

Práctica de creación de un trazador de rayos con iluminación

Iñigo Aguas Ardaiz

Computación Gráfica

Curso 2015-2016

Máster Universitario en Ingeniería Informática

Universidad Pública de Navarra - Nafarroako Unibertsitate Publikoa

29 de enero de 2016

Este documento pretende ser un resumen del trabajo que se presenta intentando mostrar cómo se ha llevado a cabo su implementación así como la formulación de trabajo que ha tenido.

1. Corrección de la entrega anterior

Para poder comenzar con esta segunda entrega, fue necesario en primer lugar corregir todos los errores de la anterior, a fin de tener un trazador de rayos funcional.

En primer lugar, la cámara no creaba de forma adecuada la base ortonormal por culpa de no hacerlo en relación al eje negativo de la Z (opuesto del vector *look*) así como de no normalizarlo. Es por esta razón, como no se guarda el vector *look* sino que se genera en el método `getLook()`, su generación también estaba incompleta sin hacer el opuesto. Se ve un fallo similar en las variables `s` y `oplook` usadas para crear la matriz de transformación.

Por otra parte, los triángulos también contenían un fallo en su cálculo. Se corrige aplicando el método del producto mixto en vez de la resolución de cramer directamente. Además, se implementa la intersección con los planos de forma más ligera. Queda un error en la clase `Perspective` ya que se usa el ángulo en grados en vez de en radianes, al olvidar el `this` necesario para hacer referencia a la variable correcta. Además, se ha implementado la proyección ojo de pez semiesférica.

Por lo demás, existen cambios pequeños de estilo, permisos de acceso y documentación que no son reseñables para el correcto funcionamiento del trazador, además de la ade-

cuación del parser para su correcta integración y funcionamiento (realmente pequeños cambios en los archivos de escenas para que el parser lo reconozca correctamente). Todo esto se ha resumido en 18.6 horas de trabajo. Un total de 39.2 para la implementación del trazador básico si tomamos las horas del trabajo anterior.

2. Planteamiento del trabajo

Se comenzó la implementación del trabajo sin tener muy claro cómo llevarlo a cabo. En particular, sin tener claro un diseño de clases y cómo estas interactuaban con otras. Este tema fue solucionado por parte del profesor al presentarnos una propuesta de diseño que se adaptaba a nuestras necesidades. Además, el uso del software 3D Studio creo que me dio un alto nivel de comprensión en la materia.

Por otra parte, uno de los mayores problemas que tuve fue el desarrollar un método de utilización de los rayos de sombra por parte de las luces. Esto se debió a dos motivos. Por un lado, como se sabe, el rayo de sombra dispone del punto P de intersección con el objeto en su trayectoria. Sabiendo esto, lo que se hace en el método de intersección más ligero es devolver un booleano sabiendo que será verdadero si se encuentra un objeto diferente al que estamos tratando (o el mismo objeto pero intersectando en otro punto). Debido a esto, se necesita disponer de una comparación entre la intersección y P que sea tolerante a un error, que finalmente se definió en 10^{-5} para hacer de forma correcta también la reflexión. Añadido a todo esto, se debe tener en cuenta que al principio no se tenía en cuenta si la intersección del rayo de sombra se encontraba entre el punto de origen del rayo y el de intersección lo que hizo que se detectaran sombras falsas. En relación a las luces, también, se debió calcular las zonas no iluminadas de las fuentes *spot* y direccionales de sección circular, utilizando en las primeras un factor de atenuación.

Con respecto a los materiales, el mayor problema ha estado en el cálculo del color. En particular, las sumas deben hacer que los colores saturen en 255. Esto hace que sea más difícil de controlar en algunos momentos, y como se calcula con multiplicaciones con irradiancia, si alguna de las componentes era 0, esto hace que no se pueda saturar a blanco (255, 255, 255), sino a la primitiva del color saturado. Probablemente la mejor solución a esto sea manejar los colores en el rango $[0, 1]$, lo que al ser un espacio normal hace que todos los cálculos con sus términos “caigan” siempre dentro del mismo espacio.

Además, la reflexión no calcula de forma correcta el vector del rayo secundario. Concretamente, se lanza en sentido contrario, lo que se soluciona invirtiendo el vector. Además, la reflexión para mi gusto no se hace de forma correcta, al no conseguir que la adición de los colores reflejen la realidad de los colores de los objetos reflejados.

Por último, el centrarse en implementar todo lo anterior ha hecho que la reforma necesaria en el parser para poder utilizar todo lo implementado no haya sido posible de llevar a cabo en su completitud. En particular, se utiliza de forma predeterminada la función de Phong con una reflectancia de Blinn con exponente de lustre de 5. El color de los objetos se mantiene. Para cambiar esto se debe acceder al parser y hacer que genere otro material al parsear un objeto. La función de reflectancia no se puede cambiar en el constructor de la clase **Material**, al igual que el hecho de ser especular o no, que está

puesto a verdadero por defecto. Todo esto es sin duda un punto de mejora, incluso con una interfaz gráfica que permita cambiarlo en directo, aunque debemos de tener en cuenta que todo lo parseable ya se puede cambiar de forma dinámica con la implementación que se ha hecho a través de la apertura de escenas. Además, se ha preparado la posibilidad de guardar los resultados en formato png.

3. Distribución temporal del trabajo y reflexión personal

Para hacer el trabajo se ha utilizado un sistema de gestión de proyectos, Redmine, tal y como se presentó en la entrega anterior. Se puede encontrar, por tanto, que se han utilizado las 29.4 horas en el siguiente enlace: <http://redmine.martinarroyo.net/projects/upnatray>. Además, se puede encontrar todo el código versionado en el siguiente enlace de github: <https://github.com/iaguas/upnaTRay>. Además, se puede encontrar toda la documentación del proyecto en: <http://iaguas.github.io/upnaTRay/apidocs/>.

Así como el documento que se entregó anteriormente hacía una reflexión un tanto negativa sobre el trabajo hecho, en este caso estoy más contento con el trabajo realizado y con el tiempo invertido. He de decir que el hacer este trabajo me ha hecho comprender mejor el fundamento teórico que desarrolla la asignatura y que el resultado obtenido, si bien dispone de algo de margen de mejora, me satisface como conocimiento de una asignatura de gráfica. También debería decir, en honor a la verdad, que bajando la implicación que he tenido con asuntos extra-académicas se ha podido obtener mejores resultados.