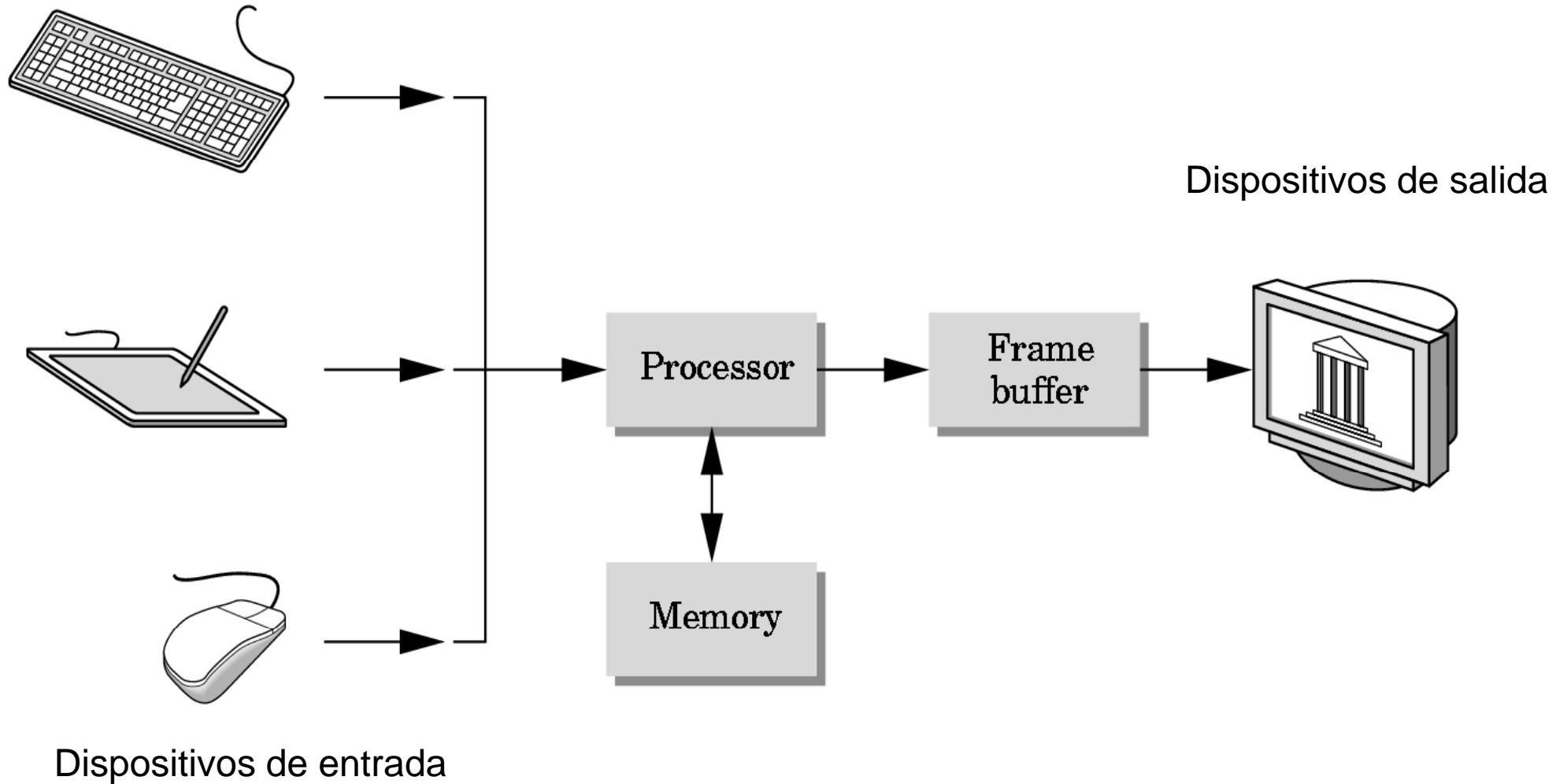


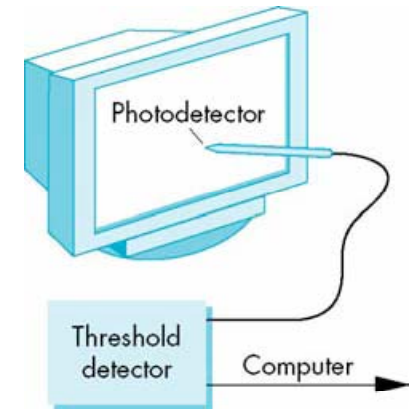
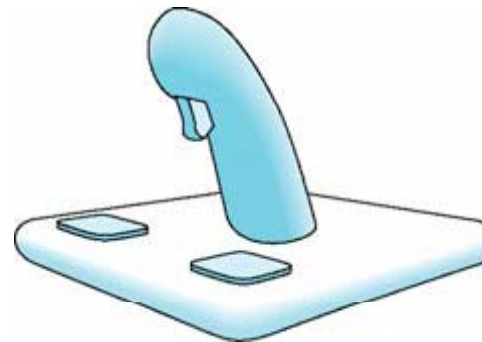
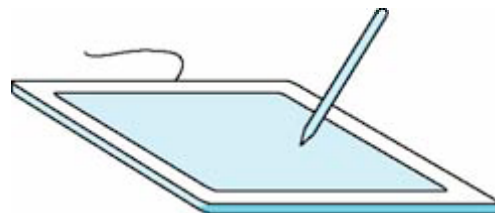
Sistema Gráfico



Dispositivos de Entrada



- ❑ Dispositivos apuntadores:
 - ❑ Localiza el apuntador en la pantalla
 - ❑ Mouse
 - ❑ Data tablet
 - ❑ Joystick
 - ❑ Touch pad / touch screen
 - ❑ Teclado



Dispositivos de entrada



Dispositivos de salida (CRT)

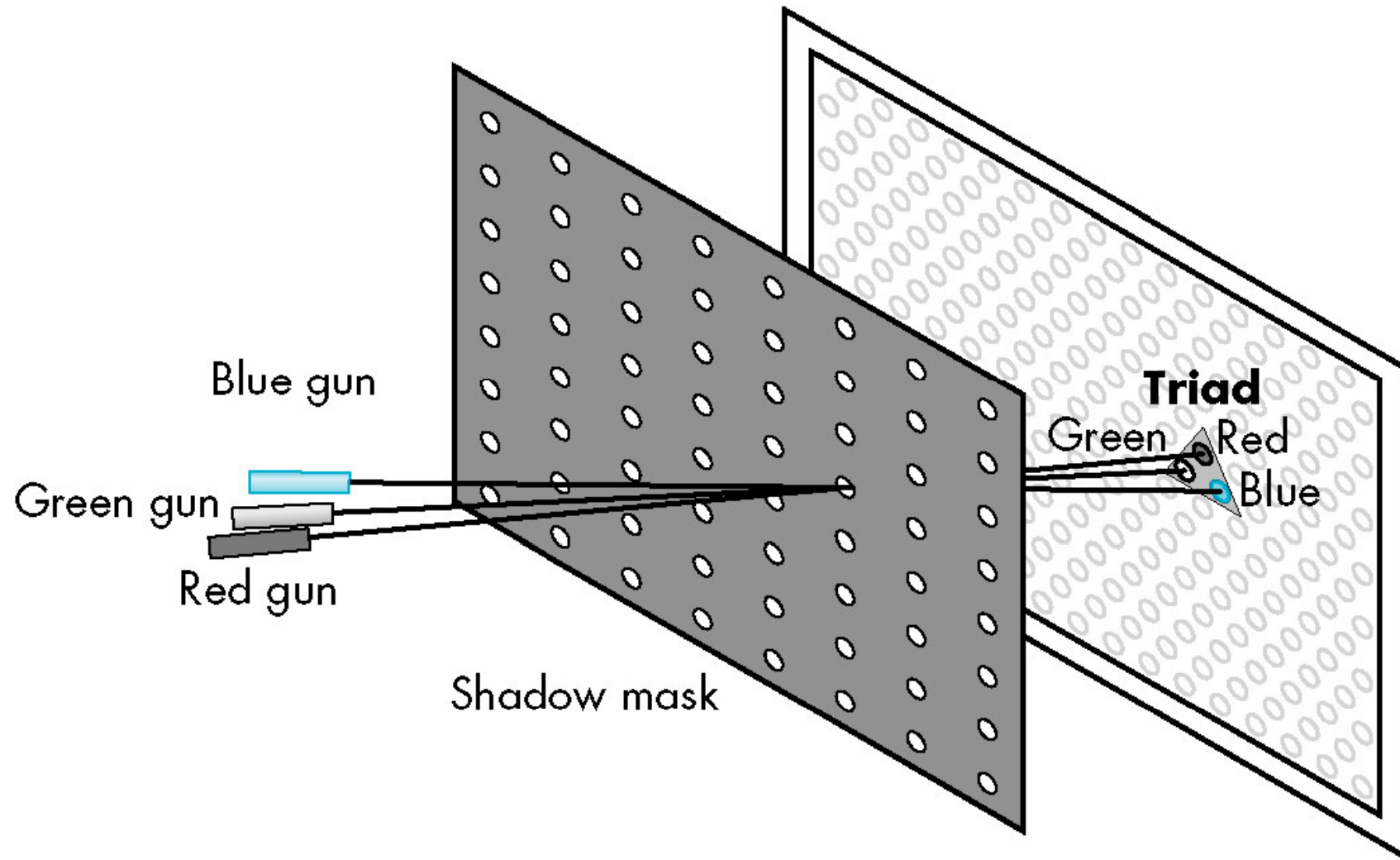


- ❑ La salida gráfica es almacenada en un *framebuffer* y convertida en voltaje por medio de un convertidor digital a analógico
- ❑ Se emite una luz
- ❑ Esta luz decae exponencialmente con el tiempo
- ❑ Refrescamiento de la pantalla
 - ❑ 60 Hz para evitar *flickers*

Dispositivos de salida



- Fosforos de colores



Dispositivos de salida



- ❑ CRT – no son tan utilizados ultimamente
- ❑ LDC – Pantallas de cristal líquido (Monitores planos)
 - ❑ Los colores son producidos a nivel de sub-pixel
 - ❑ No requieren de refrescamiento de pantalla

Pantallas HD – High Dynamics



LDC de alta resolución



HD

Frame Buffer



- Pixel
 - Una imagen esta compuesta por un arreglo o *raster* de *pixels*
- Los *pixels* son almacenados en una parte de la memoria llamada *frame buffer*
- *Frame buffer*
 - Profundidad
 - 1 bit => 2 colores
 - 8 bits => 2^8 , 256 colores
 - Sistema de color RGB
 - Resolución (número de *pixels* en un *frame buffer*)

Rasterization

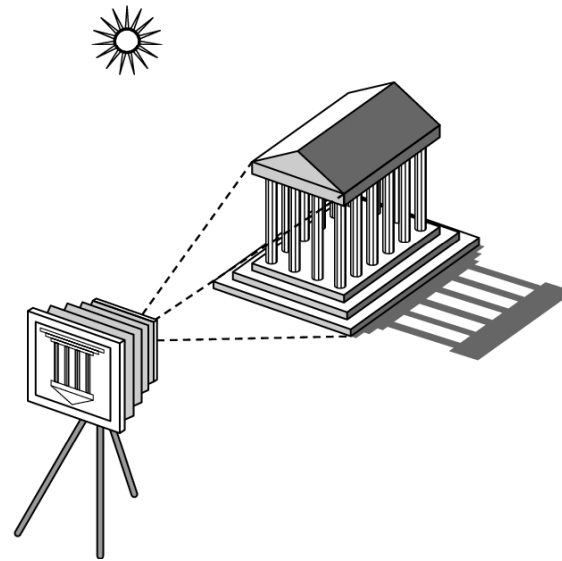


El proceso de conversión de las primitivas gráficas (líneas, puntos, círculos, etc.) generadas por una aplicación y las asigna a los *pixels* del *frame buffer* que mejor las representen

Formación de imagen



- Objeto
- Observador
- Fuente de Luz
- Material



- Propiedades que describen como interactúa la luz con el material en escena
- Independencia de los elementos



- ❑ Objeto
 - ❑ Se define en su propio espacio
 - ❑ Se forma a partir de primitivas geométricas (pologonos)
- ❑ Observador
 - ❑ Forma la imagen de los objetos según la proyección en un plano de imagen
 - ❑ Forma una imagen 2D
 - ❑ El objeto es transformado al espacio del observador



Fuente de Luz

- Sin luz es imposible generar la imagen
- Luz reflejada o transmitida
- Luz y el objeto tienen color
- Cuando interactúan el color de la luz con el color del objeto, la luz es:
 - reflejada
 - absorbida

Material



Basado en Muestra

- Pixel

- Adobe Photoshop™, GIMP™, Adobe AfterEffects™

Basado en Geometria

- Imágenes sintéticas

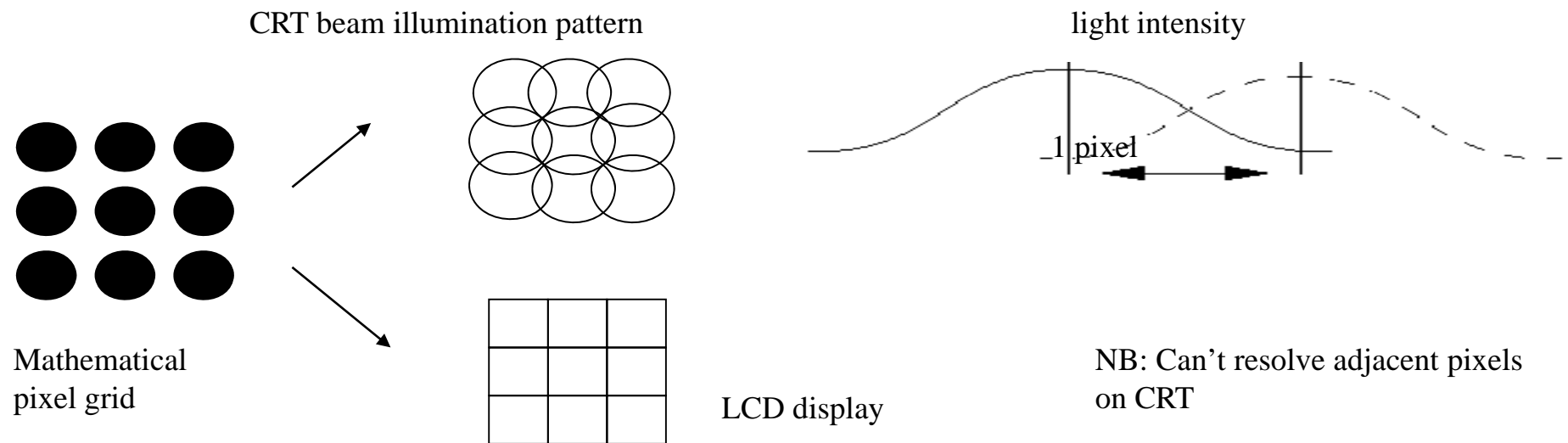
- Vectores graficos 2D como Adobe Illustrator™, Macromedia Freehand™, Corel CorelDRAW™ !

- Programas en 3D como Maya™, Autodesk's AutoCAD2005™, Discreet's 3D StudioMax™

Gráficos basado en muestra



- ❑ La imagen se forma a partir de una matriz (grid) de pixels



- Pixeles son puntos en la imagen con un valor asociado a la muestra (color, intensidad, transparencia, etc)

Gráficos basado en muestra



- ❑ A partir de la matriz de pixels
 - ❑ Editar la imagen (cortando parte de la imagen, pintando)
 - ❑ Procesamiento (modificar intensidad del color, brillo, suavizado de la imagen, realce, etc)

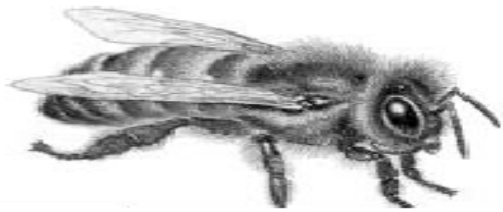
Gráficos basado en muestra



Escena 3D



Gráficos basado en muestra



Desventajas



Capture of Hair Geometry from Multiple Images, Siggraph 2004

Gráficos basado en geometría



- ❑ Son imágenes descritas a partir de un modelo geométrico
 - ❑ Líneas, polígonos, etc
 - ❑ Material, color
- ❑ Imágenes se crean muestreando el modelo a un arreglo de píxeles
- ❑ No se modifican píxeles individualmente sino se modifica el modelo y se vuelve a generar la imagen (rendering)

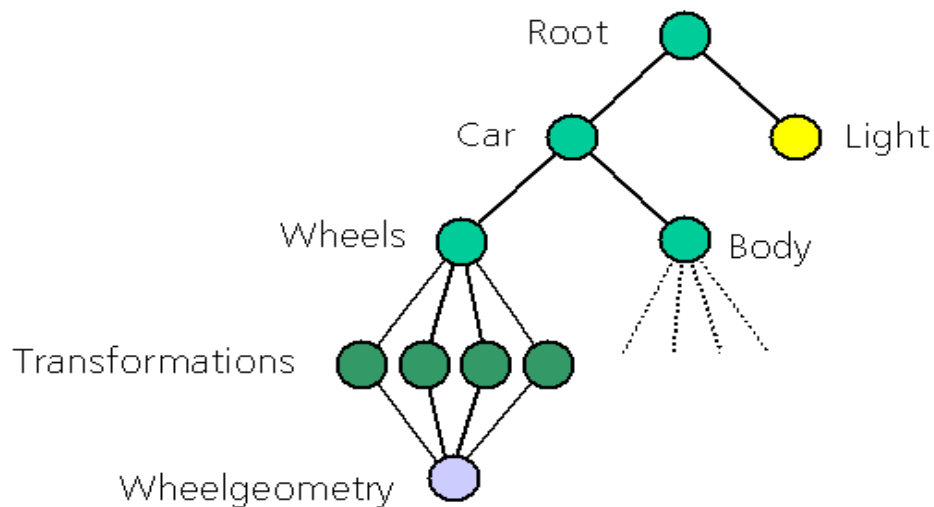


- ❑ Permite capturar el comportamiento de elementos o eventos físicos
- ❑ Real: inherente al objeto
 - ❑ físico (modelando a partir de un objeto)
 - ❑ no físico (función matemática)
- ❑ Abstracto: usado para visualización

Modelo geométrico



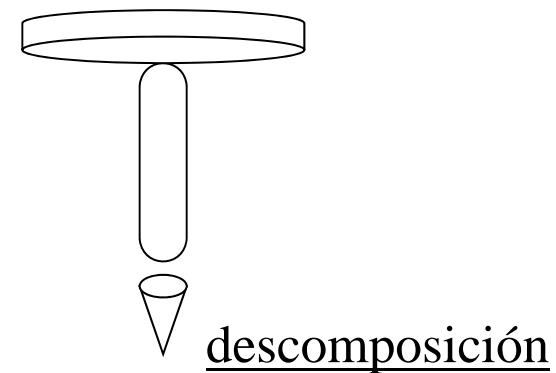
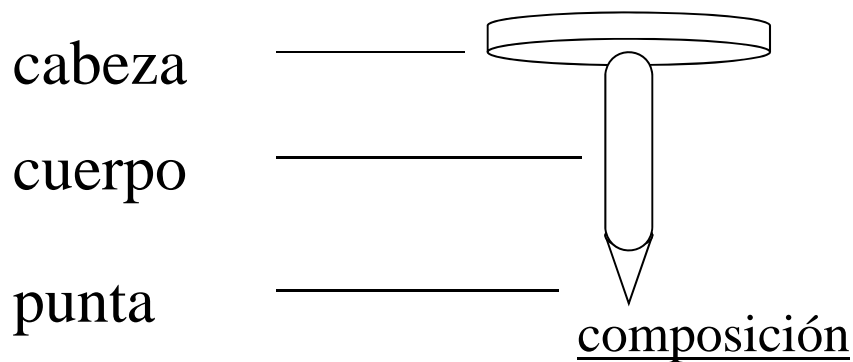
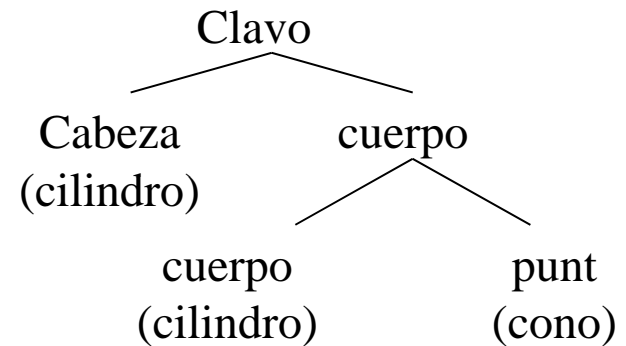
- ❑ Formas primitivas son manipuladas por transformaciones geométricas (traslación, rotación, escalamiento)
- ❑ Modelo jerárquico
- ❑ El modelo geométrico es desplegado en pantalla por medio de un mapeo de 3D a 2D para visualizar y de 2D a 3D para los dispositivos de entrada (rendering)



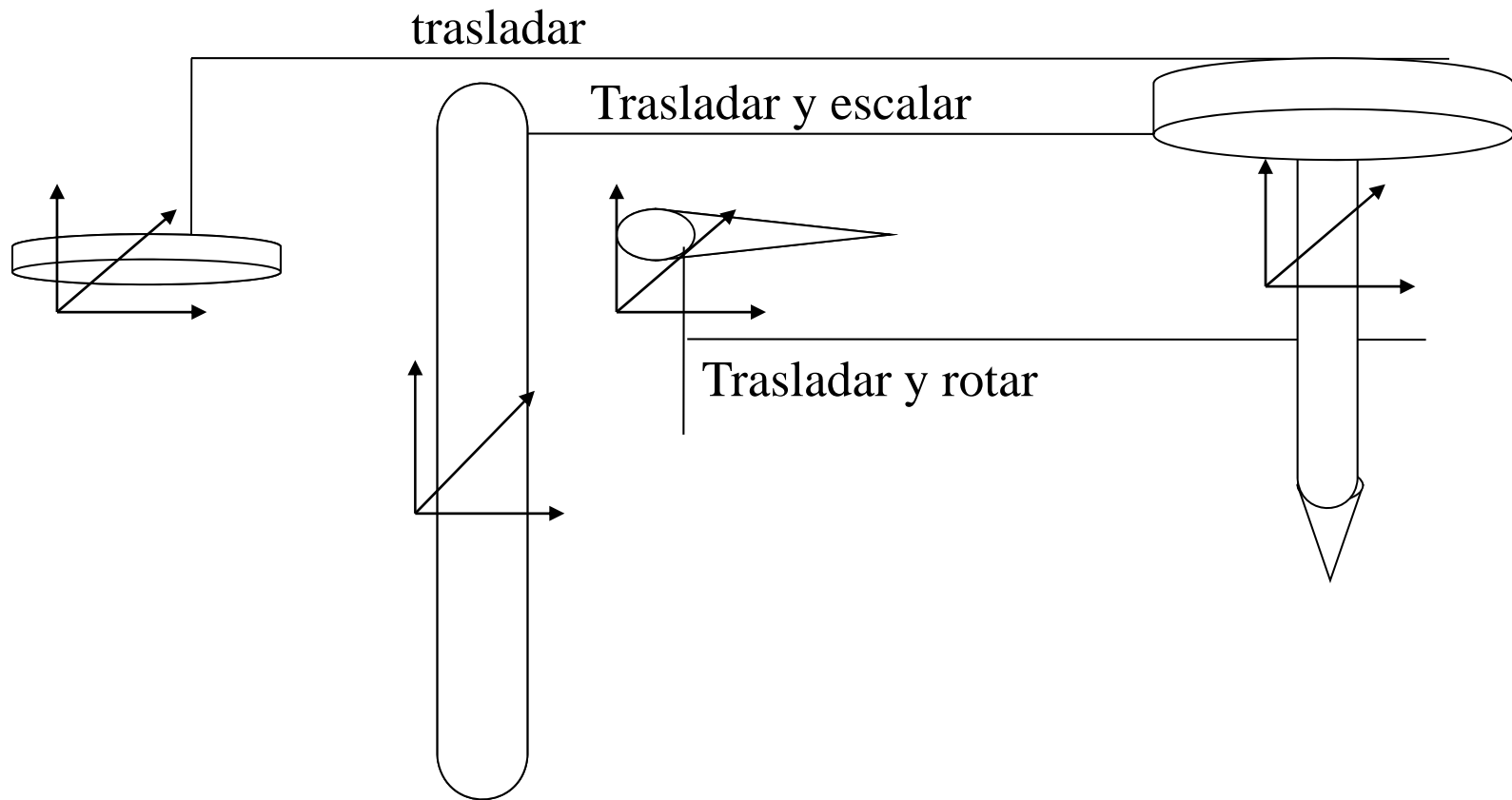
Descomposición de un modelo



- ❑ Jerarquía de los componentes geométricos
- ❑ Primitivas (cubo, cilindros, polígonos, etc)



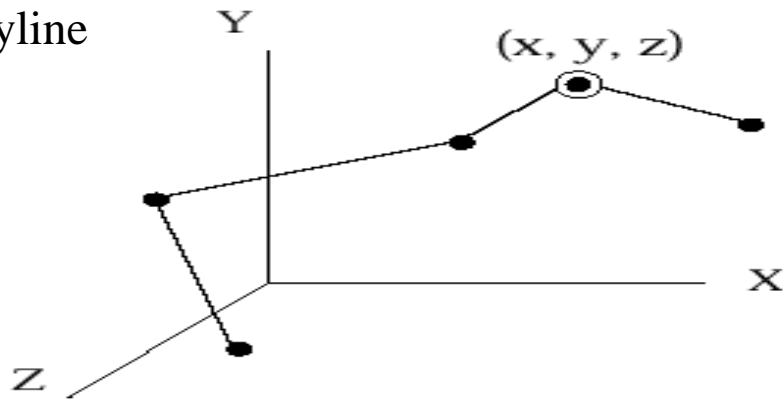
Composición de un modelo



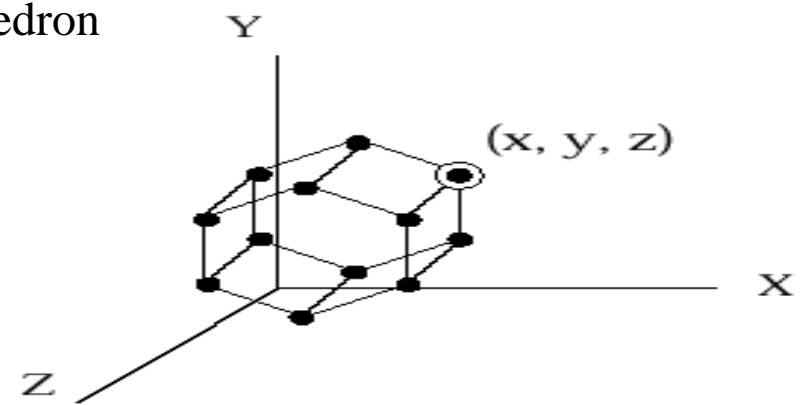
Primitivas 3D



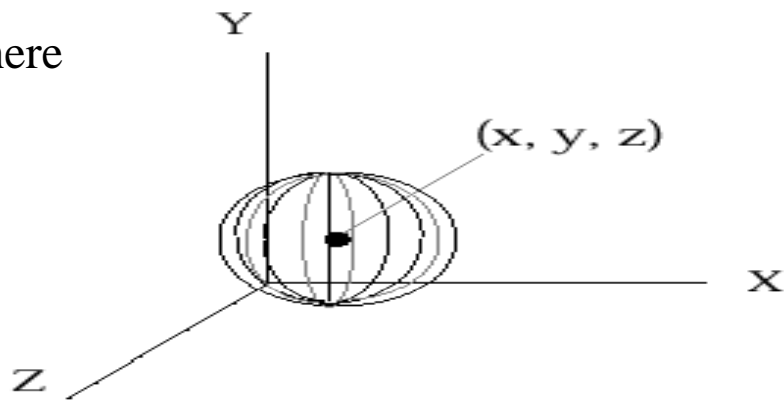
Polyline



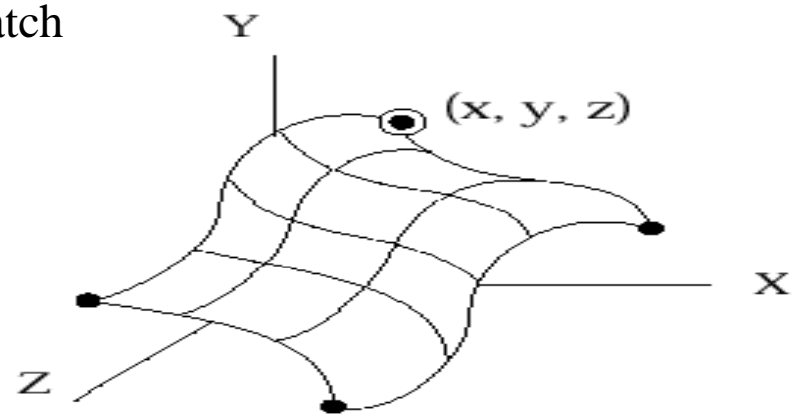
Polyhedron



Sphere



Patch





❑ **Ray tracing:** trayectoria del rayo de luz formado desde el centro de proyección hasta el objeto o al infinito

❑ Efectos globales

❑ Reflexión múltiple

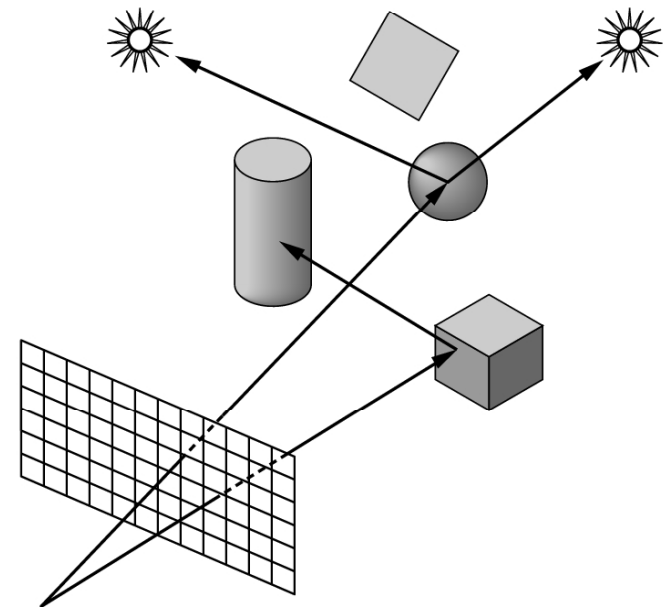
❑ Objeto traslucidos

❑ Lento

❑ Requiere de toda la data

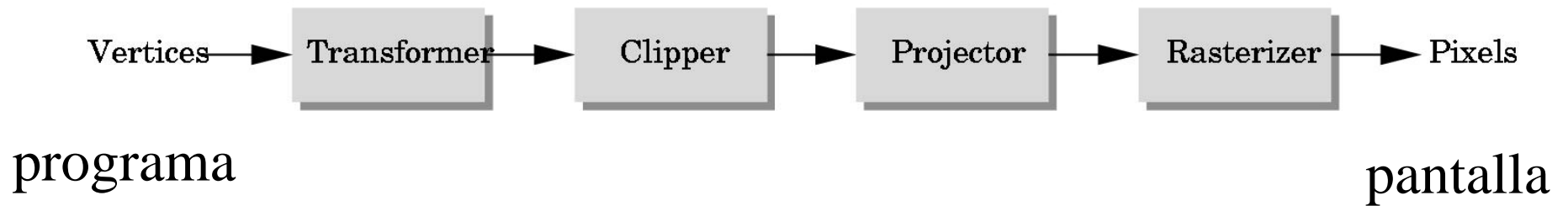
❑ **Radiosity:**

❑ Muy lento



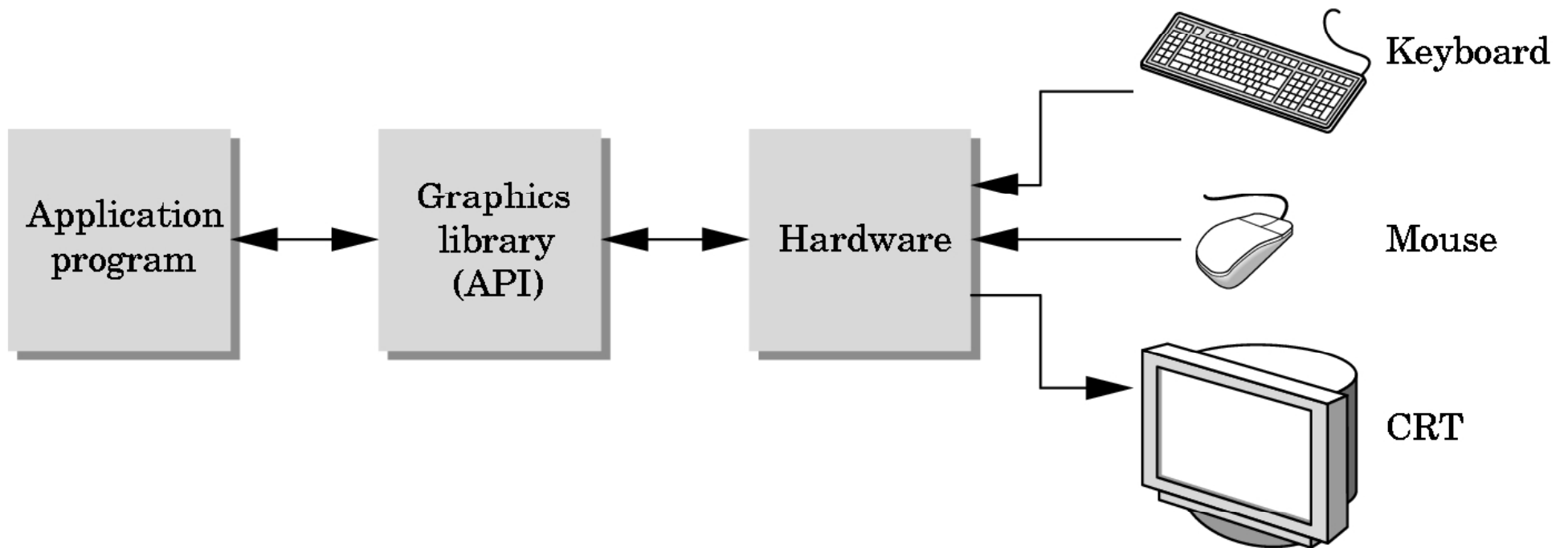


- ❑ Procesa un objeto a la vez en la medida en que son generados por la aplicación
 - ❑ iluminación local
- ❑ *Pipeline*



- ❑ Se pueden implementar en la tarjeta gráfica

API





- ❑ Funciones que especifican los objetos necesarios para formar una imagen
 - ❑ Objetos
 - ❑ Observador
 - ❑ Fuente de Luz (una o varias)
 - ❑ Material
- ❑ Adicionalmente
 - ❑ Dispositivos de E/S
 - ❑ Capacidades del sistema



- Primitivas gráficas
 - Puntos
 - Líneas
 - Polígonos
 - Curvas y superficies
 - Paramétricas
 - No paramétricas

Ejemplo



Tipo de objeto

Ubicación de un vertice

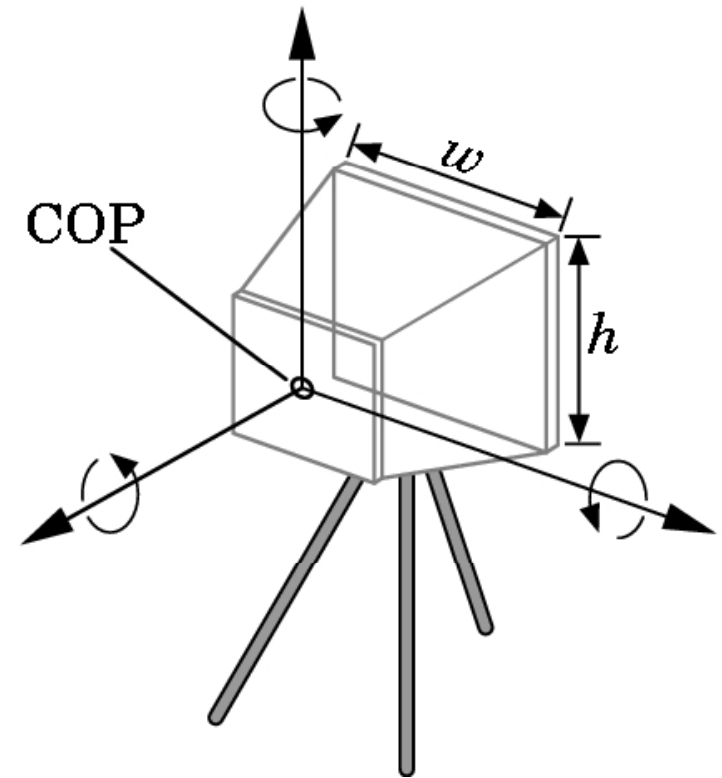
```
glBegin(GL_POLYGON)
  glVertex3f(0.0, 0.0, 0.0);
  glVertex3f(0.0, 1.0, 0.0);
  glVertex3f(0.0, 0.0, 1.0);
glEnd( );
```

Fin de la definición del objeto

Especificaciones de la cámara



- Grados de libertad
 - Posición del lente
 - Orientación
- Lentes
- Tamaño de la película
- Orientación del plano



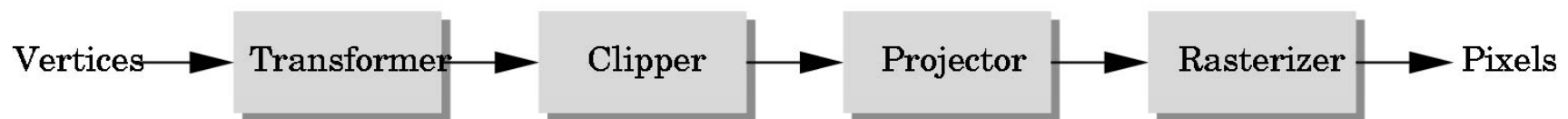


- Tipos de iluminación
 - Fuentes de luz puntuales o distribuidas
 - Spotlights
 - Fuentes de luz distantes
 - Fuentes de color
- Propiedades del material
 - Absorción
 - Tipo
 - Reflexión Difusa
 - Reflexión Especular

Transformaciones

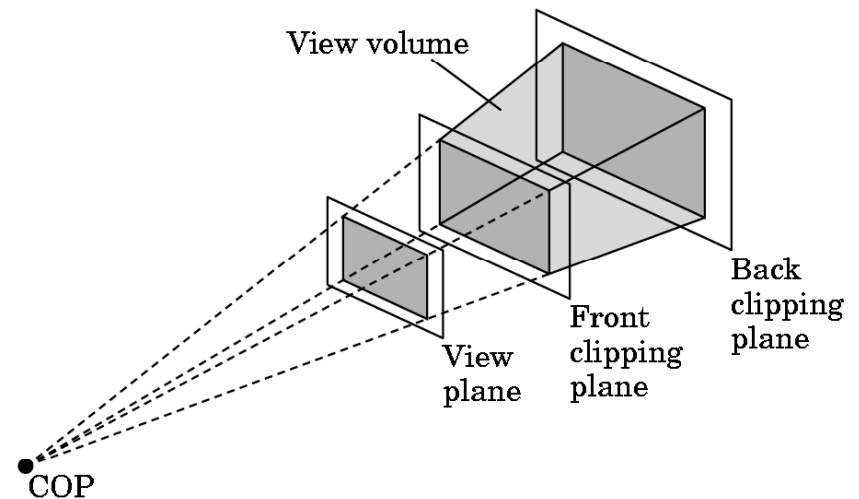
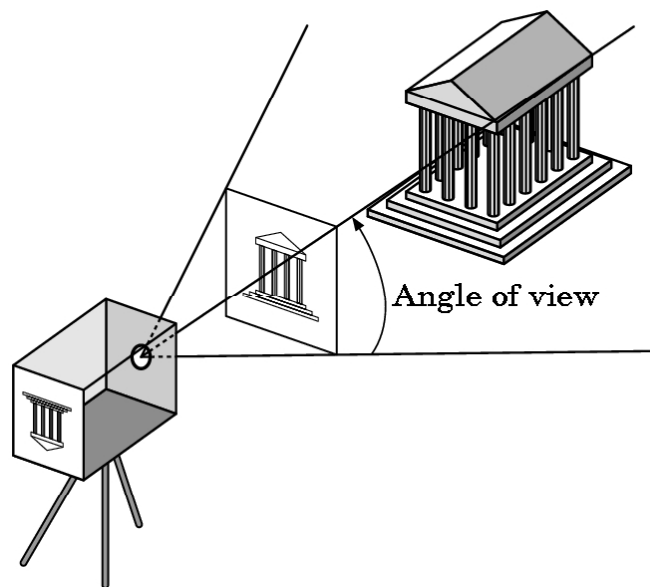


- ❑ Convertir la representación de un objeto de un sistema de coordenadas a otra
 - ❑ Coordenadas del espacio
 - ❑ Coordenadas de la cámara
 - ❑ Coordenadas de la pantalla
- ❑ Cada transformación utiliza una matriz de transformación





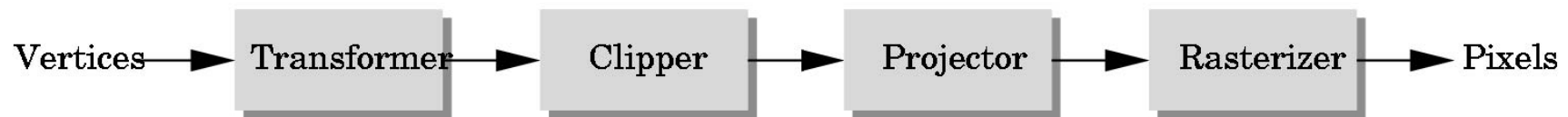
- ❑ Al igual que una cámara real, la cámara virtual no puede visualizar toda la escena
- ❑ Estos objetos que no se pueden visualizar en la escena se dice que son cortados (clipping out) de la escena



Proyección



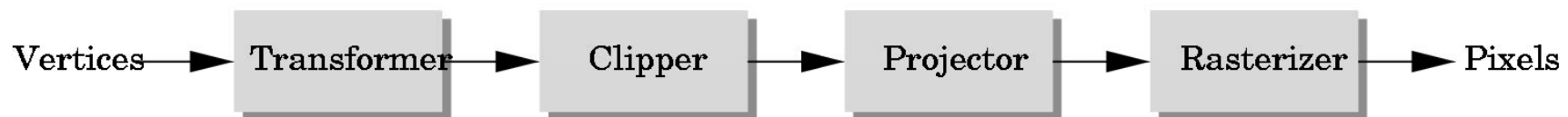
- ❑ Es el proceso que permite combinar un observador 3D con un objeto 3D para producir una imagen 2D
 - ❑ Proyección en perspectiva: los proyectores se encuentran en un centro de proyección
 - ❑ Proyección paralela: los proyectores son paralelos, el centro de proyección es reemplazado por una dirección de proyección



Rasterization



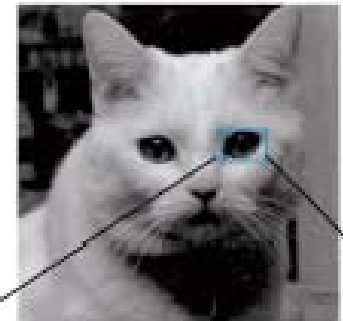
- ❑ Si un objeto es visible en la imagen, se le debe asignar un color al pixel correspondiente en el frame buffer
 - ❑ Los vertices construyen el objeto
 - ❑ Efectos de iluminación y material
 - ❑ Se rellenan los poligonos con color/sombra
 - ❑ Determinar cuales objetos van al frente (eliminar superficies escondidas)



Rasterization



- ❑ La pantalla es una matriz rectangular de puntos o pixeles
- ❑ Resolución: determina el nivel de detalle
- ❑ Número de pixeles en la imagen
 - ❑ 1024×768 , 1280×1024 , 1366×768 , etc.
 - ❑ También por ppi or dpi (pixel or dot per inch)



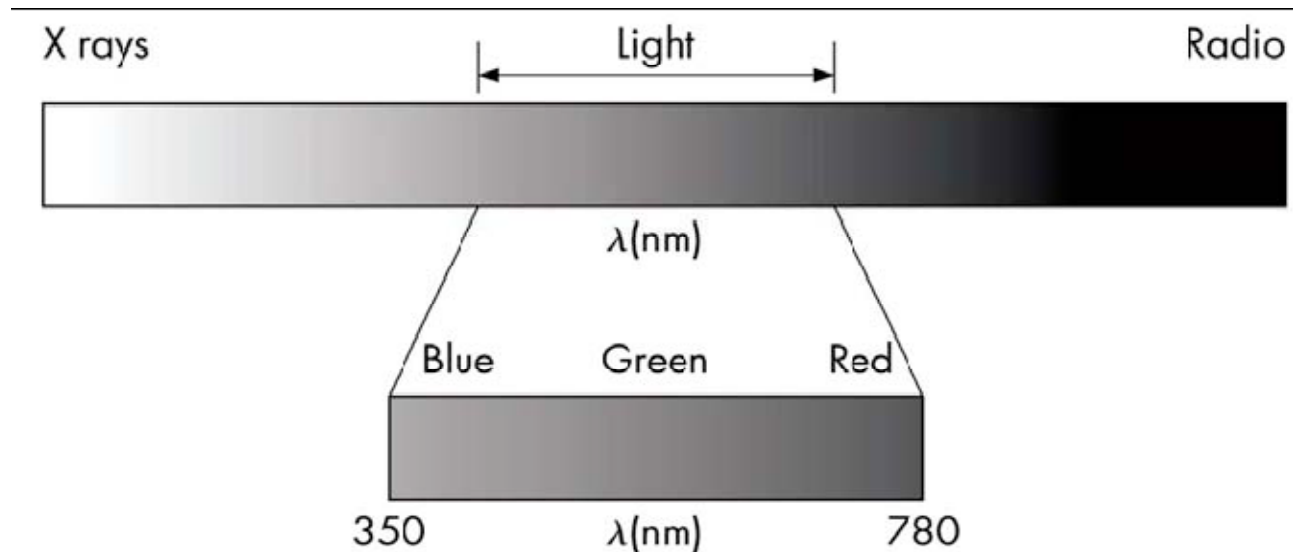
Iluminación y color



- Sin iluminación nada es visible
- Qué observa la cámara:
 - Luz reflejada
 - Luz transmitida
- Color reflejado
 - Fuente de luz y el objeto tienen color
 - Algunos colores son reflejados otros absorbidos



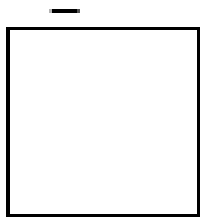
- ❑ La luz es parte del espectro electromagnético que causa una reacción a nuestro sistema visual
- ❑ Generalmente compuesto por longitudes de ondas entre 350-750 nm (nanómetros)
- ❑ Longitudes de ondas largas suelen ser más rojiza y longitudes de onda corta de color azul



Representación del color



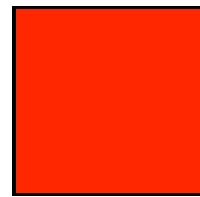
- ❑ Sistema de color RGB (R: rojo, G: verde, B: azul)
- ❑ Cada color tiene tres componentes: R, G, B
- ❑ 8 bits por componente: 24-bit
- ❑ Valor de cada componente: 0 – 255
- ❑ En OpenGL, los valores RGB estan en $[0, 1]$



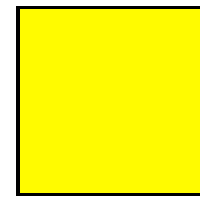
$(R,G,B) = (255,255,255)$



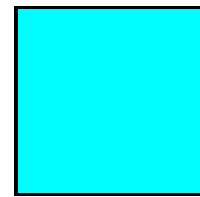
$(0,0,0)$



$(255,0,0)$

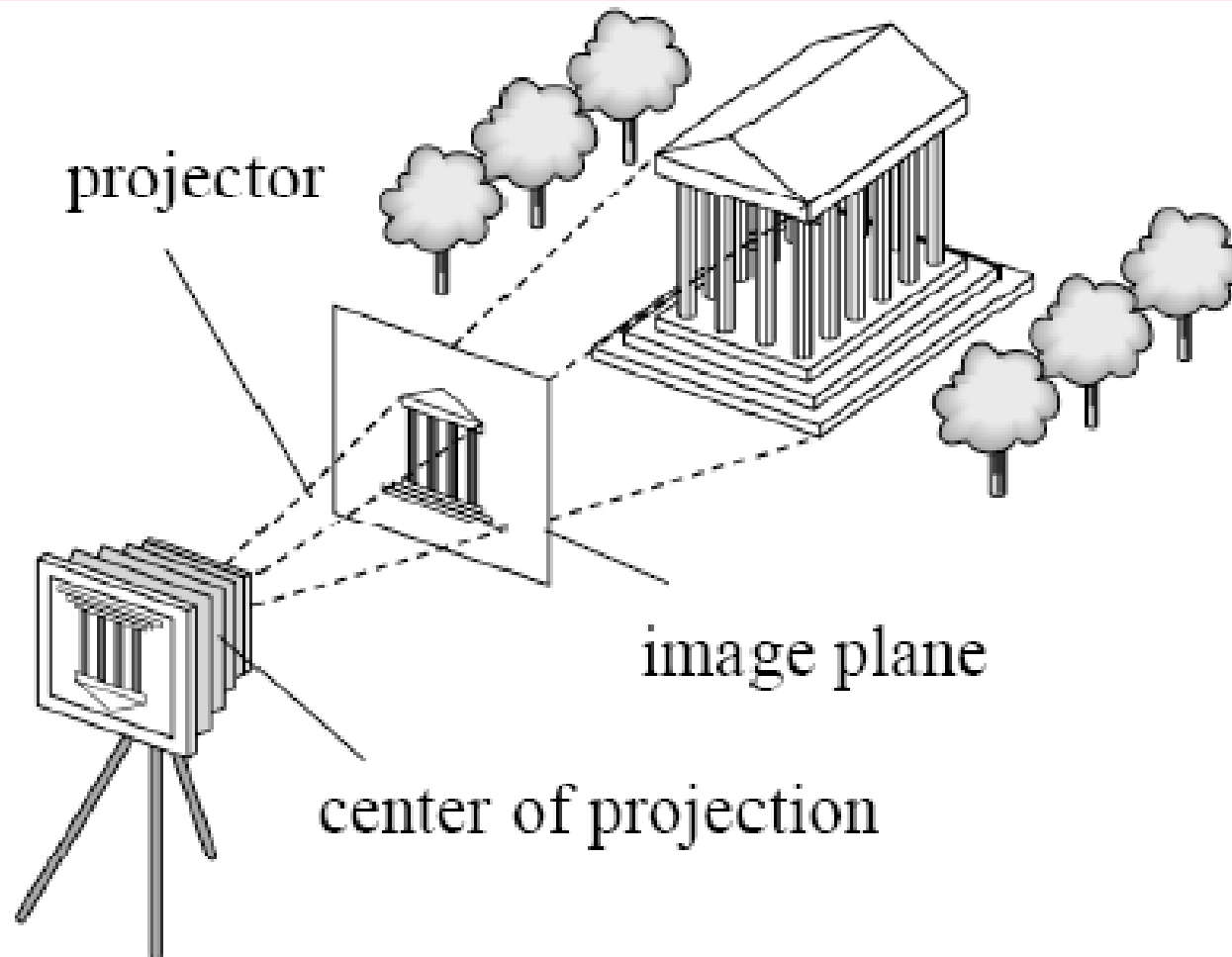


$(255,255,0)$



$(0,255,255)$

Modelo de la cámara

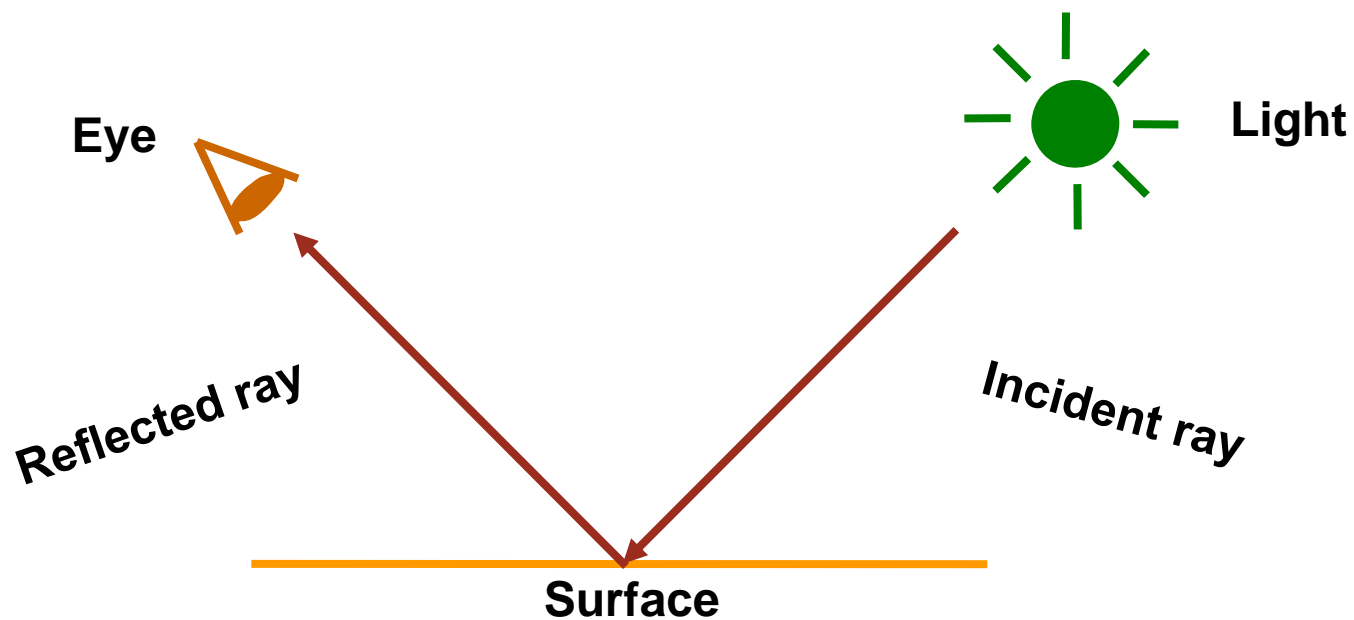


Imaging model adopted by 3D computer graphics

Idea básica



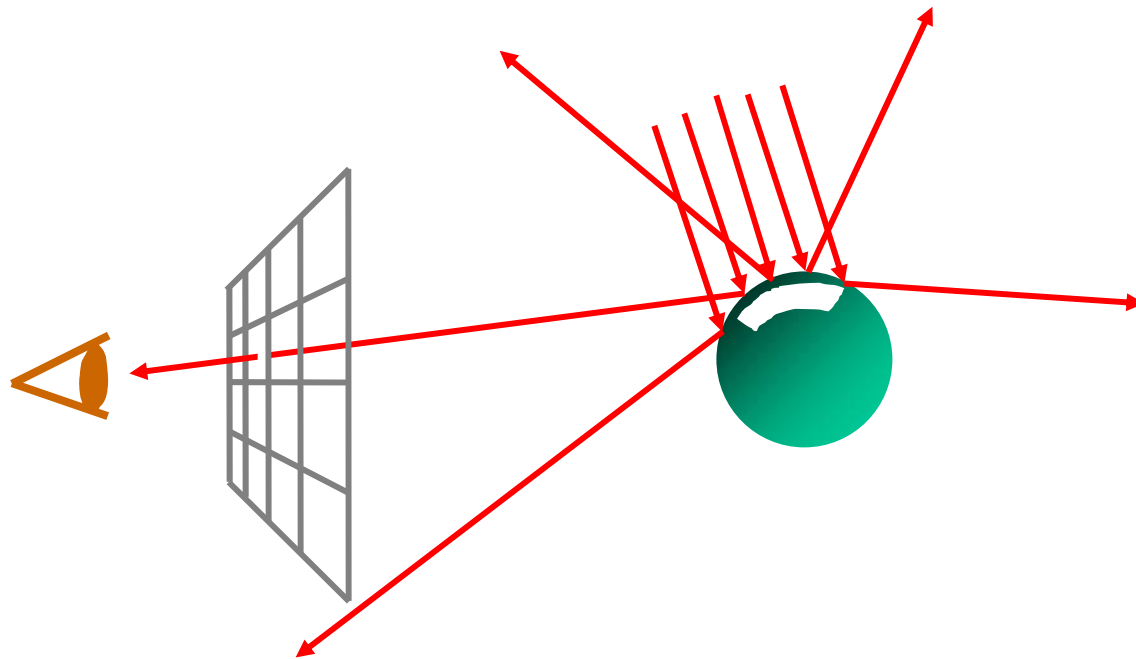
- Simular los rayos de luz desde la fuente hasta el observador



Ray Tracing



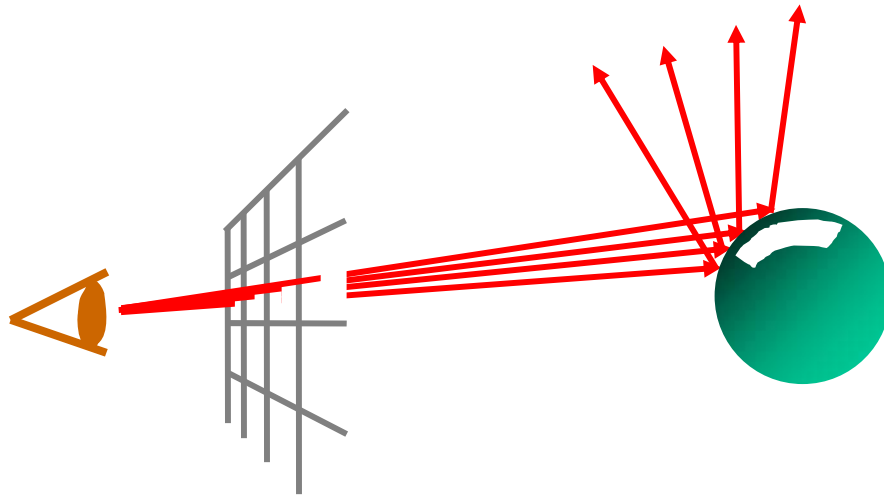
- ❑ Lanzar rayos desde la fuente de luz
- ❑ Mucho trabajo, poca contribución



Ray Tracing inverso



- ❑ Lanzar rayo desde el observador
- ❑ Trabaja solo en el área que nos interesa



Normalmente a esto nos referimos como Ray Tracing



© Machiraju/Zhang/Möller

28

The WingsPov Civilisation Museum by Eric Ouvrard



- ❑ Operaciones por poligono
- ❑ Más rapido y simple
- ❑ Procesa los poligonos en la medida en que se presentan en la imagen, no es necesario ordenarlos
- ❑ No esta diseñado para calcular el color
- ❑ OpenGL implements this fundamental algorithm

```
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB | GLUT_DEPTH); ...
glEnable(GL_DEPTH_TEST); ...
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT); ...
```

