

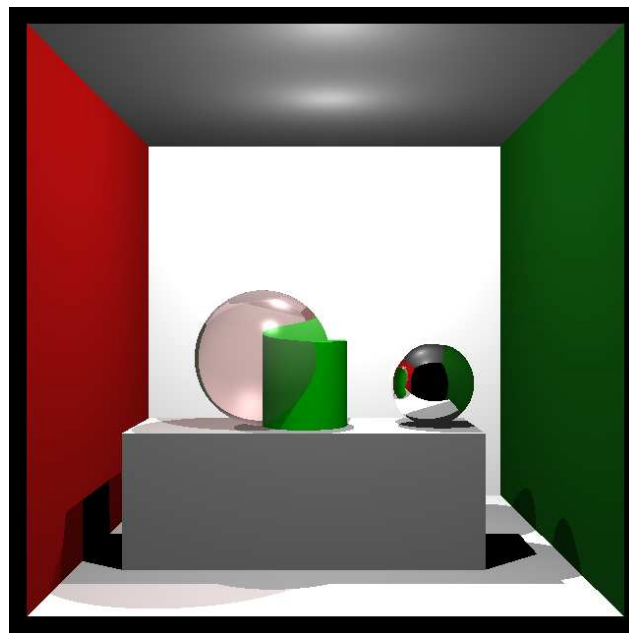
Segundo obligatorio

Introducción a la Computación Gráfica

(Junio 2014)

Se deberá implementar el algoritmo *Whitted ray tracing* [5] que permita generar un **archivo de imagen** similar a la mostrada abajo. Dicha imagen contiene:

- dos fuentes luminosas puntuales;
- una superficie espejada (la esfera de la derecha);
- superficies con componentes de luz ambiente, difusa y Phong (el cilindro del medio);
- un objeto transparente con refracción, levemente coloreado (la esfera de la izquierda);
- dos objetos inter-penetrantes (la esfera y el cilindro);
- mallas poligonales (el prisma que oficia de mesa).



Generar una imagen similar no significa una imagen igual. Pueden variar los aspectos relativos a las dimensiones, ubicación y forma de los objetos, la refracción así como detalles de las sombras. También se pueden agregar objetos o cambiar la ubicación de los mismos, aunque se exhorta contemplar las intersecciones entre objetos, con al menos uno de ellos transparente. Pero es obligatorio que se incluyan objetos del tipo esfera, cilindro, plano y malla de triángulos (por ejemplo en la mesa).

Actividades obligatorias:

La aplicación a desarrollar debe incluir:

- el manejo de múltiples fuentes de luz;
- superficies espejadas, transparentes y superficies con reflexión ambiente, difusa y Phong;
- objetos geométricos simples (esfera, cilindro, plano, triángulo);
- intersección entre objetos simples;
- ubicación de la cámara ajustable por el usuario;
- configuración por el usuario de la resolución de la imagen a generar;
- configuración de la escena mediante archivo XML (definición de las luces, de los objetos simples, de la posición y parámetros de cámara, y de la resolución de la imagen);
- generación de dos imágenes auxiliares en blanco y negro que permitan representar (por separado) los coeficientes de reflexión y transmisión (refracción) de cada objeto de la escena. Un objeto totalmente

opaco se corresponde con el color negro mientras que un objeto totalmente transparente se corresponde con el color blanco. Análogamente para la reflexión del objeto.

No es necesario que la aplicación disponga interfaz gráfica (ventanas, botones, etc.).

Para la nota serán tenidos en cuenta la optimización en la velocidad de rendering, los aspectos de creatividad gráfica, así como las cuestiones opcionales que desarrollen.

Actividades opcionales:

- Técnicas de aceleración vistas (volúmenes acotantes, partición espacial, entre otros).
- Mapeo de texturas sobre los objetos geométricos simples. Para el cálculo de la coordenada de textura asociada a cada punto de un triángulo se sugiere utilizar coordenadas baricéntricas [1].
- Utilizar alguna de las técnicas de anti-aliasing.
- Representación de sólidos mediante geometría sólida constructiva (CSG) [4].
- Uso de SDL para mostrar en tiempo real los resultados parciales del algoritmo *Whitted ray tracing*.
- Usar una textura auxiliar para modificar, en cada punto, la normal del objeto (normal mapping).
- Aplicación de transformaciones geométricas a los objetos.
- Visualizador y/o editor de la escena utilizando OpenGL.
- Generación de imágenes auxiliares que expongan de forma aislada los componentes ambiente, difuso, especular, reflexión y transmisión, correspondientes al primer nivel de recursión del algoritmo.

Resultados esperados:

- Generar un historial representativo de imágenes generadas, que expongan la evolución del obligatorio.
 - Se sugiere guardar las imágenes generadas en directorios independientes para facilitar la creación del historial. Por ejemplo, la aplicación puede crear un directorio cuyo nombre sea la fecha y hora de la ejecución.
- Informe en papel.
- CD/DVD con informe en formato digital, ejecutable, código de la aplicación, escenas de ejemplo, imágenes generadas (no menos de tres) con las imágenes auxiliares correspondientes y el historial mencionado.
- Las imágenes generadas deben ser almacenadas en algún formato estándar sin pérdida de información (png, bmp, etc.).
- En el informe debe explicitarse el funcionamiento del algoritmo, las decisiones tomadas en su implementación, bibliotecas utilizadas, etc.
- Se puede usar código externo de ayuda (por ejemplo para manejo matricial), pero se pide la implementación del renderer por completo.

Para salvar las imágenes generadas, pueden utilizar FreeImage [2], SDL_image [3] o alguna otra biblioteca similar.

Presentación del obligatorio y aclaración de dudas de letra: Lunes 9 de junio de 2014.

Entrega: Lunes 30 de junio de 2014.

[1] <http://www2.uah.es/pramos/docencia/MIG/Texturas/texturas-print.pdf>

[2] <http://freeimage.sourceforge.net/download.html>

[3] http://www.libsdl.org/projects/SDL_image/

[4] <http://www.cse.ohio-state.edu/~parent/classes/681/Lectures/19.RayTracingCSG.pdf>

[5] Andrew S. Glassner, "An Introduction to Ray Tracing", Morgan Kaufmann, 1989, ISBN 0122861604