

Capítulo 1. Introducción

1.1 Antecedentes

La realidad virtual es un área de la computación que ha cobrado gran auge en los últimos años, ya que permite la creación de imágenes muy realistas así como la navegación de mundos virtuales muy parecidos a la realidad creados de manera relativamente sencilla, pudiendo simularse un lugar que ya no existe, uno que se va a construir o uno imaginario. Este aumento en el uso e importancia de la realidad virtual se ha debido en gran medida al enorme incremento de la velocidad de las computadoras, tanto para procesar operaciones, como para mostrar gráficos.

El uso original de este concepto se refería a sistemas de completa inmersión[WODA94], que implicaban el uso de complejos cascos que proyectaban un espacio visual en 3D y trajes llenos de dispositivos electrónicos para mandar y recibir señales sobre la posición del cuerpo, creando la sensación de estar completamente en un mundo virtual e incluso pudiendo interactuar con él.

Actualmente, realidad virtual es un término extremadamente amplio: ha comenzado a referirse a casi todo lo que tiene que ver con la tercera dimensión en computadoras y es aplicado tanto en software, como navegadores, simuladores y generadores de escenarios virtuales como en hardware, creando dispositivos electrónicos como cascos, guantes y trajes.

Es importante cambiar la impresión que se tiene en general de que la realidad virtual tiene que ver únicamente con movimiento. En el área de la graficación por computadora, un campo muy estudiado es aquel que busca presentar imágenes estáticas lo más semejantes a la realidad, ya sea creadas artificialmente o bien mejorando la calidad de imágenes obtenidas por cualquier medio. La figura 1.1 muestra una cafetera generada por computadora.



Fig. 1.1.- Imagen estática de una cafetera generada utilizando el programa Studio 3D.

1.2 Mundos virtuales

Un mundo virtual es un conjunto de objetos tridimensionales que representan objetos reales o imaginarios con los que se puede interactuar en cierto grado, provocando la sensación de ser reales. Este grado de interacción se le conoce como nivel de inmersión y depende de la cantidad de sentidos que perciban sensaciones de esos mundos. Los niveles de inmersión más utilizados son los siguientes:

+*Visión*: a pesar de que los mundos tridimensionales son proyectados en una pantalla plana, el uso de técnicas de iluminación, texturas en objetos y perspectiva pueden hacer parecer una imagen plana como un cuerpo tridimensional. El realismo es aumentado mediante el uso de transformaciones sobre los objetos, como rotación, traslación y escalamiento, que simulan de una forma muy efectiva, los movimientos de una persona dentro de un escenario, como avanzar hacia delante, atrás o a los lados y girar la cabeza en cualquier sentido. Al hecho de “moverse” dentro del mundo virtual se le conoce como navegar en él.

+*Visión y oído*: el hombre escucha “tridimensionalmente” pudiendo percibir de qué dirección proviene un sonido y qué tan lejos se originó. Físicamente, el percibir la dirección de donde viene sonido ocurre porque primero llega a un oído y luego al otro; la intensidad del mismo indica su lejanía. Hardware y software especializado puede simular sonido tridimensional controlando el momento y la intensidad en que llega el sonido a cada oído.

+*Visión, oído y tacto*: el sentido del tacto es quizá, el más difícil de simular y aún está muy limitado. Guantes y trajes especiales que informan y reciben información del movimiento de la persona en el mundo virtual pueden oponer resistencia hasta cierto grado a algunos movimientos realizados por la misma, tratando de simular así el tomar objetos o topar con alguno.

Los programas capaces de permitir navegar un mundo virtual son llamados navegadores y su manera de interactuar con el usuario (recibir órdenes de movimiento)

puede ser tan compleja como el uso de guantes o tan simple como un mouse o incluso un teclado.

Actualmente la creación de mundos virtuales tiene aplicaciones muy importantes no sólo relacionadas con el entretenimiento, sino con la investigación, el entrenamiento, la enseñanza y la construcción. Varias de estas aplicaciones son [WODA94]:

+Químicos pueden ver complejas moléculas en tiempo real y visualizar nuevas formas de crear materiales.

+Estudiantes pueden interactuar a través de la computadora con la representación del sistema solar para aprender cosas como la gravedad y la masa.

+Pilotos pueden entrenarse sin la necesidad de volar o bien puede probarse un diseño costoso de un avión sin peligro de estrellarse.

+Un arquitecto y su cliente pueden “ver” una nueva casa, incluso caminar dentro de ella, antes de comenzar a construirla.

+Un turista puede visitar la torre de Pizza estando en México o conocer una embarcación hundida hace cientos de años.

1.3 Motivación

Para que un navegador pueda mostrar en una pantalla de dos dimensiones un mundo virtual tridimensional es necesario realizar una gran cantidad cálculos matemáticos, los cuales se incrementan conforme se aumenta el realismo que se desea provocar. Este

aumento de cálculos se refleja directamente en el tiempo de respuesta del navegador, haciéndolo más lento y provocando la pérdida de la sensación de estarse moviendo dentro del escenario virtual.

Un navegador simula el movimiento dentro de un mundo virtual mediante el despliegado rápido de una serie de imágenes que varían muy poco entre ellas, tal como se hace en el cine o la televisión. Estas imágenes son calculadas a partir de los objetos originales y las órdenes de movimiento del usuario. Para que el ojo humano perciba estas series de imágenes o cuadros como una animación continua, deben mostrarse al menos 6 cuadros por segundo[MUDE96]. Por lo tanto, si la escena es muy compleja y el navegador emplea más de 0.16 segundos en generar una imagen y presentarla, el efecto de movimiento continuo no se logrará.

Debido a estas razones, el estudio y desarrollo de modelados de objetos tridimensionales que faciliten cálculos sobre los mismos y hagan un uso eficiente de la memoria, así como de algoritmos veloces para transformar y presentar estos objetos en pantallas bidimensionales, haciéndolos parecer reales es muy importante.

Así como en la realidad virtual se puede hablar de niveles de inmersión, en los navegadores es más común referirse a niveles de realismo. Este nivel de realismo puede ser medido de acuerdo a la cantidad de técnicas utilizadas para dibujar los objetos, las cuáles se aplican a diferentes propiedades de los cuerpos (forma, color, textura, brillantez, transparencia, etc.) y del ambiente (luces, sombras, etc.).

En este trabajo se describe la implementación de un navegador que utiliza la estructura de árboles BSP para representar objetos y se propone un mecanismo rápido y eficiente para la construcción de estos árboles en tiempo de ejecución, obteniendo la información de los escenarios de archivos de texto que utilizan el formato desarrollado por Eric Haines (formato NFF) y el formato VRML para representar escenas, realizando finalmente un análisis de la cantidad de técnicas utilizadas para mostrar las escenas y el tiempo de respuesta del navegador, así como el tiempo de creación del árbol BSP utilizando diferentes mecanismos para crearlo.

Para las escenas con las que se realizaron las pruebas del navegador se utilizaron las rutinas en C escritas por Eric Haines que describen escenarios como esferas y tetraedros en una estructura recursiva, engranes, montañas realizadas con fractales, árboles y anillos. Además, se desarrolló un traductor para convertir archivos en formato NFF al lenguaje VRML 1.0, con la finalidad de probar éstas escenas en otros navegadores comerciales. Finalmente, se eligió un subconjunto de la especificación VRML 1.0, que comprende las instrucciones básicas para poder representar escenas relativamente complejas, y se desarrolló un traductor de este subconjunto al formato NFF.

1.4 Alcances y limitaciones

El software desarrollado tiene las siguientes características:

+Cualquier archivo en formato NFF se puede traducir al lenguaje VRML 1.0.

+Al árbol BSP generado por el navegador, el cual es llamado en este trabajo pseudo-óptimo, es obtenido en un tiempo de ejecución razonable.

+El navegador BSP permite:

- Definición de objetos en millones de colores.
- Cámara ortogonal y perspectiva.
- Creación del árbol BSP utilizando diferentes métodos para obtener el nodo raíz.
- Iluminación de polígonos usando las técnicas de Lambert, Gouraud y Phong.
- Activar y desactivar la supresión de caras ocultas.
- Activar y desactivar la luz ambiental, atenuación atmosférica, reflexión difusa y reflexión especular.
- Lectura de escenas de archivos en formato NFF.
- Lectura de escenas de archivos en formato VRML 1.0 (sólo un subconjunto de la especificación).
- Salvar y cargar una escena como árbol BSP.
- Cambiar color a todos los objetos.
- Cambiar el factor de partición para la generación de esferas, cilindros y conos.
- Activar y desactivar el proceso de pintado fuera de pantalla.
- Dibujar información sobre las particiones de los polígonos.
- Monitorear el tiempo de respuesta del navegador.

- Observar información de la escena como el número de polígonos y vértices.
- Simular cualquier movimiento de la escena.
- Tiempo de respuesta razonable.
- Robustez.

Las principales limitantes del software son:

+No todos los archivos en formato VRML 1.0 pueden ser traducidos al formato NFF debido a que como ya se mencionó, se eligió sólo un subconjunto la especificación VRML.

+No se asegura que el árbol BSP generado sea el óptimo, ya que el algoritmo que encuentra el óptimo es mayor o igual a $O(n!)$, siendo n el número de polígonos que componen el escenario.

+El navegador BSP presenta las siguientes limitaciones:

- No detecta colisiones.
- Todos los objetos son poliédricos.
- Está limitado a polígonos convexos.
- El límite de objetos en escena es finito.
- No aplica técnicas avanzadas como transparencias y sombras.
- Sólo maneja una luz.

1.5 Hardware y software utilizado

La plataforma de desarrollo y pruebas fue una PC con 64 Mb en RAM y un procesador de 400 MHz. El navegador fue implementado en el lenguaje Visual Basic 6.0 utilizando funciones de dibujo de Windows para pintar líneas y polígonos, sin embargo también se implementó en este trabajo una rutina para el dibujo de polígonos .

1.6 Organización del documento

Este trabajo está dividido en siete capítulos descritos a continuación:

+Capítulo 1.- Introducción

Describe una visión actual de la realidad virtual y de los mundos virtuales, mostrando algunos usos y problemas enfocados al desarrollo de navegadores. Presenta una justificación para la realización de este trabajo así como los alcances y limitantes obtenidos durante el desarrollo del mismo.

+Capítulo 2.- Modelado

Este capítulo es un compendio de los principales modelados de objetos tridimensionales, presentando diferentes maneras de modelar superficies, sólidos y escenas y haciendo comparaciones entre éstos.

+Capítulo 3.- Visualización

Capítulo en el que se describen técnicas y algoritmos relacionados con la visualización de objetos tridimensionales en pantalla, como la determinación de superficies visibles e iluminación de caras.

+Capítulo 4.- Fundamentos matemáticos

En este capítulo se presentan los fundamentos matemáticos y geométricos relacionados con la implementación del navegador en este trabajo, detallando las transformaciones para modificar la posición de los objetos y las proyecciones para presentarlos en pantalla.

+Capítulo 5.- Árboles BSP

Se realiza una descripción detallada de la estructura, generación y algoritmos de los árboles BSP, comparando varios métodos para la creación de los mismos y presentando aspectos relacionados como la división de polígonos por un plano y la elección del polígono raíz.

+Capítulo 6.- Desarrollo del sistema

En este capítulo se describe detalladamente las estructuras, algoritmos y esquemas utilizados para desarrollar el navegador, mencionando y explicando aspectos importantes del mismo, como la interpretación de escenas, generación de primitivas y la manera de simular movimiento.

+Capítulo 7.- Pruebas y resultados

Se muestran los resultados obtenidos al realizar diversas pruebas con el navegador implementado, como los diferentes tiempos de respuesta al utilizar diferentes técnicas para mostrar los objetos y el tiempo tomado en generar el árbol BSP utilizando diferentes enfoques para la elección del nodo raíz.

Cabe aclarar que todas las imágenes de este documento que representan objetos tridimensionales fueron creadas con el software descrito en este trabajo, excepto aquellas en cuyo pie de figura se anota explícitamente.