
Computación Gráfica

Introducción y Conceptos Fundamentales

Prof. M.C. Rivara
2010

Contenido

1. Introducción general con conceptos claves
2. Historia de Computación Gráfica
3. El proceso de rendering CG en 3D
4. OpenGL: librería gráfica. API
5. Dispositivos gráficos y gráfica interactiva

Computación Gráfica

- Primer computador digital: década de los 40
- Tecnología pantallas de rayos catódicos 1950 MIT, permite desarrollo de computadores comerciales
- Desarrollo de hardware y software financiado por industria automotriz y espacial. Primer sistema CAD y FEM 1965, Lockheed

Despegue masivo en década del 80:

- desarrollo computadores personales raster (Xerox, IBM)

A partir de década del 80

- Explosión de aplicaciones gráficas, fáciles de crear, baratas (década 80)
- Modelamiento de sólidos en Unix:
Metáforas o conceptos claves
 - ambiente desktop (escritorio) para organizar pantalla
 - conjunto de ventanas rectangulares, cada window es terminal virtual
 - despliegue de íconos (representación de archivos, programas, impresoras, etc.)
 - manipulación directa de los objetos (apuntar y clickear)
- Aplicaciones científico-tecnológicas: ciencias, ingeniería, medicina
- Otras aplicaciones de la vida diaria: TV comercial, cine, educación

Origen de Computación Gráfica interactiva moderna

(1) Tesis doctoral de Iván Sutherland 1963: sistema de dibujo Sketchpad

- estructura de datos
- almacenamiento de jerarquías de símbolos (replicación fácil de componentes estándar)
- técnicas de interacción: teclado y lápiz de luz capaz de apuntar, dibujar, hacer elecciones
- algoritmos básicos.

(2) Necesidades de la industria automotriz y aeroespacial
Actividades CAD (computer aided design) y CAM (computer aided manufacturing)

A mediados de los 60: Muchos proyectos de investigación

Computación Gráfica

- Medio más natural de comunicación con computador
Aprovecha habilidades humanas altamente desarrolladas de reconocimiento de patrones 2D y 3D que permiten percibir y procesar rápidamente datos gráficos
- Visualización científica: muy importante desde 1980
- Permite sintetizar objetos concretos y también abstractos (moléculas químicas, objetos matemáticos)

- Uso de imágenes dinámicas para visualizar: evolución de fenómenos (se necesitan más de 15 cuadros / seg). Usuario puede controlar la animación: velocidad, zoom, cantidad de detalle, relación entre objetos. Ejemplo: simuladores de vuelos

Conceptos y elementos básicos en computación gráfica

- Tubo de rayos catódicos.
- Tecnología raster, frame buffer, tasa de refresco
- Modelo de color RGB
- Rendering (proceso completo de visualización de escena 3D)
- Modelación de la superficie 3D de los objetos (en base a mallas de polígonos)
- Transformaciones geométricas básicas y transformaciones de proyección (también geométricas)
- Luces de la escena
- Modelo de iluminación
- Algoritmos (algoritmos raster, eliminación de superficies ocultas, algoritmos de clipping, etc)

CG y procesamiento de imágenes

➤ **Computación Gráfica 3D**

Síntesis “gráfica” de objetos reales o imaginarios a partir de modelos matemático / computacionales. Se usan algoritmos / procedimientos que permiten obtener la imagen raster final en 2D

➤ **Procesamiento de imágenes**

Proceso inverso: Análisis de imágenes 2D, o reconstrucción de modelos 2D y 3D a partir de sus imágenes

Por supuesto estos “mundos” se intersectan

Historia desde el punto de vista de CG (1)

Principios de los 60s: Animación computacional para simulación física;

Edward Zajac visualiza investigación satelital usando CG en 1961.

1963: Sutherland (MIT)

Sketchpad (dispositivo de manipulación directa para CAD)

Dispositivos vectoriales (caligráficos) de despliegue

Técnicas interactivas

Douglas Eglebart inventa el mouse

1968: Evans & Sutherland fundan compañía de CG

1969: Primera SIGGRAPH

Finales 60's – Finales 70s: Dinastía Utah.

Historia desde el punto de vista de CG (2)

- 1970:** Pierre Bezier desarrolla las curvas de Bezier
- 1971:** Pintado – Sombreado de Gouraud (Gouraud Shading)
- 1974:** Ed Catmull desarrolla el z-buffer (Utah)
Hunger: primer corto animado por computador
Animación por keyframes y morphing
- 1975:** Bui-Tuong Phong crea el Phong Shading (Utah)
Martin Newell modela tetera de té con parches de Bezier (Utah)
- Mediados 70's:** Gráfica Raster (Xerox)
- 1976:** Jim Blinn introduce textura y bump mapping
- 1977:** Star Wars: se usa CG en algunas secuencias
- 1979:** Turner Whitted desarrolla ray tracing

Historia desde el punto de vista de CG (3)

Mediados 70's – Mediados 80's Búsqueda del realismo

radiosity

aplicaciones en tiempo real

1982: Películas Tron, Star Trek II: Vengeance of Khan

Sistemas de partículas y CG

1984: The last startfighter

CG reemplaza modelos físicos. Primeros intentos de realismo con CG

1986: Luxo Jr. (Pixar) Primera animación CG nominada al Oscar

1989: Tin toy (Pixar) gana el Oscar de la Academia

Toy Story (Pixar y Disney), primer film full CG

Reboot Primer cartoon totalmente 3D CG

Babylon 5 primer show TV usa modelos CG como rutina

Historia desde el punto de vista de CG (4)

Fines 90's Medioambiente interactivos
 Visualización científica y médica
 Rendering artístico, etc.

00's Rendering fotorealista
 Rendering interactivo de películas?
 Hardware!

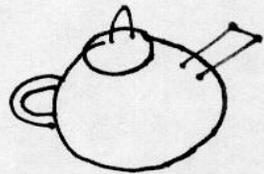
Lectura

Hearn – Baker, capítulo 1

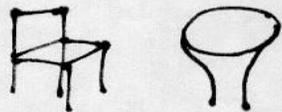
Foley

Rendering

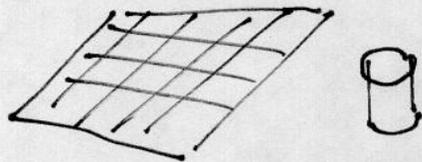
Construcción de la escena



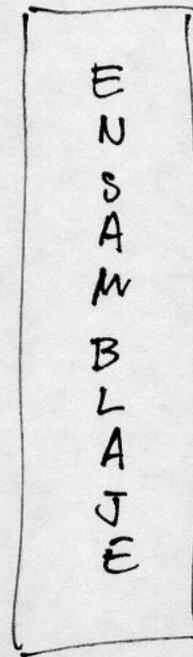
Objeto real



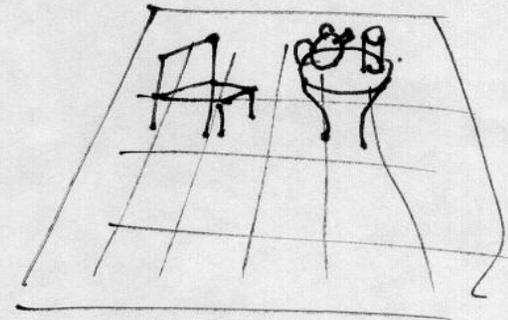
Biblioteca de
objetos



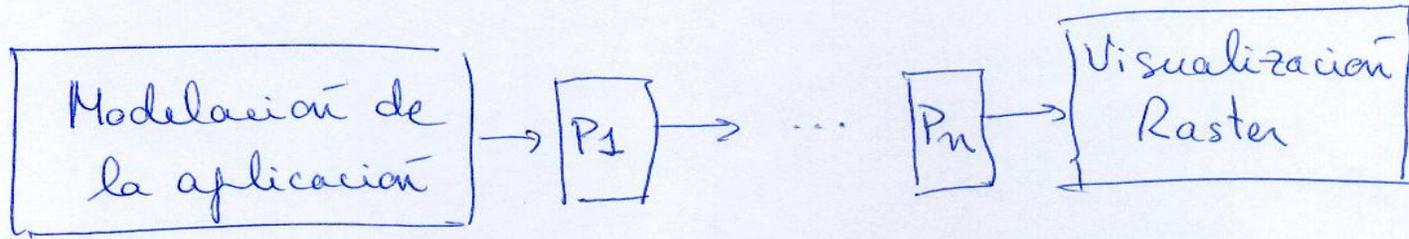
Primitivas
Gráficas



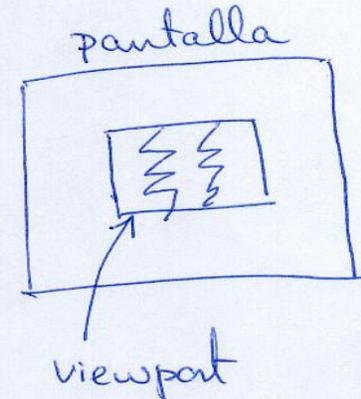
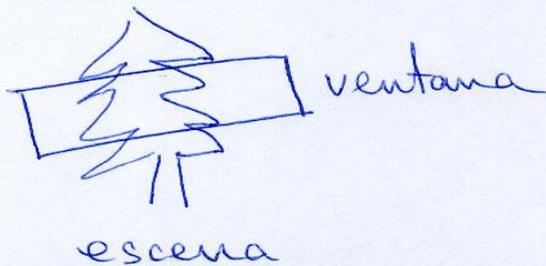
Escena
Completa



Modelo Conceptual Proceso de Visualización



Mundo 2D/3D



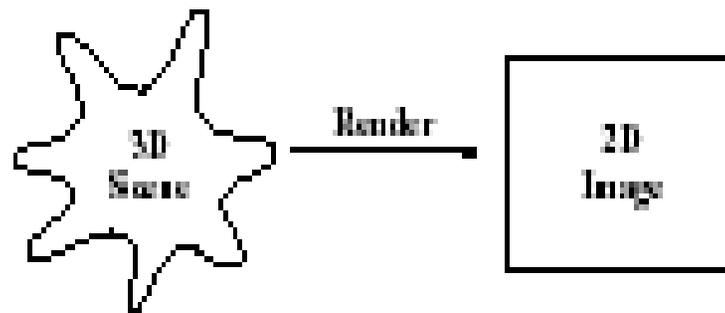
Sistema de coordenadas de la aplicación

Sistema coord. dispositivo

$P_1 \dots P_n$: pasos del proceso transformaciones, algoritmos (clipping Zbuffer, etc)

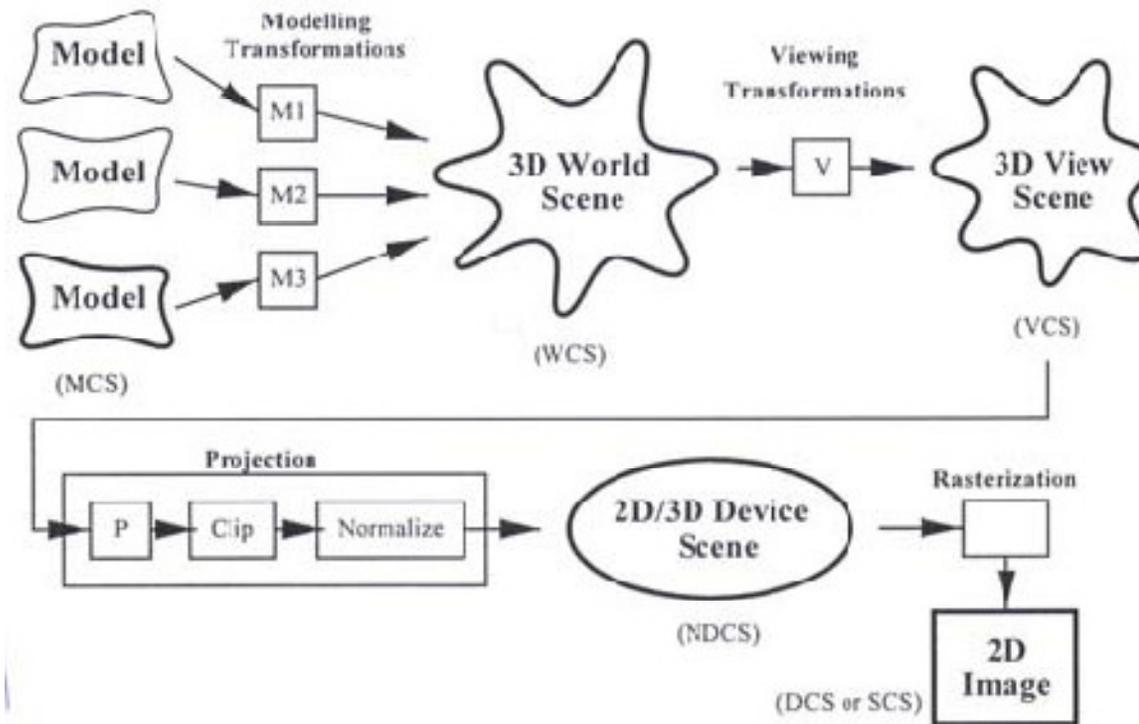
Rendering

El proceso completo que permite la conversión de una escena 3D en una imagen



Comentarios

- Las escenas se componen de modelos de los objetos: modelos de polígonos, triangulaciones, modelos de sólidos, etc.
- Los modelos se crean usando técnicas computacionales, se importan, o se construyen a mano.
- Los modelos se componen de primitivas soportados por el sistema de rendering: polígonos, triángulos, parches de Bezier, etc.
- Usaremos OpenGL, librería gráfica basada esencialmente en modelos de polígonos, y modelo de iluminación local.
- Visualización en monitor gráfico o impresora (dispositivos 2D). O se guarda archivo raster. Se requiere independencia del dispositivo.
- Clásicamente la conversión de la escena en una imagen se divide en secuencia de pasos que definen la pipeline gráfica



Hardware gráfico: GPU

- Parte del proceso pipeline se implementa en hardware gráfico para obtener velocidades interactivas adecuadas.
- Hardware moderno soporta programación por hardware
 - Se pueden implementar modelos de iluminación o transformaciones geométricas por hardware
 - Procesador gráfico programable: GPU
 - Es necesario conocer la matemática del tema y la arquitectura del hardware para usar la GPU

Elementos del rendering (1)

- Se usan varios sistemas de coordenadas
 - Sistema local de coordenadas de modelación, en general en 3D
 - Sistema global de coordenadas de mundo, en general en 3D
 - Sistema de coordenadas de visualización VCS (en la ventana de visualización en 2D)
 - Sistema normalizado de coordenadas del dispositivo en 2D
- Las primitivas se transforman al espacio del dispositivo donde se convierten en píxeles (se rasterizan)

Elementos del rendering (2)

- La Transformación entre dos sistemas de coordenadas se representa con una matriz que corresponde a una secuencia de transformaciones geométricas (producto de matrices)
- Se agrega información (de iluminación y pintado)
- Algunas primitivas se remueven (eliminación de superficies ocultas) o se modifican (clipping)

Lectura

Hearn – Baker, sección 6.1 El flujo de la visualización en 2D.

Primitivas

- Los modelos se componen de (o se convierten a) primitivas geométricas
 - Puntos
 - Segmentos de línea
 - Polígonos (en general convexos, triángulos)
- Primitivas de modelación
 - Polígonos
 - Polinomios curvos por pedazos (splines)
 - Superficies curvas por pedazos (splines)
 - Superficies implícitas (quadrics)
 - Otras
- **Software de Rendering** Puede soportar primitivas geométricas complejas o puede convertirlas en aproximaciones poligonales

Algoritmos y herramientas importantes

- Transformaciones (transformaciones geométricas, cambio de sist. de coordenadas, transformaciones de proyección, etc)
- Clipping (Recorte sobre ventanas 2D o 3D)
- Eliminación de superficies ocultas
- Rasterización: escena proyectada → conjunto de pixeles. Distintos formatos: pdf, postscript, etc.
- Otros algoritmos avanzados
- **Shading (Pintado-Sombreado) e Iluminación.** Simulan la interacción de la luz con una escena

Otros elementos de trabajo interactivo

- **Picking** selecciona objeto 3D haciendo click en dispositivo de entrada sobre la ubicación de un pixel.
- **Animación:** simula el movimiento haciendo rendering sobre una secuencia de cuadros (frames)

OpenGL (1)

- Interfaz de software para hardware gráfico. API (application programming interface)
- Provee acceso a hardware que realiza rendering vía un modelo conceptual. API oculta los algoritmos implementados por hardware.
- Interfaz independiente del hardware.
- Más de 150 funciones para modelar y renderizar escenas en base a polígonos

OpenGL (2)

- Trabajos con ventana o input del usuario no se incluyen en Open GL.
- OpenGL Utility Library (GLU) provee modelación de superficies cuádricas NURBS.
- Sintaxis de comandos.
- OpenGL como máquina de estado. Se establecen varios estados y modos que permanecen hasta que el usuario los modifica. Ej: color, proyección, etc.

¿Cómo trabaja un monitor?

Dispositivos Raster

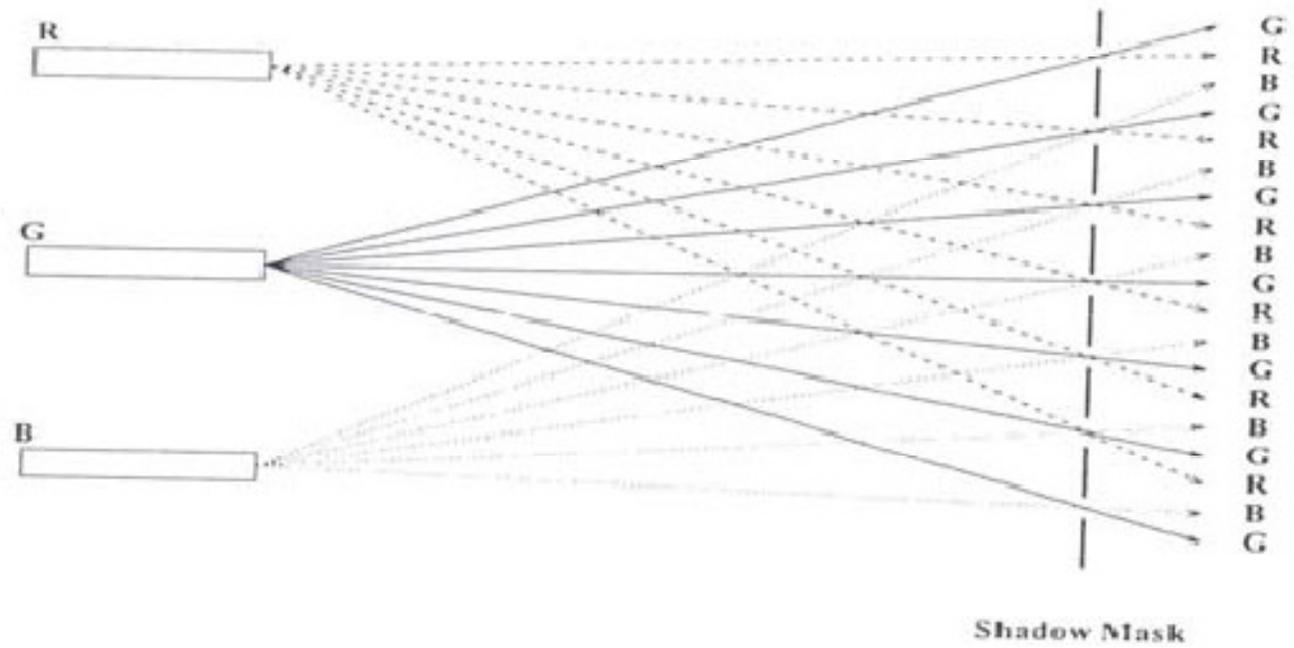
- **Tubo de Rayos Catódicos (CRTs):** dispositivo más usado
 - Haz de electrones produce estimulación de fósforo (pixel)
 - Alta resolución
 - Buenos colores
 - Alto contraste
 - Altas tasas de **refresco**

- **Pantalla de cristal líquido: LCD**

Dispositivos Raster

- El haz de electrones barre un patrón regular de líneas horizontales (**scanlines**)
- Imagen raster se almacena en el frame buffer
- Frame buffer: memoria VRAM (video RAM)
- **VRAM** es memoria dual capaz de
 - Acceso aleatorio
 - Salida serial simultánea de alta velocidad de scanlines completas a alta tasa sincronizado con un pixel clock.
- Tubo de rayos catódicos (CRT) a color usa 3 colores diferentes de fósforo y 3 cañones independientes de electrones
- En dispositivos LCD (pantallas de cristal líquido), las Shadow Masks permiten que cada cañon irradie solo un color de fósforo

Physical Devices



Pantalla de Cristal Líquido (LCDs)

- Pantalla plana
- Sin parpadeo

¿ Cómo funciona?

- Acceso aleatorio a celdas tipo memoria.
- Celdas contienen moléculas de cristal líquido que se alinean con la carga.
- Filtros polarizados permiten que la luz pase solo a través de moléculas desalineadas.

Especificación del color

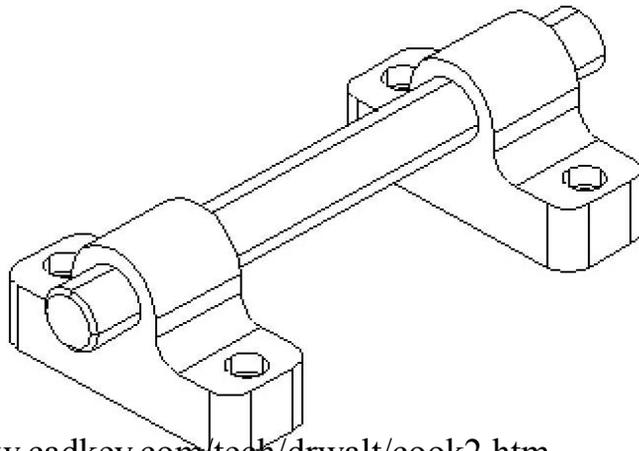
- Directamente a través del modelo RGB
 - Modelo RGB basado en tres colores primarios
 - Los colores se construyen como sumas ponderadas de los colores primarios
 - En el frame buffer se almacena n bits de información por pixel
 - ejemplo **24-Bit** , 24 bits por pixel, 8 bits por color
 - permite representar 16,777,216 colores (True color)

- Indirectamente usando Colour Lookup Table (mediante índice de color que se guarda en el frame buffer)

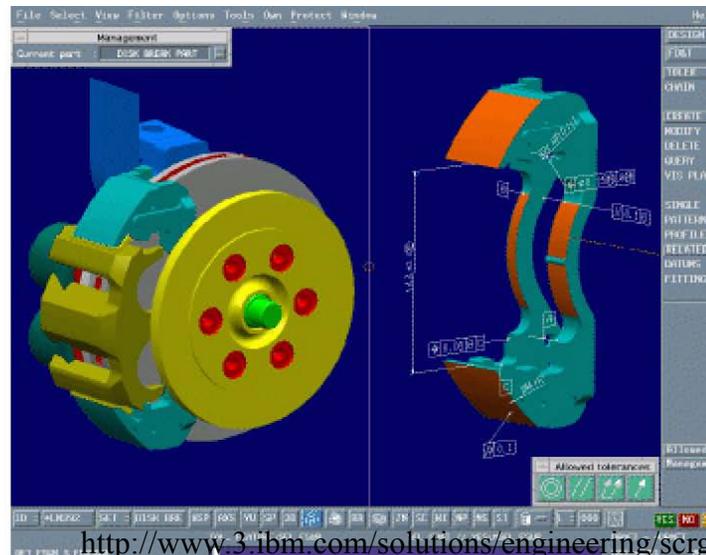
Algunos ejemplos

<http://www.siggraph.org/education/materials/HyperGraph/shutbug.htm>

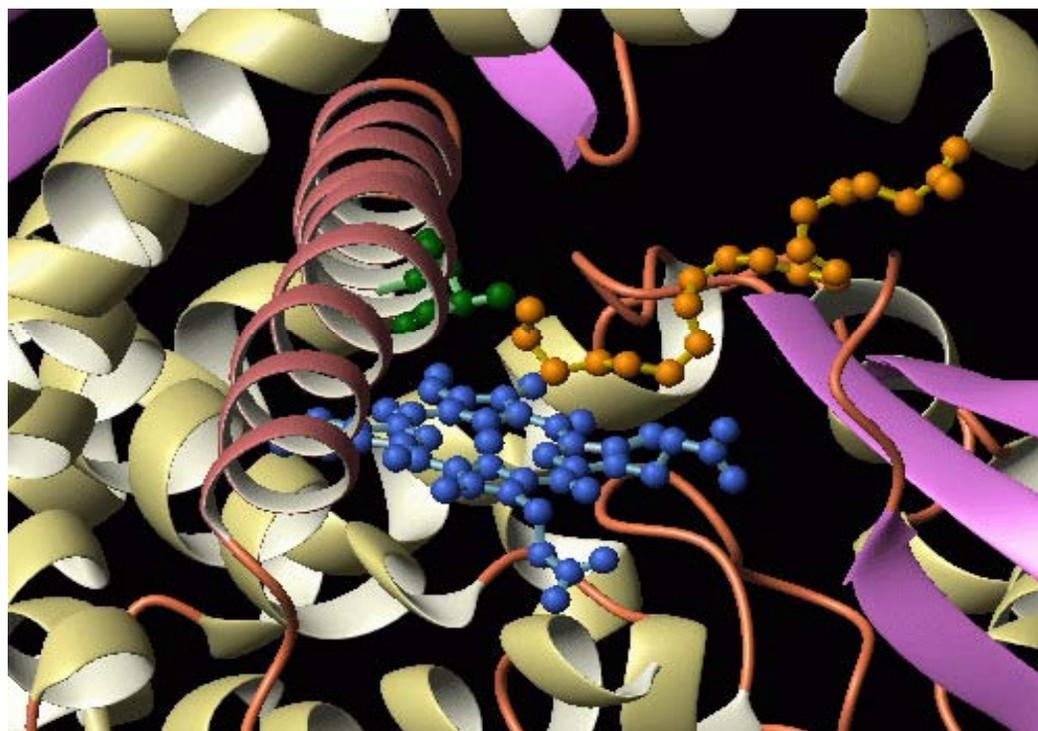
Visualisation "fil de fer" (CADKEY)



<http://www.cadkey.com/tech/drwalt/cook2.htm>



<http://www.3.ibm.com/solutions/engineering/scrgbs.fd416.gif>



MOLMOL <http://www.mol.biol.ethz.ch/wuthrich/molmol/cimages/vue1.gif>

Artes



<http://www.aec.at/prix>

Cine y juegos

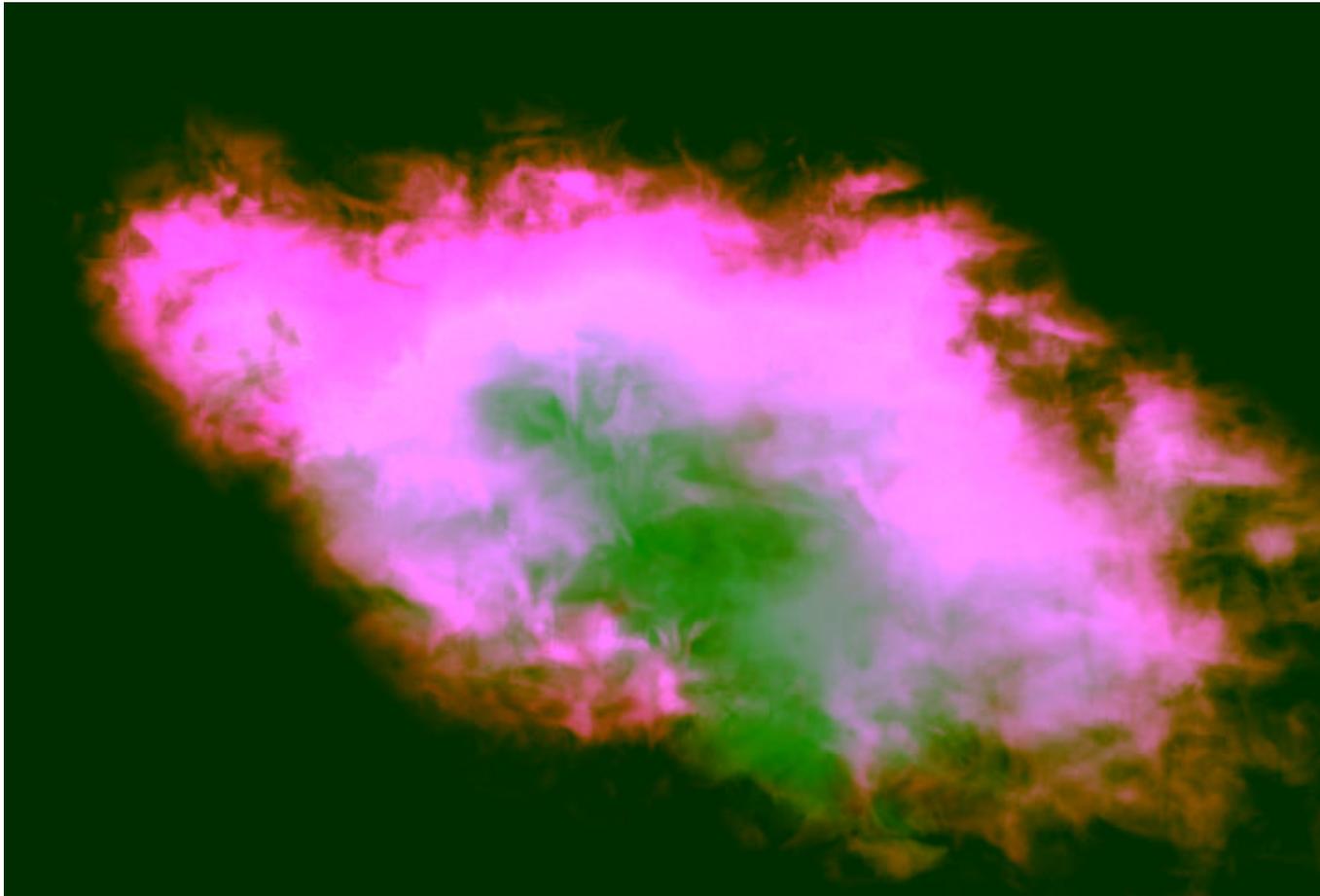


© G. Lucas “La guerra de las galaxias”

Métodos procedurales

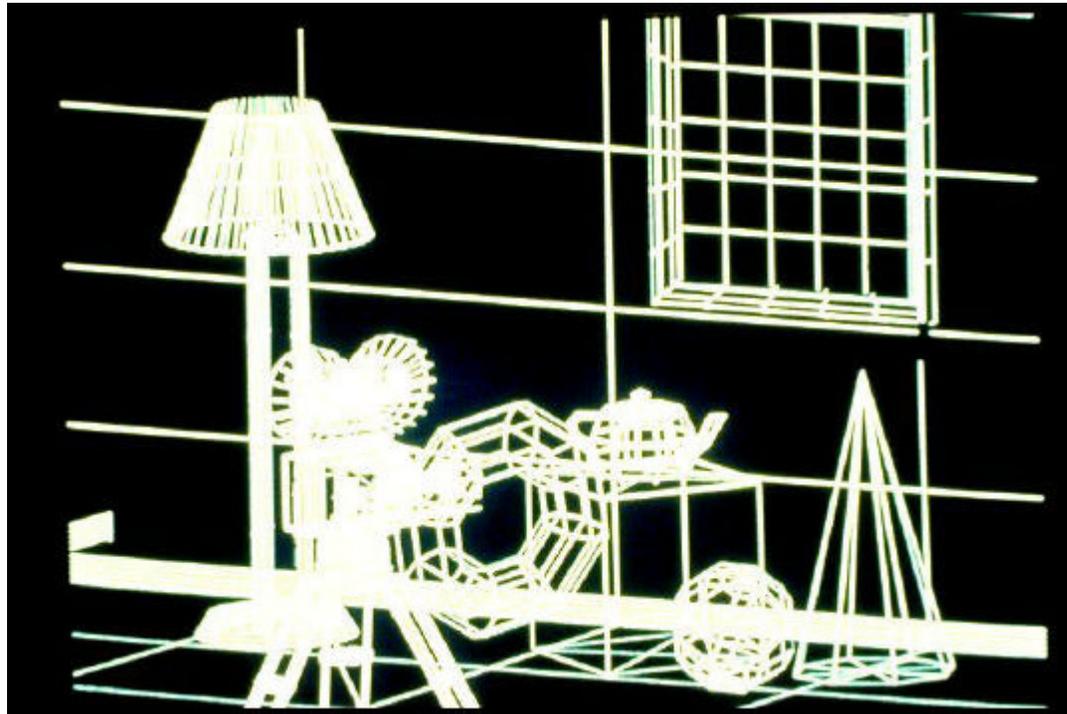


F.K. Musgrave “Alps” [Ebert et al.]

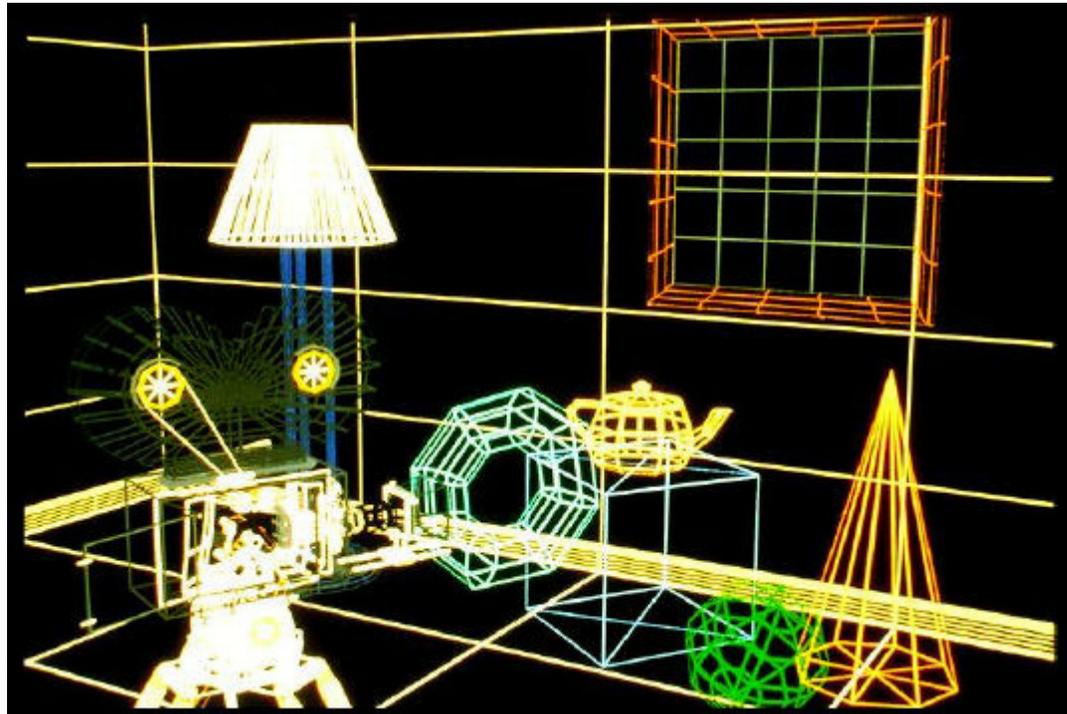


D.S. EBERT [Ebert et al.]

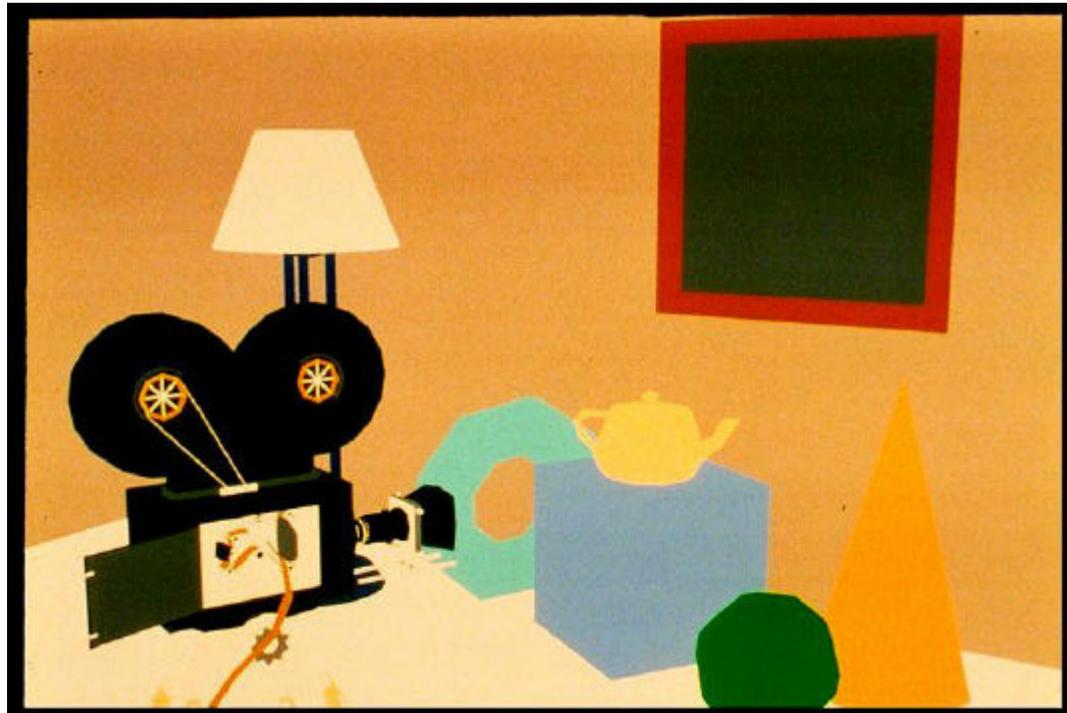
Serie “Shutterburg” de PIXAR (con Renderman)



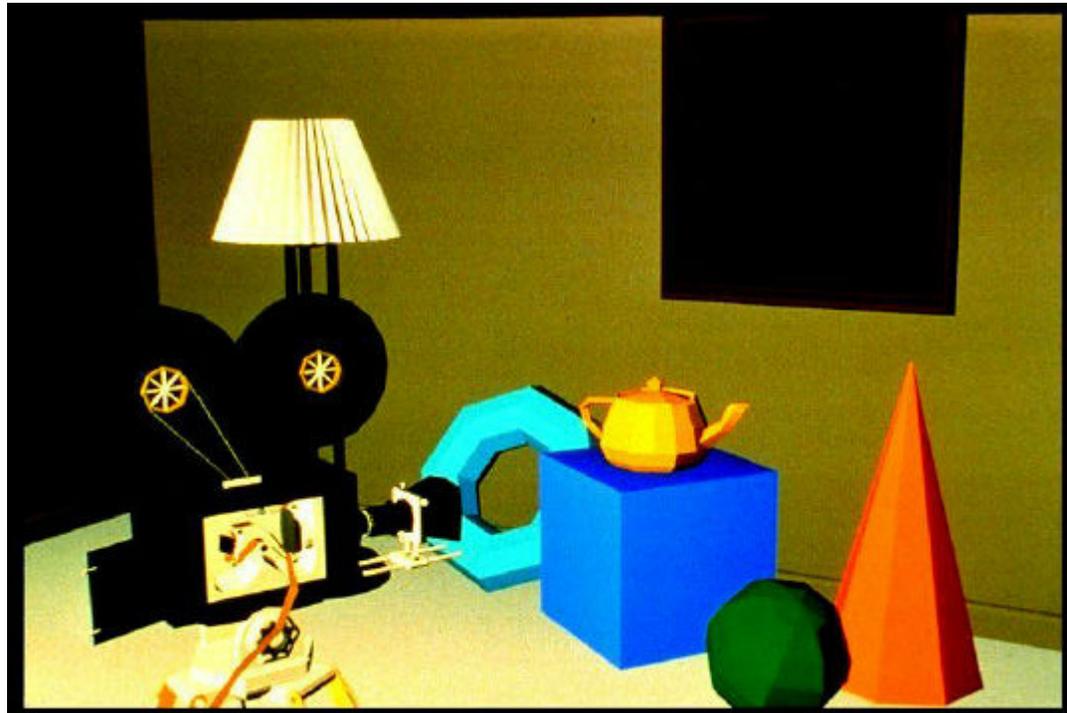
Malla de alambre (wireframe)



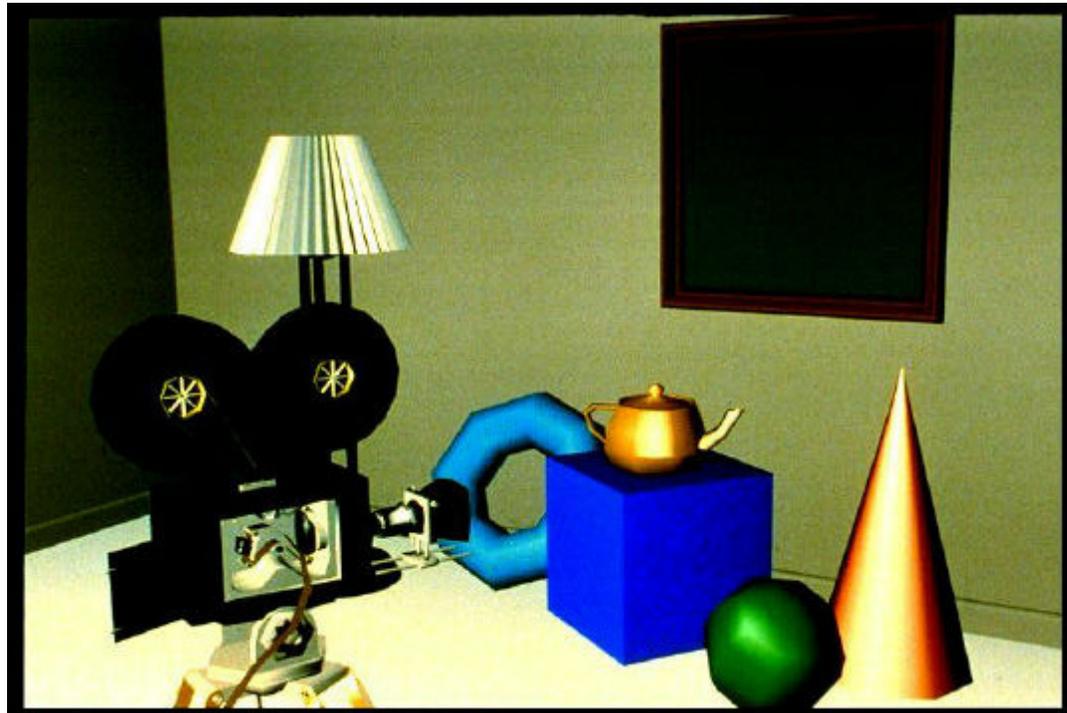
Wireframe coloreado



Coloreado uniforme



Coloreado por polígonos planos



Interpolación de Gouraud



Interpolación de Phong



Uso de texturas



Radiosidad: S. Feldman, J. Wallace Univ. Cornell