

Informática Gráfica

Tema 4. Interacción.

Juan Carlos Torres

Dpt. Lenguajes y Sistemas Informáticos
ETSI Informática y de Telecomunicación
Universidad de Granada

Índice

Informática Gráfica Tema 4. Interacción.

- 1 Introducción
- 2 Posicionamiento
- 3 Selección
- 4 Animación

Sistemas interactivos

Un sistema gráfico interactivo es un sistema que responde de forma directa a las acciones del usuario.

El sistema funciona según el siguiente ciclo:

- ▶ El usuario realiza una acción
- ▶ El programa recibe información
- ▶ Procesa la acción
- ▶ Redibuja el modelo

Interactividad

La incorporación de interactividad permite realizar aplicaciones que respondan a las acciones de los usuarios.

La mayor parte de los sistemas gráficos son interactivos.

La interactividad es esencial en

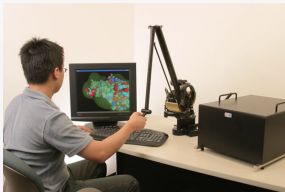
- ▶ sistemas de diseño
- ▶ juegos
- ▶ modeladores
- ▶ simuladores

Interactividad

En un sistema gráfico interactivo el usuario debe disponer de al menos un dispositivo de entrada y un dispositivo de visualización. Como mínimo teclado y monitor. Normalmente se dispone además de un ratón.

En sistemas profesionales se suelen utilizar además dispositivos de entrada especiales como:

- ▶ Tableta digitalizadora
- ▶ Sistemas de posicionamiento
- ▶ Palpadores



Interactividad

La interactividad requiere que el retardo (latencia) entre la acción del usuario y la respuesta del sistema sea suficientemente pequeño para que el usuario perciba una relación de causa efecto.

Las restricciones de un sistema de tiempo real son mayores, el retardo debe ser menor que un tiempo prefijado.

Realimentación

La realimentación es el mecanismo mediante el cual el sistema da información al usuario útil para determinar la siguiente acción a realizar. La información de realimentación que el sistema genera en cada momento depende lógicamente del estado del sistema y de la información previamente entrada por el usuario.

Se puede usar con diferentes fines específicos:

- ▶ mostrar el estado del sistema
- ▶ como parte de una función de entrada
- ▶ para reducir la incertidumbre del usuario

Técnicas

Con frecuencia, el usuario debe introducir una posición que satisfaga una determinada condición, por ejemplo tener la misma coordenada X que la posición previa.

En estas situaciones, la aplicación puede ayudar al usuario a realizar su tarea proporcionándole técnicas específicas que simplifiquen el proceso de interacción, usando técnicas de interacción.

Por ejemplo puede hacer que el cursor solo se mueva en horizontal.

Técnicas:

- ▶ Líneas elásticas
- ▶ Guías
- ▶ Gravedad

Funciones

Un sistema gráfico interactivo necesita normalmente funciones de entrada para leer texto y valores numéricos como cualquier sistema gráfico, pero necesita además poder

- ▶ leer posiciones
- ▶ seleccionar componentes del modelo geométrico

La lectura de posiciones (posicionamiento) permite generar posiciones en coordenadas del mundo. La selección permite identificar interactivamente elementos del modelo geométrico.

Dispositivos

El usuario introduce la información por medio de dispositivos de entrada. Estos pueden ser de propósito general (teclado), o específicos para la generación de datos geométricos (digitalizador).

Podemos clasificar los dispositivos de entrada gráfica atendiendo a la información que generan de forma directa. Esta puede ser:

- ▶ **posiciones 2D:** tableta digitalizadora, lápiz óptico, pantalla táctil
- ▶ **posiciones 3D:** palpador, digitalizador, tracking
- ▶ **desplazamientos 2D:** ratón, trackball, joystick

Dispositivos lógicos

La dependencia de la aplicación con el hardware de entrada se resuelve, en parte, trabajando con dispositivos abstractos, denominados **dispositivos lógicos**.

Cada dispositivo lógico representa un dispositivo físico ideal, que genera un único tipo de información de entrada. Cada dispositivo físico puede ser tratado como una realización de uno, o varios, dispositivos lógicos.

Los sistemas gráficos necesitan dos tipos de dispositivos específicos:

- ▶ **Locator**: lee posiciones
- ▶ **Pick**: lee identificadores de componentes del modelo geométrico

Procesamiento

Otro de los problemas relacionados con las operaciones de entrada es el de sincronizar las acciones del usuario con la aplicación.

La sincronización debe realizarse de forma distinta dependiendo de la función a realizar.

La sincronización se resuelve en los sistemas gráficos, haciendo uso de mecanismos de abstracción.

A las abstracciones del comportamiento se les denomina **modos de entrada**.

En cada momento, la aplicación puede asociar un modo a cada dispositivo lógico.

Hay tres modos de funcionamiento: **petición, evento y muestreo**.

Modos de entrada

El modo petición ('request') funciona como una lectura sin buffer de entrada.

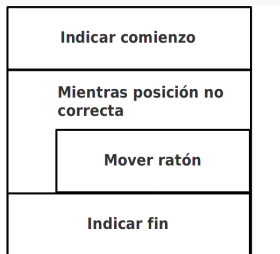
En el modo suceso ('event') utiliza un buffer, de modo que el usuario puede realizar entradas antes de que la aplicación las solicite.

El modo muestreo ('sample') tiene funcionamiento que no esta basado en la realización de operaciones individuales, sino en la asociación de un valor al dispositivo.

Posicionamiento

El objetivo de las operaciones de posicionamiento es permitir la introducción de un punto.

Desde el punto de vista del usuario, que utiliza un dispositivo que controla un cursor, el proceso de entrada es el que se muestra en la figura



Posicionamiento

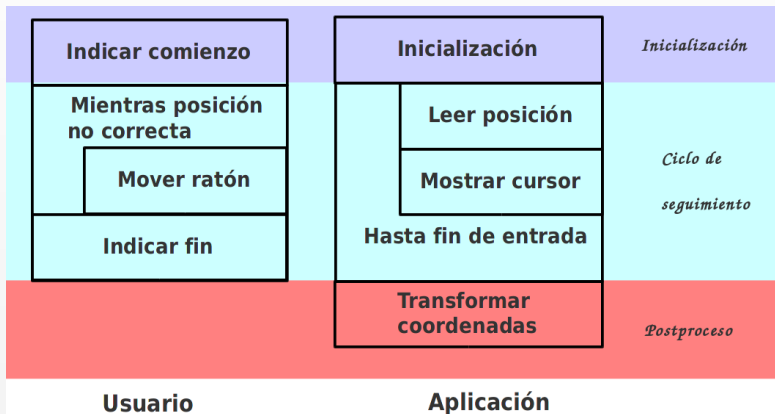
El proceso en el software de tratamiento se realiza en una estructura cíclica, que se corresponde con el ciclo de búsqueda que realiza el usuario. En este proceso podemos distinguir tres etapas:

- ▶ **inicialización**
- ▶ **ciclo de seguimiento**
- ▶ **postproceso**



Posicionamiento

Correspondencia entre operaciones del usuario y del sistema

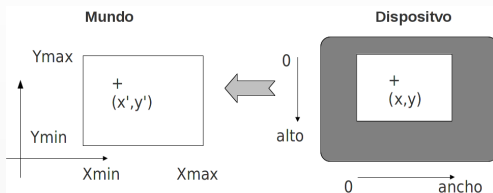


Conversión a coordenadas del mundo

La posición obtenida en la lectura está dada en coordenadas de dispositivo. Es necesario pasarla a coordenadas del mundo. Para ello se puede:

- ▶ Invertir la transformacin de visualización (si el modelo es 2D)
- ▶ Restringir el posicionamiento a un plano (calculando la posición sobre el plano)
- ▶ Utilizar un cursor 3D
- ▶ Utilizar vistas con tres proyecciones paralelas
- ▶ Invertir la transformacin de visualización (si el dispositivo de entrada es 3D)

Posicionamiento 2D



$$x' = X_{min} + x * (X_{max} - X_{min}) / ancho \quad (1)$$

$$y' = Y_{max} - y * (Y_{max} - Y_{min}) / alto \quad (2)$$

Los parámetros son los indicados a las funciones `glOrtho` y `Viewport`.

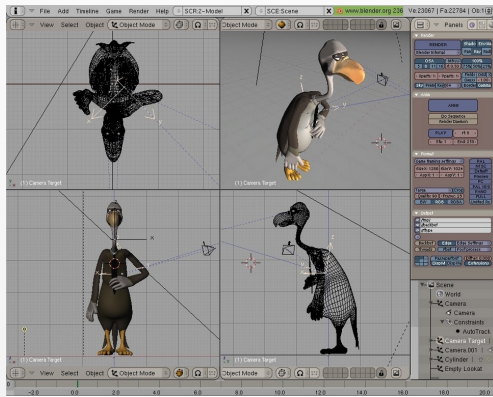
`glOrtho(Xmin,Xmax,Ymin,Ymax,Zmin,Zmax);`

`glViewport(x0,y0,ancho,alto);`

Posicionamiento 3D

Vistas con tres proyecciones paralelas

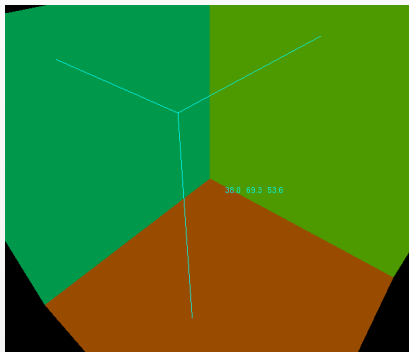
En cada vista se fijan dos coordenadas



Posicionamiento 3D: Cursor 3D

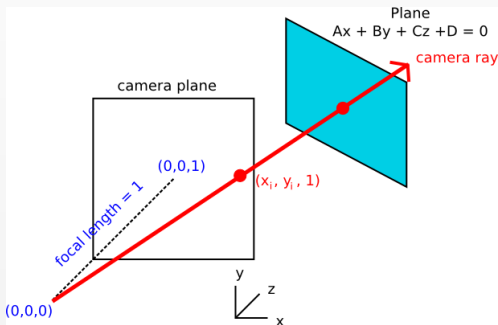
La realimentación es un cursor 3D

El desplazamiento se realiza en el plano X-Z o en el X-Y en función de los botones del ratón que estén pulsados.



Posicionamiento 3D

Si se restringe la posición a un plano que no sea perpendicular al de proyección se puede obtener una posición 3D por intersección de la recta que pasa por el punto introducido (en el plano de proyección y el centro de proyección con el plano



Entrada de transformaciones

Las transformaciones geométricas se pueden definir a partir de puntos.

- ▶ Una traslación se puede definir por el vector que va de la posición original a la posición nueva
- ▶ Una rotación por el ángulo formado por el vector que va del punto de referencia a la posición actual con la horizontal

Control de camara

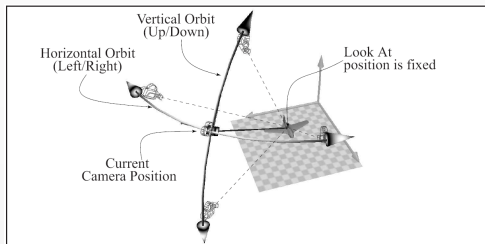
La cámara tiene demasiados parámetros para controlarlos todos interactivamente usando un ratón.

Habitualmente se controla de forma interactiva un número reducido de parámetros, seleccionados dependiendo del tipo de aplicación:

- ▶ visualización de modelos: posición
- ▶ exploración de escenarios: dirección

Cámara orbital

La cámara mira al objeto que se está visualizando, moviéndose en torno a él sobre una esfera.

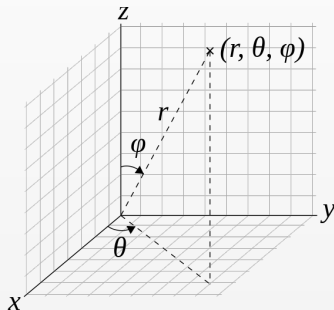


(K. Sung, P. Shirley, S. Baer: Essentials of Interactive Computer Graphics: Concepts and Implementation)

Cámara orbital

La cámara mira al objeto que se está visualizando, moviéndose en torno a él sobre una esfera.

La posición de la cámara puede especificarse en coordenadas esféricas.



V_{up} se debe calcular de forma que no sea perpendicular al plano proyección.

Cámara primera persona

La cámara está en un posición fija (un avatar o un personaje que se mueve en el escenario). El usuario puede controlar la dirección en la que mira la cámara.

La dirección se puede especificar como dos giros en direcciones ortogonales a la dirección de avance del personaje.

V_{up} se debe calcular de forma que no sea perpendicular al plano proyección.

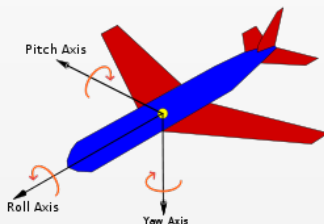
Una cámara en tercera persona sigue al personaje.

Ángulos de Euler

La orientación de la cámara (o la posición en una cámara orbital) se puede expresar como tres giros (uno respecto a cada eje).

Esta representación de la orientación se conoce como ángulos de Euler.

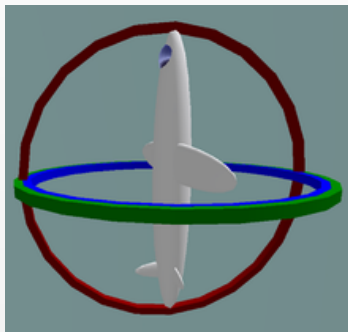
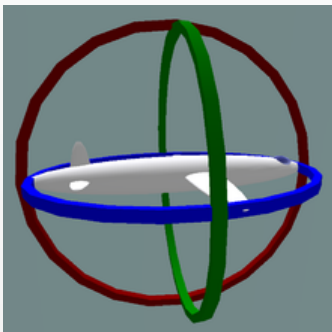
Permite controlar la orientación con tres grados de libertad (incluye control sobre V_{up}).



Gimbal lock

La utilización de ángulos de Euler tiene el problema de generar configuraciones en las que dos ejes se alinean provocando la pérdida de uno de los grados de libertad.

Este fenómeno se conoce como gimbal lock.

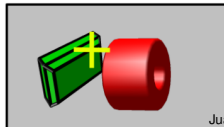
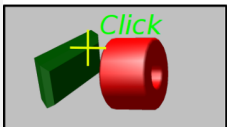
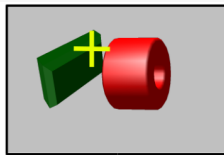
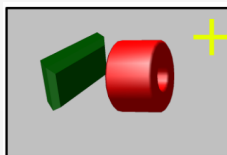


Selección

La selección permite al usuario referenciar componentes de un modelo geométrico.

Es esencial en cualquier aplicación gráfica que requiera la edición de un modelo.

El proceso de selección suele realizarse como un posicionamiento y una búsqueda en el modelo geométrico.



-> id: 23

Identificadores de objetos

Para poder realizar la selección los componentes de la escena deben tener asociados identificadores.

Los identificadores se pueden asociar a distinto nivel, permitiendo controlar el **ámbito** de la selección: vértice, primitiva, objeto.

Métodos de selección

En la selección se realiza una búsqueda en el modelo geométrico a partir de la información introducida por el usuario (usualmente una posición en pantalla).

La asociación entre posiciones de pantalla y elementos del modelo se puede establecer de varios modos:

- ▶ Identificando los objetos incluidos en un subvolumen de visión centrado en la posición dada (OpenGL)
- ▶ Intersección rayo escena
- ▶ Codificando el Id de objeto como color y leyendo del frame buffer

Animación

Generar animaciones implica:

- ▶ Regenerar la imagen periódicamente (al menos 25 veces por segundo)
- ▶ Modificar la pose de los objetos de la escena

Modificación de objetos

La modificación de los objetos de la escena puede realizarse usando distintas técnicas

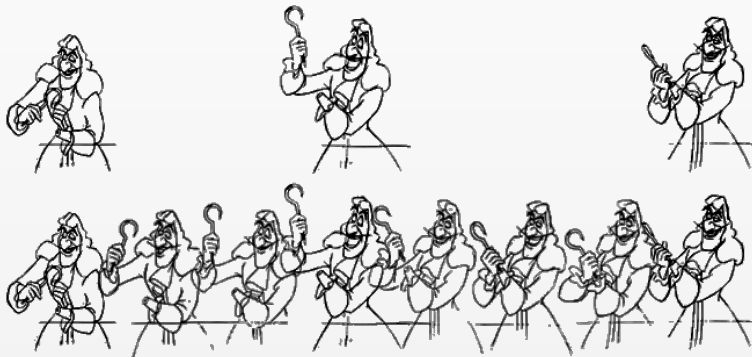
- ▶ Usando fotogramas clave (keyframe)
- ▶ Simulación física
- ▶ Controlando un esqueleto
- ▶ Proceduralmente

que se pueden combinar.

Keyframe

El animador define dos configuraciones del modelo.

El sistema calcula los fotogramas intermedios (in-betweens) por interpolación.



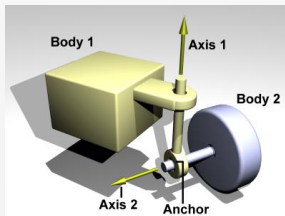
(c) Walt Disney Company, from "The Illusion of Life"

Simulación física

Para escenas con modelos físicos simples se puede calcular la configuración del escenario en cada fotograma usando las leyes de la mecánica clásica.

Existen librerías específicas para realizar esta simulación:

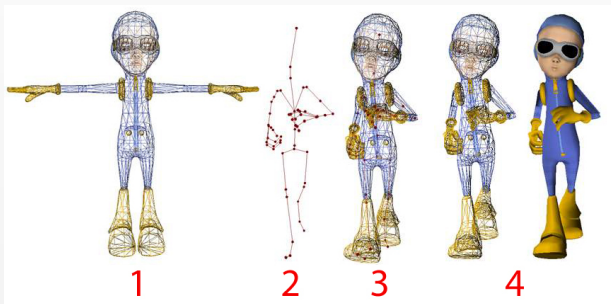
- ▶ ODE: Open Dynamic Engine.
- ▶ Newton: Newton Physics Engine
- ▶ Bullet: Physics Library



Esqueletos

Para conseguir que la animación de personajes resulte plausible se suelen usar esqueletos.

Un esqueleto es un modelo simplificado del personaje, formado por segmentos rígidos unidos por articulaciones.



Animación procedural

El comportamiento del objeto se describe mediante un procedimiento.

Es útil cuando el comportamiento es fácil de generar pero difícil de simular físicamente (p.e. la rotura de un vidrio).

