

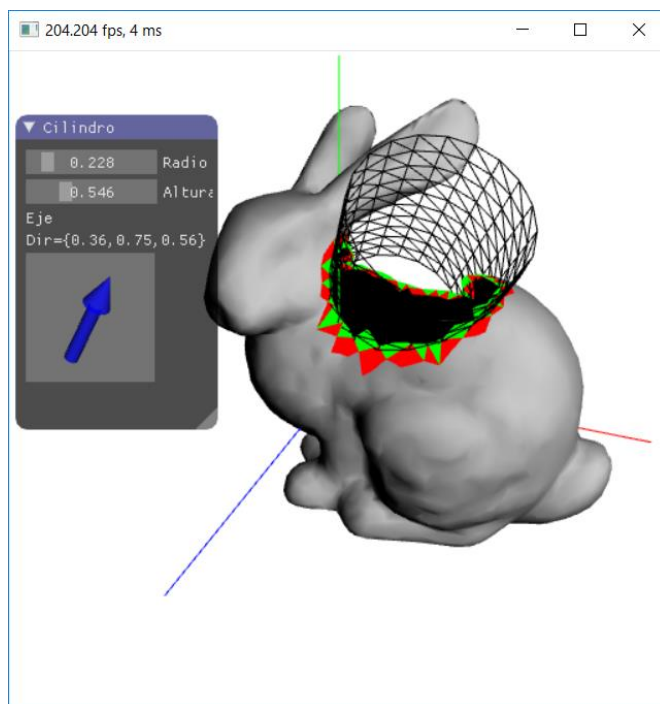


Práctica 6

Objetivo

En esta práctica se estudiará el uso de los shaders de geometría para resolver un problema que no se puede resolver usando únicamente shaders de vértice y fragmento.

Tu trabajo



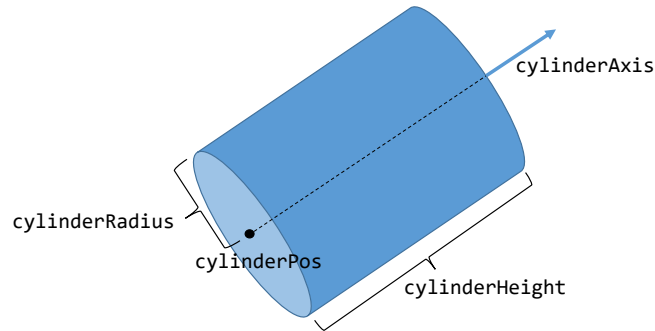
El objetivo de esta práctica es simular un volumen de recortado cilíndrico que descarte los triángulos de la escena que caigan en su interior. En la figura anterior, el cilindro ha descartado aquellos triángulos del modelo que caen completamente dentro del volumen. Los triángulos cuyos vértices se encuentran fuera del volumen no se modifican, y el resto de los triángulos se han cambiado de color: aquellos triángulos que sólo tienen un vértice dentro del volumen se dibujan en verde, y aquellos triángulos que tienen dos vértices dentro en rojo.

Deberás implementar un shader de geometría que realice dicho procesamiento.

La aplicación mantiene y actualiza las siguientes variables de tipo uniform que describen la posición, orientación y tamaño del cilindro en el sistema de coordenadas del mundo. Deberás definir en tu shader las variables del siguiente modo:

```
uniform float cylinderRadius; // radio del cilindro
uniform float cylinderHeight; // altura del cilindro
uniform vec3 cylinderAxis;    // dirección del eje (vector unitario)
uniform vec3 cylinderPos;     // centro de la base del cilindro
```

La siguiente figura muestra el significado de cada variable:



Material entregado

En esta práctica se entrega una aplicación que muestra el modelo sombreado del conejo y el cilindro en movimiento dibujado en alámbrico. La aplicación también permite al usuario definir el volumen del cilindro y su orientación. Deberás crear un shader de geometría que se encargue de procesar los triángulos del modelo a recortar según se ha descrito en el apartado anterior. Guarda el shader de geometría junto a los shaders proporcionados (en el directorio del proyecto p6), y llámalo **p6.geom** para que se cargue automáticamente.

Ampliaciones opcionales

Como ampliación para obtener más nota puedes implementar otros tipos de volúmenes de recortado (como planos o cubos), o varios volúmenes de recortado simultáneos. También puedes simular un recortado de los triángulos para que, en caso de que haya algún vértice dentro del cilindro (pero no todos), el shader subdivida el triángulo original en varios más pequeños, y aplique el procesamiento anterior a cada uno de los triángulos generados (habrá subtriángulos que desaparezcan, o subtriángulos que se mostrarán con el color original, en color rojo o en color verde, dependiendo del número de vértices que caigan dentro del volumen).