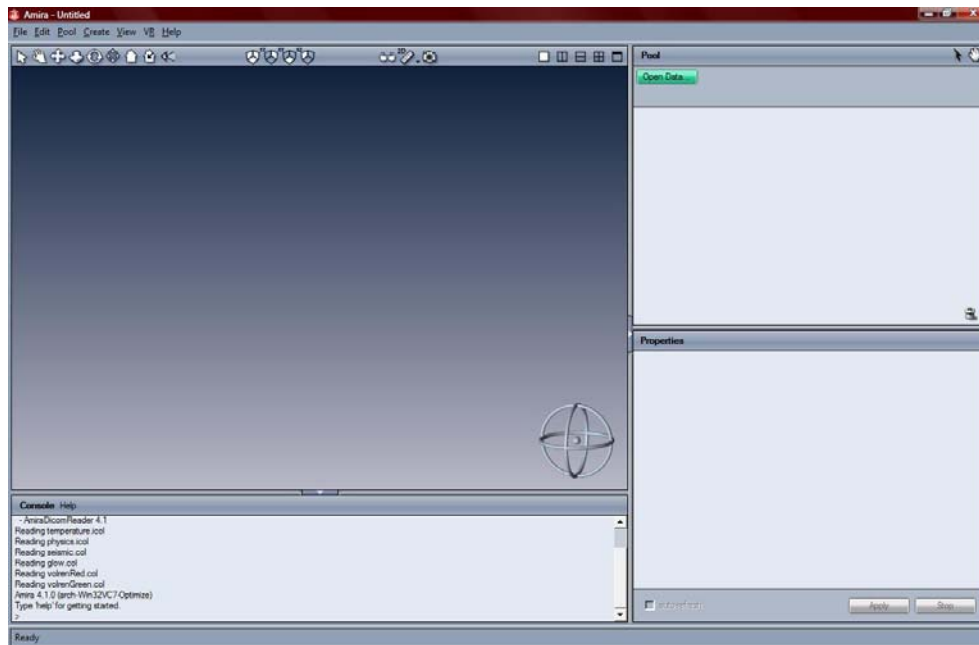


Figura 3.12: Interfaz de usuario, ventana principal de Amira.

3.4.1 Requisitos del sistema

Para poder aprovechar los componentes y las características descritas anteriormente se requiere tener el programa Amira en su versión 4.1, así como el ambiente de trabajo necesario, como lo es la tarjeta de video, la memoria utilizada, compilador C++ que sea compatible con el sistema operativo y la plataforma utilizada. En todos los casos es importante tener las cabeceras y librerías de OpenGL instaladas, esto permitirá el uso de estereoscopia.

A continuación se presentan, de una forma general, los requisitos mínimos de sistema para distintas plataformas.

1. Windows

Sistema Operativo: Windows 2000/XP/Vista.

Compilador: Microsoft Visual Studio .NET 2003 (VC++ 7.1) o superior.

Sistema Operativo: Windows XP 2003/Vista edición x64.

Compilador: Microsoft Visual Studio 2005 (VC++ 8.0).

2. Linux

Sistema Operativo: Red Hat Enterprise Linux 3.0 o superior.

Compilador: GNU gcc 3.2.3.

3. Mac OS X

Sistema Operativo: Mac OS X Tiger (10.4) o superior.

Compilador: GNU gcc 4.0.

4. *Sun Solaris*

Sistema Operativo: SunOS 5.8 (Solaris 8) o superior.

Compilador: Sun WorkShop 6.0 Update 2 (CC 5.3) o superior.

5. *SGI IRIX*

Sistema Operativo: IRIX 6.5.15 o superior.

Compilador: MIPSpro C/C++ versión 7.4 (64 bits) o superior.

6. *HP-UX*

Sistema Operativo: HP-UX 11.00.

Compilador: ANSI C/C++ versión 3.27 o superior.

3.5 Optimización de modelos 3D

La visualización científica ha permitido crear modelos 3D de diversas partes del cuerpo humano, aunque existe un pequeño problema, que estos modelos requieren de una gran cantidad de espacio de almacenamiento debido a la cantidad de polígonos que poseen.

En el área de la medicina suelen tenerse modelos 3D que requieren una gran cantidad de información por el hecho de que se requiere un cierto nivel de detalle en la imagen, de esta manera se puede dar estudiar correctamente la anatomía del cuerpo humano o quizás dar el diagnostico de algún tejido o hueso que pueda tener alguna enfermedad o deformación.

El nivel de detalle se logra con una gran cantidad de polígonos pero existen métodos que permite reducir esa cantidad de polígonos tratando de no alterar aquellos detalles que son importantes para los médicos, esta reducción de polígonos es conocida como optimización.

En computación se trata de optimizar el código, los gráficos, los sonidos, en general, todo aquello q utilice recursos computacionales y admisible de ser optimizado

La optimización de un volumen tridimensional permite aprovechar al máximo los recursos de un sistema, físicos y lógicos, por ejemplo: la tarjeta de video, memoria RAM, el espacio en disco duro, el sistema operativo y sobretodo el tiempo de ejecución y despliegue del modelo en el programa visualización Amira.

La base en la optimización de un modelo 3D reside en el número de polígonos (triangulares) y el número y tamaño de las texturas que utiliza. Por lógica, cuantos menos polígonos y menos texturas mejor, sin perder de vista la calidad que se busca en un modelo particular.

CAPÍTULO 4

ANIMACIÓN

4.1 Animación

La animación oficialmente está definida como el estado de estar vivo. Básicamente, lo que se hace a través de una animación es crear la ilusión de vida. A través de los avances que se han logrado en computación se puede decir que se tiene la capacidad de crear vida en tres dimensiones, dando a las animaciones la profundidad necesaria que se logra observando la animación, esto permite dar una experiencia a nuestro sentido visual de movimiento continuo. [10]



Figura 4.1: The ChubbChubbs (c) 2002 Sony Pictures Imageworks Inc. All rights reserved.

La animación envuelve la producción de una serie de imágenes que cuando son repetidas en una rápida sucesión, aparentan tener movimiento continuo, esta ilusión es producida por características psicológicas de nuestro cerebro, conocida como persistencia de la visión. [11]

Gracias a la persistencia de la visión podemos ver las imágenes en movimiento formadas a partir de diversos cuadros. Una parte del ojo humano, la retina, es capaz de retener momentáneamente cualquier imagen que recibe.

Para realizar cualquier tipo de animación se debe tomar en cuenta que siempre se trabajará con una acción principal generada a partir de un movimiento que dirige a la acción primaria, este es llamado movimiento de anticipación, como consecuencia de este movimiento se tiene un segundo movimiento llamado continuación (follow-through).

La unidad básica del tiempo esta determinada por el índice de grabación y proyección del cuadro individual de la película, imagen o video. Este índice se conoce comúnmente como cuadro por segundo o frame per second (fps). Si el tiempo entre las imágenes es muy largo entonces la imagen siguiente no permitirá dar la sensación de movimiento continuo. En un película se proyectan 24 imágenes distintas en un segundo (24 fps), en este caso una animación de 1 minuto requiere $24 \times 60 = 1440$ cuadros. [12]

Los cuadros por segundo cambian dependiendo del formato en el que se quiera ver, en el caso de un video PAL se maneja 25 fps, un video formato NTSC maneja 30 FPS, estas cifras son importantes a la hora de conseguir la ilusión de movimiento ya que el índice de retención de persistencia de la visión

es aproximadamente una décima de segundo, mucho mayor que los índices de proyección fps para película o video. [12]

Dentro del proceso de animación encontramos diversas técnicas, desde las más tradicionales hechas con lápiz y papel, hasta las técnicas utilizadas en la actualidad que están basadas en el uso de la computadora y dispositivos electrónicos tales como cámaras de vídeo, escáner y software especializados 2D y 3D.

La animación es especialmente apropiada en Visualización de Información para representar el cambio en el transcurso del tiempo y más específicamente para:

- Representar efectivamente la causalidad.
- Los actos que expresan comunicación o flujo.
- La reorganización o reestructuración se adapta bien a la animación, siempre que la complejidad de las interacciones no sea muy alta.
- Secuencias de acciones espaciales complejas.

Para darle un mayor realismo a la animación se puede hacer uso de exageración y dinamismo para hacer cierto que cada acción es fácilmente entendible y legible para la audiencia, esto es especialmente importante en los juegos de video para asegurar que el jugador sea capaz de decir que es lo que esta pasando durante todo el tiempo y así poder tomar decisiones necesarias para lograr sus metas. Un arte marcial debe ser capaz de patear o golpear tan rápidamente que en usuario tenga dificultad de ver cada movimiento claramente.

Como podemos observar la animación deja de ser un acto mecánico simplista de crear una secuencia de imágenes en movimiento y se convierte en una representación bastante semejante a una actuación y, como toda buena actuación, requiere paciencia y un enfoque crítico para perfeccionar el arte de la representación.

4.1.1 Timing

Un término básico que es importante dentro de la animación es el de timing. El timing de la animación al igual que los objetos animados y los propios personajes, se puede estilizar o ser naturalista, es decir, darle un sentido de realismo según el efecto requerido.

Todo el timing de la animación esta determinado por la secuencia de imágenes y su posición relativa con las imágenes anteriores y siguientes que aparecen en la pantalla. Las acciones rápidas o lentas se consiguen por las variaciones en las relaciones espaciales entre imágenes individuales dentro de una secuencia animada. No importa si estas imágenes son dibujos (Figura 4.2), modelos en 3D, imágenes generadas por computadora. Cuanto más cerca estén

entre sí las siguientes imágenes en su posición en la pantalla, más lenta será la acción; cuanto mayor sea la distancia entre las imágenes, mayor será la velocidad de la acción. [12]

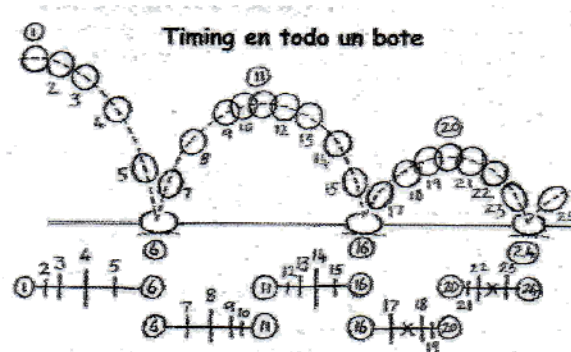


Figura 4.2: Timing del bote de una pelota.

En general es un identificador de velocidad y peso a lo largo de una acción en particular o secuencia de algún movimiento rápido, si el tiempo no es manejado correctamente entonces una animación puede parecer estar en cámara lenta o demasiado rápida, afectando la acción que se quiere animar ya que no se verá como es deseado. El timing también es referido como el ritmo (pacing) de una animación, especialmente cuando se envuelve el movimiento de un personaje.

4.2 Animación mediante Cinemática

La cinemática es el estudio de los cuerpos en movimiento sin tomar en cuenta las causas que lo producen por lo que se limita al estudio de la trayectoria en función del tiempo. Los conceptos básicos de la cinemática son el tiempo, conjunto de todos los instantes posibles, y el espacio, conjunto de todas las posiciones posibles. Esta ciencia puede ser aplicada a la computación para realizar animaciones.

En el área de la computación, la cinemática se refiere al método por el cual la computadora simplifica los cálculos requeridos para mover los elementos o miembros de un personaje, el cual es un proceso que de otra forma es difícil de manejar. Hay dos tipos, cinemática directa y cinemática inversa. [13]

La cinemática directa e inversa son técnicas de manipulación de esqueletos usadas para controlar el movimiento de los nodos a lo largo de una cadena. La cinemática directa es un método de manipulación de nodos en orden descendente de la jerarquía, mientras que la cinemática inversa afecta a los nodos en sentido ascendente de la jerarquía de la cadena de nodos.

4.2.1 Cinemática Directa

La cinemática directa es usada en programas de animación, es importante en el proceso de creación de huesos ya que muchas veces es necesario hallar la posición y orientación final de un nodo a partir de ciertos ángulos que se producen durante algún movimiento en una articulación.

Cuando se anima una secuencia, por ejemplo, el movimiento de una mano, la parte superior del hueso en una cadena que utiliza cinemática directa, el hombro, gira hacia delante, y todos los otros huesos siguen el movimiento de igual forma, limitados solamente por atributos de posición y movimiento, estos atributos son límites temporales, pero de otro modo todos los huesos de la mano importan el movimiento (o cinemática) desde el hueso padre. [13]

Para poder realizar los cálculos necesarios que ayudan a determinar posiciones, rotaciones y tamaños de diversas geometrías, que se encuentran generalmente unidas por articulaciones en una cadena de nodos (huesos).

La animación directa se crea al producir un cuadro tras otro en secuencia, este tipo de animación se realiza al crear los momentos importantes de una acción primero y luego interpolar los cuadros intermedios.

Algunas ventajas son:

- La capacidad que tiene de permitir eliminación de cuadros dentro de la secuencia
- Es una forma muy eficiente de generar animación.
- Los cambios en el timing de la animación se consiguen fácilmente utilizando este método.
- Se pueden crear una valoración bastante fiable del desarrollo de la acción, sin tener que crear todos los dibujos.
- Fomenta un enfoque hacia la animación vivo y lleno de energía.
- Esta libre del exceso de estructuración de la animación.
- Permite manejar elementos por separado que tienen su propio timing.

[12]

Algunas desventajas son:

- De no crearse buena animación podría parecer un poco rígida y en cierto sentido poco natural.
- Las correcciones no se pueden realizar fácilmente al añadir cuadros dentro de una secuencia existente.
- Es más difícil llegar a puntos específicos dentro de una secuencia y puede haber una tendencia a permitir que la duración de escenas específicas aumente o se reduzca, lo que puede causar más dificultades al director que trata de ajustarse al timing inicial de la animación.

[12]

Una aplicación de cinemática directa se observa en el desarrollo de video juegos para realizar el cálculo del tiro parabólico para el motor de física, por ejemplo, el juego de video *Snails*, en el que se aprecia claramente en la siguiente figura:



Figura 4.3: Uso de cinemática directa. *Snails*, PDAmill Game Studios, 2002-2008 ©. [M8]

4.2.2 Cinemática Inversa

Para muchas aplicaciones en tiempo real las acciones de los personajes serán creadas antes que la aplicación este en ejecución y el código del juego interactivo seleccione la mayoría de las acciones para alguna etapa del juego. Esto provee un estilo de juego efectivo, pero ¿Qué pasa cuando un personaje necesita llegar un poco más lejos que la acción almacenada para tomar un objeto? Sería bueno si la posición final de la mano del personaje pudiera ser determinado solo dando la posición en el mundo para la mano y permitiendo al código determinar las orientaciones de la mano, antebrazo, biceps, hombro, torso y caderas que son necesarios para alcanzar su objetivo. La cinemática inversa permite una solución y puede ser usada para la animación de pies y otras partes de los personajes. [14]

La técnica consiste en calcular las transformaciones necesarias en las articulaciones de una estructura, de modo que su extremo se coloque en una posición determinada, es decir, en encontrar el gesto que deben adoptar las diferentes articulaciones para que el final del sistema articulado llegue a una posición concreta. [M9]

La cinemática inversa requiere un poco mas de matemáticas que la cinemática directa. El proceso matemático inicia cuando se posiciona el final de un hueso, es decir, con el uso del último hueso. La computadora examina todos los atributos de posiciones de los huesos y los nodos desde el último hueso hasta la rotación más alta del hueso de mayor jerarquía. Esto calcula una solución que da como resultado el correcto movimiento de todos los huesos que conforman la cadena. [13]

La medicina es otra área en la cual se puede aplicar la cinemática inversa para poder realizar los movimientos de los huesos de las manos o los pies al momento de realizar una animación por computadora.

El principio de una solución para cinemática inversa para un problema de movimiento es que, dada una posición en un espacio 3D y el punto final de una cadena ligada, también llamado ultimo hueso (effector), determina las rotaciones que tiene que aplicarse a la cadena ligada con el propósito de que el ultimo hueso (effector) alcance su objetivo. Cada nodo puede ser rotado en una o mas direcciones. Un nodo simple que rota en sentido de los ejes coordenados es descrito como que tiene 3 grados de libertad. Para un objeto que tiene completa libertad de movimiento debe tener 6 grados de libertad. Por esta razón no siempre se puede alcanzar el objetivo. [14]

Para la cinemática inversa la posición del punto final es conocida y necesitamos encontrar los ángulos entre los nodos de una cadena.

Hay dos aproximaciones para resolver el problema de la cinemática inversa

- **Analítica:** Requiere de trigonometría o álgebra de matrices
- **Iterativa:** Es un poco mejor si hay muchos huesos (nodos) así como grados de libertad.

4.2.2.1 Aproximación Analítica [M10]

Si hay solo dos o tres ligas entonces puede resolverse analíticamente. Posiblemente puede dibujarse una mano con los ángulos mostrados después resolver los ángulos usando geometría, un buen conocimiento de identidades trigonométricas ayudaría. El problema es que no es una aproximación general.

Otra aproximación analítica es representar cada rotación y traslación de una articulación por una matriz. El punto final se dará por todas esas matrices multiplicadas juntas, así que necesitamos resolver el sistema de ecuaciones matricial. Después encontrar lo que la rotación representa en cada matriz.

Pueden existir muchas soluciones o puede no haber ninguna. Hay muchas formas de llegar a un punto dado o puede estar fuera del alcance. Si hay muchas soluciones entonces se necesitara aplicar restricciones adicionales. Para las articulaciones humanas pueden manejarse ciertos límites

4.2.2.2 Aproximación iterativa [M10]

Esta es una aproximación más general para programación de cadenas complejas. Inicia con un nodo en cualquier posición, después se mueve cada nodo en turno así que el movimiento toma el punto final hacia el objetivo.

Inicia con el nodo más cercano al punto final, se rota el nodo así que el punto final actual se mueve hacia el punto final requerido. Luego hace lo mismo con el siguiente nodo hacia la base y hasta que la base es rotada. Se mantiene repitiendo esto, hasta que el punto final esta cerca al punto final requerido o si después de varias iteraciones no se logra mover hacia el punto requerido.

Si se hace uso de cinemática inversa las animaciones pueden extenderse lo cual no debe ser un obstáculo para su interactividad. El uso combinado de la cinemática directa y la cinemática inversa permite una mayor flexibilidad en las animaciones.

Con el uso de nodos se permite que algunas partes del modelo parezcan más flexibles que otras al usuario. Por ejemplo, las siguientes imágenes muestran la simulación de una extremidad robótica donde se aprecian los nodos en los que se aplican las transformaciones necesarias para lograr que éste realice el movimiento correcto con ayuda de cinemática inversa.

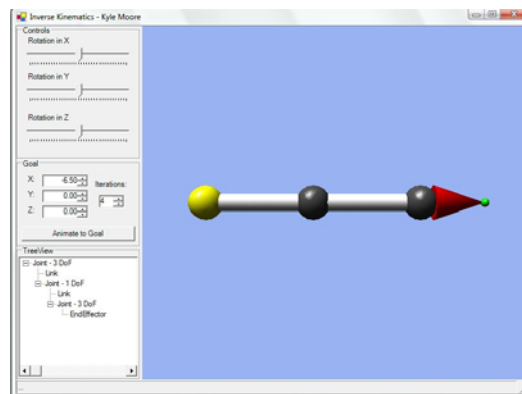


Figura 4.4: Aplicación de cinemática inversa. Posición Inicial. [M11]

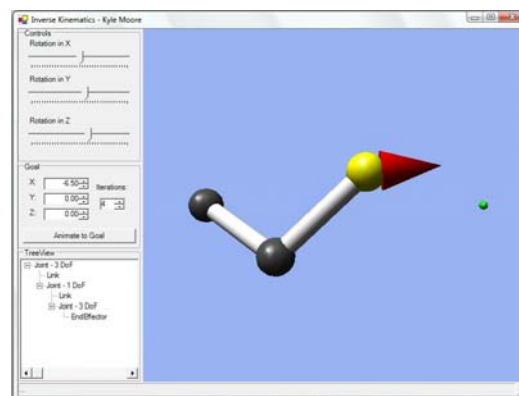


Figura 4.5: Aplicación de cinemática inversa. Posición Final. [M11]

4.2.3 Curvas Cinemáticas

Una variante del uso de cinemática inversa es llamada curva cinemática. Esta usa una curva que corre a lo largo de una serie de articulaciones o nodos de un esqueleto; Permite el manejo del flujo de la curva afectando el flujo de los nodos. Esto es usualmente más flexible para formas como tubos, tales como los nodos de una columna o tentáculos de un pulpo. Cada elemento usado en una animación puede tener su propia curva para realizar el movimiento que le corresponde y de esta manera se puede generar la gráfica de todos los elementos en distinta posición y tiempo, como vemos en la siguiente figura:

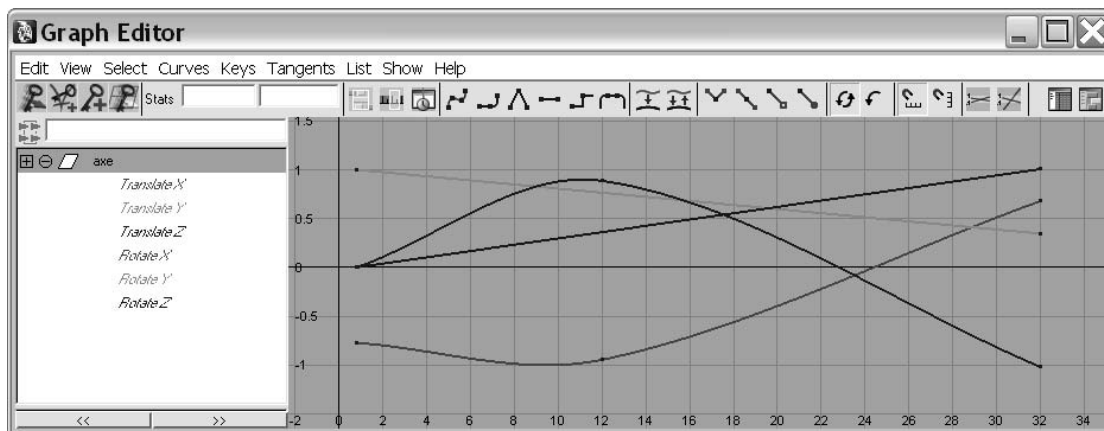


Figura 4.6: Curvas cinemáticas de distintas geometrías generadas en un periodo de tiempo.

4.3 Animación mediante Keyframes (cuadros clave)

La idea básica de la animación mediante keyframes es la de realizar cambios de valores a las matrices de transformación en un periodo de tiempo determinado.

Desde los inicios de la animación 2D se ha trabajado con el término keyframe, que se refiere a un tipo de trabajo con imágenes que son consideradas "pose a pose", es decir, una imagen inicial y una imagen final del movimiento deseado, en las que usualmente se tenía a un grupo de dibujantes que se hacía cargo de dibujar cuadros clave, mientras otros se dedicaban a dibujar las imágenes que debían haber entre estos cuadros clave en un proceso llamado *in-betweening*, o *tweening* en estos días.

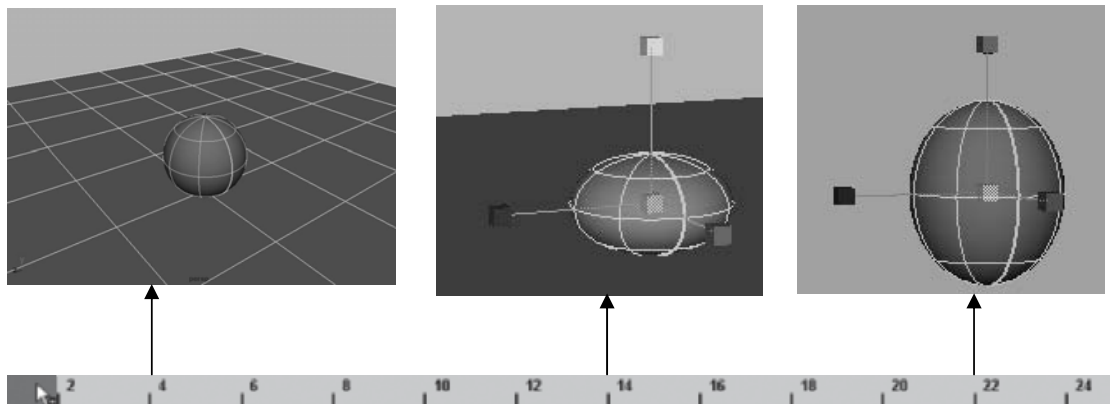


Figura 4.7: Animación del bote de una pelota mediante Keyframes.

Los sistemas basados en este tipo de animación dependen de ciertas matrices que permiten el manejo de transformaciones en las geometrías las cuales permiten operaciones sobre los valores de posición (traslación), tamaño (escalamiento) y rotación de los objetos 3D. [15]

Utilizando el tipo de animación por cuadros clave es necesario contar con ciertos elementos importantes, tales como lo el cuadro origen y el cuadro destino. Éste último puede ser utilizado como cuadro origen de la siguiente parte de la animación, mediante una técnica matemática llamada interpolación con la que se pueden obtener los cuadros que se encuentran entre un cuadro clave y otro, necesarios para realizar el movimiento que se desea plasmar. Esta operación calcula los valores intermedios de las matrices de transformación.

La forma más simple de interpolación es la interpolación lineal, éste método parte del hecho de que en las animaciones hay una relación entre dos parámetros, en este caso, entre los cuadros de animación de los cuales se puede determinar los valores intermedios de cada matriz de transformación para lograr animaciones suaves.

Esta técnica es de las más usadas dentro de la animación por computadora. Sin embargo, una desventaja es que puede resultar muy tedioso crear muchos cuadros clave para mover una geometría relativamente sencilla de una posición a otra.

El uso de interpolación nos ofrece como ventaja poder crear manualmente cada uno de los cuadros de la animación pero tampoco es necesario poner varios cuadros clave, debido a que se volverá complejo realizar correcciones posteriores así que se debe tratar de poner solo aquellos cuadros fundamentales para el movimiento que se desea proyectar.

4.4 Animación Esqueletal en Maya

Este tipo de animación es el resultado de mezclar diversas técnicas de animación tales como cinemática inversa, cinemática directa y keyframes.

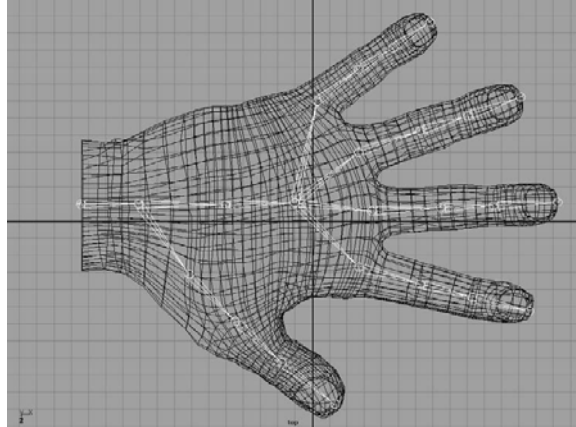


Figura 4.8: Esqueleto de una mano generado con cadenas de nodos en Maya.

En nuestro cuerpo podemos ver que los huesos y los músculos se encuentran en constante movimiento, de tal forma que si un hueso se mueve, ciertas partes de nuestro cuerpo se moverán también, es decir, los huesos son la estructura base que nos permite realizar el trabajo cotidiano.

En computación gráfica un esqueleto es un armazón, el cual a su vez es una cadena que simula los movimientos de personajes virtuales como un ser humano o animal, creado a partir de un modelo 3D que maneja la geometría cuando los huesos se mueven, el esqueleto se puede insertar en el modelo o puede ser ligado a una geometría para moverse junto con ella. Los huesos del esqueleto son animados regularmente con transformaciones (rotaciones y/o traslaciones), las cuales mueven las partes de las geometrías a las que se encuentran ligadas.

Maya ofrece una herramienta útil para generar cadenas jerárquicas de nodos (esqueleto) que permiten ligarse a las geometrías 3D (piel). Con esto se puede aplicar a cada hueso una transformación tridimensional para generar una animación, un ejemplo se muestra en la figura 4.9.

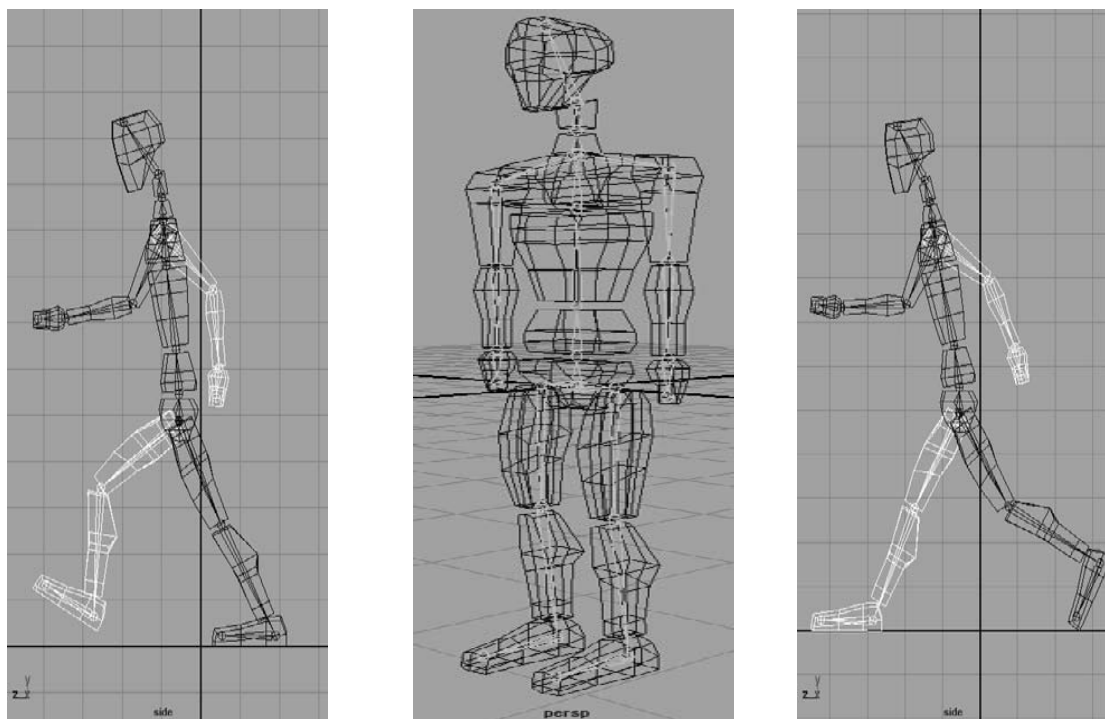


Figura 4.9: Animación Esqueletal de un cuerpo humano en Maya.

CAPÍTULO 5

**BIOMECÁNICA DE LA COLUMNA
VERTEBRAL Y MIEMBRO INFERIOR
HUMANOS: ANÁLISIS**

En los últimos años las investigaciones realizadas en diversas áreas científicas son de gran importancia para poder comprender y analizar estructuras o modelos con la ayuda de las tecnologías creadas para la visualización de datos a través de la computadora. La medicina es una de las áreas científicas que ha hecho uso de estas tecnologías que, por medio de la manipulación de órganos o partes anatómicas, muestra a los médicos el funcionamiento de algunas secciones del cuerpo humano, con el fin de enseñar o realizar técnicas de cirugía.

La representación de modelos 3D constituye un recurso de gran utilidad en el análisis y comprensión de estructuras complejas, en el campo de la anatomía humana, esta ha sido favorecida gracias a los avances tecnológicos que actualmente los médicos, en conjunto con ingenieros, han realizado en el proceso de adquisición de imágenes para diagnóstico, y el desarrollo de herramientas educativas de visualización de datos que nos permiten realizar reconstrucciones de modelos 3D con imágenes adquiridas, ya sea a partir de tomografías (TAC), resonancias magnéticas (MRI) o tomografía de emisión de positrones (PET), que permiten obtener imágenes del interior del cuerpo humano o variaciones en la fisiología del cuerpo. [16]

5.1 Caso Práctico: Biomecánica de la Columna Vertebral y Miembro Inferior

La Facultad de Medicina de la UNAM, a lo largo de los años ha impartido clases de anatomía humana, que es una base fundamental, para crear médicos de calidad, necesarios para combatir la creciente demanda de soluciones médicas a la aparición y desarrollo de múltiples enfermedades.

Gracias a la investigación médica y el desarrollo de la tecnología se han podido conjuntar la medicina con la computación a través de diversas áreas tales como: procesamiento digital de imágenes, animación, realidad virtual y cómputo gráfico. Esto ha permitido la aparición de la visualización científica, como un método alternativo de análisis médico.

La visualización científica es responsable de la transformación de datos obtenidos a partir de dispositivos electrónicos en imágenes que sirven a los médicos para entender y diagnosticar problemas internos del cuerpo humano.

Dentro de la UNAM se cuenta con la sala Ixtli de realidad virtual que permite generar un sentido de profundidad tridimensional a través de su infraestructura, destinada a la creación de proyectos de investigación que permitan lograr un avance en la docencia

En base a estos desarrollos la Facultad de Medicina del UNAM, ha decidido valerse de la visualización científica así como la infraestructura de la sala Ixtli como una herramienta de apoyo para la enseñanza de la anatomía humana.

5.2 Investigación Preliminar

Antecedentes: Cráneo Desarticulado

La Facultad de Medicina de nuestra Universidad, ha realizado diversos proyectos de visualización científica como parte de algunas medidas de apoyo a la docencia, entre las materias que reciben apoyo de este tipo de tecnologías se encuentra la asignatura de Anatomía. Anteriormente se realizó el proyecto denominado Cráneo Humano Desarticulado, el cual además se proyectó en la sala Ixtli de la Dirección General de Servicios de Cómputo Académico (DGSCA).

El proyecto del cráneo desarticulado fue el resultado de un año de trabajo, en el que se desarrollaron los modelos 3D de la cabeza humana y con él los estudiantes de primer año de la Facultad de Medicina toman clase de anatomía con el apoyo de las imágenes tridimensionales del cráneo que, además, logran la separación de cada parte para luego estudiarlas una a una.

El proyecto se llevó a cabo con la finalidad de propiciar la superación académica de los alumnos en niveles de pre y posgrado, plantear soluciones a la desvinculación que existe en los cursos de anatomía humana entre la teoría y la práctica mediante este programa interactivo, que permitirá al usuario comprender la cabeza ósea para facilitar a los alumnos la comprensión de su morfología y la articulación entre los diversos huesos que la conforman, así como la biomecánica de la mandíbula. [M12]

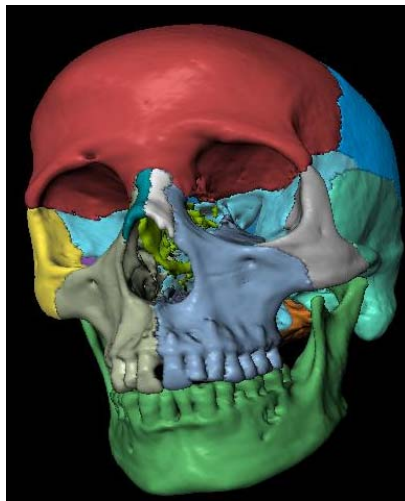


Figura 5.1: Cráneo Desarticulado.

Asimismo, se pretende lograr el aprendizaje significativo de la anatomofisiología humana, mediante la manipulación de las estructuras que conforman el cuerpo humano y dar acceso a los alumnos a las nuevas tecnologías, para apoyarlos en su formación integral, además de fomentar el interés de los estudiantes en temas que parecen difíciles.

Una vez que se comenzó a usar el programa en las clases de anatomía, se realizó un análisis de la efectividad de este programa como método de enseñanza alternativo, para esto, los profesores de la Facultad de Medicina de la UNAM, partieron de la siguiente hipótesis:

En la asignatura de Anatomía Humana, los alumnos que revisen el tema de "cabeza y cuello", empleando como apoyo el programa de cráneo en realidad virtual, tendrá mayor porcentaje de aciertos en el tema durante el examen departamental. [17]

Para obtener los datos se realizaron cuestionarios a los alumnos de la Facultad de Medicina, con un total de 40 grupos de distintos turnos, manejando la metodología siguiente:

- Selección al azar de 19 grupos de la generación 2006-2007.
- Presentación a los alumnos del programa de cráneo:
 - Antes de revisarlo en clase
 - Después de revisarlo
 - Entre las 12:00 y 15:00 hrs.
 - 3 sesiones semanales (90 días)
- Comparación de los porcentajes de aciertos alcanzados por grupo en el tema de "cabeza y cuello".

	Sin Programa	Con Programa	
Grupos	21	19	
	Matutino	Vespertino	NUCE ¹
Con Programa	16	2	1
Sin Programa	9	8	4

Los estudiantes que tuvieron acceso a la herramienta desarrollada mostraron un mejor desempeño en los temas relacionados, al final esto se vio reflejado en sus calificaciones, esto se puede observar con las tablas siguientes, donde se muestran las comparaciones con generaciones anteriores, así como los promedios globales de cada grupo seleccionado, que tuvieron o no acceso a esta herramienta.

Tabla 5.1: Antecedentes de Promedios (1er departamental / generaciones)

Generación	2003	2004	2005	2006
Promedio 1er departamental	58.7	59.7	66.2	67.2

¹ Núcleos de calidad educativa.

Tabla 5.2: Comparación Promedios 2005-2006

	Examen total	Tema de Cráneo
2005	66.2	69.6
2006	67.2	71.6

Tabla 5.3: Análisis de Aciertos en el Tema

	Promedio	Máximo	Mínimo
Con Programa	73.41	86.9	57.0
Sin Programa	69.69	83.6	57.9

Tabla 5.4: Comparación Promedios Grupos / Profesor (Si vieron programa)

Grupo	1103	1112	1114	1115	1117	1118	1125	1129	1136	1137	Promedio %
2005	71.6	69.3	65	76.7	72.9	72	70.3	56.4	60	68.3	68.2
2006	80.8	73.2	78.9	76.4	65.5	84.3	63.6	57	67.4	75.5	72.6

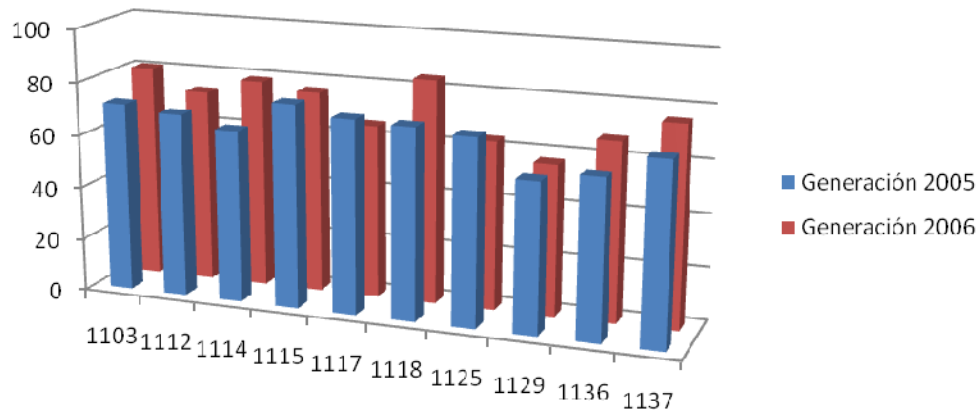


Figura 5.2: Comparación de promedios de los grupos que si vieron el programa de las generaciones 2005 y 2006.

Tabla 5.5: Comparación Promedios Grupos / Profesor (No vieron programa)

Grupo	1105	1108	1111	1119	1127	1128	1134	1135	1138	1133	Promedio %
2005	73.8	70.4	70.4	65.5	70.9	65.8	63.3	66.5	71.3	76	69.4
2006	87.5	72.5	74.6	73.4	70.9	62.8	63.4	61.7	83.6	60.1	71.0

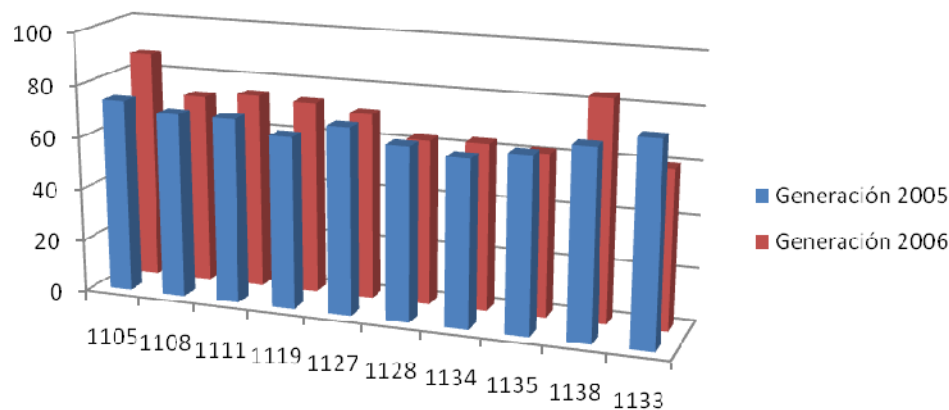


Figura 5.3. Comparación de promedios de los grupos que no vieron el programa de las generaciones 2005 y 2006.

Para realizar estas comparaciones, las personas encargadas de realizar estas estadísticas tomaron en cuenta el tipo de examen, además del orden y dificultad de las preguntas de los exámenes empleados, que exploran el tema de cráneo.

Como podemos ver los resultados muestran un ligero aumento en el número de aciertos en el examen departamental de los alumnos que fueron apoyados con el programa del cráneo. Debido a este hecho se decidió continuar con el proyecto para poder reconstruir un esqueleto humano completo, además de realizar la animación de los movimientos anatómicamente correctos, en este caso la siguiente parte fue la reconstrucción y animación de la columna vertebral y el miembro inferior.

Sala Ixtli (observatorio de visualización). [M13]

Uno de los objetivos del proyecto es que una vez terminado pueda ser utilizado por el personal docente de la Facultad de Medicina para poder impartir clases a los alumnos que cursen materias relacionadas con la anatomía humana, por este motivo se considera adecuado conocer la estructura, distribución y equipo de computo usado dentro de la sala Ixtli, que ha sido diseñada para visualizar y simular objetos complejos e imágenes en tercera dimensión.

Como resultado de la puesta en operación del observatorio de Visualización Ixtli en marzo de 2004, la facultad de medicina decide incorporar la realidad virtual a la actividad docente. Este observatorio es un instrumento con tecnología de punta que permite analizar e interactuar con imágenes complejas en tercera dimensión, mediante un sistema de realidad virtual inmersiva.



Figura 5.4: Sala de Visualización iXtli.

Para lograr que el cerebro humano logre una sensación de profundidad es necesario producir imágenes diferentes para el ojo derecho y el izquierdo. Estas imágenes se proyectan en forma alternada a gran velocidad. Los lentes de cristal líquido bloquean la visión de los ojos al seguir la sincronía proporcionada por la computadora; de esta forma, el ojo recibe sólo la imagen que corresponde, de manera tan rápida que engaña al cerebro y le hace creer que ve una sola.

Las imágenes son proyectadas en una pantalla semicilíndrica que cubre gran parte del campo de visión del espectador para realzar y mejorar las representaciones de los diferentes proyectos de investigación con el fin de comprender mejor la realidad y los resultados de las investigaciones.

El elemento clave de la sala es su sistema de Realidad Virtual Inmersiva, cuyo propósito es lograr que los usuarios tengan la sensación de encontrarse dentro del mundo creado por computadora, esto se produce al generar, en tiempo real, imágenes estereoscópicas que el usuario percibe con profundidad y que, además, responden a sus órdenes o movimientos. Las imágenes calculadas en tiempo real se generan a partir de un modelo tridimensional almacenado previamente en el equipo gráfico de alto rendimiento que está a disposición de la sala.

Nuevas imágenes se generan siguiendo las órdenes y movimientos de los usuarios, los cuales, son capturados por medio de dispositivos de rastreo de movimiento, guantes o ratón tridimensionales. Asimismo, se cuenta con un sistema de sonido que proporciona sensaciones auditivas relacionadas con el ambiente creado.

Características

CUPO

El espacio de iXtli está diseñado para albergar hasta 42 personas utilizando lentes estereoscópicos.

PROYECTORES

Las imágenes son generadas por tres proyectores Christie Digital Mirage 2000, basados en tecnología de alta resolución SXGA (1280x1024). Cada proyector produce una tercera parte de la imagen y utiliza equipos de corrección de geometría y mezcla de imágenes, para producir una sola imagen de resolución 3520 x 1024, que cubre la totalidad de la pantalla.

PANTALLA

Las imágenes son proyectadas en mono o en estéreo en una pantalla semicilíndrica de 140 grados, que mide 8.90 m de largo por 2.55 m de ancho. La forma cilíndrica permite cubrir gran parte del campo de visión del usuario lo que le ayuda a sentirse dentro del ambiente virtual.

DISPOSITIVOS

La sala cuenta con un sistema de rastreo de movimiento InterSense IS-900 para detectar con precisión la posición y orientación de las partes del cuerpo, en relación con el modelo visualizado. Se tienen dos sensores, uno para el rastreo del movimiento de la cabeza y otro que rastrea la posición y movimientos de un mouse tridimensional o wanda. Además, se cuenta con un guante inalámbrico 5DT Data Glove, con dos sensores por dedo, para medir tanto la flexión como la abducción entre los dedos.

AUDIO

La sala iXtli posee un sistema de sonido de alta tecnología: Equipo Dolby Surround 5.1 que a través de tres bocinas delanteras, dos bocinas traseras y un sub-woofer, envuelve al público con sonido tridimensional. Está equipada para reproducir formatos CD, DVD-A, Wav y MP3, entre otros.

CÁMARAS

iXtli cuenta con tres cámaras robóticas en la parte superior de la pantalla que pueden integrarse a un sistema de videoconferencia o de utilizarse como dispositivos para interacción con algoritmos de visión por computadora

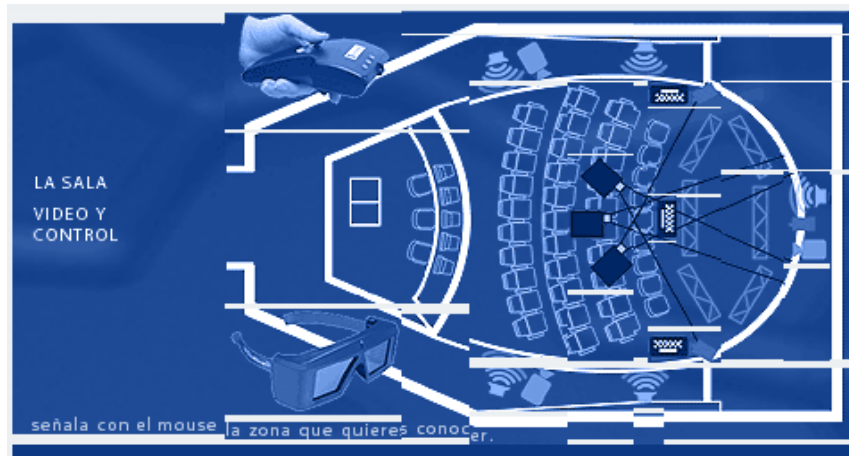


Figura 5.5: Distribución de la sala Ixtli. [M13]

5.3 Análisis

Actualmente los gráficos 3D ofrecen la posibilidad de crear ambientes virtuales en una computadora, lo cual unido a la visualización científica permite a los investigadores entender nuevos problemas.

Esto es posible gracias al avance en el desarrollo de algoritmos de visualización, lenguajes de programación orientados a objetos y programas de modelado 3D que ofrecen la posibilidad de crear software de mejor calidad.

La Realidad Virtual Inmersiva encuentra su aplicación en una importante gama del conocimiento humano, en las áreas científicas, sociales, de humanidades y las artísticas, es por eso que también puede ser aplicado a la medicina actual, como una herramienta auxiliar para los médicos.

El caso práctico que desarrollamos para este trabajo trata de un modelo tridimensional de la biomecánica articular de la columna vertebral y los miembros inferiores humanos. El objetivo fue crear dicho modelo lo más apegado a la realidad posible, con ayuda de especialistas en el campo, para ser utilizado en la actividad docente de la Facultad de Medicina.

Nuestro trabajo radica en desarrollar un segundo proyecto (el primero fue el del Cráneo Desarticulado) en el cual se modelará la Biomecánica de la Columna Vertebral y Miembro Inferior de tal manera que se pueda apoyar con una nueva herramienta de Visualización Científica las labores docentes que en la Facultad de Medicina se llevan a cabo. Las metodologías y conocimientos necesarios para llevar a cabo este objetivo significan una preparación en diversas áreas de Ingeniería en Computación.

Con el antecedente del proyecto del Cráneo Desarticulado que ya se utiliza activamente en la enseñanza de anatomía en la Facultad de Medicina, se cree que éste proyecto puede ser igual de útil debido a que se cumplirán

algunos los objetivos planteados inicialmente por la Facultad de Medicina, entre ellos se encuentran:

- Crear una herramienta auxiliar para la enseñanza de la anatomía humana.
- Practicar la docencia asistida por la realidad virtual generada por computadora.
- Desarticular cada uno de los huesos de un mismo elemento (columna y miembro inferior), con la mínima pérdida de información entre sus fronteras.
- Conservar un alto nivel de calidad en las geometrías extraídas ya que esto fomenta la cercanía con la realidad.

PROBLEMAS

Algunos de los problemas encontrados al iniciar el proyecto fueron:

- El formato DICOM que se maneja en las imágenes médicas.
- En las tomografías adquiridas inicialmente se puede observar que los huesos están sobre una plancha que sirve como soporte para acomodar los huesos.
- Se debe contar con un software que permita el apilamiento de las imágenes para poder generar el volumen.
- Realizar el procesamiento de las imágenes originales.
- La búsqueda del software adecuado para poder cambiar el formato de las geometrías para que tanto programa de visualización como el de animación puedan trabajar correctamente con los modelos.
- Un software adecuado para elaborar la animación es primordial.
- La falta de conocimiento en el área médica por parte nuestra.
- Tomar en cuenta que las geometrías originales contienen una gran cantidad de polígonos así que se debe lograr la optimización de las geometrías para que el proyecto pudiera ser usado en la sala Ixtli, y de ser posible en un equipo de cómputo normal.
- Los recursos de Hardware con los que cuenta la sala iXtli para el correcto despliegue del modelo 3D final.

METODOLOGÍA

La metodología general para ofrecer una solución de la situación planteada en el desarrollo de este proyecto, se realizó en base a un esquema típico para resolver los problemas de ingeniería.

➤ Identificación del Problema

El problema, básicamente, era crear un modelo anatómicamente fiel a la columna vertebral y los miembros inferiores humanos, y hacer una animación de la biomecánica de los mismos, y presentarla de modo que pueda ser visualizada en la sala Ixtli.

➤ Análisis del problema

Realizar la búsqueda de herramientas de software, así como las técnicas de ingeniería más apropiadas para ofrecer una solución viable en base a la información proporcionada por la Facultad de Medicina, UNAM.

➤ Desarrollo del problema

A través de la investigación y asesoría médica especializada en esta etapa, se evaluarán las posibles soluciones planteadas, según las necesidades a cubrir.

Elegida la solución, será necesario describir las características y el funcionamiento de dicha solución.

➤ Solución del problema

Se pretende implementar la tecnología de cómputo, que permita la optimización de los recursos actuales de visualización y Realidad Virtual.

SOLUCIONES

Se requirió de un análisis de cada problema para poderle dar la solución más adecuada según los elementos con los que se contaban.

El primer paso para realizar el proyecto era conocer cómo eran adquiridas las imágenes médicas que combinan los datos de los huesos de un cuerpo con cualquier tipo de imágenes diagnósticas. Las imágenes médicas adquiridas se encontraban en formato DICOM, por lo que se necesitaba encontrar el programa más adecuado para poder separar la plancha de cada uno de los huesos.

Un análisis realizado en base a las características anatómicas del modelo, determinó que era necesaria una técnica de edición de imágenes que nos ayudara a segmentar mejor cada geometría para evitar la pérdida de información y no alterar la calidad del modelo.

Un programa el que se pudieron visualizar las imágenes por separado y a la vez generar el volumen fue AMIRA, pero este tenía el inconveniente de no proporcionar la información necesaria para poder detectar el rango de valores que constituían el segmento de la plancha. Entonces se decidió hacer pruebas con Adobe Photoshop, afortunadamente este programa si fue capaz de proporcionar un rango de valores, en este caso la escala de grises y coordenadas cartesianas en las que se encontraba la plancha, datos necesarios para poder eliminar la plancha.

Una vez obtenidos los datos anteriores, era necesario buscar el software que permitiera hacer el procesamiento digital de todas las imágenes para poder eliminar la plancha definitivamente. Un programa sumamente robusto que es usado en muchas aplicaciones de ingeniería es MATLAB (MATrix LABoratory).

Matlab contiene una caja de herramientas (toolbox) especializada en imágenes que permiten hacer operaciones con imágenes por ejemplo: filtrar ruido (permite el paso de ciertas frecuencias y después se realiza la decimación en la imagen ya filtrada), realce (destaca ciertos valores o regiones así como mejorar una imagen), segmentación (separar imágenes por zonas u objetos), entre otras, pero realmente la operación que nos permitirá finalmente separar la plancha es la convolución de imágenes.

Para la parte de cómo se debería realizar la animación surgió la idea de utilizar el programa de 3D Studio Max, pero al realizar las pruebas no ofrecía una buena calidad en las geometrías y para los médicos el nivel de detalle es algo de mucha importancia además de no ofrecer una método de animación acorde con el movimiento natural de una persona real.

Al realizar la prueba con Maya los doctores quedaron muy satisfechos con el nivel de detalle en los huesos porque no existía gran pérdida de información en comparación con 3d Studio y otra ventaja que nos ofreció el software fue el manejo de la técnica de animación esquelética, muy acorde a las necesidades del proyecto, por este motivo se decidió usar Maya.

Durante esta etapa de análisis se observó que los programas de visualización y de animación no compartían el mismo formato para manejar las imágenes que generaban el modelo, es decir, no se podían comunicar, por lo que se tuvo que apoyar en un programa llamado Deep Exploration que permitía cambiar de un formato a otro para que se pudiera llevar a cabo el proceso de animación y visualización en diferentes programas.

En cuanto a lo relacionado con la falta de conocimiento en el área médica, se contó con el apoyo del personal especializado que ayudaría a orientarnos en cómo deberían estar acomodados anatómicamente correctos los huesos de la columna y los pies, así como el rango de movimiento permitido en cada hueso, en conjunto e individualmente.

Un problema inicial que se encontró durante la realización de este trabajo es que las geometrías originales contenían una gran cantidad de polígonos así que era necesario reducir esa cantidad ya que la computadora no poseía los recursos suficientes (la cantidad de memoria RAM suficiente, procesador así como la tarjeta de video), para realizar el procesamiento de los datos para que al momento de proyectar las imágenes, la computadora fuera capaz de realizarlo sin mucha dificultad.

Los resultados esperados pueden plasmarse de diversas maneras, por ejemplo, la Facultad de Medicina UNAM lanzó una convocatoria en la cual se invitaba a los diversos proyectos de investigación que se llevan a cabo en esta facultad, a participar mediante la realización de un cartel informativo en el que se mostraría el avance y resultados de cada proyecto, así que se decidió realizar un cartel con estas características y se muestra a continuación:

Unidad Mixta
Centro de Enseñanza
y Adiestramiento Quirúrgico

BIOMECÁNICA DE LA COLUMNA VERTEBRAL Y EL MIEMBRO INFERIOR EN REALIDAD VIRTUAL

*LET HUMANOID HILION EN HENTHO HILION
INOLECUNCA DE TV COGUNA HILIONENT*

I. INTRODUCCIÓN

Actualmente los gráficos 3D ofrecen la posibilidad de crear ambientes virtuales en una computadora, lo cual unido a la técnica de visualización, permite que en el área científica se exploren y entiendan más fácilmente sistemas complejos.

Esto es posible gracias al avance en el desarrollo de algoritmos de visualización, lenguajes de programación orientados a objetos, y programas de modelado 3D que ofrecen la posibilidad de crear software de mejor calidad.

II. METODOLOGÍA

Se obtuvieron imágenes de la columna vertebral, la cintura pélvica y las extremidades inferiores por medio de tomografía axial computarizada (TAC), que es una técnica radiológica consistente en hacer pasar un haz de rayos X de manera sucesiva alrededor del objeto deseado.

A partir de los datos obtenidos de múltiples tomografías y utilizando el programa AMIRA que es capaz de realizar algoritmos complejos para la reconstrucción en 3D, se obtuvieron las geometrías de los huesos con una fidelidad muy similar a la realidad.

Para poder crear las imágenes animadas, se empleó el programa MAYA que permite el uso de técnicas de animación por computadora conocidas como Animación Esquelética, con el cual fue posible crear una cadena de uniones y huesos relacionados, cada hueso se transforma en una imagen tridimensional que incluye posición, escala y orientación, así como un hueso opcional conocido como "padre" y que nos proporciona una referencia. De esta manera, se jerarquizan los huesos y se obtiene el movimiento de una manera más apropiada a la realidad.

III. RESULTADOS

Se obtuvieron modelos 3D que pueden ser empleados en la enseñanza de la Anatomía humana, asignatura esencial en las carreras de Medicina y Biología.

La animación lograda en la columna vertebral, permite estudiar el movimiento de cada vértebra de manera independiente con lo que pueden analizarse los ángulos de movimiento que tiene cada uno de los componentes vertebrales y facilitar la comprensión de posibles alteraciones.

En el caso de los componentes del miembro inferior, el trabajo para dar la animación necesaria, fue completamente diferente dado que los huesos largos que la constituyen se mueven con una sola unidad.

IV. CONCLUSIONES

- Las técnicas de visualización utilizadas para el anterior programa de cráneo dieron origen a una metodología que facilita el trabajo en el resto de los sistemas a desarrollar, en este caso, la columna vertebral y el miembro inferior humanos.
- La animación esquelética resultó una técnica útil al permitir un mayor control de los movimientos óseos y una animación más natural y fiel a la realidad.
- Gracias al empleo de la realidad virtual inmersiva, es posible obtener simulaciones en tiempo real que permiten adquirir conocimientos y destrezas, en este caso médico quirúrgicas, a un mayor número de estudiantes en menor tiempo antes de su ensayo en material biológico.

V. BIBLIOGRAFÍA

- G. Lucet Lagriffoul and D. Espinoza Juárez. *Imm as Environment for Virtual Reality Aided and Disovering*. *Monografía Biotécnicas Vol. XXXIII* páginas 77-191 UNAM, México 2004.
- LM de la Cruz, J. García, Y. Godoy and E. Ramos. *Case study: Parallel Lagrangian Visualization Applied to Natural Convective Flows*

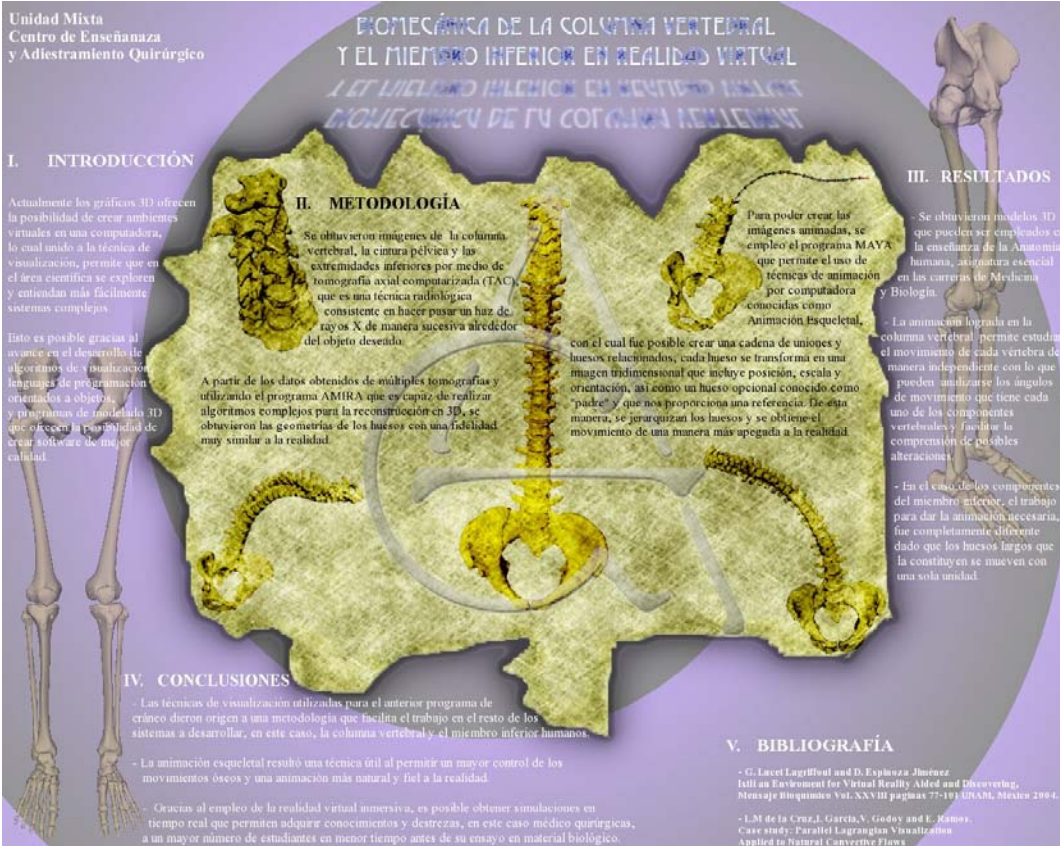


Figura 5.6: Cartel Informativo del proyecto Biomecánica de la columna vertebral y el miembro inferior.

CAPÍTULO 6

**BIOMECÁNICA DE LA COLUMNA
VERTEBRAL Y MIEMBRO INFERIOR
HUMANOS.: DESARROLLO**

Antecedentes Anatómicos

Nuestro esqueleto es una estructura sobre la cual los músculos, tendones, grasa y piel se encuentran unidos. Es por eso que nuestro cuerpo toma la forma del esqueleto y es lo que le da proporción. Protege y da soporte a lo que se encuentra dentro (órganos vitales) mientras sirve como un soporte para las partes externas tales como los músculos, la grasa y la piel. La estructura interna de nuestro cuerpo también afecta al contorno de nuestra figura.

La columna vertebral es una compleja estructura osteofibrocartilaginosa y articulada, en la parte dorsal del torso, que funciona principalmente como elemento de sostén, recubrimiento y protección de la médula espinal, y es uno de los factores que ayudan a mantener el centro de gravedad de algunos animales: los vertebrados. Consta de 33 piezas óseas que constituyen la columna vertebral, denominadas vértebras, las cuales están conformadas de tal manera que goza la columna de flexibilidad, estabilidad y amortiguación de impactos durante la moción normal del organismo. [M19]

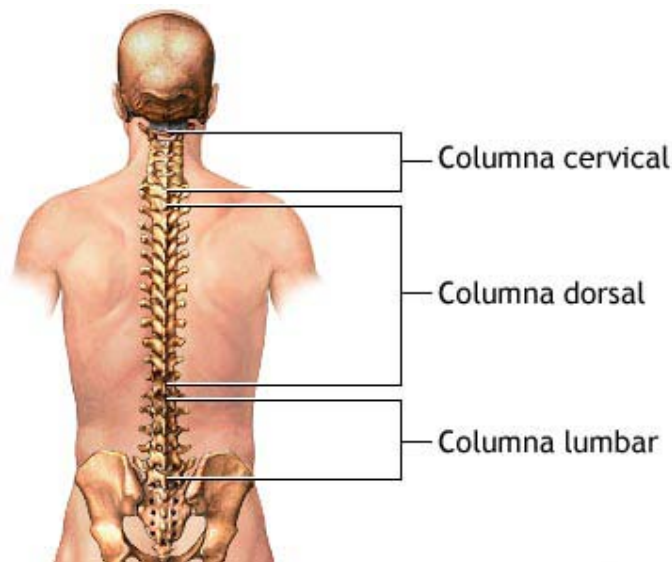


Figura 6.1: Columna Vertebral Humana, vista dorsal. [M14]

El miembro inferior del cuerpo humano se encuentra formado por diversos huesos: en la parte superior de la pierna hay un hueso llamado fémur seguido de la rotula, hay dos más abajo llamados tibia y fibula. Los huesos de la pierna son más pesados y por lo tanto tienen menos grados de libertad de movimiento esto debido al hecho de que los huesos de la pierna se han desarrollado para la movilidad y carga de peso.

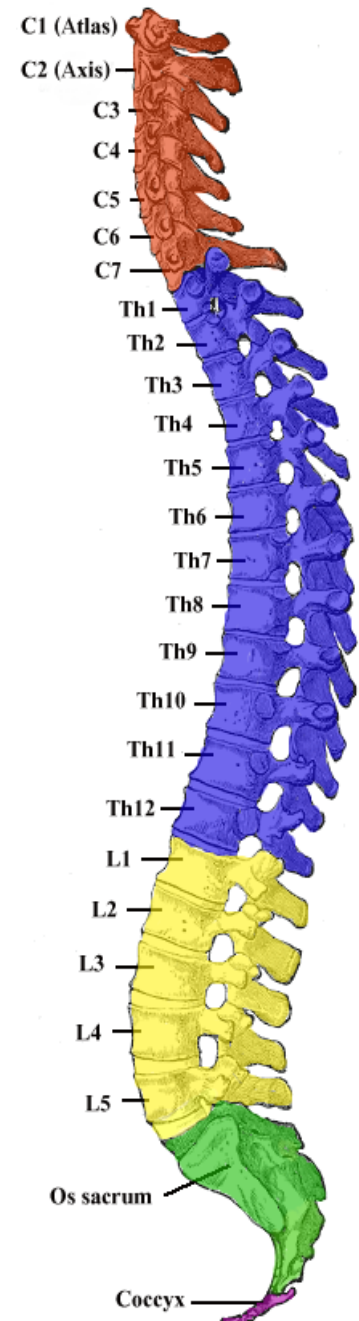


Figura 6.2: Columna Vertebral Humana, vista lateral. [M15]

Para que los huesos de la pierna se unan es necesario trabajar con la pelvis la cual tiene huecos en los que se unen perfectamente con el fémur para formar la articulación que restringe el movimiento en cada dirección. La tibia permite soportar el peso del fémur. La fibula es ligeramente y que no carga directamente el peso del cuerpo pero sirve como una unión para los músculos.

En el pie se encuentran huesos de menor tamaño, entre éstos encontramos: calcáneo, astrágalo (hueso del tarso que se articula con la tibia y la fibula para formar la articulación del tobillo. Es el único hueso del tarso que tiene contacto con los huesos de la pierna), escafoides, cuboides; para simplificar la nomenclatura, algunos huesos están divididos en grupos: cuñas, metatarsos y falanges (dedos).

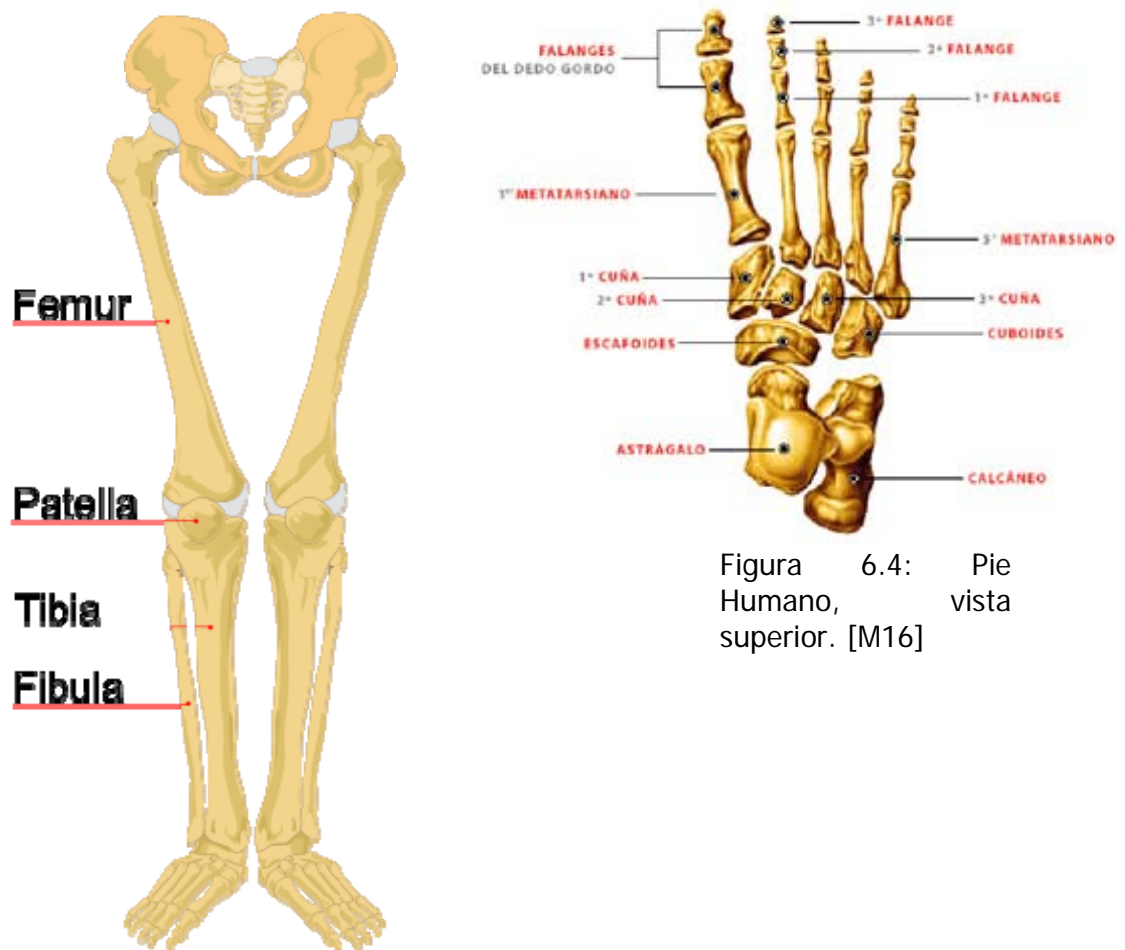


Figura 6.4: Pie Humano, vista superior. [M16]

Figura 6.3: Miembros Inferiores Humanos, vista frontal. [M38]

6.1 Desarrollo Modelado

6.1.1 Adquisición de imágenes

Recibimos las imágenes tomográficas con las siguientes especificaciones:

- Resolución de 512 x 512.
- 0.6 mm de distancia entre cada imagen.
- Escala de grises a 16 bits de profundidad.
- Formato DICOM¹.



Las imágenes que recibimos originalmente contenían un elemento no relevante para nuestra reconstrucción, el soporte del dispositivo donde fueron puestos los huesos para realizar las tomografías.

Figura 6.5: Tomografía en formato DICOM de los huesos de la cadera.

Para poder tener las imágenes tomográficas de los huesos solamente, fue necesario aplicar técnicas de procesamiento digital de imágenes, las cuales involucran álgebra matricial.

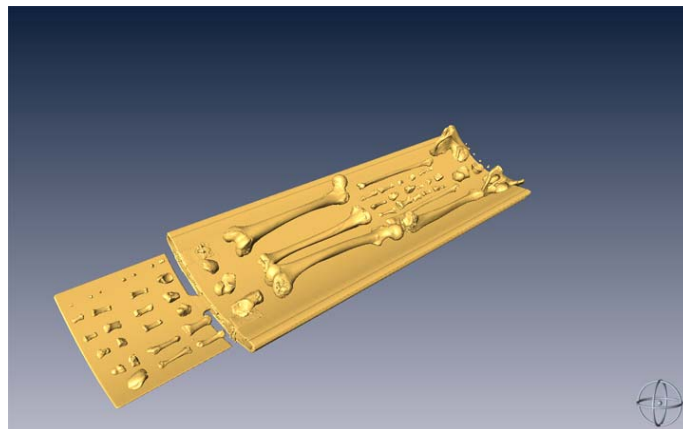


Figura 6.6 Representación 3D en Amira mostrando el elemento sobrante de la cama.

¹ Digital Imaging and Communication in Medicine

6.1.2 Procesamiento de imágenes con Matlab

El procedimiento que seguimos para deshacernos del elemento no relevante fue crear una matriz tipo máscara con las mismas dimensiones de las imágenes tomográficas, pero con únicamente conteniendo los elementos 1 y 0. Utilizando el software Photoshop, y basándonos en las siguientes imágenes, creamos las máscaras que se desharían de las características de los dos tipos de plancha que interfiere (la cabecera, y el resto de la cama).



Figura 6.7: Ejemplo de Tomografía, con la cama mostrándose.



Figura 6.8: Filtro para la cabecera.



Figura 6.9: Filtro para la cama.

Posteriormente, se realizó el proceso de la eliminación de los elementos no deseados mediante procesamiento digital de imágenes utilizando MATLAB. Debido a la gran cantidad de imágenes que debíamos procesar, se realizó un script que automatizaría el procedimiento para todas ellas.


```
clc;
clear;

A=dir('*.*');
filenames={A.name}';
frames=numel(filenames);

a=imread('mCama.tiff');

K=0;

for K=3:811 %La cabecera de la cama empieza en la
imagen 191-: 1007-191+2

    I=dicomread(filenames{K});
    info=dicominfo(filenames{K});

    procesamiento=a.*I;

    dicomwrite(procesamiento,filenames{K},info);
    [x map]=dicomread(filenames{K});
    figure;
    imshow(x,map)
    title(filenames{K});
    pause(0.01)
    close all;

end
```

A grandes rasgos, el script realiza una convolución² de ambas matrices, la máscara y la imagen que se desea procesar, y guarda las imágenes procesadas en formato DICOM.

² Operador [.*] en MATLAB.

6.1.3 Generación de Modelos 3D con Amira

Una vez completado el procesamiento de las imágenes, se cargaron en el software de visualización Amira, el cual las utilizó para realizar un modelo tridimensional.

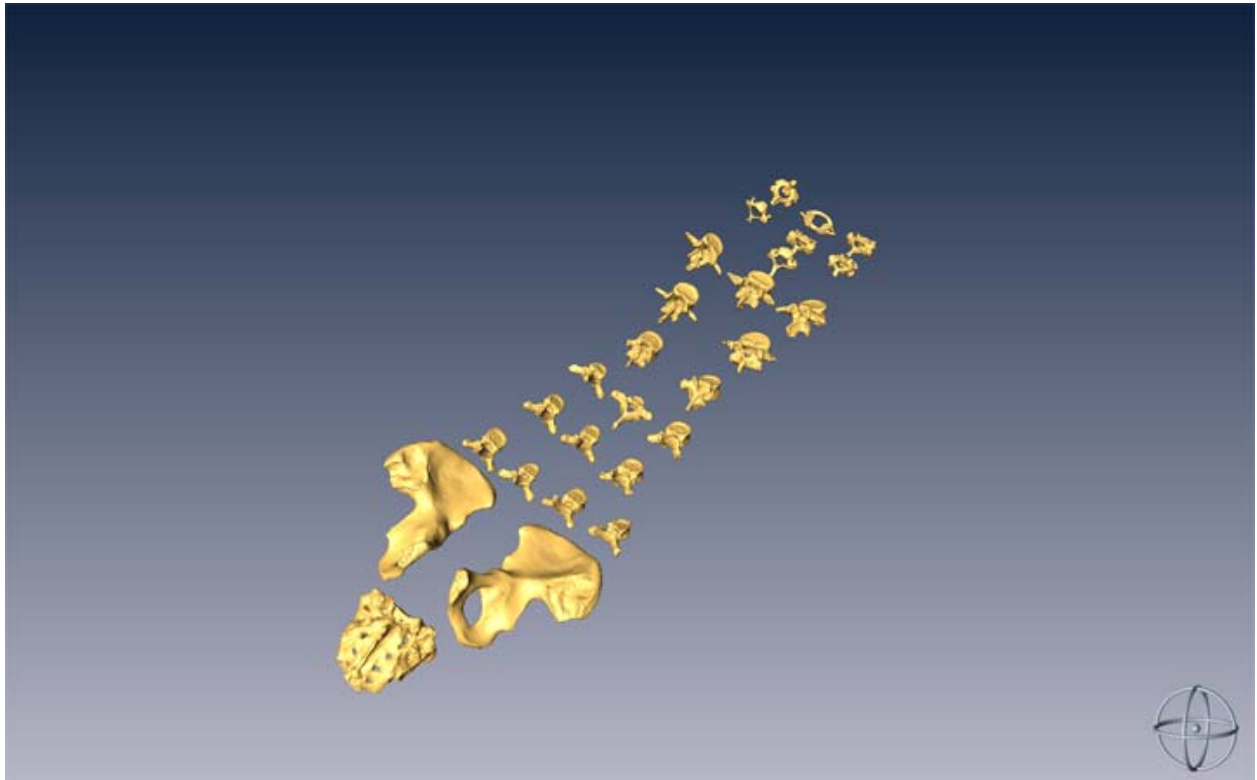


Figura 6.10: Representación 3D en Amira de las tomografías procesadas.

6.1.4 Segmentación De huesos

Como los huesos en este modelo en conjunto eran solamente un modelo, fue necesario realizar una segmentación, es decir, separar hueso por hueso, para poder ser manejados individualmente, paso necesario para poder ensamblarlos en una representación anatómicamente exacta de la columna vertebral y los miembros inferiores. Este proceso fue de lo más tardado, como cada hueso debía ser segmentado individualmente, con diferentes parámetros, para mantener una calidad y exactitud anatómica deseable en todos ellos.

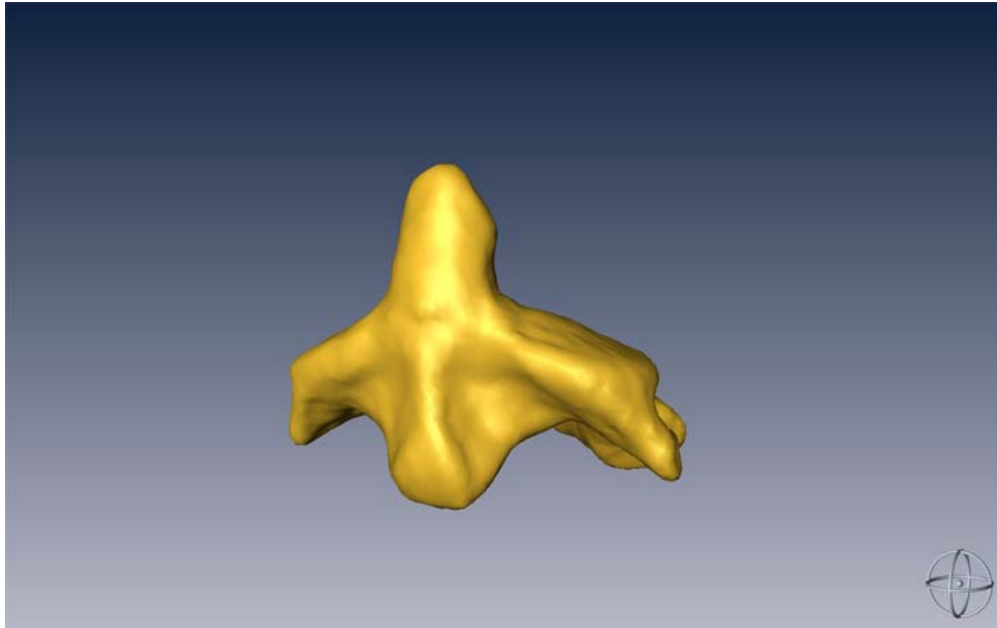


Figura 6.11: Modelo tridimensional de la vértebra Axis (C2) segmentada.

6.1.5 Exportación de formato Amira - Maya

Ya separados, se hizo una pequeña optimización de las geometrías que conformaban cada hueso, pues en conjunto, el número de triángulos que conforman el sistema es muy grande, y, aunque el equipo que utilizamos es bastante poderoso, la pura cantidad de datos a manejar era demasiada.

Originalmente, se deseaba realizar la biomecánica en Amira, sin embargo, las capacidades de animación que ofrece son bastante limitadas, por lo que decidimos utilizar un software más especializado en animación. Para esto, exportamos desde Amira nuestras geometrías en formato OBJ, formato bastante portable dentro del software 3D.

Primero intentamos en 3D Studio MAX de Autodesk, pero resultaba poco eficiente, pues las geometrías que nos mostraba al importar nuestros archivos OBJ se veían bastante burdas, y era necesario hacer algunos ajustes que aumentaban considerablemente el número de triángulos que conformaban los modelos, lo cual inutilizaba nuestros esfuerzos anteriores en optimizar geometrías. Tras buscar mejores alternativas, decidimos utilizar Maya de Autodesk, pues de entrada, al importar nuestras geometrías, mostraban la calidad vista en Amira, además de que es uno de los programas más poderosos y utilizados en el campo de la animación por computadora.

6.1.6 Reconstrucción Anatómica

Ya una vez teniendo los huesos desarticulados en Maya, bajo orientación médica especializada, reconstruimos la columna vertebral, nombrando cada hueso, y ordenándolos de acuerdo a su posición en un sistema óseo real.

Para asegurar que nuestra animación se apegaba lo más posible a la realidad, contamos con la asesoría de un médico ortopedista, que nos ayudó a restringir los grados de libertad con los que contarían las vértebras. Esta restricción se hizo solamente en los nodos padres de cada grupo de vértebras.

De igual manera, el desarrollo de la reconstrucción anatómica del miembro inferior requirió conocimiento anatómico, previamente proporcionado por médicos ortopedistas de la Facultad de Medicina, ya que sin su asesoría no se hubiera logrado ensamblar cada uno de los huesos correctamente.

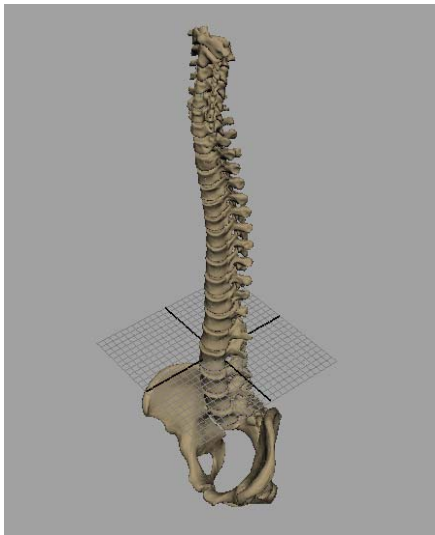


Figura 6.12: Modelo de la Columna Vertebral ensamblada en Maya.

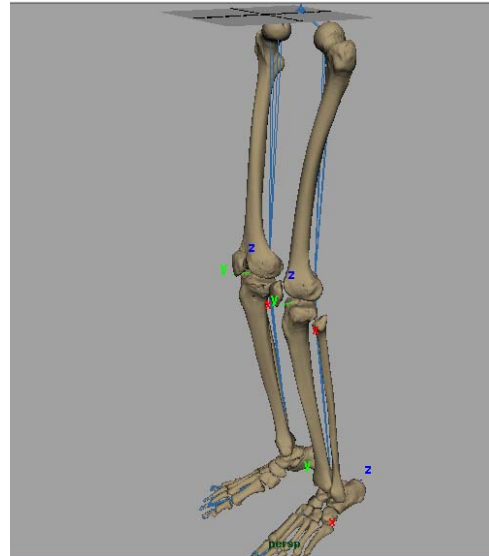


Figura 6.13: Modelo de los Miembros inferiores ensamblados en Maya.

6.1.7 Animación Esqueletal en Maya

Debido a los objetivos del proyecto, de mostrar la biomecánica de los sistemas óseos reconstruidos, era necesario decidir cómo se iba a hacer la animación. La decisión fue simple, considerando que la animación esquelética está diseñada para imitar el movimiento de un cuerpo articulado, como lo es un esqueleto, decidimos utilizarla. El hacer uso de esqueletos en animaciones 3D permite estirar o realizar deformaciones de las geometrías unidas a los huesos que conforman al esqueleto, esto le da un mayor realismo a la animación.

6.2 Desarrollo Animación

Para la realización de la animación en Maya, contamos con 2 opciones, cinemática inversa (IK-Handles) o la utilización de un script en el lenguaje propietario de Maya (MEL). Se probaron ambos métodos, sin embargo el de cinemática inversa resultó inadecuado para la columna vertebral, así que terminamos utilizando un script para el movimiento de la columna vertebral, y cinemática inversa para los miembros inferiores, que resultaba mucho más adecuado para ello.

6.2.1 Joints (nodo, articulación o unión)

El primer paso para realizar una animación esquelética es lo que se conoce como rigging, es decir, crear un esqueleto hecho por *joints* (uniones o nodos), y emparentar las geometrías a dichos joints. Esto constituye un modelado jerárquico, y debimos decidir cual joint debe ser la raíz. En el caso de la columna vertebral, este fue L5, y todas las demás heredarán las transformaciones geométricas que sufra este joint. Se fue emparentando cada vértebra con un joint, moviéndonos desde las vértebras inferiores (lumbares), pasando por las medias (torácicas), hasta las superiores (cervicales), y así, se aseguraba que las transformaciones de la anterior afectarían a la vértebra siguiente, aparte de sus propias transformaciones.

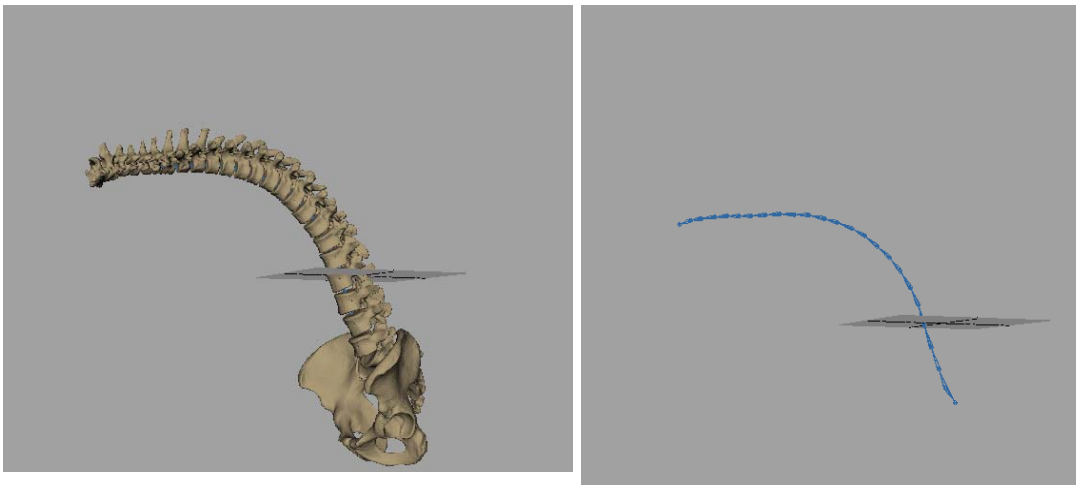


Figura 6.14: El modelo, izquierda, afectado por las transformaciones hechas al esqueleto, derecha.

En el caso de los esqueletos existe una parte importante que es el manejo de jerarquías ya que los huesos son creados por medio de uniones las cuales estos parten de un nodo raíz (root) que es el padre de todos los demás nodos creados en la jerarquía de cada hueso.

La herramienta en maya (joint tool) es usada para crear y colocar nodos para crear una jerarquía esquelética. Para colocar los nodos es recomendado usar una vista ortográfica, es decir, con la cámara posicionada desde arriba, de frente o de lado para que de esta manera permita una mejor colocación de cada nodo.

Al colocar un nodo adicional automáticamente se convierte en parte de la actual cadena de nodos e hijo del anterior nodo previamente colocado. Estos nodos son los puntos de movimiento en las articulaciones de un esqueleto.

Si tomamos como ejemplo el caso del cuerpo humano podemos ver que existen diversos nodos, a partir de éstos pueden derivarse más nodos para conformar todos los huesos que son parte de un esqueleto.

Un nodo en la cintura puede ser el nodo raíz a partir del cual se pueden derivar los nodos de una pierna del esqueleto el cual a su vez puede producir más nodos: en la rodilla, este nodo a su vez será padre de los que pertenecen a los nodos que se encuentran en el talón y este último a los nodos de los cinco dedos del pie.

Si volvemos a partir del mismo nodo ubicado en la cintura, esta vez se puede crear otra jerarquía para la columna vertebral se puede tomar el mismo nodo que está en la cadera pero esta vez se manejaría un solo padre y los demás nodos se colocan en cada una de las vértebras hasta llegar al cráneo.

Una vez realizada la reconstrucción anatómica correctamente, se puede continuar con la elaboración del esqueleto con joints (rigging), para esto se manejan dos grupos similares de cadenas de joints, una para cada pierna, ambas cadenas tienen como nodo padre el joint ubicado en la cadera a partir del cual se derivarán los nodos hijos.

Al igual que con los nodos de la columna vertebral, los nodos de la pierna fueron creados en un orden jerárquico, en el que se heredarán transformaciones y movimientos del nodo padre (cadera), sucede lo mismo para cada uno de los nodos generados para el pie.

En el caso de las piernas se tiene una estructura de una cadena de nodos que inicia por la pelvis (1 nodo) → rodilla (1 nodo) → rotula (1 nodo) → (1 nodo) astrágalo (tobillo)

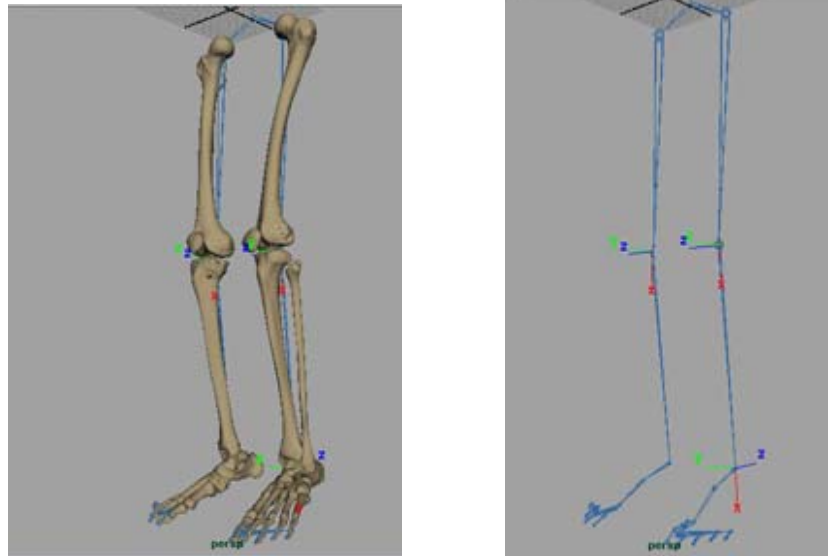


Figura 6.15: Rigging del modelo de los miembros inferiores.

A partir del tobillo se tomó otra estructura que permitiera tener un mayor número de nodos que fueran clave para crear un movimiento natural. La cadena de forma general fue creada de la siguiente manera: astrágalo (1 nodo) → escafoides (1 nodo, parte central) → cuñas (1 nodo, parte central) → metatarsos (5 nodos) → falanges (19 nodos), como se observa en la Figura 6.16.

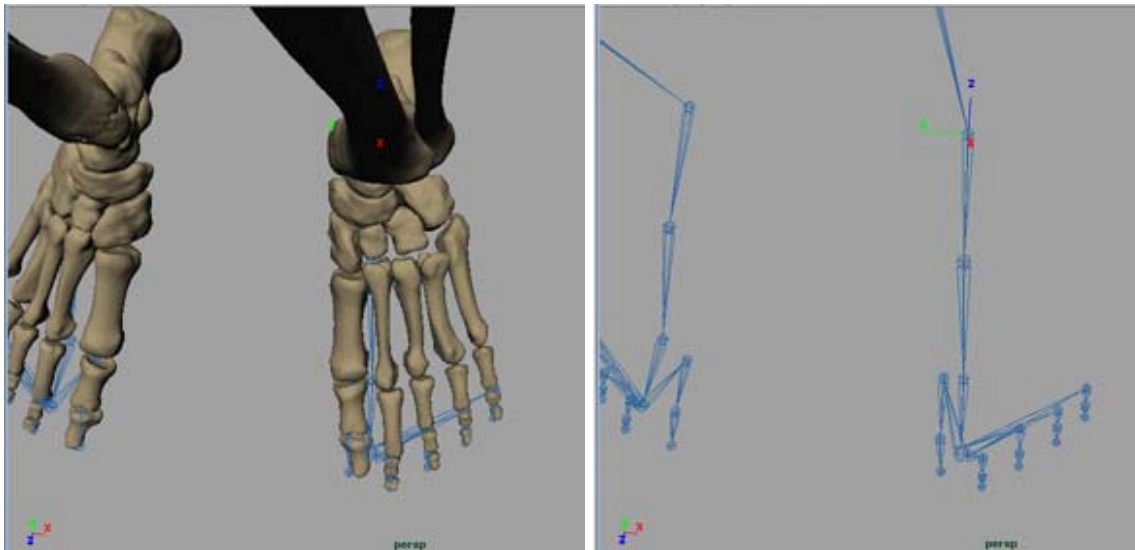


Figura 6.16: Rigging de los pies.

Una vez generada la jerarquía se prosiguió a emparentar la cadena de nodos con cada una de las geometrías correspondientes a cada nodo.

También se necesitó limitar los atributos (rotaciones) de los nodos principales para que no pudieran existir movimientos que no son físicamente posibles en un ser humano normal.

6.2.2 IK-Handle

El IK-Handle es una herramienta incluida en el paquete de Maya, se refiere al método de animación mediante cinemática inversa para indicar que una cadena de huesos hará uso de cinemática inversa para realizar una animación.

Para poder hacer uso de esta herramienta se tiene que crear un esqueleto, activar la opción para dar un clic izquierdo en un nodo que servirá como nodo base y será afectado por cinemática inversa (IK), después de este paso se tiene que dar otro clic en el nodo final, éste nodo servirá como el manipulador de la cadena (handle).

Como observamos anteriormente en las piernas se manejó una estructura de nodos: cadera → rodilla → tobillo → dedos del pie. Para hacer uso del IK-Handle primero se debe activar la herramienta luego seleccionar el nodo de la cadera y después el nodo del tobillo, de esta manera aparecerá de forma gráfica el IK-Handle.

Si el nodo o el IK-Handle en la escena son muy grandes o pequeños, se puede modificar el tamaño de cualquiera de los dos, esto facilita la visibilidad y manejo del esqueleto creado. Se puede modificar los atributos del IK-Handle al igual que con los nodos, como lo son rotación, traslación y escalamiento

El IK-Handle de la pierna se realizó desde el nodo padre que se encuentra en la articulación de la cadera con el fémur y hasta el nodo del tobillo, de esta manera se podrá tener control en el movimiento de la pierna completa para realizar movimientos de flexión y extensión, un ejemplo de este movimiento es el de caminar.

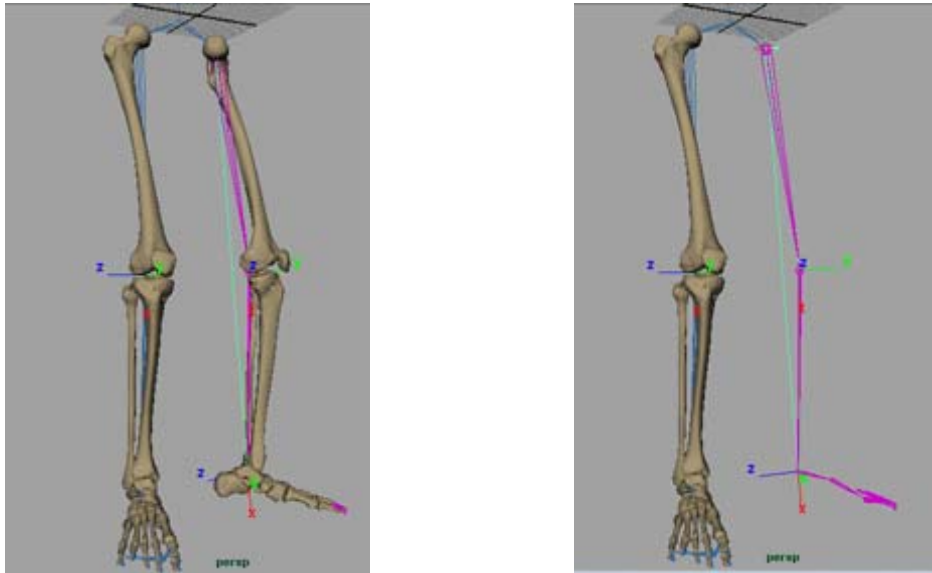


Figura 6.17: IK-Handle en el modelo de la pierna.

Los movimientos realizados por la pierna son heredados por el pie, pero el pie tiene sus propios movimientos que son independientes de los realizados por la pierna, es por esto que se decidió hacer uso de muchos IK-Handle para poder realizar la animación de los movimientos, por ejemplo de los dedos.

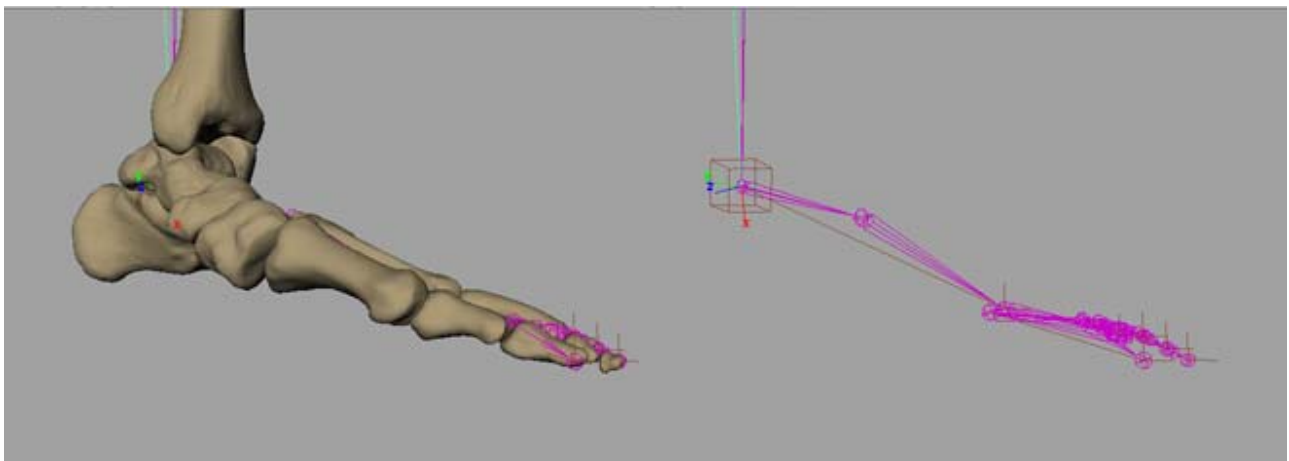


Figura 6.17: IK-Handles en el modelo del pie

6.2.3 Script MEL

MEL es el acrónimo de Maya Embedded Language. Es un lenguaje de programación personalizado, diseñado específicamente para trabajar dentro de Maya. Debido a su sintaxis y estructura simple, es fácil y más accesible que una interfaz de programación de C++. Una ventaja de MEL es que es un lenguaje interpretado, mientras que los lenguajes de programación típicos requieren que se compile y después se ligue el código fuente, un lenguaje interpretado puede ser ejecutado inmediatamente. No hay necesidad de usar compiladores o depuradores externos. [18]

Como MEL es un lenguaje interpretado puede correr un poco más lento que un código equivalente en lenguaje de alto nivel C++. Un programa en un lenguaje de alto nivel es compilado desde el código fuente para generar instrucciones el código máquina lo cual lo hace relativamente un poco más rápido que MEL al momento de ejecutarse.

Un script MEL permite crear gráficos personalizados para animar a un personaje y modificar atributos de diversas geometrías creadas en Maya. La interfaz que proporciona Maya es muy intuitiva de usar porque todos los controles están etiquetados y con instrucciones para su uso además de que es más rápido posicionar teclas con funciones de MEL. En cambio una instrucción en Maya primero debe ser interpretada y después convertida en una instrucción en código máquina.

```
jTRoot.rotateX=T12.rotateX;  
jT11.rotateX=T12.rotateX;  
jT10.rotateX=T12.rotateX;  
jT9.rotateX=T12.rotateX;  
jT8.rotateX=T12.rotateX;  
jT7.rotateX=T12.rotateX;  
jT6.rotateX=T12.rotateX;  
jT5.rotateX=T12.rotateX;  
jT4.rotateX=T12.rotateX;  
jT3.rotateX=T12.rotateX;  
jT2.rotateX=T12.rotateX;  
jT1.rotateX=T12.rotateX;
```

Script MEL de una de las vértebras³

Para la columna vertebral, se hizo una pequeña animación que mostrara los movimientos posibles, siempre bajo la supervisión de un médico, para asegurar la veracidad. En la siguiente figura se muestran algunos de los movimientos generados.

³ Este script se hizo de manera muy similar para cada sección de la columna vertebral, es decir, uno para la sección lumbar, otro para la sección torácica, y otro para las cervicales. Las variables rotX, rotY, y rotZ son atributos creados por nosotros limitados de acuerdo a la asesoría médica.

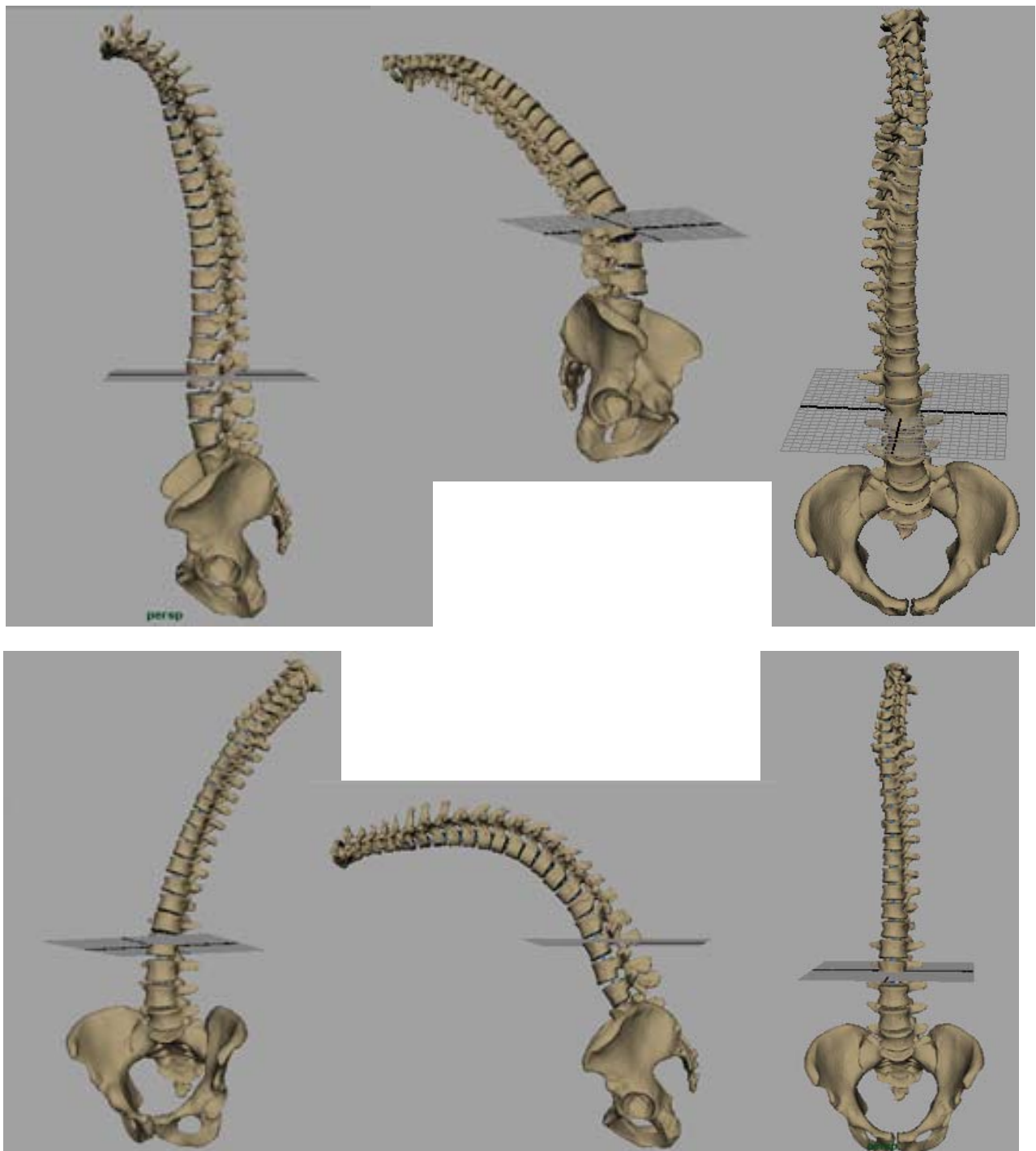


Figura 6.18 Algunos movimientos de la animación de la columna vertebral.

En este punto, aún cuando ya se habían hecho optimizaciones previas, con la adición de la animación, los datos de la reconstrucción crecieron bastante, entonces, optamos por tener dos versiones, una de alta resolución (conteo de polígonos) que fuera estática, y una de menor resolución que sería utilizada solamente en la animación. Afortunadamente, en la versión de baja resolución, es apenas apreciable la diferencia, y solamente si uno se acerca demasiado a las geometrías. A la distancia necesaria para apreciar la animación completa, no se nota la diferencia.



Figura 6.19: Imagen alta resolución estática.



Figura 6.20: Imagen baja resolución de la animación.

Utilizando las geometrías optimizadas, volvimos a hacer la animación de la columna vertebral, y se hizo una animación pequeña que mostrara algunos movimientos característicos de los miembros inferiores. Debido a la versatilidad de las articulaciones de los pies, resultaba muy difícil hacer una animación con todos los movimientos posibles, por lo que optamos por hacer una pequeña animación representativa. Algunos movimientos de la misma se muestran en la siguiente figura:



Figura 6.21: Animación de los miembros inferiores.

6.3 Pruebas

Las pruebas realizadas para el proyecto se realizaron primero en la computadora otorgada por la facultad de medicina

- Dell Precision 470 Workstation
- Intel Xeon Dual-Core x2 2.8 GHz
- 3 GB RAM
- nVIDIA Quadro FX 1300 128 MB VRAM

Con esta computadora se realizaron prácticamente todas las pruebas:

- Visualización de Geometrías (Amira y Maya)
- Eliminación de soporte (MATLAB)
- Compatibilidad de formatos entre Maya y Amira (Deep Exploration)
- Animación (3D Studio vs. Maya)
- Optimización (reducción de polígonos con Rational Reducer)
- Estereoscopía (Uso de Amira)

La prueba de mayor importancia para que el proyecto pudiera considerarse exitoso fue la realizada en la sala Ixtli.

Debido a que inicialmente se tenía un modelo con una cantidad de polígonos muy elevado y la supercomputadora no era capaz de procesar tanta información gráfica, así que la primera prueba resultó ser un fracaso.

Una vez realizada la optimización de las geometrías se volvió a realizar la prueba en la supercomputadora de la sala Ixtli y debido a la reducción de polígonos, afortunadamente esta vez sí se logró procesar la información, por lo que la prueba fue exitosa.

6.4 Aportación a la Docencia

Finalmente, el propósito de este proyecto es ser utilizado en clases de anatomía en la sala Ixtli o el laboratorio de Realidad Virtual propio de la Facultad de Medicina, así que tener la animación en formato de Maya era inútil, pues Maya no soporta estereoscopía. Era necesario volver a Amira, por lo que se exportaron ambas animaciones en formato VRML, que podía ser leído por Amira.

Una vez en Amira, se hizo un pequeño demo (Figura 6.22 y 6.23), para ser presentado en la sala Ixtli, en este demo se muestran los modelos, tanto de alta resolución (estáticos) como los animados de más baja resolución.

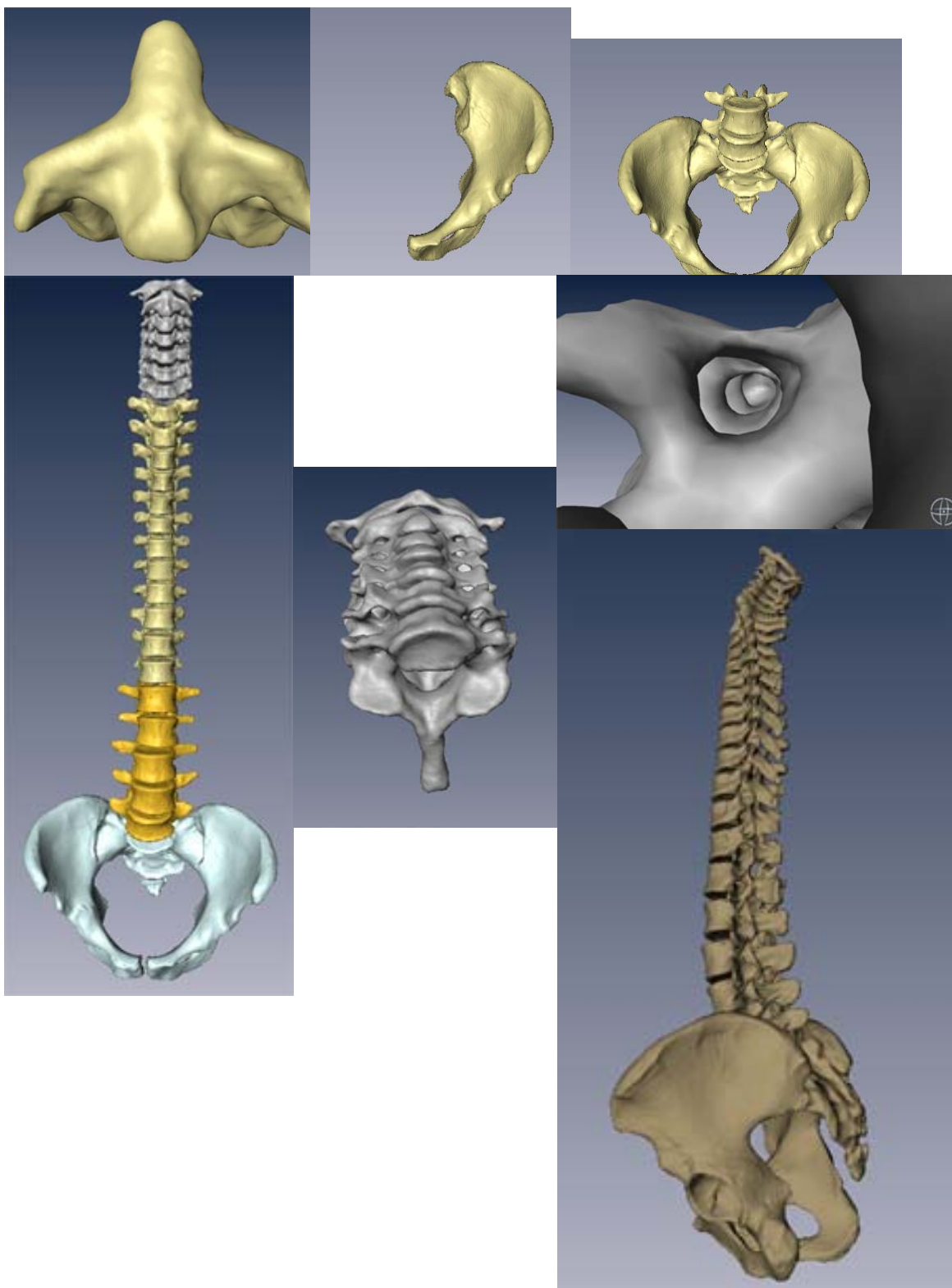


Figura 6.22: Demo presentado en la Sala Ixtli parte I.



Figura 6.23 Demo presentado en la Sala Ixtli parte II.

6.5 Resultados

Se pretende que en un futuro inmediato la herramienta desarrollada sea utilizada en el acto docente y permita a los profesores una técnica alternativa de la enseñanza de la anatomía humana que permita a los estudiantes comprender de una forma más sencilla los conceptos mostrados por el profesor para un mejor rendimiento académico.

Se obtuvieron modelos 3D que pueden ser empleados en la enseñanza de la anatomía humana, asignatura esencial en la carrera de Medicina así como en carreras relacionadas con el área de Ciencias biológicas y de la salud.

La animación realizada en la columna vertebral permite estudiar el movimiento de cada vértebra de manera independiente con lo que pueden analizarse los ángulos de movimiento que tiene cada uno de los componentes vertebrales y de esta manera facilitar la comprensión de posibles alteraciones.

En el caso de los componentes del miembro inferior, el trabajo para dar la animación necesaria, fue completamente diferente al planteado en la columna vertebral dado que los huesos largos que la constituyen se mueven como una sola unidad.

CONCLUSIONES

A lo largo del desarrollo de este trabajo de tesis, pudimos aplicar muchos de los conocimientos adquiridos durante nuestra carrera, y principalmente en nuestro módulo de especialización, Sistemas Inteligentes y Cómputo Gráfico.

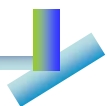
El proyecto realizado de la Biomecánica de la Columna Vertebral y Miembro Inferior consiste en una simulación, que hace uso de realidad virtual y gráficos 3D. Funciona como una herramienta auxiliar en la docencia para poder estudiar cada uno de los huesos así como el movimiento de los mismos, con una calidad de imagen que fomenta la cercanía con la realidad; con estos puntos anteriores podemos decir que los objetivos iniciales del proyecto han sido cumplidos de forma satisfactoria.

Las tomografías son el método de adquisición más adecuado para este proyecto, principalmente porque, al estar trabajando nosotros con tejido no vivo, pudimos aspirar a la máxima calidad de imagen, aplicando una cantidad de radiación que podría ser peligrosa en tejido vivo. Al contar con imágenes tomográficas de alta calidad, aseguramos que la reconstrucción tridimensional de los huesos era lo más fidedigno a la realidad.

Durante la realización del proyecto, nos vimos en la disyuntiva de las posibles soluciones de software, ya que no todas cubrían las necesidades particulares del problema a resolver. Para el procesamiento de las imágenes, como mencionamos en capítulos anteriores, se utilizó MATLAB, por la robustez que ofrece, y que es una herramienta que conocíamos con anterioridad. Para la visualización, se consideró OpenDX, pero Amira resultaba más adecuado, ya que es más amigable y permitía una mayor libertad para la realización de Demos Interactivos. Para el modelado tridimensional y animación se consideró 3DStudio MAX, pero decidimos utilizar Maya, por tener mejores características pertinentes a nuestro proyecto. Utilizar este software nos permitió enfocarnos al problema de crear la simulación, sin tener que ocupar un tiempo considerable en crear una herramienta para ello.

Con el software de visualización científica actual, mediante los algoritmos de reconstrucción 3D, como Volume Rendering, directo o indirecto, se puede tener un compromiso entre calidad y rendimiento. Desde el principio se buscaba la máxima exactitud anatómica de nuestras representaciones tridimensionales sin tener a un lado la eficiencia en recursos computacionales. Debido a esto, consideramos que la optimización realizada en los modelos fue apropiada, pues, el demo final puede ser ejecutado tanto en la supercomputadora del Laboratorio Ixtli, como en una computadora personal.

Las técnicas de visualización utilizadas para el anterior programa del cráneo dieron origen a una metodología que simplificó de gran manera el trabajo a lo largo de la elaboración del presente proyecto.



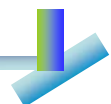
La animación esquelética resultó ser una técnica esencial para el manejo de nuestras geometrías al permitir un mayor control de los movimientos óseos y un movimiento más natural y fiel a la realidad. Esto, aunado a la cinemática inversa, nos permitió representar todos los movimientos que pueden realizar nuestros objetos de interés, es decir la columna vertebral y los miembros inferiores.

Gracias al empleo de la realidad virtual inmersiva, es posible obtener simulaciones en tiempo real que permiten adquirir conocimientos y destrezas, en este caso médico quirúrgicas, a un mayor número de estudiantes en un menor tiempo antes de su ensayo en material biológico, es decir, permite el estudio de los huesos de forma intensiva, en caso de encontrar alguna enfermedad o deformación, es posible determinar las causas que las producen, ya que con esta herramienta se tiene un entorno controlado, donde es posible reproducir las mismas condiciones sin poner en riesgo a personas reales (pacientes).

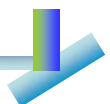
El proyecto antecesor a este, del Cráneo Desarticulado, desde su implementación en las clases, ha tenido resultados muy positivos en el proceso de aprendizaje en los alumnos de la Facultad de Medicina de la UNAM, lo cual se refleja en su desempeño académico, comparados con aquellos alumnos que tomaron sus clases sin el apoyo de dicha herramienta. Se espera que los modelos y la simulación realizada de la biomecánica tengan al menos los mismos resultados positivos. A la fecha aún no se utiliza en clases regulares, sin embargo, la recepción ha sido satisfactoria en varias conferencias y congresos presentados por los doctores a cargo del proyecto, y se espera que en este nuevo ciclo escolar ya sea utilizado regularmente por algunos profesores en sus clases.

En el estado actual del proyecto, la animación en el demo de Amira se encuentra en un ciclo de repetición constante. Es posible realizar una animación donde el presentador tenga control sobre ella, por ejemplo, detener el movimiento, y mostrar una sola vértebra. Sin embargo, esto requiere de una gran cantidad de espacio de almacenamiento y de RAM. Por ejemplo, para una sola vértebra animada de este modo, el almacenamiento requerido para una cantidad pequeña de cuadros era en el orden de Gigabytes, y reproducir la animación acababa con todos los recursos disponibles (Procesadores y RAM). Esto lo vuelve un poco inconveniente, aunque tal vez, con los avances tecnológicos, sea mucho más factible realizar una de estas animaciones en el futuro.

Se prevé que este proyecto llegue a abarcar el cuerpo humano completo (huesos, arterias, órganos, etc.), iniciando por el esqueleto. Por lo tanto una vez terminado este proyecto se pueda hacer uso del proyecto anterior y colocar un cráneo, con lo cual el esqueleto empieza a tomar su forma completa para un estudio más particular.



Se debe tomar en cuenta que una vez añadidos los nuevos modelos de los huesos restantes, aún es posible realizar mejoras, en el desempeño final del modelo 3D, en el entorno visual y en la información que se le pueda proporcionar al usuario, alumnos en este caso, además de tomar en cuenta los recursos de hardware para que pudiera ser utilizado en máquinas con menor cantidad de memoria y menor velocidad de procesador.



GLOSARIO

Absorción. Proceso de transferencia de la energía de las ondas electromagnéticas o sonoras a un medio, cuando lo atraviesan o inciden sobre él. En el caso de una escena 3D los distintos objetos que se encuentran en ella tendrán la facultad que tienen algunos sólidos para captar o recoger dentro de sí parte de la energía emitida por la fuente de luz. [M17]

Aliasing. Es el efecto que causa que señales continuas distintas se tornen indistinguibles cuando se les muestra digitalmente. Cuando esto sucede, la señal original no puede ser reconstruida de forma unívoca a partir de la señal digital, es por esto que en una imagen o escena 3D se pueden llegar a observar los artefactos de construcción como pixels, voxels o polígonos. [M18]

Alpha Blending. Es el proceso de combinar un color translúcido de primer plano con el color de fondo produciendo un nuevo color. [M19]

Anisotrópico. Este término se refiere a la propiedad de ser dependiente de la dirección, es decir, que no se tiene la misma forma, así que es posible especificar una forma de un objeto en diferentes direcciones. [M20]

Compactibilidad. Reducir el tamaño de un volumen o una geometría en cuanto a su cantidad de polígonos o tamaño de archivo para ocupe menos espacio al almacenarlo.

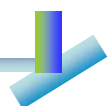
Complejidad. Cualidad de lo que está compuesto de varios elementos. Difícil de comprender o resolver. En computación se refiere al costo de los algoritmos en base a diferentes parámetros requeridos durante el cómputo de un algoritmo para resolver un problema. Los recursos comúnmente estudiados son el tiempo y el espacio. [M21]

Cuadro (Fotograma). Se le llama así a cada una de las imágenes individuales captadas por cámaras de video y registradas analógica o digitalmente.

Cuantos. Cantidad mínima determinada de energía que puede ser absorbida, propagada o emitida por la materia. [M22]

Difracción. Fenómeno físico por el cual un rayo de luz se desvía al interferir con otro o al pasar por el borde de un cuerpo opaco o por una abertura estrecha. También es un movimiento ondulatorio en el que una onda de cualquier tipo se extiende después de pasar junto al borde de un objeto sólido, en lugar de seguir avanzando en línea recta. [M23]

Dispersión. Descomposición de un haz de luz blanca, cuando incide sobre una superficie de vidrio, en una serie de haces coloreados. Fenómeno de separación de las ondas de distinta frecuencia al atravesar un material. [M24]



Eficiencia. Capacidad para lograr un fin empleando los mejores medios posibles. La óptima utilización de los recursos disponibles para la obtención de resultados deseados. [M25]

Emisión. Lanzamiento de ondas que transmiten sonidos e imágenes. Transmisión de una señal.

Emisor. Es un dispositivo electrónico que permite enviar mensajes a distancia a través de ondas. Que emite o envía hacia fuera.

Frame buffer. Es un espacio de memoria reservado para almacenar imágenes en un mapa o arreglo de bits que son enviadas al monitor en forma de píxeles. Regularmente el frame buffer es almacenado en la memoria RAM o en la tarjeta de video, si es que se cuenta con ella. Es el espacio de memoria que se utiliza como regulador y sistema de almacenamiento intermedio entre dispositivos de un sistema informático.

Histograma. Permite representar la distribución de intensidad de los píxeles de una imagen e indica la probabilidad de que un determinado nivel de gris aparezca en la imagen. Gráfico utilizado para representar la distribución de frecuencias de datos estadísticos, agrupados en intervalos, en forma de barras sobre un eje de coordenadas. [M26]

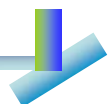
Imagen estereoscópica (o estéreo). Es aquella que, de una u otra forma, puede verse en 3D a pesar de estar sobre una superficie plana como en un papel o algún monitor. [M3]

Isosuperficie. Superficie definida por todos los puntos de un campo escalar que tienen un mismo valor.

Isotrópico. Homogeneidad en todas las direcciones, es decir, se conserva la misma forma de un objeto en todas sus direcciones. [M20]

Píxel. La parte más pequeña de la pantalla del monitor es un punto cuadrado o rectangular que recibe el nombre de píxel. La palabra píxel surge de la combinación de dos palabras inglesas comunes, picture (imagen) y element (elemento). Un píxel se describe de forma más correcta como una unidad lógica, y no física, ya que el tamaño físico de un píxel individual lo determina el fabricante del monitor. El tamaño de un píxel se mide en milímetros (mm). [M27]

Proceso estocástico. Es un tipo de proceso aleatorio que evoluciona con el tiempo y se representa por un conjunto de funciones aleatorias. Permite representar todos y cada uno de los pasos necesarios para realizar una actividad, además de las formas o maneras en que cada uno de los pasos puede ser llevado a efecto y sus respectivas probabilidades. [M28]



Raster. Es una malla rectangular de pixels que representan datos gráficos de un área determinada de la pantalla. [M29]

Receptor. Dispositivo electrónico que recibe señales eléctricas, telegráficas, telefónicas o radiofónicas y las convierte en sonidos o señales que se pueden oír o ver. [M30]

Reflexión. Es el cambio en la dirección o en el sentido de la propagación de una onda o rayo de luz que ocurre en la superficie de separación entre dos medios, de tal forma que regresa al medio inicial. [M31]

Refracción. Es el cambio de dirección que experimenta un rayo de luz cuando pasa de un medio material a otro o de un medio transparente a otro también transparente. Este cambio de dirección está originado por la distinta velocidad de la luz en cada medio. [M32]

Relación señal/ruido. Cualquier señal no deseada, en este caso señal eléctrica no deseada que circula por el interior de un equipo electrónico. También es una relación de señal deseada a indeseado de la señal (ruido) en el promedio de nivel de potencia de una transmisión. Si hay demasiado ruido en un circuito, la relación señal / ruido es baja. Si el circuito es de buena calidad, la relación señal / ruido será alto. [M33]

Render. Es una interpretación de todos los elementos de una escena para obtener una imagen final. Proceso de generación de una imagen a partir de información tal como modelos matemáticos de objetos tridimensionales, luces, mapas de texturas, etc.

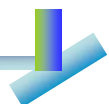
Resolución. Se refiere a la agudeza y claridad de una imagen. El término se utiliza normalmente para describir monitores, impresoras e imágenes. Para monitores, la resolución de pantalla significa el número de puntos (pixels) en toda la pantalla

Rigging. Es el proceso de establecer controles en un modelo para permitirle moverse y ser animado. En muchos casos se hace uso de un esqueleto, el cual contiene una jerarquía de nodos (joints) establecidos a lo largo del esqueleto.

Ruido. Se denota por ruido cualquier entidad en las imágenes, datos o resultados intermedios que no son interesantes para la computación que se pretende llevar a cabo. Señal extraña que impide o dificulta una comunicación.

Segmentación Proceso utilizado para identificar sólo partes específicas del total de datos.

Sensor. Dispositivo electrónico que detecta, o sensa manifestaciones de cualidades o fenómenos físicos, como la energía, velocidad, aceleración, tamaño, cantidad, etc. con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo. [M34]



APÉNDICE I

FORMATO DICOM

DICOM [M35]

El estándar DICOM (Digital Imaging and Communications in Medicine) fue creado por la National Electrical Manufacturers Association (NEMA) para ayudar en la distribución y visualización de imágenes médicas, como las TAC, resonancias magnéticas y ultrasonidos. La parte 10 del estándar describe el formato de archivo para la distribución de imágenes.

Un archivo DICOM contiene un header que posee información sobre el nombre del paciente, el tipo de escaneo, las dimensiones de la imagen, etc. Y aparte los datos de imagen. A diferencia de otros formatos, los archivos DICOM pueden ser comprimidos mediante variantes con o sin pérdida del formato JPEG, o con un formato Run-Length Encoding, idéntico a la compresión que se encuentra en algunas imágenes en formato TIFF.

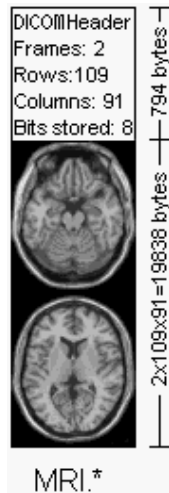
Estructura de los datos DICOM

En resumen, los datos DICOM contienen los siguientes componentes, llamados grupos:

- **Identificación.** Esta sección contiene información esencial para encontrar una serie de datos de imagen en una base de datos. Esto incluye la fecha y hora de la adquisición de imagen.
- **Parámetros de Adquisición.** Esta sección contiene información concerniente a la modalidad de la imagen (TAC, Resonancia Magnética, etc), los parámetros de escaneo específicos (nombre de secuencia, variantes de secuencia) y la administración de un agente de contraste. También, el nombre del fabricante y su producto se almacenan. Finalmente, la posición y orientación del paciente se mencionan aquí. Alguna información es específica a ciertas modalidades (y restringida a ellas), como la intensidad del campo magnético, eco, y tiempo de repetición para datos de Resonancia Magnética.
- **Datos del Paciente.** Contiene el nombre, identificador único, fecha de nacimiento, y sexo del paciente. La estatura y peso del paciente son datos opcionales que pueden ser útiles.
- **Datos de Imagen.** Para la localización y evaluación de imágenes, se requieren varios números, los cuales están contenidos en esta sección. Algunos ejemplos son el identificador de estudio, número de serie, grupo de imagen, y un número de adquisición.
- **Presentación de Imagen.** Esta sección contiene la distancia de corte y el espaciado de pixel, que describe la resolución en el plano. El rango de valores, máximo y mínimo, y el número de bits asignados también se encuentran en esta sección. Finalmente, se puede especificar una

configuración por defecto para el despliegue de datos (*WindowWidth*, *WindowCenter*).

El Header DICOM



Usando como ejemplo la imagen anterior, los primeros 794 bytes de la imagen contendrían el header DICOM, que describe las dimensiones de imagen y retiene otros datos de texto sobre el escaneo. El tamaño del header depende de cuanta información de header se almacene. En este caso, el header define una imagen de dimensiones 109x91x2 voxels, con una resolución de datos de 1 byte por voxel, lo cual da como resultado una imagen de tamaño 19838. Los datos de imagen siguen a la información del header.

La siguiente es una lista más detallada del header. DICOM requiere de un preámbulo de 128 bytes, usualmente con valor cero, seguido por las letras 'D', 'I', 'C', 'M'. A su vez seguido por la información del header, organizada en 'grupos'. Por ejemplo, el grupo 0002hex es el grupo de metainformación de

archivo. Los elementos DICOM requeridos dependen del tipo de imagen, y están listados en la parte 3 del estándar DICOM.

First 128 bytes: unused by DICOM format
 Followed by the characters 'D','I','C','M'
 This preamble is followed by extra information e.g.:

```
0002,0000,File Meta Elements Group Len: 132
0002,0001,File Meta Info Version: 256
0002,0010,Transfer Syntax UID: 1.2.840.10008.1.2.1.
0008,0000,Identifying Group Length: 152
0008,0060,Modality: MR
0008,0070,Manufacturer: MRlcro
0018,0000,Acquisition Group Length: 28
0018,0050,Slice Thickness: 2.00
0018,1020,Software Version: 46\64\37
0028,0000,Image Presentation Group Length: 148
0028,0002,Samples Per Pixel: 1
0028,0004,Photometric Interpretation: MONOCHROME2.
0028,0008,Number of Frames: 2
0028,0010,Rows: 109
0028,0011,Columns: 91
0028,0030,Pixel Spacing: 2.00\2.00
0028,0100,Bits Allocated: 8
0028,0101,Bits Stored: 8
0028,0102,High Bit: 7
0028,0103,Pixel Representation: 0
0028,1052,Rescale Intercept: 0.00
0028,1053,Rescale Slope: 0.00392157
7FE0,0000,Pixel Data Group Length: 19850
7FE0,0010,Pixel Data: 19838
```

Por ejemplo, la modalidad de la imagen a la izquierda es 'MR' (como dice el elemento del grupo 0008:0060) por lo que debe contener elementos que describan el tiempo de eco de la resonancia magnética. La ausencia de esta información en la imagen es una violación del estándar, aunque en la práctica, muchos visualizadores de DICOM ignoran la presencia de estos datos, y extraen solo la información del header que describe la imagen.

De particular importancia es el elemento de grupo 0002:0010. Esto define el **Identificador Único de Sintaxis de Transferencia** (siguiente tabla). Este valor reporta la estructura de los datos de imagen, revelando si los datos están

o no comprimidos. Las imágenes DICOM pueden ser comprimidas tanto por el esquema de compression con pérdidas JPEG común, donde se pierde alguna información de alta frecuencia como por el esquema sin pérdidas JPEG, rara vez visto fuera de la imagenología médica, de Huffman, diferente al más reciente y eficiente algoritmo JPEG-LS. Estos códigos se describen en la parte 5 del estándar DICOM.

Aparte de reportar el algoritmo de compresión, si existe, el UID de Sintaxis de Transferencia también reporta el orden de bytes para los datos raw. Computadoras diferentes almacenan valores enteros de maneras diferentes, las llamadas 'big endian' y 'little endian'. Considerando un entero de 16 bits de valor 257, el byte más significativo guarda el valor 01 (255) mientras que el menos significativo guarda el valor 02. Algunas computadoras lo almacenarían como 01:02, mientras que otras lo almacenarían como 01:02.

Aparte del UID de Sintaxis de Transferencia, la imagen también está especificada por las Muestras por Pixel (0028:0002), Interpretación Fotométrica (0028:0004), Bits Asignados (0028:0100). Para la mayoría de las imágenes TAC y de resonancia magnética, la interpretación fotométrica es un continuo monocromático (escala de grises). En DICOM, se le da a esta clase de imágenes una interpretación fotométrica de 'MONOCHROME1' (valores bajos=claros, valores altos=oscuros) o 'MONOCHROME2' (valores bajos=oscuros, valores altos=claros). Sin embargo, muchas imágenes de ultrasonido y fotografías médicas incluyen color, y éstas son descritas por diferentes interpretaciones fotométricas (Palette, RGB, CMYK, YBR, etc.).

APÉNDICE II

FORMATO VRML

VRML (Virtual Reality Modeling Language) [M36]

El lenguaje de modelado de realidad virtual (VRML) es un formato estándar de archivo para describir mundos y objetos 3D. Además las escenas con objetos estáticos, tienen un tiempo de definición integrado que permite la construcción de objetos animados. La animación inicia en un punto de tiempo determinado y prosigue hasta el fin de la animación.

A pesar de que la repetición de la animación es posible, para el espectador no es posible parar la animación o ir hacia atrás o hacia delante con velocidades diferentes. Pero esta característica es importante si los datos temporal-espaciales son examinados, como estaban los objetos en el pasado, como estarán en el futuro y más importante como se desenvuelve desde el pasado hacia el futuro en un rango de tiempo. Para este propósito una definición diferente del tiempo y objetos con representaciones dependientes son necesarios.

VRML está compuesto de nodos gráficos ordenados en una estructura jerárquica, estos son usados para definir más de un constructor gráfico, además de incorporar la noción de estado.

Los cambios en un estado afectan el modo en el que el constructor gráfico desplegará la imagen del archivo VRML. Por ejemplo, una traslación o una rotación de nodo afectan la ubicación del subsecuente constructor gráfico. Los nodos de materiales afectan a la apariencia del constructor gráfico definido en la escena

Cada nodo en un archivo VRML tiene una o más de las siguientes propiedades: [19]

- Tipo (Type): Muchos tipos de nodos están actualmente definidos, incluyendo los nodos que definen las propiedades y transformaciones de las primitivas gráficas, además de las cámaras y luces. Hay también nodos que permiten a otros nodos agruparse juntos.
- Campos (Fields): Los campos son los parámetros usados para personalizar características específicas de los nodos. Un nodo puede ser único con el uso de campos, por ejemplo, un cono puede tener un radio único para su base y un único valor de su altura.
- Nombre (Name): Nombrar un nodo es opcional. Un nodo que tiene nombre permite a su definición ser re-usada a lo largo del archivo VRML. El comando DEF es usado para definir un nodo con su nombre. Esta definición puede ser re-usada con el comando USE.
- Hijos (Children). Una clase especial de nodo permite tener hijos (subnodos). De esta manera se puede crear una jerarquía.

Sintaxis del formato de archivo VRML. [19]

La sintaxis para un archivo es simple. Es definido usando la gramática corta BNF¹:

```
// <name> Es un no terminal.
// ::= Es una asignación.
// ...|... Significa este constructor O ese constructor.
// [...] Significa que el constructor corchete es opcional.
// {...} Significa que el constructor llave es repetido una o más veces.
// '...' Es una definición literal de cadena.

<Start> ::= <VRML_ID> [Node]
;
<VRML_ID> ::= '#VRML V1.0 ascii'
| '#VRML V1.1 utf8'
;
<Node> ::= [ 'DEF' <ObjectName> ]
<ObjectType> '{'
[ <FieldDef> ]
[ <FieldUsage> ]
[ <SubNodes> ]
}'
| 'USE' <ObjectName>
;
<ObjectType> ::= 'AsciiText' | .... | 'WWWInline'
;
<FieldDef> ::= 'fields' '[' {<Type> <FieldName>} ']'
;
<Type> ::= 'SFLong' | 'SFFloat' | ..... 'MFFloat'
;
<FieldUsage> ::= { <FieldName> <Value> }
;
<SubNodes> ::= {Node}
;
<FieldName> ::= ('a'...'z' | 'A'...'Z')
[ {'a'...'z' | 'A'...'Z' | '0'...'9' | '_' | '+'} ]
;
```

¹ Backus-Naur form. Es una meta sintaxis usada para expresar gramáticas libres de contexto.

Un ejemplo de un archivo VRML que define un cono verde y una esfera roja iluminada por un punto de luz definido en la ubicación de la cámara:

```
# VRML V1.0 ascii

Separator {
  # Define cámara mirando a <0, 0, 0>
  PerspectiveCamera {
    Position 0 0 15
  }
  # Define la luz en la posición de la cámara
  PointLight {
    Location 0 0 15
  }
  # Define una esfera roja centrada en (3, 0, 0)
  Separator {
    Material {
      ambientColor 0.1 0.0 0.0
      diffueColor 0.7 0.0 0.0
    }
    Translation {
      translation 3 0 0
    }
    Sphere {
      Radius 2
    }
  }
  #Define un cono verde centrado en (-3 0 0)
  Separator {
    Material {
      ambientColor 0.0 0.1 0.0
      diffueColor 0.0 0.7 0.0
    }
    Translation {
      translation -3 0 0
    }
    Cone {
      bottomRadius 2
      hright 3
    }
  }
}
}
```

APÉNDICE III

FORMATO OBJ

OBJ [M37]

OBJ es un formato de archivo que permite definir geometrías así como algunas propiedades, inicialmente creado por Wavefront Technologies para su paquete de animación "Advanced Visualizer".

Los archivos de objeto pueden encontrarse en formato ASCII (.obj) o en formato binario (.mod). El formato de archivo es abierto y ha sido adoptado por otros desarrolladores de aplicaciones gráficas 3D además de poder ser importado/exportado desde otros programas como Poser, Autodesk Maya, Avid Softimage|XSI, blender, MeshLab, Misfit Illusion, GLC player, entre otros. Para la mayoría es un formato universalmente aceptado.

OBJ es simplemente un formato de datos que representa geometrías 3D, es decir, la posición de cada vértice, las coordenadas de la textura asociada con el vértice, la normal de cada vértice, y las caras que dan forma a cada polígono.

Este formato también soporta objetos poligonales y objetos de forma libre. Una geometría poligonal usa puntos, líneas y caras para definir los objetos, mientras que los objetos de forma libre usan curvas y superficies.

También permite soportar parámetros de suavizado para objetos curvados, además de ofrecer la posibilidad de nombrar grupos de polígonos.

Estructura del archivo

Para que un programa de gráficos 3D pueda hacer uso de un archivo de tipo obj es necesario conocer la estructura que se maneja, como lo son las palabras clave que permiten identificar cada elemento, de esta forma podrá ser interpretado correctamente. Algunos de los tipos de datos que pueden incluirse en un archivo están en inglés y se listan a continuación:

Datos De Vértice

- Vértices geométricos (v)
- Vértice de textura (tv)
- Normal de vértice (vn)
- Vértices espacio parámetro (vp)

Curvas de Forma Libre / atributos de superficie

- Matriz base, Bezier, B-Spline, Cardinal, Taylor (cstype)
- Grados (deg)
- Matriz Base (bmat)
- Tamaño de paso (step)

Elementos

- Punto (p)
- Línea (l)
- Cara (f)
- Curva (curv)
- Curva 2D (curv2)
- Superficie (surf)

Curvas de forma libre / cuerpo de superficie

- Valores de parámetro (parm)
- Ciclo de corte exterior (trim)
- Ciclo de Corte interior (hole)
- Curva especial (scrv)
- Punto especial (sp)
- Acuerdo final (end)

Conectividad entre superficies de forma libre

- Conexión (con)

Agrupar

- Nombre de grupo (g)
- Grupo suave (s)
- Unión De grupo (mg)
- Nombre de objeto (o)

Despliegue /atributos de render

- Interpolación bevel (bevel)
- Interpolación de color (c_interp)
- Interpolación de disolución (d_interp)
- Nivel de detalle (lod)
- Nombre de material (usemtl)
- Librería de material(mtl-lib)
- Shadow casting (shadow_obj)
- Ray tracing (trace_obj)
- Técnica de aproximación de curva(ctech)
- Técnica de aproximación de superficie (stech)

Sintaxis del formato de archivo OBJ [19]

El formato de archivo obj tiene una organización realmente de forma libre. Este archivo tiene muchas especificaciones de lenguajes además de los definidos aquí. Las especificaciones de lenguajes pueden ser resueltos con los primeros caracteres. Si una especificación no es reconocida, la especificación deberá ser ignorada.

```
// <name> Es un no terminal.
// ::= Es una asignación.
//...|... Significa este constructor O ese constructor.
// [...] Significa que el constructor corchete es opcional.
// {...} Significa que el constructor llave es repetido una o más veces.
// '...' Es una definición literal de cadena.
//<CR> Carácter de nueva línea.
<Start> ::= {<Vertices> <Groups>}
;
<Vertices> ::= {<Vertex>}
;
<Vertex> ::= 'v' <Float> <Float> <Float> <CR>
| 'vn' <Float> <Float> <Float> <CR>
| <CR>
;
<Groups> ::= {[<Group>] <Faces>}
;
<Group> ::= 'g'{<String>} <CR>
;
<Faces> ::= {<Face>}
;
<Face> ::= 'f'{<VertexIndex>
| '/' [<TextureIndex>
| '/' <NormalIndex>]]} <CR>
;
<VertexIndex> ::= <Integer>
;
<TextureIndex> ::= <Integer>
;
<NormalIndex> ::= <Integer>
;
<Integer> ::= {'0'..'9'}
;
<Float> ::= ['-'] {'0'..'9'} [ '.' {'0'..'9'} ]
| ['e' ['-'] {'0'..'9'} ]
;
<String> ::= ('a'...'z' | 'A'...'Z')
| {'a'...'z' | 'A'...'Z' | '0'...'9' | '_' | '+' } ]
;
;
```

Un ejemplo de un archivo en formato obj es el siguiente:

```
# Este es un comentario
# Primer vértice con coordenadas(x,y,z).
v 0.123 0.234 0.345
v ...
...

#Coordenadas de textura
vt ...
...

#Normales en (x, y, z); normales no deben ser unitarios.
vn ...
..

#Cada cara está dada por un conjunto de índices vértice/textura/normal
#Un arreglo de coordenadas que precede esto.
#De aquí f 1/1/1 2/2/2 3/3/3 es un triangulo que tiene coordenadas de
#textura y normales para sus tres vértices, y teniendo el vértice 1 de la lista
#"v", coordenada de textura 2 de la lista "vt" y la normal 3 de #la lista "vn"

f v0/vt0/vn0 v1/vt1/vn1 ...
f ...
...

#cuando se nombran grupos de polígonos o grupos de materiales las
#siguientes etiquetas aparecen en la sección de la cara

g [grupo nombre]
usemtl [material nombre]

# La segunda línea permite llamar definición de materiales en un archivo
externo .mtl.
# Cada etiqueta aplica a todas las caras siguientes, hasta que otra etiqueta del
mismo tipo aparece.
...
...
```

Otro ejemplo más claro podría ser:

#Para definir una caja

v	-0.00000	0.50000	0.70711
v	-0.00000	-0.50000	0.70711
v	0.70711	-0.50000	0.00000
v	0.70711	0.50000	0.00000
v	-0.70711	0.50000	-0.00000
v	-0.70711	-0.50000	-0.00000
v	0.00000	-0.50000	-0.70711
v	0.00000	0.50000	-0.70711

g box Side1

f 1 2 3 4

g box Side2

f 8 7 6 5

g box Side3

f 4 3 7 8

g box Side4

f 5 1 4 8

g box Side5

f 5 6 2 1

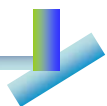
g box Side6

f 2 6 7 3

REFERENCIAS

BIBLIOGRAFÍA

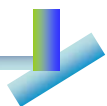
- [1] Preim, Bernhard, Bartz, Dirk, *Visualization in Medicine*, Morgan Kaufmann Publishers, 2007.
- [2] González, Rafael C., Woods, Richard E., *Digital Image Processing*, Segunda Edición, Prentice Hall, 2002.
- [3]. Hernández H., Emmanuel, Vázquez R., Gabriel, *Interfaz Gráfica 3D para simular y controlar robots móviles usando realidad virtual*, México 2002.
- [4]. Hansen, CharlesD, Johnson, Christopher R., *The Visualization Handbook*, Elsevier, 2005.
- [5]. ACM SIGGRAPH, *Visualization in Scientific Computing*, 1987.
- [6]. Lohmann, Gabriele, *Volume Image Analysis*, WileyBlackwell, 1998.
- [7]. Watt, Alan, *3D Computer Graphics*, 3era edición. Addison-Wesley, 2000.
- [8]. Lichtenbelt, Barthold, Crane, Randy, *Introduction to Volume rendering*, Prentice Hall, 1998.
- [9]. Engel, Klaus et al., *Real-time Volume Graphics*, A K Peters, 2006.
- [10]. Mc Kinley, Michael, *The Game Animator's Guide to Maya*, Sybex, 2006.
- [11]. O' Rourke, Michel, *Principles of three-Dimensional Computer Animation. Modeling, Rendering and Animating with 3d Computer Graphics*, W. W. Norton & Company, tercera edición, 2003.
- [12]. Webster, Chris, *Técnicas de animación*, Anaya Multimedia, 2006.
- [13]. Avgerakis, George, *Digital Animation Bible, Creating Professional Animation with 3ds max, LightWave, and Maya*, McGraw-Hill 2004.
- [14]. Lever, Nik, *Real-time 3D Character Animation with Visual C++*, Focal Press, 2002.
- [15]. Derakhshani, Dariush, *Introducing Maya 2008*, Wiley Publishing, Inc.
- [16]. González F., Alejandro, Reyes T., Joaquín, Ibarra R., Víctor Hugo, *Tridimensional Model of a human skull and it's bones*, 2005.



[17]. Ibarra R, Víctor H., Reyes T., Joaquín, González F. Alejandro, Tomasini, Patricia, *Análisis Preliminar de la Incidencia de los apoyos didácticos en Realidad Virtual, sobre el Aprendizaje de los Alumnos de la Facultad de Medicina de la UNAM*, 2007.

[18]. A. D. Gould, David. *Complete Maya Programming. An Extensive Guide to MEL and the C++ API*, Morgan Kaufmann Publishers, 2003.

[19]. Keith Rule, *3D Graphics File Formats*, Addison- Wesley Developers Press 1996.



MESOGRAFÍA

[M1]. El 40% de la actividad del cerebro humano se dedica a la visión.
17 de Mayo de 2008.

http://smc.neuralcorrelate.com/files/inpressfiles/DiCYT_dec14.pdf

[M2]. A Critical History of Computer Graphics and Animation.
4 de abril de 2008.

<http://design.osu.edu/carlson/history/lesson17.html>

[M3]. Imágenes estereoscópicas.

2 de abril 2008

<http://www.econectados.com/2006/09/imagenes-estereoscopicas/>

[M4]. Imagen de un casco de realidad virtual.

4 de abril 2008.

<http://www.neoteo.com/en-busca-de-una-realidad-virtual-inmersiva.neo>

[M5]. Imagen de la película The Matrix.

4 de abril 2008

<http://realidad1p.spaces.live.com>

[M6]. Presentación Marching cubes.

<http://ma1.eii.us.es/miembros/rogodi/td0708/32/pre.pdf>

4 d abril de 2008

[M7]. Direct Volume Rendering con Amira.

4 de abril de 2008

<http://www.visualizacion.unam.mx/media/docs/amira.pdf>

[M8]. Eon Games, Imagen Snails.

28 de mayo de 2008

[http://www.eongames.com/\(page_7\)/games/snails/shots.html](http://www.eongames.com/(page_7)/games/snails/shots.html)

[M9]. Cinemática inversa.

28 de mayo de 2008

<http://www.x-robotics.com/cinematica.htm>

[M10]. 3D Theory - Kinematics – Joints.

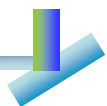
28 de mayo de 2008

<http://www.euclideanspace.com/physics/kinematics/joints/index.htm>

[M11]. Computer Character Animation Course. Aplicación de cinemática inversa.

28 de mayo de 2008

http://www.kylejmoore.com/?page_id=11



[M12]. Gaceta electrónica Facultad de Medicina.
10 de julio de 2008.
<http://www.facmed.unam.mx/publicaciones/gaceta/dic102k5/modelo.html>

[M13]. Datos de la Sala de visualización iXtli.
10 de julio de 2008.
<http://www.ixtli.unam.mx/>

[M14]. Imagen Columna vertebral, vista dorsal.
21 de julio de 2008.
<http://metodopilates.blogspot.com/>

[M15]. Imagen Columna vertebral, vista lateral.
18 de julio de 2008.
http://es.wikipedia.org/wiki/Columna_vertebral

[M16]. Imágenes de los huesos del pie humano.
21 de julio de 2008.
<http://www.juntadeandalucia.es/averroes/~29701428/salud/ssvv/pie.htm>

[M17]. Definición de absorción.
28 de agosto de 2008.
<http://www.wordreference.com/definicion/absorci%F3n>

[M18]. Definición de alpha blending.
28 de agosto del 2008.
http://www.fastgraph.com/help/alpha_blending.html

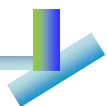
[M19]. Definición de aliasing.
28 de agosto de 2008.
http://dac.escet.urjc.es/rvmaster/asignaturas/TAG3D/Tema2_2p.pdf

[M20]. Definición de anisotrópico e isotrópico.
28 de agosto de 2008.
<http://simscience.org/cracks/glossary/isotropic.html>

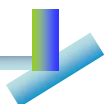
[M21]. Definición de complejidad.
28 de agosto de 2008.
http://es.wikipedia.org/wiki/Complejidad_computacional

[M22]. Definición de cuantos.
28 de agosto de 2008.
<http://es.thefreedictionary.com/cuanto>

[M23]. Definición de difracción.
28 de agosto de 2008.
<http://es.thefreedictionary.com/Difracci%C3%B3n>



- [M24]. Definición de dispersión.
28 de agosto de 2008.
<http://es.thefreedictionary.com/Dispersi%C3%B3n>
- [M25]. Definición de eficiencia.
28 de agosto de 2008.
<http://www.wordreference.com/definicion/eficiencia>
- [M26]. Definición de histograma.
28 de agosto de 2008.
<http://es.thefreedictionary.com/histograma>
- [M27]. Definición de pixel.
28 de agosto de 2008.
<http://support.microsoft.com/kb/253680/es>
- [M28]. Definición de proceso estocástico.
http://enciclopedia.us.es/index.php/Proceso_estoc%C3%A1stico
- [M29]. Definición de raster.
28 de agosto de 2008.
<http://www.pctechguide.com/glossary/WordFind.php?wordInput=Raster>
- [M30]. Definición de receptor.
28 de agosto de 2008.
<http://es.thefreedictionary.com/receptor>
- [M31]. Definición de reflexión.
28 de agosto de 2008.
<http://www.wordreference.com/definicion/reflexi%C3%B3n>
- [M32]. Definición de refracción
28 de agosto de 2008.
http://teleformacion.edu.aytolacoruna.es/FISICA/document/fisicaInteractiva/Op%20Geometrica/reflex_Refrac/Refraccion.htm
- [M33]. Definición de relación señal a ruido
28 de agosto de 2008.
<http://www.sonido-zero.com/biblioteca-de-sonido/la-relacion-senal-ruido.html>
- [M34]. Definición de sensor
28 de agosto de 2008.
<http://robots-argentina.com.ar/glosario.htm>
- [M35]. Formato DICOM.
22 de agosto de 2008.
<http://www.sph.sc.edu/comd/rorden/dicom.html#intro>



[M36]. Formato VRML.
20 de agosto de 2008.
<http://www-winfo.uni-siegen.de/vrmlHistory/docs/partVH/introduction.html>

[M37]. Formato OBJ.
22 de agosto de 2008.
<http://local.wasp.uwa.edu.au/~pbourke/dataformats/obj/>

[M38]. Miembros Inferiores Humanos, vista frontal.
22 de agosto de 2008.
http://en.wikipedia.org/wiki/Leg_bone

