

## **Métodos de ocultamiento**

Los algoritmos de detección de superficies ocultas son utilizados en gráficos por computadora para determinar qué objetos en una escena tridimensional son visibles desde una determinada perspectiva o cámara, y cuáles están ocultos detrás de otros objetos.

### **1. Z-Buffering (Buffer de Profundidad):**

- Definición: El Z-Buffering es un método que utiliza un buffer para almacenar la profundidad (coordenada Z) de cada píxel en la escena. A medida que se renderizan los objetos, se compara la profundidad de cada píxel con el valor almacenado en el buffer para determinar la visibilidad.
- Ventajas: Es eficiente y adecuado para escenas complejas. Puede manejar oclusiones y escenas con objetos no convexos.
- Desventajas: Requiere memoria adicional para almacenar el buffer de profundidad. No es adecuado para transparencias ni efectos más avanzados.

### **2. Algoritmo del Pintor (Painter's Algorithm):**

- Definición: En este método, los objetos se dibujan en orden de profundidad, desde los más lejanos hasta los más cercanos. Los objetos más cercanos pueden ocultar los objetos detrás de ellos.
- Ventajas: Es simple y fácil de implementar. Puede funcionar bien para escenas con objetos convexos.
- Desventajas: No maneja bien los objetos no convexos y puede sufrir del "aliasing de polígonos" en escenas complejas.

### **3. Ordenamiento de Bordes (Edge Sorting):**

- Definición: Este método se basa en la ordenación de los bordes de los polígonos en función de su coordenada Y. Luego, se rellenan los píxeles entre los bordes en el orden correcto.
- Ventajas: Es útil para objetos planos y es eficiente en términos de memoria.
- Desventajas: No es adecuado para escenas tridimensionales complejas y puede no manejar bien la intersección de polígonos.

#### **4. Algoritmo de Puntillismo (Scanline Algorithm):**

- Definición: Divide la pantalla en líneas horizontales (scanlines) y procesa cada scanline por separado, determinando la visibilidad de los objetos en esa línea y rellenando los píxeles visibles.
- Ventajas: Es eficiente para objetos planos y escenas relativamente simples.
- Desventajas: No es adecuado para escenas tridimensionales complejas y puede no manejar bien la transparencia.

#### **5. Algoritmo de Ray Tracing:**

- Definición: Ray tracing simula el comportamiento de la luz al seguir rayos de luz desde el punto de vista de la cámara y determinar qué objetos son visibles. Los rayos no bloqueados generan la imagen final.
- Ventajas: Produce imágenes realistas con sombras suaves y reflejos. Puede manejar escenas muy complejas y efectos avanzados.
- Desventajas: Es computacionalmente intensivo y lento en tiempo real. Requiere mucho poder de procesamiento y es menos eficiente que otros métodos para la interactividad en tiempo real.

#### **6. Algoritmo de Eliminación de Superficies Ocultas (Hidden Surface Removal - HSR):**

- Definición: Los algoritmos de HSR incluyen varios métodos, como el Z-Buffering y el algoritmo del pintor, que se utilizan para eliminar oclusión de superficies en gráficos 3D.
- Ventajas: Proporciona una variedad de opciones para la detección de superficies ocultas, lo que permite adaptarse a diferentes escenarios.
- Desventajas: La elección del método HSR adecuado depende de la escena y los requisitos de rendimiento específicos.