

TENDENCIAS EN TÓPICOS EN COMPUTACIÓN GRÁFICA

Integrantes: Kelvin Paul Pucho Zevallos
Angelo Aldo Perez Rodriguez
Luis Armando Sihuinta Perez
Josnick Chayña Batallanes

Profesor: Diego Alonso Iquira Becerra

Fecha de realización: 7 de junio de 2022

Fecha de entrega: 8 de agosto de 2022

Arequipa - Perú

Índice de Contenidos

1. Culling	1
1.1. Introducción	1
1.2. Occlusion Area	1
1.3. Occlusion culling	2
1.3.1. Cuando usar occlusion Culling	2
1.3.2. Como funciona el occlusion culling	3
1.4. Ejemplos	3
Referencias	5

Índice de Figuras

1. Occlusion Area	2
2. Uso de Occlusion Culling con areas	3
3. Occlusion Culling OFF - Occlusion Culling ON	4

1. Culling

1.1. Introducción

Es una función que desactiva la representación de objetos cuando la cámara no los observa actualmente porque están oscurecidos (ocuidos) por otros objetos. Esto no sucede automáticamente en los gráficos 3D presentados de manera predeterminada en la computadora, ya que la mayoría de las veces los objetos más alejados de la cámara se dibujan primero y los objetos más cercanos se dibujan sobre ellos (esto se denomina "sobredibujar"). El Occlusion Culling es diferente de Frustum culling. Frustum Culling solo deshabilita los procesadores de objetos que están fuera del área de visualización de la cámara, pero no deshabilita nada oculto a la vista por sobredibujado. Teniendo en cuenta que Occlusion Culling, seguirá beneficiándose de Frustum Culling.

1.2. Occlusion Area

Para aplicar occlusion culling a objetos en movimiento se tiene que crear un Occlusion Area y luego modificar su tamaño para que se ajuste al espacio donde van a estar ubicados los objetos en movimientos , objetos que no son estaticos. Esto pueden ser creados con Occlusion Areas agregando el componente Occlusion Area a un Game Object vacío (Component -> Rendering -> Occlusion Area in the menus).

Después de crear una Occlusion Area, se debe marcar la casilla de verificación Is View Volume para ocultar objetos en movimiento.

Características del Occlusion Area

- **Size** Define el tamaño del Occlusion Area.
- **Center** Coloca el centro del Occlusion Area.
- **Is View Volume** Define en dónde puede estar la cámara. Se debe Verificar esto para ocluir objetos estáticos que están dentro de este Occlusion Area.

Múltiples áreas pequeñas pueden ser más precisas en comparación con una sola área más grande, pero puede aumentar el tiempo de baked y el tamaño del archivo de datos.

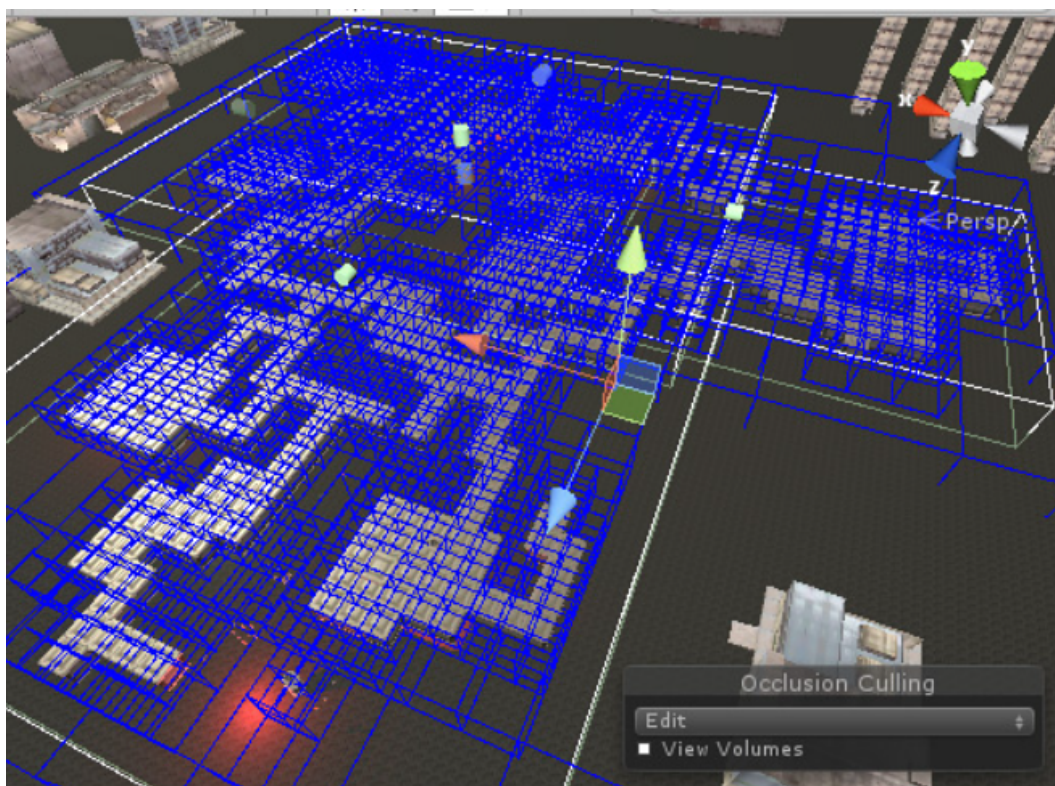


Figura 1: Occlusion Area

1.3. Occlusion culling

A diferencia del frustum culling, que es un proceso automatizado, el occlusion culling debe ser configurado y hacer bake de manera manual en nuestro proyecto.

Para este proceso todos los Game objects con las etiquetas Occludeer Static y Occludee Static se incluyen en el proceso de baked. Si la cámara sale de la Occlusion Area, no se aplicará ninguna oclusión, lo que provocará errores como objetos invisibles y agujeros en el terreno.

1.3.1. Cuando usar occlusion Culling

Para determinar si es probable que el occlusion culling mejore el rendimiento del tiempo de ejecución de su proyecto, considere lo siguiente:

- La prevención de las operaciones de renderizado desperdiciadas puede ahorrar tiempo de CPU y GPU. Ya que el occlusion culling integrada de Unity realiza cálculos de tiempo de ejecución en la CPU, lo que puede compensar el tiempo de CPU que ahorra. Por lo tanto, es más probable que el occlusion culling genere mejoras en el rendimiento cuando un proyecto está vinculado a la GPU debido a un sobregiro.
- Unity carga datos del occlusion culling en la memoria en tiempo de ejecución.

- El oclusion culling funciona mejor en escenas en las que áreas pequeñas y bien definidas están claramente separadas entre sí por GameObjects sólidos. Un ejemplo común son las habitaciones conectadas por pasillos.
- Puede usar el oclusion culling para ocluir GameObjects dinámicos, pero GameObjects dinámicos no puede ocluir otros GameObjects. Si un proyecto genera geometría de escena en tiempo de ejecución, entonces el oclusion culling integrada de Unity no es adecuada para su proyecto.

1.3.2. Como funciona el oclusion culling

El oclusion culling genera datos sobre tu escena en el editor de Unity y luego usa esos datos en tiempo de ejecución para determinar lo que puede ver una cámara. El proceso de generación de datos se conoce como baking.

Cuando haces baking a los datos de oclusion culling, Unity divide la escena en celdas y genera datos que describen la geometría dentro de las celdas y la visibilidad entre celdas adyacentes. Luego, Unity fusiona las celdas donde sea posible, para reducir el tamaño de los datos generados. Para configurar el proceso de baking, puede cambiar los parámetros en la ventana Eliminación de oclusión y usar áreas de oclusión en su escena.

En el tiempo de ejecución, Unity carga estos datos bakeados en la memoria y, para cada cámara que tenga habilitada su propiedad eliminación de oclusión, realiza consultas en los datos para determinar qué puede ver esa cámara. Cuenta que cuando el oclusion culling esta habilitado, las cámaras realizan tanto la selección de frustum como la selección de oclusión.

1.4. Ejemplos

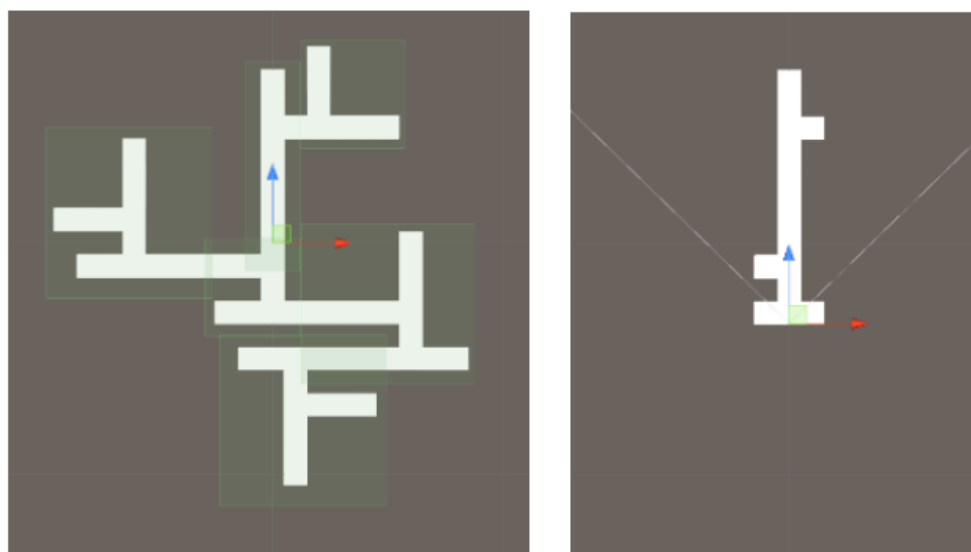


Figura 2: Uso de Occlusion Culling con areas

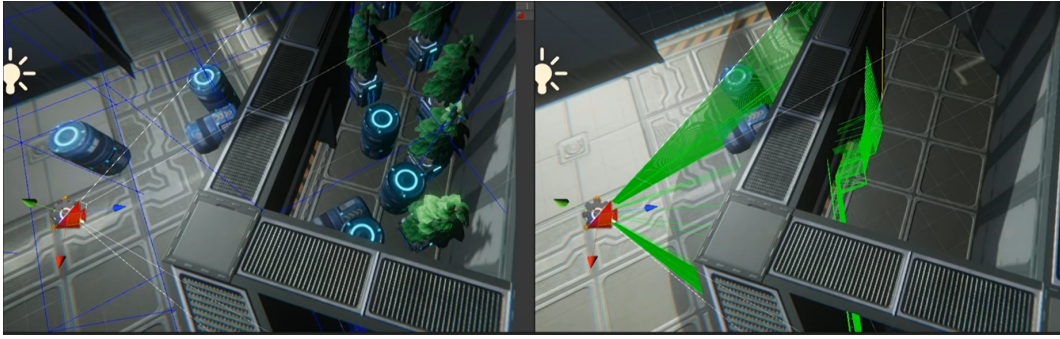


Figura 3: Occlusion Culling OFF - Occlusion Culling ON

Referencias

- <https://docs.unity3d.com/Manual/OcclusionCulling.html>
- <https://docs.unity3d.com/es/2018.4/Manual/OcclusionCulling.html>
- <https://www.youtube.com/watch?v=18KMqxJlhmw>
- <https://www.youtube.com/watch?v=C0qqC650mQY>
- <https://www.youtube.com/watch?v=OmuQmydipGg>