



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

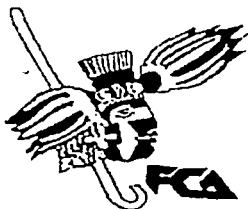
FACULTAD DE CONTADURÍA Y
ADMINISTRACIÓN

**REQUERIMIENTOS
PARA EL DESARROLLO
DE UNA OFICINA VIRTUAL**

SEMINARIO DE INVESTIGACIÓN INFORMÁTICA

Que para obtener el título de:

LICENCIADO EN INFORMÁTICA



Presenta:

Julián Jesús Romero López

Asesor:

L.A.E. Y M.A. Luis Eduardo López Castro

México, D.F.
Actualizado a:

1996
2001

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Paginación

Discontinua

AGRADECIMIENTOS.

Quiero darle gracias a Dios que me da la oportunidad de terminar con este proyecto importante para mí.

A mi esposa Leticia quien con sus sacrificios, desvelos y regaños me ayudo a terminar y seguir adelante.

Agradecer a Gilda y Julián, Luz Irene y Sergio Ricardo Romero López por su paciencia, apoyo, comprensión y sobre todo el soportar mi carácter.

A ti Mamiluz que con tus buenos consejos y tu forma de educarnos me hiciste reflexionar y darle un valor importante a la vida.

A Javier López Ramos quien fue la inspiración en mi persona para concluir mis estudios.

Un valioso agradecimiento al M. en A. y L.A.E. Luis Eduardo López Castro quien demostró interés, apoyo e inquietud para la realización de este proyecto.

A todos los López Ramos, Guerrero Dichi, López Leal y sus familias, a la familia Buendía Romero, Ortiz Soriano y a Julián e Irene Romero, por creer en mis errores y aciertos, alegrías y tristezas, sueños y fantasías.

A Constantino López (botitas) y a Maria Leal por todas sus atenciones desde el primer día que me permitieron entrar a su casa.

Al Churro, Tililica, Sorullo, Bombón, Champion, Pérez y a Nacho, quienes con su compañía, humor, y buenos momentos que compartimos me ayudaron a formarme como persona.

A Hector Perales y su familia quienes sin conocerme se desprendieron de su tiempo para compartir su interés en la Realidad Virtual y demostraron con su amistad que todavía hay personas que saben cual es el significado de esta palabra.

Al Oax. mary y nemesio por su amistad en tan corto tiempo dentro y fuera del trabajo.

A BETBIRAI por llenar un hueco más en mi vida y ser mi sobrina CONSENTIDA.

Y en general a todas aquellas personas que formaron y forman parte de mi vida y que colaboraron con sus experiencias, anécdotas, sueños y cualquier comentario para motivarme a realizar esta investigación.

GRACIAS A TODOS USTEDES.

12

Agradecimientos		III.
Contenido		V.
Parte I. Marco Problemático		
	Prefacio	IX.
Parte II. Planteamiento del Problema		
	Definición del Problema	Pag. 1.
	Fundamento	Pag. 1.
	Objetivo de este Trabajo	Pag. 1.
	Objetivos Específicos	Pag. 2.
	Alcances	Pag. 2.
	Limitaciones	Pag. 3.
Parte III. Marco Teorico Conceptual		
	Capitulo 1 Conceptos de Realidad Virtual (RV).	
	1.1 Introducción	Pag. 4.
	1.2 Que es la Realidad virtual	Pag. 6.
	1.3 Captar la Realidad Mediante la Ilusión	Pag. 8.
	Capitulo 2 Elementos de la Realidad Virtual (RV)	Pag. 10.
	2.1 Inmersión	Pag. 10.
	2.2 Niveles de Inmersión	Pag. 10.
	2.3 Sensorama	Pag. 13.
	2.4 Viaje a las Estrellas	Pag. 14.
	2.5 Navegación	Pag. 14.
	2.6 Manipulación	Pag. 15.
	Capitulo 3 Ciberespacio.	
	3.1 Historia y Generalidades	Pag. 16.
	Capitulo 4 Requerimientos para crear un Mundo Virtual.	
	4.1 Entrada-Salida	Pag. 22.
	4.2 Dispositivos Estandar de Entrada y Salida (E/S)	Pag.23.
	4.3 Guantes de Datos	Pag. 23.
	4.4 Posición Absoluta	Pag. 28.
	4.5 Dispositivos de Visualización en Cascos	Pag. 29.
	4.6 Posición Absoluta	Pag. 32.
	4.7 Rastreadores	Pag. 32.

4.8	Características de Rastreo de los Equipos de Cabeza	Pag. 33.
4.9	Investigación de Rastros Alternativos	Pag. 35.
4.10	Booms	Pag. 36.
4.11	Salida	Pag. 37.
4.12	Luz	Pag. 37.
4.13	Efectos 3D	Pag. 39.
4.14	Desde los Estereoscopios hasta los Cascos HMD	Pag. 40.
4.15	Anaglifos	Pag. 41.
4.16	Películas en 3D	Pag. 41.
4.17	Las Computadoras También Pueden Hacerlo	Pag. 42.
4.18	Sistema de Obturación LCD de SEGA	Pag. 42.
4.19	Tecnología Estereográfica de PI-CELL	Pag. 43.
4.20	El Movimiento de la Cabeza es un Hecho de la Vida	Pag. 44.
4.21	Sonido en 3D	Pag. 45.
4.22	Facilidades de Audio del Equipo de Cabeza	Pag. 46.
4.23	Tacto y Propiocepción	Pag. 48.
4.24	Retroalimentación Táctil	Pag. 48.
4.25	Gravedad	Pag. 49.
4.26	Traje	Pag. 49.
4.27	Bolas de Fuerza	Pag. 51.
4.28	Joystick y Joybox	Pag. 51.
4.29	Varas	Pag. 51.
4.30	Controladores de Voz	Pag. 51.
4.31	Facilidades Táctiles	Pag. 52.
4.32	Pequeñas Bolsas y Microagujas	Pag. 53.
4.33	Termodos	Pag. 54.
4.34	Cintas de Andar	Pag. 55.
4.35	Bicicleta	Pag. 56.
Capítulo 5	Presentación de Imágenes Virtuales.	
5.1	El Sistema RV. Máquinas y Aparatos	Pag. 57.
5.2	Gráficas Inmersivas Tridimensionales (3D)	Pag. 58.
5.3	Opaco o Transparente	Pag. 58.
5.4	Generación de Polígonos en Tiempo Real	Pag. 60.
5.5	Mejora Háptica	Pag. 62.
5.6	Reproducción de Imágenes en el Espacio Real	Pag. 64.
5.7	Distribución	Pag. 65.
5.8	Modos de Presentación: el Punto de Vista del Usuario	Pag. 65.
5.9	Presentaciones Proyectadas	Pag. 67.
5.10	Complejidad de Imágenes	Pag. 68.
5.11	Refinamiento Adaptativo	Pag. 69.

Parte IV.	Diseño del Experimento.	
Capítulo 6	Investigación y Desarrollo de un Guante Virtual	Pag. 70.
6.1	El Diseño del Power Glove	Pag. 70.
6.2	Modificaciones al Guante para Funcionar en la PC	Pag. 71.
6.3	Método del Puerto Paralelo	Pag. 72.
6.4	Examen del Cableado del Power Glove	Pag. 74.
6.5	Uso del Cable Original	Pag. 75.
6.6	Trabajo de Investigación.	Pag. 75.
6.7	La conexión a la Computadora	Pag. 77.
6.8	Conexión de la Alimentación	Pag. 78.
6.9	Alimentación Externa	Pag. 78.
6.10	Prueba de la Conexión	Pag. 80.
6.11	El Sistema de Posicionamiento por Ultra Sonidos	Pag. 81.
6.12	Giro	Pag. 84.
6.13	Programación del Power Glove	Pag. 85.
6.14	Modo de Baja Resolución	Pag. 85.
6.15	Señalización del Guante	Pag. 86.
6.16	Bucle para los 8 Bits	Pag. 86.
6.17	Uso de los 8 Bits	Pag. 87.
6.18	Modo de Alta Resolución	Pag. 87.
6.19	Algo más Cercano al Código C	Pag. 88.
6.20	Retardo	Pag. 90.
6.21	Lectura del Guante y código	Pag. 90.
6.22	Software de Representación	Pag. 93.
6.23	Entrada del Usuario	Pag. 93.
6.24	Transformación y Proyección de Vértices	Pag. 94.
6.25	Ordenación de Objetos por su Coordenada Z	Pag. 95.
6.26	Eliminación de Superficies Ocultas	Pag. 96.
6.27	Color	Pag. 97.
6.28	Dibujo	Pag. 97.
6.29	Creación de un Objeto	Pag. 97.
6.30	Creación de un Archivo PLG	Pag. 100.
Capítulo 7	Jugando al Frontón	Pag. 103.
7.1	Jugando al Frontón	Pag. 103.
7.2	Programación del Juego de Frontón	Pag. 103.
7.3	Construcción del Equipamiento	Pag. 104.
7.4	La Pista de Juego de Frontón	Pag. 104.
7.5	La Pelota	Pag. 109.

7.6	La Raqueta	Pag. 111.
7.7	Movimiento de la Pelota	Pag. 113.
7.8	Animación	Pag. 114.
7.9	Movimientos del Cuerpo y del Punto de Vista	Pag. 115.
7.10	Interfaz de Movimientos	Pag. 117.
7.11	El bucle de Control	Pag. 117.
7.12	El Teclado	Pag. 118.
7.13	Up y Down (Arriba y Abajo)	Pag. 119.
7.14	Left y Right (Izquierda y Derecha)	Pag. 120.
7.15	Giro de la Cabeza	Pag. 120.
7.16	Detección de Colisiones	Pag. 121.
Parte V.	Análisis de Resultados.	
Capítulo 8	Ventajas y Desventajas de la RV.	
8.1	Ventajas y Desventajas	Pag. 123.
8.2	Escritorios Virtuales	Pag. 125.
8.3	Conocimientos Ejecutivos y Modelos de Corporación	Pag. 126.
8.4	Mercado Mejorado	Pag. 128.
8.5	Factores Humanos en Ambientes Virtuales	Pag. 129.
Parte VI	Reporte Final.	
	Conclusiones	Pag. 131.
	Temas Pendientes para Futuras Investigaciones	Pag. 133.
	Ventajas y Desventajas de Nuestro Enfoque	Pag. 134.
Parte VII	Apéndices.	
A.	Glosario y Referencias	Pag. 137.
B.	Bibliografía	Pag. 161.
C.	Fuentes Adicionales de Información	Pag. 162.
D.	Estadísticas del Mercado de la RV	Pag. 163.
	Costos de Dispositivos HMD's	Pag. 163.
	Costos de Hardware	Pag. 163.
	Software	Pag. 164.
E.	Código del Programa	Pag. 165.

PARTE I. MARCO PROBLEMÁTICO.

PREFACIO.

Todavía me acuerdo del primer vídeo juego casero el Atari Pong, eran dos líneas cortas simulando las raquetas y un cuadro simulando la pelota, un fondo negro que era la cancha, no me imagine que los videojuegos serian la base de lo que hoy en día se conoce como Realidad Virtual, una nueva generación de software e interfases que combinada con la Informática se ven beneficiadas áreas como la medicina, los negocios, la educación, finanzas, gobiernos, etc. Por mencionar algunas.

El ser un aficionado a los videojuegos, películas, documentales y comentarios de personas que creen en la Realidad Virtual me motivaron e inspiraron a realizar este trabajo de investigación sobre Realidad Virtual, donde espero que los estudiantes de hoy día se motiven para continuar experimentando con ella.

¿Que es la Realidad Virtual? Soñar es una forma de realizar todos aquellos deseos que se nos niegan de día. Soñando podemos volar, alcanzar las mas altas cumbres del planeta o descender a las mas profundas simas. Por desgracia, en general no podemos dominar los sueños y al final nuestra realidad acaba limitada a lo que nos brinda la percepción de nuestros sentidos. Franquear la barrera de los sentidos y ofrecer sensaciones de extremo realismo debería ser casi como un sueño controlado.

En la introducción de este trabajo, se presenta el problema que abordamos, el objetivo de este experimento, así como sus alcances y limitaciones.

En el capítulo 1 se presenta la RV, ha tratado de ofrecer una interfaz entre nuestros sentidos y las maquinas, pero entonces que es, un teclado y una pantalla también nos ofrecen una interfaz adecuada aunque no nos brindan una experiencia realista. El termino clave para definir qué es la RV, radica en la "inmersión". La frontera entre los sistemas de Realidad Virtual y los sistemas convencionales está en que mientras que en los primeros percibimos la información como una verdad representada mediante caracteres, imágenes, audio o incluso gráficos tridimensionales, en los segundos nos encontramos cara a cara con la información, tal como si compartiese el mismo espacio en el que nos encontramos. Nos hallamos inmersos en esa realidad.

La frontera entre un sistema inmersivo y uno que no lo es, depende más bien de los gustos de cada cual y de su capacidad de inmersión, que de una ley que marque la diferencia. Un videojuego de lo más sencillo puede resultar una experiencia totalmente realista para un niño, mientras que la más sofisticada máquina con los más complejos dispositivos puede no dejar de ser un videojuego para un adulto escéptico.

En el capítulo 2 se consideran los Elementos Básicos que se deben incorporar en cualquier Sistema de Realidad Virtual, no existe una regla fija que mencione cuales deben de ser, pero casi la mayoría incluye: La Inmersión, Navegación y Manipulación.

El capítulo 3 trata sobre la Historia y Generalidades del Ciberespacio se recopila información con respecto al ciberespacio que son todos aquellos espacios y paisajes tridimensionales que se encuentran en la computadora es decir, que engañan a la vista y al oído e incluso al sentido del tacto, conocido también como telepresencia, que no es otra cosa que la presencia aparente en lugares muy lejanos.

Es decir el ciberespacio representa la primera interfaz, los diseños tridimensionales con vistas múltiples, imágenes sombreadas o animación, en donde los usuarios no tendrán dificultad en comprender, ni vacilación en adoptar una tecnología que les permita recoger una parte de ADN y hacerla girar para comprender su forma, volar como Supermán a través de una galaxia, o simplemente formar piezas usando herramientas y ver enseguida los resultados.

El capítulo 4 se refiere a los Requerimientos para crear un mundo virtual: Probablemente el dispositivo más usual para el control y entrada en un sistema virtual es un guante, "instrumentado" con fibras ópticas flexibles que recorren cada una de las articulaciones de la mano. Los equipos de cabeza (HMDs Head-Mounted Displays) son denominados unidades de presentación montados sobre la cabeza algunos parecen cascos mediante los cuales los dispositivos de visión quedan suspendidos enfrente de los ojos del usuario. El sonido es importante para la percepción espacial de una persona y es más efectivo cuando las ayudas visuales son mínimas. Los efectos de sonido con frecuencia reavivan los efectos visuales. El traje es básicamente un guante de datos específico para todo el cuerpo. Está instrumentado con el mismo tipo de cable de fibra óptica que recorre un guante de datos. Al moverse, curvarse o hacer señas el usuario, el sistema toma coordenadas para cada parte del traje, rastreando dinámicamente una extensa serie de acciones, 20 o más sensores recogen la mayoría de las articulaciones del cuerpo.

Cada uno de estos sistemas refuerza un aspecto de la ilusión del usuario durante su inmersión en el mundo virtual. Todos ellos están integrados por unos sistemas accesibles de hardware, software y electrónica, de tal forma que actúan coordinadamente.

El capítulo 5 analiza la Presentación de Imágenes Virtuales, Un sistema RV está compuesto de muchos sistemas. Apoyados en un fundamento básico de hardware, software y electrónica, se trata de sistemas independientes desarrollados para producir efectos visuales, auditivos y táctiles que son utilizados en entornos virtuales.

En el capítulo 6 se presenta una aplicación con un Guante Virtual. Como ya sabemos, el Power Glove fue diseñado para el Nintendo, no para una computadora. Y este se usa como un dispositivo de entrada a la computadora que trata de realizar movimientos como lo hace la mano.

La computadora puede determinar dónde está la mano del usuario en un espacio tridimensional, si el puño está cerrado o no y si la mano se está girando o no.

En el capítulo 7 se explica que es lo que se necesita para poder jugar al Frontón, y lo primero es conectar a la computadora el Power Glove y su interfaz, la creación de los elementos básicos del juego, considerar la movilidad del guante en el mundo virtual. Y finalmente, llegar a la cuestión de la tridimensionalidad durante el juego, es decir utilizar la percepción de profundidad para determinar dónde está la raqueta en relación con nosotros mismos.

Las Ventajas y Desventajas de la Realidad Virtual se discuten en el capítulo 8. Se investigaron las diversas categorías de aplicaciones profesionales de la Realidad Virtual y los posibles daños físicos, sociológicos, financieros psicológicos que los usuarios de esta tecnología podrían tener.

Finalmente se presentan las conclusiones sacadas de este experimento con el guante.

PARTE II. PLANTEAMIENTO DEL PROBLEMA.

DEFINICION DEL PROBLEMA.

¿ Cómo fomentar el uso y desarrollo de Sistemas Virtuales dentro de las Instituciones Públicas y Privadas ?

FUNDAMENTACION.

Actualmente las computadoras no son muy amigables, es complicado su uso y aprovechamiento.

He notado que en nuestro país no se le ha dado la importancia necesaria a la computadora como una herramienta más de trabajo, tanto en las empresas medianas o chicas, como en gran parte del sector público.

Se le tiene miedo o un gran respeto al decir: Si oprimo una tecla diferente se van a borrar o a perder los datos, la computadora me va a sustituir o a dejar desempleado por eso no aprendo a utilizarla o simplemente se dice son muchos comandos y en un idioma diferente, no entiendo.

Por lo anterior, es digno de atención si estamos conscientes de que la mayor parte de los sistemas de información desarrollados en la actualidad se pueden catalogar como herramientas administrativas y utilizando los beneficios de la RV se pueden obtener en primera instancia una transmisión de información rápida concisa y clara, en vista de que el usuario forma parte del sistema y por ende el desempeño de sus funciones mejora y en segunda instancia por todo lo que la RV ofrece, como un entorno creativo que influye en gran medida sobre la percepción del usuario.

OBJETIVO GENERAL.

Que el lector entienda los conceptos, elementos y requerimientos básicos, que se necesitan para crear interfases tecnológicas en las diferentes áreas de una empresa (finanzas, producción, formación y Selección de personal, mercadotecnia, etc.) donde se pueda aplicar la Realidad Virtual.

OBJETIVOS ESPECIFICOS.

Mostrar un método más conveniente, sencillo y barato de conexión (interfaz) entre el guante y la computadora.

Crear una fuente de alimentación (corriente) externa para el guante.

Utilización de un sistema de posicionamiento por medio de ultrasonido.

Desarrollo de los algoritmos y la programación para controlar el guante.

Aplicación de archivos PLG (Objetos).

Elaborar un juego de frontón simulado que incluya un campo de juego o pista, raqueta, pelota así como el manejo de la raqueta y golpear la pelota.

ALCANCES.

1.- El estudio y experimento del guante se desarrollo en una Computadora con un procesador 386 con una velocidad de 120 mhz., con un Monitor a color VGA de 14", con disco duro de 1.2 G.B. de capacidad, un puerto paralelo, un puerto serial, entrada para teclado y mouse, sistema Operativo Windows 95.

2.- Se modifiko el guante para que funcione en la computadora investigando y proporcionando una lista completa de componentes pa4ra la interfaz del guante con el puerto paralelo.

3.- Se logro utilizar el cable original de guante, la conexión con la computadora y la utilización de una alimentación de corriente alterna.

4.- Se utilizo un compilador turbo C/Borland, para recompilar el código de alta resolución escrito en ensamblador y proporcionar una versión en lenguaje "C".

5.- Se consiguió una herramienta llamada REND386 para la creación de objetos virtuales y del juego de frontón ya que al parecer la forma mas fácil y sencilla es utilizando polígonos.

6.- Se consiguió un guante para el juego de entretenimiento Nintendo junto con su propio sistema de posicionamiento por medio de ultrasonido.

LIMITACIONES.

1.- Se limito a utilizar una PC por lo tanto no se comprobó si se puede utilizar en Estaciones de Trabajo ó en MainFrame.

2.- No se utilizo en otro tipo de Sistema Operativos Robustos o en varias maquinas como por ejemplo UNIX.

3.- No existe la inmersión porque para el juego únicamente se limito a usar el guante y no un casco o gafas virtuales.

4.- No se utilizó un traje virtual por lo tanto no se aprecia completamente otro elemento básico para sistemas virtuales que es la navegación.

5.- Se limito a utilizar un solo lenguaje de programación para la creación de mundos virtuales REND386.

6.- De los tres elementos básicos de la realidad virtual únicamente se utilizo la Manipulación y la Navegación faltando experimentar la Inmersión.

7.- Con este tipo de guante (Power Glove) no es independientemente al cien por ciento hay algunos movimientos que se tienen que realizar mediante el teclado.

8.- El guante (Power Glove) esta limitado a los movimientos que realizaría un joystick en mundos virtuales.

PARTE III. MARCO TEORICO CONCEPTUAL.

CAPITULO I.

CONCEPTOS DE REALIDAD VIRTUAL.

INTRODUCCIÓN.

La Realidad Virtual no es intimidatoria ni es del dominio exclusivo de adictos a los videojuegos y a la tecnología. Sus aplicaciones tampoco están restringidas a lo puramente tecnológico o científico; es un medio creativo de comunicación. Tendrá influencia en como diseñar las cosas o llevar nuestros negocios, como enseñar a los niños y como tratar enfermedades. Si se tiene en mente la sensación de estar soñando, se puede tener una idea de lo que es la Realidad Virtual.

Del mismo modo que en sueños convive lo que tiene sentido y lo que no lo tiene, en la Realidad Virtual se mezclan libremente lo lógico y lo ilógico. ¿Como reconstruiríamos un sueño si pudiéramos conectar nuestra imaginación al motor de una potente computadora? inmediatamente imágenes visuales vendrían a nuestra mente y podríamos maniobrar con ellas. La Realidad Virtual nos ofrece la posibilidad de sumergir nuestros sentidos en experiencias nuevas.

Los entornos o escenarios de la Realidad Virtual pueden ser predefinidos o enfocados de tal manera que el usuario obtenga una destreza específica, como si estuviese realmente en ellos. Un entorno de Realidad Virtual independiente del idioma o la cultura puede hacerlo todo mucho mas claro y puede suponer un salto hacia delante para cualquiera desde un piloto de combate hasta un neurocirujano, pasando por un comerciante, y llegando a la educación.

Los escenarios virtuales pueden promover un intercambio dinámico de factores que tienen influencia en las decisiones. Esto es de especial importancia en el comercio. En la fabricación, los participantes pueden acceder a un proceso interactivo de diseño sin los costos y retrasos que la actual construcción lleva consigo. Los clientes también se benefician de esta nueva forma de experimentar las cosas. Los escenarios virtuales ofrecen una oportunidad de probar productos o servicios antes de comprarlos o invertir grandes cantidades de dinero para desarrollarlos.

De esta forma un ejecutivo de la industria automovilística, que debe estar convencido de que la financiación del diseño de un nuevo coche esta garantizada, podría de una manera sencilla ser invitado a sentarse y a conducir una realización virtual del modelo propuesto. Dependiendo de su reacción, el diseño puede ser descartado, alterado o regenerado para que el pueda volver a probarlo. Solo después de la aprobación del producto se inicia su construcción.

De nuevo, todo esto se realiza con unos recursos humanos mínimos y sin los grandes costos y retrasos de la construcción paso a paso. Esta convergencia crítica del poder de la reproducción de imágenes y la computación que llamamos Realidad Virtual ha abierto posibilidades para compartir experimentos y experiencias mas allá de lo que nadie pudiera imaginar. Ofrece la posibilidad no solo de aumentar los esfuerzos por el bienestar de la humanidad, sino también de despertar nuestras ilusiones, sueños o fantasías.

Aunque la eficacia del conocimiento de experiencias reales sea única, las experiencias virtuales pueden acortar las distancias entre lo que debemos aprender y lo que somos capaces de aprender mediante la experiencia. Se pueden inventar experiencias virtuales que son imposibles en la realidad e incluso se pueden ampliar las fronteras de nuestros sentidos mas allá de las que ahora conocemos.

Ya no se trata de si se puede hacer o no, sino de como aplicarlo de la mejor forma. Cada vez que la tecnología de la Realidad Virtual se utiliza, es la variación de un tema que tiene su punto de partida en el mundo real. Y hay espacio más que suficiente para distintos mundos generados por computadora desde aquellos basados en la realidad hasta los que hacen divagaciones en la fantasía.

Es importante mencionar que ya existe una investigación en la Biblioteca de la Facultad de Contaduría y Administración, con el nombre de Sistemas de Bases de Datos en Entornos Virtuales, la cual consiste sin lugar a dudas señalar (tomando como caso de estudio a la Realidad Virtual) el ciclo de desarrollo e implantación de nuevas tecnologías y el como, la convivencia de éstas, con tecnologías ya aceptadas constituye el camino más seguro para su integración.

La intención es que este trabajo sea una interesante contribución a la bibliografía existente en este tema y en otro sentido la posibilidad de compartir una experiencia.

Asimismo todas las imágenes y síntesis de información contenida en cada uno de los capítulos fue realizada por medio de las bibliografías, revistas y consultas en internet que se mencionan en la parte de Apéndice.

QUE ES LA REALIDAD VIRTUAL.

El propósito central del sistema interactivo es el que se conoce como una virtualidad del sistema. Este es un término muy general que abarca todos los campos en los que la mente, los efectos y las ilusiones son lo que importa.

Entendiendo por virtualidad de una cosa aquello que parece ser como distinto de su "Realidad" más concreta, que puede no ser importante. El otro término más próximo es "Entorno Mental". Algunas personas en especial estudiantes han presionado para que se retenga el término "Virtualidad" aun cuando cause confusión entre los usuarios de los así llamados "Sistemas Virtuales", que significan sistemas reales configurados por una computadora.

Un sistema interactivo es una serie de presentaciones destinadas a afectar la mente, igual que en una película. Esto no es una analogía casual; éste es el tema central. Se usa el término "Virtual" en su sentido tradicional, como contrario de "Real". La realidad de una película abarca la forma en que se pintó el decorado y donde estuvieron colocados los actores entre las tomas, la virtualidad de una película es lo que parece estar en ella. La realidad de un sistema interactivo incluye su estructura de datos y en qué lenguaje fue programado.

Una "Virtualidad", entonces es una estructura de la apariencia, la percepción conceptual de lo que se está creando. Es ese entorno, con sus cualidades para reaccionar y sentir, lo que importa, no la "Realidad" irrelevante de los detalles de implementación. Y crear esa apariencia como un todo integrado es la verdadera tarea de diseñar e instrumentar la Virtualidad. Esto es tan cierto para una película como lo es para un procesador de palabras.

Entonces podemos decir que la palabra VIRTUAL se define como "Que existe o resulta en esencia o efecto pero no como forma, nombre o hecho real". Y REALIDAD es "La cualidad o estado de ser real o verdadero". Otras definiciones alternativas de RV son:

- Un entorno en tres dimensiones sintetizado por computadora en el que varios participantes acoplados de forma adecuada pueden atraer y manipular elementos físicos simulados en el entorno y, de alguna manera, relacionarse con las representaciones de otras personas pasadas, presentes o ficticias o con criaturas inventadas.

- Un sistema interactivo computarizado tan rápido e intuitivo que la computadora desaparece. La Realidad Virtual, o técnicas del ambiente virtual, cambiará la manera en la que el hombre actúa recíprocamente con los sistemas de la computadora.

- O simplemente entenderlo como un mundo de animación donde nosotros creamos y formamos parte de él. Las ideas subyacentes de realidad virtual emergieron casi hace treinta años cuando investigadores empezaron a sumergir participantes humanos (operadores) en sistemas de teleoperación. Un sistema de teleoperación es aquel que es operado y controlado por un operador humano que está en una situación remota de cualquier ambiente. En el cual lleva un aparato en forma de visor montado en la cabeza y la cual despliega imágenes, y que se mueve en la dirección a la que dirija su cabeza.

Aunque la ciencia del último siglo, con la física, se ha dedicado con especial empeño a cuestionar las más antiguas y firmemente asentadas dicotomías: espacio y tiempo, materia y energía, movimiento y reposo, cuerpo y mente y ahora la ciencia y la tecnología, con la informática parecen dispuestas a cruzar la última frontera, la que separa lo irreal de lo real.

Con el teléfono y la radio, el oído supera la distancia y suplente la voz real del hablante con una reproducción electrónica convincente. Con la televisión, el ojo hace lo mismo con la imagen. Si podemos ampliar la ilusión a los demás sentidos (sobre todo al tacto), la "telepresencia" será un hecho.

Al igual que hay sistemas acústicos de alta fidelidad tan perfectos que hacen imposible distinguir una grabación, por ejemplo, una orquesta real, es sólo cuestión de perfeccionamiento técnico el crear ilusiones multisensoriales prácticamente indistinguibles de una experiencia auténtica.

Desde nuevas y revolucionarias formas de enseñanza o de rehabilitación de minusválidos, hasta nuevas y tal vez definitivas modalidades de drogadicción electrónica. Desde el sexo a distancia (o con seres creados por computadora) hasta la guerra teledirigida, siguiendo el refrán de que en el amor y la guerra todo se vale. Junto con la inteligencia artificial, la realidad virtual es sin duda el campo de investigación más importante del momento, el más prometedor y el más inquietante, el *mysterium tremendum* que el hombre tendrá que afrontar.

Lo que la humanidad puede soñar, la tecnología lo puede conseguir" es la consigna de Fujitsu (una de las empresas más seriamente comprometidas con el desarrollo de la realidad virtual), y parece a punto de cumplirse plenamente. Que sueños o pesadillas llegue a realizar la tecnología en los próximos años, depende en buena medida de que estamos informados sobre sus posibilidades y tendencias.

CAPTAR LA REALIDAD MEDIANTE LA ILUSIÓN

La computadora, con su capacidad para manejar enormes volúmenes de datos y simular la realidad, abre una nueva ventana sobre esa visión de la naturaleza. Tal vez comencemos a ver la realidad de manera diferente sólo porque la computadora produce conocimiento de modo diferente al de los tradicionales instrumentos analíticos. Brinda un diferente ángulo de la realidad. Vivimos en un mundo físico con cuyas propiedades nos hemos familiarizado llegando a conocerlas bien. Sentimos un compromiso con ese mundo físico que nos da la capacidad de predecir dónde caerán los objetos, qué aspecto tienen las formas bien conocidas vistas desde otros ángulos y cuánta fuerza se requiere para empujar los objetos venciendo la fricción.

Se tiene el conocimiento correspondiente de las fuerzas que actúan sobre las partículas cargadas, fuerzas en los campos no uniformes, los efectos de las transformaciones geométricas no proyectivas y el movimiento de baja fricción y alta inercia. Una pantalla conectada con una computadora digital nos brinda la ocasión de familiarizarnos con conceptos que no se pueden materializar en el mundo físico. Es un espejo que nos lleva al mundo maravilloso de la matemática.

Si la misión de la pantalla es servirnos de espejo para introducirnos en el mundo maravilloso de la matemática, construido en la memoria de una computadora, debería atender a tantos sentidos como sea posible. Hasta donde se sabe, nadie propuso seriamente displays de computadora olfativos o gustativos. Existen excelentes displays de audio, pero lamentablemente tenemos poca aptitud para hacer que la computadora genere sonidos significativos.

En Ames Research Center de la NASA en Mountain View, California. Se desarrolló la computadora que ellos llamaban "la máquina de la realidad" Una especie de máscara de aluminio semejante a la de un SCUBA cubre el rostro y una televisión binocular tridimensional llenaba el campo de visión con espejismos electrónicos, cualquiera que fuera la dirección en la que se moviera la cabeza.

El cuerpo no estaba en el mundo de la computadora en el que se podía ver alrededor de uno, pero una de las manos había acompañado a los ojos sobre la vasta llanura electrónica que parecía rodear. Un guante con un reticulado de sensores sincronizaba los gestos físicos en el cuarto en el que el cuerpo estaba ubicado, con los movimientos de el guante que flotaba en el mundo creado por la computadora "Ciberespacio".

Una representación gráfica de los datos extractados de los bancos de cada computadora en el sistema humano. Líneas de luz ordenadas en el no espacio de la mente, esta máquina es

una televisión envolvente con programas tridimensionales, incluso sonido tridimensional y objetos sólidos que puede alzar, manipular y hasta tocar con los dedos y la manos. Inmerso en un mundo artificial que explora activamente en lugar de mirarlo con atención desde una perspectiva fija, mediante una pantalla plana en un cinematógrafo o en un televisor o una computadora. Imagínese que es el creador tanto como el consumidor de su experiencia artificial, y que dispone del poder de usar un gesto o palabra para volver a moldear el mundo que esta viendo, oyendo y sintiendo. Esta parte no es ficción. Los displays de cabeza (HMDs) y los gráficos tridimensionales de la computadora, dispositivos de entrada y salida, modelos computacionales que constituyen un sistema RV, hacen posible hoy que usted se sumerja en un mundo artificial, lo alcance y lo reforme.

Se le a llegado a llamar "Simulador" también. La tecnología RV se parece a los simuladores de vuelo que la Fuerza Aérea y las aerolíneas comerciales usan para entrenar a los pilotos. En los simuladores de vuelos convencionales, los pilotos aprenden a manejar un avión sin abandonar el suelo, practicando con una réplica de los controles del aeroplano; el "parabrisas" de un simulador de vuelos es una pantalla indicadora con gráficos computacionales, sobre la cual se va presentando un paisaje cambiante de acuerdo con el curso que toma el piloto.

Toda la simulación está montada sobre una plataforma móvil que se desplaza de acuerdo con los movimientos del aeroplano simulado. La realidad virtual también es un simulador, pero en lugar de mirar una pantalla plana bidimensional y operar un joystick, la persona que experimenta la RV está rodeada por una representación tridimensional generada por computadora, y es capaz de desplazarse en el mundo virtual y verlo bajo distintos ángulos para introducirse en él, asirlo y remodelarlo. Aunque esto suena a ciencia ficción, y la palabra "ciberespacio" se originó de hecho en una novela de ciencia ficción, la realidad virtual ya es una ciencia, una tecnología y un negocio sostenido por fondos significativos que provienen de las industrias de computación, comunicaciones, diseño y entretenimiento en todo el mundo.

- En la Universidad de Carolina del Norte se puede caminar por un edificio que existió en el ciberespacio antes de que fuera construido en la realidad física.
- En la NASA existen robots para mantenimiento en un espacio exterior virtual.
- En Tsukuba, existe una de las originales "ciudades de la ciencia" en Japón uno puede tener la fantástica experiencia de observarse así mismo a través de los ojos de un telerobot: una experiencia " fuera del cuerpo" electrónicamente amplificada.
- En Silicon Valley, hay una empresa en donde se fabricaban gafas y guantes que se han usado en muchos otros lados.

CAPITULO II.

ELEMENTOS DE LA REALIDAD VIRTUAL.

Aunque no existen reglas fijas y específicas sobre lo que deben o no incorporar los sistemas de Realidad Virtual, los mejores sistemas utilizan tres elementos básicos: Inmersión, Navegación y Manipulación.

INMERSIÓN.

Sumergirse en un sistema de Realidad Virtual es sentir que se está experimentando desde dentro una Realidad Alternativa, no simplemente verla a través de una ventana.

Es como si estuviese dentro de un juego de vídeo, relacionándose con los otros personajes, que ahora tienen tamaño natural y que pueden aparecer por detrás a la izquierda o a la derecha, e incluso por encima de la cabeza (recuerde la película TRON, donde el protagonista principal se encuentra dentro del juego que ha programado luchando contra los otros protagonistas del juego.)

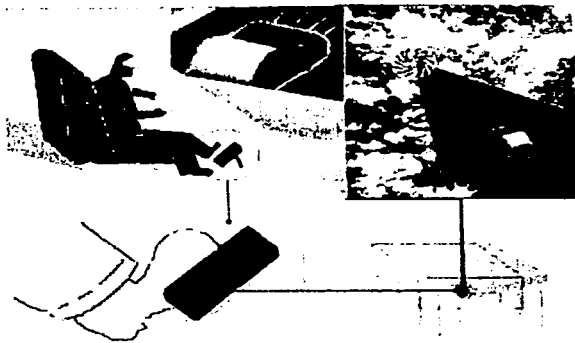
Cuando se acelera en este simulador de conducción, la pantalla incrementa la velocidad de presentación de los cuadros para dar la impresión de que se está moviendo más rápidamente.

NIVELES DE INMERSIÓN.

Muchos diferentes niveles y grados de inmersión sensorial están incluidos en el campo de la RV. Cada aplicación puede ser diseñada para acomodar diversos grados de creencia suspendida o de manipulación que le presente. Los espacios primitivos en 3-D disminuyen, así como las imágenes y objetos endoscópicos y borrosos.

Al ver un espacio en 3-D, una persona es expuesta a un nivel elemental de inmersión virtual, considerando el espacio, los objetos son definidos para la percepción del usuario y la manipulación de las imágenes es posible, el usuario queda expuesto a otro nivel de inmersión.

La inmersión "total", en cualquier caso, requiere que toda referencia al mundo real sea bloqueada de forma efectiva, proporcionar estímulos sustitutivos, y que el usuario éste convencido de que es real. Está es la inmersión a nivel del entorno. El potencial y la demanda de aplicaciones existe a todos éstos niveles, los cuales son espacio, objetos y entorno.

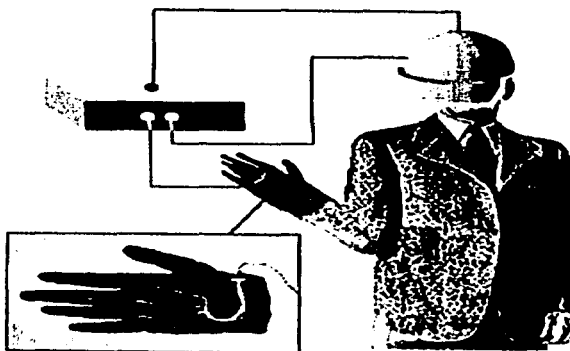


La inmersión es función específica del Hardware. Algunos sistemas de Realidad Virtual, como el aparato VIRTUALITY que funciona con la Pesadilla de los Dáctiles (juego de Vídeo), utilizan un casco con pantalla (Head-Mounted Display o HMD) en vez de una pantalla normal de computadora para así estimular el sentido de la vista. Mostrando a cada ojo una imagen imperceptiblemente distinta, la pantalla del casco aprovecha el fenómeno del paralaje biocular para crear el efecto tridimensional, añadiendo así más realismo al mundo virtual.

EL HMD también estimula el sentido del oído a través de auriculares estereofónicos, dándole pistas en relación a la fuente y dirección del sonido presentando una fracción de segundo antes a un oído que al otro. El casco también puede gestionar los movimientos de la cabeza, suministrando a la computadora un flujo de datos respecto a su posición y orientación en un espacio tridimensional.

La computadora, a su vez ajusta la pantalla del casco, produciendo un estímulo visual que esta sincronizado con los movimientos de la cabeza. Algunos sistemas de realidad virtual utilizan dispositivos de entrada adicionales, como guantes que perciben la orientación y posición de las manos. Estos guantes de datos, actúan de la misma forma que el casco con pantalla, suministrándole a la computadora más datos. La figura anterior muestra un sistema típico de realidad virtual en un ambiente de inmersión, podrá sentirse en una cabina y volar o conducir utilizando un volante y pedales.

Observar el mundo virtual a través de la pantalla del casco, mientras la cabina oscila o se inclina a tomar una curva mientras se recorre la pista imaginaria, se sentirá como si realmente estuviera volando o conduciendo. La máquina de realidad virtual se ha apoderado de sus sentidos y lo a separado de la realidad "verdadera".



Sistema típico de realidad virtual, con casco y guante de datos.

SENSORAMA.

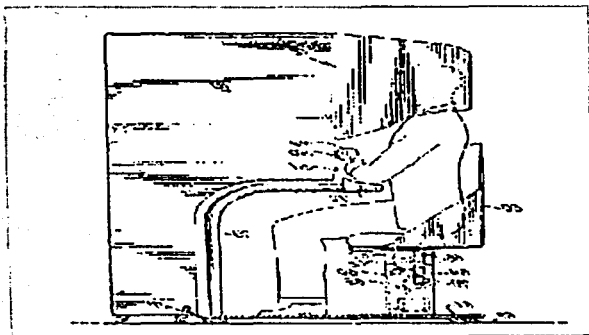
La máquina Sensorama de Morton Heilig, dispositivo multisensorial diseñado en 1962 para estimular los sentidos de la vista, tacto, oído y olfato del usuario, es un ejemplo de sistema de realidad virtual no computarizado que utiliza las técnicas de inmersión.

El usuario se sienta en un taburete frente a una pantalla de retroproyección y se asía a un par de palancas.

El sensorama conducía al usuario a través de varias escenas, incluyendo un paseo en bicicleta por Brooklyn y sobre las dunas de California. Estimula el sentido del tacto a través de las vibraciones de las palancas y de un asiento que se movía básicamente a medida que la "bicicleta" se desplazaba.

Estimula los sentidos de la vista y del oído mostrándole al usuario una película con los sonidos sincronizados, y el sentido del olfato lanzando aromas a través de pequeños surtidores que apuntaban hacia la cara del usuario.

Aunque la experiencia del Sensorama se basaba en una tecnología anticuada, supo sumergir al usuario en un entorno virtual durante corto tiempo. El aparato de Heilig no se fabricó nunca en cantidad, y lo único que queda de él es el prototipo. Hoy en día, el primer sistema de realidad virtual no computarizado permanece abandonado. En la figura, El Sensorama de Morton Heilig llevaba al usuario a un "paseo" a través de distintos lugares.



La máquina Sensorama de Morton Heilig

VIAJE A LAS ESTRELLAS.

Viaje a las estrellas, una de las atracciones de Disneylandia proporciona un ejemplo actual de un paso hacia la inmersión.

El público del viaje a las estrellas experimenta un recorrido escalofriante por el espacio mientras un robot piloto evita las colisiones y los desastres. Aunque las sensaciones de las violentas aceleraciones y desaceleraciones son asombrosamente reales, la cápsula del viaje a las estrellas nunca va a ninguna parte, ya que todo el viaje tiene lugar sobre suelo.

Las sensaciones de movimiento y gravedad son creadas por medio de las rápidas oscilaciones longitudinales y laterales del asiento y de la propia cápsula, mientras que el sonido que crea a través de canales estereofónicos múltiples añaden un toque más de realismo.

La inmersión es la diferencia entre mirar una realidad alternativa a través de una ventana (la realidad alternativa tenga la fama de un libro, una película o un vídeo juego) y mirar dicha realidad desde dentro, como un participante. La inmersión es el elemento unitario más importante de un sistema de realidad virtual, y es lo que hace "virtual" a la realidad.

NAVEGACIÓN.

La inmersión lo lleva a engaño y lo hace pensar que se encuentra en una realidad alternativa, mientras que la navegación le da la oportunidad de explorarla. La navegación es la habilidad de desplazarse dentro del ciberespacio generado por la computadora, explorarlo e interactuar con él a voluntad. Naturalmente, esto no significa que realmente vaya a una parte; es la sensación de que puede moverse dentro, lo que hace que un entorno sea "virtual".

Un entorno virtual es como un sueño: Puede estar en una cama en un mundo real, y sin embargo explorar reinos mágicos, o viajar en vuelos intercontinentales, en un mundo virtual.

Aunque el Sensorama de Heilig supo sumergir al usuario en un mundo virtual, la experiencia fue sin embargo, pasiva, ya que el usuario era sólo un pasajero en el viaje. No queremos de ninguna manera minimizar la importancia de la máquina (o de su inventor). La tecnología necesaria para crear un sistema de realidad virtual, navegable no existía en 1962, y no estaría disponible por lo menos durante otros 25 años.

Para incluir el control de la navegación en el sensorama usando la tecnología de 1962 se hubieran necesitado diferentes películas para cada intersección de carreteras durante el recorrido, y alguna forma de poder cambiar rápidamente de una película a otra, una tarea que aún hoy día sería un gran reto.

Los simuladores de vuelo y conducción de tipo industrial frecuentemente proporcionan tanto una etapa como muchas en un mundo virtual que es tan real como el mundo verdadero. Por ejemplo un simulador de conducción de camiones, es tan real que a un operador que se le indica que saque el camión de la carretera, normalmente es incapaz de hacerlo. Esto es Realidad Virtual.

MANIPULACIÓN.

La tercera variable que da a la realidad virtual sus ribetes de realidad es la habilidad del usuario para manipular de alguna forma el entorno. La manipulación es simplemente la posibilidad de alcanzar y llamar una puerta virtual, o disparar sobre un adversario virtual, y hacer que el mundo virtual responda de la manera apropiada. Por ejemplo, podrá ponerse un guante de datos, asir y rotar un objeto dentro del campo de visión virtual. El objeto responderá como si fuese un objeto real que gira en un mundo real. lo bueno de esto es que "SENTIRA" como hacer rotar el objeto, por medio de pequeños estimuladores táctiles (llamados Tactors) incorporados al guante.

CAPITULO III.

CIBERESPACIO.

HISTORIA Y GENERALIDADES.

El CIBERESPACIO es una tecnología de propósitos generales de interacción con las computadoras; nada en él es específico del diseño gráfico tridimensional, como tampoco las interfaces de quinta generación, con pantallas de trama, son sólo útiles para dibujos bidimensionales. Sin embargo, cuando los displays gráficos se desarrollaron por primera vez, se les usó para aplicaciones netamente gráficas tales como el dibujo o el procesamiento de la imagen. Sólo posteriormente, a medida que la tecnología de la representación gráfica se abarataba y los displays gráficos pudieron obtenerse fácilmente, el público llegó a ver que el uso adecuado de los gráficos bidimensionales podía ayudar a clarificar incluso tareas exclusivamente orientadas a textos o números.

Así ocurrirá con el ciberespacio. El ciberespacio representa la primera interfaz tridimensional de computadora. Los usuarios que luchan para comprender los diseños tridimensionales con vistas múltiples, imágenes sombreadas o animación, no tendrán dificultad en comprender, ni vacilación en adoptar, una tecnología que les permita recoger una parte y hacerla girar para comprender su forma, volar como Superman a través de un dibujo complejo o formar piezas usando herramientas y ver enseguida los resultados.

La VPL (Virtual Programming Language (Laboratory, Inc.) hizo posible la investigación de la RV para muchas personas en la comunidad informática, pero muchos otros, fuera de los círculos esotéricos dedicados a la investigación de la interfaz humana, empezaron a tomarla en cuenta cuando Autodesk ingresó en el negocio del ciberespacio, cuando exhibió la primera presentación interna de su proyecto Siberia en la primavera de 1989.

La sala estaba más que nunca repleta con el personal de Autodesk, sus invitados eran observadores de tecnología independiente, como Stewart Brand, el editor de Whole Earth Catalog, Hugh Daniels, el primer director técnico de la comunidad virtual de Whole Earth, WELL, el futurista Peter Schwartz, ex proyectista estratégico de la SRI International Royal Dutch/Shell, y el London International Stock Exchange, presidente ahora de su

propia firma consultora, Global Business Network. Theodor Nelson, tecnoprofeta, miembro de Autodesk y ejecutor de la revolución de la computadora personal, Esther Dyson, tábano de la industria informática, John Barlow, compositor de canciones para los Grateful Dead, republicano de Wyoming y que se autodenominaba "maniático tecnólogo", también gente de Apple. La mayoría de los presentes ya habían echado una mirada en el ciberespacio considerando el refinamiento técnico en el cual no había necesidad de explicar ya que ellos habían conocido la etapa preliminar del ciberespacio, en el que la base era el hardware.

Se hizo mención de futuras aplicaciones en visualización, juegos, salud, deportes, educación, entrenamiento, teatro, danzas, fisioterapia y telerobótica.

La entrada de Autodesk en el campo de la investigación y desarrollo del ciberespacio marcó como una industria seria aunque naciente. John Walker impone respeto entre los programadores, no sólo por su peso financiero, sino porque Autodesk se conoce como una empresa de programadores. Nació porque Walker y otros apostaron a que los arquitectos y diseñadores que habían estado usando computadoras personales para otras aplicaciones pagarían varios cientos de dólares por una versión algo menos poderosa de un software de CAD de varios miles de dólares que grandes estudios de arquitectura y firmas constructoras usaban en las computadoras grandes. El AutoCAD se convirtió en un Best Seller de software para computadoras personales.

A medida que maduraban las técnicas gráficas de computadora y crecía el potencia del hardware, el juego de herramientas (TOOL KIT) de Autodesk empezó a agrandarse, evolucionando desde una versión de pantalla plana de un dibujo mecánico hasta representaciones en 2 1/2 D de espacios y objetos.

Desde que la computadora puede crear espacios y paisajes tridimensionales, los hombres sueñan con entrar en este mundo de ilusiones y moverse libremente dentro de él. Los nuevos desarrollos en el entorno de la RV hacen que este sueño esté muy cercano a convertirse en realidad.

El usuario se coloca en la cabeza unas gafas negras de plástico con dos pequeñas pantallas incorporadas. Su mano derecha se desliza en el interior de un guante de licra dotado de sensores y cables de fibra de vidrio. Sobre sus hombros ondean dos cuerdas del tamaño de

un dedo. Su equipo es completo. A través del EyePhone y del guante de datos esta conectado a un sistema de computadora.

La instalación se pone en marcha. Se oye el murmullo de los aparatos al funcionar y al principio, el usuario ve todo de color negro. Después de 30 segundos, sus ojos perciben la

luz y puede distinguir una imagen. Se sumerge en el cielo azul a través de un extenso paisaje lleno de colinas.

Si mueve la cabeza, su campo de visión cambia totalmente. Justo enfrente de él flota una mano sintética de color gris que imita todos los gestos de su mano con un mínimo impar de tiempo. Sin embargo, no está mirando una ventana con varias pantallas desde fuera, sino que es el usuario el que está dentro de este mundo, formado por cortinas de luces de colores que se desarrolla en el espacio y donde no existe la gravedad. Un mundo que sólo obedece las leyes de la computadora.

Mientras tanto a sus espaldas se encuentra en la calle donde esta su casa, señala la puerta de entrada con el guante de datos e inmediatamente se dirige a ella volando. Transmite una orden errónea y, de repente, esta en el interior de la casa a través de la pared exterior. Ahora abre su mano derecha y aterriza en el acto. La computadora sabe que cuando extiende su dedo índice significa "volar"; una mano abierta, "parada", y el puño, "coger".

Esta en un enorme pasillo sin amueblar que termina en un gran salón con chimenea y amplios ventanales. Se acerca al único mueble que hay, una silla giratoria de color rojo, la levanta y la vuelve a colocar en una esquina de la habitación.

Ahora quiere volar otra vez para conseguirlo extiende su dedo índice verticalmente hacia arriba, inmediatamente comienza a elevarse, atravesando el techo de la habitación y el ático sale al exterior, abre su mano para no perderse en la inmensidad del espacio artificial, esta suspendido en el aire por encima de la casa y, al mirar hacia abajo, ve a lo lejos el balcón. Por un momento tiene miedo de caerse, la ilusión de la altura es tan fuerte que incluso le produce vértigo. En realidad esta sentado, con los dos pies apoyados en el suelo, en la planta baja de un edificio de oficinas de la Compañía de Software Autodesk, donde desde el laboratorio "Cyberia II" entro a un alucinante viaje electrónico y sus sentidos se han dejado engañar por la simulación de este mundo, que sólo existe como conjunto de datos en las memorias de la computadora.

Autodesk proyecta lanzar al mercado un sistema con el que cualquier arquitecto podrá crear directamente un modelo a partir de los datos de una casa construida por la computadora. De esta forma se evitarían muchos mal entendidos entre arquitectos y constructores, sobre todo en el caso de grandes complejos de viviendas y oficinas. En este

sentido, los técnicos de la firma van a iniciar la fabricación de unas gafas con pantallas, para lo que deben examinar la casa proyectada en el denominado ciberespacio.

Ciberespacio, un neologismo derivado de "espacio cibernético", fue creado por el escritor de ciencia ficción William Gibson. En su novela "New Romancer", publicado en 1984,

miles de millones de seres humanos viajan diariamente a un "espacio de alucinaciones que experimentan de forma conjunta", pero no con la ayuda de unas gafas con pantallas sino mediante electrodos implantados en la cabeza. Es por ello que no resulta extraño que en California el proyecto Ciberespacio no consista simplemente en un técnica para poder pasear por los gráficos tridimensionales de una computadora.

Hace tiempo que para muchos amantes de la informática, Ciberespacio se ha convertido en una droga digital que amplía notablemente los conocimientos. La ideología de esta nueva subcultura de alta tecnología se difunde en prestigiosas revistas como "Mundo 2000" o "Reality Hackers", en todos ellos se fantasea sobre la colonización de un continente fantástico que sólo existe en la computadora; sobre "realidad artificial", es decir, que engañan a la vista y al oído e incluso al sentido del tacto y sobre "teleasistencias" (también conocido como telepresencia), que no es otra cosa que la presencia aparente en lugares muy lejanos.

En Autodesk, donde un cartel colocado en el pasillo anuncia "CYBERSEX" con una mujer informatizada sintética, el experto en robótica Randy Walser dirige experimentos de Ciberespacio, además de la demostración arquitectónica, Walser y su grupo han desarrollado un juego de tenis, en el tenis dos jugadores conectados con cables entre sí y colocados en lugares totalmente distintos pueden ver en las pantallas de sus gafas el mismo campo sintético de tenis.

Con raquetas reales en sus manos cuyas posiciones pueden registrarse constantemente con el casco del guante de datos, los jugadores pegan y devuelven una pelota de tenis que no existe en la realidad. En la casa, con la bicicleta de Ciberespacio se tiene la sensación de pedalear a través de un paisaje, y si vamos muy de prisa podemos llegar incluso a volar "hablamos de un universo totalmente nuevo, los hombres entraran en el ciberespacio para trabajar, jugar, entrenar o bien para conversar un rato".

Hasta ahora solamente los institutos de investigación como consorcios japoneses de alta tecnología como Matsushita y Fujitsu, autoridades militares y las autoridades norteamericanas de aviación y astronáutica NASA, pueden permitirse el costoso equipo de Ciberespacio. En el Nasa Ames Research Center de Moffet Field en Silicon Valley, Scott Fisher, Michael McGreevy y Stephen Ellis experimentan desde mediados de los ochenta con "head-mounted displays" y "mundos virtuales".

La sensación espacial en el mundo artificial se refuerza enormemente cuando se oyen los pasos y las voces de otras personas que se mueven en la misma escena del ciberespacio. La psicóloga de la NASA Elizabeth Wenzel ha desarrollado un sistema tridimensional acústico con el que nos podemos orientar en el espacio exterior.

Si en el mundo ficticio una persona lanza una piedra a una ventana situada a 10 metros puede oír desde allí el ruido de los cristales al romperse. Si va a dar un paseo por el jardín del ciberespacio, escuchará según su localización el murmullo del agua de la fuente o el ruido de la calle. Del mismo modo, un controlador aéreo que trabaja con el sistema acústico tridimensional puede diferenciar más fácilmente los mensajes radiotelegráficos de los pilotos a través de sus auriculares porque escucha las voces en estéreo. Este sistema ayudará notablemente al controlador de radares y servirá para evitar colisiones.

La "falsa" realidad es aun más perfecta ya que, desde hace poco, en el ciberespacio también se puede sentir. En el Advanced Robotics Research Center en Manchester, el usuario se pone el guante "teletáctico", agarra un objeto ficticio y con la punta de sus dedos sentirá como lo toca. El truco consistía en que en este particular guante hay cosidas unas minúsculas almohadas neumáticas que, controladas por una computadora, se inflan rápidamente tan pronto como alguien que se encuentra en el ciberespacio toca la mano de otra persona o una pared.

Pese a todo, el único Artificial Reality System ofrecido hasta el momento pertenece a la empresa californiana VPL. El prototipo presentado de un sistema RB2 (Reality Built for 2) envuelve todo el cuerpo en un traje de datos equipados con sensores. De este modo, dos personas con el EyePhone y el traje de datos puestos reciben así dobles sintéticos que imitan en el acto todos sus movimientos.

Ellos mismos pueden determinar a voluntad la figura, el tamaño del cuerpo y el espectro de estas marionetas, de esta forma es posible que deambulen por una fábrica que existe como modelo de ciberespacio en una computadora de Nueva York. Durante esta vuelta de inspección en el laboratorio se encuentran en Sidney y Tokio. Naturalmente pueden conversar y ver los gestos y movimientos de la otra persona.

Para poder alterar hasta tal punto nuestra imagen habitual de la realidad es preciso que en cada uno de los tres lugares se encuentran computadoras de gran capacidad, encadenados entre sí a través de una conexión de banda ancha. A pesar de que la nueva tecnología está dando sus primeros pasos, los profetas de la cultura del ciberespacio predicán que, en el futuro, todo será posible. Célebres personajes como el fundador de VPL, Jaron Lanier un músico corpulento y amante de la informática, ha subrayado que "la RV nunca podrá alcanzar la belleza y la riqueza de detalles del mundo físico".

Los productos de Autodesk para la visualización y la manipulación tridimensional de los objetos y estructuras podían verse como la punta de lanza de algo tan importante al menos como la industria de las computadoras personales que sólo había alcanzado hasta ahora una pequeña porción de su mercado potencial. Walker expuso la tesis de que "la fascinación de hoy por interfases de usuario es un artilugio del modo en que operamos normalmente las

computadoras con pantallas, teclados e indicadores, del mismo modo que los lenguajes de control surgían de los sistemas Batch de tarjetas perforadas. Los próximos desarrollos tecnológicos prometen reemplazar las interfases actuales de usuario con algo muy diferente". Ese "algo muy diferente" es el display de inmersión total que Walker prefiere llamar sistema Ciberespacial.

Walker reconoció que la idea de entrar en el mundo de la computadora tiene sus orígenes en la ciencia ficción y citó a William Gibson y a Frederick Pohl de la serie "Heechee", "el segundo de los cuales, Beyond the Blue Event Horizon, inspiró a Autodesk y cuya idea desempeñó una parte significativa en la formación de una empresa". Walker adujo que "Realidad Artificial" y "Realidad Virtual" son oximorones y propuso como término mejor Ciberespacio.

Si la tecnología ciberespacial es la dirección que emprenderá la próxima revolución de las computadoras, y los usuarios de CAD son la vanguardia del mercado del Ciberespacio, y si Autodesk y su base para usuarios están en una perfecta situación para explorar las oportunidades del mercado, primero debía bajarse el costo del Sistema Ciberespacial, así como el costo de hardware, guantes, gafas y sensores de posición.

El Ciberespacio no solamente proveerá nuevas experiencias como paseos en una feria de diversiones. Más que cualquier mecanismo inventado hasta ahora, cambiará el modo en que se perciben los humanos, en un nivel muy fundamental y personal. En el Ciberespacio, no hay necesidad de moverse en un cuerpo como el que uno posee en la realidad física.

"Imagínese una fiesta de disfraces en la cual usted adopta no solamente un nuevo conjunto de vestimenta, sino un cuerpo nuevo, una nueva voz y en cierto sentido fundamental y literal una nueva identidad. Ahora imagínese que usted lo hace no sólo en una fiesta, sino todos los días como parte integrante de su vida. ¿Quién es usted entonces? parecería, desde su actual punto de vista en la realidad física, que usted estará centrado como está ahora, en su cuerpo físico.

¿Ocurrirá eso cuando las consecuencias de sus acciones y decisiones en sus posibles personalidades tengan consecuencias físicas, sociales, económicas, artísticas, técnicas y éticas tan significativas como las de su personalidad "original"? ¿La personalidad alternativa, que sólo actúa en el ciberespacio, consistirá legalmente una persona"?.

CAPITULO IV.

REQUERIMIENTOS PARA CREAR UN MUNDO VIRTUAL.

ENTRADA Y SALIDA.

Un sistema eficiente de RV necesita una correspondencia entre la Entrada (los datos suministrados a la computadora) y la Salida (los datos que salen). En la mayoría de los casos, el tipo de datos que alimenta un sistema de RV consiste en señales generadas por algún tipo de dispositivo de entrada, como un guante, un casco virtual o un volante y pedales.

Podemos distinguir entre este tipo de dispositivo de entrada, que tienen cierta relación con el mundo virtual que se explora y otros dispositivos de entrada que no la tienen tales como teclados o ratones. Por ejemplo, si el mundo virtual es de los que se manipulan por el usuario manualmente, haciendo y moviendo objetos, entonces el guante de datos es el dispositivo más apropiado. En este caso, el teclado será el menos apropiado, ya que no ayudaría a mantener la ilusión del entorno virtual. De manera similar, si el mundo virtual es del tipo que puede ser explorado por el usuario mediante el movimiento, entonces lo más apropiado será un suelo sensible a la presión.

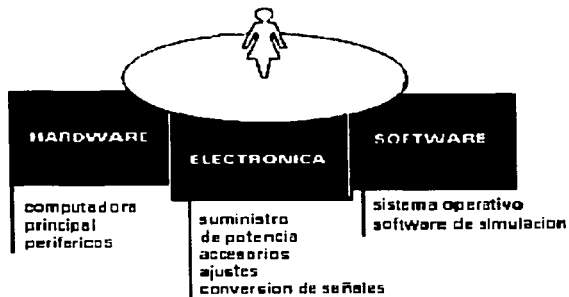


figura 1

También podría ser de gran ayuda el saber cómo algunos de los componentes básicos de la RV se introducen en un sistema. Un buen número de sistemas independientes se unen con la base de hardware, software y electrónica figura 1, y otros proporcionan efectos auditivos, visuales y táctiles.

DISPOSITIVOS ESTÁNDAR DE ENTRADA/SALIDA (E/S).

Los dispositivos de entrada son piezas del equipamiento usadas para introducir información. En un principio los procedimientos de entrada consistían en tarjetas que llevaban una serie de agujeros, que tenían significado para una máquina binaria. Después, y hasta hoy en día, el dispositivo de entrada es el teclado.

Otros dispositivos por citar algunos son:

- Un ratón (Mouse)
- Pantallas sensibles al tacto
- Un bolígrafo o puntero
- Botones y palancas
- Escáneres
- Sintetizadores de voz
- Indicadores situados en la cabeza
- Biocontroladores

Un dispositivo de salida es cualquier cosa mediante la cual los datos son recibidos por medio de la computadora. Los primeros dispositivos de salida fueron muy parecidos a los teletipos; a partir de información que puede ser leída por la computadora, producían paquetes de datos que los humanos podían entender. Después vinieron las impresoras, las pantallas de vídeo, las voces sintetizadas y otros dispositivos que traducían información legible por la máquina y la sacaban al mundo exterior. Algunos dispositivos llamados canales de entrada/salida E/S tienen un doble objetivo: un tráfico en dos direcciones, por ejemplo un módem telefónico es un dispositivo de E/S.

A continuación se dan algunos ejemplos de los dispositivos de entrada utilizados en un sistema típico de RV. No tiene por qué tenerlos todos pero mientras mas tenga, el efecto será más realista.

GUANTES DE DATOS.

Probablemente el dispositivo más usual para el control y entrada en un sistema virtual es un guante, "instrumentado" con fibras ópticas flexibles que recorren cada una de las articulaciones de la mano.

Este dispositivo consiste en un guante que emplea técnicas electrónicas para obtener la posición y orientación de la mano que lo lleva. Mientras la mano se mueve en tres dimensiones, el guante envía una corriente de datos electrónicos a la computadora en forma de coordenadas tridimensionales.

La computadora utiliza entonces los datos para manipular un objeto en la pantalla, por ejemplo, puede mover su mano como si rotara un cubo en el espacio. La computadora utiliza los datos generados por el guante para rotar un cubo en la pantalla, en perfecta sincronización con los movimientos de la mano. En la figura 2, el efecto puede ser bastante real, especialmente si utiliza gafas 3D y la pantalla de salida utiliza técnicas de imágenes dobles en 3D. Al Girar el guante de datos la imagen de la pantalla hará lo mismo.

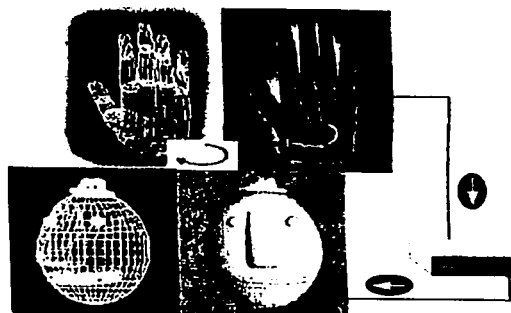


figura 2

Aunque se han utilizado distintas tecnologías en los guantes para obtener la sensación de flexión y movimiento, los más elegantes (por ejemplo los VPL DataGlove) utilizan fibras ópticas. Con esta técnica, el guante incorpora una red de fibras ópticas (de las que se utilizan en las redes telefónicas) incorporadas a lo largo de los dedos. La técnica implica una luz de intensidad que ilumina un extremo de la red de fibras, y luego medir la intensidad de la luz que emerge por el otro extremo. Algunas de las fibras están integradas en las articulaciones de los dedos y los nudillos, los que ocasionan una pérdida de luz cuando se doblan los dedos. La luz que emerge por el extremo de las fibras integradas se compara con la luz que sale de las fibras no integradas ("grupo de control"). En la figura 3, la computadora utiliza estos datos para determinar qué articulaciones fueron flexionadas y

hasta qué punto, y luego ajusta la imagen en concordancia. Así funciona el VPL DataGlove. Las fibras ópticas determinan el grado de flexión y arqueo de los dedos.



figura 3

Otra técnica, ésta utilizada por el Dexterous Hand Master, de Exos, implica un intrincado exoesqueleto de imanes y sensores que miden el ángulo de flexión de cada articulación de la mano. Hay una serie de imanes y sensores conectados a cada dedo con cinta velcro, haciendo que el conjunto parezca un dispositivo robótico extraterrestre como se muestra en la figura 4. El exoesqueleto de Exos mide la flexión de los dedos por medio de imanes y sensores.

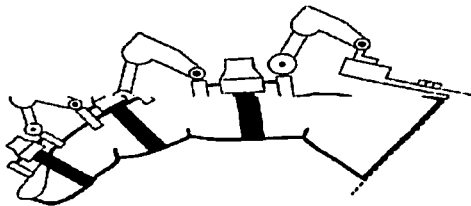


figura 4

La opción entre los VPL DataGlove y el Exos Hand Master es un problema de la actividad que se va a realizar el DataGlove pone énfasis en la comodidad y el estilo (no existen sensores salientes). Por otro lado el Hand Master, pone su énfasis en la exactitud junto con ciertas condiciones de comodidad.

Otros tipos elásticos menos incómodos están siendo desarrollados, quizás entre los modelos más modernos están los guantes sin dedos que permiten al usuario manipular las cosas en el entorno real (como el teclado) mientras están conectados al entorno virtual. Los guantes sin dedos figura 5 son también más cómodos porque las áreas de los dedos y las palmas están expuestas al aire y ventiladas.



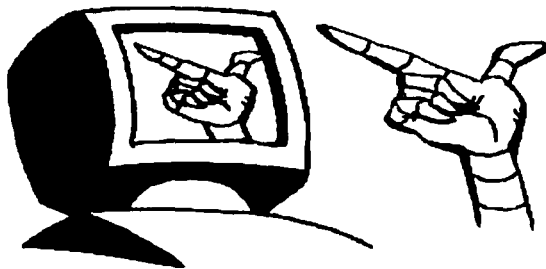
figura 5

Al introducir al sistema un lenguaje de signos, los movimientos particulares del usuario (como apuntar) o los conjuntos de movimientos combinados (como algunos gestos) sirven para dar mandatos al programa. El programa interpreta éstos y genera imágenes que satisfacen estos signos. Por ejemplo, los dedos cerrados en el puño, al ser elevados en la dirección de un objeto, podrían representar una orden para dar un golpe de zoom en el objeto. Este sistema generara entonces una secuencia de imágenes que da a la persona la sensación de acercamiento al objeto.

Los guantes son utilizados primordialmente para transmitir al sistema señales en una dirección; en cualquier caso, algunos están adaptados también para transmitir señales en la otra dirección del sistema e incrementar la credibilidad del entorno virtual. Esta realimentación puede incluir sensaciones táctiles como presión, calor o textura, para elevar lo que el usuario experimenta. Para una realimentación táctil, los objetos reales son abarcados con un guante de desarrollo. Una muestra de fuerzas es medida y almacenada y es evocada más adelante para producir el mismo efecto en el guante del usuario.

De los veinte o más sensores que puede contener un guante, no todos necesitan ser activos. Quien use el sistema puede elegir utilizar sólo aquellos basados en información relevante

para su trabajo particular; conservando por tanto recursos del sistema. También; dependiendo de la naturaleza de la realimentación requerida, los datos pueden ser tomados en intervalos, en lugar de continuamente. La mayoría de los guantes requieren una graduación separada o específica para cada usuario y el rastreador electromagnético puede ser ajustado a las peculiaridades magnéticas de una habitación. Aunque la información para cada usuario puede ser almacenada en la computadora para otro uso, extraerla y reinstalarla cada vez puede suponer una incomodidad.



Las aplicaciones apropiadas de entrada por guante incluyen:

- Entornos de diseño CAD/CAM, de tal forma que el usuario pueda coger objetos en la pantalla;
- Entrenamiento biomédico y colaboración cuando, por ejemplo, los enfermeros necesitan asesoramiento sobre la función y ejecución de las manos;
- Robótica, para que los gestos puedan ser utilizados para dirigir a un robot;
- Telemanipulación, en la que las acciones de manipulado y presión de un robot puedan ser ejecutadas en tiempo real;
- Animación, para la cual un guante u otros sensores puedan ser transformados en caracteres generados por la computadora y controlar sus acciones;
- Investigación y enseñanza en el reconocimiento del lenguaje de signos y letras impresas;
- Simulación, en la que las señales del guante puedan ser incorporadas para controlar la información.

Generalmente, un guante permite al usuario trabajar con objetos virtuales de la misma manera que trabajaría con objetos reales, alcanzándolos, tocándolos, cogiéndolos o, de otra forma, manipulándolos por medio de la mano animada.

Un usuario puede seleccionar objetos y actuar en o con ellos de una forma muy natural, sin tener que recurrir a teclear en la computadora o a una interacción esotérica y formal con la computadora.

Una versión muy simple de estos guantes llamados "guantes de potencia" (PowerGlove) está en el mercado para ser usada en los videojuegos domésticos.

Muy representativos de los guantes más avanzados son los guantes provistos de resistencias sensibles a la fuerza en sus superficies internas.

Cuando los objetos reales son tomados, una muestra distinta de fuerzas es generada en la mano de una persona de la presión proporcional de estas fuerzas puede ser medida y almacenada como parte de la base de datos del entorno o en una base de datos creada específicamente con este propósito.

POSICIÓN ABSOLUTA.

El uso de medidores de tensión o fibra óptica sirve sólo para determinar la posición y orientación de la mano, pero no su posición absoluta en el espacio tridimensional. Para determinar la posición absoluta se hace necesario el uso de técnicas adicionales.

El PowerGlove emite una señal de ultrasonidos que es recogida por un receptor del monitor. En realidad son tres receptores, que utilizan la triangulación para determinar con exactitud la posición del guante.

Además de la posición absoluta, los receptores también pueden medir el grado de giro e inclinación del guante.

Tanto VPL como Exos utilizan la tecnología de localización, Polhemus para sus guantes de alta precisión. Ello hace necesario el uso de campos magnéticos y de sensores que pueden detectar la posición del guante dentro del campo magnético.

Este es el mismo procedimiento utilizado por Ascensión, otro fabricante de tecnología de detección de posición.

El PowerGlove de Mattel utiliza la técnica de triangulación para determinar con exactitud la posición del guante en el espacio tridimensional figura 6.

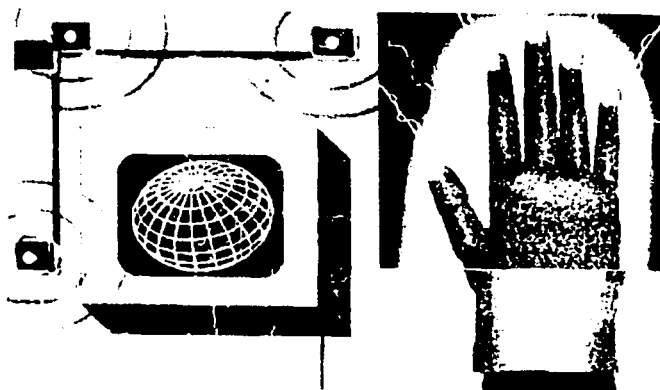


figura 6

DISPOSITIVOS DE VISUALIZACION EN CASCOS.

Los ángulos desde los que recibimos entradas por nuestros sentidos son: detrás a los lados o delante, influyen en nuestra percepción del mundo que nos rodea. Sintetizando todas las indicaciones de localización recibidas a través de nuestros canales sensoriales, determinamos dónde está algo en relación a nuestro cuerpo.

Un buen número de estos canales sensoriales (por ejemplo, los ojos para la visión, los oídos para el sonido, la nariz para el olfato y la boca para el gusto) están situados en la cabeza; por tanto, no es sorprendente que el equipo de cabeza sea normalmente la parte más importante del arsenal para un visitante de un entorno virtual. Es también uno de los más complejos.

Los equipos de cabeza son denominados unidades de presentación montados sobre la cabeza (HMDs Head-Mounted Displays). Algunos parecen cascos mediante los cuales los dispositivos de visión quedan suspendidos enfrente de los ojos del usuario figura 7. Otros parecen gafas de bucear sin tubo y los más simples son gafas suspendidas de una cinta en la cabeza.



figura 7

Los dispositivos más eficaces de entrada de datos para los sistemas de RV son, sin lugar a duda, los cascos. Estos dispositivos que gestionan tanto el aspecto de entrada como el de salida de la RV, se acopla a la cabeza del usuario. El HMD verifica los movimientos laterales y verticales de la cabeza, y envía el correspondiente flujo de datos sobre su posición a la computadora.

La computadora utiliza estos datos para crear imágenes estereoscópicas en el entorno virtual. Las imágenes estereoscópicas se envían a un par de pantallas miniaturizadas integradas en el HMD (aspecto de salida de la RV).

Como cada ojo recibe su propia imagen, y debido a que las imágenes difieren ligeramente, el efecto que producen es el de visualizar un mundo tridimensional.

En la figura 8 se muestra el casco fabricado por Virtual Research en Sunnyvale, California. Debido a que estas imágenes estereoscópicas están sincronizadas con los movimientos de la cabeza, el efecto producido es muy notable, dando la impresión de moverse realmente en una realidad alternativa.

Posteriormente veremos los aspectos estereoscópicos de esta técnica cuando estudiemos la visión estereoscópica y las gafas 3D (véase Efectos 3D, más adelante). Este casco fabricado por Virtual Research incluye pantallas gemelas y auriculares estereofónicos.

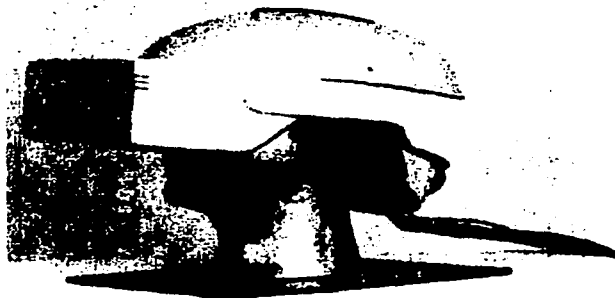


figura 8

Dos tipos de aparatos de visión son usados hoy en día para presentar realizaciones virtuales dispositivos de tubos de rayos catódicos (CRTs) y dispositivos de presentación en cristal líquido (LCDs). Los CRTs son básicamente pequeños televisores que presentan imágenes brillantes y de alta resolución figura 9, pero son incómodos sobre todo para llevar en la cabeza; suelen ser pesados, voluminosos y gruesos. Ya que los dispositivos de cristal líquido son ligeros y planos, como los que podemos encontrar en calculadoras y relojes digitales.

Los recientes logros tecnológicos han hecho posible pequeñas presentaciones en CRT de alta resolución y el LCD está siendo ahora reemplazado en muchos sistemas. Pequeñas pantallas de televisión de una pulgada pueden ahora presentar imágenes claras con una resolución de 1.000* 1.000 pixels.

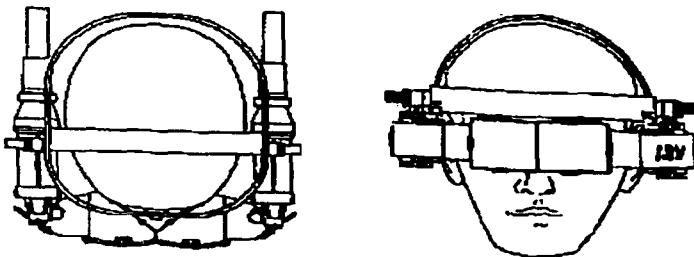


figura 9

Las presentaciones montadas sobre la cabeza son utilizadas en la reproducción de imágenes médicas, realizaciones moleculares, incursiones arquitectónicas y algunos videojuegos. Cuando el usuario se mueve, la escena cambia en la dirección opuesta y la persona siente como si estuviese en ella.

Las mejoras en los campos de visión son complicadas de especificar y desarrollar, normalmente, un observador no tiene una visión periférica. En HMDs de alta ejecución, los campos de visión no exceden los 70 grados por ojo. La meta es fundamentalmente mejorar los sistemas ópticos para que el usuario pueda obtener un campo de visión de 360 grados en cualquier momento.

POSICIÓN ABSOLUTA.

La mayoría de los HMD no sólo controlan la orientación de la cabeza sino también su posición absoluta. Con el añadido de este tipo de datos, el mundo virtual se convierte en uno bien real. Una manera de controlar la posición absoluta es a través del transmisor montado en el HMD y de un receptor instalado en o cerca de la computadora. A medida que el usuario se desplaza, la computadora recibe un flujo de datos, que utiliza para modificar el tamaño del objeto dentro del campo visual.

Estos datos, añadidos a los anteriores ayudan a crear una realidad artificial que aparenta ser de tanta profundidad y dinámica como el mundo real. Otros sistemas utilizan sensores magnéticos y campos magnéticos que rodean el casco. Mientras el usuario se desplaza dentro del campo magnético, el sensor establece su posición y la envía a la computadora. La VPL Research, una de las pioneras en guantes de datos, anuncio recientemente la creación de un sistema de RV completo denominado MicroCosm.

El MicroCosm se basa en el Macintosh Quadra 900 y esta integrado por un casco, uno o dos guantes DataGlove, y una "caja negra" que controla toda la entrada y salida.

RASTREADORES.

Para objetivos de rastreo, un dispositivo es sujetado al objeto o al usuario para que los movimientos de la cabeza o las manos puedan ser detectados. Esto es llevado a cabo con medios electromagnéticos u ópticos. Los movimientos son expresados en coordenadas de posición y orientación que son descifradas por la computadora. Es entonces, cuando las

imágenes correspondientes a ese punto de vista son presentadas. Los dos tipos de sensores de posición, electromagnéticos y ópticos, ejercen su presión sobre el usuario.

Normalmente alguien que está usando un dispositivo de rastreo electromagnético esta restringido a un área de trabajo de sólo 4 ó 5 pies. Los rastreadores de techo que usan diodos emisores de luz (LEDs) extienden el campo de trabajo considerablemente pero todavía no excede el tamaño de una habitación.

La mayoría de los sistemas de rastreo requieren una clara línea de visión en todo momento entre el sensor y el objetivo o el usuario. El estado de latencia es un factor en ambos tipos de rastreo, aunque el retraso es más problemático en sistemas electromagnéticos que en los ópticos.

CARACTERÍSTICAS DE RASTREO DE LOS EQUIPOS DE CABEZA.

Dentro de los sistemas electromagnéticos, el dispositivo más popular, el Polhemus, sigue los cambios de orientación o posición del usuario leyendo campos magnéticos alternativos.

Para rastrear se sitúa un pequeño sensor en lo alto de la unidad del equipo de cabeza figura 10. Los sensores pueden ser situados en cualquier sitio, en los guantes o dentro de dispositivos de bola.

Debido a la difusión de los campos magnéticos en los dos extremos hay una pérdida de una fracción de segundo entre el movimiento de la cabeza del usuario y su correspondiente respuesta desde el generador de presentaciones.

Esta diferencia puede ser disminuida. Al mismo tiempo que el usuario se mueve, los cálculos son hechos y las imágenes son constantemente regeneradas y actualizadas.

Estas imágenes reflejan cambios en la escena virtual, correspondiendo a lo que el usuario vería si realizase movimientos parecidos en un entorno real.

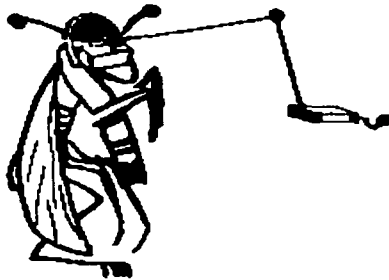


figura 10

En los sistemas ópticos, los paneles del techo llevan acoplados unos 1.000 haces infrarrojos LED que sirven como faros de navegación.

Como la posición de cada luz en el techo es conocida, la computadora conecta aquellos LEDs que se encuentran en el campo de visión de las cámaras situadas en el equipo de cabeza del usuario.

Los fotodiodos de efecto lateral hacen posible que las cámaras recojan información de los faros encendidos.

Entonces la computadora usa un algoritmo llamado <<recepción espacial por colinealidad>> para calcular la posición y orientación de la cabeza del usuario y generar una presentación apropiada.

Normalmente el número de actualizaciones varía según el número de LEDs visibles. El rastreo óptico figura 11 es usualmente posible con un hardware mínimo, pero la gestión de los elementos de la información recogida requiere una gran atención.

La posición exacta de los faros, el estacionamiento correcto de las cámaras y el calibrado de los fotodiodos para obtener información precisa son elementos cruciales para que la aplicación tenga éxito.

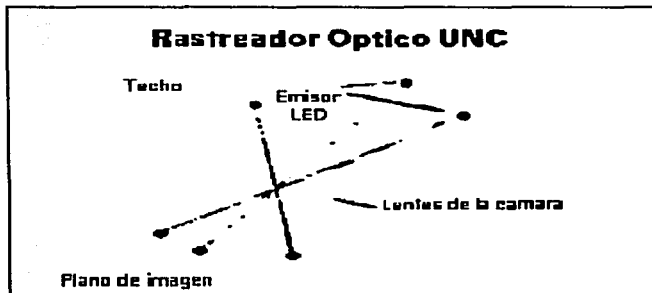


figura 11

INVESTIGACIÓN DE RASTREOS ALTERNATIVOS.

Los dispositivos ultrasónicos, giroscópicos y mecánicos son también usados en misiones de rastreo en aplicaciones de RV. Los ultrasónicos son relativamente baratos pero son muy sensibles a ruidos exteriores y requieren camino libre entre los emisores de señales y los micrófonos; los giroscópicos son precisos y compactos, pero son caros y con el tiempo se deterioran; y los mecánicos son sencillos y baratos pero físicamente limitados.

Una lanzadera podría poner en órbita muchos satélites, cada uno de los cuales emitiría señales de radio que una computadora o un receptor podría convertir en marcas de orientación. Si se usan dos satélites, sería posible un mapa de lectura de altitud. La única desventaja que tiene este tipo de rastreo es que debe haber una línea clara de visión entre cada satélite y el objetivo a rastrear. Además, es útil, primordialmente para las unidades exteriores como las implementadas por las compañías de camiones en vehículos de seguimiento y de ruta.

Un sistema de posicionamiento global denotado GPS (Por Global Positionig System) está siendo utilizado en lo militar y su rastreo mediante satélite es preciso hasta alrededor de 7 pies.

Las investigaciones sobre receptores de GPS se están realizando combinando sistemas. El autorrastreo, los rastreos por radar óptico y el láser óptico están siendo examinados, para las aplicaciones en las que el usuario no puede ser atado a un laboratorio, como las aplicaciones al aire libre, en grandes espacios o para su uso en presentaciones virtuales.

BOOMS.

BOOM son las iniciales de Monitor de Omni-Orientación Binocular (Binocular Omni-Orientation Monitor figura 12), un tipo de dispositivo de visión estereoscópica que se ha convertido en una alternativa popular a los dispositivos montados sobre la cabeza.

La caja de visión se suspende desde una estructura giratoria en un extremo, su peso es compensado por una carga en el otro extremo. Dos asas están situadas cerca de la base de la caja y son utilizadas para empujar y sujetar la caja cómodamente en la cabeza, dirigiéndola para obtener diferentes vistas. Los modelos actuales presentan escenas en monocromo y color, en un espacio de presentación cilíndrica de aproximadamente 5 pies de diámetro y 5 pies de altura.



figura 12

Su campo de visión, en cualquier instante, es aproximadamente 140 grados horizontal por 90 grados vertical. Su efecto estéreo está logrado básicamente mediante la reproducción de imágenes en tonalidad roja para el ojo izquierdo y en imágenes en tonalidad azul para el ojo derecho.

Cualquier cálculo de rastreo es una composición de las medidas tomadas simultáneamente de las tres diferentes posiciones y orientaciones. Los codificadores ópticos en cada uno de

los seis pliegues del BOOM alimentan al sistema con esta información de alta precisión sobre la posición y la orientación de la cabeza del usuario. El uso del BOOM tiene ciertas ventajas sobre el uso de un dispositivo montado sobre la cabeza, especialmente si va a ser utilizado durante un largo período de tiempo.

Además, como se mencionó anteriormente, la mayoría de las presentaciones montadas sobre la cabeza están equipadas con ligeros LCDs, y éstos sólo producen imágenes programadas y de baja resolución (720 * 480 pixels). El peso de un BOOM, en cualquier caso, no es llevado y conducido por el usuario, luego los BOOMS pueden utilizar la tecnología de presentación de los CRTs, pequeños y pesados televisores con una resolución de 1.280 por 1.024 pixels por ojo. Esto es suficiente para obtener imágenes nítidas y detalladas.

El BOOM puede ser suspendido sobre la estación de trabajo de una persona de tal manera que alguien tenga acceso a la escena virtual mientras está sentado en una terminal, pero puede ser retirado para realizar otro trabajo. Los dispositivos mecánicos de rastreo que llevan incorporados los BOOMS también eliminan la necesidad de un sistema de rastreo electromagnético, evitando los problemas de distorsión que a menudo están asociados a las interferencias magnéticas.

Los BOOMS tienen su mejor aplicación en visión directa o navegación simple y son particularmente apropiados en proyectos de visualizaciones y simulaciones científicas y para construir módulos de entrenamiento y entretenimiento.

SALIDA.

La forma más importante de salida desde un sistema de RV es la luz visible, bien sea desde un monitor normal o desde las pantallas binoculares instaladas en un casco. Resulta claro que cuanto mayor sea la variedad de salida, más real será el efecto obtenido, la mayoría de los sistemas de RV también utilizan el sonido y la presión sobre la piel, además de la luz, para incrementar la ilusión de un mundo alternativo.

LUZ.

Nuestros ojos nos proporcionan información en relación a la calidad y cantidad de luz en el entorno que nos rodea. Trabajando de forma simultánea, un par de ojos pueden suministrar

más información acerca del tamaño y la posición relativa de los objetos situados en nuestro campo visual que un par de ojos que trabajen independientemente. Cualquiera que haya estudiado un curso de introducción a la psicología sabe algo acerca de los impulsos que necesita el cerebro para determinar el tamaño y posición de los objetos en el espacio. Por ejemplo, dada una fuente luminosa situada a espaldas del observador, el cerebro interpretara que un objeto pobremente iluminado estará mas alejado del observador que uno que lo este intensamente figura 13.

De la misma forma, un objeto que aparece de menor tamaño se considerara más alejado que uno que aparece con un tamaño mayor, que es como se vería un objeto detrás del otro. Estas señales monoculares, que no dependen de un par de ojos trabajando al unísono, son bastante simples de programar en un sistema de RV.

De mayor interés para la RV son las señales visuales que dependen del trabajo simultáneo de ambos ojos. Estas señales binoculares son la base para la visión estereoscópica, que es a su vez uno de los elementos fundamentales del sistema de realidad visual que utilizan los cascos HMD. Señales monoculares de profundidad.

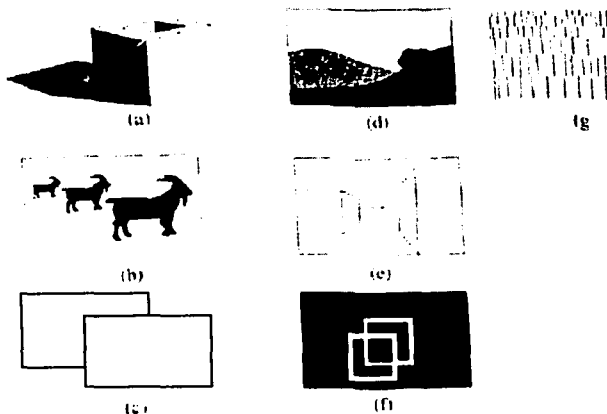


figura 13

a) Luz y Sombra

d) Textura Gradiante

g) En Profundidad

b) Tamaño Relativo

e) Perspectiva aérea

c) Interposición

f) Perspectiva

EFECTOS 3D.

Sucedan dos hechos de gran importancia cuando los ojos se enfocan en un objeto. Primero, los ojos convergen de manera que el objeto está en la línea de visión directa de cada ojo como se muestra en la figura 14. Esta convergencia binocular se produce a través de la acción de los músculos que controlan el movimiento del ojo, y suministra al cerebro la retroalimentación necesaria en relación a la distancia relativa con respecto al objeto, cuanto mayor sea la convergencia, más cercano estará el objeto.

El cerebro se apoya en la retroalimentación propioceptiva que envían estos músculos al intentar determinar la distancia relativa, mientras mayor es la tensión en estos músculos, mayor será la convergencia, y así sucesivamente (la propiocepción es simplemente la percepción por parte del cerebro del estado del cuerpo, por ejemplo su postura u orientación).

Segundo, cada ojo recibe una imagen del objeto ligeramente diferente de la otra. Así, el ojo izquierdo ve un poco más del lado izquierdo del objeto que el ojo derecho, que ve un poco más del lado derecho de dicho objeto. El cerebro utiliza el paralaje binocular para hacer una interpretación más exacta de la distancia relativa del objeto, cuanto mayor es la disparidad entre las dos imágenes, más cerca estará el objeto.

Estos dos factores combinados forman la base de la estereopsis, o sensación binocular de la profundidad. Repasemos en qué medida los experimentos más antiguos con la 3D han contribuido a la evolución de los actuales cascos HMD. Los ojos convergen para enfocar una imagen, cuanto más cerca se encuentre la imagen, mayor será la convergencia binocular.

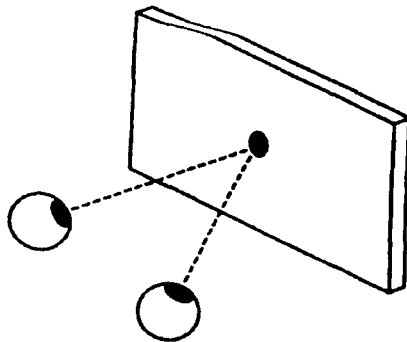


figura 14

DESDE LOS ESTEREOSCOPIOS HASTA LOS CASCOS HMD.

La historia de la tecnología que ha llevado hasta los cascos HMD se puede remontar directamente hasta Sir Charles Wheatstone y su estereoscopio de espejo de 1833. El aparato de Wheatstone semejaba unos binoculares con alas. Estas "alas" eran realmente los espejos que enviaban cada uno la respectiva imagen al ojo correspondiente.

A diferencia de los estereoscopios posteriores, que utilizaban un único cuadro con dos imágenes situadas una al lado de la otra, las imágenes de Wheatstone estaban sobre cuadros diferentes y colocadas a los lados y no de frente.

El aparato de Wheatstone, y los estereoscopios que le siguieron son excelentes para un solo observador. Se hizo por lo tanto necesario buscar un procedimiento diferente para proporcionar la sensación 3D a grandes grupos. El estereoscopio de espejos de Wheatstone montaba las imágenes de lado a lado figura 15.

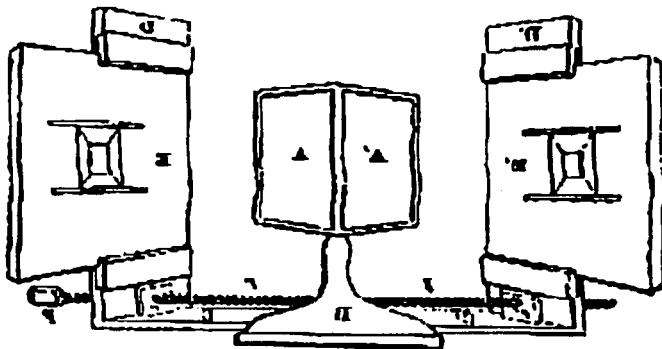


figura 15

ANAGLIFOS.

La técnica anaglífica fue demostrada en 1858 por un hombre llamado D'Almeida, que utilizó filtros rojos y verdes tanto en las lentes del proyector como en las gafas. La idea en la que se apoyaba su demostración era, primero proyectar la imagen derecha a través de un filtro verde y la izquierda a través de un filtro rojo, colocar delante del ojo izquierdo un filtro rojo para hacer invisible la imagen de la derecha a dicho ojo y un filtro verde delante del ojo derecho, como el filtro verde elimina la imagen verde y el filtro rojo elimina la roja, da como resultado que cada ojo recibe su propia y única imagen, el cerebro las funde para crear el efecto de profundidad figura 16.

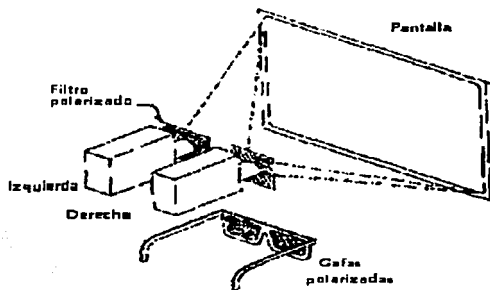


figura 16

PELÍCULAS EN 3D.

El sistema de filtros rojos y verdes fue el utilizado en Hollywood en 1937 para filmar la película con el apropiado título de Third-Dimension Murder (Asesinato en la Tercera Dimensión).

Aunque tuvo escaso éxito, sus limitaciones se hicieron evidentes a medida que Hollywood entraba en el mundo del color. (Como esta técnica trabajaba con filtros rojos y verdes para separar ambas imágenes, era imposible utilizarla en películas de color). Un avance sensacional se produjo en 1952 con el lanzamiento de la película Bwana Devil, la primera película en color y 3D.

Bwana Débil y las películas en 3D que les siguieron utilizaron lentes polarizadoras en los proyectores y en las gafas utilizadas por el público. La idea en que se apoyaba este sistema es idéntica a la de los filtros rojo y verde de D'Almeida, con la diferencia de que se utilizaba luz polarizada para separar las imágenes derecha e izquierda. La imagen derecha era polarizada en una dirección (vertical), mientras que la imagen izquierda era polarizada en la otra (horizontal). La lente izquierda transparentaba la luz polarizada verticalmente, pero no la luz polarizada horizontalmente, la otra lente hacía lo contrario muestra una técnica de proyección típica en 3D utilizando filtros polarizadores. La luz polarizada en una dirección se hace invisible al atravesar una lente polarizada en la dirección opuesta.

LAS COMPUTADORAS TAMBIÉN PUEDEN HACERLO.

La popularidad de las películas en 3D se difuminó con la misma celeridad con que apareció. Desde 1954 solamente unas cuantas películas han sido filmadas usando técnicas de 3D en comparación con las cientos que se filmaron dos años antes. Pero la pérdida de Hollywood se transformaría en una ganancia para Silicon Valley, al hacerse cargo de ello los creadores de software a partir de 1980. Hasta el momento, se han lanzado al mercado un cierto número de productos comerciales que utilizan tecnología en 3D. La mayor parte del software para 3D utiliza gafas con filtros rojos y verdes para crear los efectos de 3D, por lo tanto limita la salida a imágenes monocromáticas.

SISTEMA DE OBTURACION LCD DE SEGA.

Las gafas 3D suministradas para el Master Gamme System de Sega utilizan una técnica totalmente diferente para proyectar una imagen única para cada ojo. En vez de presentar simultáneamente la imagen apropiada para cada ojo, la técnica de Sega se basa en alternar la imagen de la pantalla entre la visión de la izquierda y la de la derecha sesenta veces por segundo, sincronizadas por medio de los "obturadores" incorporados a las gafas de LCD.

Cuando la imagen del ojo izquierdo es visible (durante un sesentavo de segundo), la lente derecha se hace opaca; y cuando la imagen del ojo derecho es visible, es la lente izquierda la que se hace opaca. El cerebro funde las imágenes de la izquierda y de la derecha para producir una visión estereoscópica. Si posee unas de estas gafas de LCD con obturadores, se puede construir un pequeño panel de circuito, o modificar el de Sega, que le permitirá utilizar estas gafas.

TECNOLOGÍA ESTEREOGRÁFICA DE PI-CELL.

Mientras que las gafas Sega están orientadas hacia un público ahorrativo, las que suministra la StereoGraphics Corporation de San Rafael California, están dirigidas a un público con poderes adquisitivos superiores. Las gafas estereográficas incorporan la tecnología punta de las pi-cell de LCD, capaces de transmitir y recibir ondas infrarrojas figura 17.

El transmisor infrarrojo se coloca encima del monitor y envía la señal que actúa sobre los obturadores, eliminando así la existencia de cables de conexión entre las gafas y el monitor. La figura de abajo muestra el sistema CrystalEye StereoGraphic.

El resultado es una visión estereoscópica a todo color y sin pestañeo, se use con un sistema cerrado de televisión o con imágenes generadas por una computadora. El efecto resulta muy natural, es decir, hasta que mueva la cabeza y cambie la perspectiva. El sistema StereoGraphic CrystalEye utiliza un emisor de rayos infrarrojos en el monitor (a) y gafas LCD (b) para crear el efecto 3D en la pantalla.

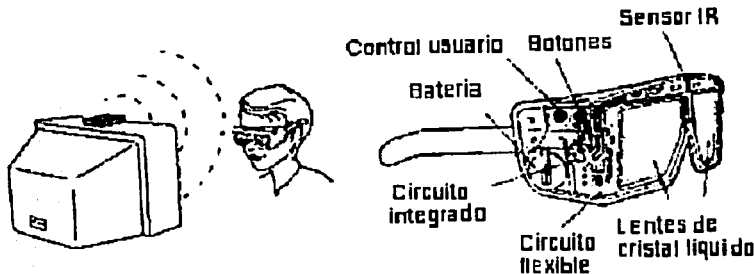


figura 17

EL MOVIMIENTO DE LA CABEZA ES UN HECHO DE LA VIDA REAL.

El problema con la mayor parte de la tecnología de 3D es la retroalimentación irreal e inadecuada que se recibe cuando se mueve la cabeza mientras se visualiza una imagen en 3D. Realice la siguiente experiencia mire a través de una ventana (o hacia un pasillo bastante largo) y mueva la cabeza a la izquierda y a la derecha. Notará que los objetos más cercanos se desplazan una mayor distancia que los que están más alejados, mientras que los más remotos prácticamente no se mueven.

Este fenómeno se conoce como "paralaje del movimiento", y fue utilizado por primera vez por Walt Disney en su clásico Blancanieves y los siete enanos para crear la sensación de profundidad al realizar los fondos de las tomas. (Antes de Blancanieves, los fondos en los dibujos animados eran simples telones planos que se desplazaban uniformemente, sin considerar la distancia aparente desde el observador).

El problema con las imágenes en 3D (sean creadas por computadora o no) es la ausencia del paralaje del movimiento; al mover la cabeza hacia los lados mientras ve una imagen en 3D no se produce ningún movimiento relativo de los objetos en la pantalla. Y tampoco da ninguna información adicional sobre la imagen, como serían los datos de los lados, alto y profundidad de los objetos.

En vez de ello, toda la imagen se mueve de una manera extraña. StereoGraphics incorpora un minúsculo ratón transmisor de ultrasonidos Logitech de 3D en un par de gafas StereoGraphics de LCD.

El receptor correspondiente, utilizando técnicas de triangulación, recibe los impulsos de sonido y calcula la posición del transmisor en el espacio en figura, al usarlo con las gafas estereográficas, el sistema de ratón 3D de Logitech sigue la pista de la posición absoluta de las gafas, y ajusta la imagen de la pantalla para que coincida.

Al mover la cabeza a la derecha se ve una parte mayor del lado derecho del objeto, mientras que si se mueve la cabeza hacia arriba se verá una porción mayor de la parte superior del objeto.

Uno de los programas de demostración en 3D, le permite rotar un objeto tridimensional en la pantalla mientras lo observa a través de las gafas 3D. El objetivo, es mover su cabeza a los lados mientras ve los distintos objetos, y notará que la retroalimentación no coincide con la realidad.

Ahora imagínese moviendo su cabeza hacia la derecha y viendo una porción mayor de ese lado del objeto y se hará una idea del sistema StereoGraphic. El receptor triangular localiza las gafas en un espacio de tres dimensiones para crear una retroalimentación real sobre la pantalla figura 18.

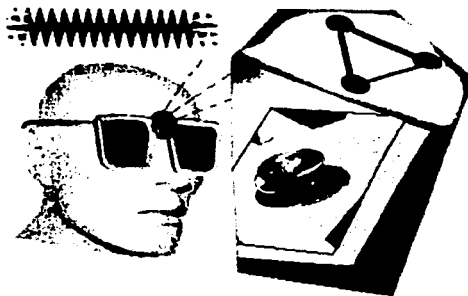


figura 18

SONIDO EN 3D.

Tal como descubrió la industria del sonido en la década de los 50, el mundo visual no era el único que existía en tres dimensiones. La gente tenía dos oídos por una razón: [dos oídos permiten que el cerebro localice el sonido en el espacio, tomando en cuenta que un sonido llega a un oído una fracción de segundo antes que al otro]. Estas mínimas diferencias entre los sonidos fueron las bases para el "nuevo" sonido estereofónico de dicha década, que se grababa usando dos micrófonos para simular los dos oídos del oyente. Por supuesto que el cerebro tiene un sentido adicional para la percepción de la profundidad auditiva, de manera que se puede distinguir, por ejemplo, el sonido que procede de arriba del que viene de abajo. Mientras que la industria del sonido prosigue en su camino para llegar a la meta de una reproducción fiel de la música (de aquí la "alta fidelidad"), la industria informática persigue su propia meta: creando o mejorando un mundo virtual a través del estímulo auditivo. Imagínese caminando dentro de un mundo virtual, escuchando sus propios pasos mientras resuenan en un pasillo muy largo. En algún lugar a la derecha oye una voz que murmura, "¡Shsss! Se detiene y gira para ver una figura inclinada que se esconde tras una puerta. Ella (pues es una mujer) desdobra un arrugado papel y se lo entrega.

La inclusión del sonido en un sistema de RV añade una dimensión real adicional al entorno, de la misma forma que el añadir el sonido sincronizado otorgó una nueva dimensión a las películas a partir de los años 30. Un dispositivo desarrollado por Crystal River Engineering y llamado Convolvotron utiliza 128 o más procesadores para producir un entorno auditivo tridimensional. Comercialmente, el uso del sonido en 3d está limitado a dos o cuatro canales, como lo hacen los aparatos Virtuality que están actualmente instaladas en los videojuegos.

FACILIDADES DE AUDIO DEL EQUIPO DE CABEZA.

El sonido es importante para la percepción espacial de una persona y es más efectivo cuando las ayudas visuales son mínimas. Los efectos de sonido con frecuencia reavivan los efectos visuales.

Hay muchos ejemplos en los que los diseñadores de RV incluyen características auditivas para aumentar lo visual, ya que todos tendemos a girarnos en la dirección en la que oímos algo para ayudar a nuestros ojos.

Los ejemplos más visibles son las aplicaciones en juegos y en entretenimientos, pero entre otras aplicaciones audio están también las presentaciones de torres de control o cabinas, la actividad telerobótica en entornos peligrosos, la exploración y observación de información multidimensional y algunos tipos de teleconferencias.

Los componentes auditivos que contienen un subsistema de procesamiento de señales en tiempo real sintetizan sonidos externos en 3-D. Estos recrean los efectos filtrantes que nosotros experimentamos por la forma externa irregular de nuestras orejas y proporcionan indicaciones direccionales.

Pero los humanos no somos precisos juzgando a qué distancia está el foco del sonido. Los investigadores tratan de producir focos sintetizados de sonido que no presenten ambigüedades.

Pequeñas sondas microfónicas son situadas cerca de los dos tímpanos de una persona colocada en una habitación sin eco. Se registran sonidos de unas 144 localizaciones. A partir de las diferentes formas de filtrar los sonidos se construye un mapa que contenga la localización de los filtros.

Dos alimentadores de sonido distintos son incorporados en los auriculares del equipo de cabeza, proporcionando al usuario sonidos simultáneos procedentes de al menos cuatro focos distintos que están sincronizados con el programa.

Los sonidos son generalmente seleccionados de una librería CD-ROM de efectos de sonido pregrabados generados por una computadora que se utiliza como un proveedor específico de sonidos digitales. Hay disponibles alrededor de 1.000 sonidos, como, por ejemplo, golpes secos cuando se produce algún contacto entre objetos virtuales, efectos de vuelo figura 19, etc.



figura 19

El propósito de los sonidos es elevar la ilusión de realidad o aumentar la información que se proporciona al usuario a través de otros canales. El usuario oye sonidos omnidireccionales que provienen de enfrente, detrás, ambos lados, arriba y abajo.

Cuando el usuario gira la cabeza, el sonido parece estar situado en el espacio virtual, y los ajustes son realizados para que la localización de "fuentes" particulares de sonido permanezca constante, independientemente de la orientación de la cabeza.

Por ejemplo, se pueden oír sonidos muy fuertes de pasos que se acercan por detrás; y si uno se da la vuelta, el sonido parecerá que se está acercando. Si se da un paso a un lado se podrá ver pasar al objeto acompañado de una ejecución en efecto Doppler del sonido de los pasos. Este tipo de sonido envolvente estereoscópico, a veces denominado sonido espacial, está siendo utilizado en muchos sistemas no virtuales, como los sistemas de navegación y rastreo para los pilotos.

Lo que se detecta en el radar se refuerza mediante indicaciones aurales direccionales, proporcionando al piloto más información para poder trabajar en situaciones potencialmente peligrosas.

TACTO Y PROPIOCEPCIÓN.

El sentido del tacto permite interactuar con y recoger información acerca del entorno sintiéndolo con sus dedos u otras partes del cuerpo. Imagínese su mano virtual penetrando en el ciberespacio y asiendo un objeto suspendido allí, o moviéndola y pulsando un transmisor, y sintiendo realmente el objeto en las puntas de sus dedos. Observará cómo su mano virtual pulsa el transmisor en la pantalla, y sentirá la presión en su dedo índice mientras lo hace, es la propiocepción la retroalimentación que suministra nuestro cuerpo acerca de su estado, postura y posición en un momento dado.

Este sentido propioceptivo es tan importante para los sistemas de RV como el sentido del tacto, ya que una parte importante de la ilusión estriba en la percepción del usuario de su relación con el mundo virtual. Para un usuario con los ojos vendados, o para otro que utiliza un casco HMD, existe muy poca diferencia entre impulsar un molino de rueda o caminar sobre una acera, especialmente si las otras señales sensoriales están ayudando a crear un entorno virtual coherente.

RETROALIMENTACIÓN TÁCTIL.

La tecnología actual permite que esto sea una realidad. Una de las maneras de afrontar este aspecto utiliza metales con memoria plástica, aleaciones que "recuerdan" su forma original y la recobran al ser calentadas. Estos metales son la base para los estimuladores táctiles, que se encastran en los guantes que lleva el usuario.

Cuando la computadora detecta una colisión entre su mano virtual y el objeto virtual en la pantalla, envía una corriente eléctrica al tacto, haciendo que la lámina de metal cambie de forma enviando un "duende" que aplica una presión contra la punta de su dedo.

Al incluir los tactores, un guante de entrada de datos puede actuar también como dispositivo de salida, suministrándole retroalimentación táctil en su relación con el entorno virtual. Imagínese moviendo su mano y sintiendo los lados de un cubo virtual suspendido en el ciberespacio, y sintiendo la sensación táctil en el momento de llegar a la arista del cubo. O imagínese recorriendo con sus dedos una superficie virtual rugosa, haciendo que los tactores se enciendan y se apaguen rápidamente creando la sensación de la misma acción en el mundo real. Los guantes de retroalimentación táctil de Xtensory le permiten sentir los objetos virtuales tal como los ve en la pantalla.

GRAVEDAD.

Aunque no sea un candidato a convertirse en un séptimo sentido, otro de los sentidos del tacto de suma importancia es la percepción por parte del cuerpo de los efectos producidos por la fuerza de gravedad. Si se conduce un coche de carrera, o se está volando en un reactor, el cuerpo de alguna manera sabe cuándo se realiza un giro cerrado, o cuándo se vuela hacia arriba o hacia abajo. Parte de esta sensación está relacionada con un sentido del tacto más general, ya que la acción de la fuerza de gravedad sobre el cuerpo se traduce en distintas presiones sobre los receptores de la piel. La otra sensación es la que ha sentido cualquiera que se haya montado en una montaña rusa, la sensación de que el estómago ha alcanzado una existencia independiente, y que en ese momento anda buscando un nuevo lugar donde ubicarse.

Los simuladores de vuelo y de conducción utilizan la sensación de gravedad sobre el cuerpo para engañarlo y hacerle creer que se mueve, sin ser esto cierto. Al inclinar el asiento del conductor hacia la izquierda (sentido contrario a las agujas del reloj) produce una sensación similar a la que se siente cuando un vehículo gira a la derecha. De forma similar, al inclinar el asiento hacia atrás se produce la sensación de aceleración, mientras que al inclinarlo hacia adelante produce la de frenado o desaceleración, las fuerzas de gravedad son las bases del Viaje a las Estrellas en Disneylandia, que crea la sensación de movimiento haciendo que los asientos individuales y toda la habitación oscile y se incline.

TRAJE.

El traje es básicamente un guante de datos específico para todo el cuerpo figura 20. Está instrumentado con el mismo tipo de cable de fibra óptica que recorre un guante de datos. Al moverse, curvarse o hacer señas el usuario, el sistema toma coordenadas espaciales para cada parte del traje, rastreando dinámicamente una extensa serie de acciones, 20 o más sensores recogen la mayoría de las articulaciones del cuerpo.

Estas señales digitalizadas son traducidas por la computadora a una realización, es decir, un cuerpo virtual que está expuesto en una pantalla o en un escenario virtual. La imagen generada queda sujeta a las señales dictadas por los movimientos del usuario y es regenerada continuamente. El usuario se identifica con la realización.

Las aplicaciones particularmente apropiadas para la entrada y alimentación por medio de trajes incluyen:

- Evaluación y prueba de movimientos incluyendo supervisión del movimiento y medidas del paso, y asesoramiento físico;
- Kinesología y aplicaciones y entrenamiento de la medicina deportiva, especialmente para mediciones funcionales y de ejecución:
- Terapias de rehabilitación para víctimas de golpes o lesiones y ayuda preventiva para lesiones repetitivas:
- Biomecánica:
- Terapia sexual y erotismo.



figura 20

BOLAS DE FUERZA.

Algunos dispositivos de bola se mueven muy poco. En su lugar, son equipados para que las fuerzas direccionales (empujes, sacudidas y giros) aplicadas por la mano del usuario, sean medidas como entradas. Al empujar fuerte se intensifica la medición, y el empuje se reduce cuando la presión es liberada. Estas se denominan "bolas de fuerza". Algunas bolas de fuerza están comercializadas con características definidas por el usuario, adaptadas con interruptores para salidas de emergencia del programa.

JOYSTICK Y JOYBOX.

Un joystick o una pareja de joysticks moduladores de velocidad, parecidos a los controles de un helicóptero, permiten al usuario total libertad de movimiento. Los arquitectos prefieren esto como dispositivo de control.

VARAS.

Como una varita mágica con un sensor en su extremo y un sensor de seis grados de libertad en su base, este dispositivo es probablemente el dispositivo de control más sencillo utilizado en entornos virtuales. Algunas varas tienen interruptores o botones de selección adheridos. Al blandir una vara, un participante de un mundo virtual puede utilizarla para pintar, interactuarse con otros objetos o emitir un conjunto limitado de órdenes.

Algunas versiones emiten un rayo láser desde su parte frontal. Cuando se apunta un objeto en la escena virtual, lo selecciona o recupera.

CONTROLADORES DE VOZ.

La entrada y salida de voz sintetizada son ventajas, permitiendo un control sin manos de un procedimiento. A menudo incluyen la eficacia de las entradas dictadas por un médico en una operación o por un soldado en el campo de batalla. Además la integración de la voz en las aplicaciones computarizadas está siendo justificada no solo desde este punto de vista del usuario sino como fuerza de asociación entre campos como la medicina y otros.

Como con todas las nuevas tecnologías, la conveniencia del reconocimiento y la síntesis de voz debe verse bajo la luz de una serie de cosas:

- La gran inversión inicial requerida para proporcionar equipamiento en aplicaciones individuales;
- Los recursos y el talento aportados de otros proyectos para su desarrollo;
- El volumen impracticable de los sistemas actuales disponibles para tales usos;

FACILIDADES TÁCTILES.

Los investigadores que diseñan entornos virtuales se esfuerzan en incorporar cualquier característica que eleve el realismo de su diseño figura 21. El grado de convicción de un usuario de que el mundo virtual es real es una fuerte medida de lo bien que ha resultado el diseño.

Los factores que son generalmente reconocidos como contribuyentes a una sensación táctil son una sensación de presión, una sensación de textura, una ausencia ó presencia de calor o frío y la sensación de olor; y los esfuerzos se están desarrollando para transmitir estas sensaciones a los participantes en entornos virtuales.

Las configuraciones de accesorios físicos y transductores, dispositivos que convierten una forma de energía en otra, son utilizados para elevar la ilusión de una persona de una inmersión sensorial total.

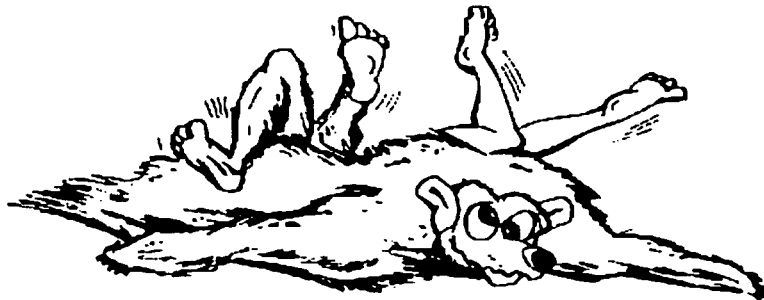


figura 21

PEQUEÑAS BOLSAS Y MICROAGUJAS.

Algunos modelos de guantes y trajes están equipados con microagujas y pequeñas bolsas inflables que proporcionan efectos hápticos y táctiles. Fueron usados por primera vez en simuladores de vuelo para dar el efecto de aceleración gravitacional. Las agujas y bolsas están conectados a la computadora y controlados para cambiar la posición y la forma muy rápidamente, en sincronización con el programa.

Así un usuario que se aproximase y cogiese la perilla de una puerta virtual figura 22, por ejemplo, recibiría no sólo una realimentación visual para elevar la ilusión del acto, si no también una realimentación programada a través de las agujas y las bolsas del guante, que simula las presiones en una mano cuando se agarra y gira una perilla real.

Cuando las coordenadas de la mano virtual coinciden con la perilla virtual, se produce un movimiento único de combinación. La ilusión continúa en armonía con la acción hasta que el contacto virtual permanece. Cuando se quita la mano virtual de la perilla, los microactuadores son desconectados, y el usuario nota un cese abrupto de la "sensación".



figura 22

Los usuarios que recogen o maniobran con un objeto virtual pueden "sentir" su peso o ligereza por el aumento de presión y los movimientos de las agujas del guante o traje.

Torsión y resistencia pueden ser simuladas mediante presiones y movimientos contrarios a los realizados por el usuario, así como hay sensores disponibles que transmiten la temperatura, permitiendo al usuario "sentir" calor o ausencia del mismo.

TERMADOS.

La mayoría de las aplicaciones de la RV no necesitan sensores de calor figura 23. En algunos casos, sin embargo, la habilidad para simular calor o frío, aunque sólo sea sobre un rango limitado, es especialmente valiosa. Una forma de hacer esto es encajar pequeños termodos, como collares, sobre los dedos del usuario.

Estos funcionan como pares térmicos. Cada uno contiene un sensor de temperatura y una pequeña bomba termoeléctrica que mueve calor dentro o fuera de un foco para producir sensación de calor o frío en el dedo del usuario. Los termodos y otros dispositivos de transmisión no tienen por qué estar limitados a las yemas de los dedos. Independientemente de su situación, su operación básica no cambia y, por tanto, pueden ser situados en cualquier lugar.

Independientemente del dispositivo térmico utilizado, éste utiliza señales de la computadora que controla el entorno virtual. Los valores de temperatura son asignados a objetos, lugares o superficies en el entorno virtual y registrados en el programa del sistema.

Cuando el usuario se acerca a cada uno, el sistema envía órdenes que desencadenan la respuesta correspondiente en los termodos del cuerpo. Para propósitos telerobóticos, los sensores de temperatura pueden ser grabados y las señales correspondientes mandadas al sistema, que luego lo transmite en los termodos del usuario.

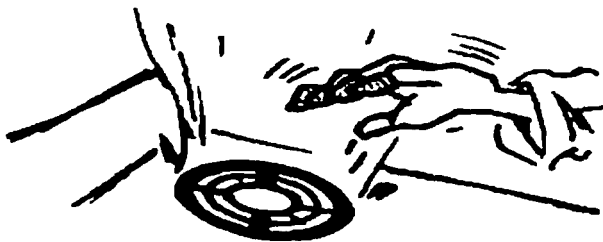


figura 23

CINTAS DE ANDAR.

Algunas áreas virtuales, como el diseño arquitectónico de un edificio o de un vestíbulo, son más grandes o extensas que aquellas que pueden ser cubiertas mediante un rastreador estacionario. Una cinta de andar es a menudo preferida por los usuarios que quieren unos medios de entrada más naturales. El dispositivo es modificado para medir la distancia recorrida y las mediciones se incorporan al programa figura 24. Estos son entonces reflejados en el mundo virtual que el caminante ve. Los mangos (normalmente de bicicleta) sirven como dispositivos de conducción cuando el usuario los gira, la escena virtual parece cambiar acordermente.

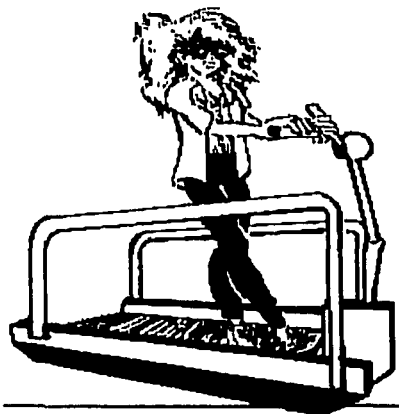


figura 24

BICICLETA.

Otro dispositivo de conducción y control es una bicicleta estacionaria que es pedaleada a través de un mundo virtual creado para simular un itinerario. Por ejemplo, un usuario puede elegir una ruta específica por cualquier país o una ruta subrealista que recorra galaxias psicodélicas. El usuario ve la escena en una gran pantalla situada delante de la bicicleta o a través de un dispositivo montado sobre la cabeza. La velocidad a la que la escena virtual pasa es determinada por lo rápido o despacio que el usuario pedalea, y la dirección de los mangos determina la orientación de la escena figura 25.

Los dispositivos de frenado controlados por computadora actúan sobre la rueda trasera, proporcionando realimentación al usuario para elevar los efectos de terreno desigual o abrupto. La computadora explora el progreso del usuario mediante un seguimiento de la velocidad y distancia relativa que ha recorrido en el camino virtual. Este tipo de estructura es también utilizado en simulaciones de vuelo, en cuyo caso se acomoda el mecanismo de dirección para efectos simulados de elevación.



figura 25

CAPITULO V.

PRESENTACIÓN DE IMÁGENES VIRTUALES.

EL SISTEMA RV: MÁQUINAS Y APARATOS.

Un sistema RV está compuesto de muchos sistemas. Apoyados en un fundamento básico de hardware, software y electrónica figura 1, se trata de sistemas independientes desarrollados para producir efectos visuales, auditivos y táctiles que son utilizados en entornos virtuales.

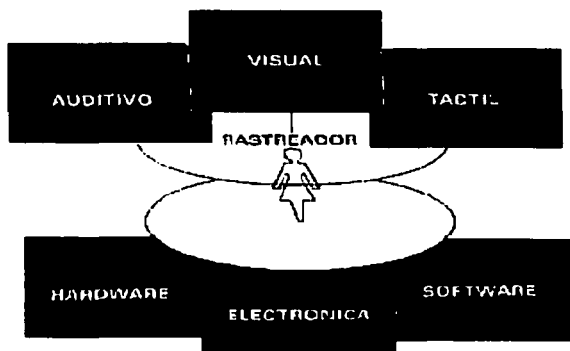


figura 1

Cada uno de estos sistemas refuerza un aspecto de la ilusión del usuario durante su inmersión en el mundo virtual. Todos ellos están integrados por unos sistemas accesibles de hardware, software y electrónica, de tal forma que actúan coordinadamente. Sus componentes son descritos con más detalle en la Parte IV.

GRÁFICAS INMERSIVAS EN 3-D.

La RV ofrece al usuario un “mundo que experimentar” figura 2. Se trata de un mundo animado electrónico.

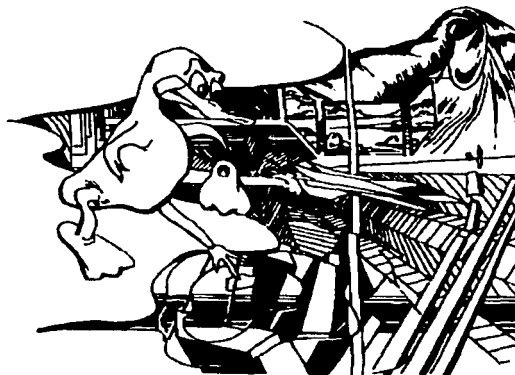


figura 2

Como cualquier arte visual, las gráficas inmersivas se basan en el deseo de creer del usuario. De cualquier forma, es un arte en el cual una persona puede sentirse inmersa.

OPACO O TRANSPARENTE

Una imagen exhibida puede aparecer ante el usuario de dos formas: *opaca*, en la que la visión del entorno virtual bloquea y suplanta la visión del mundo real; o *transparente*, en la que las imágenes virtuales parecen superponerse al mundo real, que continúa visible a través del cristal del casco.

En efecto, una presentación opaca reemplaza al mundo real, una tapa lo recubre figura 4. Cuando el usuario escoge la presentación transparente, en vez de la opaca, y continúa manteniendo el mundo real como marco de referencia, éste es más una "realidad aumentada" que una "realidad virtual".

La capa o cubrimiento transparente se consigue mediante unas superficies ópticas recubiertas de plata, en frente de cada ojo.

Las imágenes generadas aparecen como grabaciones electrónicas en superficies del mundo real, ayudas visuales y notas que son vistas como superposiciones en el mundo real.

La elección entre presentación opaca y transparente es frecuentemente conflictiva figura 3.

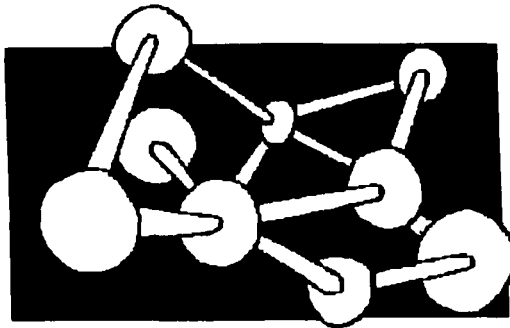


figura 3

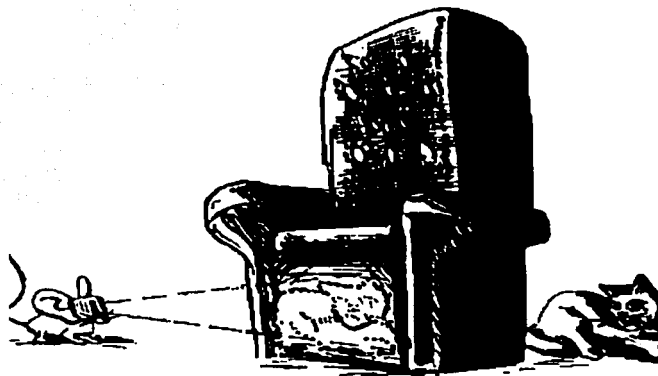


figura 4

GENERACIÓN DE POLÍGONOS EN TIEMPO REAL.

La velocidad de las máquinas utilizadas para producir imágenes virtuales es importante. Generar imágenes lo suficientemente rápido como para que parezcan reflejar acciones en tiempo real es el mayor problema al que se enfrentan los creadores de entornos virtuales.

Las máquinas de más alta calidad pueden producir imágenes complejas y bien definidas, pero lentamente. Por supuesto, también pueden producir otras rápidamente, pero simples y borrosas. Pocas máquinas son capaces de producir rápidamente imágenes complejas de alta resolución. Se ha trabajado mucho en la producción y presentación de buenas imágenes. Después de su primera aparición en el mundo virtual hace 35 años, Ivan Sutherland se aventuró en la exploración de ideas innovadoras aunque nunca abandonó definitivamente el "mundo virtual" figura 5.



figura 5

Cada imagen es un mosaico de pequeños polígonos, cada uno de los cuales es generado separadamente por la computadora. Para acercarnos a secuencias visuales realistas en tiempo real, la computadora tiene que generar 30,000 polígonos por segundo y el escenario debe actualizarse un mínimo de seis veces por segundo para conseguir la impresión de fluidez de movimiento figura 6.



figura 6

Siempre existe una correspondencia entre velocidad y calidad de imagen. Los algoritmos que convierten datos de información en pequeños y brillantes pixels dentro de la pantalla o en una escena virtual consumen, por lo general, una gran cantidad de recursos de la computadora.

El último dispositivo para estaciones de trabajo es un "acelerador" de gráficos, tiene el propósito de producir imágenes a una velocidad dos veces superior a la de otros sistemas. Otros experimentos con multicomputadoras son capaces de conseguir 2,000,000 de polígonos por segundo con una resolución de 1,280*1,624 pixels, ligeramente más nítidos que en la pantalla de una buena estación de trabajo.

Las pantallas hechas a mano, encargadas para el proyecto secreto "Super Cockpit" de las Fuerzas Aéreas de Estados Unidos, son capaces de simular condiciones de vuelo realistas y los investigadores han reconocido el realismo de los accidentes simulados cuando han sido sometidos a ellos.

De hecho, un piloto entrenado se puede desmayar cuando falla a una "aproximación". Hasta hoy sólo el "Super Cockpit" y otras pocas aplicaciones militares como los entornos de aviones de combate usados en el entrenamiento de pilotos para la Guerra del Golfo, se aproximan a un nivel lo suficientemente realista como para engañar completamente a los usuarios. Lo mejor que el resto de las aplicaciones ofrecen es una alta calidad de animación.

Esto no sabotea necesariamente un programa, pero se están realizando esfuerzos para minimizar la distracción causada por una reproducción de imágenes poco realista, incluyendo imágenes de fondo real. El trabajo está siendo consolidado en sistemas "captadores" de imágenes para permitir a los que lo desarrollan almacenar y modificar escenas reales para utilizarlas en entornos virtuales. Este ideal - reproducción fotorealista de imágenes desarrollada en tiempo real - y el fracaso de muchos sistemas en generar tales imágenes de alta calidad a altas velocidades es el mayor obstáculo para los entornos realistas virtuales.

Lo que se puede hacer de forma computacional está todavía muy lejos de lo que se puede hacer fotográficamente y lo efectivo de estos entornos depende mucho de la buena voluntad del observador para pasar por alto las imperfecciones.

MEJORA HÁPTICA.

Otro tipo de presentación virtual, que es utilizada al mismo tiempo que la visual, es la presentación háptica. En la presentación háptica las cosas parecen empujar, tirar, o irradiar en diferentes direcciones con grados diversos de fuerza.

Por ejemplo, los objetos que en el mundo virtual pueden ser provistos de campos de fuerza, torsión, fricción, calor y presión; son notables para que una persona interactúe con ellos.

Las presentaciones hápticas son usadas en juegos para elevar las sensaciones de velocidad o gravedad y son utilizadas por farmacéuticos químicos para encontrar lugares óptimos de enlace en moléculas, por ejemplo, en las enzimas.

La interacción en entornos hápticos requiere paciencia y una aguda vigilancia de las indicaciones. Un sistema utiliza un dispositivo mecánico llamado GROPE.

En efecto, en un conjunto virtual espacial más que obtener indicaciones reales, el usuario tiene la sensación de ir a tientas aleatoriamente, al estilo de un ciego, hasta que es provisto de algún tipo de realimentación positiva, como si fuese "el que busca" en el juego del escondite, sólo con ayuda de pistas como "caliente, caliente" o "frío, frío". Las señales de una presentación háptica han de ser interpretadas por el usuario para ajustar sus movimientos, de tal forma que puedan alcanzar alguna meta figura 7.



figura 7

Los participantes de mundos virtuales se acostumbran a ciertas indicaciones y las usan para medir la profundidad, la fuerza, o la distancia real. Los diseñadores ayudan incluyendo en el modelo muchas líneas y ángulos reveladores, forzando a que las cosas tomen una perspectiva ordenada.

Excepciones a esto serían aquellos ejemplos en los que se buscan estructuras paralelas en átomos o moléculas. En estos casos, la incorporación de rejillas al fondo puede ocultar las estructuras.

La realimentación de fuerzas, incluso para los objetos virtuales más sencillos, es una tarea muy difícil, y las presentaciones hápticas no están diseñadas como máquinas "al tacto", sino como entornos en los que una persona es capaz de obtener conocimientos a partir de las propiedades asociadas a los objetos representados.

Guantes y trajes están siendo gradualmente incorporados en las áreas que proporcionan efectos hápticos y táctiles a sus portadores. Se conectan a la computadora y se controlan para proporcionar efectos realistas a las presentaciones. Se describen con más detalle en la Parte IV.

REPRODUCCIÓN DE IMÁGENES EN EL ESPACIO REAL.

Una imagen que aparece en un espacio virtual pero que depende totalmente de algo en el espacio real se denomina imagen en espacio real. Una imagen en espacio real aparece sólo bajo unas condiciones muy específicas y en un lugar específico del entorno. Estas imágenes, que pueden parecerse prácticamente a cualquier cosa, sólo son informativas y útiles cuando son contempladas en relación a cosas basadas en el mundo real o en el contexto de las condiciones del mundo real y el usuario debe compartir su espacio virtual para verlas. La imagen, generada en el espacio virtual, pero totalmente dependiente de coordenadas conectadas al espacio real, aparece normalmente ante el usuario como un tipo de capa transparente en la escena figura 8.

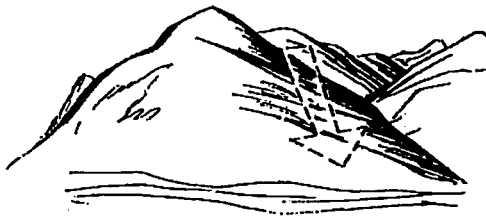


figura 8

Según se crean los objetos virtuales, cualquier elemento específico del mundo real en el que estén basados por ejemplo, condiciones relevantes, contexto conceptual, cultural, altitud, latitud y longitud es catalogado y registrado en una base de datos conectada al sistema. Así se convierten en imágenes "sensibles al contexto" que son visibles sólo cuando tienen significado completo. Las imágenes son exhibidas cuando las coordenadas de las imágenes registradas en la base de datos concuerdan con aquellas del entorno virtual que está siendo explorado.

En otras palabras, si el objeto virtual es un subconjunto de cualquier entorno virtual y pertenece a él, entonces aparece. Permanece mientras las condiciones del lugar y el contexto concuerdan.

Si el usuario sale de este entorno o si el contexto o las condiciones cambian de tal forma que la imagen deja de ser relevante, ésta desaparece.

DISTRIBUCIÓN.

Un sistema RV de distribución necesita proporcionar a los ojos del usuario imágenes de ángulo abierto, de alta resolución y bien enfocadas.

Los sistemas de distribución de imágenes para entornos virtuales deben crear escenas que sean detalladas de forma realista. Estas deben ser presentadas para corresponderse con escenas que el usuario vería si se moviese por ese entorno, si éste fuese real.

Una vez que las realizaciones de mundos virtuales han sido creadas, pueden ser almacenadas electrónicamente y movilizadas o transmitidas a través de redes.

MODOS DE PRESENTACIÓN: EL PUNTO DE VISTA DEL USUARIO.

Las imágenes son normalmente presentadas cilíndrica o esféricamente. Algunas aplicaciones para entretenimiento utilizan imágenes proyectadas, en las que se detectan los movimientos y acciones del jugador y se suponen imágenes fantasmales en tiempo real en pantalla figura 9.

En el modo de presentación cilíndrico, imágenes panorámicas de 360 grados envuelven al espectador lateralmente, pero no aparecen imágenes ni en el techo ni en el suelo.

Aun así, el usuario tiene una lograda sensación de estar en un entorno real.

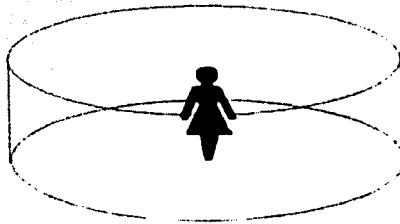


figura 9

Alternativamente, las imágenes pueden desarrollarse en un hemisferio visual o como una cúpula alrededor del usuario, en cuyo caso el movimiento vertical de la cabeza es acomodado en una especie de escenas enmarcadas. Ocasionalmente una especie de reja es añadida como punto de referencia del plano del suelo figura 10.

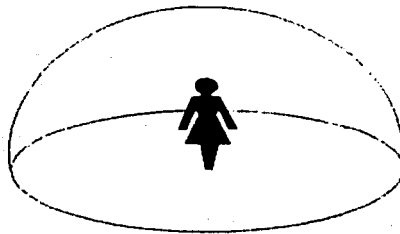


figura 10

Como la presentación debe ser calculada y recalculada en el momento que el usuario gira, se puede dar una notable latencia, un retraso entre el movimiento de la cabeza y la visualización, que puede deteriorar el efecto de realismo. Simplemente con que el usuario gire la cabeza, la escena cambia.

El objetivo es minimizar el retraso, ya que diferencias de milisegundos son advertidas agresivamente por el usuario. En la mayoría de los casos el usuario ve el mundo virtual desde un punto de vista egocéntrico, es decir, él está en el centro de este mundo.

Ocasionalmente, lo puede ver desde un punto de vista exterior. En estos casos la situación relativa del usuario puede ser marcada y cambiada por una flecha virtual, como una indicación móvil "tú estás aquí" en un centro comercial o guía turística.

PRESENTACIONES PROYECTADAS.

Este tipo de presentación no requiere ni cascos, ni guantes, ni trajes. En su lugar, un sistema de vídeo combina una imagen en tiempo real del usuario con una imagen en 2-D generada por computadora, que luego es proyectada como una imagen distante, llenando el campo de visión del usuario.

En los sistemas proyectados, como el dispositivo de presentación no es portado por el usuario, la fatiga es menor. Además los costos de mantenimiento son generalmente más bajos y la distorsión de las imágenes es mínima cuando la presentación está conectada a dispositivos de alta resolución, como las pantallas de computadora figura 11.

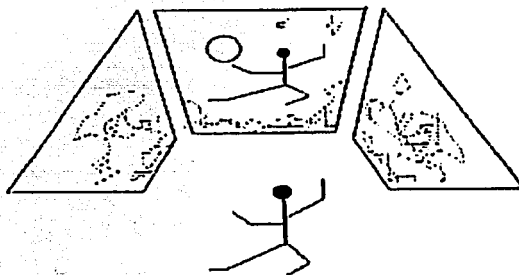


figura 11

Los usuarios combinan sus movimientos con la presentación y parecen interactuarse entre ellos y con las imágenes retratadas. Ser capaz de verse a uno mismo como parte del entorno que hay en la pantalla proporciona una sensación de inmersión al usuario. Es una técnica muy utilizada en el sector gubernamental para simuladores de armas de gran apertura de ángulo.

COMPLEJIDAD DE IMÁGENES.

Imágenes Precalculadas. Hay una ventaja en limitar la generación de imágenes a una presentación de los bordes laterales. Las presentaciones panorámicas están menos expuestas a los desfases entre tiempo y calidad antes mencionados que las presentaciones esféricas.

Al limitar el movimiento del usuario (sólo al extremo en que la credibilidad no sea amenazada), la pérdida de la máquina se reduce enormemente como la máquina reduce los gastos de generar cientos de trozos de imágenes poligonales, se liberan más recursos hasta el punto de que pueden ser distribuidos, en lugar de preparar imágenes anticipadas y almacenarlas, dejándolas disponibles en la memoria. Por tanto, ya no se necesita constantemente recalcular y generar una imagen en tiempo real cada vez que la cabeza del usuario se mueve figura 12.

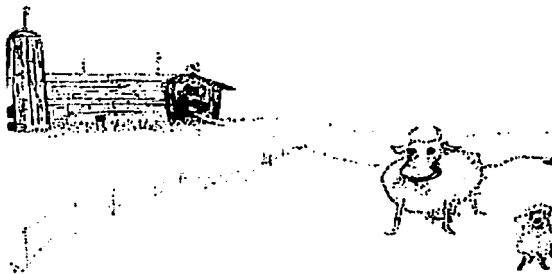


figura 12

La NASA lo ha hecho en sus sistemas, una panorámica de 360 grados es preprogramada y precalculada y luego almacenada anticipadamente en la memoria de imágenes - el almacén temporal de la computadora. Como estas imágenes ya no son computarizadas en el proceso, los usuarios no están expuestos a retrasos. En cambio, se les ofrecen imágenes previamente reproducidas, tan pronto como un cambio de posición y orientación es decir, de movimiento. También, como el factor de rapidez es suprimido del proceso de generación de imágenes, las imágenes precalculadas tienden a ser más detalladas, complejas y de un aspecto más realista que aquellas producidas en una rápida sucesión en tiempo real.

REFINAMIENTO ADAPTATIVO.

Otra forma de hacer frente a la complejidad computacional del usuario es generar escenas totalmente detalladas sólo cuando el usuario permanezca quieto. Cuando el nivel de actividad se eleva, la periferia parece desaparecer o regresar a una forma simplificada. En una presentación virtual, una persona que se mueve rápidamente de un lugar a otro advierte que el nivel de realidad baja y sube de nuevo tan pronto como el o ella "permanece quieto". En otras palabras, realismo e interactividad luchan entre si. Esta correspondencia entre realismo de imágenes e interactividad, denominada refinamiento de adaptación, aparece en algún grado en todos los sistemas actuales. Al reducir cualquier necesidad de interactividad, se permite un nivel más alto de realismo.

También, reproduciendo sólo lo que éste a la vista del usuario (es decir, un subconjunto del modelo total) la pérdida del sistema es reproducida; por lo tanto, la presentación es acelerada y el efecto de realismo es inmediato. El uso de la tecnología y el equipamiento es lo suficientemente refinado como para que sea transparente para la mayoría de los usuarios y la tendencia de nuestros cerebros de rellenar lo que falta en sensaciones casi completas aumenta el poder de sugestión de un RV. Aunque el ser acoplado a un equipo de cabeza se sacrifica algo de realismo, una persona percibe - como si se tratase de un pensamiento deseoso - que el o ella está en el centro de un entorno como una parte íntegra y activa de el. Los usuarios de sistemas de RV están generalmente deseando que sus mentes y sus ojos sean engañados figura 13.

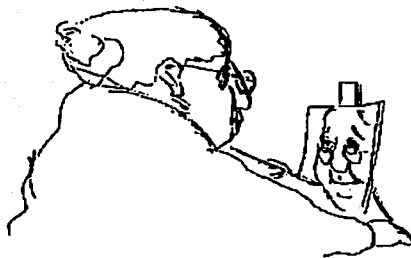


figura 13

PARTE IV. DISEÑO DEL EXPERIMENTO.

CAPITULO VI.

INVESTIGACIÓN Y DESARROLLO DE UN GUANTE VIRTUAL.

¿Nos acordamos todavía del nombre del primer videojuego doméstico?. Si la respuesta es Atari Pong, hemos acertado. No era muy sofisticado, pero fue el comienzo. Utilizando un pequeño botón de la consola de videojuegos se podía mover una pequeña raqueta en la pantalla con la que se podía golpear una pelota rectangular. Posteriormente, la misma compañía lanzó al mercado el sistema de entretenimiento doméstico Atari. Ahora, podemos utilizar una raqueta o un joystick. Durante este tiempo, han aparecido muchos otros dispositivos de entrada, como el joystick de pulgar de Mattel Intellivision y el joystick de cursos de Nintendo. Lo que todavía faltaba era un dispositivo de entrada que tuviera en cuenta la forma en que realmente se mueve nuestro cuerpo.

Hace ya algunos años, varias empresas estuvieron pensando en estas cosas, y se inventó el Power Glove. El Power Glove es un periférico del sistema de entretenimiento doméstico de Nintendo. Con este dispositivo, la computadora puede determinar dónde está la mano del usuario en un espacio 3-D, si el puño está cerrado o no y si la mano se está girando o no.

EL DISEÑO DEL POWER GLOVE.

En 1981, Thomas Zimmerman empezó a tener curiosidad por la música por computadora, y sobre si era posible tocar una guitarra de aire. Lo que se necesitaba era un dispositivo que imitara los movimientos de la mano. Tuvo una idea, fue a una tienda y compró un guante normal y algunos tubos de plástico. Empleando una técnica utilizada en fibras ópticas, creó un dispositivo de entrada que replicaba correctamente el movimiento de los dedos. Colocando los tubos de forma especial debajo de los dedos, se podía decir de si estaban doblados y en qué medida. En uno de los extremos hay una fuente de luz llegará al otro extremo del tubo. Midiendo la cantidad de luz que llegaba de la fuente en relación con la recibida en el otro extremo, Zimmerman podía saber el grado de curvatura del dedo. Obviamente, si el puño estaba cerrado, la cantidad de luz resultante sería mínima. En 1982, Zimmerman patentó su invento con el número 4542291.

Algún tiempo después, Zimmarman entró en contacto con Jaron Lanier, que estaba muy interesado en la música y había pensado en el uso del invento del guante. Se fundó la compañía VPL. En octubre de 1987, la revista Scientific American presentó un Dataglove de VPL en la portada. El Dataglove utilizaba fibras ópticas avanzadas para medir la curvatura de los dedos. Además, incorporaba un mecanismo de seguimiento que permitía a una computadora decir dónde estaba localizado el guante en el espacio e identificar tres posiciones angulares diferentes: inclinación, balanceo y desvío. El sistema de seguimiento fue desarrollado por Polhemus y utilizaba campos magnéticos para determinar esas propiedades.

MODIFICACIONES AL POWER GLOVE PARA QUE FUNCIONE EN LA COMPUTADORA.

Como ya sabemos, el Power Glove fue diseñado para el Nintendo, no para una computadora. En 1990, un artículo de la revista Byte explicaba cómo se podía conectar el Mattel Power Glove a una computadora de IBM o compatible. Se incluía un código fuente completo para interpretar el modo joystick del guante, denominado ahora el modo de baja resolución. Cuando se corrió la noticia, todo el mundo empezó a trabajar a un ritmo febril para descifrar los secretos del modo de alta resolución.

A finales de 1991, aparecieron varios mensajes en una lista de correo electrónico denominada glove-list, y en el grupo de noticias de USENET denominado sci.virtual-worlds. Se resolvieron los enigmas del modo de alta resolución y se dejó el código correspondiente en lugares de dominio público para que todo el mundo pudiera utilizarlo. Desde entonces, se ha actualizado el código para conseguir versiones más útiles. Algunas de esas personas son Greg Alt, Dave Stampe, Bernie Roehl y Mark Pflaging. Continuaremos el capítulo con un estudio en profundidad del Power Glove, de su modo de funcionamiento y de la disponibilidad de interfaces hardware y software con la computadora de IBM o compatibles. Así comenzamos nuestras indagaciones en la realidad virtual.

La conversión del Power Glove consta de dos partes:

- La primera es un ligero recableado sobre el guante,
- La segunda, la utilización del software.

Hay dos formas de recablear el guante, dependiendo de si vamos a conectarlo al puerto serial o al paralelo de nuestra computadora. Existe para ambos métodos, pero es más conveniente pensar en el método de conexión al puerto paralelo, ya que la mayoría del

software diseñado para el Power Glove sólo funciona si el guante está conectado en ese lugar.

MÉTODO DEL PUERTO PARALELO.

Éste es el método más conveniente. Es más sencillo y más barato, y además hay más software escrito para esta configuración.

Lista de componentes para una interfaz del Power Glove con el puerto paralelo

Para conectar el Power Glove al puerto paralelo de la computadora, se necesitan los siguientes elementos:

1. Por supuesto, la computadora
2. El Power Glove
3. Un cable de extensión. No es necesario, pero es una buena idea
4. Un puerto paralelo bidireccional en la computadora
5. Un conector paralelo DB-25 macho, en versión para cablear o para soldar (RS-232)
6. Un conector de caperuza DB-25 (RS-15)
7. unos cuantos metros de cable normal calibre 18
8. Cortadores de cable o un cuchillo X-acto
9. Un destornillador normal
10. Una fuente de alimentación (las conexiones se examinan más adelante)
11. Un soldador y cable de soldar (opcional)
12. Un multímetro (opcional)

La figura 1 muestra el diseño completo de la interfaz con el Power Glove.

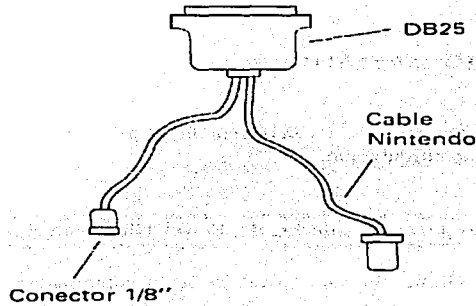


Fig. 2 La interfaz con el Power Glove.

Es fundamental que nuestra computadora tenga un puerto paralelo bidireccional. Cuando se diseñó originalmente la computadora, el puerto paralelo sólo estaba pensado para transmitir datos de salida hacia una impresora.

Pero sucedió que el puerto paralelo tenía varias características que lo hacían útil como puerto de entrada. A medida que los fabricantes de clónicos empezaron a ampliar los periféricos, diseñaron sus propios puertos paralelos genéricos y sus tarjetas multipuerto como bidireccionales, lo que significaba que la información se podía transmitir en sentido de entrada y de salida.

En nuestro caso, vamos a enviar datos al Power Glove y no recibir datos de retorno. Por lo tanto, si el guante y la interfaz no se comunican bien, lo más probable es que el puerto paralelo no sea bidireccional.

EXAMEN DEL CABLEADO DEL POWER GLOVE.

Lo primero que tenemos que hacer es determinar qué conexiones del cable del Power Glove se conectan con los pines del conector del otro extremo. Si estamos utilizando el conector original sin cable de extensión, buscamos la caja en el extremo de los receptores en triángulo. Debe existir un cable corto que sale de la caja. Debemos cortar aproximadamente 1.2 cm desde el conector del extremo (como muestra la figura 3). Si estamos utilizando el cable de extensión, el extremo de dicho cable se conecta con el extremo del conector que sale de la caja, y se corta el extremo restante del cable de extensión.

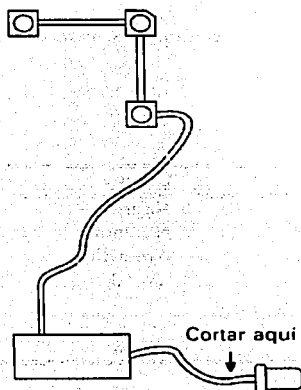


Fig. 3 Cortado del cable del Power Glove.

Se pela la cubierta negra aproximadamente 3.5 cm para dejar al descubierto los cables. Después se pela la cubierta de cada uno de los cables hasta que quedan al descubierto aproximadamente 2 cm de cable metálico.

USO DEL CABLE ORIGINAL.

Esta sección es útil si se está utilizando el cable original (no el cable de extensión). Si no es así, se puede pasar a la sección titulada Trabajo de investigación. Comenzar el proceso de conexión observando los cables coloreados en el extremo del cable que acabamos de cortar. Debemos ver cables de los siguientes colores: Negro; Naranja; Amarillo; Verde y Rojo. Existen otros dos cables, pero no nos interesan por ahora. Cada uno de esos colores se refiere a una patilla en particular del conector (ver tabla A).

Debemos verificar que existen los cinco colores mencionados y que están conectados a la patilla correcta. Si es así, podemos pasar a la patilla denominada conexión. Si falta un color o los colores son diferentes a los antes mencionados, vamos a la siguiente sección.

Tabla A.

No. de patilla	Color	Función
1	Negro	Tierra
2	Naranja	Reloj
3	Amarillo	Latch
4	Verde	Salida
5		Nada
6		Nada
7	Rojo	+5 voltios

TRABAJO DE INVESTIGACIÓN.

Si no encontramos los colores mencionados en el párrafo anterior, tomamos el conector de Nintendo que eliminamos del extremo de la caja pequeña o del extremo del cable de extensión.

Se corta el plástico negro hasta que aparezcan dos o tres centímetros de cables coloreados.

Se elimina aproximadamente un centímetro del plástico de cada uno de los cables coloreados para dejar al descubierto el conductor metálico. Hay que procurar dejar visible algo del plástico coloreado, de forma que se puedan reconocer los cables. La siguiente operación requiere un multímetro puesto en resistencia o un montaje, como en la figura 4.

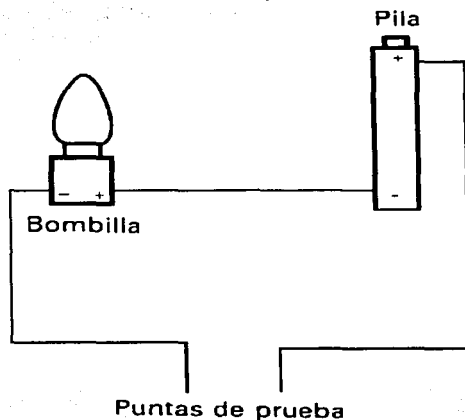


Fig. 4 Circuito de prueba de los cables del Power Glove.

Utilizando el multímetro o el circuito con la bombilla junto con el conector de Nintendo, se conecta uno de los extremos a uno de los hilos coloreados. Se escribe en un papel el color del cable que hemos conectado al cable de prueba. Con el otro extremo, vamos tocando cada una de las patillas del conector de Nintendo hasta que el multímetro indique que hay resistencia, o la lámpara se encienda. Cuando eso sucede, indica que el cable del color correspondiente y la última patilla que hemos tocado están conectados. Podemos entonces crear una lista como la Tabla A, escribiendo el color del cable y el número de patilla correspondiente. La operación se repite para cada uno de los cables. Cuando hayamos terminado, habremos establecido la relación entre los cables y las patillas.

LA CONEXIÓN A LA COMPUTADORA.

Ahora tenemos que conectar el guante al puerto paralelo. Para esta parte de la interfaz, utilizaremos un conector macho DB-25. Los más fáciles de utilizar son los que tienen las patillas para enganchar, aunque la desventaja es que la conexión no es tan sólida como los que se soldan. Necesitaremos también cortar los trozos de cable, aproximadamente 15 cm de largo, para alimentar el cable. Los cables del Power Glove y los dos cables que hemos preparado anteriormente se conectan como se muestra en la figura 5.

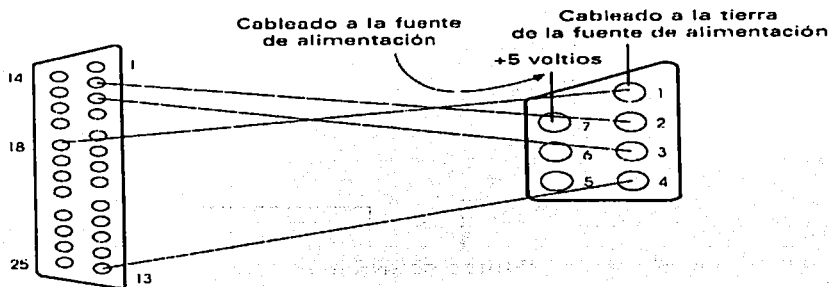


Fig. 5 Cableado del Power Glove al conector DB-25.

Cuando hayamos terminado, el esquema de cableado debe ser el siguiente:

Patilla del Power Glove	Se conecta con
Pin 1	Pin 18 del conector paralelo y tierra (a la fuente de alimentación)
Pin 2	Pin 2 del conector paralelo
Pin 3	Pin 3 del conector paralelo
Pin 4	Pin 13 del conector paralelo
Pin 7	+ 5 voltios (fuente de alimentación)

CONEXIÓN DE LA ALIMENTACIÓN.

La alimentación se puede conectar al guante de cuatro formas diferentes: (1) desde el teclado, (2) desde una fuente externa, (3) desde un conector de periférico externo y (4) desde una batería. Existen razones para escoger cada una de las computadoras. Si queremos poder conectar el guante a distintas computadoras, las opciones del teclado y de la fuente de poder externa son las mejores.

Cualquiera que sea nuestra elección, debemos asegurarnos de que utilizamos sólo +5 voltios, que la tierra está conectada a la patilla 1 (o el color asociado) y que los +5 voltios están conectados a la patilla 7 (o el color asociado). Esto es muy importante. Si la polaridad de las conexiones se cambia, es muy probable que la placa base de la computadora resulte dañada al conectar el Power Glove al puerto paralelo.

ALIMENTACIÓN EXTERNA.

Una de las formas más simples de alimentar nuestro Power Glove es mediante una fuente de alimentación externa. Que suministre 5 voltios. Aunque se ha sugerido que se podrían utilizar 4.5 o 6 voltios para alimentar el guante, no es conveniente, ya que la electrónica del guante espera ser alimentada con 5 voltios, y existe la posibilidad de que el guante resulte dañado.

Fuente de alimentación externa

Se necesita lo siguiente:

1. Transformador con salida de 5 voltios
2. Conector mono de 1/8"
3. Clavija mono de 1/8"

Hay dos formas de conectar la fuente de alimentación al guante. En el caso del modelo de Coleco, el conector del extremo no tiene utilidad.

Por tanto, hay que cortarlo. Así la alimentación se puede conectar al guante directamente, la otra opción, especialmente cuando se utiliza una fuente nueva, es utilizar un sistema de clavija-conector. Esto permite desmontar rápidamente la fuente de alimentación si se desea. La figura 6 muestra la conexión.

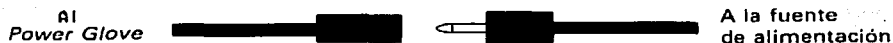


Fig. 6 Clavijas para la alimentación desde una fuente externa.

El primer paso es determinar si es necesario conectar la clavija a la fuente de alimentación. Si hemos comprado una fuente de alimentación de desecho, existe la posibilidad de que el conector del extremo no sea utilizable. Necesitaremos entonces conectar una clavija mono de 1/8" a las líneas de +5V y de tierra (GND). Habitualmente, la fuente de alimentación dispondrá de un diagrama que indicará los voltajes de cada cable. Se busca la línea de tierra (GND) y se conecta la parte larga de la clavija ver figura 7.

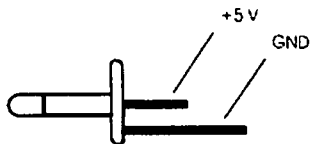


Fig. 7 Conexión de tierra (GND) y +5v a la clavija macho.

Después se localiza la línea de +5V y se une al conector de menor longitud de la clavija. Para verificar que la conexión es correcta, se puede utilizar un multímetro para medir la tensión de la clavija. La punta positiva, generalmente roja, debe tocar la punta de la clavija, y la negativa, generalmente negra, la base. El multímetro debe marcar +5V cuando la fuente de alimentación esté conectada.

El siguiente paso es conectar una clavija hembra a los hilos del Power Glove. Para ello, basta con repetir exactamente los mismos pasos anteriores, sólo que utilizando los terminales +5V y tierra (GND) del guante. La figura 8 muestra un ejemplo de conexión.

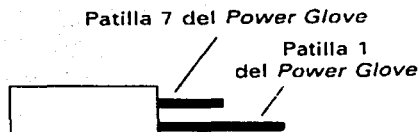


Fig. 8 Conexión de la clavija hembra al Power Glove.

PRUEBA DE LA CONEXIÓN.

Una vez alimentado nuestro guante, es hora de probarlo. Antes de realizar una prueba completa, es conveniente verificar la conexión de la alimentación para que no haya riesgo de dañar la computadora. Si ya hemos enchufado el conector paralelo a nuestra computadora, lo desconectamos.

De momento, no tiene que haber conexión física entre el guante y la computadora. Si estamos utilizando un cable de extensión para el guante, también hay que desconectarlo. Se conecta la fuente de alimentación al guante. Al conectar la alimentación, debe oírse un pitido, y las luces del sensor en triángulo pueden encenderse o no. Si se encienden, puede ser que no lo hagan todas. Si no oímos el pitido, es que algo va mal.

Si sale humo, es seguro que algo va mal. Volvemos a comprobar las conexiones con el multímetro; comprobamos si el voltaje del conector Nintendo es el correcto tocando la patilla 7 del cable Nintendo con la punta positiva, y la patilla 1 con la punta negativa. La lectura debe de ser de + 5 voltios. Si no es así, hay que comprobar la fuente de alimentación. Si el multímetro indica un valor negativo, puede ser que hayamos cableado incorrectamente la fuente de alimentación. Utilizando de nuevo el multímetro, comprobamos la fuente de alimentación para asegurarnos de que la patilla 7 está a +5V, y la patilla 1 a tierra (GND).

Una vez verificado que el cable de Nintendo está al voltaje correcto, se recomienda volver a comprobar todas las conexiones. Se une el montaje del Power Glove a la interfaz y se conecta el conector DB-25 al puerto paralelo. Se enciende la computadora. Si todo está funcionando correctamente, el guante emitirá un pitido y las luces del receptor en triángulo podrían parpadear una o dos veces. Si esto sucede, la conexión fue un éxito, si no revisar todo desde el principio.

EL SISTEMA DE POSICIONAMIENTO POR ULTRA SONIDOS.

El sistema de posicionamiento por ultrasonidos consta de dos transmisores ultrasónicos localizados en la parte superior del Power Glove y tres receptores dispuestos en forma de triángulo. ¿Como convierte el sistema la información recibida en forma de ultrasonidos en posiciones x , y , z y giro de la mano?. Explicaremos primero la base matemática, y después lo ilustraremos con un ejemplo.

Los receptores de ultrasonido del Power Glove están dispuestos en forma triangular, como se muestra en la figura 9. Etiquetaremos cada uno de los receptores con las letras R1, R2 y R3. R1 es el receptor de la esquina superior izquierda, R2 es el de la esquina superior derecha y R3 es el receptor de la esquina inferior derecha. Si suponemos que el receptor R1 está en la posición 0, 0, 0, podemos determinar la posición de los otros dos receptores. La distancia del receptor R1 al receptor R2 es de 16 unidades, lo que significa que R2 está en la posición 16, 0, 0. El receptor R3 está a 16 unidades desde R1 y a 16 unidades desde R2, así que está en la posición 16, 16, 0. Nótese que todas las posiciones anteriores tienen 0 en la coordenada z . Esto es debido a que los tres receptores están en el plano de coordenadas x - y .

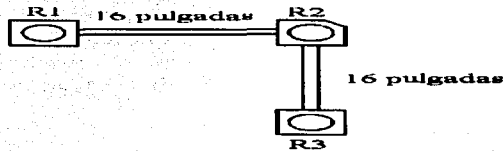


Fig. 9 Localización de receptores de ultrasonidos.

Imaginemos por un instante que cada uno de los receptores son los puntos centrales de tres esferas. Lo que estamos intentando encontrar es la posición x , y , z del transmisor de ultrasonidos. Esta posición estará localizada en cada una de las esferas. La electrónica del Power Glove nos proporciona la distancia del transmisor a cada uno de los tres receptores.

Denominemos a esos valores $R1d$, $R2d$ y $R3d$. Estos valores representarán entonces la distancia de cada uno de los receptores a uno de los transmisores. Si esto es así, entonces $R1d$ es el radio de una esfera con centro en el punto $R1$ y que pasa por la ubicación del transmisor, x , y , z . Lo mismo se aplica para los otros dos receptores. El problema que tenemos entonces es encontrar los valores x , y , z que satisfagan la ecuación de la esfera para cada uno de los receptores. La ecuación de una esfera es:

$$\text{Radio}^2 = (x-x_{\text{centro}})^2 + (y-y_{\text{centro}})^2 + (z-z_{\text{centro}})^2$$

Si sustituimos las coordenadas de los tres receptores, tenemos estas tres ecuaciones de la esfera:

$$R1^2 = (x-x1centro)^2 + (y-y1centro)^2 + (z-z1centro)^2$$

$$R2^2 = (x-x2centro)^2 + (y-y2centro)^2 + (z-z2centro)^2$$

$$R3^2 = (x-x3centro)^2 + (y-y3centro)^2 + (z-z3centro)^2$$

Si sustituimos ahora la información anterior sobre el origen de los tres receptores, obtenemos las ecuaciones:

$$R1^2 = (x)^2 + (y)^2 + (z)^2$$

$$R2^2 = (x-16)^2 + (y)^2 + (z)^2$$

$$R3^2 = (x-16)^2 + (y-16)^2 + (z)^2$$

Podemos despejar las incógnitas x e y de la primera y segunda ecuaciones, con lo que obtenemos la ecuación:

$$R1^2 = R2^2 + 32x - 256$$

despejando x obtenemos:

$$x = \frac{(R1^2 - R2^2 + 256)}{32}$$

Repitiendo el procedimiento anterior para cada una de las restantes ecuaciones, se obtiene lo siguiente:

$$y = \frac{(R1^2 - R2^2 + 256)}{32}$$

y

$$z = \text{SQRT}(R1^2 + x^2 + y^2)$$

Si observamos nuestras incógnitas, vemos que una vez que el receptor tiene las distancias R1, R2 y R3, la electrónica puede determinar dónde está localizado el transmisor, y por lo tanto dónde está localizada la mano.

Hagamos ahora algunos números. Supongamos que los receptores miden las siguientes distancias: R1d = 25, R2d = 30, R3d = 10. Sustituyendo en las anteriores ecuaciones, podemos determinar que el guante está situado en la posición (0, 33, 41). No obstante, es un valor falaz, ya que está basado en la posición del espacio donde está centrado el guante.

GIRO.

El giro del guante es mucho más simple. No hay que olvidar que el guante tiene dos transmisores. La única razón de su existencia es determinar el giro del guante.

Primero tenemos que localizar la posición de los dos transmisores utilizando la técnica vista anteriormente.

Con un poco más de matemática, podemos ver que si dividimos la diferencia de los valores de y entre la diferencia de los valores de x, obtendremos la tangente del ángulo de giro.

$$\tan(\text{ángulo de giro}) = \frac{(y_{t1} - y_{t2})}{(x_{t1} - x_{t2})}$$

Ahora todo lo que necesitamos es hacer una simple función tangente y tendremos el ángulo deseado. Pero la función tangente no es práctica, de forma que crearemos una tabla de referencias con los valores de la tangente ya precalculados.

La electrónica se ocupa de buscar esa tabla el valor más cercano al obtenido en la ecuación anterior. Cada uno de los valores precalculados tiene un número asociado que va de 0 a 11. La relación de los números con el giro del guante se indica a continuación:

- 0 - El guante está situado con la palma hacia abajo.
- 3 - El guante está situado con la palma mirando a la izquierda.
- 6 - El guante está situado con la palma hacia arriba.
- 9 - El guante está situado con la palma hacia la derecha.
- 11 - El guante está situado con la palma casi en la posición 0 anterior.

Todos esos valores son aproximados. Si tenemos el guante en una posición con la palma hacia la izquierda, los números variarán de 0 a 4. Ésta es toda la matemática necesaria para el diseño de un seguidor ultrasónico.

PROGRAMACIÓN DEL POWER GLOVE.

Ahora que hemos finalizado el hardware de nuestra interfaz con el Power Glove, hay que ver la forma de controlar el Power Glove mediante software. Una de las primeras cosas a tener en cuenta de la interfaz hardware es que todo está controlado mediante un reloj, un latch y una línea de datos. Se menciono anteriormente que éramos capaces de obtener información de la posición y el giro del guante. El guante envía esa información serialmente, un bit cada vez. Además existen temporizaciones específicas entre el guante y la computadora que deben ser coordinadas. Pero vayamos por partes. Empezaremos por analizar el modo joystick de baja resolución del Power Glove.

MODO DE BAJA RESOLUCIÓN.

El software para el modo de baja resolución se incluía en el artículo original sobre la interfaz paralela aparecido en la revista Byte. El código estaba escrito en lenguaje ensamblador. Lo que vamos a hacer es analizar una versión en C del código y tratar de entender el proceso de comunicación con el guante.

El puerto paralelo de la computadora está localizado en la posición 378 en hexadecimal. Esta es la posición de salida de datos. Al comunicarse con una impresora, ésta es la posición donde debe colocarse el carácter de salida.

La posición 379 en hexadecimal corresponde a un registro de estado. En ella se lee al byte de estado de la impresora. Este byte en realidad está formado por cinco líneas de estado en las que la impresora puede poner 0 o 5 voltios para indicar un problema.

En el modo de baja resolución del Power Glove, el guante está diseñado para emular un joystick. Utilizando un simple algoritmo, podemos preguntarle al guante lo que está haciendo, y éste responderá con un solo byte cuyo valor nos dará la información necesaria. El algoritmo para acceder a la información de baja resolución es:

```
Bucle
    señalar el guante
    bucle para los 8 bits
    utilizar los 8 bits
fin bucle
```

SEÑALIZACIÓN DEL GUANTE.

El guante se señala enviando un valor a las señales de latch y de reloj de la interfaz. Empezaremos poniendo a 0 la línea del latch manteniendo inicializada la línea de reloj en `outportb(0x378, 1)`.

La orden `outportb` envía el parámetro deseado, un byte, al primer parámetro, una dirección. Tras borrar el latch, cargamos la línea del latch y enviamos un pulso `outportb(0x378, 3)`. La señalización finalizada borrando la línea del latch: `outportb(0x378, 3)`. El guante sabe en este momento que desearíamos obtener información sobre su posición actual.

BUCLE PARA LOS 8 BITS.

Tan pronto como señalizamos el guante, éste comienza a enviarnos la información, un bit cada vez. Necesitamos poder recibir los bits y formar un byte completo con ellos. El código para realizar esta operación podría ser el siguiente:

```
b=0;
for (i=0; i<8; i++)
{
    b <= 1;
    b+=(inportb(0x379) & 0x16)>>4;
    outportb(0x378, 0);
    outportb(0x378, 1);
}
b*=0xFF;
```

Comenzamos inicializando a cero la variable en la que vamos a almacenar la información: `b=0`;. Seguidamente, entramos en un bucle con una cuenta total de 8 iteraciones: `for(i=0; i<8; i++)`.

Toda la información contenida en nuestra variable se desplaza hacia la izquierda para hacer sitio a un nuevo valor: `b<=1`;. Leemos el puerto paralelo y sumamos el valor del bit en nuestra variable: `b+=(inportb(0x379)&0x16)>>4`;. El bucle de iteración finaliza enviando un pulso a la línea de reloj del guante: `outportb(0x378, 0); outportb(0x378, 1)`;

USO DE LOS 8 BITS.

Lo último que tenemos que hacer es utilizar los 8 bits de datos que hemos recibido. El significado de cada bit en nuestra variable de almacenamiento es el que sigue:

- Bit 7- Un valor de 1 indica que la información es válida.
- Bit 6- Se pone a 1 si el guante está cerrado o se ha pulsado un botón.
- Bit 5- Se pone a 1 si está pulsado el botón SELECT.
- Bit 4- Se pone a 1 si está pulsado el botón START.
- Bit 3- Se pone a 1 si el guante se mueve hacia arriba.
- Bit 2- Se pone a 1 si el guante se mueve hacia abajo.
- Bit 1- Se pone a 1 si el guante se mueve hacia la izquierda.
- Bit 0- Se pone a 1 si el guante se mueve hacia la derecha.

MODO DE ALTA RESOLUCIÓN.

Este modo es más complicado que el modo de baja resolución. El motivo principal es que en este caso el guante espera que la información se transfiera de acuerdo con una temporización específica.

El guante funciona de forma que da y recibe. Se le envía un pulso de reloj y el guante envía un bit (literalmente) de datos.

El primer paso para obtener datos del guante en alta resolución es configurarlo en modo de alta resolución. Todo el software de dominio público está de acuerdo en que el algoritmo de inicialización para el modo de alta resolución es:

- Enviar un pulso de inicialización al guante
- Leer 4 bits del guante y descartarlos
- Establecer la comunicación con el guante utilizando las líneas de reloj y de latch
- Enviar un código de 7 bits - 0x06, 0xC1, 0x08, 0x00, 0x02, 0xFF, 0x01
- Retardo
- Liberar la línea de inicialización
- Esperar un tiempo

ALGO MÁS CERCANO AL CÓDIGO C:

```

// Enviar un pulso de inicialización al guante
outputb (0x378, 0x01);
outputb (0x378, 0x03);
retardo (3-4 microsegundos); // Nota: No son milisegundos
outputb (0x78, 0x02);
// leer 4 bits y descartar
retardo (304 microsegundos);
outputb (0x378, 0);
outputb (0x378, 0x01);
retardo (304 microsegundos);
outputb (0x378, 0);
outputb (0x378, 0x01);
retardo (304 microsegundos);
outputb (0x378, 0);
outputb (0x378, 0x01);
retardo (304 microsegundos);
outputb (0x378, 0);
outputb (0x378, 0x01);
// Establecer comunicaciones
outputb (0x378, 0x01);
retardo (7212 microsegundos);
outputb (0x378, 0x03);
retardo (2260 microsegundos);
// Enviar 7 bits
hi_res_codes[7] = {0x06, 0xC1, 0x08, 0x00, 0x02, 0xFF, 0x01};
for (i=0; i<7; i++)
{ x = hi_res_code[i];
  for (j=0;j<8;j++)
  { // Salida de los 8 bits a un tiempo
    if (x&0x80)
    {
      outputb (0x378, 0x03);
      outputb (0x378, 0x02);
      outputb (0x378, 0x03);
    }
    else
    {
      outputb (0x378, 0x01);
    }
  }
}

```

```
        outputb (0x378, 0x00);
        outputb (0x378, 0x01);
    }
    x=x<<1;
    retardo (3-4 microsegundos);
}
retardo (85-150 microsegundos);
}
retardo (892 microsegundos);
outputb (0x378, 0x01);
retardo (50000 microsegundos);
```

Veamos un par de componentes de este código. El primero es la salida del puerto paralelo. La lectura de un bit de datos procedente del guante se realiza poniendo a cero tanto el reloj como el latch: `outputb (0x378, 0);`. Después, la línea del reloj se pone a nivel alto `outputb (0x378, 0x01)`.

Seguidamente se lee el bit procedente de la línea de datos del guante: `inportb (0x379)&0x01`. Recuérdese que la línea de reloj del Power Glove está conectada al bit 0 del puerto paralelo.

El guante se inicializa enviando un pulso por la línea de latch. La línea de latch se pone a estado cero, pero el reloj se mantiene a nivel alto: `outputb (0x378, 0x01);`. Después, la línea de latch se pone a nivel alto, junto con la del reloj: `outputb (0x378, 0x03);`.

Es necesario introducir un retardo para permitir que el guante lea la línea de latch; después, la línea del latch se vuelve a poner en estado cero: `outputb (0x378, 0x01);`.

Se puede enviar un bit al guante utilizando dos combinaciones de reloj/latch, en función de si el bit es un cero o un uno. Si el bit es un uno, las líneas de reloj y de latch estarán a nivel alto: `outputb (0x378, 0x03);`.

Seguidamente, se produce un pulso de reloj poniéndolo primero a cero, y seguidamente a nivel alto: `outputb (0x378, 0x02); outputb (0x378, 0x03);`.

Si el bit es un cero, se ejecuta la misma secuencia, pero el latch se mantiene en estado cero: `outputb (0x378, 0x01); outputb (0x378, 0x00); outputb (0x378, 0x01);`. Entre dos bits consecutivos se produce un retardo de 3-4 microsegundos, y entre cada byte, un retardo de 85-150 microsegundos.

RETARDO.

En este momento es indispensable comentar algo sobre los retardos de microsegundos que deben ocurrir en las secuencias de transferencia de datos. El guante fue diseñado para trabajar con un solo microprocesador de la máquina del Nintendo. Este procesador es el 6502, el mismo que se utilizaba en la serie de microcomputadoras APPLE II. Esto significa que los diseñadores tenían que preocuparse de las transferencias de bits a la velocidad del 6502, y no de otras máquinas.

Desde el diseño de la interfaz con la computadora, los diseñadores de software se las han tenido que ver con los retardos. La solución típica ha sido utilizar un bucle de temporización con dos variables, que se utilizaban para incrementar o reducir el tiempo de recorrido del bucle en función de la máquina que se utilizara.

Este paquete utiliza un método simple que realiza los cálculos para encontrar los valores a colocar en los bucles de temporización en la máquina en la que se está ejecutando el programa. Así aumenta mucho la fiabilidad del Power Glove cuando se conecta a una computadora.

LECTURA DEL GUANTE.

Lo último que debemos hacer con la interfaz software es comenzar a leer datos. La inicialización del guante se realiza mediante el siguiente algoritmo:

```
Bucle
Al leer enviar a la salida de datos el valor 0xA0
Enviar la posición x
Enviar la posición y
Enviar la posición z
enviar el valor de rotación
enviar el valor de los dedos
enviar el valor de los botones
FinBucle
```

EL CÓDIGO DE ESTE ALGORITMO ES:

```

while (! kbhit ())
{
    while ((a = get_glove_byte( )) != 0xA0
           delay (2000-4000 microsegundos);
    x = get_glove_byte( );
    y = get_glove_byte( );
    z = get_glove_byte( );
    rot = get_glove_byte( );
    fingers = get_glove_byte( );
    buttons = get_glove_byte( );
    discard = get_glove_byte( );
    discard = get_glove_byte( );
}

```

Para utilizar este código, necesitaremos la función `get_glove_byte()`. Esta función es:

```

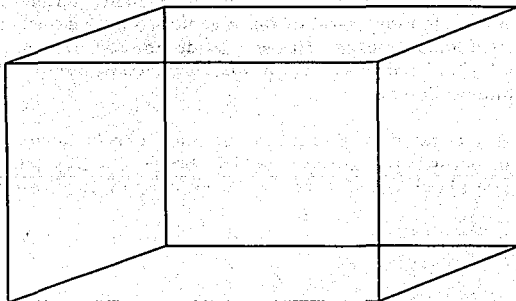
unsigned char get_glove_byte( );
{
    // Inicializar guante
    outportb (0x378, 0x01);
    outportb (0x378, 0x03);
    delay (3-4 microsegundos);
    outportb (0x378, 0x01);
    // La inicialización anterior también indica al guante que debe enviar
datos, ya que ya vimos el 0xA0
    // Leer 8 bits
    for (i=0; i<8; i++)
    {
        val = val<<1;
        val += inportb (0x379) & 0x01;
        outportb (0x378, 0x00);
        outportb (0x378, 0x01);
    }
    delay (85-150 microsegundos);
    return x;
}

```

Una vez que el valor 0xA0 ha sido detectado y se han leído todos los bytes del guante, podemos utilizarlos para los propósitos que estemos necesarios. Debido a la interfaz con la computadora, la lectura de datos del guante no es tan simple, pero tampoco es compleja. Ahora que lo sabemos todo sobre comunicaciones del Power Glove y tenemos acceso a un código que nos permite la comunicación, es el momento de escribir software para hacer cosas. En los capítulos anteriores, hemos visto una introducción, la descripción de términos y cómo se puede conectar un dispositivo a nuestra computadora que permiten añadir cierto grado de realismo a nuestro mundo virtual. El paso siguiente es programar nuestra computadora para crear los mundos virtuales y para programar el hardware. El software para construir mundos virtuales se denomina PAQUETE DE REPRESENTACIÓN (Rendering Package). Este software acepta como entrada números como por ejemplo:

8	6	0		
0	0	0		
0	2	0		
2	2	0		
2	0	0		
2	0	-2		
2	2	-2		
0	2	-2		
0	2	-2		
0	0	-2		
4	1	2	3	4
4	4	3	6	5
4	5	6	7	8
4	8	7	2	1
4	2	7	6	3
4	8	1	4	5

y los transforma en la siguiente imagen:



Los programas de representación no solamente transforman los números de un objeto y posteriormente lo visualizan. El software debe tener en cuenta dónde estamos en nuestro mundo virtual. Lógicamente, no tiene sentido representar objetos que están detrás de nosotros, y por lo tanto fuera de nuestro campo de visión. Además de donde estamos, el software de representación debe tener en cuenta desde que perspectiva estamos mirando al mundo. ¿Estamos tumbados mirando hacia arriba, o estamos en el aire mirando hacia abajo? Una vez que el software de representación posee toda esta información, determina que objetos se deben representar y cómo se deben representar en pantalla.

SOFTWARE DE REPRESENTACIÓN.

Sean de alto o bajo costo todos los software de representación funcionan de la misma manera:

- Bucle
- Obtener entrada del usuario
- Transformar y Proyectar
- Vértices
- Ordenar objetos
- Eliminar facetas ocultas
- Color
- Dibujar
- FinBucle

ENTRADA DE USUARIO.

El usuario proporciona la entrada generalmente por el teclado o el ratón; por tanto, el software de representación debe ser capaz de gestionar tanto entradas de teclado como movimientos del ratón. En algunos casos, el ratón se utiliza para controlar una mano virtual dentro de un mundo. Además de esos dispositivos de entrada, tenemos que considerar el Power Glove, el ratón 3-D, el sistema de seguimiento de los movimientos de la cabeza y el software de reconocimiento de voz.

Como el Power Glove y el ratón 3-D se han construido con la misma interfaz de entrada básica, se pueden considerar simultáneamente. El software de representación debe de ser capaz de comunicarse con la electrónica del Power Glove para leer los valores de x, y y z,

el valor de giro, además del grado de curvatura de los dedos. El Power Glove se puede utilizar muy adecuadamente para controlar una mano virtual, de forma que el software de representación sea capaz de determinar en que parte del mundo virtual debe colocar la mano. Obviamente, esta colocación estará relacionada con la dirección en la que estamos mirando.

Para determinar a donde estamos mirando en el mundo virtual, el software de representación puede utilizar la información que suministra el sistema de seguimiento de la posición de la cabeza. Si este sistema indica que estamos mirando a la izquierda, el software de representación debe cambiar apropiadamente el aspecto del mundo que vemos. En lo que respecta al software de reconocimiento de voz, el software de representación debe estar preparado para aceptar entradas de teclado, ya que nuestro sistema de voz activará dicho software a través del teclado.

TRANSFORMACIÓN Y PROYECCIÓN DE VÉRTICES.

Cuando se diseña un objeto para incluirlo en un mundo virtual, los puntos de los objetos se posicionan utilizando coordenadas del mundo, que utilizan valores de x, y y z referidos a algún origen 0, 0, 0.

El software de representación se ocupará de convertir los objetos a valores de pixel x, y de la pantalla. Esta conversión se denomina proyección, y requiere sólo dos pasos.

Coordenadas del mundo	Coordenadas de visualización	Coordenadas de pantalla
-----------------------	------------------------------	-------------------------

En el primer paso es transformar los vértices expresados en coordenadas del mundo a coordenadas del espacio de visualización. Hay varios sistemas de coordenadas diferentes que se pueden utilizar en el espacio de visualización, de los cuales el más común es la perspectiva. Las fórmulas para convertir coordenadas del mundo en coordenadas de visión perspectiva son:

$$\text{Projected_X} = \frac{(x * d)}{z}$$

$$\text{Projected_Y} = \frac{(y * d)}{z}$$

Nótese que la coordenada z tiene el mismo valor en los dos sistemas de coordenadas. El segundo y último paso en la proyección es convertir las coordenadas de visión en coordenadas de espacio de pantalla, que son las posiciones x , y de los puntos que van a componer los objetos en la pantalla de la computadora. Como la coordenada z no existe en la pantalla de la computadora, simplemente la descartamos. Los valores x , y se obtienen utilizando las ecuaciones:

$$\text{Screen_X} = \text{Projected_X} = \frac{\text{Screen_Width}}{2}$$

$$\text{Screen_Y} = \text{Projected_Y} = \frac{\text{Screen_Height}}{2}$$

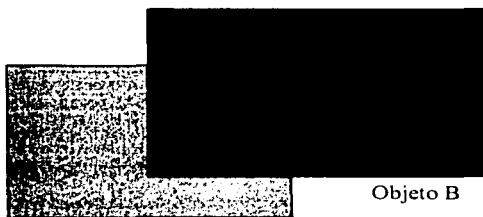
Además de la proyección de los vértices de los objetos, la computadora debe mover los objetos de acuerdo con las interacciones que realiza el usuario utilizando el teclado u otros dispositivos de entrada. Si el usuario desea mover un objeto a una distancia determinada en la dirección de la coordenada x , la computadora deberá ajustar los vértices del objeto adecuadamente. Este ajuste se realiza generalmente utilizando matrices de transformación. La matriz de transformación correspondiente a la traslación (movimiento) es:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ tx & ty & tz & 1 \end{bmatrix}$$

Todos los vértices de un objeto se deben transformar utilizando multiplicación de matices. Los cálculos son muy costosos desde el punto de vista tiempo, si tenemos en cuenta que el sistema de representación debe trabajar en tiempo real.

ORDENACIÓN DE OBJETOS POR SU COORDENADA Z.

Una vez que todos los objetos tienen asignadas coordenadas de visión y de pantalla, los ordenaremos basándonos en su coordenada z . Esto se hace para determinar que objetos están delante de cuáles. Si tenemos dos objetos y el objeto A está delante del objeto B, tendremos que dibujar primero el objeto B y después el objeto A. El resultado sería:



Objeto A

Objeto B

Si dibujáramos primero A y después B, obtendríamos el resultado contrario:



Objeto A

Objeto B

Ordenando todos los objetos, podremos dibujarlos siguiendo una lista atrás adelante. En la práctica, la lista se mantiene ordenada todo el tiempo. Cuando un objeto se transforma utilizando una matriz de traslación o de rotación, se localiza en la lista y se reposiciona en la lista utilizando su nueva coordenada z.

ELIMINACIÓN DE SUPERFICIES OCULTAS.

La eliminación de superficies ocultas se realiza para ahorrar tiempo de representación. Si tenemos un cubo en nuestro mundo y miramos a uno de los lados, no será necesario representar el lado opuesto, ya que no lo veremos. La eliminación de superficies ocultas se determina a partir del vector normal para un polígono específico. Si el vector normal está dirigido hacia el usuario, la superficie deberá representarse. Si está dirigido hacia la parte opuesta del usuario, se puede eliminar.

COLOR.

El color es un elemento importante para dotar de realismo a nuestro mundo virtual. Cada luz poseerá una dirección y un color. Cuando el software de representación comienza a dibujar una nueva pantalla, determinará de que forma afectan las luces a los objetos basándose en el ángulo que forma las luces con cada objeto. Si el objeto está directamente enfrente de la luz, dicho objeto recibirá toda nuestra intensidad de la luz y deberá ser coloreado en consecuencia. Si el objeto forma algún ángulo con la fuente de la luz deberá utilizarse alguna fracción de la intensidad disponible. Empleando algún procedimiento de sombreado, cada uno de los objetos puede tener sombras del mismo color en función de la intensidad de iluminación.

DIBUJO.

El último paso del procedimiento de representación es el dibujo. Los desarrolladores de paquetes de software de representación han dedicado un esfuerzo y tiempo considerable a este asunto. Ello es debido a la cantidad de tiempo que se dedica a dibujar en la pantalla de la computadora. Cuanto más rápidas sean las rutinas de dibujo de líneas, más rápido podrá el programa de representación actualizar la pantalla tras una determinada entrada por parte del usuario. La mayoría de este código se escribe en lenguaje ensamblador altamente optimizado.

CREACIÓN DE UN OBJETO.

Existen varias formas diferentes de crear un archivo 3-D. La primera es dibujar el objeto en un papel. Dependiendo del objeto que sea, esto puede ser muy sencillo o complicado, ya que en papel sólo podemos hacer representaciones 2-D. Cuando los objetos se complican tanto que dibujarlos sobre papel resulta imposible, podemos utilizar un sistema CAD, con el que podremos dibujar en tres dimensiones.

El único problema es que el formato PLG no está muy extendido. De hecho, el único sistema CAD que puede producir archivos PLG es NorthCad de Quest. A efecto de demostración diseñaremos una casa sobre papel.

El primer paso es dibujar un cubo utilizando una técnica que tal vez se calificará de infantil. Las figuras A a la F muestran los pasos necesarios para dibujar un cubo tridimensional sobre papel. Seguidamente tenemos que asignar coordenadas a cada uno de los tres vértices del cubo. La forma más sencilla es asignar a la esquina inferior izquierda de la parte delantera la coordenada $0, 0, 0$. La figura G muestra los cuatro vértices inferiores etiquetados con sus coordenadas. Podemos ahora movernos a los vértices superiores replicando las cuatro coordenadas de los vértices inferiores modificando los valores de 0 de la componente y . La figura H muestra todos los vértices y sus coordenadas. Lo siguiente que necesitamos es un tejado para la casa. Podemos construirlo centrando un punto encima del cubo y dibujando líneas desde los vértices superiores a ese punto. También tenemos que darle a ese punto una coordenada, como muestra la figura I. Lo que hemos hecho es crear sobre papel un objeto muy simple junto con sus coordenadas. El paso siguiente es poner las definiciones del objeto en un archivo PLG:

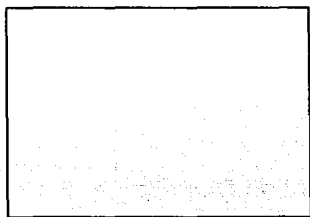


figura A

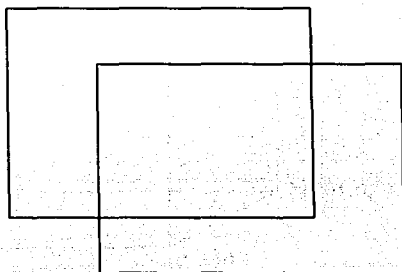
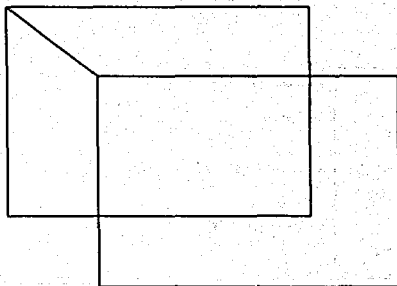
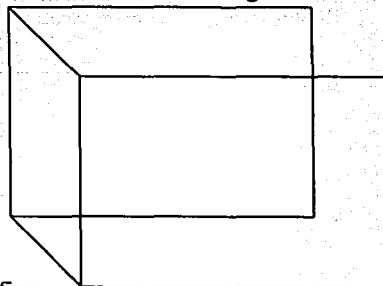


figura B

figura
Cfigura
D

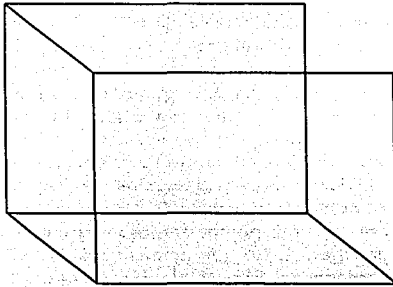


figura E

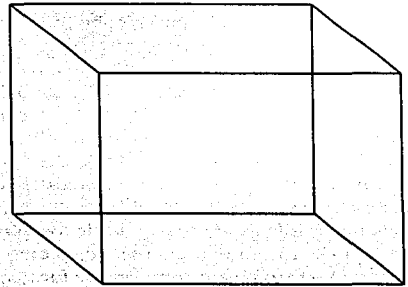


figura F

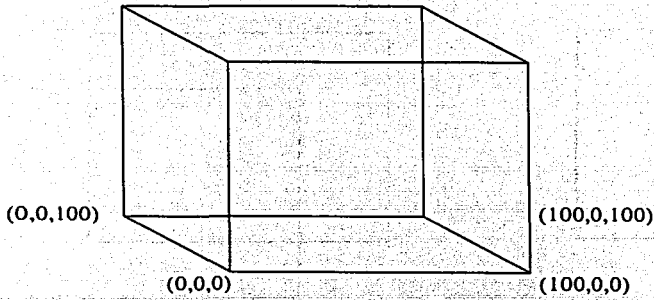


figura G

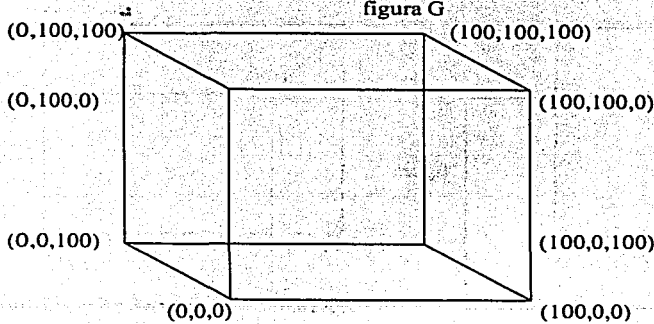


figura H

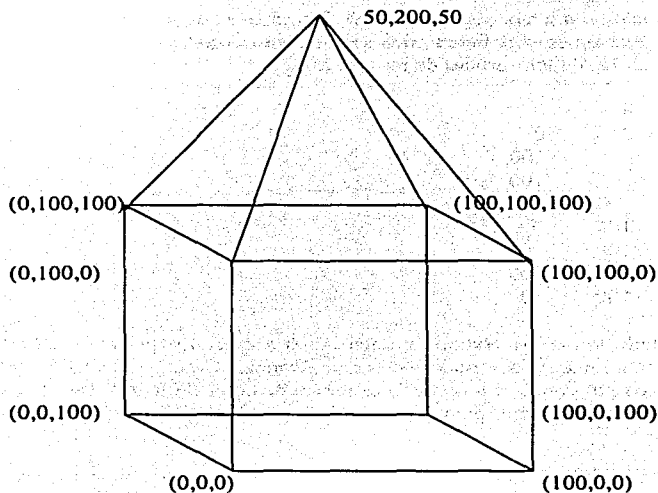


figura I

CREACIÓN DE UN ARCHIVO PLG.

Todos los archivos PLG comienzan con una línea de esta forma: **NOMBRE # vértices # polígonos**, como se diseñó una casa, elijeremos el nombre de **CASA**. Para encontrar el número de vértices, sólo tenemos que observar nuestro objeto. Hemos puesto coordenadas en nueve vértices, de modo que la palabra **CASA** debe ir seguida del número 9. Sigue el número de polígonos del objeto.

Un polígono es un objeto creado a partir de tres o más puntos conectados de forma que las líneas no se corten. Un polígono con tres vértices es un triángulo. Si observamos de nuevo la casa que hemos diseñado, vemos que tiene cuatro lados, una planta y un tejado formado por cuatro polígonos. Esto da un total de nueve polígonos, de manera que escribimos otro nueve, y el encabezamiento de nuestro archivo PLG queda así: **CASA 99**.

Después de la línea de encabezamiento, sigue la lista de los vértices de nuestro objeto. El orden no importa, pero generalmente es buena idea seguir algún esquema ordenado. Esta podría ser la lista de los valores de los vértices de nuestra casa.

```

0      0      0
100    0      0
100    0      100
0      0      100
0      100    0
100    100    0
100    100    100
0      100    100
50     100    50

```

Una vez que se han listado todos los vértices, comenzamos a crear los polígonos. Cada polígono tiene asociada una línea de descripción con este formato: COLOR # VERTICES v1 v2 v3 ... el color de los polígonos se basa en el esquema de color de REND386. No se entrará en detalle sobre dicho esquema, pero sí se presenta una lista con los 16 primeros colores y sus valores:

NEGRO	0
ROJO	1
NARANJA	2
ROSA	3
MARRÓN	4
AMARILLO CLARO	5
AMARILLO	6
VERDE CLARO	7
VERDE	8
CIAN	9
AZUL CLARO	10
AZUL	11
ROSA BRILLANTE	12
MAGNETA	13
GRIS	14
BLANCO	15

VERTICES es el número de vértices que componen los polígonos, y $v_1 v_2 v_3 \dots$ son los números de índice de los vértices de la lista del principio del archivo PLG. Hay que tener en cuenta que los índices de los vértices comienzan en 0, no en 1. Los números de índice de los vértices se deben indicar en el sentido contrario a las agujas del reloj, suponiendo que se mira al polígono de frente. Para el suelo de la casa, por ejemplo, podríamos imaginarnos de pie en la casa y mirando hacia el suelo. La descripción del polígono sería color 4 1 0 3 2. Por cada uno de los polígonos que forman nuestro objeto debe existir una línea como ésta. Una vez que el archivo PLG de objetos está terminado, podemos añadir el objeto al archivo del mundo. Todos los objetos se añaden utilizando un descriptor de objeto y este formato: OBJECT nombrepkg Sx Sy Sz Rx Ry Rz Tx Ty Tz.

La palabra OBJECT se utiliza para indicar a las rutinas de análisis gramatical que se trata de una línea descriptora OBJECT. Cuando el programa de análisis gramatical ve esa palabra, espera encontrar el nombre del archivo PLG correspondiente al objeto. Ese nombre se utiliza para abrir dicho objeto. Los nueve valores restantes se utilizan para determinar las características específicas del objeto cuando éste se situó en el mundo.

El primer grupo, Sx Sy y Sz, son los valores de escala del objeto en cada una de las coordenadas. Esos valores se utilizan para crear versiones de diferentes tamaños del objeto utilizando el mismo archivo PLG. De este modo, si deseamos tener una casa grande, basta con utilizar valores grandes para esos tres parámetros. Los valores 1, 1, 1 dejarán al objeto exactamente con el tamaño definido en el archivo PLG. Los valores menores que 1 reducirán su tamaño. Los valores Rx Ry y Rz se utilizan para rotar el objeto alrededor de cada eje de coordenadas. Por ejemplo, si deseamos que nuestra casa quede cabeza abajo, utilizaremos los valores 180, 0, 0. Los valores de Tx Ty y Tz se utilizan para trasladar al objeto a un lugar específico del mundo. Así es como creamos nuestro mundo virtual.

Los objetos se pueden situar en cualquier punto del mundo virtual simplemente colocando los valores adecuados en los parámetros de traslación. A veces es muy útil dibujar un mapa con la situación de los objetos en nuestro mundo y situar después los objetos de acuerdo con ese mapa. Una vez que hemos colocado los objetos en el archivo del mundo, estamos listos para explorar dicho mundo en el programa principal.

CAPITULO VII.

JUGANDO AL FRONTÓN.

En capítulos anteriores, se han presentado los conceptos, antecedentes e historia, interfases, periféricos, de la realidad virtual y los detalles del software de presentación denominado REND386. Vamos a utilizar este software de presentación para crear una aplicación: de un frontón virtual. Por varias razones.

- Podemos ver la teoría de la realidad virtual llevada a la práctica.
- El número de polígonos necesario para construir el frontón virtual es pequeño, de forma que la velocidad del juego será alta, incluso en computadoras no muy rápidas.
- Podrá utilizar la mayoría de los proyectos hardware que se puedan construir.

JUGANDO AL FRONTÓN.

Para poder jugar, necesitaremos tener conectado a la computadora el Power Glove y su interfaz. La forma más simple de jugar es congelar la raqueta y después ir hacia el otro lado de la pista. Esto nos da oportunidad de golpear la pelota tanto cuando proviene del frente de la pista como cuando proviene de la parte de atrás.

PROGRAMACIÓN DEL JUEGO DE FRONTÓN.

Antes de empezar la codificación, necesitamos analizar el problema que tenemos. Deseamos que el juego de frontón simulado sea lo más parecido posible al frontón real. Esto significa varias cosas:

1. Necesitamos contar con varios elementos del juego:
 - * Raqueta
 - * Pelota
 - * Un campo de juego o pista
2. Necesitamos poder manejar la raqueta y golpear la pelota
3. El juego debe proporcionar una sensación real de profundidad, no sólo debe utilizar gráficos.

Hay que analizar cada componente por separado. En primer lugar, la creación de los elementos del juego es simple. Sólo hay que diseñarlos sobre papel y crearlos con el software. La pelota es un elemento de particular interés, ya que nuestros bloques básicos de diseño son polígonos. Tenemos que determinar cuántos polígonos son necesarios para crear este objeto esférico.

En segundo lugar, tenemos que considerar nuestra movilidad en el mundo virtual. En el mundo real manejamos la raqueta con nuestras manos, que están unidas a nuestro cuerpo.

Finalmente, llegamos a la cuestión de la tridimensionalidad. Durante el juego del frontón, se gasta una buena parte del tiempo en utilizar la percepción de profundidad para determinar dónde está la raqueta en relación con nosotros mismos. Esto será muy importante en nuestro juego virtual.

La pista del juego de frontón se construirá utilizando el sistema REND386 descrito en el capítulo 6. Se utilizará la línea de compiladores Turbo C/Borland. Para poder recompilar el código, necesitaremos obtener previamente el sistema de desarrollo REND386 a partir de las fuentes mencionadas en el capítulo 6.

CONSTRUCCIÓN DEL EQUIPAMIENTO.

El equipamiento necesario para el juego de frontón es lo suficientemente simple como para diseñarlo sobre el papel con muy poco esfuerzo. Comenzaremos con la pista de juego.

LA PISTA DE JUEGO DE FRONTÓN.

Una de las consideraciones más importantes cuando se diseña la pista de juego de frontón es el realismo.

Todas las dimensiones deben ser adecuadamente escaladas de forma que el usuario tenga la sensación de que realmente está jugando en una área grande. Esto suena bien en teoría, pero en práctica hay que resolver varios problemas. El primero el tamaño de la pista. El segundo problema al que nos enfrentamos está relacionado con el primero: no tenemos una forma realista de movernos por un área tan grande. Nadie ha desarrollado un sistema de seguimiento que permita a los usuarios moverse por áreas tan grandes. El tercer problema es el de la seguridad. Si jugamos al frontón virtual en el caso de que existiera una pantalla montada en un casco, no podremos ver los obstáculos, y podríamos hacernos daño.

La solución a esos problemas es escalar la pista hasta un tamaño que permita a los usuarios permanecer de pie o sentados en una posición fija y utilizar el Power Glove para golpear la pelota. Aunque esto limita el realismo del juego. En REND386, cada unidad del mundo virtual corresponde aproximadamente a 1 mm del mundo real. Las dimensiones de la pista de frontón serán de 3200 unidades de ancho, 4000 unidades de longitud y 2400 de altura.

Esto se convierte en 3.2 metros x 4.0 metros x 2.4 metros. La pista se muestra en la figura 1. Podemos ver que la pista no tiene parte de atrás. Se ha eliminado para mantener el sentido de orientación del usuario. Si todas las paredes estuvieran representadas, sería muy fácil perderse. La pérdida de una de las paredes no afectará los movimientos de la pelota, ya que podemos controlar su movimiento mediante software.

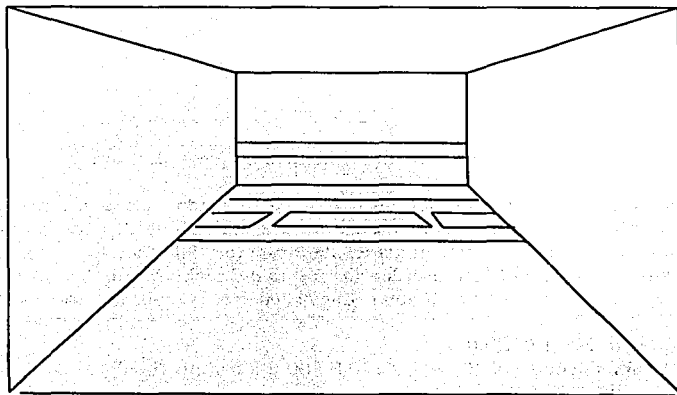


Fig. 1 Vista delantera de la pista de juego de frontón virtual.

Empezaremos nuestra construcción de la pista por el suelo, que es un polígono sencillo que se puede construir utilizando la orden polyobj en un archivo de mundo. La orden completa para construir el suelo es:

```
polyobj 4 sgray 800,0,0 800,0,4000 4000,0,4000 4000,0,0
```

Además, existen en el suelo varias líneas que se utilizan durante el juego normal del frontón. Esas líneas se dibujan con las órdenes:

```
polyobj 4 sred 800,0,3000 800,0,3100 4000,0,3100 4000,0,3000
polyobj 4 sred 800,0,2200 800,0,2300 4000,0,2300 4000,0,2200
polyobj 4 sred 1200,0,2300 1200,0,3000 1300,0,3000 1300,0,2300
polyobj 4 sred 3500,0,2300 3500,0,3000 3600,0,3000 3600,0,2300
```

La figura 2 muestra el diseño de la pista con el suelo completo.

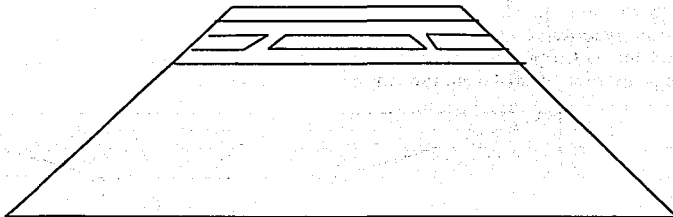


Fig. 2 Suelo de la pista de juego de frontón

Lo siguiente que añadiremos son las paredes de los lados. Cada una de ellas se crea a partir de un archivo PLG denominado courtw12plg. Este archivo PLG es la representación de un polígono de dos lados de 100 unidades de longitud y 50 unidades de anchura. Utilizaremos los parámetros de escala en la orden de objeto del archivo mundo. Además, debemos rotar las paredes para ponerlas en su lugar. Las órdenes para realizar esto son:

```
split 800,1,0 1,0,0 0
object courtwal 40,84,4 0,270,0 800,0,0 0 walls fixed
split 4000,1,2000 1,0,0 0
object courtwal 40,84,4 0,270,0 4000,0,0 0 walls fixed
```

La figura 3 muestra la pista con las dos paredes incorporadas.

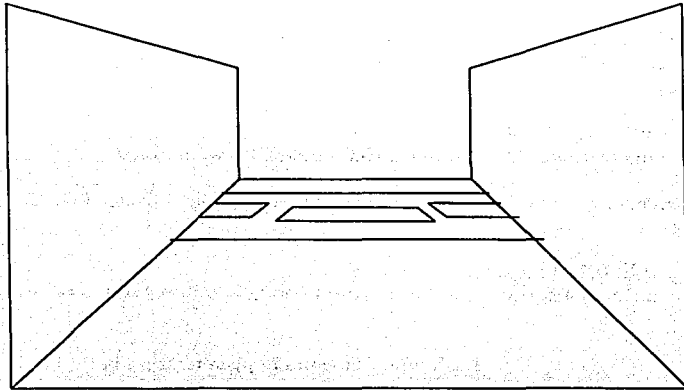


Fig. 3 Paredes de la pista de frontón

A continuación añadimos la pared frontal. Esta pared se diseña a partir de un archivo PLG denominado courtwal.plg, que es exactamente lo mismo que el archivo PLG courtw12.plg excepto que utiliza un valor diferente para la orden surfacedef de coloreado de polígonos del archivo del mundo.

Habremos notado que hemos utilizado varios colores diferentes para cada pared de la pista en lugar de utilizar un color uniforme. La razón de esto es que REND386 no se ocupa directamente de la iluminación. Si creamos dos polígonos, como nuestras paredes, y las colocamos formando un ángulo entre sí, como la esquina de una casa, encontraremos que el sistema no es capaz de cambiar por sí mismo los colores de las paredes de forma que seamos capaces de diferenciarlas.

Al no poder diferenciarlas con la vista, lo que vemos es un gran bloque de pared. La esquina de unión de las dos paredes resulta invisible. Debido a eso, tenemos que colorear las paredes manualmente para conseguir el efecto realista esperado.

La pared frontal tiene también marcas que se utilizan en el juego, lo mismo que el suelo. Por tanto, hay que incluir esas marcas al dibujarla. Las órdenes que se utilizan para dibujar la pared son:

```
object courtwall 32,48,4 0,0,0 800,0,4800 0 walls fixed
object line 3200,100,10 0,0,0 800,300,3900 0 lines fixed
```

La última pared por dibujar es la pared trasera. Se forma utilizando la información del archivo courtw12.plg, lo mismo que las paredes laterales. Las órdenes para dibujar esta pared son:

```
split 4000,1,2000 1,0,0 0
object courtw12 40,48,4 0,270,0 4000,0,0 0 walls fixed
```

Finalmente, terminamos el diseño de la pista incluyendo el techo. Para ello usamos la orden polyobj:

```
split 0,2400,0 800,0,4800 0
polyobj 4 sgray 4000,2400,0 4000,2400,2400 800,2400,2400 800,2400,0
```

La figura 4 muestra la pista terminada.

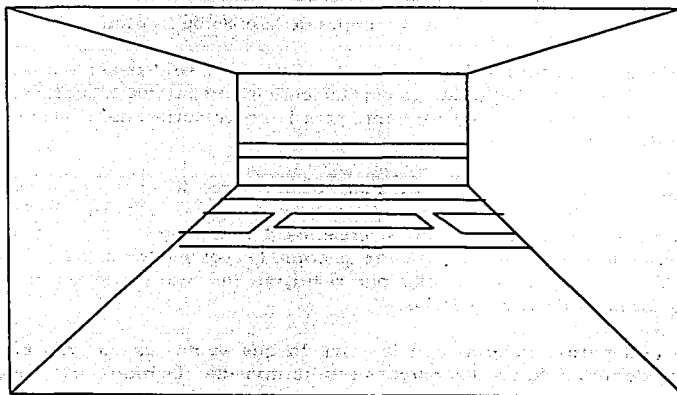


Fig. 4 La pista de frontón completa

LA PELOTA.

Ahora que ya tenemos una pista, necesitamos crear una pelota. Uno de los objetos más difíciles de crear en un entorno virtual (o en entornos gráficos tradicionales) es un círculo o una esfera. El motivo es que para crear el arco del círculo o de la esfera tenemos que unir en forma de escalera los píxeles de la pantalla de la computadora. La figura 5 muestra una sección de un círculo dibujado en una pantalla.

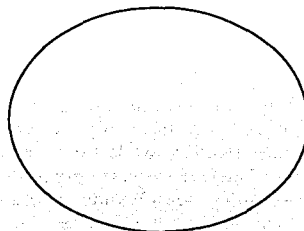


Fig. 5 Un círculo generado por computadora

Veamos cómo cambian los píxeles en sentido horizontal y vertical para poder formar el círculo. Imaginemos ahora que el círculo se transforma en una esfera tridimensional.

La situación se complica aún más debido a que sólo disponemos de polígonos bidimensionales para construir nuestra pelota. Una solución es utilizar múltiples bandas angulares de polígonos. Una sencilla forma de imaginar esto es pensar en una señal de alto. Si tumbáramos la señal de alto en el suelo y por la parte superior de nuestra señal de alto y unimos cada vértice de la señal con los vértices del polígono. Crearíamos otra banda que forma un ángulo con la primera banda. Si repetimos esto con la parte inferior de la señal de alto, tendríamos una aproximación a la esfera. Las figuras 6 a 8 muestran la secuencia de operaciones descritas.

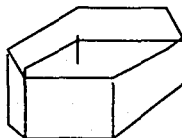


Fig. 6 Banda central de la esfera virtual

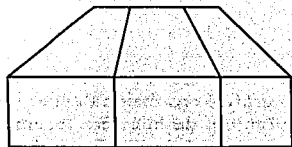


Fig. 7 Bandas central y superior de la esfera virtual

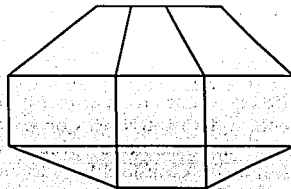


Fig. 8 La esfera virtual completa

Para hacer que el objeto sea aún más parecido a una esfera, todo lo que tenemos que hacer es crear más bandas de pequeños polígonos y aumentar el número de polígonos por banda. En el ejemplo anterior, tenemos un total de tres bandas, cada una con ocho polígonos. Ésta es la esfera que vamos a utilizar para nuestro juego de frontón debido a la necesidad de que el número total de polígonos no sea elevado. Las figuras 9 a 12 muestran varias esferas diferentes diseñadas utilizando diversas bandas con diferentes números de polígonos en cada banda.

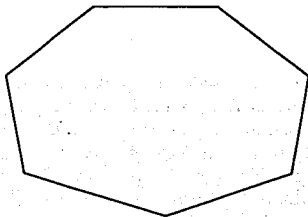


Fig. 9 Primera aproximación a la esfera

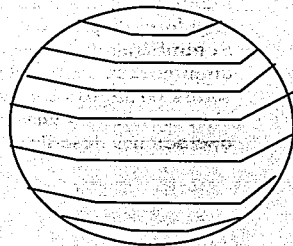


Fig. 10 Una esfera mejor

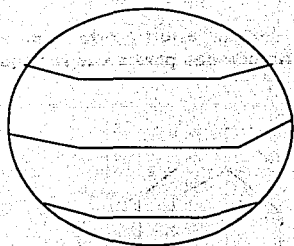


Fig. 11 Una esfera aún mejor

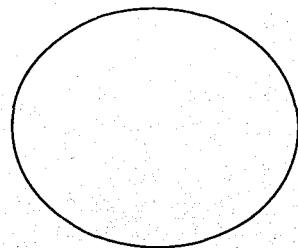


Fig. 12 Nuestra esfera final

LA RAQUETA.

El último componente del equipo que necesitamos construir es la raqueta. A primera vista, podría parecer que la raqueta es muy complicada de diseñar para un entorno virtual. Parece que tiene excesivos detalles. Básicamente, una raqueta es una elipse con un mango recto. De nuevo, podemos utilizar la idea de la señal de alto para aproximar la forma de la elipse. Si tomáramos los dos extremos y tiráramos de ellos, suponiendo que las barras centrales se pueden estirar, la forma resultante se parecería bastante a una raqueta. La figura 13 muestra el modelo.

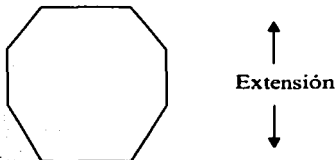


Fig. 13 Una señal de alto estirada

La señal de alto estirada será el armazón de nuestra raqueta. Como la raqueta se puede mover en todas direcciones, necesitamos diseñarla utilizando objetos que se puedan mover

en todas direcciones. El objeto básico para poder realizar este tipo de tarea es el cubo. Utilizaremos ocho cubos orientados en diferentes direcciones para crear la raqueta que se muestra en la figura 14.

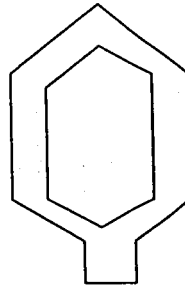


Fig. 14 Armazón de la raqueta

Con el armazón ya completo, tenemos que añadir las cuerdas. En nuestro entorno virtual, deseamos que las cosas parezcan tan reales como sea posible. Cuando tengamos la raqueta en la mano, desearemos ver sus cuerdas. Si creáramos las cuerdas utilizando un polígono simple (un polígono muy estrecho), sólo podríamos verlas por un lado. Tendríamos que utilizar un segundo polígono para el otro lado de la raqueta.

Del mismo modo, si miráramos a la raqueta desde arriba o desde abajo, parecería que no tiene cuerdas, debido a la forma en que los polígonos han sido dibujados. Necesitaríamos entonces cuatro polígonos para cada cuerda. El número de polígonos necesarios para las cuerdas sería superior al número de polígonos necesarios para los otros objetos.

Afortunadamente, hay una solución. REND386 tiene varias características que lo hacen especialmente útil para dibujar polígonos.

Si definimos un polígono de sólo un punto, el sistema no ejecutará ningún código correspondiente a polígonos; simplemente colocará un punto en la posición deseada. Si definimos un polígono con dos puntos (es decir, una línea), el sistema dibujará la línea donde debe estar.

La línea podrá verse desde cualquier dirección. Por tanto, podemos utilizar este tipo de polígonos para las cuerdas de nuestra raqueta. La raqueta terminada se muestra en la figura 15.

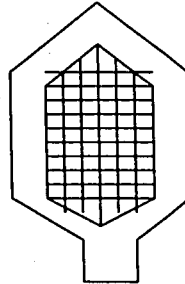


Fig. 15 La raqueta terminada

Ahora que ya hemos diseñado todo el equipo necesario para jugar al frontón, tenemos que añadir acción al sistema.

MOVIMIENTO DE LA PELOTA.

Como somos los únicos participantes en nuestro entorno virtual, necesitamos una forma de mover la pelota a nuestro alrededor. Sólo tenemos una mano, y se utiliza para manejar la raqueta, de modo que necesitaremos animar la bola por sí misma. Es decir, el sistema moverá la pelota automáticamente y nosotros intentaremos cambiar su trayectoria golpeándola.

Esto es todo lo que nos podemos acercar a un juego real de frontón, y nos da una idea de lo compleja que puede llegar a ser la simulación de la realidad.

ANIMACIÓN.

La animación de la pelota se realizará mediante la modificación de sus coordenadas x , y y z . Obviamente, si cambiamos el valor de la coordenada (x), la pelota se moverá de un lado a otro. El valor de (y) controlará la altura de la pelota sobre el suelo y el valor de (z) controlará el movimiento hacia adelante y hacia atrás. La pelota partirá de una posición predefinida. En nuestro caso, la definida por las coordenadas 1000,500,2000. El movimiento de la pelota será de tal forma que sus tres coordenadas tendrán valores positivos. Su movimiento real vendrá gobernado por tres variables globales, `inc_val_x`, `inc_val_y` e `inc_val_z`. El valor inicial de esas variables será de 50,40 y 40, respectivamente.

Cuando arranca el programa, el control de la pelota se realiza mediante una función denominada `move_ball`, que suma las variables de incremento en curso a las tres variables de posición, denominadas `x_position`, `y_position`, y `z_position`. La primera vez que se llama a la función, estas variables se incrementarán, partiendo de sus valores iniciales, para tomar los valores de 1050,540 y 2040, respectivamente.

Este incremento del valor de las variables continuará hasta que se alcance alguna de las condiciones que definen los límites. Las condiciones de los límites las definen las sentencias del tipo `if...elseif`. Veamos cómo son las condiciones en la dirección del eje (x), tiene la posibilidad de golpear contra la pared de la derecha o de la izquierda. Esto se puede expresar así:

```
if ( x_position > RIGHT_WALL )  
  {}  
else if ( x_position < LEFT_WALL )  
  {}
```

Las condiciones `RIGHT_WALL` y `LEFT_WALL` se definen 50 unidades hacia el interior desde la posición real de la pared. Las 50 unidades son para tener en cuenta el tamaño de la pared y de la pelota, de forma que no parezca que la pelota se introduce en la pared y vuelve a salir. Si se cumple alguna de esas condiciones, el valor de incremento se cambia de signo. Se realizan pruebas independientes de este tipo para las paredes, el techo y el suelo de la pista. En todos los casos, la pelota cambiará de dirección automáticamente. Aunque el movimiento de la pelota es bastante realista, se podría mejorar en el sentido de añadir el efecto de una gravedad simulada. En lugar de cambiar simplemente la dirección del movimiento y aplicar los valores de incremento a la posición de la pelota, se podría aplicar cierta gravedad a la misma. Necesitaríamos alguna manera de levantar la pelota del suelo.

En este estado del proyecto, tenemos una pista de frontón completa y una pelota que se mueve por su interior. Lo siguiente que necesitamos es ser capaces de movernos nosotros mismos por la pista y controlar la raqueta.

MOVIMIENTOS DEL CUERPO Y DEL PUNTO DE VISTA.

Debemos concretar cual va a ser el concepto de movimiento en nuestro entorno virtual para jugar al frontón. En el mundo real, podemos mover independientemente nuestros torsos, nuestros miembros y la cabeza. Podemos caminar, charlar y mover la cabeza, y todo ello al mismo tiempo. Sin embargo, cuando movemos nuestros torsos utilizando las piernas, el resto de nuestro cuerpo se mueve también. Nuestra cabeza no se queda charlando y moviéndose de un lado a otro en una esquina mientras el torso cruza la calle (aunque un mundo virtual así podría ser interesante). Necesitamos imitar los movimientos naturales del cuerpo en nuestro entorno virtual.

Empezamos por definir un cuerpo, sin objetos adosados hasta que añadamos la mano virtual utilizando un segmento. No olvidemos (capítulo 7) que un segmento es una estructura de datos que podemos utilizar para manejar objetos o como padre de otros segmentos. Es esta última función la que nos interesa. El segmento para nuestro cuerpo se define así:

```
SEGMENT*body_seg;
```

La variable `body_seg` es un puntero a un tipo de estructura definida por `SEGMENT`. Tras definir el segmento, necesitamos crear la estructura de datos real en memoria con la sentencia:

```
body_seg = newseg (NULL);
```

Ahora que tenemos un cuerpo, podemos empezar a añadir los elementos necesarios para lograr un movimiento completo en nuestro entorno virtual. Lo primero que necesitamos añadir es nuestra mano virtual. En el sistema REND386, podemos utilizar el Power Glove como dispositivo de entrada, pero una de las condiciones para utilizarlo es que se debe especificar algún archivo FIG para su representación virtual. El sistema incluye una mano virtual, denominada HANDSM.FIG, que es bastante realista. Hagamos saber a REND386 que deseamos utilizar HANDSM.FIG como representación de nuestra mano poniendo la variable `gpcursor` al valor de la cadena `handsm`. También tenemos que especificar qué driver está utilizando el sistema para controlar el Power Glove. El driver por defecto es el Power Glove. La variable `gpdname` se pone a `pglove`.

La inicialización del guante se realiza mediante la función `gloveptr_init`, que necesita varios parámetros y devuelve un puntero `PDRIVER` que utilizaremos cuando accedamos al guante. En el sistema se ha definido la variable `gd` como de tipo `PDRIVER` y también ha definido algunas otras que utilizan para salvaguardar valores. El aspecto de la llamada completa a la función es:

```
gd = manip_device = menu_device = gloveptr_init ( gpdname, 2*65536L,  
2*65536L, 2*65536L, 65536L, 65536L, 65536L);
```

Ya conocemos el cometido de la variable `gpdname` utilizada en la función. Los seis últimos parámetros son valores de escala para el dispositivo guante. Los tres primeros valores son los valores de escala para las componentes direccionales `x`, `y` y `z`.

Esto significa que por cada unidad que se mueva el guante, el sistema informará que la mano virtual se ha movido 2 unidades. Esto presenta una ventaja, debido a que los sensores de ultrasonidos del guante tienen una área de detección pequeña. Una vez que el guante ha sido inicializado, necesitamos cargar la representación virtual de la mano mediante la sentencia:

```
if (gd) load_glove_cursor (body_seg, manip_device, gpcursor);
```

Nótese que le estamos proporcionando a la función el segmento que define nuestro cuerpo, `body_seg`. El sistema creará automáticamente dos segmentos hijos de `body_seg`. `Wrist_seg` es un descendiente directo de `body_seg`, y `hand_seg` es un descendiente directo de `wrist_seg`. Esto quiere decir que cualquier acción que se realice sobre el segmento `body_seg` se pasará automáticamente a sus hijos `wrist_seg` y `hand_seg`. Los objetos reales en el archivo `HANDSM.FIG` se encuentran adscritos al segmento `hand_seg`. Lo que todo esto significa es que controlamos la mano virtual utilizando tres segmentos. Si movemos nuestros cuerpos utilizando `body_seg`, las manos virtuales seguirán de forma natural este movimiento del cuerpo, ya que el segmento `body_seg` es su padre. Pero también es posible mover la mano virtual utilizando `wrist_seg` sin mover nuestro cuerpo. Es una forma muy natural de simular los movimientos de la mano.

La segunda tarea en la construcción de nuestro cuerpo es dotarlo de ojos. El sistema `REND386` crea, o presenta, las imágenes basándose en el lugar a donde están mirando nuestros ojos: el punto de vista. Utilizaremos una variable denominada `current_view` para controlar esto. El punto de vista no se encuentra ligado al cuerpo como lo está la mano, de forma que tendremos que moverlo independientemente del cuerpo, pero siempre a la vez. Pongamos ahora todo en posición.

Empezaremos situando nuestro cuerpo de forma que tengamos una buena panorámica de la pista. La posición adecuada es la definida por las coordenadas 2250, 800, -300, que sitúa el centro de nuestro cuerpo en la mitad de la pista, en la parte de atrás y aproximadamente a 1 metro del suelo (la altura correcta para una persona promedio). Nótese que la mano virtual se moverá a una posición relativa al segmento `body_seg`. Las sentencias son:

```
abs_move_segment ( body_seg, 2250,800,-300 );  
update_segment ( body_seg );
```

Seguidamente movemos para golpear la pelota. Pero todavía no tenemos control sobre la raqueta.

INTERFAZ DE MOVIMIENTOS.

Ahora que tenemos nuestro cuerpo y el punto de vista en las posiciones iniciales adecuadas, necesitamos una forma de movernos por la pista. La forma más lógica de movernos es utilizar un joystick. El sistema REND386 nos permite utilizar el joystick, ya que proporciona las funciones necesarias para localizar el joystick y leer valores del mismo. Comenzaremos por la localización. La sentencia:

```
if ((joy_return = joystick_check()))
```

Localizará el joystick en el sistema si es que existe uno. Si `joy_return` tiene el valor de 0, el sistema no ha sido capaz de localizar un joystick en la computadora. Si este es el caso, presentaremos como salida una simple sentencia que indique que no hemos sido capaces de encontrarlo, lo que quiere decir que tendremos que utilizar el teclado para introducir los movimientos. Si `joy_return` no vale 0, tiene que valer 1 o 2. Un valor de 1 indica que hay un joystick conectado al puerto 0, y un valor de 2 indica que el joystick está en el puerto 1. Utilizando estos datos, podemos inicializar el joystick mediante la sentencia:

```
joystick_init (&joy_data, 0) o joystick_init(&joy_data, 1)
```

EL BUCLE DE CONTROL.

Seguidamente necesitamos configurar el sistema para los movimientos. Tenemos cuatro posibles movimientos: hacia adelante, hacia atrás, a la derecha y a la izquierda. Además, tenemos que ser capaces de mover nuestro punto de vista 90° a la izquierda. Esto nos permitirá mirar a la pista hacia adelante, mantener nuestros ojos fijos en la pelota y mover

nuestro cuerpo. El movimiento del cuerpo se controlará mediante el joystick o las flechas del movimiento del teclado.

Para determinar si el sistema necesita precisar un movimiento, se establece un lazo de control que observa el teclado y el joystick. Si se pulsa una tecla o se mueve el joystick, el sistema reaccionará. El aspecto del bucle es éste:

```
while ( ejecutando )
{
    if (bioskey (1)) handle_key(getkey());
    if (joystick_return) check_joystick();
    refresh_display();
}
```

EL TECLADO.

El bucle de control se ejecutará mientras la variable ejecutando no valga 0. El bucle comienza comprobando si se ha pulsado o no una tecla. Si no se ha pulsado, no necesitamos ningún proceso posterior. Si se ha pulsado, hay que procesar la tecla y realizar las acciones que se supone que la pulsación de la tecla va a originar.

El proceso se realiza en la función denominada handle_key. El único parámetro necesario para esta función es el valor de la tecla pulsada. La función handle_key consiste en una única sentencia switch para determinar a qué teclas estamos reaccionando.

```
Switch ( c )
{
    case 'q':
    case UP
    etc.....
}
```

Una de las teclas más importantes para nuestra respuesta es la tecla Q. Pulsarla es la única forma de salir del programa. En este caso, el sistema pondrá a 0 la variable ejecutando y devolverá el control al bucle principal.

En cuanto esto suceda, la condición del bucle será falsa, y el programa finalizará. Las otras teclas a las que queremos responder son los movimientos de las flechas. Estas teclas tienen asignado los siguientes valores numéricos:

```
UP      0x4800
DOWN    0x5000
LEFT    0x4B00
RIGHT   0x4D00
```

Hemos utilizado la sentencia `define` de C para definir estas constantes al comienzo del programa. Cuando se pulsa cualquiera de las teclas, necesitamos mover nuestro cuerpo y el punto de vista en la dirección adecuada. Veamos la acción de las teclas.

UP Y DOWN (ARRIBA Y ABAJO).

Cuando el usuario pulsa la tecla de flecha UP, nuestro cuerpo y el punto de vista se moverán hacia adelante en el entorno virtual. Para ello, debería bastar con incrementar la componente z del segmento del cuerpo y del punto de vista. Sin embargo, REND386 está basado en un sistema de coordenadas polares, de forma que el sistema de incrementar sólo la coordenada z no funcionará. El código para realizar el movimiento hacia adelante es éste:

```
x = 150 * sine(current_view ->pan);
y = 150 * cosine(current_view ->pan);
current_view ->ex += x;
current_view ->ez += z;
rel_move_segment ( body_seg, x,0,z);
update_segment ( body_seg );
```

En la programación de los sistemas de realidad virtual siempre se deben tener cuidado con los cálculos extras que deben realizar los programas.

Por ejemplo, los cálculos para el movimiento hacia adelante se basan en la variable `pan` del punto de vista, y los valores de los cálculos se almacenan en variables temporales. En otras palabras, deseamos movernos hacia adelante basándonos en lo que estamos mirando actualmente. Sería muy difícil navegar por un mundo virtual si camináramos hacia adelante sin tener en cuenta hacia adonde estamos mirando.

En el mundo real, casi siempre caminamos en la dirección de nuestra línea de visión. Nótese que estamos poniendo en nuestros cálculos una escala de 150. De este modo, nuestra zancada virtual se incrementa desde aproximadamente 1 milímetro a aproximadamente 15 cm. Si aumentáramos este valor, nuestros movimientos en el entorno virtual serían todavía más rápidos. Sin embargo a medida que aumentamos la velocidad de los movimientos, la escena va apareciendo más discontinuamente.

Deseamos escoger un paso natural que haga parecer que las imágenes se muevan lentamente. El movimiento hacia atrás se realiza simplemente cambiando el signo del valor de escala de 150 a -150.

LEFT Y RIGHT (IZQUIERDA Y DERECHA).

El movimiento a izquierda y derecha está de nuevo referido a nuestro punto de vista. Para ver la importancia de determinar hacia donde estamos mirando, quedémonos de pie mirando en una dirección y movámonos hacia la izquierda. Fijémonos cómo nos movemos. Giremos 180° y volvamos a movernos hacia nuestra izquierda. Nos estamos moviendo en dirección contraria a la primera vez. Si no tuviéramos en cuenta la dirección hacia donde miramos, al movernos hacia la izquierda lo haríamos siempre en la misma dirección, lo que sería muy confuso. En relación con la computadora, el movimiento a la izquierda o a la derecha es muy semejante al movimiento hacia adelante o hacia atrás. Para moverse a la izquierda, los valores calculados de x y z serían:

```
x = -150 * cosine(current_view ->pan);  
y = 150 * sine(current_view ->pan);
```

Nótese que ahora no sólo los signos han cambiado, sino que el valor de x utiliza la función coseno, a diferencia de la función seno que utilizaba en el movimiento hacia adelante y hacia atrás. Para moverse hacia la derecha, basta con cambiar los signos de los valores de escala.

GIRO DE LA CABEZA.

Una última tecla que tenemos que considerar es la tecla Y, que nos permitirá girar nuestra cabeza en incrementos de 90°. Si alguna vez hemos deseado mirar detrás de nosotros sin girar nuestro cuerpo, ahora es nuestra oportunidad. Utilizamos la variable pan de nuestro punto de vista utilizada en el código presentado anteriormente. Para cambiar la posición de la cabeza con respecto a esa variable, simplemente sumamos o restamos un valor en grados:

```
current_view ->pan += 90*65536L
```

Esta sentencia hará que nuestro punto de vista cambie 90°. El valor de la multiplicación 65536L es un requerimiento del esquema numérico utilizado por REND386 para acelerar

los cálculos aritméticos. Poniendo esta sentencia en nuestro switch para conmutar en función de los valores del teclado, podemos girar nuestra cabeza, orientándola hacia la parte frontal de la pista mientras la raqueta permanece en posición para golpear la pelota, el último paso es coger la raqueta y golpear la pelota.

DETECCIÓN DE COLISIONES.

Al principio, la raqueta está flotando en el aire frente a la pista y la pelota se mueve al rededor. El juego comienza cuando el usuario toma la raqueta con su mano virtual y comienza a golpear la pelota. Para tomar la raqueta con la mano virtual, hay que detectar la colisión entre los dos objetos. Cuando el usuario ha tomado la raqueta en sus manos, tenemos que comprobar constantemente si se produce una colisión entre la raqueta y la pelota. Para hacer esto, utilizaremos un método simple.

Una función denominada `sphere_pretest` calcula la distancia entre cualquier objeto y un punto en una posición determinada por sus coordenadas (x), (y) y (z). Si queremos determinar si ha tenido lugar una colisión entre dos objetos, suministraremos a la función un puntero a uno de los objetos y punto central del otro objeto: el punto central del objeto se puede obtener utilizando la función `get_object_bounds`. En nuestro caso, encontraremos la posición en curso de la pelota y suministraremos la raqueta como objeto a la función `sphere_pretest`. La secuencia es:

```
get_onject_bounds ( ball_object, &x, &y, &z );  
dist = sphere_pretest ( raquet_object, x, y, z );
```

Una vez que esas dos sentencias se han ejecutado, la variable `dist` contendrá la distancia en unidades entre los dos objetos. Para determinar si están colisionando, hay que tener en cuenta el tamaño de `dist` y de los propios objetos. Como cada objeto que estamos comprobando tiene un tamaño determinado en unidades, necesitamos especificar una distancia de colisión que tenga en cuenta esos tamaños.

En el juego de frontón tenemos que decidir entre dos tipos distintos de colisiones. El primer tipo es entre la mano virtual y la raqueta, y el segundo es entre la raqueta y la pelota. Lógicamente, no hay que comprobar la colisión entre la raqueta y la pelota hasta que no haya ocurrido una colisión entre la raqueta y la mano. Para controlar qué tipo de colisión es la que está activa, utilizamos una variable denominada `NO_RACQUET`. Si `NO_RACQUET` no es 0, no tenemos en posición la raqueta, y hay que comprobar si se produce una colisión entre la raqueta y la mano. En cada iteración del bucle principal se determinará si ha tenido lugar esa colisión o no. La sentencia que utilizamos para decidir si se ha producido una colisión entre la raqueta y la mano tiene dos partes.

La primera es que la distancia de colisión debe ser inferior a 100 unidades, lo que implica que la mano debe estar muy cercana a la raqueta. La segunda parte de la sentencia de colisión se basa en los dedos del guante. La interfaz para el Power Glove que proporciona REND386 tiene la posibilidad de detectar algunas posturas simples de la mano. La postura que toma el guante al tomar la raqueta con el puño cerrado. Cuando la mano se encuentra con el puño cerrado y lo suficientemente cerca de la raqueta, ocurre una colisión, con lo que la raqueta se convertirá en hija de la mano virtual y se moverá siempre en consonancia con la mano. Como se ha producido una colisión con la raqueta, la variable NO_RACQUET se pondrá a 0.

Cuando la variable NO_RACQUET vale 0, el bucle principal busca una colisión entre la raqueta y la pelota. La idea es básicamente la misma que con la raqueta y la mano, excepto que la distancia de colisión se incrementa hasta 300 unidades, para tener una mejor oportunidad de golpear la pelota. Cuando ocurre una colisión entre la raqueta y la pelota, la pelota debe reaccionar al golpe moviéndose en la dirección opuesta.

Para conseguirlo, determinamos en qué dirección se estaba moviendo la pelota cuando recibió el golpe (es decir, si su dirección era positiva o negativa) e invertimos el sentido del movimiento modificando el valor de la variable inc_val_z. Esta variable se utiliza para seguir los movimientos de la pelota. Además añadimos algo de velocidad a la pelota poniendo inc_val_z a cinco veces su velocidad normal de 10. Seguidamente, cada vez que se mueve la pelota, su velocidad va disminuyendo hasta volver a alcanzar el valor de 10. De ese modo se crea la sensación de que la pelota ha sido golpeada realmente.

PARTE VI. ANALISIS DE RESULTADOS.

CAPITULO VIII.

VENTAJAS Y DESVENTAJAS DE LA REALIDAD VIRTUAL

Prototipado/Modelado: Las aplicaciones de prototipado son aquellas en las que se utiliza la realidad virtual para establecer un modelo de cómo sería un determinado desarrollo que aún está en sus etapas más tempranas. Esto es muy útil en áreas como las del diseño industrial (por ejemplo, podemos pasearnos en el último modelo de todo terreno antes de que se haya realizado sobre él una sola soldadura), arquitectura, planificación de operaciones (por ejemplo misiones en el espacio o lugares de alta inaccesibilidad), o sencillamente podemos realizar paseos virtuales sobre terrenos existentes o inexistentes.

Entrenamiento de Pilotos: Una de las aplicaciones más populares es el entrenamiento de pilotos mediante técnicas de realidad virtual o simulada para responder ante situaciones reales o de practica de vuelo que podrían costar vidas y materiales costosas, se utilizan plataformas enteras de simulación donde el factor más importante es el envolvente de vuelo o el que se ve más sometido el piloto, creando un perfecto entorno así como texturas de todo tipo, que pueden incluso llegar a crear psicológicamente hablando sensaciones de todo tipo a el piloto y demostrar su valía para el puesto en el que se le prepara.

Entrenamiento de Médicos: Otra aplicación interesante para el entrenamiento y perfeccionamiento de especialistas en operaciones laparoscópicas. Aquí se utiliza una estación de trabajo Silicon graphics que recibe los datos de las herramientas que maneja el médico, iguales a las que utilizaría en una operación real, para procesarlos y generar una imagen fotorealista en un monitor de forma que nada lo distinga de una operación real. Este sistema tiene un gran interés, ya que es mucho más barato formar futuros especialistas de esta manera que con operaciones reales, además de permitir que muchas más personas aprendan o mejoren sus habilidades, ya que solamente es cuestión de adquirir más máquinas que pueden funcionar con turnos mucho más flexibles que las operaciones reales.

Aplicaciones Metafóricas: estas aplicaciones pretenden utilizar la RV para dotarnos de una representación visual que aumente nuestra capacidad de comprensión sobre un determinado proceso. Las aplicaciones metafóricas típicas suelen ir encaminadas a la representación de

datos científicos (por ejemplo, el túnel de viento virtual desarrollado en la NASA Ames Center, que representa los flujos de aire sobre un determinado modelo virtual), pero no

están limitadas a este campo (por ejemplo, Maxus System International desarrolló en 1992 un sistema de este tipo que se empleaba para manipular carteras de valores de bolsa).

Simulación de Estructuras: Otra de las aplicaciones tanto en diseño como comercialmente hablando, son los diseños virtuales con simulación perfecta de estructuras externas e internas que ayudan a el arquitecto a crear el edificio (o vivienda) perfectos, valorando los espacios muertos y las memorias de calidades de su proyecto, así como igualmente es aplicable para fines comerciales, aunque eso sí quizás demasiado costoso por el momento para este último, ya que la simulación virtual requiere potentes máquinas y periféricos para ser tan perfecta como para recibir este nombre.

Aplicaciones en minusvalías físicas: Es uno de los principales proyectos futuristas aunque mucho más costosos, ya que su diseño tendría que incluir, sobre todo en el caso de las personas invidentes, unos sensores especiales, que adaptados, conseguirían una visión simulada del terreno permitiendo dotar de visión (en este caso) a estas personas, e incluso en algunos casos, dotar de facultades superiores a las humanas mediante esta realidad simulada y real al mismo tiempo.

Teleoperación y tele-robótica: Esta área está fuertemente ligada a los sistemas de telepresencia y de realidad aumentada, y se emplea en ámbitos como la programación de robots (programar movimientos sobre un robot virtual que reproducirá un robot real), los sistemas de exploración sobre terrenos inaccesibles o en las cada vez más cercanas aplicaciones de telemedicina.

Aplicaciones de Entrenamiento táctico y militar terrestre: sobre todo en Norteamérica, grupos especiales de asalta, y grupos militarmente especialmente preparados, usan entrenamientos simulados virtualmente ambientados de modo impresionante donde se le obliga a el soldado a responder y actuar en situaciones críticas del modo correcto, preparándole a actuar en segundos e incluso décimas en estas situaciones. Estas situaciones son creadas o ambientadas de un modo diferente a los propuestos anteriormente, ya que en esta realidad virtual, se provee de panorámicas y escenarios montados previamente, y con sonido realista dependiendo de la situación, pero es, como no un entrenamiento simulado, un entrenamiento virtual.

Aplicaciones de Ocio: cada vez es más frecuente el uso de sistemas inmersivos en parques de atracciones, cada vez los videojuegos resultan más complejos y las empresas que los desarrollan más poderosas, es cuestión de tiempo el que estas se lancen a los sistemas de Realidad Virtual y que estas lleguen a nuestras casas.

El cortador de césped, la película de Stephen King interpretada por Pierce Brosnan, (surgió a principios de los 90) gira en torno a una exploración sobre la realidad virtual y la experimentación en seres humanos. Se puede observar una silla flogiston. Esta silla, que proporciona una postura de tensión mínima, incorpora una almohadilla sensible al calor y paneles de control. Figura 1. Esta película que abordaba el tema de RV, la sociedad hizo eco de esta nueva tecnología que incluso idealizaba la posibilidad de hacer realidad cualquier sueño, al menos de forma sintética.



figura 1

ESCRITORIOS VIRTUALES

Independientemente de lo grande que sea una pantalla de computadora, siempre se acaba el espacio. La introducción de demasiadas ventanillas sólo crea un desorden de la información, como el que podemos encontrar encima de una mesa; incluso peor, ya que está todo condensado en un espacio mucho menor.

Un escritorio virtual del mismo tamaño que uno real reemplaza al escritorio real. Su gran pantalla permitirá a los usuarios explorar las posibilidades de un mundo de información electrónica, ayudado de un guante y un casco estereoscópicos. La nueva oficina podría posiblemente estar equipada con tales superficies de trabajo, con pantallas de televisión de alta definición (HDTV por High-definition TeleVision) de una anchura de 4 pies y una profundidad de alrededor de 2 pies. Como en estos tiempos es posible una resolución de 100 puntos por pulgada (ppp), se puede disponer de alrededor de 13 millones de píxeles y con esta resolución no se cansaría la vista.

Igual que se elaboran paralelamente informaciones y proyectos diversos en el escritorio real, se podría entrar y trabajar en cuatro o cinco proyectos simultáneamente por medio de ventanas virtuales. Los proyectos de diseño en equipo podrían ser trabajados al mismo tiempo y comparados. Las tendencias y cifras financieras podrían ser conectadas en tiempo real de tal forma que el flujo del mercado cambiase antes que los ojos, proporcionando un margen competitivo para las tendencias dependientes del momento.

Los registros literarios y los videojuegos serían posibles desde otros puntos de acceso, y las ventanillas podrían verse extendidas, archivadas o reorganizadas más rápidamente que los papeles de trabajo de una mesa real. Con la utilización del escritorio virtual, el guante virtual y el casco estereoscópicos se podría participar desde cualquier punto donde sé este de una sesión virtual en donde el intercambio de ideas se daría entre el jefe los clientes y los compañeros de oficina figura 2.

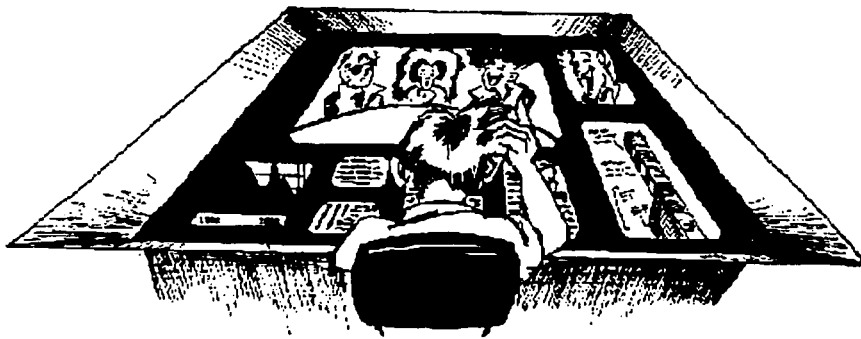


figura 2

CONOCIMIENTOS EJECUTIVOS Y MODELOS DE CORPORACIÓN

Para tener éxito en los negocios, los ejecutivos competentes están al tanto del impacto en las corporaciones de las nuevas tecnologías, mercados o inversiones de capital. Tienen habilidades para evaluar los factores de riesgo, pero necesitan tener a su disposición unas herramientas y un entrenamiento que les proporcione un margen competitivo de información.

Para el proceso de toma de decisiones se ve a menudo influido por la estructura de los modelos utilizados para el análisis y la información. Por tanto, los ejecutivos deben asegurarse que el sistema de información utilizado representa con precisión a la compañía como "ellos" la ven, no como la ve el departamento de procesamiento de datos. Los creadores de decisiones ejecutivas necesitan facilidades que les permitan visualizar la información relevante y utilizarla con rapidez.

Deben de ser capaces de ensayar estrategias a la luz de los desafíos de la vida real y de contemplar las soluciones posibles y sus consecuencias en la organización. Los modelos de simulación deben incluir modelos de la historia y la cultura de la compañía, es decir lo que funciona ahora y lo que funcionó en el pasado, para que el ejecutivo pueda anticipar lo que va a funcionar en el futuro.

Esto supone procesar millones de datos relevantes para la situación de la compañía, aprovechar los conocimientos acumulados y la capacidad de aprendizaje de la organización y, por último, visualizar la información de formas significativas.

Estas son las áreas en las que las técnicas de la RV de exhibición y de interfaces pueden ser más aprovechadas. La visualización debe facilitar el análisis de aquellos factores que más puedan afectar a la empresa, y el software debe proporcionar al ejecutivo los medios para manipular y cuestionar los datos.

Al proporcionar una herramienta para una conceptualización a alto nivel y un entorno virtual en el que la manipulación de las ideas es fácil, la tecnología RV respalda las decisiones intuitivas informadas. Por ejemplo, se puede reservar una ventanilla para una representación gráfica variable de las cotizaciones de las acciones, las tendencias de mercado, el inventario, los intereses de socios, los movimientos competitivos de otras compañías, etc.

Todos los que trabajan en los conocimientos corporativos pueden seleccionar visualizaciones de estas gráficas, mejoradas por la perspicacia del ejecutivo. Así, todos los empleados de la organización pueden compartir y aprovechar la intuición, y se pueden difundir innovaciones figura 3.

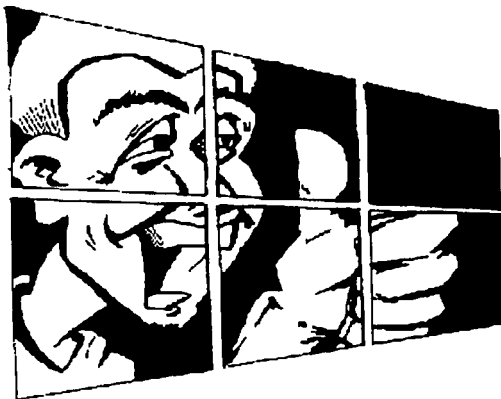


figura 3

MERCADO MEJORADO

EL mercado corporativo se aprovecha de la tecnología RV para analizar archivos masivos de datos comerciales (como los recolectados por las compañías de seguros y bancos) y para visualizar realizaciones interpretativas de los mismos. El personal de mercado puede determinar mejor la estrategia más rentable obtener mejores resultados gracias a la capacidad de visualizar el mercado, cuantificar sus riesgos y manipularlo de acuerdo con diferentes aproximaciones. Las visualizaciones dinámicas de la información del mercado y los sistemas de previsión incrementarán las posibilidades de éxito en la introducción de nuevos productos figura 4.

Las hojas electrónicas visualizadas, que están conectadas a las bases de datos en tiempo real, estimulan la acción oportuna de las personas responsables de las decisiones financieras. Los programas de entrenamiento fabricados para dirección y asesoría técnica proporcionarán a los usuarios situaciones verosímiles en las que puedan ser desafiados, de acuerdo con sus propios estilos de aprendizaje y sus niveles de habilidad. La experiencia virtual en el trabajo se puede obtener sin la barrera de decisiones fallidas o errores de procedimiento. Eventualmente, todas las computadoras que procesan imágenes de alta resolución, utilizadas en las aplicaciones descritas, deberían ser provistas de sistemas para

copiar, teleproductores de imágenes, lectores de texto o escánares. Esto afectará a un montón de aplicaciones en la oficina y elevará el apoyo a los esfuerzos del mercado.

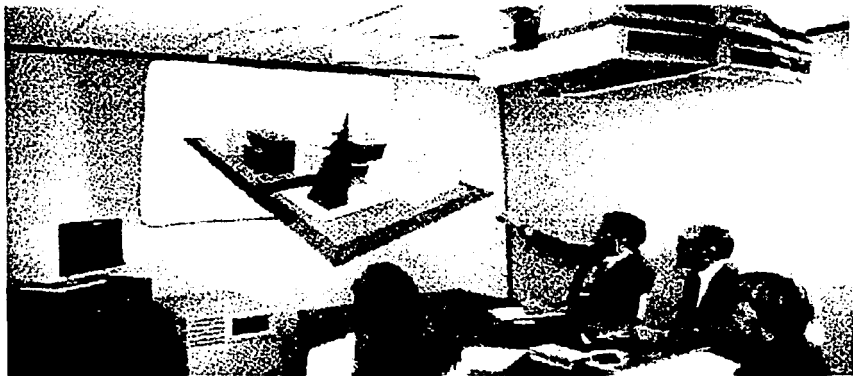


figura 4

FACTORES HUMANOS EN AMBIENTES VIRTUALES

Pero esta tecnología desgraciadamente tiene graves efectos secundarios. En el Imax, Por ejemplo, las películas que se visualizan no llegan a cuarenta minutos porque hay personas que se marean, aparecen extraños efectos visuales en la vista, pierden el equilibrio, tienen dolor de cabeza, de estomago e incluso vómitos. Esto sucede tanto en películas de Inmersión como en los cascos con gafas 3D. Y todo ello por la limitación de nuestros organismos.

En la Realidad Virtual el principal problema es que se puede engañar al cerebro y hundirlo en un mundo virtual imaginario, pero no al oído interno. Este órgano es el encargado de hacernos mantener el equilibrio así como el que nos indica si nos movemos, nos agachamos, si la información que llega a través de los ojos nos dice que estamos avanzando

pero el oído interno dice que estamos parados, se produce un trastorno que se presenta con efectos como los comentados anteriormente.

Pero todos los inconvenientes se superarán con el tiempo, y una tecnología que nos abre las puertas a nuevos mundos y nuevas formas de investigación, nos deleitara con aplicaciones que cambiarán nuestra forma de vivir en este siglo que empieza. Pero hay efectos que

seguirán sin conocerse hasta pasado un largo plazo y no existen animales de laboratorio en los que probar efectos psicológicos. Habrá que seguir experimentando con humanos.

Incluyen factores de salud como el vértigo por el resplandor el cual puede incluir un ataque, daño auditorio y el oído interno causado por el audio de alto volumen, movimientos prolongados repetitivos los cuales causan heridas por el sobreuso y daños de cabeza, del cuello o espina vertebral debido al peso o posición de los cascos. Los factores de seguridad también necesitan ser considerados. Las manos y los brazos pudieran ser pinchados o sobre extendidos si un dispositivo de retroalimentación falla, el usuario podría ser desorientado dañado si falla la computadora y repentinamente manda al usuario a la realidad, interrumpiendo el sentido d presencia.

No cabe duda que la electrónica y las nuevas herramientas con las que cuenta la computación harán en el próximo siglo un mundo con más esperanza, más humano, ya que el acceso al conocimiento sería más fácil y rápido y por ende una educación personalizada, eficiente, clara, efectiva y dinámica.

Todo lleva consigo un riesgo; ejemplo, el fomento a la violencia, escape de la realidad, pornografía, proliferación de grupos extremistas, juegos enajenables. Pero el ser humano deberá adaptarse y basándose en su capacidad minimizar o desaparecer esas influencias negativas para bien de todos.

PARTE VI. REPORTE FINAL.

CONCLUSIONES.

Se estudio en profundidad el guante, su modo de funcionamiento y de la disponibilidad de interfaces hardware y software con la computadora, por lo tanto, en resumen de nuestro experimento podemos sacar las siguientes conclusiones:

- 1.- se adapto el guante de un periférico del sistema de entretenimiento doméstico Nintendo. Para que con este dispositivo, la computadora pueda determinar dónde está la mano del usuario en un espacio tridimensional, si el puño está cerrado o no y si la mano se está girando o no.
- 2.- se determino que conexiones del cable del guante se conectan con los pines del conector (descrito en la parte IV capítulo 6 inciso 6.4) así como el poder utilizar su cable original (inciso 6.5).
- 3.- se realizo la conexión del guante al puerto paralelo, utilizando un conector Macho DB-25 (descrito en el inciso 6.7).
- 4.- se construyo una fuente de alimentación externa (descrito en el inciso 6.9).
- 5.- Se utilizó un sistema de posicionamiento por ultrasonido que consta de dos transmisores localizados en la parte superior del guante y tres receptores en forma de triángulo, este sistema tiene la ventaja de que ya están integrados al guante, se puede utilizar el cable original, la lectura del guante es más rápida y sencilla de programar (inciso 6.21).
- 6.- Se utilizo código del modo de alta resolución para la creación de la interfaz, es decir se realizo la conversión del guante mediante un ligero recableado para conectarlo al puerto paralelo de la computadora. Se pudo haber usado el puerto serial pero es más conveniente y fácil de programar el método de conexión al puerto paralelo, la segunda , la utilización del software (descrito en el inciso 6.18).
- 7.- Se utilizó el software de presentación REND386 para crear una aplicación: de un frontón virtual por varias razones: Podemos ver la teoría de la realidad virtual llevada a la práctica. El número de poligonos necesario para construir el frontón virtual es pequeño, de forma que la velocidad del juego será alta, incluso en computadoras no muy rápidas. Podrá utilizar la mayoría de los proyectos hardware que se puedan construir.

8.- El juego de frontón simulado es muy parecido al frontón real. Esto significan varias cosas:

Necesitamos contar con varios elementos del juego:

- * Raqueta
- * Pelota
- * Un campo de juego o pista

Podemos manejar la raqueta y golpear la pelota. El juego proporcionar una sensación real de profundidad, no sólo utiliza gráficos.

9.- Se analizó cada componente por separado. En primer lugar, la creación de los elementos del juego es simple. Sólo hay que diseñarlos sobre papel y crearlos con el software. La pelota es un elemento de particular interés, ya que nuestros bloques básicos de diseño son polígonos. Tenemos que determinar cuántos polígonos son necesarios para crear este objeto esférico.

En segundo lugar, tenemos que considerar nuestra movilidad en el mundo virtual. En el mundo real manejamos la raqueta con nuestras manos, que están unidas a nuestro cuerpo. Hemos mostrado dos de los elementos básicos de la RV "la manipulación de objetos", y "la navegación la forma como nos movemos en el juego virtual.

Finalmente, llegamos a la cuestión de la tridimensionalidad. Durante el juego del frontón, se gasta una buena parte del tiempo en utilizar la percepción de profundidad para determinar dónde está la raqueta en relación con nosotros mismos. Esto será muy importante en nuestro juego virtual.

10.- La pista del juego de frontón se construyó utilizando el sistema REND386 descrito en el capítulo 7. Se utilizó la línea de compiladores Turbo C/Borland. Para poder recompilar el código.

11.- La RV no es solamente un instrumento de entretenimiento y de diversión. Se puede también aplicar como herramienta de trabajo en empresas Públicas y Privadas.

12.- El guante podría sustituir los comandos de los Sistemas Operativos actuales ejemplo (windows), por movimientos reales, es decir en lugar teclear "Control-C, para copiar" y luego "Control-V, para pegar" sería más sencillo si con el guante se toma de una carpeta, dispositivo físico o lógico y se suelta o deposita en otro.

13.- Se esta consciente de que el costo para algunos dispositivos (guantes, cascos, trajes, etc, más sofisticados) puede ser muy alto, ya que en la actualidad no están muy comercializados, al igual que el software, pero esperamos que creando y distribuyendo el

software por internet, en libros, revistas, se abaraten los costos de producción de hardware y poder adquirir periféricos virtuales menos caros, partiendo de que a mayor volumen de producción, reducción en costos (principalmente los fijos) y por ende mejores alternativas de precio, y por otra parte a mayor oferta menor precio.

14.- La Realidad Virtual es una nueva generación de la informática donde conjuntada con las bases de datos, nos permitirá tomar mejores decisiones, sin basarnos en simples suposiciones como se han tomado hasta ahora. Esto será posible al facilitar la forma de acceder a los datos.

15.- Para las grandes empresas la RV sería una nueva forma de comercializar sus productos sin tener que invertir en importaciones o exportaciones para que conozcan sus productos o servicios, aunque parezca que al invertir en cualquier proyecto para RV el costo es muy alto. La verdad es que el beneficio alcanzado a largo plazo es muy grande.

16.- Cada día se difunde más el uso de la RV las compañías de emisiones por cable están acoplándose a redes de telecomunicaciones, los productores de cine están invitando a expertos en computadoras a desarrollar efectos especiales. En los hogares se está entrando a una nueva variedad de facilidades en el entretenimiento y en las compras, se están haciendo accesibles los variados servicios "virtualmente realistas", (El Internet).

Entonces podemos decir que la Realidad Virtual no es una Útopía es nuestro presente, empezó con un juego casero y se convirtió en una rama de la informática, desde una pequeña aplicación de un sueño hasta un complejo sistema de pilotaje espacial. En estos tiempos se habla de sexo virtual, entonces podemos hablar de la Realidad Virtual como una materia de estudio.

TEMAS PENDIENTES PARA FUTURAS INVESTIGACIONES.

La tecnología de la Realidad Virtual, puede mejorar mediante investigaciones sobre los siguientes temas:

Diseñar nuevos Lenguajes de programación para Sistemas Virtuales.

Habrà que reescribir muchas de las analogías que se utilizan en los Sistemas Operativos actuales y que todos damos por válidas (por ejemplo que el "cortar" y "pegar" es algo que se realiza sobre el plano mientras que tal vez sobre el espacio deberíamos empezar a hablar de "coger" (tomar) y "dejar").

Una cosa es el posicionarse en un escenario y otra bien distinta es poder interactuar con los propios objetos. Obtener una información sobre el tacto, dureza o peso de los objetos nos podría ayudar a realizar sobre el mundo virtual las mismas acciones que realizamos sobre nuestro mundo real y que nos resultan evidentes, de forma más natural.

Con respecto a las unidades de visualización RV, hay muchos problemas por solucionar. Entre los citados están:

- Baja calidad de imagen de pequeños LCDs (es decir, baja resolución);
- Coste y disponibilidad de pequeños sistemas de CRTs;
- Eliminación del retraso entre el movimiento del usuario y la respuesta del sistema;
- Amplio campo de visión en estéreo que está sobrepuesto a la visión del mundo real;
- Superposición de objetos virtuales en el mundo real de tal manera que tenga sentido para el sistema visual humano;
- Lucha entre confort e incomodidad de los aparatos que el usuario de la RV lleva en la cabeza y en el cuerpo;
- Ejecución de los sistemas operativos en tiempo no real;
- Habilidad para modelar mundos virtuales;
- Generación de imágenes para escenas complejas.

El concepto de "comunidad virtual" empieza a tomar importancia, tal vez en un futuro bastante más cercano de lo que nos creemos, la vídeo conferencia sea sustituida por salas virtuales en las que los personajes que representan a los usuarios del mundo real charlen e intercambien información de una forma tan natural como si todos ellos estuviesen presentes en la misma estancia.

VENTAJAS Y DESVENTAJAS DE NUESTRO ENFOQUE.

VENTAJAS	DESVENTAJAS
El guante se diseño para sustituir al mouse y al joystick.	El guante se diseño para el juego de video Nintendo no para la computadora.
El guante se diseño para versiones de alta Resolución, este modo permite al guante responder a movimientos x,y y z, a movimientos de los dedos y al poder girar la mano.	El guante resulta ser muy pesado y engorroso para utilizarlo como joystick.

VENTAJAS	DESVENTAJAS
Se puede desarrollar cualquier tipo de software para este dispositivo.	Desafortunadamente sin hardware ni software de apoyo el dispositivo no sirve.
El software REND386 tiene varios cientos de rutinas en C de Borland que permiten la creación, visualización y manipulación de objetos tridimensionales.	La necesidad de utilizar una fuente externa de alimentación de energía para el guante.
La utilización del formato de objetos PLG que es una manera de representar objetos tridimensionales utilizando números.	Problemas entre el guante y la conexión, ya que se esta conectando un producto comercial y una interfaz realizada por uno mismo.
El guante puede realizar los movimientos básicos en cualquier Sistema de RV, como cerrar - abrir , arriba - abajo, izquierda - derecha, doblar los dedos y girar el guante.	El guante funciona con uno solo y no con otros, la razón más común es que algunos programas permiten que el guante este en otro puerto paralelo además del LPT1.
El guante y las bases de datos no están peleados se pueden realizar muchos experimentos en conjunto.	Cuando se agarra la raqueta no se experimenta la sensación de que se esta cogiendo un cuerpo sólido.
	No se puede sentir el efecto de calor, viento o líquidos.
	El guante esta en tiempo de latencia (instante desde el que ejercemos nuestra acción sobre el mundo hasta que percibimos los resultados) y en que es un único usuario el que participa en la experiencia.

En particular sobre el trabajo que se ha presentado aquí, sugerimos que se podrían continuar investigando los siguientes puntos:

- Guantes con características y materiales diferentes.
- Otras alternativas de posicionamiento ya sea para el guante o para otros tipos de dispositivos virtuales
- Interfaz la creación de entradas específicas para dispositivos virtuales.
- Programación: Otros paquetes de Representación de Mundos Virtuales (Rendering Package).

- Conexión con las Bases de Datos relacionales mediante SQL, Informix, Oracle por mencionar algunas.
- Aplicación con Sistemas Operativos Robustos.
- las interfaces gráficas del usuario, campo en el que parece evidente que algún día deberemos migrar de nuestras interfaces planas actuales a nuevas y espectaculares interfaces gráficas tridimensionales.

PARTE VII. APENDICES.

A.- GLOSARIO.

Abducción

Movimiento por el cual se aleja un miembro del plano medio del cuerpo; abducción del brazo, del ojo.

Actuador

Normalmente son medios mecánicos (hidráulicos) o eléctricos para proporcionar una realimentación táctil que dé fuerza al usuario.

Ad-hoc

Para objetos o situaciones específicas; hecho <<al vuelo>> según se necesite.

Algoritmo

Fórmula que consta de una secuencia de pasos, para resolver un problema o hacer cálculos para que algo suceda; procedimiento computacional mecánico o recursivo.

Algoritmo de reproducción

Fórmula matemática para generar imágenes a partir de datos.

Ancho de Banda

El rango particular de las frecuencias eléctricas (y por tanto, los tipos de comunicación) que pueden ser manejadas por un dispositivo de comunicación.

Animatrónico

generación y manipulación electrónica que parece <<viva>>.

Anteojos de obturación

Gafas estereoscópicas para visualizar imágenes tridimensionales generadas por computadora; usualmente a pilas; también son llamadas gafas de obturación.

Aplicación

Un único conjunto de instrucciones de software diseñadas para resolver un problema particular o ejecutar una tarea particular. Las hojas de cálculo electrónicas o los paquetes de procesamiento son aplicaciones. Los grupos de aplicaciones forman un sistema de información.

Argonne Remote Manipulador

Denotado ARM; sistema de asas de mano a menudo utilizado para proporcionar realimentación al usuario en aplicaciones que conllevan fuerzas o torsiones; dispositivo de control y entrada con un movimiento de 6 grados de libertad.

Ásirlo

Cogerlo, agarrarlo.

Ático

Cuerpo de arquitectura que disimula el tejado/Piso que está bajo el tejado de una casa.

Atributo

Un adjetivo, cualidad, rasgo distintivo o característica de una persona, lugar o cosa; parte de los datos que contienen una única información sobre algo.

Autor (RV)

Diseñador o programador de entornos virtuales. Un experto que proviene de una disciplina externa es a veces llamado autor colaborativo.

Banda ancha

Una facilidad de transmisión multifrecuencia (normalmente un cable coaxial) que transmite simultáneamente un número de canales (por ejemplo, voz, datos, video), cada uno de ellos asignado a una diferente frecuencia, para que no se den interferencias.

Base de conocimiento

Conocimientos registrados electrónicamente, en lugar de ser transmitidos táctil o verbalmente, a los que se puede acceder o transmitir por medio de una interfaz electrónica; conocimientos agregados y acumulados que son clonados, realizados, aumentados, almacenados o transmitidos para resolver problemas. Las reglas de la Inteligencia Artificial son utilizadas para codificar los conocimientos de tal forma que se puedan realizar inferencias a partir de ellos. Los sistemas expertos son ejemplos de conocimientos encerrados.

Base de datos

Datos recolectados y almacenados de forma estructurada para facilitar el acceso a la información; contiene registros y campos.

Bases de datos en espacio real

Un registro de coordenadas basadas en el mundo real sobre objetos virtuales que se basan en una posición relativa al mundo real, para tener significado y contexto; colección de coordenadas específicas de altitud, latitud, y longitud asignadas a objetos virtuales (y las únicas en las que se pueden encontrar dichos objetos virtuales).

Biocontroladores (Biosensores)

Dispositivos que detectan y procesan la mayoría de las señales bioeléctricas (por ejemplo, la actividad eléctrica de los músculos, el movimiento del ojo, etc.) convirtiéndolas en señales digitalizadas.

Biocular

Mostrar la misma imagen a cada ojo; a veces se hace para conservar los recursos de la computadora cuando la percepción de profundidad no es fundamental (Ver estereoscópico).

Bit

Señal de activo-inactivo enviada a la computadora, representada por un 0 o un 1; la unidad de memoria más pequeña en una computadora; parte de un byte, que está compuesto de 8 bits.

Bolsas

Pequeñas bolsas de aire acopladas a los guantes o trajes, controladas por computadora, que se inflan o desinflan en sincronización con el programa; transmiten sensaciones de presión o tangibles al usuario que contacta con un objeto dentro de un entorno virtual; utilizadas con los microagujas.

Blob

Personaje generado por computadora, creado en la Universidad de Carnegie Mellon; programado para dar la apariencia de tener emociones y actuar independientemente.

Boom

Monitor de Omni-orientación Binocular; utilizado en la RV, se trata de un dispositivo que consiste en dos anillos montados de tal manera (formando ángulo recto) que el monitor utilizado para visualizar el entorno virtual queda suspendido en un plano horizontal entre ellos, independientemente de su plataforma de movimiento; dispositivo similar al que se utiliza en un barco para suspender una brújula.

Buffer

Espacio de la computadora de acceso inmediato, utilizado para el almacenamiento temporal de datos, por ejemplo, compartir textos enviados por la computadora más rápido de lo que la impresora puede imprimirlos.

Byte

Unidad común de almacenamiento 8 bits (señales digitales) de información; normalmente equivale a un carácter. Un bit de cada <<octeto>> puede ser un bit de <<paridad>> utilizado para la detección y corrección de los errores de software.

CAD

Diseño asistido por computadora; utilizado en la arquitectura y en el diseño de productos.

CAE

Ingeniería asistida por computadora.

CAM

Fabricación asistida por computadora; cuando las salidas del CAD operan la maquinaria que crea productos.

Cardan

Un dispositivo que consiste en dos anillos (formando un ángulo recto) montados de tal forma que el objeto está suspendido en un plano horizontal entre ambos independientemente del movimiento de la plataforma en la que se apoyan es utilizado en un baco para suspender una brújula y en el laboratorio para sujetar dispositivos como el monitor para visualizar entornos virtuales (denominado BOOM).

CAT Scan

Tomografía asistida por computadora; examen del cuerpo por rayos X.

CD-ROM

(Compact Disk-Read Only Memory). Dispositivo de almacenamiento; contiene una librería llena de textos; un dispositivo óptico de este tipo con escritura es denominado Worm (Write-Once Read-Many times). Los tipos programables 8 con una pequeña cantidad de datos que pueden ser actualizados) se denominan CD-PROM.

Ciberespacio

Término creado por William Gibson, escritor de ciencia ficción, para describir espacios tridimensionales sintetizados por computadora; lugar fantástico de <<alucinación consensual>>. Los términos <<ciberespacio >> y <<realidad virtual>> son a menudo intercambiado, pero algunos investigadores los distinguen basándose en los requisitos de cada uno. Muy a menudo, la realidad virtual incluye <<experiencias>> y el ciberespacio se utiliza más para visualizar información y acceder a ella.

Ciclo de vida

Rango completo de las etapas de desarrollo de un plan o producto, desde su concepción hasta su realización y lanzamiento al mercado.

Código binario

Grupos de señales electrónicas compuestos de dos dígitos, ceros (<<activos>>) y unos (<<inactivos>>) que pueden ser leídos por la computadora; como el sistema decimal, que está basado en 10 dígitos, del cero al nueve.

Código de simulación

Software que controla las acciones del usuario y el comportamiento de objetos, sonidos y fuerzas en un mundo virtual. También controla la graduación del entorno virtual y las cosas que pertenecen a él. El código de simulación de algunas aplicaciones permite a los usuarios coger objetos virtuales.

Conceptual

Concebido mentalmente (por ejemplo, una idea, un plan, pensamiento o diseño)

Conflicto de indicaciones

Una <<enfermedad>> del movimiento causada cuando el cuerpo intenta interpretar indicaciones conflictivas recibidas por los sentidos; a veces es atribuida a un fallo de graduación en los dispositivos oculares.

Conocimiento

Entendimiento obtenido mediante la experiencia o estudio; la suma cognoscitiva e intelectual de lo que es percibido, descubierto o inferido; abarca más que la información.

Convergencia

El punto en el que las imágenes derecha e izquierda se funden en una sola imagen.

Convolucionar

Filtrar y entrelazar señales (por ejemplo, sonidos) y representarlos en 3 dimensiones; utilizado en aplicaciones de RV para recrear sonidos que proporcionan indicaciones direccionales.

Coordenadas

Conjunto de números que determinan la localización de un punto en el espacio; números dados para posicionar un punto, curva o plano; en cpmkimtp <<x,y>> que relaciona un punto con los ejes verticales y horizontales.

CPU

Unidad central de procesamiento; la parte principal de una computadora, que interpreta y ejecuta órdenes según son recibidos; parte que contiene un circuito lógico de computación.

Creación de prototipos

Una técnica para construir versiones rápidas y toscas de un sistema o de las partes de dicho sistema; la creación de prototipos permite a los usuarios y diseñadores de los sistemas detectar las imperfecciones e inventar formas para mejorar el sistema o para explorar un sistema que no es entendido plenamente.

CRTs

Tubos de rayos catódicos; descubierto en el siglo XIX; terminales de presentación o pantallas de TV; pesados y voluminosos para muchas tareas, a veces son reemplazados por los LCDs. Las imágenes son producidas por electrones disparados a una pantalla de fósforo emisora de luz.

Cyborg

Humano robótico modelado directamente a partir de lecturas digitales de una persona y transformado en un <<gemelo>> animado y fotorealista como los que muestran las películas <<Terminator 2>> y <<Robocop>>; personaje sin espíritu producido a través de una metamorfosis ilusoria.

DARPA

Defense Advanced Project Agency, Agencia de Proyectos Avanzados de Defensa del Gobierno de los EE.UU.; ahora es ARPA, sin la D.

Datos

Factores y figuras organizados de los que se pueden obtener conclusiones.

Deformación de objetos

Habilidad del sistema para calcular y presentar imágenes que reflejan elasticidad, por ejemplo, estirar o apelmazar la capa de una piel virtual u otros objetos no rígidos (de tal forma que parezca deformarse más que moverse).

Detección de colisión

Habilidad computacional para calcular si un usuario comparte sus coordenadas espaciales con un objeto virtual, es decir, si el objeto fuese real, cuándo se produce el contacto. El código <<software>> es escrita para que, cuando la colisión virtual sea detectada, la realimentación avise al usuario y el objeto virtual parezca ser modificado o movido.

Dextrous Master

Estructura mecánica de entrada (manual); construida de pequeñas bombas neumáticas, motores, poleas, cables, indicadores de tensión, dispositivos de graduación y sensores de posición; utilizando en lugar de los guantes o conjuntamente en la telerrobótica, incluyendo una realimentación de fuerzas; a veces es denominado <<mano robótica>>.

Digital, digitalizado

Convertido o traducido electrónicamente a dígitos (normalmente los ceros y unos binarios) que pueden ser procesados y visualizados por computadora.

Dispositivo de entrada manual

Aparato controlado por la mano que se utiliza para transmitir señales precodificadas al programa RV.

Dispositivo E/S

Dispositivo E/S utilizados para presentar o recibir datos a través de la computadora.

Distribución espacial de los datos

Asignar orientación (giros, inclinación) y coordenadas de posición (x,y,z) a sonidos digitales que provienen de los datos.

Distorsión metálica

Cantidad medible de interferencia de sonidos o actuación degradada de los rastreadores electromagnéticos cuando son utilizados cerca de grandes objetos metálicos o equipamientos operativos.

Domino

Área particular de conocimiento, especialidad; campo de estudio.

Efecto Doppler

Un aparente incremento de la frecuencia de sonido (o luz) según su fuente se aproxima al observador y una disminución según se aleja (análogo al que produce un tren cuando se acerca y se aleja); su nombre se debe a un físico austriaco.

Efectores

Dispositivos e actuación utilizados en los entornos virtuales para entradas/salidas, sensaciones táctiles y rastreo, por ejemplo, guantes, bolsas y agujas, unidades de sonido.

Eficacia de los datos

Selección y utilización de aquellos tipos de datos que ayudan de forma efectiva a resolver un problema particular; eliminación de los datos no esenciales.

Endoscopia

Fino instrumento con forma de tubo insertado en una cavidad u órgano del cuerpo para realizar un examen médico; pequeñas puntas transmiten las imágenes a los dispositivos de visualización.

Entorno (virtual)

Espacio en el que el usuario de la tecnología RV se imagina a sí mismo y en el que se produce la interacción; visualización de un mundo o escenario generada por computadora.

Entorno de inmersión por redes

Lugares de redes conectadas de actividad Rvm, suma de todos los sitios remotos que se conectan a un espacio virtual compartido, por ejemplo, el Museo de Arte Virtual, redes de entrenamiento militar, etc.

Entornos sintéticos

Término usado por algunas agencias gubernamentales en lugar de entornos virtuales; están entre las siete ideas estratégicas para el avance de la ciencia y tecnología del Departamento de Defensa.

Epcot

Experimental Portotypical Community of Tomorrow.

Esotérico

Incomprensible para las personas no iniciadas, de difícil interpretación.

Escáner

Un dispositivo similar a una fotocopiadora pero que está conectado a una computadora; un dispositivo que contiene software para convertir una imagen en un código que la computadora puede leer; similar a un facsímil. Las imágenes son examinadas electrónicamente ópticamente (mediante unos rayos de luz o electrones bien enfocados) y convertidos a una forma digitalizada para transmitir las, modificarlas o integrarlas por medio de un programa.

Estación de trabajo

Una microcomputadora de un solo usuario, generalmente con una capacidad de gráficos a alta resolución y alta velocidad, que puede poner en marcha las aplicaciones de forma independiente o conjuntamente con otras computadoras por medio de una red. Las estaciones de trabajo son consideradas más potentes que las computadoras personales; sin embargo, muchos PCs modernos superan a las antiguas estaciones de trabajo.

Estereoscópico

Que proporciona un efecto tridimensional; cada ojo recibe una imagen ligeramente diferente de tal forma que, cuando son visualizadas juntas, lo que se ve parece tener profundidad (ver biocular).

Estereo

Estereofonía.

Estereofonía

Reproducción de los sonidos destinada a dar la impresión del relieve acústico.

Evaluación del riesgo

Determinar y evaluar aquellos rasgos de un plan o entorno que contribuyen a su éxito o fracaso.

Exoesqueleto (Esqueleto externo)

Estructura conectada mecánicamente para controlar y recibir realimentación de una aplicación (ver Dextrous Master).

Fisión

Producción de energía como resultado de la escisión de un átomo pesado en el núcleo de dos más ligeros (ver fusión).

Flujo

Movimiento o circulación continua; movimiento suave característico de los fluidos.

Franja

Una de las muchas imágenes quietas que, visualizada secuencialmente a una velocidad prescrita transmite movimiento, por ejemplo, una toma del rollo de una película.

Fuerza

Potencia que es operativa cuando se enfrenta a una resistencia o esfuerzo; fuerza operativa.

Fusión

Formación del núcleo de átomos pesados a partir de otros ligeros, acompañada de un descargo de energía. La fusión termonuclear resulta de aplicar un calor extremo a los átomos ligeros. La fusión fría pretende efectuar la formación de núcleos de átomos más pesados y energía en laboratorios a temperatura ambiente (ver Fisión).

Gesticulación manual

Movimientos predeterminados de la mano o conjuntos de movimientos de la mano que son registrados por un guante de entrada para provocar cambios o manipular un entorno virtual; movimientos sencillos y conjuntos (combinados para obtener gestos) que son interpretados como órdenes por la computadora, que genera las imágenes.

Graduación

Reducción o aumento del tamaño proporcionalmente, de acuerdo a una medición fijada.

Gráficas inmersivas

Reproducciones multidimensionales (dibujos, sonidos y efectos táctiles) que, al ser combinados todos los sentidos, quedan inmersos y la línea entre los mundos real e ilusorio desaparece.

Gráficos (por computadora)

Dibujos o pinturas creados o introducidos en la computadora mediante escáneres o fotografías; pueden ser almacenados, recuperados y manipulados electrónicamente; normalmente son percibidos como figuras geométricas o lineales.

Guante de entrada

Movimientos programados de la mano y las articulaciones que son supervisados a través de un guante que lleva sensores de fibra óptica a lo largo de los dedos y que traduce estos movimientos a rutinas de computación que son reflejadas en un entorno virtual.

Guante de potencia

Versión VPL del Data Glove (Guante de Datos) para uso doméstico.

Háptico/a

Perteneciente a las sensaciones táctiles, presión, temperatura, giro, etc. mediante la piel, los músculos, los tendones o las articulaciones; que representa sensaciones táctiles (aunque no está limitado a ellas).

Hardware

Dispositivos físicos en cada parte del sistema y las conexiones de redes entre distintos lugares.

HDTV

High Definition Television, televisión de alta definición; sistema de transmisión de señales de vídeo de alta resolución.

Hipermedia

Información en forma de texto, diagramas, diagramas de movimiento (animados), imágenes, imágenes en movimiento (televisión), sonidos o programas de computadora; información que se encuentra bajo el control de una computadora y los medios del usuario para navegar a través de ella. El término se utiliza para documentos complejos, como los manuales de entrenamiento, manuales de seguridad y otros extensos documentos almacenados en discos ópticos.

Holograma

Una imagen realizada en película de alta resolución que toma muestras de las ondas luminosas que emanan de un objeto cuando es iluminado por un láser. Cuando la luz brilla a través de esta película especial, las muestras de luz son reproducidas en 3-D.

Imagen

Una reproducción mental, manual o generada por computadora de la apariencia de alguien o algo; un dibujo o pintura; generalmente es percibida como una serie de pixels.

Imitación, Imitado

Representación de la Naturaleza; exhibir una reproducción de algo real; término utilizado para describir cómo el comportamiento de personajes artificiales o ficticios reproduce el mundo real.

Inferencia

Aplicación de reglas a ciertos factores para obtener otros nuevos; razonamientos deductivos.

Información

Colección aleatoria de material no sintetizado; es diferente de los datos, que son información organizada para el análisis, y del conocimiento, que abarca un material empírico, inferido o interpretado.

Inteligencia superpuesta

Potenciación o extensión de las capacidades humanas o el intelecto por encima de la simulación de los procesos humanos cognoscitivos; término creado por Frederick Brooks (ver Inteligencia Artificial, Realidad aumentada).

Inteligencia Artificial (IA)

Programas de computadora que intentan ejecutar eficientemente computaciones enfocadas a ciertas tareas (por ejemplo, memorizar listas) o simular aspectos cognoscitivos del comportamiento humano (por ejemplo, seleccionar las listas que hay que memorizar); un intento de representar, procesar y transferir los conocimientos (ver Inteligencia Amplificada).

Interactivo

Que tiene rasgos que permiten al usuario influenciar o manipular el curso de la acción; permite una interdependencia entre el usuario y el sistema.

Interfaz

Cualquier cosa que se utilice una parte del equipo o una opción del menú para conectar al usuario con un programa; cualquier cosa que se utilice para conectar una computadora con un dispositivo interno (generalmente es una tarjeta insertada en la unidad de computación) como una impresora, un modem o una red.

ISDN

Integrated Services Digital Network, Red Digital de Servicios Integrados; un nuevo tipo de red con mayor anchura de banda, voz completamente digital, datos y sistema de señalización basado en los estándares internacionales; es el acrónimo de muchos chistes (<<I Sill Doesn't Work>>. ¿Todavía no funciona?, etc.).

Kinestesis

Sensaciones derivadas de los músculos, tendones y articulaciones, simuladas por el movimiento y la tensión.

Laparoscopia

Instrumento fino con forma de tubo que es introducido a través de una pequeña incisión y utilizado para un examen abdominal y cirugía mínimamente invasora; una punta especial devuelve imágenes al dispositivo de visión.

Latencia

Espacio entre el movimiento del usuario y la respuesta del sistema; lapso de tiempo que a veces es medido en franjas; retraso entre el cambio real de posición y su reflejo en el programa; similar a tiempo de retraso en la respuesta.

LBE

Location-Based Entertainment; sistemas instalados en lugares fijos, generalmente locales comerciales (por ejemplo, las maquinas recreativas).

LCD

Pantalla de cristal líquido, un tipo de pantalla plana utilizada en las calculadoras y los relojes digitales. Se aplica un campo eléctrico a una superficie de moléculas de cristal líquido, causando que actúen como filtros de polarización de luz; considerado un avance sobre los CRT's aunque ambos pueden ser reemplazados por las pantallas de matriz activa que requieren poca potencia pero son todavía muy costosas y difíciles de fabricar las pantallas de matriz pasiva son otra alternativa.

LEDs

Diodos Emisores de Luz montados sobre el techo o fijos; en las aplicaciones RV son pequeños aparatos utilizados como unos faros en combinación con los sensores de reproducción de imágenes (generalmente montados sobre el equipo de cabeza) para tareas de rastreo; también son situados en los extremos de las secciones articuladas de los guantes para ayudar a medir los grados de curvatura.

Maquinas específicas

Computadora u otro dispositivo designado y reservado para un propósito específico.

Marioneta

Imagen de un usuario tal y como es representado visualmente en un espacio generado por computadora, imagen de una persona que está utilizando el sistema.

Megabyte

MB o Meg; aproximadamente 1 millón de bytes; de forma precisa, 1024 kilobytes o 1048576 bytes.

Memoria de imagen

Facilidad de almacenamiento temporal de acceso inmediato.

Metamorfosis

Ilusión de metamorfosis generada por computadora creada mediante gráficos fotorrealistas y efectos animatrónicos.

Microagujas

Grupos de pequeñas clavijas acopladas a los guantes o a los trajes, conectadas a una computadora y sincronizadas para moverse con un programa, transmitiendo sensaciones de presión, textura y tangibilidad al usuario que entra en contacto con los objetos virtuales, utilizados conjuntamente con las bolsas.

Micromáquina

Dispositivos muy pequeños compuestos de motores, palancas, aparatos, interruptores y poleas infinitamente pequeñas básicamente se trata de las máquinas convencionales hechas miniatura; un pequeño chip de silicio que lleva unos componentes muy pequeños.

MIDI

Musical Instrument Digital Interface; un código estándar que acepta como órdenes diferentes tipos de señales (incluyendo las señales de los bicontroladores) que luego utiliza para controlar instrumentos electrónicos, por ejemplo, sintetizadores de música, procesadores de palabras, dispositivos domésticos, etc.

MIRT

Mario Interactive Real Time; videojuego de Nintendo.

Modelo (RV)

Base de datos gráfica de los colores y coordenadas que corresponden a las piezas con formas poligonales del mundo virtual; una simulación generada por computadora de algo real.

Modem

Modulador/Demodulador, dispositivo utilizado junto con una computadora y una línea telefónica para convertir (modular) datos digitales en tonos análogos o viceversa.

Motor de realidad

Sistema informático con software para generar objetos y mundo virtuales y permitir la interacción del usuario.

MRI

Magnetic Resonance Imaging; Imagen del cuerpo derivada de lecturas de energía magnética; generalmente para uso médico.

Multicomputadora

Una clase de máquinas paralelas conectadas por redes con múltiples nodos de computación, cada una con su propia memoria de datos y órdenes.

Multidimensional

Que tiene muchas dimensiones, como tamaño físico, alcance y resolución.

Multimedia

Combinación de textos, imágenes, videos de movimiento y sonidos; requieren una amplia anchura de banda y potencia computacional.

NASA

National Aeronautics and Space Administration; una división del gobierno federal de los EE.UU.

Nodo

Punto de conexión con una red un interruptor, una terminal, etc.

NPSNET

Red de simulaciones de estaciones de trabajo establecida en el U.S. Naval Postgraduate School (Monterrey) para la investigación, el entrenamiento y la educación; utilidad perteneciente al Gobierno que será conectada a las bases de datos de SIMNET.

NTSC

National Television System Comitee; establece los estándares oficiales para las señales de transmisión por video.

Numéricamente intensivos

Que requieren extensos cálculos matemáticos; triturador de números.

Objeto (Virtual)

Imagen tridimensional generada por computadora en un entorno virtual, representaciones en el espacio virtual con el que el usuario interactúa.

Obturación

Cerrar una abertura introduciendo o aplicando un cuerpo en ella.

Orientado al objeto

Agrupado conceptualmente en unidades autónomas; cualquier cosa inteligible que pueda ser percibida por la mente.

Origen de las imágenes

Líneas, polígonos y superficies y formas curvas.

Par térmico

Un dispositivo que consiste en dos metales distintos unidos de tal forma que, al aplicarles energía, el calor transportado de uno a otro.

Paradigma

Un ejemplo o modelo conceptual utilizado para ilustrar nuevas aproximaciones o formas de pensar.

Parámetros

Factores de medidas o límites determinados.

Paralaje del movimiento

Mirando a través de una ventana (o hacia un pasillo bastante largo) y al mover la cabeza a la izquierda y a la derecha. Sé notará que los objetos más cercanos se desplazan una mayor distancia que los que están más alejados, mientras que los más remotos prácticamente no se mueven.

Pixels

Los puntos de una pantalla que forman letras o dibujos. El número de pixels por pulgada determina la nitidez de la imagen (en reproducciones con volumen se utilizan los <<boxels>>).

Pod (Vaina)

Cápsula, coche o cabina cerrada en la que se puede sentar el usuario.

Polígono

Figura plana formada por tres o más líneas rectas; figura de muchos lados; la pieza más pequeña de una presentación virtual.

Procedimiento

Actividad definida o secuencia de ejecuciones que indican cómo se hace algo, no lo que se hace; es diferente de Proceso.

Proceso

Actividad definida o secuencia de ejecuciones que indican cómo se hace algo, no lo que se hace; la ejecución repetida de tareas lógicas identificada en términos de entrada/salida; es diferente de Procedimiento.

Producto adicional

Equipo Periférico o software adicional que se vende por separado del sistema y puede ser añadido a éste para aumentar su grado de eficacia o para añadir nuevas características; denominados procesadores de aplicación por el AT&T, para servicios telefónicos (como el correo verbal).

Programador

Persona que prepara o escribe un programa.

Propiorreceptor

Un receptor sensorial en los músculos, articulaciones y tendones que responden a estímulos del organismo en la vida real, a partir de dispositivos externos en entornos RV.

Propioceptiva

Es la retroalimentación que suministra nuestro cuerpo acerca de su estado, postura y posición en un momento dado y con respecto a cualquier objeto.

Prototipos experimentales

Aplicación en la que un diseñador construye algo (un edificio o un automóvil, generalmente algo de mayor tamaño que un ser humano) que necesita explorar y ensayar antes de iniciar la construcción; creados para mejorar la comunicación entre las personas que participan en un proyecto y permitir el ensayo y la revisión de los cambios antes de iniciar la producción física, de tal forma que el resultado final responda a las especificaciones (Ver Creación de Prototipos y Prototipos Virtual).

Prototipos virtuales

Una realización de un producto o diseño para ilustrar las características de dicho producto o diseño al usuario, antes de su construcción real; normalmente utilizado como una herramienta de exploración para los desarrolladores o como un accesorio de comunicación para las personas que revisan los diseños propuestos.

<<¿Qué pasaría si...?>>

Escenario hipotético usado para proyectar y explorar las posibles consecuencias a los cursos de acción bajo condiciones variables, por ejemplo, ¿Qué le pasaría al tipo bancario si todo el mundo pagase los préstamos?, ¿...si Argentina invadiese las Malvinas?, ¿...si tu suegra se mudase a tu casa?, ¿...si ocurriesen las tres cosas?

Radioidad

Grados de los colores y la brillantez, como los reflejos difusos en los interiores de un edificio.

RAM

Random Acces Memory; memoria electrónica de la computadora donde se almacenan programas y datos mientras la computadora está en marcha; no es lo mismo que ROM, Read Only Memory, que almacena información permanentemente. La RAM puede ser sobrescrita pero pierde la información cuando se apaga la computadora, permite un acceso rápido a su información. Las memorias convencionales extendidas son tipos de RAM.

Ratón

Dispositivo de conducción manual usado en computadoras personales y estaciones de trabajo.

Razones de proporción

Razón entre la anchura y la altura; normalmente 4:3 del área de las pantallas de televisión y computadora; debe ser consistente entre las presentaciones (pantalla, papel o microfilm) o la imagen es estrecha para llenar el área; por ejemplo, una imagen cuadrada puede aparecer rectangular si las razones de orientación son distintas.

RB1/RB2

Reality Built for 1, Realidad construida para 1 persona; producto VPL creado en 1980, uno de los primeros sistemas comerciales <<on line>> en tiempo real para un solo usuario; permite al usuario interactuar con objetos virtuales. Reality Built for 2 (RB2), realidad construida para 2 personas; permite a dos usuarios interactuar con objetos del entorno y entre ellos; este software funciona con un producto nuevo, el Micocosm.

Realidad artificial

Espacios simulados generados por computadora; una combinación de sistemas computacionales y videosistemas; término creado por Myron Krueger para el título de su libro (Krueger83).

Realidad aumentada

Comparada a realidad <<virtual>>; percepción mejorada; cuando una persona escoge fijarse del mundo real como franja de referencia pero utiliza una presentación transparente (y no opaca) u otros medios no intrusivos para aumentarla, por ejemplo, una superposición esquemática del motor de un coche (ver Inteligencia Amplificada, Artificial).

Realidad proyectada

Una imagen de los movimientos del usuario es proyectada junto con otras imágenes en una extensa pantalla, donde el usuario puede verse a sí mismo como si estuviese en la escena; una sofisticación entre las 3-D y la inmersión total.

Realimentación de fuerza

Una inferencia de la sensación de resistencia proporcionada al usuario, sensación de los actuadores, no sólo de contacto, sino también de la fuerza involucrada (para distinguirla de la Realimentación táctil).

Realimentación táctil

Realimentación dirigida a través de la simulación del sentido de tacto o sensación física; sensación de contacto distinta de la Realimentación de Fuerza, que refleja la magnitud de la fuerza.

Realización (RV)

Actualización de una imagen en un entorno virtual; convertir datos en objetos visualizables; virtualización.

Reconocimiento de voz

Reconocimiento de la voz humana como entrada a la computadora, transcribiendo lo que es hablado, analizando las muestras de sonido y convirtiéndolo en texto digitalizado.

Red de realidad

Un servicio experimental que respalda las conexiones a larga distancia entre usuarios simultáneos en un espacio virtual remoto común; conexiones individuales a larga distancia en las que la gente puede comunicarse y colaborar, aunque no se encuentren en el mismo lugar físico.

Refinamiento de adaptación

Aprovechar el nivel reducido de interacción entre un usuario y el sistema (e.g. andar lentamente hasta detenerse para aumentar el nivel de detalle de sus alrededores); un intercambio inversamente proporcional entre los niveles de actividad y detalle en un escenario de realidad virtual.

Relleno de textura

Lenar los polígonos con muestras almacenadas; sustituir en las superficies los efectos multicolores, de granulado y de construcción por colores únicos en los polígonos individuales; es utilizado en lugar de las muestras creadas independientemente para conservar los recursos de computación.

Representación ciberespacial

Otro término para denotar una visualización tridimensional o una realización RV (Virtualización).

Reproducción

Traducción a otra forma, por ejemplo, convertir señales en un dibujo; ceder o reducir a otro estado o interpretación; ejecutar cálculos de pixels para la visualización.

Reproducción de imágenes en espacio real

Gráficos que dependen de coordenadas en el mundo real; imágenes registradas en lugares reales.

Reproducción de imágenes en tiempo real

Gráficos o imágenes sincronizadas con el tiempo o los acontecimientos del mundo real.

Resolución

Medida de la calidad de la imagen, generalmente expresada en puntos por pulgada (<<dots per inch>>, dpi); cuantos más puntos más nítida es la imagen; a veces es utilizado para medir la calidad en líneas por pulgada (<<lines per inch>>, lpi).

Respuesta de alarma

Indicador sobre la credibilidad de la telepresencia o la presentación virtual; por ejemplo, ¿se asustará el operador humano de un robot si algo le ataca?

RGB

Los colores primordiales, rojo, verde, y azul (Red, Green and Blue); colores aditivos usados en las presentaciones en pantalla de color. Las combinaciones e intensidades de estos tres colores son utilizadas para representar el espectro completo.

ROM

Read-Only Memory; datos almacenados de forma que no pueden ser borrados ni alterados (por ejemplo, un chip acoplado a la máquina para retener información importante no variable, como el sistema operativo); no es lo mismo que RAM, que se utiliza en una sesión y desaparece cuando la máquina es apagada.

RV

Realidad Virtual; también denominada realidad artificial; modelo digital de un entorno; término creado por Jaron Lanier; los <<mundos>> virtuales fueron predichos por Ivan Sutherland en los años sesenta; la convergencia de la simulación por computadora y la visualización que intenta eliminar la separación entre el usuario y la máquina.

Salto cuántico

Un cambio dramático o paso adelante, especialmente en los niveles de conocimiento e información .

Seis grados de libertad

6 DOF (Degrees of Freedom); seis medidas diferentes que pueden asignarse a cualquier movimiento. Por ejemplo, cualquier movimiento puede llevar asociado cambios en al menos tres posiciones (horizontal, vertical y profundidad) y tres grados de orientación (rotación, giro e inclinación). La capacidad para asignar los seis a un determinado movimiento, viene incorporado en muchos dispositivos, como los mecanismos de rastreo de seis grados de libertad.

Sensor de disposición

Dispositivo electromagnético u óptico que traduce los movimientos de la mano o la cabeza del usuario a coordenadas que un programa puede descifrar y utilizar para determinar la situación y orientación del usuario dentro de un espacio; un dispositivo de rastreo.

Silla Flogiston

Silla reclinable diseñada para trabajos intensivos a largo plazo, colocada para reducir al mínimo el estrés y que el ocupante pueda visualizar cómodamente una imagen proyectada sobre su cabeza y manipulada a través de un ratón montado en un brazo de la silla. Algunas versiones son instaladas con una plataforma móvil y dispositivos de realimentación.

SIMMET

Red de simulación de entornos no inmersivos de campos de batalla, que comenzaron en el Institute for Simulation & Training de la Universidad de Central Florida, para el entrenamiento y comunicación militares de la Armada de los EE.UU., fundada por el ARPA (Ver NPSNET).

Simulación

Un proceso o aparato para generar condiciones de ensayo que se aproximan a las condiciones reales u operacionales, por ejemplo, el uso de simuladores de vuelo para entrenar pilotos; in sistema de imitación o que se usa para simular algo distinto.

Síntesis de voz

Reproducción de señales digitalizadas (texto) en voz electrónica.

Sistema experto

Programa computacional que se basa en conocimientos o razonamientos para emular las actuaciones de expertos humanos; contiene reglas codificadas (en una base de conocimientos o de reglas) que reflejan conocimientos individuales o específicos de un campo y luego ejecuta los <<razonamientos>> a través de un intérprete de reglas para

tomar una decisión, llegar a una conclusión y desistir (si no se puede obtener lo pretendido).

Sistema de información

Sistema comercial diseñado para respaldar la toma de decisiones en una organización y para proporcionar un acceso flexible a una consistente colección de datos organizados; conjunto de aplicaciones relativas.

Sistema incompatible

Sistema controlado por diferentes tipos de software que no pueden funcionar juntos.

Sistema operativo

Conjunto básico de instrucciones de software que ponen en marcha una computadora (por ejemplo, el DOS); software de supervisión; proporciona apoyo a los programas de aplicaciones y a las interfaces usadas con el sistema.

Sistema de reproducción de imágenes

Hardware y software diseñado específicamente para capturar, almacenar, manipular, transmitir y reproducir imágenes a partir de una computación real, entornos reales, modelos o documentos; reproduce representaciones realistas de pixles; no es lo mismo que los gráficos. Las aplicaciones comerciales son los videosistemas, las cámaras y los discos ópticos.

Sistema de respaldo de decisiones

Un sistema diseñado para ayudar a los usuarios a tomar mejores decisiones en áreas específicas; base de datos estructurada y conjunto de herramientas de software que facilitan el estudio de decisiones de gestión y de los factores que influyen; software para el análisis y la computación de los factores que influyen en las decisiones.

Software (herramienta)

Programas codificados que dicen a la computadora lo que hacer para realizar tareas específicas; un conjunto de instrucciones lógicas detalladas para operar una computadora.

Sonido espacial

Notas y tonos que parecen emanar de diferentes y variables distancias; reproducido en audiosferas de realidad virtual para elevar el realismo; tipo de sonido envolvente.

Sonorización de datos

Asignar sonidos a datos digitalizados, puede incluir filtrarlos para dar la ilusión de sonido localizado.

Sustitución sensorial cruzada

Sustitución de un estímulo sensorial por otro, por ejemplo, ver en lugar de oír, oler en lugar de gustar. En escenarios RV, señales de un sentido son convertidas en señales de otro y presentadas como tales, por ejemplo, la voz es convertida en un texto que puede ser leído.

Sutherland, Iván

Fue el precursor en este campo en 1965 cuando mencionó la pantalla como <<una ventana a través de la cual se impide ver un mundo virtual>>, que <<parecería real, sonaría real y se percibiría como real>>; construyó el primer equipo de cabeza en Utah, en 1968.

Tableta digitalizadora

Un dispositivo de entrada con una tabla de gráficos para introducir datos en la computadora.

Tecnología de visualización

Sistema o programa que se necesita para generar electrónicamente dibujos e imágenes por computadora.

Teledidónica

Encuentro sexual simulado a través de una conexión telefónica entre dos computadoras; entorno generado por computadora que proporciona realimentación táctil por medio de un programa compartido; descrito.

Tiempo Real

El momento justo en que algo sucede; para resolver problemas con la computadora, el tiempo entre la entrada de datos y la solución; utilizado cuando la respuesta a una entrada es lo suficientemente rápida como para afectar las entradas posteriores.

Triturador de números

Cálculos matemáticos iterativos y repetitivos.

Torsión

Fuerza de giro; rotación alrededor de un eje.

Transmisión por fibra óptica

Enviar grandes cantidades de datos como latidos de luz a través de unos finos filamentos de cristal; es costoso pero preciso, fiable y rápido, especialmente si las distancias son largas.

Transportador

Un potente chip procesador que contiene una computadora, capacidad de memoria y canales E/S y comunicación; es adecuado en problemas que se pueden dividir en subproblemas independientes que, a su vez, pueden ser resueltos simultáneamente.

Trazado de rayos

Simulación de efectos de luz.

Tridimensional (3-D)

Ver 3-D.

Valores absolutos

Posición y orientación dentro de un espacio virtual como medida de un punto simple y constante que sirve de origen, p.e. una <<esquina>> de un espacio virtual usado para rastrear. Si el objeto es movido, se ignoran las medidas previas y se toman nuevas medidas de posición y orientación (ver Valores relativos).

Valores relativos

Valores de posición y orientación relativos a la posición y orientación anteriores en un espacio virtual; calculados desde cero cada vez que un objeto virtual es movido (ver Valores absolutos).

Vaporwave

Palabra del argot de la industria informático que se refiere a los productos software que no han sido todavía demostrados o distribuidos (<<hyperware>> para los productos hardware); productos de pensamiento deseoso; tácticas pre-anunciadas para apoderarse del mercado y ganar tiempo para el desarrollo; están muy extendidas en la industria.

Variable

Algo que puede cambiar y sobre lo que pueden depender otras cosas; algo no conocido sin una cantidad fijada.

Vértigo

Mareo, desmayo/Apresuramiento de la actividad.

VIEW

Virtual Interface Environment Workstation; desarrollado por el Aerospace Human Factors Research Division del Ames Research Center de la NASA (Laboratorio VIEW).

Virtualización

El proceso mediante el cual un humano interpreta una impresión sensorial como un objeto en un entorno distinto al entorno en el que el objeto existe físicamente.

Visión de escena

Presentación virtual visualizada en una extensa pantalla a través de una ventana terminal más que con dispositivos inmersivos.

Visualización

Tomar datos (generalmente científicos), explorar su significado y hacerlos más comprensibles presentándolos en una simulación intuitiva; es usada principalmente en física, química y aplicaciones médicas.

Visualización montada sobre la cabeza

Casco o aparato montado sobre la cabeza que lleva dispositivos ópticos y visuales (localizados o suspendidos enfrente de la cabeza del usuario).

Vortex

Flujo de fluidos rotando alrededor de un eje, como un remolino.

VPL

Virtual Programming Language (Laboratory, Inc.), lenguaje de programación virtual, una de las primeras incursiones comerciales para desarrollar y producir dispositivos de control (por ejemplo, El DataGlove) para entornos virtuales, fundado por Jaron Lanier.

Waldo

Nombre de pila de un robot o dispositivo teleoperado por control remoto.

WIMP

Acrónimo de Widget, icon, mouse and pull down menu (widget, icono, ratón y menú desplegable).

Zona <<Barf>>

Casos en los que se producen debido a que los ritmos de presentación son demasiado rápidos o lentos o debido a un fallo en la sincronización.

#-D

Algunas veces es usado en sistemas en los que cualquier número de dimensiones puede ser mostrado, por ejemplo, sistemas 6-D que tienen tres fuerzas y tres torsiones.

3-D

Una visualización, medio o realización que da la apariencia de altura, anchura y profundidad.

B.- BIBLIOGRAFÍAS.

Mundos Virtuales Realidad Virtual y Ciberespacio. Nicholas Lauroff; México febrero 1994.

Realidad Virtual. Howard Rheingold. Gedisa; vol. 24.

The Science of Virtual Reality and Virtual Environments. Roy S. Kalawsky; Addison Wesley Publishing Company.

Virtual Reality Through the New Looking Glass. Ken Pimentel/Kewin Teixeira; McGraw-Hill.

Virtual Reality Construction Kit. Joe Gradecki; John Wiley Sons, Inc.

Artificial Reality. Krueger, Mayron W; Massachusetts: Adison-Wesley, 1991.

Realidad Virtual. Del Pino, González; España: Paraninfo, 1995.

Realidad Virtual: aplicaciones prácticas en los negocios y la industria. México. Chorafas, Dimitris N. y Steinmann, Heinrich; México: Prentice-Hall Hispanoamericana, 1996.

BETTETINI, G. "Las nuevas tecnologías de la comunicación", Paidós, Barcelona, 1995.

LARIJANI, C. "Realidad Virtual", McGraw-Hill, Madrid, 1994.

C.- FUENTES ADICIONALES DE INFORMACIÓN.

Revista On Off. Número 18,20,23

Revista Información Científica y tecnológica.

Revista Otros Mundos.

Revista Muy Interesante, Especial, Junio 1994.

Mundo Científico, num. 148. Vol 14

Periódico PC Semanal.

Revista Electronic Gaming Monthly.

Revista Electronic Games. Volumen 3.

Revista PCMedia México, marzo, mayo, junio, noviembre 1995.

Revista Mundo 2000.

Revista Reality Hackers.

Virtual Reality 3D Site Projects; <http://www.3dsite.com/cgi/virtual-reality-index.html>.

Virtual World; <http://www.vworlds.com/worlds/>.

Virtual World; <http://www.redcientifica.com>.

Virtual World; <http://www.OzOne.com/avantgo.php3>.

Virtual World; <http://cecusac.gdi.iteso.mx/virtual>.

Virtual World; <http://www.siggraph.org.mx>

Virtual World; <http://www.rtz.es>

Virtual World; <http://www.sec.puc.cl>

Virtual World; <http://www.wmaestro.com/webmaestro>

Virtual World; <http://www.wired.com>

Virtual World; <http://webspace.sgi.com/webspace>

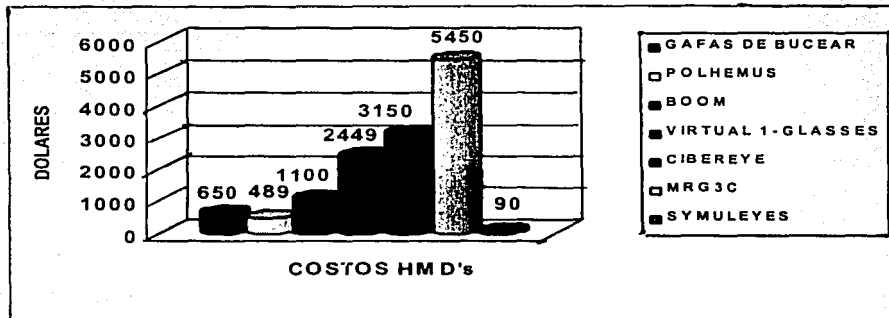
Virtual World; <http://www.cs.umd.edu/projects/hcil/eve.restore>

Virtual World; <http://www.cs.gatech.edu/gvu/virtual>

D.- ESTADISTICAS DEL MERCADO DE LA RV

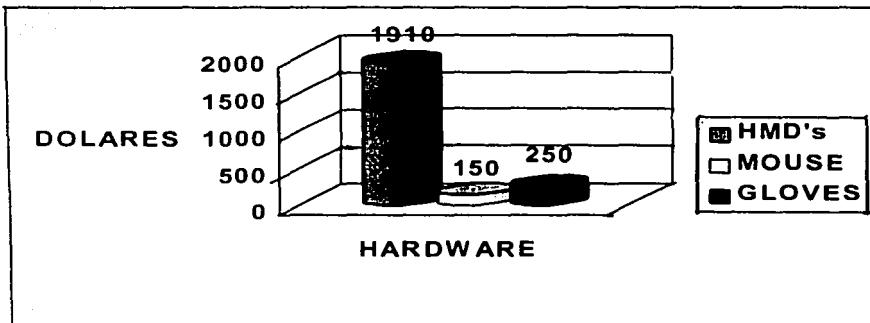
COSTOS DE DISPOSITIVOS HMD's.

A continuación se presenta una gráfica en la que se puede observar la enorme diferencia de costos entre los periféricos de RV.



COSTOS DE HARDWARE.

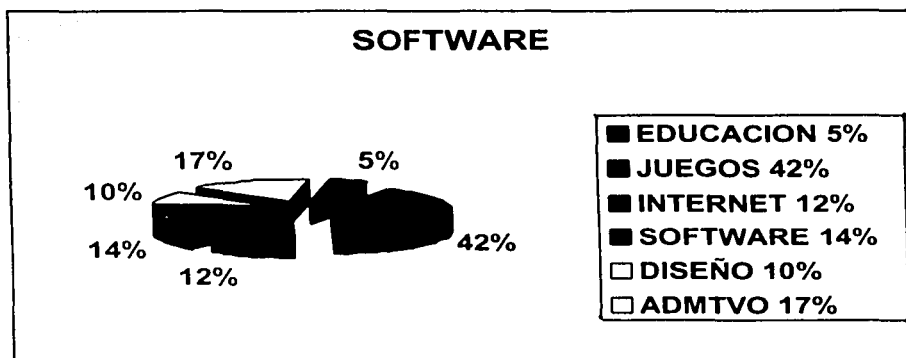
Tomando en cuenta el promedio de los costos de las tres categorías más utilizadas de



Tomando en cuenta el promedio de los costos de las tres categorías más utilizadas de dispositivos para RV por un alto costo, gran parte de los usuarios no los utilizan por su falta de versatilidad además no son bien vistos como inversión en equipos.

SOFTWARE

Actualmente la mayoría del software se orienta hacia los juegos de entretenimiento dejando en ultimo paso a la educación, poniéndole poca atención al diseño y desarrollo, el Internet y los sistemas administrativos comienzan a tener fuerza. Según la siguiente gráfica.



E.- CÓDIGO DEL PROGRAMA.

/* Este programa requiere de la versión 4.01 de Rend386 para compilar */

```
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <math.h>
#include "rend386.h"
#include "userint.h"
#include "plg.h"
#include "splits.h"
#include "tasks.h"
#include "pointer.h"
#include "cursor.h"
#include "segasupp.h"
#include "segio.h"
#include "intmath.h"
#include "gdc.h"

#define to_rad(a) ((a) * 3.14159262 / 180.0)
#define sine(x) sin(to_rad(x/65536L))
#define cosine(x) cos(to_rad(x/65536L))

#define CEILING 2400
#define FLOOR 0
#define LEFT_WALL 500
#define RIGHT_WALL 3650
#define BACK_WALL 0
#define FRONT_WALL 4000

extern unsigned _stklen = 25000;

/* Variables globales*/
/* Esta es nuestra lista de objetos. Archivos utilizados: keyboard, world, cursors. */
OBJLIST *objlist;
```

Apéndice

```
/* Estos son los valores iniciales para una visión en estereo. Archivos utilizados: keyboard,
world, render. */
STEREO                default_stereo = {600,240,320,50,600, 1*65536L};

/* Esta variable indica el tipo de campo de visión. Declarada en el archivo: render.
Archivos utilizados: keyboard, gloveptr, world, render. */
int                   stereo_type = MONOSCOPIC;

/* Esta variable declara la visión inicial en el programa. Archivos utilizados: keyboard,
world. */
VIEW                  default_view = {
                                0,0,-1000,
                                0,0,0,
                                9*65536L,
                                1000,15000,-5000,
                                0,319,0,199,
                                1,100000,
                                1/1.25*65535L,
                                0
                                };

/* Este apuntador indica el valor inicial de la visión. Archivos utilizados: keyboard, world,
render. */
VIEW                  *current_view = &default_view;

/* Este apuntador es el principio para dividir el árbol del programa. Archivos utilizados:
keyboard, world, render, cursors, hdmnip. */
SPLIT                  *split_tree = NULL;

/* Esta variable aloja un "palette". Archivos utilizados: world. */
unsigned char         palette[256*3];

/* Este apuntador es el inicio del grupo de tareas. Archivos utilizados: world. */
TASK                   *tasklist = NULL;

/* Este apuntador guarda el controlador de vídeo instalado para la mano. Archivos
utilizados: main program. */
void                   *v_driver_pointer = NULL;

/* Esta variable guarda la ruta especificada dentro del archivo de configuración ó en el
archivo "world". Archivos utilizados: world. */
char                   loadpath[100] = "";
```

```
/* Esta variable guarda un nombre de archivo temporal incluyendo la ruta "loadpath".
Archivos utilizados: main program. */
static char      tempname[100];

char             framefname[100];

/* Declara una función externa. */
extern void      *screen_data();

/* Esta variable es utilizada para guardar información referente a la configuración actual de
la pantalla. Archivos utilizados: gloveptr, mouseptr, render, userint. */
extern struct    Screeninfo *screeninfo;

/* Esta variable es utilizada dentro del archivo "userint". */
PDRIVER         *menu_device = NULL;
PDRIVER         *manip_device = NULL;

/* Estas variables son utilizadas como de configuración por el dispositivo apuntador
guante, traslación y rotación. Archivos utilizados: world. */
float           gpdo_x = 2, gpdo_y = 2, gpdo_z = 2,
               gpdo_rx = 1, gpdo_ry = 1, gpdo_rz = 1;

/* Estas variables son utilizadas como de configuración por el dispositivo "head tracker",
posición relativa hacia el cuello y rotación de la cabeza. */
float           hdo_x = 0, hdo_y = 0, hdo_z = 0;
float           hdo_rx = 0, hdo_ry = 0, hdo_rz = 0;

/* Estas variables son utilizadas como de configuración por el comando "STEREOLEFT"
en el archivo "world". Archivos utilizados: world keyboard. */
int            sl_xflip = 0, sl_xoff = 0;
long           sl_left = -1, sl_top, sl_right, sl_bottom;

/* Estas variables son utilizadas como de configuración por el comando "STEREORIGHT"
en el archivo "world". Archivos utilizados: world keyboard. */
int            sr_xflip = 0, sr_xoff = 0;
long           sr_left = -1, sr_top, sr_right, sr_bottom;

/* Estas variables son utilizadas en la configuración para el comando de rotación
"STEREORIGHT" y "STEREOLEFT" en el archivo "world". Archivos utilizados: world
keyboard. */
float          sl_xrot = 0, sr_xrot = 0;
```

```
/* Esta variable indica si se está usando un guante. 0 = falso. Archivos utilizados: cursors
hdmanip. */
int          use_glove = 1;

/* Esta variable indica si se utiliza el Sensor de cabeza. 0 = falso. Archivos utilizados:
render. */
int          use_ht = 0;

/* Esta variable indica si se utiliza el sistema de pantalla VGA "doble ángulo de amplitud"
en la cabeza. 0 = falso. Archivos utilizados: render. */
int          use_wide = 0;

/* Esta variable indica si se utiliza el sistema Monocromático. 0 = falso. Archivos
utilizados: render. */
int          use_BW = 0;

/* Esta variable es igual a "1" en el caso de controlar los ojos al utilizar los lentes "Sega de
3ra. Dimensión". Archivos utilizados: render. */
int          swap_eyes = 0;

/* Esta variable es igual a imaginar en la pantalla que se retrocede. Archivos utilizados:
keyboard render. */
int          fancy_background = 0;

/* Esta variable pone un dibujo y lo un refleja al fondo de la pantalla. Archivos utilizados:
keyboard render. */
int          reflection_pool = 0;

/* Esta variable se utiliza para indicar desplegar el "logo". Archivos utilizados: render. */
int          have_logo = 0;

/* Esta variable se utiliza para mostrar el "logo", si es que se tiene uno. Archivos utilizados:
render keyboard. */
int          show_logo = 0;

/* Esta variable se utiliza para mostrar la ubicación actual dentro de la pantalla. Archivos
utilizados: render keyboard. */
int          show_location = 0;
```

```
/* Esta variable se utiliza para mostrar el compás de "3-D" dentro de la pantalla. Archivos
utilizados: render keyboard. */
int          show_compass = 0;

/* Esta variable se utiliza para mostrar la cantidad de cuadros por segundo (velocidad de
Imágenes). Archivos utilizados: render keyboard. */
int          show_framerate = 0;

/* Esta variable se utiliza para limpiar la pantalla en cada cuadro ó imagen. Archivos
utilizados: render world keyboard. */
int          do_screen_clear = 1;

/* Esta variable se utiliza para desplegar un cuadro o segmento de imagen. Archivos
utilizados: world render. */
int          use_frame = 0;

/* Esta variable se utiliza para definir la ubicación de un cuadro ó imagen. Archivos
utilizados: world render. */
int          frame_x = 0, frame_y = 0, frame_w = 320, frame_h = 200;

/* Esta variable se utiliza para desplegar el modo actual de la pantalla. Archivos utilizados:
render world. */
int          vdmode = 0x14;

/* Esta variable se utiliza si se requiere reprocesar la visión o vista actual. Archivos
utilizados: world keyboard. */
int          review = 1;

/* Esta variable se programa a "1" si se requiere copiar el "gram back" en pantalla.
Archivos utilizados: keyboard world colormap render rendrep. */
int          reframe = 0;

/* Esta variable se utiliza si la tecla izquierda de mayúsculas es usada por el usuario.
Archivos utilizados: main program keyboard. */
int          shifted = 0;

/* Esta variable se utiliza para guardar la página visual que está utilizando. Archivos
utilizados: render userint. */
int          v_page = 0;
```

```
/* Esta variable indica si el guante esta disponible. Archivos utilizados: world cursors. */
int             have_glove = 1;

/* Esta variable indica si el apuntador esta disponible. Archivos utilizados: world cursors.
*/
int             have_ptr = 0;

/* Esta variable se utiliza para poder usar el mouse. Archivos utilizados: keyboard. */
int             manip_2D_avail = 0;

/* Esta variable es utilizada para correr ciclo principal de ejecución. Archivos utilizados:
main. */
int             execution = 1;

/* Esta variable se utiliza para indicar si es necesario el "REDRAW" (limpiado de
pantalla). Archivos utilizados: keyboard anim world cursors render hdmanip. */
int             redraw = 1;

/* Esta variable se utiliza para trazar el horizonte. Archivos utilizados: keyboard render
colormap. */
int             do_horizon = 1;

/* Esta variable no es necesaria ya que se define en cualquier parte y puede ser programada
con el archivo "world". Esta define el color de la parte media superior de la pantalla
después de que se haya limpiado, si es que se programa con valor igual a 1. Archivos
utilizados: world render colormap. */
extern          sky_color;

/* Esta variable no es necesaria ya que se define en cualquier parte y puede ser programada
con el archivo "world". Esta define el color de la parte media inferior de la pantalla después
de que se haya limpiado, si es que se programa con valor igual a 1. Archivos utilizados:
world render colormap. */
extern          ground_color;

/* Esta variable no es necesaria ya que se define en cualquier parte y puede ser programada
con el archivo "world". Esta define el color de la pantalla después de que se haya limpiado,
si es que se programa con valor igual a 0. Archivos utilizados: world render colormap. */
extern int      screen_clear_color;
```

```
/* Esta variable tiene el valor del color mas alto posible. Archivos utilizados: render
colormap. */
extern          highest_color;

/* Esta variable activa a 0 si no esta cargada la paleta y non-zero si esta cargada. Archivos
utilizados: world. */
int            npalette = 0;

/* Esta variable tiene el nombre actual del manejador del interruptor. Esta variable activa a
0 si no esta cargada la paleta y non-zero si esta cargada. Archivos utilizados: world. */
char          swdname[40] = "sega";

/* Esta variable tiene el nombre actual del manejador del apuntador del guante. Archivos
utilizados: world. */
char          gpdname[40] = "pglove";

/* Esta variable tiene el nombre actual del manejador que rastrea la cabeza. Archivos
utilizados: world. */
char          hdname[40] = "none";

/* Esta variable tiene el nombre del apuntador del manejador del guante y la figura del
cursor. Archivos utilizados: world. */
char          gpcursor[40] = "handsm.fig";

/* Esta variable tiene el nombre actual del manejador de vídeo. Archivos utilizados: main
program y world. */
char          vdname[40] = "vd256.rvd";

/* Esta variable almacena el nombre del controlador del mouse. Archivos utilizados: world.
*/
char          mdname[40] = "mouse";

int head_device = 0,
    PORT = 1,
    BALL_SPEED = 10;
extern SEGMENT *body_seg;
PDRIVER *mouser;
SEGMENT *BALL, *RACQUET, *hand_joints[20],
        *our_hand, *glove_joints[20];
OBJECT *palm_object;
```

```
int use_powerglove = 0, use_mouse = 0, use_3dmouse = 0, use_gdc = 0, use_cyberman =
0;
POINTER mouse_data;
int one_attached = 0, palm_color = 0, click = 0, hit = 0;
OBJECT *racquet_object;
long rx, ry, rz;
```

```

/*****
Esta función es utilizada para el archivo "render.c" para desplegar diferentes objetos en la
pantalla del usuario. Es opcional. Sin embargo, "render.c" requiere accederla.
*****/

```

```

***/
void status_on_screen()
{

```

```

/*****
Esta función conecta el directorio de carga con el nombre diferente a "\\" y "/" y el
directorio de carga no este vacío. Archivos utilizados: world.c.
*****/

```

```

char *fix_fname(char *name)
{
    if (loadpath[0] && !strchr(name, '\\') && !strchr(name, '/'))
        sprintf(tempname, "%s\\%", loadpath, name);
    else
        strcpy(tempname, name);
    return tempname;
}

```

```

/*****
Esta función es llamada cuando el usuario finaliza el programa de gráficos. Desactiva el
sistema de gráficos y termina el programa "renders" así como el programa principal.
*****/

```

```

void closeall() {
    exit_graphics();
    reset_render();
    exit(0);
}

```


/******

Esta función es para leer la entrada del teclado.

```
unsigned getkey() {
    unsigned c;
    union REGS regs;

    regs.h.ah = 2;
    int86(0x16, &regs,&regs);
    shifted = (regs.h.al & 3);
    if ((c=bioskey(0)) & 0xff) c &= 0xff;
    else if ( shifted ) c |= 1;
    return c;
}
```

/******

Esta función maneja las teclas pulsadas por el usuario pulsando cualquier tecla por el usuario.

```
void handle_key ( unsigned int c, int *rac ) {
    FILE *infile;

    switch ( c ) {
        case 'q':
            case 'Q':popmsg ( "Deseas salir del Juego? (y/n)" );
                if ( toupper(getkey())=='Y' ) execution = 0;
                else redraw = 1;
                break;

        /* Mirar hacia arriba. Paso por 1 el valor del ángulo.*/
        case 0x4800: current_view->tilt += 65536L;
            redraw = 1;
            break;

        /* Mirar hacia abajo. Paso por 1 el valor del ángulo. */
        case 0x5000: current_view->tilt -= 65536L;
            redraw = 1;
            break;

        /* Mirar a la izquierda. Paso por 1 el valor del ángulo. */
        case 0x4B00: current_view->pan -= 65536L;
            redraw = 1;
            break;
    }
}
```

```

/* Mirar a la derecha. Paso por 1 el valor del ángulo. */
case 0x4D00: current_view->pan += 65536L;
             redraw = 1;
             break;

case 'Y':
case 'y': rel_rot_segment ( body_seg, 0L, 90*65536L, 0L, RYXZ );
          update_segment ( body_seg );
/*current_view->pan += 90*65536L;*/
          redraw = 1;
          break;

case 'b':
case 'B': current_view->ex -= 50L * sine(current_view->pan);
          current_view->ez -= 50L * cosine(current_view->pan);
          rel_move_segment ( body_seg, -50L*sine(current_view->pan),
                             0,
                             -50L*cosine(current_view->pan));
          update_segment ( body_seg );

          redraw = 1;
          break;

case 'f':
case 'F': current_view->ex += 50L * sine(current_view->pan);
          current_view->ez += 50L * cosine(current_view->pan);

          rel_move_segment ( body_seg, 50L*sine(current_view->pan),
                             0,
                             50L*cosine(current_view->pan));
          update_segment ( body_seg );
          redraw = 1;
          break;

case 'r':
case 'R': current_view->ex += 50L * cosine(current_view->pan);
          current_view->ez -= 50L * sine(current_view->pan);
          rel_move_segment ( body_seg, +50L*cosine(current_view->pan),
                             0,
                             -50L*sine(current_view->pan));
          update_segment ( body_seg );
          redraw = 1;

```

```

        break;
    case 'l':
    case 'L': current_view->ex -= 50L * cosine(current_view->pan);
              current_view->ez += 50L * sine(current_view->pan);
              rel_move_segment ( body_seg, -50L*cosine(current_view->pan),
                                0,
                                +50L*sine(current_view->pan));
              update_segment ( body_seg );
              redraw = 1;
              break;

    case ',':

        if ((click==0)&&(!one_attached))
        {
            abs_rot_segment(hand_joints[1], 0L,30*65536L+18061L*127,0L, RYXZ);
            abs_rot_segment(hand_joints[2], -20*65536L,90*65536L,-
5*65536L+38700L*127, RYZX);
            abs_rot_segment(hand_joints[3], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[4], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[5], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[6], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[7], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[8], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[9], 38700*90,0L,0L, RYXZ);
            abs_rot_segment(hand_joints[10],38700*90,0L,0L, RYXZ);

            hit = 0;
            if ( sphere_pretest ( palm_object, rx, ry, rz ) < 100 ) hit = 1;
            if ( hit )
            {
                attach_segment ( RACQUET, our_hand );
                one_attached = 1;
                *rac = 0;
            }
            update_segment ( our_hand );
            click = 1; redraw = 1;
        }
        else if ((click ==1)&&(!one_attached))
        {
            abs_rot_segment(hand_joints[1], 0L,30*65536L,0L, RYXZ);
            abs_rot_segment(hand_joints[2], -20*65536L,90*65536L,-5*65536L, RYZX);

```

```

abs_rot_segment(hand_joints[3], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[4], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[5], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[6], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[7], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[8], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[9], 0,0L,0L, RYXZ);
abs_rot_segment(hand_joints[10],0,0L,0L, RYXZ);
update_segment ( our_hand );
click = 0; redraw = 1;
}
}
}

/*****
Esta función ejecuta la actualización después de cualquier objeto o perspectiva en
movimiento. Se hace la mayoría del trabajo en la función screen_refresh en el archivo
render.c
*****/
void refresh_display(){

    initialize_screen_factors ( current_view );
    fast_view_factors ( current_view );
    screen_refresh ( current_view );

}

int    x_position,
        y_position,
        z_position,
        inc_val_x,
        inc_val_y,
        inc_val_z;

/*****
Esta función mueve la pelota acorde con la raqueta golpeando en los límites de la cancha.
*****/
void move_ball(SEGMENT *BALL)
{
    y_position+=inc_val_y;
    if ( y_position > CEILING )
    {

```

```

    inc_val_y = -BALL_SPEED; /* (rand() % 50);*/
    y_position+=inc_val_y;
}
else if ( y_position <= FLOOR )
{
    inc_val_y = BALL_SPEED; /* (rand() %50); */
    y_position+=inc_val_y;
}

x_position+=inc_val_x;
if ( x_position > RIGHT_WALL )
{
    inc_val_x = -BALL_SPEED; /*(rand() %50);*/
    x_position+=inc_val_x;
}
else if (x_position < LEFT_WALL )
{
    inc_val_x = BALL_SPEED; /*(rand() % 50);*/
    x_position+=inc_val_x;
}
if ( inc_val_z > BALL_SPEED ) inc_val_z -= 1;
if ( inc_val_z < -BALL_SPEED ) inc_val_z += 1;
z_position+=inc_val_z;
if ( z_position > FRONT_WALL )
{
    inc_val_z = -BALL_SPEED; /*(rand() % 50); */
    z_position+=inc_val_z;
}
else if ( z_position <= BACK_WALL )
{
    inc_val_z = BALL_SPEED; /* (rand() % 50);*/
    z_position+=inc_val_z;
}
abs_move_segment ( BALL, x_position, y_position, z_position );
update_segment ( BALL );

;

void prprint(int x,int y, int color, char *t)
{
    int bk = (color>8) ? 0 : 15;
    printxyr(x,y,bk,t,0);
}

```

```

        printxyr(x+1,y+1,color,t,0);
    }

void mouse_update(int *rac)
{
    long palm_x, palm_y, palm_z;
    unsigned mouse;

    mouse = pointer_read (mouser, &mouse_data);

    if (!one_attached)
    {
        hit = 0;
        // get_object_bounds ( palm_object, &palm_x, &palm_y, &palm_z );
        if ( sphere_pretest ( palm_object, rx, ry, rz ) < 100 ) hit = 1;

        if ((hit) && (palm_color == 0)) { set_poly_color(palm_object, 0, 3 ); palm_color
= 1; }
        if ((!hit) && (palm_color == 1)) { set_poly_color(palm_object, 0, 0x14EF );
palm_color = 0; }
    }

    if ((mouse_data.buttons & 1) && (click==0))
    {
        abs_rot_segment(hand_joints[1], 0L,30*65536L+18061L*127,0L, RYXZ);
        abs_rot_segment(hand_joints[2], -20*65536L,90*65536L,-
5*65536L+38700L*127, RYZX);
        abs_rot_segment(hand_joints[3], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[4], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[5], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[6], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[7], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[8], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[9], 38700*90,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[10],38700*90,0L,0L, RYXZ);
        hit = 0;
        // get_object_bounds ( palm_object, &palm_x, &palm_y, &palm_z );
        if ( sphere_pretest ( palm_object, rx, ry, rz ) < 100 ) hit = 1;
        if ( hit )
        {
            attach_segment ( RACQUET, our_hand );
        }
    }
}

```

```

    one_attached = 1;
    *rac = 0;
    ;

    update_segment ( our_hand );
    click = 1; redraw = 1;
    ;

if (!one_attached)
{
    if (((mouse_data.buttons & 1)==0) && (click ==1))
    {
        abs_rot_segment(hand_joints[1], 0L,30*65536L,0L, RYXZ);
        abs_rot_segment(hand_joints[2], -20*65536L,90*65536L,-5*65536L, RYZX);
        abs_rot_segment(hand_joints[3], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[4], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[5], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[6], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[7], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[8], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[9], 0,0L,0L, RYXZ);
        abs_rot_segment(hand_joints[10],0,0L,0L, RYXZ);

//      if ( one_attached ) detach_segment ( RACQUET);

//      one_attached = 0;
        update_segment ( our_hand );
        click = 0; redraw = 1;
    ;
    ;

if ( mouse & PNEW_POS )
{

    if (mouse_data.dx)
    {
        if ( mouse_data.dx > 127 ) mouse_data.dx = 127;
        if ( mouse_data.dx < -127 ) mouse_data.dx = -127;

        rel_move_segment ( our_hand, 5*mouse_data.dx*cosine(current_view->pan), 0,
        -5*mouse_data.dx*sine(current_view->pan) );
        update_segment ( our_hand );
    }
}

```

```

    |
    |
    | if((mouse_data.dz))
    | {
    |
    |     if (mouse_data.dz > 127 ) mouse_data.dz = 127;
    |     if (mouse_data.dz < -127) mouse_data.dz = -127;
    |     rel_move_segment ( our_hand, 0, 5*mouse_data.dz, 0 );
    |     update_segment ( our_hand );
    |
    |
    | if (mouse_data.dy)
    | {
    |     if ( mouse_data.dy > 127 ) mouse_data.dy = 127;
    |     if ( mouse_data.dy < -127 ) mouse_data.dy = -127;
    |
    |     rel_move_segment ( our_hand, 5*mouse_data.dy*sine(current_view->pan), 0,
    | 5*mouse_data.dy*cosine(current_view->pan) );
    |     update_segment ( our_hand );
    |
    | }
    |
    | redraw = 1;
    |
    |
    | void gdc_update()
    | {
    |     unsigned int linear, rotation, buttons;
    |     gdc_read ( &linear, &rotation, &buttons );
    |
    |     if (!one_attached)
    |     {
    |         hit = 0;
    |         if ( sphere_pretest ( palm_object, rx, ry, rz ) < 100 ) hit = 1;
    |
    |         if ((hit) && (palm_color == 0)) { set_poly_color(palm_object, 0, 3 ); palm_color = 1; }
    |         if ((!hit) && (palm_color == 1)) { set_poly_color(palm_object, 0, 0x14EF ); palm_color
    | = 0; }
    |
    |
    |         if ( linear )
    |         {
    |             if (linear & GDC_MOVE_UP) rel_move_segment ( our_hand, 0, 15, 0 );
    |

```



```

    if (linear & GDC_MOVE_DN) rel_move_segment ( our_hand, 0, -15, 0);
    if (linear & GDC_MOVE_LF) rel_move_segment ( our_hand, -15*cosine(current_view->pan), 0, -200*sine(current_view->pan) );
    if (linear & GDC_MOVE_RT) rel_move_segment ( our_hand, 15*cosine(current_view->pan), 0, 200*sine(current_view->pan) );
    if (linear & GDC_MOVE_FW) rel_move_segment ( our_hand, 15*sine(current_view->pan), 0, 200*cosine(current_view->pan) ); update_segment ( our_hand );
    if (linear & GDC_MOVE_BK) rel_move_segment ( our_hand, -15*sine(current_view->pan), 0, -200*cosine(current_view->pan) ); update_segment ( our_hand );

    update_segment ( our_hand );
}

```

```

;
SEGMENT *palm;
OBJECT *handle;
/*****

```

Esta función ejecuta las funciones del programa. El loop termina cuando execution = 1.

*****/

```

void main_loop(SEGMENT *BALL, SEGMENT *RACQUET){
    POINTER gp;
    OBJECT *ball_object;
    long x,y,z,rad;
    int NO_RACQUET = 1;
    char buf[100];
    long dist;
    racquet_object = seg_get_object(find_segment_by_name(RACQUET, "racquet"));
    ball_object = seg_get_object (BALL);
    rad = get_object_bounds ( handle, &rx, &ry, &rz );

```

```

while ( execution )
{
    if (bioskey(1))          handle_key (getkey(), &NO_RACQUET);
    if (redraw)             refresh_display();
    move_ball(BALL);

    if ( use_powerglove ) glove_update( manip_device, &gp );
    if ( use_mouse ) mouse_update(&NO_RACQUET);
    if ( use_gdc ) gdc_update();
    update_segment ( body_seg );

```

```

    if ( NO_RACQUET )

```



```
if ( v_driver_pointer == NULL )
{
    fprintf ( stderr, "Controlador de vídeo malo/n" );
    exit ( 1 );
}

/*****
Esta función es usada al leer la raqueta en el juego y asigna una llamada al segmento
ball_seg.
*****/
SEGMENT * read_ball()
{
    FILE *in;
    OBJECT *ball;
    SEGMENT *ball_seg;

    if ((in = fopen ( "ball.plg", "r" )) == NULL )
    {
        fprintf ( stderr, "No se puede abrir el archivo ball.\n" );
        exit(1);
    }

    set_loadplg_offset ( 0, 0, 0 );
    set_loadplg_scale ( 5, 5, 5 );
    ball = load_plg ( in );
    if ( load_err != 0 )
    {
        fprintf ( stderr, "Error al cargar el archivo ball.plg.\n" );
        exit(1);
    }

    add_to_objlist ( objlist, ball );
    ball_seg = new_seg ( NULL );
    seg_set_object ( ball_seg, ball );

    update_segment ( ball_seg );

    x_position = 1000;
    y_position = 500;
    z_position = 2000;
    inc_val_x = 50;
```

```
inc_val_y = 40;
inc_val_z = 40;
```

```
return ball_seg;
}
```

```
/******
Esta función es usada al leer la raqueta en el juego y asigna una llamada al segmento
racquet_seg.
*****/
```

```
SEGMENT * read_racquet()
```

```
{
FILE *in;
SEGMENT *racquet_seg;
if ((in = fopen ( "racquet.fig", "r" )) == NULL )
{
fprintf ( stderr, "No se puede abrir el archivo racquet.\n" );
exit(1);
}
}
```

```
/* set_loadplg_offset (1000, 500, 2000); set_loadplg_scale (3, 3, 3); */
set_readseg_objlist ( objlist );
set_readseg_scale ( 3, 3, 3 );
racquet_seg = readseg (in, NULL);
if ( readseg_err != 0 )
{
fprintf ( stderr, "Error al cargar el archivo racquet.plg.\n" );
exit(1);
}
}
```

```
/*add_to_objlist ( objlist, racquet ); racquet_seg = new_seg ( NULL ); seg_set_object (
racquet_seg, racquet ); */
```

```
abs_move_segment ( racquet_seg, 1500, 500, 2000 );
// abs_rot_segment ( racquet_seg, 0L, 90*65536L, 0L, RXYZ );
update_segment ( racquet_seg );
```

```
handle = seg_get_object (find_segment_by_name(racquet_seg, "handle" ));
if ( handle == NULL )
{
fprintf ( stderr, "No se encuentra el segmento handle.\n" );
exit(1);
}
```

```
    }  
  
    return racquet_seg;  
}  
  
void get_glove()  
{  
    PDRIVER *gd;  
    extern PDRIVER *gloveptr_init(char *gname, long sx, long sy,  
        long sz, long srx, long sry, long srz );  
  
    gd = gloveptr_init ( gpdbname, gpdo_x*65536.0, gpdo_y*65536.0, gpdo_z*65536.0,  
        gpdo_rx*65536.0, gpdo_ry*65536.0, gpdo_rz*65536.0 );  
  
    manip_device = menu_device = gd;  
    if(gd)  
        load_glove_cursor ( body_seg, manip_device, gpcursor );  
}  
  
void read_hand()  
{  
    FILE *in;  
    if ((in = fopen ( "handsm.fig", "r" )) == NULL )  
    {  
        fprintf ( stderr, "No se puede abrir el archivo handsm.\n" );  
        exit(1);  
    }  
    set_readseg_objlist ( objlist );  
    set_readseg_seglist ( hand_joints, 20 );  
    set_readseg_scale ( 1, 1, 1 );  
    our_hand = readseg (in, NULL);  
    if ( readseg_err != 0 )  
    {  
        fprintf ( stderr, "Error al cargar el archivo handsm.plg.\n" );  
        exit(1);  
    }  
}  
  
// palm_obj = seg_get_object(find_segment_by_name(our_hand, "palm"));  
  
update_segment(our_hand);  
attach_segment ( our_hand, body_seg );
```

```

rel_move_segment (our_hand, 0, 0, 600 );
;

void main(int argc, char *argv[])
{
char *fname, *in_filename;
FILE *in;
int i;

if (argc == 1)
{
printf ( "\n\nusage: rb [-][/] ?pm3gi");
printf ( "\n\n    ? ,h = Ayuda");
printf ( "\n\n    P = PowerGlove");
printf ( "\n\n    M = Mouse");
printf ( "\n\n    G = Global 3D Controller COM1");
printf ( "\n\n    I = Global 3D Controller COM2");
exit(0);
}

setup_render(40,800);
atexit(closeall);
set_global_split_root ( &split_tree );
initial_world_split ( &split_tree );
set_move_handler ( split_move_handler );
objlist = new_objlist();

if ((in = fopen ( "court.wld", "r" )) == NULL )
{
fprintf ( stderr, "Error al abrir al archivo world.\n" );
exit(1);
}
if (read_world(in))
{
fprintf ( stderr, "Error al leer el archivo world.\n");
exit(1);
}
fclose(in);
load_video_driver ( vdname );
screeninfo = screen_data();
highest_color = screeninfo->colors-1;
preset_default_colors();

```

```

frame_x = screeninfo->xmin;
frame_y = screeninfo->ymin;
frame_w = screeninfo->xmax - screeninfo->xmin+1;
frame_h = screeninfo->ymax - screeninfo->ymin+1;

```

```

default_view.left = screeninfo->xmin;
default_view.top = screeninfo->ymin;
default_view.right = screeninfo->xmax;
default_view.bottom = screeninfo->ymax;
default_view.aspect = screeninfo->aspect;

```

```

if(enter_graphics() ) {
    fprintf ( stderr, "No se puede entrar al modo de graficos\n\n");
    exit(1);
}

```

```

screen_clear_color = 0;

```

```

sky_color = 1; ground_color = 2;
body_seg = new_seg(NULL);
set_readseg_scale ( 1,1,1 );

```

```

initialize_screen_factors ( current_view );
fast_view_factors ( current_view );

```

```

BALL = read_ball();
RACQUET = read_racquet();

```

```

for (i=1; i < argc; ++i)
{
    if (argv[i][0] == '/' || argv[i][0] == '-')
    {
        switch(toupper(argv[i][1]))
        {
            case 'h':
            case '?': printf ( "\n\nusage: rb [-]/[?]pm3gc");
                    printf ( "\n\n    ?,h = Ayuda");
                    printf ( "\n\n    P = PowerGlove");
                    printf ( "\n\n    M = Mouse");
                    printf ( "\n\n    G = Global 3D Controlador COM1");
                    printf ( "\n\n    I = Global 3D Controlador COM2");

```

```

        exit(0);
        break;

    case 'P': use_powerglove = 1; get_glove(); break;
    case 'M': use_mouse = 1; read_hand();
        menu_device = mouser = mouseptr_init(mdname);
        pointer_reset ( mouser );
        break;
    case 'G': use_gdc = 1;
        PORT = 1;
        gdc_init(PORT);
        read_hand();
        break;
    case 'I': use_gdc = 2;
        PORT = 2;
        gdc_init(PORT);
        read_hand();
        break;
    }
}
}

abs_move_segment ( body_seg, 2250, 800, -300 );
update_segment ( body_seg );

palm = find_segment_by_name (body_seg, "palm" );
rel_rot_segment ( palm, 0L,0L, -45*65536L, RZXY );
update_segment ( palm );
if ( palm == NULL )
{
    fprintf( stderr, "No se encuentra el segmento palm.\n" );
    exit(1);
}
palm_object = seg_get_object ( palm );

main_loop(BALL, RACQUET); /* empezar o esta desconectado. */
}

```


Apéndice

thumb 8 4

0 0 -50
0 260 -50
-85 320 -50
-160 260 -50

40 0 85
50 200 80
-85 320 50
-160 260 50

0x14FF 4 0 1 2 3
0x14FF 4 7 6 5 4
0x14EF 4 1 5 6 2
0x14EF 4 0 3 7 4

worldscale 1.0

hither 10
yon 2000000
skycolor 16
groundcolor 16

start 2000,1000,-400 0,0,0 1

ambient 76

surfacedef sred	0x11FF
surfacedef sorange	0x12FF
surfacedef smagenta	0x13FF
surfacedef sbrown	0x14FF
surfacedef slight_yellow	0x15FF
surfacedef syellow	0x16FF
surfacedef slight_green	0x17FF
surfacedef sgreen	0x18FF
surfacedef scyan	0x19FF
surfacedef slight_blue	0x1AFF
surfacedef sblue	0x1BFF
surfacedef slight_pink	0x1CFF
surfacedef spink	0x1DFF
surfacedef sgray	0x1EFF

THUMB.PLG

Apéndice

surfacedef swhite	0x1FDF
surfacedef swhite2	0x1FDD
surfacedef plaster0x1fd8	
surfacedef brick	0x11ff
surfacedef wood	0x12df
surfacedef painted	0x13f8
surfacedef glass1	0x3e00
surfacedef glass2	0x3a00
surfacedef tin	12
surfacedef carpet	0x4a
surfacedef greypaint	239
surfacedef bright	255
surfacedef grain	0x14e8
surfacedef artwork	4
surfacedef beige	0xfd
surfacedef blue	8
surfacedef red	9
surfacedef orange10	
surfacedef cobble239	
surfacedef marble	0x1Df8
surfacedef steel1	0x28eF
surfacedef steel2	0x25eF
surfacedef teal	0x12ff
surfacedef mirror	0x2a00
surfacedef uglypaint	12
surfacedef white	15
surfacedef black	1

surfacemap walls
surface 1 plaster
surface 2 swhite2

surfacemap lines
surface 1 red

PISO DE LA CANCHA

split 0,0,0,0,1,0 1
polyobj 4 sgray 800,0,0 800,0,4000 4000,0,4000 4000,0,0
polyobj 4 sred 800,0,3000 800,0,3100 4000,0,3100 4000,0,3000
polyobj 4 sred 800,0,2200 800,0,2300 4000,0,2300 4000,0,2200
polyobj 4 sred 1200,0,2300 1200,0,3000 1300,0,3000 1300,0,2300

Apéndice

polyobj 4 sred 3500,0,2300 3500,0,3000 3600,0,3000 3600,0,2300

TECHO DE LA CANCHA

split 0,2400,0,0,1,0 2

polyobj 4 sgray 4000,2400,0 4000,2400,4000 800,2400,4000 800,2400,0

REGRESAR DE UN LADO DE LA PARED

split 800,1,0,1,0,0 3

object courtwl2 40,48,4 0,270,0 800,0,0 0 walls fixed

FRENTE DE UN LADO DE LA PARED

split 4000,1,2000,1,0,0 4

object courtwl2 40,48,4 0,270,0 4000,0,0 0 walls fixed

REGRESAR DE LA PARED DE LA CANCHA

split 800,1,4000,0,0,1 5

object courtwall 32,48,4 0,0,0 800,0,4000 0 walls fixed

object line 3200,100,10 0,0,0 800,300,3990 0 lines fixed

endsplits

COURT.WLD