

InformaticaGraficatodalateoria.pdf



ajzave



Informática Gráfica



3º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de
Telecomunicación
Universidad de Granada



MÁSTER EN

Inteligencia Artificial & Data Management

MADRID

Formamos
talento para un futuro
Sostenible

saber más



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositista. Consulta más información en inglés.

ALFONSO JULIAN ZAPATA VELASCO

INFORMÁTICA GRÁFICA

Resúmenes de estudio del temario

1. TEMA: Introducción	1
• Conceptos generales	1
• Sistema gráfico	2
2. TEMA: Modelado de objetos	4
• Introducción	4
• Modelos de fronteras (B-rep) Boundary Representation	5
Suavizado Plano (Flat Shading)	7
Suavizado Gouraud	8
Suavizado Phong	8
• Transformaciones geométricas	10
• Generación de modelos de frontera	14
3. TEMA: Visualización de objetos	15
• Transformaciones de proyección	16
• Proceso de síntesis de imágenes en 3D	18
• Explicación de las diapositivas (26-31) Pregunta de examen posible	20
• Transformación de vista 3D	23
• Transformación de normalización:	29
• Algoritmo Z-buffer:	32
• Luz como onda partícula	34
• Texturas	38
4. TEMA: Interacción	39
• Introducción	39
• Posicionamiento	41
• Selección	42
• Animación	43

1. TEMA: Introducción

• Conceptos generales

Individuo real → El profesor

Modelo → Una figura en 3D del profesor

Consulta condiciones aquí



WUOLAH

Imagen → Un selfie del profesor

Modelo subyacente → Representación matemática o lógica de un objeto tridimensional o de una escena que se utiliza para crear imágenes y animaciones en ordenador

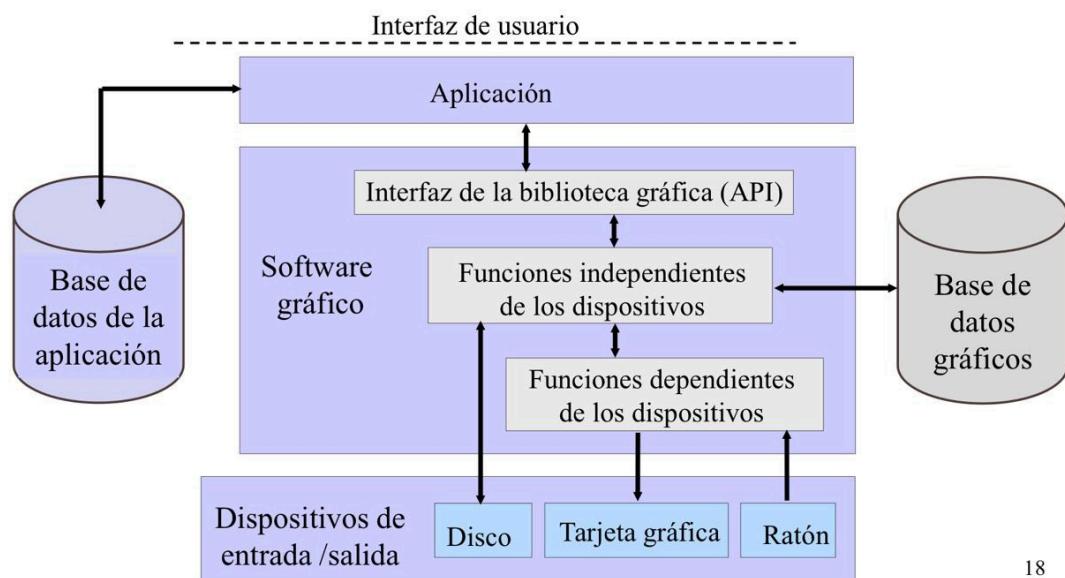
Dentro de un modelo (objeto sintético) podemos hacer una **representación vectorial o continua** (vértices, aristas, ...) o una **representación discreta o matricial** (Matriz bidimensional de elementos discretos, píxeles).

La primera, tendrá una **resolución infinita**, es decir, al hacer zoom no se deforma, mientras que la segunda tendrá una **resolución finita**.

La síntesis de un modelo en informática gráfica se refiere al proceso de generar imágenes o representaciones visuales a partir de un modelo subyacente tridimensional o de datos de entrada. Implica aplicar técnicas de renderizado, iluminación, sombreado y otras transformaciones para crear una representación visual del modelo en una pantalla o en un archivo de imagen.

La informática gráfica supone la creación, manipulación o modificación y almacenamiento de objetos (o modelos) e imágenes. Es interactiva, es decir, se pueden cambiar las características de los objetos.

● Sistema gráfico



18

Una librería gráfica debe reunir las siguientes funciones:

- Creación y edición de modelos geométricos
- Síntesis
- Entradas gráficas

Existen dos **sistemas de coordenadas** principales:

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](#)

Que te den 10 € para gastar
es una fantasía.
ING lo hace realidad.

Abre la **Cuenta NoCuenta** con el código
[WUOLAH10](#), haz tu primer pago y llévate 10 €.

Quiero el cash

[Consulta condiciones aquí](#)



do your thing

Informática Gráfica



Comparte estos flyers en tu clase y consigue más dinero y recompensas



- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanear y acceder a apuntes
- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR

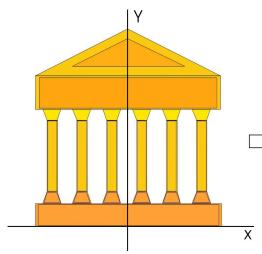
Banco de apuntes de la

WUOLAH

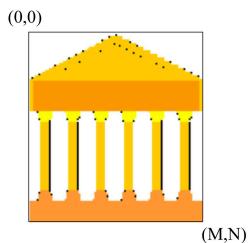


2D

- Coordenadas mundiales:

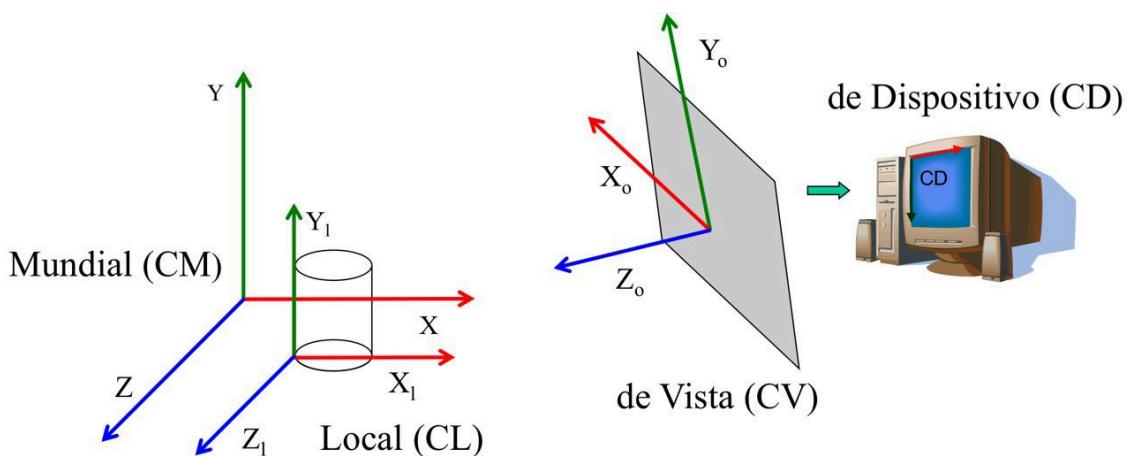
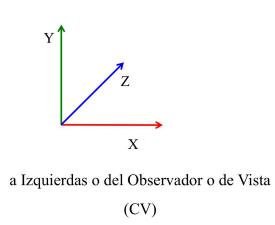
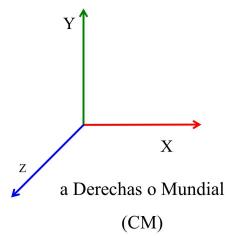


- Coordenadas de dispositivo:



3D

- Mundial o a derechas(CM)
- De vista o del observador o a izquierdas (CV)



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](#).

Primitivas → Las primitivas son las funciones de salida imprescindibles o básicas para construir los datos gráficos de un modelo, es decir, la representación más básica sobre la que trabajamos, puntos, líneas y polígonos. cuadrado (**x, y, azul, borde**)

Elementos más complejos se crearán mediante bloques de estas primitivas. La apariencia o el aspecto de estas primitivas está controlada por los atributos. Hay dos filosofías en la especificación de atributos:

- Como parámetro de una llamada a una función de salida cuadrado (**x, y, azul, borde**)
- Fijados mediante una colección separada de funciones cuadrado (**x, y**)
color(**azul**)
borde(**borde**)

Las primitivas con las que trabajaremos son los puntos, con atributos de estilo, tamaño y color, las líneas (conjunto de vértices conectados por líneas rectas) con atributos de estilo, tamaño y color y los polígonos y los polígonos (conjuntos de caras planas), con atributos del contorno, estilo, tamaño y color y con atributos del interior, hueco, sólido, textura, trama.

2.TEMA: Modelado de objetos

● Introducción

El **modelado geométrico** consiste en la construcción de un modelo matemático de la forma de un objeto físico, tanto bidimensional como tridimensional.

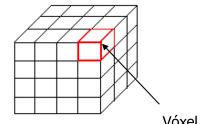
Un **modelo geométrico computacional** será aproximado ya que no se puede representar en un ordenador toda la complejidad de un objeto real.

Existen modelos computacionales de: Curvas, superficies, sólidos...

Un sólido modelado por las fronteras (B-rep) Boundary Representation “Representación de límites”, se forma por un conjunto de caras planas (polígonos) que delimitan su superficie. El número de vértices puede ser cualquiera. Lo normal es considerar mallas formadas por triángulos.



Un sólido modelado por enumeración espacial se representa por un conjunto de volúmenes (vóxeles) que permiten modelar su interior. La



Vóxel

Consulta condiciones aquí



do your thing

WUOLAH

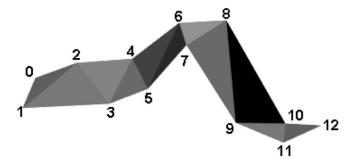
representación de un sólido con véxeles es mediante una matriz tridimensional

● Modelos de fronteras (B-rep) Boundary Representation

Se representan usando mallas de polígonos (caras planas).

El número de vértices puede ser cualquiera. Lo normal es considerar mallas formadas por triángulos.

La malla es una “tira de triángulos”. Cada triángulo de la tira comparte una arista y dos vértices que en principio no hay que compartir.

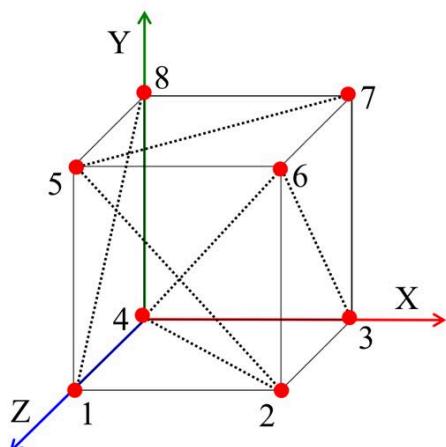


La información la almacenamos por separado de la geometría (vértices) y de la topología (caras y aristas).

- Lista de vértices: Una entrada por cada vértice, por tanto no se repiten.
- Lista de aristas: Una entrada por arista que contiene los índices a los vértices extremos de la arista.
- Lista de caras: Una entrada por triángulo que contiene los índices a los vértices correspondientes de la lista de vértices.

Permite visualizar en modo de alambre de forma eficiente.

• Ejemplo



Lista de triángulos	Lista de vértices	Lista de aristas
1,2,5	1 x_1, y_1, z_1	1,2
2,6,5	2 x_2, y_2, z_2	2,3
2,3,6	3 x_3, y_3, z_3	3,4
3,7,6	4 x_4, y_4, z_5	4,1
3,4,7	5 x_5, y_5, z_5	1,5
4,8,7	6 x_6, y_6, z_6	2,6
4,1,8	7 x_7, y_7, z_7	3,7
1,5,8	8 x_8, y_8, z_8	4,8
5,6,7		5,6
5,7,8		6,7
4,2,1		7,8
4,3,2		8,5

Implementación y visualización de las mallas:

La construcción de los modelos ha de estar separada de la visualización.

Tipos de datos básicos (usando C)

```
struct vertex
{
    // Tres valores reales que representan un vértice 3D
```

```

    float coord[3];           // Acceder con: coord[0], coord[1], coord[2]
};

struct face
{
    int innd_ver[3];          // Tres valores reales que representan un vértice 3D
};
```

- **Dibujar en modo punto. Válido en todos los casos menos para listas de triángulos aislados**

```

void DrawPoints (vertex *puntos, int n, float r, float g, float b, int size)
{
    int i;
    glColor3f(r, g, b);
    glPointSize(size);
    glBegin(GL_POINTS);
    for( i = 0; i < n; i++){
        glVertex3f(puntos[i].coord[0], puntos[i].coord[1], puntos[i].coord[2]);
    }
    glEnd();
```

También podemos llevar a cabo una **malla de triángulos aislados**, una **malla de vértices y triángulos** o una **malla de vértices, aristas y triángulos**, además de otros modelos de visualización como el “**ajedrez**”.

Para facilitar la codificación con C++ se pueden usar:

- Plantillas (templates)
- STL (Standard Template Library)

```
_vertex3f Vertex;
Vertex.x = 0;
Vertex.y = 5;
Vertex.z = 10;
```

// Estructuras para una malla de vértices, aristas y puntos

```
vector<_vertex3f> Vertices;           // Lista de vértices
vector<_vertex3i> Triangulos;         // Lista de triángulos
vector<_vertex2i> Aristas;            // Lista de aristas
```

- **Visualización en modo puntos con la STL**

```
...
int i;

...
glBegin(GL_POINTS);
for(i = 0; i < Vertices.size(); i++)
```

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.

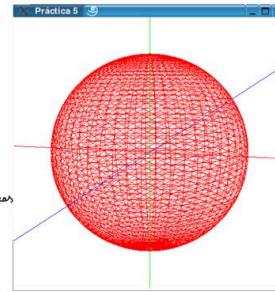


```
glVertex3f (Vertices[i].X, Vertices[i].y, Vertices[i].z);  
);  
glEnd();
```

- STL vector

```
// Visualizando en modo alambre
```

```
...  
int Vertex_1,Vertex_2, i; // Se almacena para almacenar los índices de los vértices  
...  
glBegin( GL_LINES ); // Indica el inicio de una serie de llamadas a 'glVertex' que definen líneas.  
for( i= 0 ; i < Aristas.size() ; i++ )  
{  
    Vertex_1 = Aristas[i]._0 ;  
    Vertex_2 = Aristas[i]._1 ;  
    glVertex3f( Vertices[Vertex_1].x,  
    Vertices[Vertex_1].y,Vertices[Vertex_1].z ) ; } }  
    glVertex3f( Vertices[Vertex_2].x,  
    Vertices[Vertex_2].y,Vertices[Vertex_2].z ) ; }  
}  
glEnd() ;
```

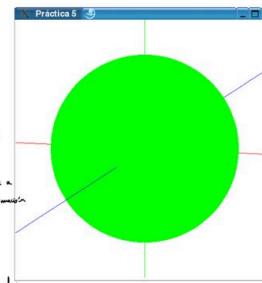


2

- STL vector

```
// Visualizando en modo sólido
```

```
...  
int Vertex_1,Vertex_2,Vertex_3; // Se almacenan los índices de los 3 vértices  
...  
glBegin( GL_TRIANGLES ); // Tercer de los tipos de triángulos. GL_TRIANGLES es el modo que le dice a OpenGL que debe conjuntar los vértices consecutivos que se especifican a continuación.  
for ( i= 0 ; i < Triangulos.size() ; i++ )  
{  
    Vertex_1 = Triangulos[i]._0 ;  
    Vertex_2 = Triangulos[i]._1 ; } }  
    Vertex_3 = Triangulos[i]._2 ;  
    glVertex3f( Vertices[Vertex_1].x, Vertices[Vertex_1].y, Vertices[Vertex_1].z ); } }  
    glVertex3f( Vertices[Vertex_2].x, Vertices[Vertex_2].y, Vertices[Vertex_2].z ); } }  
    glVertex3f( Vertices[Vertex_3].x, Vertices[Vertex_3].y, Vertices[Vertex_3].z ); } }  
}  
glEnd() ;
```



30

Suavizado Plano (Flat Shading)

Es la técnica más simple y rápida de las tres. Aquí, cada polígono de un objeto 3D (usualmente un triángulo o cuadrilátero) se rellena con un solo color o tono. El color se calcula generalmente basándose en la dirección de una única normal al polígono y una fuente de luz. Como resultado, no hay transición suave entre polígonos adyacentes, por lo

Consulta condiciones aquí



WUOLAH

que se pueden ver claramente los bordes y la malla del objeto, dando un aspecto facetado, como un diamante.

Suavizado Gouraud

Mejora el suavizado plano al calcular el color no solo para cada polígono, sino para cada vértice. Luego, el color se interpola a lo largo de la superficie del polígono. Esto significa que si dos triángulos adyacentes tienen diferentes colores en un vértice compartido, el color se mezcla suavemente a lo largo de la arista que comparten. Esto da como resultado una transición más suave y menos apariencia de facetas en comparación con el suavizado plano. Sin embargo, puede no manejar bien los reflejos especulares ya que no considera el espectador o la dirección de la vista para calcular la iluminación.

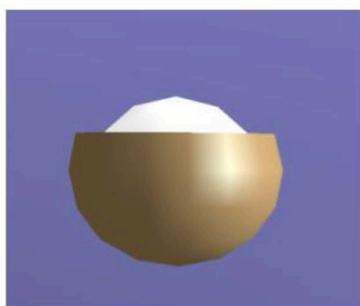
Suavizado Phong

Es un método más sofisticado y computacionalmente intensivo. A diferencia del Gouraud, el suavizado Phong interpola las normales de los vértices a lo largo de la superficie del polígono y calcula el color para cada píxel individualmente. Esto tiene en cuenta la dirección de la luz, la posición del espectador y las propiedades del material para cada píxel, lo que resulta en efectos de iluminación más precisos y detallados, como los reflejos especulares brillantes. Es más realista que el Gouraud, especialmente en superficies brillantes o reflectantes, pero también requiere más recursos computacionales.

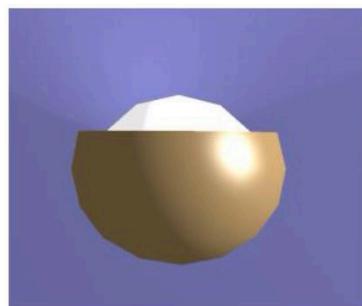
- Cálculo de normales, a utilizar en iluminación
 - Normal a las caras: suavizado plano
 - Normal a los vértices: suavizado de Gouraud o Phong



Suavizado plano



Suavizado Gouraud

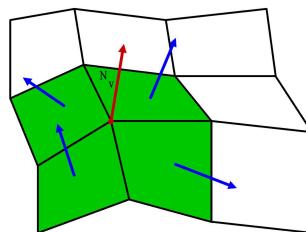
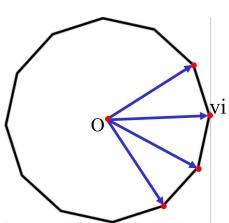


Suavizado Phong

El suavizado plano es verosímil para objetos que se puedan modelar con caras planas, pero no para objetos curvos (suaves).

Hay dos formas de cálculo de la normal a los vértices:

- A partir de la forma de un objeto (Desde el centro hasta el vértice)
- Una normal media de las normales de las caras que se reúnen en un vértice dado



• Aristas aladas (winged-edges, Baumgart72)

Esta estructura pone el énfasis en las aristas en lugar de las caras.

Para cada arista tenemos:

- Sus dos vértices
- Las dos caras adyacentes a la arista (cara derecha e izquierda)
- Las aristas predecesoras y sucesoras en las caras adyacentes

“El modelo de “Aristas Aladas” (Winged-Edges) es una estructura de datos en la computación gráfica diseñada para almacenar información sobre un modelo poligonal, como una malla 3D. A diferencia de otras estructuras que pueden enfocarse en los vértices o las caras, el modelo de Aristas Aladas se centra en las relaciones entre aristas, lo que facilita la navegación y manipulación de la geometría de la malla.”

El modelo de Aristas Aladas representa cada arista con la siguiente información:

- **Sus dos vértices:** Cada arista tiene dos vértices en sus extremos. Estos son puntos en el espacio 3D que definen la posición de la arista.
- **Las dos caras adyacentes a la arista:** Cada arista es compartida por dos caras, que en la estructura de Aristas Aladas son referenciadas como la cara derecha e izquierda. Estas caras son las superficies planas o curvas que forman la malla.
- **Las aristas predecesoras y sucesoras en las caras adyacentes:** Para cada una de las dos caras adyacentes, la estructura almacena referencias a las aristas que preceden y suceden a la arista actual en el orden de las aristas que definen la cara. Esto significa que para cada cara, se sabe qué arista viene antes y cuál viene después, permitiendo un recorrido ordenado alrededor de la cara.

El nombre “Aristas Aladas” viene de la forma en que se visualiza esta estructura de datos: si imaginas cada arista como el centro de una mariposa, las caras serían las alas y las aristas conectadas a los extremos serían las puntas de las alas.

La ventaja de usar el modelo de Aristas Aladas es que **facilita ciertas operaciones geométricas y topológicas**. Por ejemplo, es fácil encontrar el anillo de caras alrededor de un vértice o seguir el borde de un hueco en la malla. También es útil para operaciones como subdivisión de mallas, detección de bordes, y para algoritmos que requieren caminar alrededor de una malla poligonal de manera eficiente.”

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

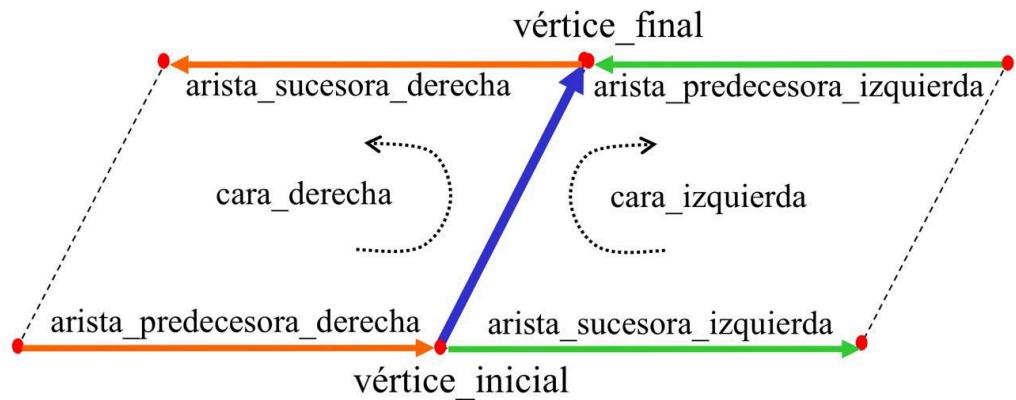
Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.



● Transformaciones geométricas

El movimiento de cualquier objeto que expresa mediante transformaciones geométricas. El manejo de instancias de objetos gráficos desempeñan un papel muy importante. Una instancia es un objeto gráfico definido en su propio sistema de coordenadas cartesianas.

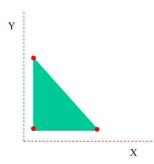
Una transformación geométrica cambia la geometría (los vértices) no la topología (las caras).

Un objeto se puede considerar como un conjunto de puntos. Las transformaciones geométricas serán operaciones para calcular nuevas posiciones de los puntos en un objeto.

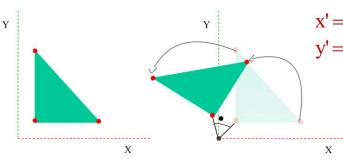
Tipos de transformaciones geométricas:

- Traslación → Mueve un objeto de un lugar a otro sin cambiar su orientación ni tamaño. Por ejemplo, desplazar una silla a lo largo de una habitación.
- Rotación → Gira un objeto alrededor de un eje fijo. Piensa en girar una llave dentro de una cerradura.
- Escalado → Cambia el tamaño de un objeto, agrandándolo o reduciéndolo. Imagina que estás ajustando el tamaño de una fotografía digital con una función de zoom.
- Deformación (cizalla) → Inclina las formas de un objeto, alterando su proporción pero sin modificar su área o volumen. Piensa en empujar la parte superior de una caja hasta que se convierta en un paralelogramo.
- Reflexión (caso particular del escalado) → Voltea un objeto como si se mirara en un espejo. Un ejemplo sería ver la letra 'b' convertirse en 'd' cuando se refleja horizontalmente.

• Translación



• Rotación con ángulo en el sentido contrario a las agujas del reloj



$$\begin{aligned}x' &= x + t_x \\y' &= y + t_y\end{aligned}$$

$$\begin{aligned}x' &= x \cos \theta - y \sin \theta \\y' &= y \cos \theta + x \sin \theta\end{aligned}$$

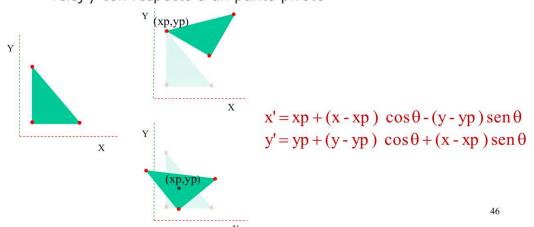
Consulta condiciones aquí



do your thing

WUOLAH

- Rotación con ángulo en el sentido contrario a las agujas del reloj y con respecto a un punto pivote

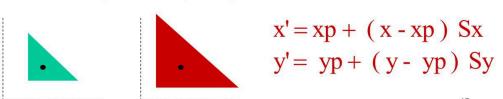


- Escalado con respecto al origen



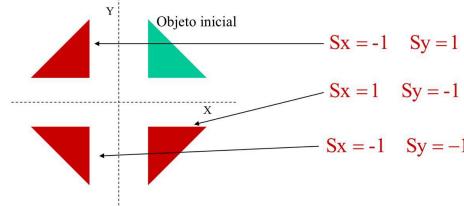
47

- Escalado con respecto a un punto pivote

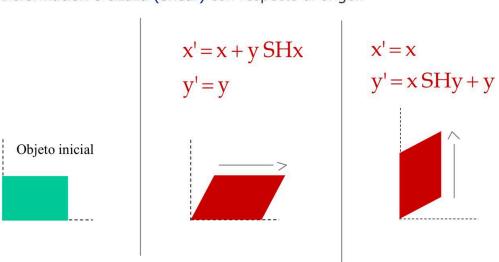


- Reflexión (Flip). Caso particular del escalado "incluido un eje negativo"

$$\begin{aligned} x' &= x \cdot S_x \\ y' &= y \cdot S_y \end{aligned}$$



- Deformación o cizalla (Shear) con respecto al origen



Al conjunto de transformaciones geométricas que se aplican a un objeto se le denomina transformación de modelado.

Representación matricial

Las reglas para componer ecuaciones son complejas si se usan funciones.

Se usa la representación matricial para las transformaciones. Componer transformaciones equivale a multiplicar matrices.

Se emplean las coordenadas homogéneas, en las que un punto se representa por el vector (x', y', h) (2D) o por (x', y', z', h) (3D) y se usan matrices 3×3 (2D) o 4×4 (3D) para representar las transformaciones.

Para no realizar normalizaciones se toma $h = 1$ y la última columna de las matrices 3×3 como $(0, 0, 1)$ o bien $(0, 0, 0, 1)$ para las matrices 4×4 .

"Las coordenadas homogéneas son una forma de representar puntos en un espacio proyectivo que se utilizan comúnmente en gráficos por computadora para facilitar las transformaciones geométricas. La "h" en la representación de un punto (x', y', h) en 2D o (x', y', z', h) en 3D es la coordenada homogénea que permite incluir transformaciones de traslación en las operaciones matriciales."

En coordenadas cartesianas normales, solo podemos usar transformaciones lineales, como la rotación o el escalado, mediante multiplicación de matrices. Sin embargo, la translación no es una transformación lineal y no puede ser representada como una multiplicación de matrices en coordenadas cartesianas. Aquí es donde las coordenadas homogéneas son útiles: permiten representar todas las transformaciones geométricas, incluida la translación, como multiplicaciones de matrices.

La coordenada "h" es un factor de escala que se utiliza para convertir las coordenadas homogéneas de vuelta a las coordenadas cartesianas. Por ejemplo, un punto en coordenadas homogéneas (x', y', z', h) se convierte a coordenadas cartesianas (x, y, z) dividiendo cada componente por "h": $(x'/h, y'/h, z'/h)$.

En la práctica, normalmente se establece "h" a 1 para simplificar los cálculos. Esto significa que no se requiere la normalización (división por "h") después de aplicar una transformación, porque el factor de escala no cambiará. Además, en las matrices de transformación 3x3 (para 2D) o 4x4 (para 3D), la última columna se puede configurar a (0, 0, 1) o (0, 0, 0, 1) respectivamente, lo que corresponde a no aplicar ninguna transformación adicional en la coordenada "h".

3. Transformaciones geométricas

3.2. Representación matricial

$$(x, y, z) \begin{pmatrix} x, y, z \\ x', y', z' \\ x'', y'', z'' \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \begin{pmatrix} x & x' & x'' \\ y & y' & y'' \\ z & z' & z'' \\ 1 & 1 & 1 \end{pmatrix}$$

- Hay dos representaciones para las matrices:

- A) si un punto se considera como vector fila $(x, y, 1)$ se multiplica por la izquierda de la matriz
- B) si un punto se considera como columna $(x, y, 1)^T$ se multiplica por la derecha

$$T_{tx,ty} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{pmatrix}$$

$$S_{sx,sy} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_\theta = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$T^t_{tx,ty} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$$

$$S^t_{sx,sy} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Forma A

Forma B

52

La diapositiva presenta cómo las transformaciones geométricas pueden ser representadas y aplicadas utilizando matrices en la computación gráfica. Se explica que hay dos maneras de utilizar las coordenadas homogéneas de un punto (x, y) para la multiplicación por estas matrices: como vector fila (Forma A) o como vector columna (Forma B).

Aquí está lo que cada matriz representa y cómo se compone en ambas formas:

Transformación de Translación (T)

Forma A: La matriz se coloca a la derecha del vector fila $(x, y, 1)$

Forma B: La matriz se coloca a la izquierda del vector columna $(x, y, z)^T$ “donde T denota la transposición”

La matriz contiene los valores t_x t_y en la primera fila o columna respectivamente, que representan cuánto se moverá el objeto en las direcciones x e y.

Transformación de Rotación (R)

Forma A: La matriz de rotación se coloca a la derecha del vector fila.

Forma B: La matriz de rotación transpuesta se coloca a la izquierda del vector columna.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositista. Consulta más información en [ing.es](#)



Las matrices utilizan $\cos(\alpha)$ y $\sin(\alpha)$ para definir la rotación alrededor del origen por un ángulo(α).

Los signos cambian en la transposición para mantener la misma rotación.

Transformación de Escalado (S)

Forma A: La matriz de escalado se coloca a la derecha del vector fila.

Forma B: La matriz de escalado transpuesta se coloca a la izquierda del vector columna.

Los valores Sx y Sy son los factores de escala en las direcciones x e y, respectivamente.

Transformación de Deformación o Cizalla (SH)

Forma A: La matriz de cizalla se coloca a la derecha del vector fila.

Forma B: La matriz de cizalla transpuesta se coloca a la izquierda del vector columna.

Los valores SHx y SHy indican cuánto se desplazarán las coordenadas x y en la dirección de la otra coordenada, creando un efecto de inclinación o deslizamiento.

En todos los casos, el elemento inferior derecho de la matriz es 1, lo que permite que la coordenada homogénea h se mantenga en 1 tras la transformación, simplificando así el cálculo y evitando la necesidad de normalizar las coordenadas después de aplicar la transformación.

Consulta condiciones aquí

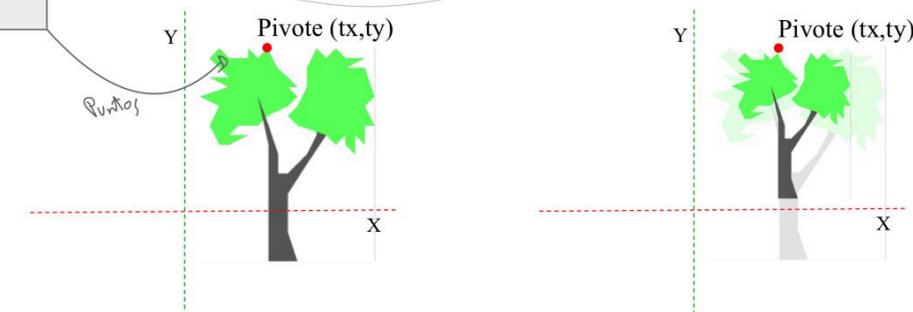


do your thing

WUOLAH

- Ejemplo de escalado con respecto a un punto pivote, con la representación A)

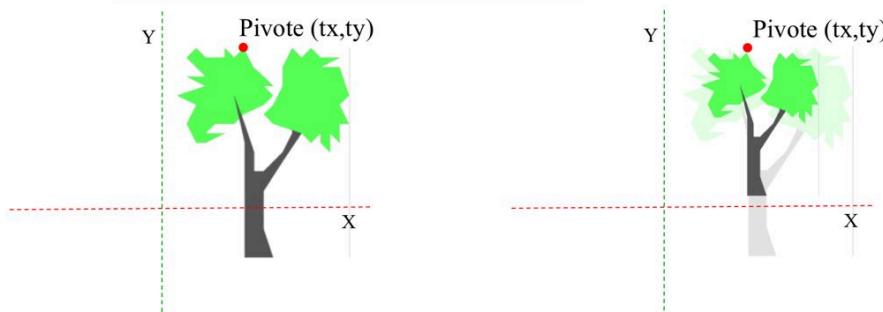
$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \begin{pmatrix} y_1 & 1 \\ \vdots & \vdots \\ y_n & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -tx & -ty & 1 \end{pmatrix} \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tx & ty & 1 \end{pmatrix} = \text{Objeto} \times T_{-tx,-ty} \times S_{sx,sy} \times T_{tx,ty} \equiv \text{Objeto} \times M$$



53

- Ejemplo de escalado con respecto a un punto pivote, con la representación B)

$$\begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 & \dots & x_n \\ y_1 & \dots & y_n \\ 1 & \dots & 1 \end{pmatrix} = T_{tx,ty}^t \times S_{sx,sy}^t \times T_{-tx,-ty}^t \times \text{Objeto} \equiv M \times \text{Objeto}$$



54

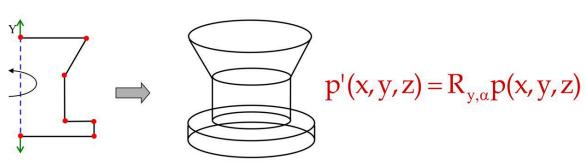
Toda transformación digital tiene inversa. Fácil de calcular con matrices.

Las transformaciones de coordenadas son inversas a las transformaciones geométricas.

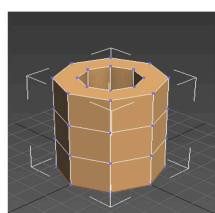
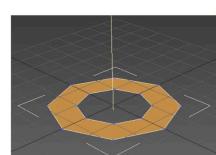
● Generación de modelos de frontera

● Revolución y extrusión

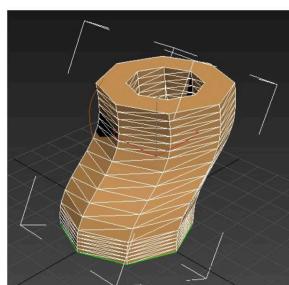
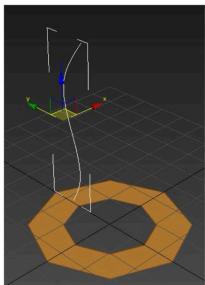
- Extrusión según una recta



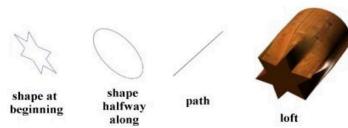
$$p'(x, y, z) = T_{tx,ty,tz} p(x, y, z)$$



- Extrusión según una curva



- Combinación de perfiles



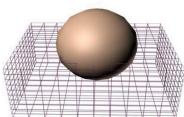
- Extrusión local



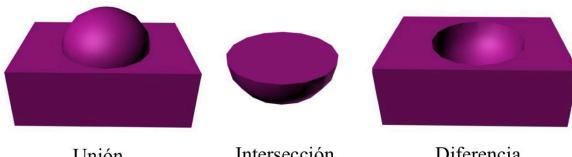
• Operaciones booleanas



Objeto 1



Objeto 2



Unión

Intersección

Diferencia

• Geometría constructiva de sólidos (CSG)

Técnica de modelado en la que se crean formas complejas combinando objetos sólidos simples mediante operaciones booleanas como la unión, intersección y diferencia.

• Modelos jerárquicos

Permiten la creación de modelos complejos a partir de modelos más simples.

Tipos de modelos jerárquicos:

- Objetos compuestos
- Objetos articulados
- Escenas formadas por varios objetos

Se representan mediante grafos dirigidos acíclicos (Directed Acyclic Graph o DAG). No hay un tipo de grafo normalizado.

- En las hojas tenemos objetos simples.
- En los nodos transformaciones geométricas e instancias o bien un objeto y en los arcos transformaciones geométricas.

Para la realización de estos grafos es conveniente utilizar las coordenadas maestras.
(Colocar al menos uno de los vértices en el (0,0,0)).

3.TEMA: Visualización de objetos

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene una cobertura máxima de hasta 100.000 euros por depositante. Consulta más información en [ing.es](#)



● Transformaciones de proyección

Permiten transformar un vector de un espacio de dimensión N a otro de dimensión N-1. En informática gráfica, los objetos tridimensionales se proyectan a un espacio bidimensional.

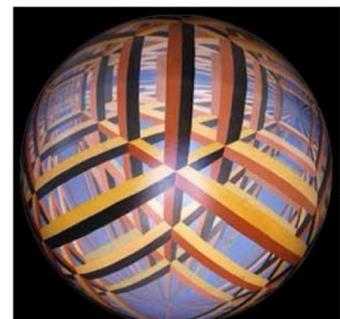
Hay dos clases de proyecciones:

- Planar → Proyectan objetos 3D en un plano 2D, como una sombra proyectada sobre una pared plana, manteniendo la uniformidad y sin distorsión por perspectiva.
- No planar → Utilizan superficies curvas o métodos más complejos para proyectar, como la proyección en un globo o en un cilindro, lo que puede incluir distorsión para representar la profundidad o la curvatura del espacio.

Planar

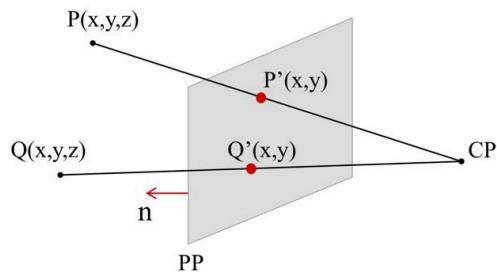


No planar

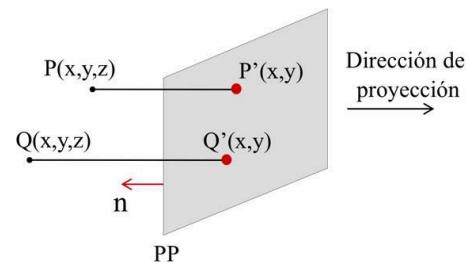


Clasificación de las proyecciones:

Las proyecciones geométricas planares se dividen en dos grandes grupos: perspectiva y paralelas. Esta distinción se relaciona con la posición del centro de proyección. Para especificar una proyección paralela, damos una dirección y para la perspectiva un punto.



Perspectiva



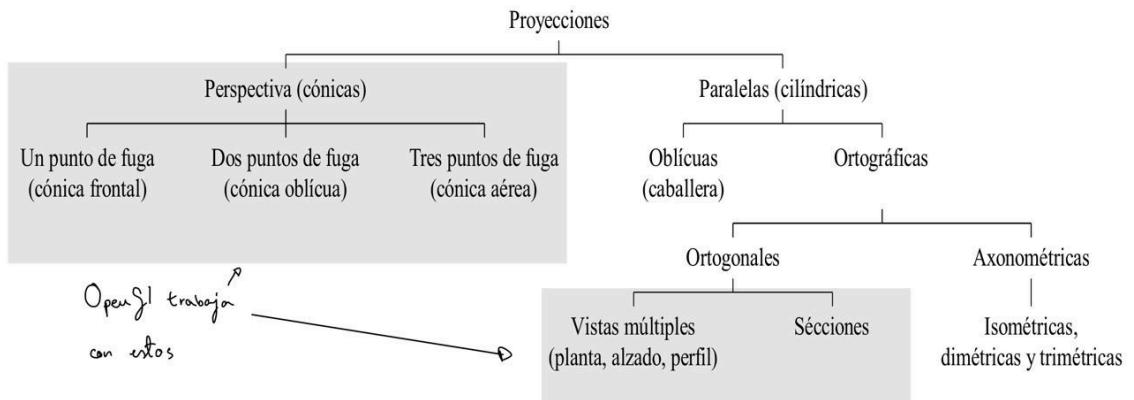
Paralela

Consulta condiciones aquí



do your thing

WUOLAH



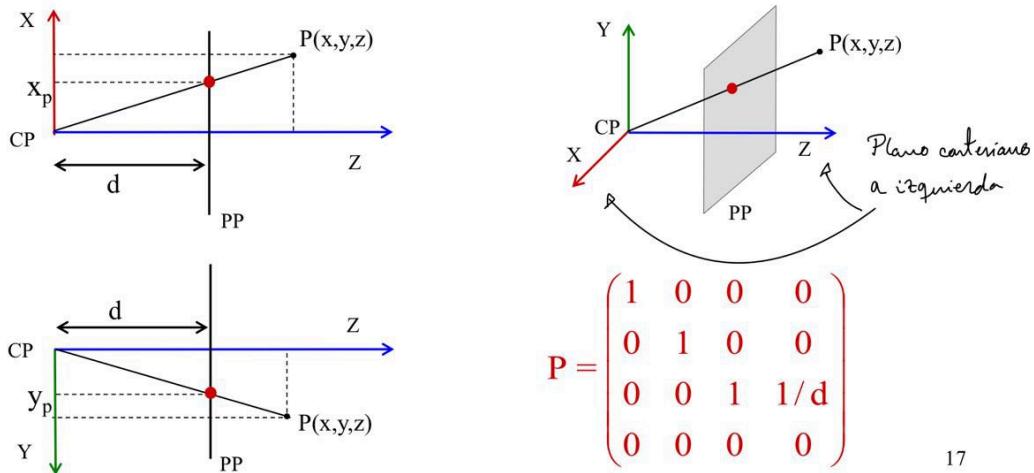
● Proyección en perspectiva.

Produce la ilusión de disminución en el tamaño de los objetos cuando nos vamos alejando del centro de proyección. No se pueden tomar medidas, ángulos y distancias exactas, salvo ángulos en caras paralelas al plano de proyección.

La matriz de transformación para un punto de fuga se puede calcular de dos formas respecto a las coordenadas del observador:

- A) Plano de proyección en $z=d$, centro de proyección en $z=0$
- B) Plano de proyección en $z=0$, centro de proyección en $z=-d$

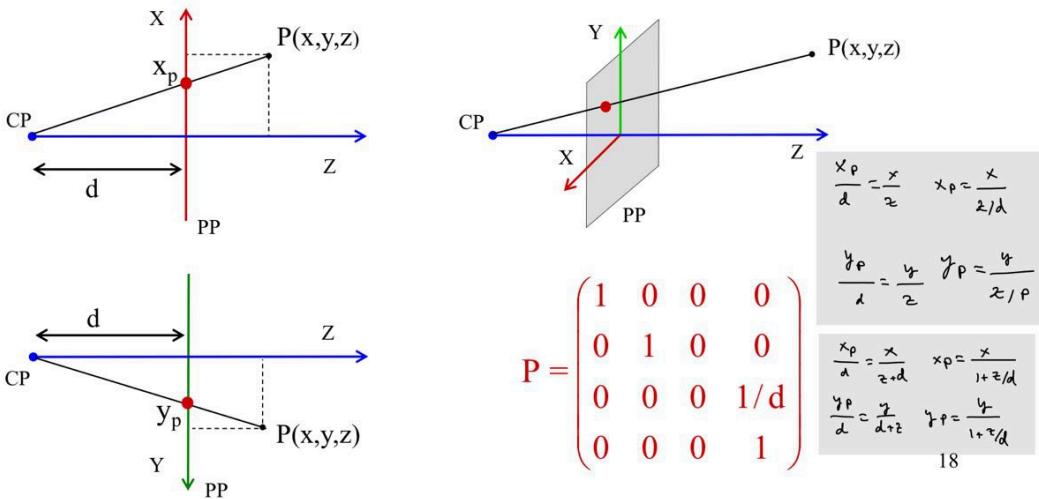
• A) Plano de proyección en $z=d$, centro de proyección en $z=0$



17

1.2. Proyección en perspectiva

- B) Plano de proyección en $z=0$, centro de proyección en $z=-d$



Las matrices de perspectiva con dos y tres puntos de fuga se obtienen aplicando rotaciones.

Caballera

Permite medir ángulos y distancias para las caras paralelas al plano de proyección y solamente distancias con un cierto factor de escala para las restantes caras.

Vistas múltiples (triédricas)

Permiten medir tanto ángulos como distancias directamente sobre los dibujos, pero es difícil deducir las características 3D de los objetos.

● Proceso de síntesis de imágenes en 3D

Cámara:

Una cámara es un objeto de la escena que se puede ubicar en cualquier posición para crear una imagen de la escena. Es como si un observador se situase en la escena para captar una imagen a través de una proyección de la escena sobre un plano.

Una cámara virtual ha de comportarse de forma similar a las reales.

A la hora de definir una cámara virtual debemos decir:

- Situación o ubicación de la cámara en el espacio → ¿Desde dónde miramos?
- Punto de mira o centro de atención → ¿Hacia dónde miramos?
- Cantidad de escena que podemos ver → ¿Cuánto ángulo abarcamos?
- Tipo de proyección (paralela o perspectiva).

Proceso de síntesis de imágenes en 3D.

Etapas:

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

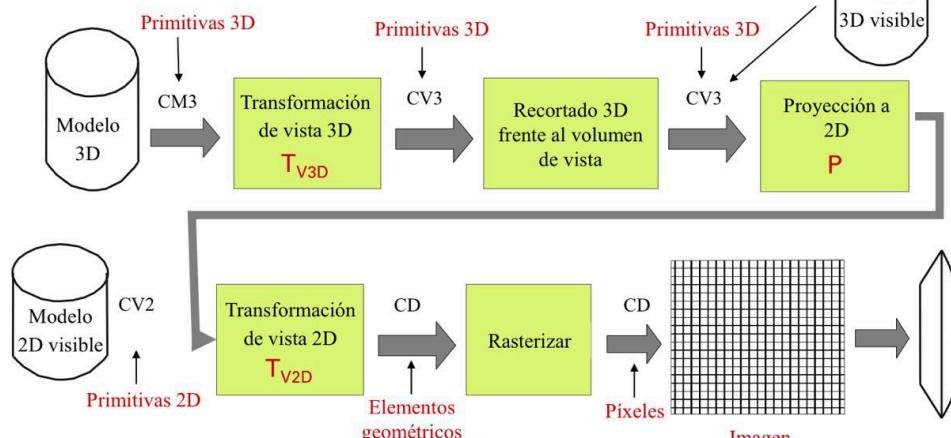
ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.



1. *Modelo 3D y sus primitivas:* Se comienza con un modelo tridimensional compuesto de primitivas geométricas (como vértices, aristas y caras).
2. *Transformación de vista 3D (T_{V3D}):* El modelo se transforma desde un sistema de coordenadas mundial 3D (CM3) a un sistema de coordenadas de la vista 3D (CV3), que depende de la posición y orientación del observador (cámara).
3. *Recortado 3D frente al volumen de vista:* Se realiza un recorte de las partes del modelo que no están dentro del volumen de vista, es decir, se descartan las partes que el observador no puede ver.
4. *Proyección a 2D (P):* El modelo 3D visible se proyecta en un espacio 2D para crear una representación bidimensional, como la imagen que se mostrará en pantalla.
5. *Modelo 2D visible y sus primitivas:* El modelo ahora en 2D tiene sus propias primitivas geométricas (como puntos y líneas).
6. *Transformación de vista 2D (T_{V2D}):* Se transforma el modelo 2D a un sistema de coordenadas dispositivo (CD), que está alineado con los píxeles de la pantalla.
7. *Rasterizar:* Los elementos geométricos 2D se convierten en píxeles, un proceso conocido como rasterización, donde se decide qué píxeles deben iluminarse y con qué color.
8. *Imagen:* El resultado final es una imagen rasterizada que puede ser mostrada en una pantalla compuesta por una cuadrícula de píxeles.

2. Proceso de síntesis de imágenes en 3D

2.2. Etapas



CM3=Coordenadas mundiales 3D

CV3=Coordenadas de vista 3D (observador)

CV2=Coordenadas mundiales 2D

CD=Coordenadas dispositivo

24

Volúmen de vista 3D (perspectiva)

Consulta condiciones aquí



WUOLAH

Parámetros:

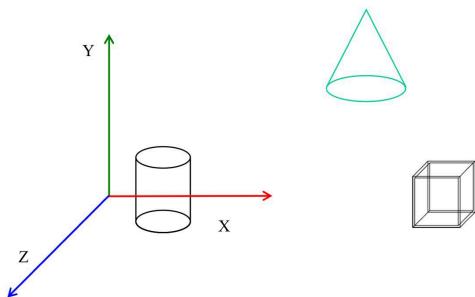
- Punto en el plano de proyección (PO). En coordenadas mundiales 3D
- Punto de mira (PM) o posición a la que apunta la cámara. En coordenadas mundiales 3D
- Normal al plano de proyección (PM-PO)
- Vectores de inclinación de la cámara (VI). En coordenadas mundiales.
- Centro de proyección (CP) o posición de la cámara. En coordenadas mundiales 3D.
- Planos de recorte frontal (F) y posterior o trasero (T). Distancia sobre el eje Z del sistema de coordenadas del observador.
- Ventana en el plano de proyección (Wxmin, Wxmax, Wymin, Wymax). En coordenadas mundiales 2D.

Solo estos dos son los que hay que indicar en OpenGL.

• **Explicación de las diapositivas (26-31) Pregunta de examen posible**

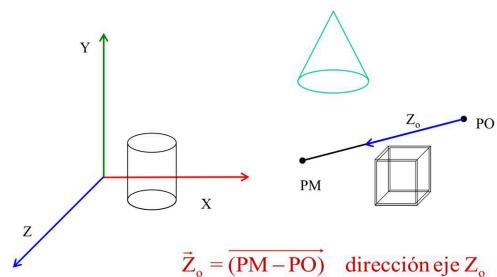
Pasos para el proceso de síntesis de imágenes en 3D.

1º Establecemos el sistema de coordenadas mundiales:



Sistema de coordenadas mundiales

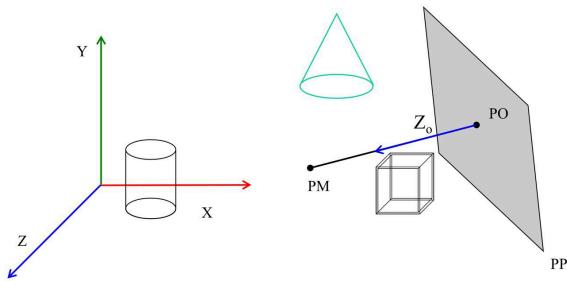
2º Establecemos el punto en el plano de proyección (PO) y el punto de mira (PM).



En la imagen, el eje Z_o representa la dirección de la vista en un sistema de coordenadas de cámara o de proyección.

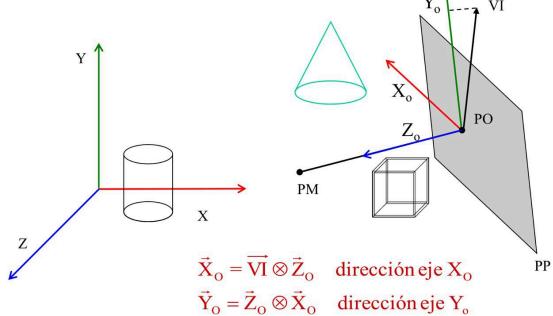
Punto en el plano de proyección (PO) y punto de Mira (PM)

3º Establecemos el plano de proyección.



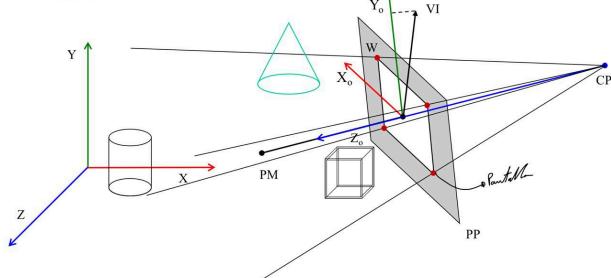
4º Establecemos los vectores de inclinación de la cámara (VI)

– Construcción



Define la orientación de la cámara en el espacio tridimensional. Estos vectores de inclinación son fundamentales para establecer la transformación que lleva los puntos del modelo 3D al espacio de la cámara, y son esenciales para realizar la proyección final en el plano de imagen (PP).

– Construcción



5º Establecemos el centro de proyección (CP) y la ventana en el plano de proyección.

La ventana en el plano de proyección sirve como un marco de referencia para determinar qué parte del espacio tridimensional será capturada y visualizada en la imagen final 2D. Actúa como el "campo de visión" de la cámara en el contexto de la proyección

gráfica.

6º Establecemos los planos de recorte frontal (F) y posterior o trasero (T)

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

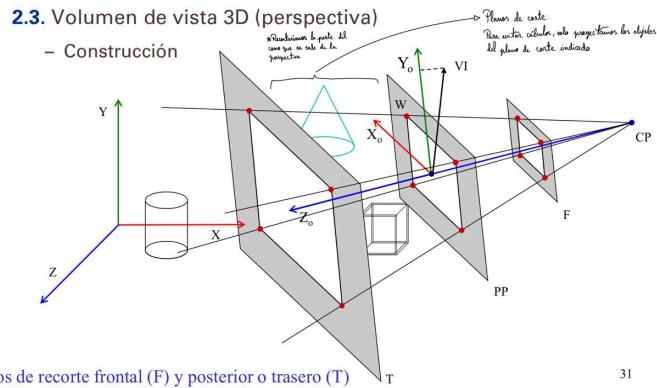
Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

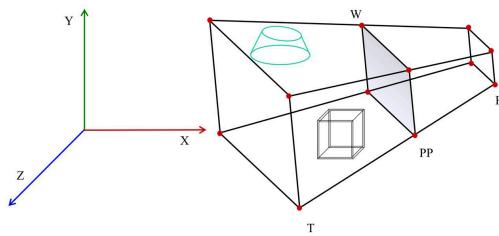
Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositista. Consulta más información en inglés.



31

7º Por último realizamos el recortado conveniente



Volúmen de vista 3D (paralela)

Parámetros:

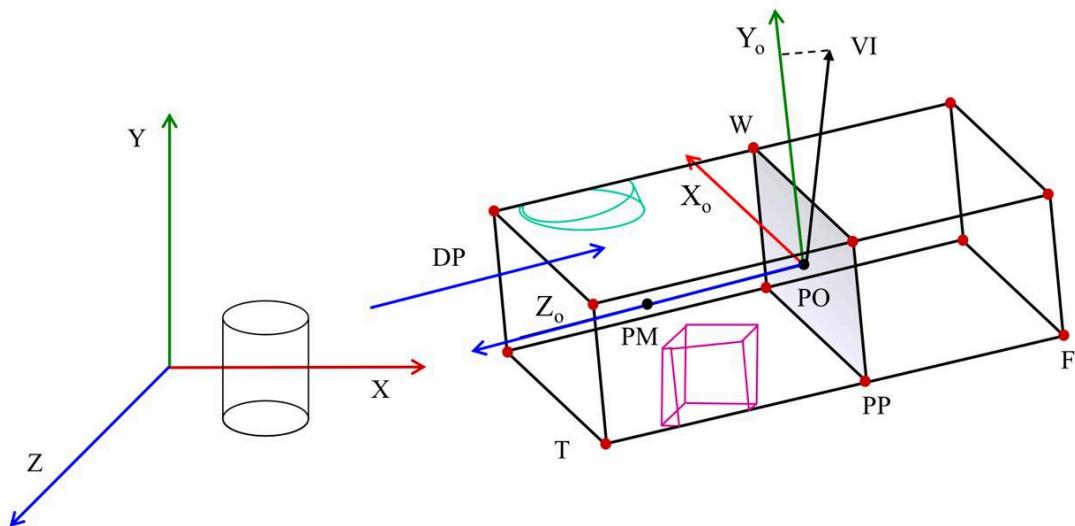
- Punto en el plano de proyección (PO). En coordenadas mundiales 3D
- Punto de mira (PM) o posición a la que apunta la cámara. En coordenadas mundiales 3D.
- Normal al plano de proyección (PM-PO).
- Vector de inclinación de la proyección (VI). En coordenadas mundiales 3D
- Vector de dirección de proyección (DP). En coordenadas mundiales 3D
- Planos de recorte frontal (F) y posterior o trasero (T). Distancia sobre el eje Z del sistema de coordenadas del observador.
- Ventana en el plano de proyección (Wxmin, Wxmax, Wymin, Wymax). En coordenadas mundiales 2D.

Consulta condiciones aquí



do your thing

WUOLAH



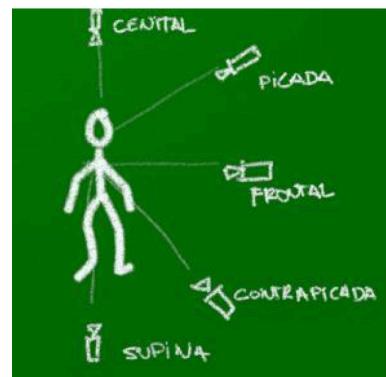
Volúmen de vista 3D (perspectiva) definido por el ángulo del campo de visión en el plano X-Y y la proporción entre el ancho y el alto (aspect=w/h)

Tipos de ángulos:

- Normal: Aproximadamente 46°. Se corresponde con el ángulo de un ojo humano.
- Angular: Cuando es mayor de 46°. Al abarcar más campo de visión, los objetos parecen más pequeños. Para sacarlos al mismo tamaño, nos tenemos que acercar a ellos obteniendo grandes distorsiones.
- Tele: Cuando es menor de 46°. Los objetos parecen más grandes. Para sacarlos al mismo tamaño, nos tenemos que alejar, comprimiendo los planos.

Tipos de planos: Según orientación de la cámara

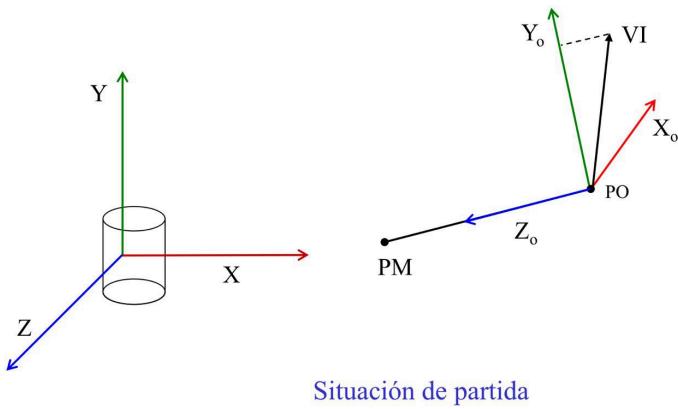
- Frontal: Proporciona una visión neutra y objetiva de la escena
- Picado: Hace que los personajes parezcan pequeños y débiles (cámara a 45°)
- Contrapicado: Hace que los personajes parezcan grandes y heróicos (cámara a -45°)
- Subjetivo (POV): Sitúa la cámara en el punto de vista de un personaje
- Cenital: Cámara a 90°
- Supina: Cámara a -90°



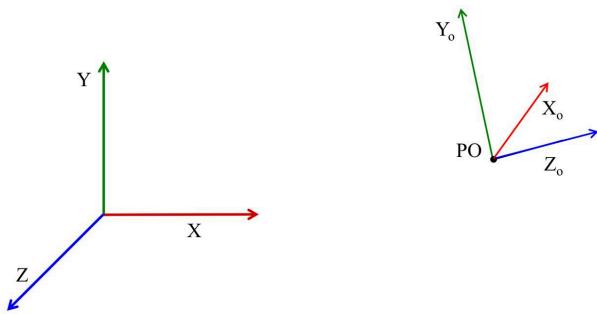
● Transformación de vista 3D

Transformación de vista 3D (opción 1) Opción compleja

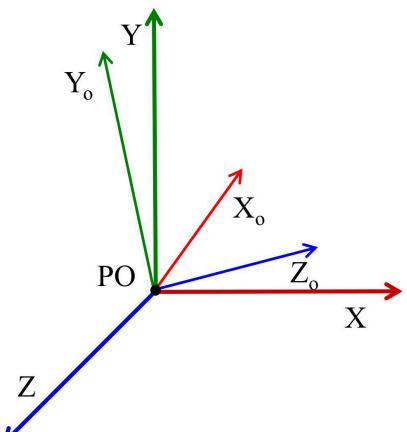
0° Situación de partida



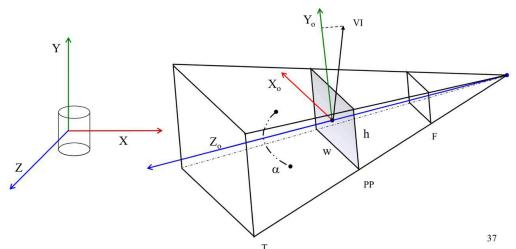
1º Paso 1. Inversión del eje Zo



2º Paso 2. Traslación al origen del sistema de coordenadas del observador al origen de coordenadas a nivel mundial.



3º Paso 3. Rotación del ejerce Zo con respecto a X



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

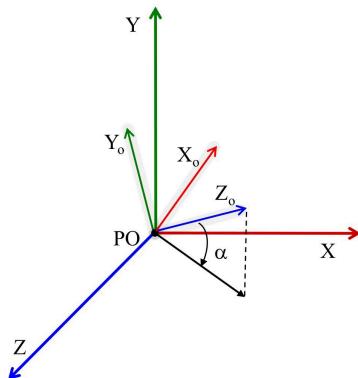
Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

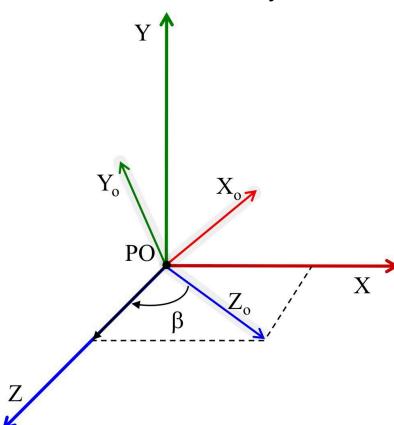
1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

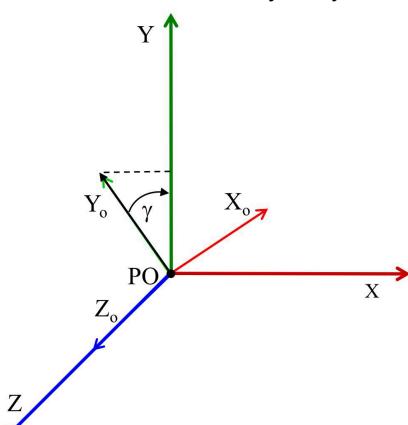
ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.



4º Paso 4. Rotación del eje Z_o con respecto a Y



5º Paso 5. Rotación del eje Y_o y X_o con respecto a Z



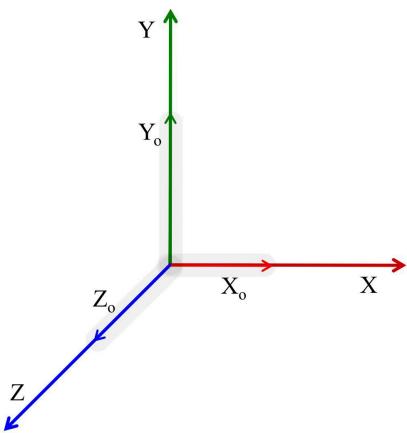
6º Paso 6. Los dos sistemas de coordenadas ya alineados

Consulta condiciones aquí



do your thing

WUOLAH



$$T_{V3D} = Tr(-PO_x, -PO_y, -PO_z)R(\alpha, X)R(-\beta, Y)R(\gamma, Z) =$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -PO_x & -PO_y & -PO_z & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \operatorname{sen} \alpha & 0 \\ 0 & -\operatorname{sen} \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(-\beta) & 0 & -\operatorname{sen}(-\beta) & 0 \\ 0 & 1 & 0 & 0 \\ \operatorname{sen}(-\beta) & 0 & \cos(-\beta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \gamma & \operatorname{sen} \gamma & 0 & 0 \\ -\operatorname{sen} \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

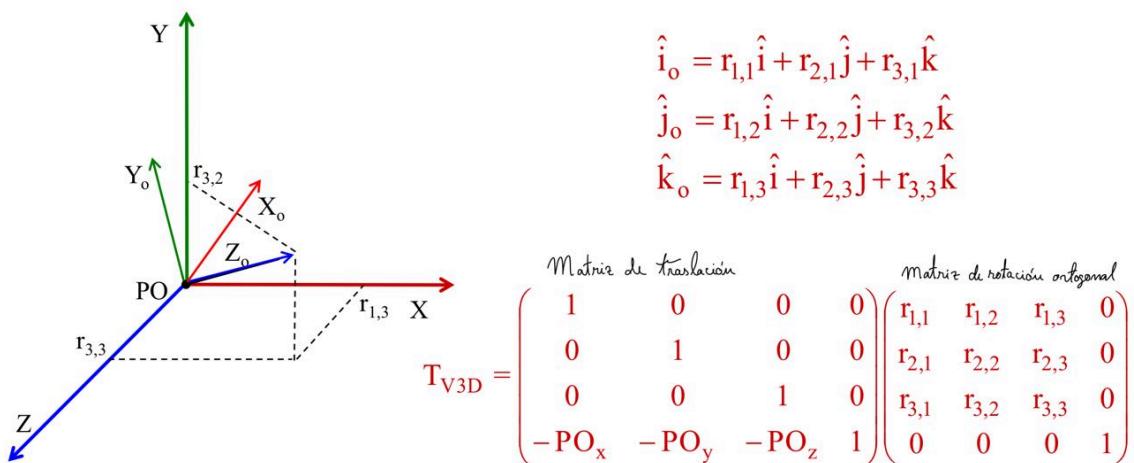
La transformación de vista 3D, TV3D, se compone de una traslación seguida de tres rotaciones alrededor de los ejes del sistema de coordenadas:

1. Translación $Tr(-Pox, -Poy, -Poz)$. Esta matriz mueve el sistema de coordenadas de la escena para que el punto de observación (PO) esté en el origen. Esto implica una traslación negativa por el vector de posición del punto de observación.
2. Rotaciones $R(\alpha, X) R(-\beta, Y) R(\gamma, Z)$: Estas son tres matrices de rotación que rotan la escena alrededor de los tres ejes principales (X, Y, Z). La notación $R(\theta, Eje)$ indica una rotación por un ángulo θ alrededor del eje especificado. Las rotaciones se aplican en el siguiente orden:
 - Alrededor del eje X por un ángulo α
 - Alrededor del eje Y por un ángulo $-\beta$ (el signo negativo indica la dirección de la rotación).
 - Alrededor del eje Z por un ángulo γ .

Las matrices de rotación utilizan funciones trigonométricas de los ángulos de rotación: seno (sen) y coseno (\cos) para definir cómo se rotará la escena en cada eje. La combinación de estas transformaciones permite que la cámara (o el punto de vista) se mueva y rote alrededor del objeto o escena, cambiando así la perspectiva desde la cual se visualizará el modelo 3D.

Transformación de vista 3D (opción 2) Opción simple

- Se aplica la propiedad de ortogonalidad de Euler



Esto implica la creación de una matriz de transformación que cambia la perspectiva de un modelo tridimensional para que pueda ser visto desde un punto de vista particular en un espacio bidimensional.

La transformación de vista se compone de dos partes principales:

Matriz de rotación ortogonal: Basada en los vectores unitarios i, j, k del sistema de coordenadas de la cámara (o vista). Estos vectores se expresan en términos de los vectores unitarios del sistema de coordenadas mundial i, j, k , con los coeficientes $r_{1,1}$ a $r_{3,3}$ definiendo cómo se alinean las coordenadas de la vista con respecto a las del mundo. La ortogonalidad significa que los ejes de la cámara son perpendiculares entre sí y cada uno es una combinación lineal de los ejes del mundo.

Matriz de traslación: Desplaza el sistema de coordenadas para que el punto de observación (PO) se mueva al origen. Esto se logra mediante la multiplicación por una matriz de traslación que contiene los valores negativos de la posición del punto de observación ($-PO_x, -PO_y, -PO_z$) para mover el origen del sistema de coordenadas del mundo al punto de vista de la cámara.

La matriz $TV3D$ resultante es una combinación de estas dos transformaciones, que se utiliza para convertir las coordenadas del mundo 3D a coordenadas de cámara 3D. Al aplicar esta matriz a un modelo 3D, se puede manipular el modelo para que parezca que se ve desde la posición y orientación específicas de la cámara, preparándolo para la proyección final en una imagen 2D.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene una cobertura garantizada de hasta 100.000 euros por depositista. Consulta más información en inglés.

$$PO = (0,0,1)$$

$$PM = (0,0,0)$$

$$VI = (0,1,0)$$

$$Z_o = (0,0,-1) \rightarrow \hat{k}_o = \hat{k}$$

$$X_o = \frac{\vec{VI}}{|\vec{VI}|} \otimes \hat{k}_o \rightarrow \hat{i}_o = \hat{i}$$

$$\hat{j}_o = \hat{k}_o \otimes \hat{i}_o = \hat{j}$$

$$T_{V3D} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La matriz cambia la perspectiva de una escena 3D para que pueda ser vista desde una cámara virtual situada en un punto específico y orientada en una dirección particular.

Los datos proporcionados son:

PO (Punto de Observación) está en las coordenadas (0,0,1).

PM (Punto de Mira) está en el origen (0,0,0).

VI (Vector hacia arriba de la Vista) está en las coordenadas (0,1,0).

Los vectores unitarios del sistema de coordenadas de la cámara son definidos como:

Zo es (0,0,-1), lo que indica la dirección en la que la cámara está mirando.

Xo se calcula como el producto vectorial de VI y Zo, resultando en el vector unitario io que es igual a i.

Jo se calcula como el producto vectorial de Zo y Xo, resultando en el vector unitario jo que es igual a j.

La matriz de transformación de vista TV3D resultante es:

```

1 0 0 0
0 1 0 0
0 0 1 0
-POx -POy -POz 1

```

Esta matriz es una combinación de una matriz de identidad, que no altera la orientación ya que los ejes de la cámara y del mundo coinciden, y una matriz de traslación que mueve el sistema de coordenadas mundial al punto de observación.

El resultado de “Transformación de vista 3D (opción 1/2)” nos da una vista isométrica.

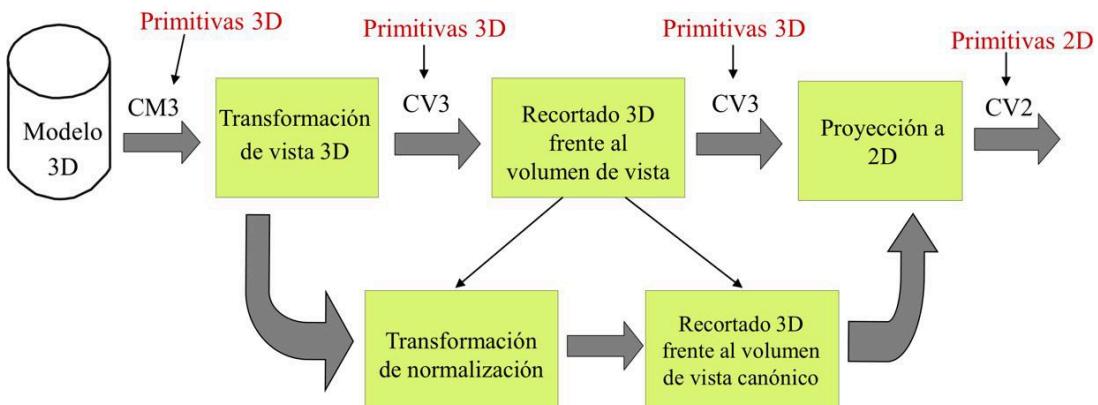
Consulta condiciones aquí



WUOLAH

• Transformación de normalización:

La operación más costosa en la síntesis 3D es el recortado, así que se consideran volúmenes de vista normalizados o canónicos en los que recortar es más eficiente.



La diapositiva muestra las etapas de procesamiento de un modelo 3D para su visualización en una pantalla 2D, enfocándose en la etapa de normalización y recortado.

El proceso comienza con un modelo 3D compuesto por primitivas geométricas en un sistema de coordenadas mundial (CM3). Primero, este modelo se transforma a un sistema de coordenadas de vista (CV3) mediante una transformación de vista 3D. Esta transformación sitúa el modelo en relación con la posición y orientación de una cámara virtual.

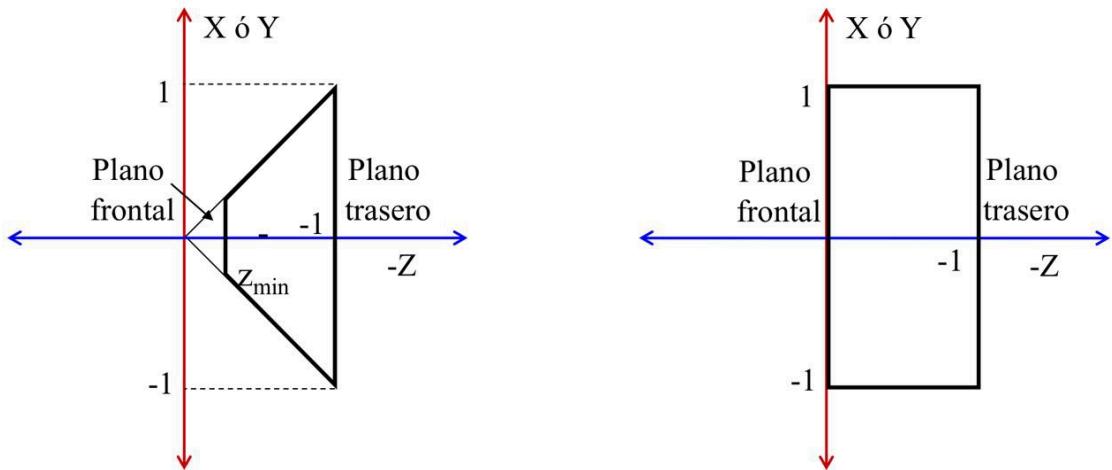
Una vez que el modelo está en el espacio de la cámara, se realiza un recortado 3D frente al volumen de vista. Esto significa que se eliminan las partes del modelo que están fuera del campo de visión de la cámara, es decir, no contribuirán a la imagen final y, por tanto, no necesitan procesarse más.

La transformación de normalización es un paso adicional de recortado y escalamiento que asegura que el volumen de vista transformado encaje dentro de un volumen de vista canónico estandarizado. Este volumen canónico es un espacio de referencia predefinido que simplifica la última etapa de proyección a 2D, ya que ahora todos los objetos están dentro de un rango conocido y manejable.

Finalmente, el modelo recortado y normalizado se proyecta en un espacio bidimensional (CV2) para mostrarlo en la pantalla. Este proceso de proyección convierte las coordenadas tridimensionales en coordenadas bidimensionales, lo cual es esencial para poder visualizar la escena 3D en una superficie plana como una pantalla de ordenador.

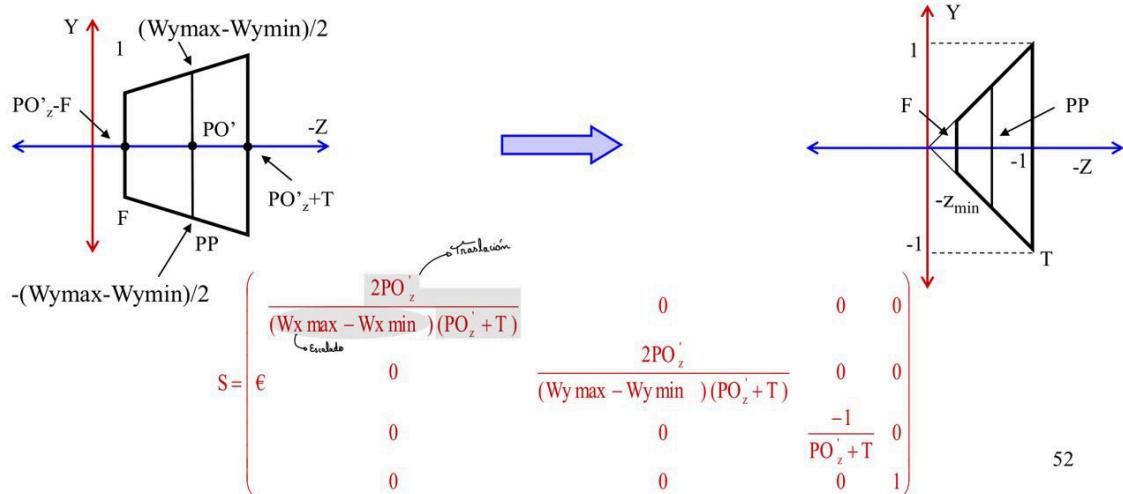
Los volúmenes de vista normalizados o canónicos son:

- Perspectiva
- Proyección paralela



Transformación de normalización

Transformación de volumen de vista a volumen de vista normalizado para una proyección en perspectiva. Caso simple: CP, PO y centro de la ventana del mundo en el eje Z del observador.



Esta transformación es una parte esencial del proceso de renderizado 3D, que convierte el volumen de vista tridimensional específico del mundo de la cámara (donde se ha realizado el recorte de objetos que no están dentro del campo de visión) en un volumen de vista normalizado.

El volumen de vista normalizado es un espacio estandarizado que típicamente tiene una forma de cubo o caja con rangos definidos, por ejemplo, entre -1 y 1 en todas las direcciones. Esto simplifica la proyección final al plano de imagen, ya que todos los puntos ahora se encuentran en un rango conocido y manejable. La perspectiva también se tiene en cuenta durante esta transformación para asegurar que los efectos de la profundidad se mantengan en la visualización final.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.

La transformación se realiza en varios pasos:

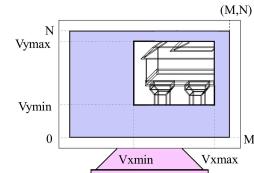
1. Escalado: El espacio de vista se escala para que coincida con el rango del volumen de vista normalizado. Esto se realiza mediante la matriz de escalado (S) mostrada en la diapositiva, donde (ϵ) parece ser un factor de escala para la coordenada X, y los términos para la coordenada Y y la coordenada Z se basan en la altura y la profundidad del volumen de vista original (definido por los valores W_{max} , W_{min} , W_{xmax} , W_{xmin} y los límites de Z).

2. Traslación: Además del escalado, la transformación también implica trasladar el volumen de vista para que su centro coincida con el origen del volumen de vista normalizado. Esto se realiza ajustando las coordenadas en la dirección Z, como se indica en la matriz con los términos ($P_0 \cdot z + T$).

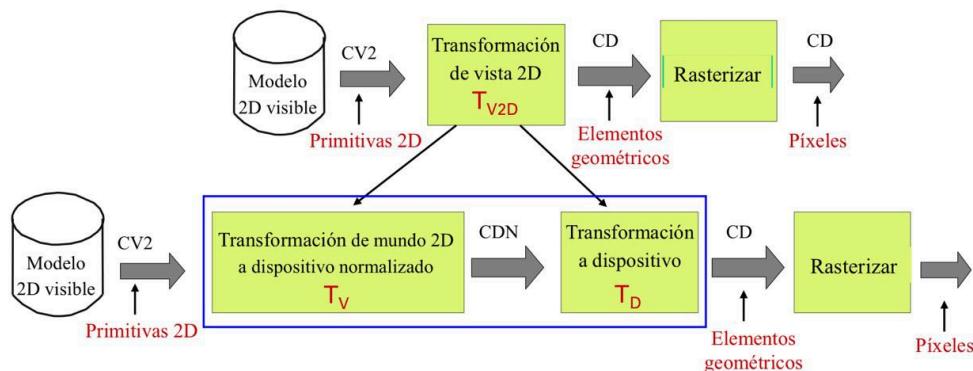
Al aplicar esta transformación, el volumen de vista desde la cámara se normaliza y se prepara para la proyección en el plano de imagen, que será la última etapa antes de convertir la escena en una imagen 2D que se puede mostrar en una pantalla.

Transformación de vista 2D

- Se transforma el modelo proyectado (modelo 2D) a coordenadas del dispositivo de salida.
- La coordenada Z de los vértices se conserva para hacer después eliminación de partes ocultas.
- Se introduce el concepto de viewport. Una zona rectangular del dispositivo de salida donde se dibujará la imagen.



Se utilizan coordenadas normalizadas de dispositivo



CDN=Coordenadas de dispositivo
normalizado

54

Consulta condiciones aquí



WUOLAH

Transformación de vista 2D en el proceso de renderizado de gráficos por computadora, que es el paso donde se convierten las coordenadas 2D del mundo de la escena a coordenadas 2D del dispositivo de visualización, como una pantalla.

La secuencia es la siguiente:

- **Modelo 2D visible:** La escena ya ha sido proyectada desde 3D a 2D y contiene solo los elementos visibles dentro del volumen de vista.
- **Transformación de vista 2D (T_V2D):** Se aplica una transformación para convertir las coordenadas del mundo 2D (CV2) a coordenadas normalizadas del dispositivo (CDN). Esto implica ajustar el tamaño y la posición de la imagen para que se ajuste a un rango estandarizado, generalmente el rango que el dispositivo puede mostrar.
- **Transformación de mundo 2D a dispositivo normalizado (T_V):** Esta es otra representación de la transformación de vista 2D, y se muestra cómo las coordenadas 2D se llevan a un sistema de coordenadas de dispositivo normalizado.
- **Transformación a dispositivo (T_D):** Convierte las coordenadas normalizadas a coordenadas específicas del dispositivo de visualización (CD), teniendo en cuenta las dimensiones y la resolución del dispositivo para que la imagen se muestre correctamente.
- **Rasterizar:** Este paso convierte la geometría vectorial en píxeles para su visualización final. Cada elemento geométrico se convierte en un conjunto de píxeles que forman la imagen final que se verá en la pantalla.

Rasterización → Rasterización (discretización): Los elementos geométricos son continuos, pero si los queremos visualizar en un dispositivo raster se tienen que representar por puntos discretos que se corresponderán a pixeles.

- **Eliminación de partes ocultas** → Se visualizan los objetos presentando únicamente porciones visibles para el observador.

● **Algoritmo Z-buffer:**

- Devuelve para cada píxel el color del objeto más cercano al observador. Puede emplearse en cualquier tipo de objeto, no sólo los definidos para caras planas.
- El algoritmo Z-Buffer puede ser fácilmente implementado en hardware. Necesita que se tenga disponible una memoria donde ir almacenando valores de profundidad para cada píxel, de ahí su nombre.
- Es el algoritmo de eliminación de partes ocultas más ampliamente utilizado. Por ejemplo es utilizado por OpenGL.

```
void Z-Buffer(...)
```

```

{
    int pz, y, x;
    for(y=0; y<YMAX; y++)
        for(x=0; x< XMAX; x++)
    {
        put-pixel (x, y, color_fondo);
        put_z_buffer(x, y, ZMAX);
    }

    for (cada polígono Q de la escena)
        for (cada pixel (x, y) de la rasterización de Q)
            {pz = valor de z del polígono Q en el pixel (x,y);
             if (pz <= read_z_buffer(x,y))
            {
                put_pixel(x,y,color_pol_Q_en_pixel(x,y));
                put_z_b(x,y,pz);
            }
        }
}

```

Algoritmo Z-Buffer, que es una técnica común en gráficos por computadora para la eliminación de partes ocultas de una escena 3D cuando se renderiza en una imagen 2D. El algoritmo Z-Buffer mantiene un búfer que registra la profundidad (valor Z) de cada píxel en la pantalla para determinar qué objetos están en primer plano y cuáles están ocultos detrás de otros.

Aquí está la explicación del pseudocódigo paso a paso:

Se declara una función Z-Buffer con un conjunto no especificado de parámetros.

Dentro de la función, se declara la variable pz que almacenará el valor de profundidad de los píxeles, así como las variables y y x para iterar a través de los píxeles en la pantalla.

El primer bucle for inicializa cada pixel en la pantalla al color de fondo y establece su valor de profundidad en el Z-Buffer al máximo valor Z posible (ZMAX). Esto es equivalente a "limpiar" la pantalla y el Z-Buffer para la siguiente fase de dibujo.

El segundo conjunto de bucles for itera sobre cada polígono Q en la escena y luego sobre cada píxel (x,y) que resulta de la rasterización del polígono Q. La rasterización es el proceso de convertir un polígono en una serie de píxeles en la pantalla.

Dentro del bucle más interno, pz obtiene el valor Z del polígono Q en el pixel (x,y). Si este valor Z es menor que el valor almacenado actualmente en el Z-Buffer para ese píxel (lo que significa que el polígono está más cerca de la cámara que lo que se había dibujado antes), se actualiza el pixel en la pantalla con el color del polígono Q y se actualiza el Z-Buffer con el nuevo valor Z (pz).

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

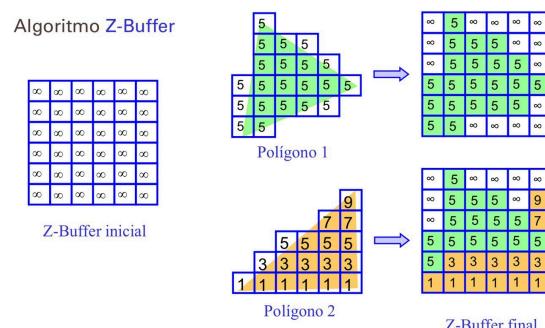
1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositista. Consulta más información en inglés.

Esta técnica asegura que solo los objetos visibles desde la perspectiva de la cámara sean dibujados y que los objetos que estén detrás de otros no se muestren, manejando así la visibilidad y el ocultamiento correctamente en la escena renderizada.

El buffer inicial se inicializa con todos los valores a “infinito”.



• Luz como onda partícula

Luz como onda partícula

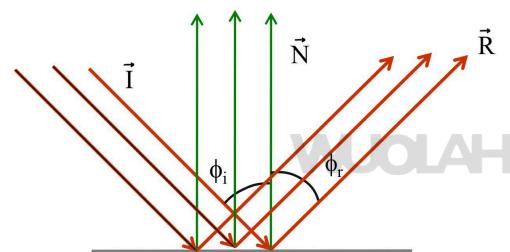
- La luz tiene una doble naturaleza ondulatoria-corpuscular.
- Como partícula, la luz consiste en pequeños paquetes de energía llamados fotones que se propagan en línea recta a gran velocidad.
- Como onda, la luz se propaga mediante un frente de onda en el que cada uno de sus puntos emite ondas esféricas elementales, siendo la posición del frente de onda al cabo de un tiempo la envolvente de dichas ondas esféricas (principio de Huygens)
- La óptica geométrica se ocupa de la propagación de la luz como energía, mientras que la óptica física u ondulatoria se ocupa de la propagación de la luz según las leyes del movimiento ondulatorio.

Interacción de la luz con la materia

- **Reflexión especular o regular** → La reflexión especular o regular es un tipo de reflexión de la luz que ocurre en superficies lisas y pulidas, como un **espejo o un lago tranquilo**, donde los rayos de luz que inciden en la superficie se reflejan en **una sola dirección**. Esta dirección se calcula según la ley de reflexión, que establece que el ángulo de incidencia es igual al ángulo de reflexión, y ambos ángulos se miden con respecto a la normal de la superficie en el punto de impacto.

En gráficos por computadora, la reflexión especular se simula para dar a los objetos un aspecto brillante y destacar la textura de la superficie. Se calcula utilizando modelos de

Reflexión especular o regular



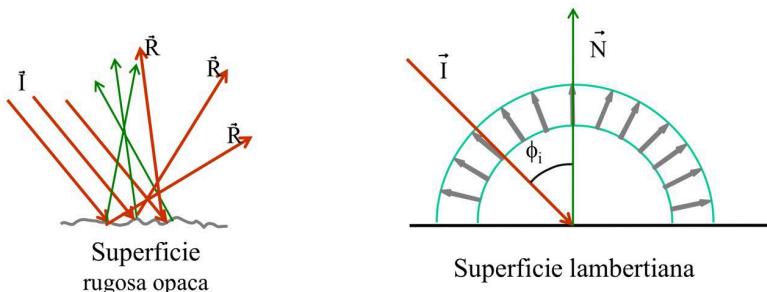
Consulta condiciones aquí



iluminación que toman en cuenta la posición de la luz, la posición del observador y las propiedades del material de la superficie para determinar el brillo o resplandor especular que contribuye al color final de un píxel. Esto produce los puntos brillantes y los reflejos nítidos que se ven en los objetos **metálicos** o húmedos en las imágenes renderizadas.

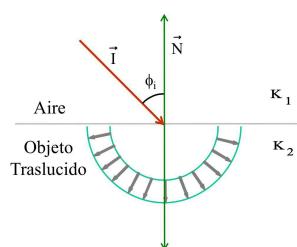
- **Reflexión difusa** → La reflexión difusa es el fenómeno que ocurre cuando la luz incide sobre una superficie **áspera o mate** y se refleja en **muchas direcciones diferentes**. A diferencia de la reflexión specular, que es direccional y mantiene la coherencia de los rayos de luz, la reflexión difusa dispersa la luz, lo que resulta en una iluminación suave y uniforme sobre la superficie del objeto.

En gráficos por computadora, la reflexión difusa es modelada para representar cómo la luz interactúa con superficies que no son reflectantes de manera especular. Es un componente clave del modelo de iluminación de Lambert, que asume que la luz se refleja igualmente en todas las direcciones y la cantidad de **luz reflejada depende únicamente del ángulo entre la luz incidente y la normal de la superficie**. Esto hace que los objetos con reflexión difusa no tengan brillos o reflejos definidos, sino más bien un color que parece venir uniformemente de la superficie, lo cual es característico de materiales como la arcilla, el papel y la piel.



- **Transmisión especular** → Aire a objeto transparente
- **Transmisión difusa** → Aire a objeto translúcido
- **Transmisión difusa ideal** → Proceso en el cual la luz pasa a través de un material y se dispersa uniformemente en todas direcciones, similar a cómo una luz suave y uniforme atraviesa un vidrio esmerilado o un tejido fino.

Transmisión difusa ideal



Luz como partícula:

- Explica la absorción y emisión de luz
- Explica la reflexión especular. Las partículas luminosas son elásticas y en la reflexión se produce un choque elástico.
- Explica de forma incompleta la transmisión

Luz como onda

- Explica la transmisión, incluyendo que en medios con mayor índice de refracción la velocidad de la luz es menor
- Explica las interferencias y la difracción

Luz como onda y partícula

- La luz es una energía electromagnética con longitudes de onda comprendidas entre 400-700 nm.
- Los tres términos físicos más importantes para describir el color son:
 1. La longitud de onda dominante
 2. La pureza de excitación o contenido en gris
 3. La luminosidad o intensidad de luz
- La cantidad de energía presente para cada longitud de onda se representa por medio de una gráfica de distribución de energía espectral $P(\lambda)$
- Existen colores que teniendo distintas distribuciones espectrales $P(\lambda)$ aparecen al ojo como iguales (metámeros)
- Así el color se puede especificar mediante el empleo de tres luces especialmente simples mezcladas en distintas proporciones.
- Las luces se toman de tres zonas concretas del espectro. En el sistema RGB se toman 700 nm para el rojo, 546,1 nm para el verde y 458,3 nm para el azul.
- Hay colores que no pueden obtenerse por mezclas de tres luces RGB.
- Modelo CIE se define con tres valores primarios llamados X, Y y Z, que no se corresponden con ningún valor real y son positivos.

El primario Y coincide con la eficiencia de luminosidad. Cada color real C se define por las cantidades (x, y, z) necesarias matemáticamente para producir ese color.

En el modelo CIE, el gamut son los colores comprendidos dentro de un triángulo formado con tres colores.

Consideraciones en los modelos de iluminación local

En general, un modelo de iluminación o shading model determina qué intensidad luminosa $I\lambda$ que ha de asignarse a cada punto de una superficie.

Los modelos de iluminación local son llamados modelos empíricos locales o de primer orden.

Son empíricos porque están basados en la óptica geométrica y locales porque se considera que la única luz que se puede iluminar una superficie procede de las fuentes de luz de la escena (luz directa).

La iluminación de una escena no genera sombras definidas detrás de los objetos, ya que no hay bloqueo u obstrucción ("occlusiones") de la luz por parte de esos objetos. Esto implica que la luz se difunde de tal manera que alcanza todas las superficies directamente, y como resultado, las fuentes de luz son visibles desde cualquier ángulo de visión que esté orientado hacia ellas, sin ser ocultadas por otros objetos en la escena.

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

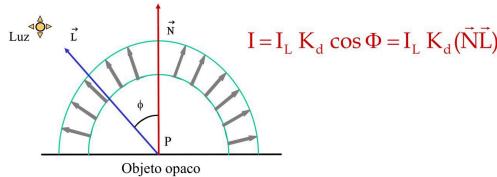
ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.

En lugar de considerar todas las longitudes de onda λ para los cálculos de iluminación, se usan las tres longitudes del modelo RGB.

El modelo más simple es el ambiental que aproxima toda la iluminación indirecta. Asignar un valor constante de intensidad a cada punto de una cara.

Reflexión difusa

- Reflexión isotrópica de la luz que llega a una superficie
- La intensidad en una superficie sigue la ley de L'Ambert. Depende del cociente de reflexión difusa K_d y del ángulo formado entre el vector de dirección al foco de luz y la normal de la superficie.



El $\cos(\alpha)$ ha de ser mayor o igual a cero

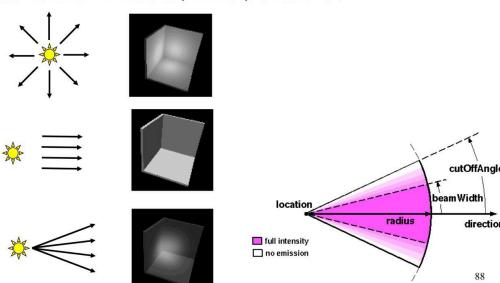
- La dependencia con la longitud de onda se puede escribir de dos formas:

$$I_{\lambda} = I_{a\lambda} K_{a\lambda} + I_{L\lambda} K_{d\lambda} (\vec{N}\vec{L})_+$$

$$I_{\lambda} = I_{a\lambda} K_a O_{d\lambda} + I_{L\lambda} K_d O_{d\lambda} (\vec{N}\vec{L})_+$$



3.3. Reflexión difusa (luz puntual y direccional)



Reflexión especular (El modelo de Phong)

- El modelo de Phong modela el brillo en las superficies
- Divide la reflectividad en una parte difusa y una especular. La difusa es independiente del observador, en cambio, no lo es la especular

“El modelo de Phong es una técnica de iluminación en gráficos por computadora que simula cómo la luz interactúa con las superficies, utilizando una combinación de reflexión difusa, reflexión especular y luz ambiental para dar a los objetos un aspecto tridimensional realista.”

Consulta condiciones aquí



do your thing

WUOLAH

● Texturas

Con las transformaciones de textura se añaden detalles a la superficie de los objetos sin aumentar su número de caras.

En una transformación de textura ($R3 \rightarrow R2$) se consideran los aspectos siguientes:

- Su aplicación a una superficie 3D (texturas planas)
- Parámetros de la superficie del objeto que se modulan para dar la impresión de textura:
 - a. Coeficientes de reflexión difusa y especular
 - b. Perturbación del vector normal
 - c. Transparencia
- Síntesis de texturas

Texturas planas

Una textura plana supone pegar una trama o imagen sobre una superficie. La imagen se suele expresar en dos parámetros u, v normalizados (entre 0 y 1). El valor del color RGB en un punto u, v determinará normalmente el valor de las constantes K_a, λ y K_d, λ de un modelo de iluminación local.

A los píxeles de una imagen usada para una textura se les llama texels (texture elements). Para aplicar una textura tiene que existir una función T tal que $(u, v) = T(x, y, z)$, con (x, y, z) un punto de una superficie.

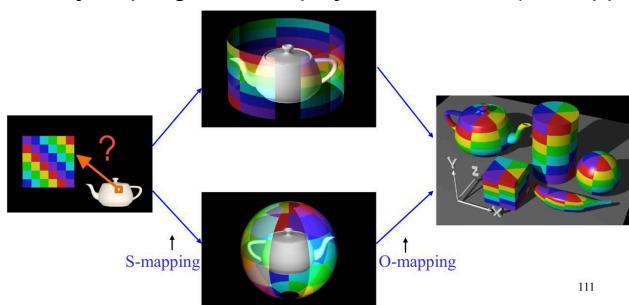
Asignación de coordenadas de textura

Esta asignación se puede realizar de tres modos:

- Manual; para objetos muy simples se asigna una coordenada de textura a cada vértice
- Procedural; para objetos sencillos y objetos complejos donde no importe que se pueda deformar la textura

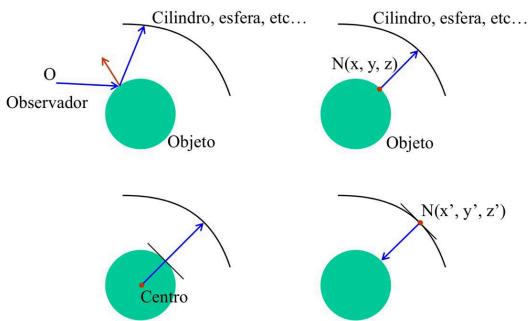
Para un objeto poligonal complejo se opera en dos etapas:

1. Transformación del espacio bidimensional de la textura a un objeto 3D sencillo como un plano, cubo, esfera o cilindro que envuelva al objeto poligonal (S-mapping).
2. Transformación de la textura obtenida en el objeto 3D sencillo a la superficie del objeto poligonal complejo en cuestión (O-mapping).



111

- Hay cuatro tipos de transformaciones **O-mapping**



- Asistida; para objetos muy complejos. Se usan herramientas CAD.

4.TEMA: Interacción

● Introducción

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

El sistema gráfico funciona según el siguiente ciclo:

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

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

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

En sistemas más profesionales, se suelen utilizar dispositivos de entrada especiales como una tableta digitalizadora, un sistema de posicionamiento o un palpador.

Latencia: Para un sistema gráfico interactivo, la latencia (retardo entre la acción del usuario y la respuesta del sistema) ha de ser lo suficientemente pequeña como para que el usuario perciba una reacción de causa efecto (*del orden de 100 milisegundos*).

Realimentación: Es el mecanismo mediante el cual el sistema da información al usuario útil para determinar la siguiente acción a realizar. Se puede usar con diferentes fines:

- Mostrar el estado del sistema
- Como parte de una función de entrada
- Para reducir la incertidumbre del usuario

Técnicas que simplifican el proceso de interacción:

- Líneas o curvas elásticas
- Rejillas
- Puntos de gravedad

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en inglés.

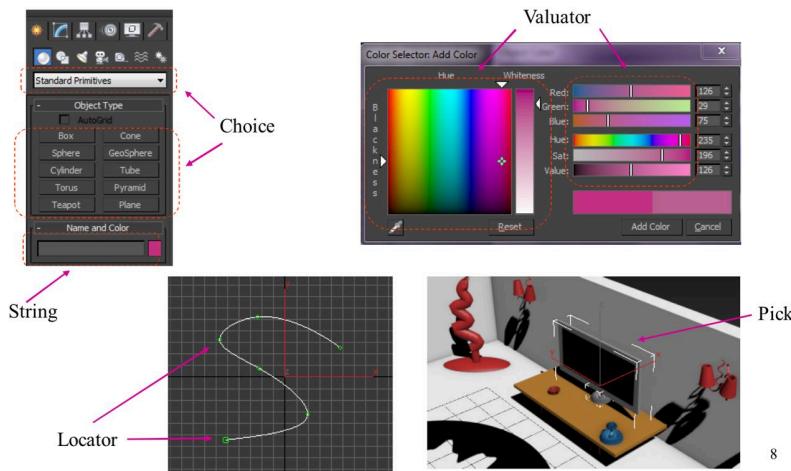


En un sistema gráfico se utilizan dispositivos de entrada abstractos, denominados dispositivos de entrada lógicos.

Cada dispositivo lógico representa un dispositivo físico ideal, que genera un único tipo de información de entrada.

Hay cinco tipos de dispositivos de entrada lógicos:

- Locator (posicionamiento): Un punto en coordenadas mundiales.
- Valuator: Un valor numérico dentro de algún rango
- Choice: Selección de un valor dentro de un conjunto de alternativas
- String: Una cadena de texto
- Pick (selección): Un identificador de un componente de un modelo gráfico



8

Un dispositivo de entrada físico se puede corresponder con uno o varios dispositivos de entrada lógicos.

Los dispositivos de entrada lógicos, en principio, se pueden hacer corresponder con cualquier dispositivo físico de entrada, pero algunos dispositivos hardware son más apropiados que otros para ciertos tipos de datos de entrada.

Para un sistema gráfico hay dos tipos de dispositivos lógicos más específicos: Locator (posicionamiento) y Pick (selección).

- El posicionamiento permite generar posiciones en coordenadas del mundo.
- La selección permite identificar interactivamente elementos del modelo geométrico.

La forma en cómo interactúan los dispositivos de entrada y una aplicación depende de cómo se sincronicen 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 abstracción de las operaciones de entradas.

Se definen tres modos de entrada: *petición (request)*, *evento (event)* y *muestreo (sample)*.

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

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

Consulta condiciones aquí



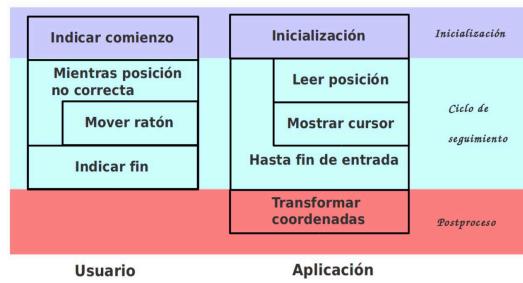
WUOLAH

- En el modo *evento* (*event*) se utiliza un buffer, de modo que el usuario puede realizar entradas antes de que la aplicación las solicite. Se almacena una cola de sucesos (valores y dispositivo que lo produjo).
 - El modo *muestreo* (*sample*) se asocia un valor al dispositivo por defecto al dispositivo que el usuario puede modificar y la aplicación puede consultar su valor.

• Posicionamiento

- El objetivo de las operaciones de posicionamiento es permitir la introducción de un punto.
 - En la inicialización se muestra el estado inicial del dispositivo. La rejilla es frecuentemente usada como técnica.
 - En el seguimiento, se usa con frecuencia la línea elástica como técnica.

- Correspondencia entre operaciones del usuario y del software de tratamiento



La posición obtenida en la lectura está dada en **coordenadas de dispositivo**, por lo tanto es necesario transformarla a **coordenadas del mundo**. Para ello se puede:

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

Posicionamiento (control de la cámara)

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

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

- Visualización de modelos: posición (**cámara orbital**)
La cámara mira al objetivo, moviéndose en torno a él sobre una esfera.
La posición de la cámara se puede especificar en coordenadas esféricas.
El vector de inclinación se debe calcular de forma que no sea perpendicular al plano de proyección.
 - Exploración de escenarios: dirección (**cámara en primera persona**)
La cámara está en un avatar o un personaje que se mueve en el escenario (punto de observación).

El usuario puede controlar la dirección en la que mira la cámara (punto de mira), que se puede especificar como dos giros en direcciones ortogonales a la dirección de avance del personaje.

El vector de inclinación se debe calcular de forma que no sea perpendicular al plano de proyección.

Una cámara en tercera persona puede seguir al personaje.

- **Cámara en ángulos de Euler**

La orientación de la cámara (punto de mira de la cámara) se puede expresar como tres giros (uno respecto a cada eje).

Esta representación de la orientación se conoce como ángulos de Euler que permite controlar la orientación con tres grados de libertad (incluyendo el control sobre el vector de inclinación).

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.

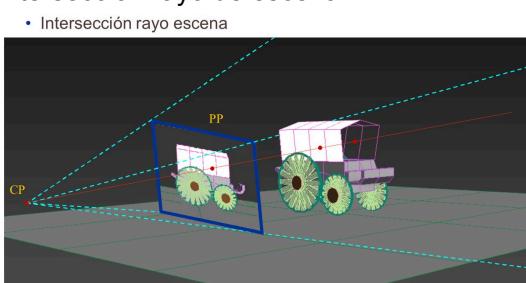
El proceso de selección suele realizarse como un posicionamiento y una búsqueda dentro del modelo jerárquico.

Para poder realizar la selección, los componentes de la escena deben tener asociados identificadores o etiquetas. Éstas se pueden asociar a distinto nivel, permitiendo controlar el ámbito de la selección: vértice, primitiva, objeto.

Métodos de selección:

La asociación entre posiciones de pantalla y elementos del modelo se puede establecer usando distintos métodos de selección:

1. Intersección rayo de escena



2. Identificando los objetos incluidos en un subvolumen de visión centrado en la posición dada (OpenGL)

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

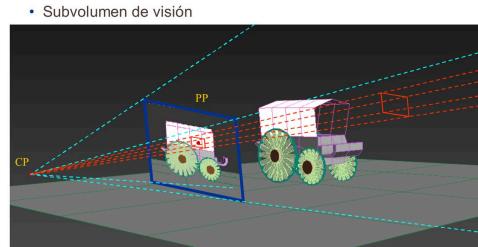
Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

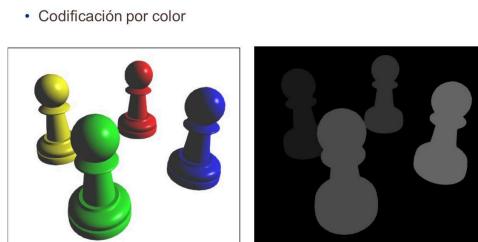
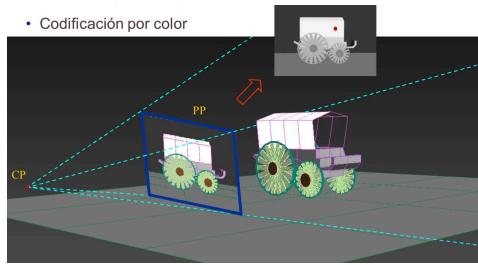
1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos y tiene como una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](#)



3. Codificando el identificador de objeto como color leyendo del frame buffer



● Animación

Animación: Creación de la ilusión de que las cosas cambian.

No sólo debemos atenernos al cambio de los objetos de sitio, sino también a los cambios de forma, los cambios de color o intensidad de la luz.

La ilusión de la animación se debe al fenómeno de la persistencia de la visión. Cuando presentamos rápidamente una serie de imágenes sucesivas que varían ligeramente, el sistema visual las mezcla.

Percepción de movimiento ⇒ 24 Imágenes/segundo

Existen diversos criterios de clasificación de la animación:

- Según el agente que produce la animación:
Hombre → Animación convencional
Ordenador → Animación por ordenador
- Según el medio o herramienta que se use:
Mano → Animación convencional
Ordenador → Animación asistida por ordenador

Consulta condiciones aquí



WUOLAH

- Según la apariencia visual
 - Aspecto bidimensional
 - Aspecto tridimensional

La animación por ordenador está orientada sobre todo a animación tridimensional, manejando entornos más complejos que la tradicional.

- Ventaja → Automatismo y manejo de grandes cantidades de información.
- Desventaja → Una buena animación requiere igual tiempo que hecha a mano.

En la animación por ordenador, éste tiene un papel verdaderamente importante. No es una herramienta de apoyo, sino un elemento básico.

La flexibilidad de la animación por ordenador reside en la posibilidad de crear mundos virtuales, con objetos y reglas virtuales que pueden ser cambiadas modificando la definición de los mismos.

Centrándonos en la modificación de los objetos de una escena, podemos usar las siguientes técnicas, que pueden combinarse:

- Fotogramas clave (keyframe)
- Esqueleto
- Simulación física
- Procedural

Keyframe ⇒ Se basa en la definición de una serie de posiciones o estados clave en determinados instantes de tiempo para un objeto dado. Después, se generarán los pasos intermedios mediante el concepto de intercalado: dadas dos situaciones clave inicial y final, obtener las intermedias mediante interpolación entre ellas.

Las posiciones clave han de ser aquellas de las que fácilmente se pueden obtener otras posiciones intermedias. No es fácil que el ordenador haga un buen intercalado si los fotogramas clave son muy diferentes.

Se suelen utilizar curvas de movimiento que sirven para realizar diagnósticos de la animación, así como para modificarla editando las curvas. Normalmente se incluyen tres curvas de movimiento para cada uno de los ejes: X, Y o Z (traslación, rotación, escalado).

Esqueleto o armature ⇒ Un esqueleto es un modelo simplificado del personaje, formado por segmentos rígidos (huesos o bones) unidos por articulaciones. En lugar de utilizar la figura completa, se utiliza el esqueleto, donde es más fácil hacer corresponder posiciones entre dos keyframes.

Simulación física ⇒ Para escenas con modelos físicos simples se puede calcular la configuración del modelo en cada fotograma usando las leyes de la mecánica clásica. Librerías para simulación física: ODE, Newton, Bullet...

Cinemática inversa → Se mueven las partes más bajas de la jerarquía de un objeto y el movimiento se transmite a los ancestros de las mismas (igual que ocurre en la realidad). Por ejemplo, si queremos acercar la mano de un personaje a un objeto, se colocan los dedos sobre el objeto, y el resto del brazo los seguirá automáticamente.

La cinemática inversa es desde el punto de vista matemático un problema mal definido (hay diversas soluciones).

Debemos indicar el modo exacto en que las articulaciones se coordinan, y las limitaciones de sus movimientos (por ejemplo, valores de rotación) para limitar el número de soluciones.

Procedural ⇒ El comportamiento del objeto se describe mediante procedimientos (funciones).

Es útil cuando el comportamiento es fácil de generar, pero difícil de simular físicamente.