PATH TRACER

INFORMÁTICA GRÁFICA 2017/2018

Escuela de Ingeniería y Arquitectura

Grado en Ingeniería Informática

Alba Clemente Villafranca – 652204

Alberto Milián Barquín - 681048

índice

[**1.-Introducción** 1](#_Toc523672797)

[**2.- Decisiones de implementación** 1](#_Toc523672798)

[**3.- Requerimientos básicos** 1](#_Toc523672799)

[**4.- Requerimientos Opcionales** 1](#_Toc523672800)

[**5.- Conclusión** 1](#_Toc523672801)

# **1.-Introducción**

El presente proyecto consiste en la implementación de un Path Tracer capaz de generar (renderizar) imágenes simulando el transporte y la interacción de la luz con los objetos geométricos de una escena, así como la generación y estudio de dichas imágenes.

# **2.- Decisiones de implementación**

El lenguaje de programación elegido para la implementación del programa ha sido Java, debido al conocimiento que los componentes del equipo poseen del mismo.

Las geometrías que conforman la escena se encuentran implícitas en el código, lo cual quiere decir que para generar diferentes escenas se requiere una modificación del código directamente, no obstante, la sección de código que realiza esta función se encuentra bien señalizada mediante comentarios.

Las imágenes generadas son de formato png, se ha seleccionado este formato debido a que da una buena calidad de imagen en este caso y es compatible con la mayoría de plataformas y sistemas operativos.

Para la ejecución del Path Tracer, la línea de comendo debe seguir el siguiente esquema:

|  |
| --- |
| java -jar pathTracer.jar <antialiasing> <anchura> <altura> <nombreDeLaImagen> |

Tal que:

* <antialiasing>: es el número de rayos (o caminos, paths) que se lanzan por cada píxel de la imagen.
* <anchura> y <altura>: la anchura y altura en píxeles que tendrá la imagen generada.
* <nombreDeLaImagen>: El nombre de la imagen, sin añadir la extensión (.png), ya que eso lo hará el programa.

Un ejemplo de línea de comandos para una ejecución del programa sería:

|  |
| --- |
| java -jar pathTracer.jar <100> <1080> <1080> <scene1> |

# **3.- Creación de la escena**

La escena esta compuesta de una serie de objetos geométricos, una cámara, los píxeles que componen la imagen y los puntos de luz que iluminan la escena.

La cámara está representada por un punto, que será el origen de los rayos (*paths*), y tres direcciones, perpendiculares entre sí, que conforman la base de coordenadas en base a la cual se generará la imagen.

En cuanto a los objetos geométricos, la mayoría de las escenas tienen lugar en una *Cornell Box*, que consiste en 5 planos infinitos que forman entre sí un habitáculo con uno de los lados abiertos, en el cual está situada la cámara. La pared izquierda es de color rojo, la pared derecha verde, y el resto de paredes (superior, fondo e inferior) grises.

Las escenas pueden tener uno o varios puntos de luz, generalmente situados en el plano superior de la imagen, el techo de la *Cornell Box.*

# **4.- Requerimientos básicos**

Los requerimientos básicos que cumple este Path Tracer son:

* Luz directa: Obtención de la luz directa que se observa en el píxel correspondiente, calculada utilizando la técnica de BRDF de Phong para determinar radiancia (el color).

//IMAGEN LUZ DIRECTA

* Luz indirecta: La implementación de esta iluminación permite generar el efecto denominado *color bleeding*, entre otras. Cuando un rayo impacta contra un objeto y este recibe luz directa, refracta parte de la energía recibida con el color alterado en base al color del objeto y el color del rayo que lo impacta. Si este nuevo rayo de energía llega a impactar con otro objeto puede llegar a iluminar con ese nuevo color resultante dicho objeto, siempre y cuando la distancia no sea suficientemente grande como para que la energía del rayo sea insignificante.

//IMAGEN CON LUZ INDIRECTA

//IMAGEN SOLO CON INDIRECTA

* Varios puntos de luz: La mayoría de las escenas han sido renderizadas con dos puntos de luz, generados aleatoriamente cada uno dentro de un triángulo. Ambos triángulos están dispuestos en el plano superior de la escena y formando un rectángulo, de tal forma que simule ser una superficie rectangular luminosa.
* Next Event Estimation:

At each interaction of each path, the path tracer will estimate direct lighting from all

the point light sources by casting shadow rays and accounting for the inverse square law.

Notice that the number of light sources will affect rendering time.

In order to compare the effect of next event estimation, another test scene could be a

Cornell Box in which the light source is a point light source in the upper half of the box

(instead of a whole planar primitive).

If all these required features (path tracer + next event estimation with point lights) are

accomplished correctly, the students will get a mark of 7. For higher marks, the students

should do any of the optional extensions.

# **5.- Requerimientos Opcionales**

# **6.- Conclusión**