

# Tutorial

## Introducción a los algoritmos de *Ray Tracing* y *Path Tracing* para la generación de imágenes sintéticas

Miguel A. Astor

Universidad Central de Venezuela, Facultad de Ciencias  
Escuela de Computación, Centro CICORE, Laboratorio ICARO  
Urb. Valle Abajo, San Pedro, Caracas, Venezuela  
Email: miguel.a.astor@ucv.ve

**Resumen**—En este tutorial se presentarán las nociones fundamentales para la implementación del algoritmo de *Ray Tracing* de Whitted como algoritmo para la generación de imágenes sintéticas. Este es un algoritmo clásico del campo de la Computación Gráfica que permite la generación de imágenes que incluyen múltiples efectos visuales como la iluminación y sombreado de superficies, reflexión y refracción. Adicionalmente, se presentará la extensión del algoritmo de *Ray Tracing* conocida como algoritmo de *Path Tracing*, utilizado para generar imágenes con efectos de iluminación global tales como cáusticas e iluminación difusa indirecta. El enfoque del tutorial es teórico-práctico. Durante el desarrollo del mismo se presentarán los conceptos teóricos necesarios para el planteamiento y funcionamiento de los algoritmos, los cuales podrán ser practicados por los participantes con la construcción de un *ray tracer* básico en el lenguaje C++, utilizando plantillas de código provistas por el instructor.

**Palabras clave**—*Ray Tracing*, *Path Tracing*, generación de imágenes sintéticas, Computación Gráfica, iluminación global, C++.

### I. INTRODUCCIÓN

La generación de imágenes sintéticas o síntesis de imágenes por computadora consiste en el uso de computadoras para producir imágenes, actividad que hace uso de múltiples técnicas y algoritmos para obtener resultados que pueden variar ampliamente en cuanto la calidad de las imágenes generadas y los efectos visuales que se pueden generar [1]. Dependiendo de los algoritmos y técnicas utilizadas, las imágenes generadas pueden poseer un alto nivel de realismo visual, o pueden poseer distintos niveles de estilización. Es precisamente en el campo de la síntesis de imágenes realistas donde tiene aplicación el algoritmo de *Ray Tracing* y sus muchos derivados [2]. *Ray Tracing* como un algoritmo de generación de imágenes sintéticas fue introducido por Turner Whitted en 1979 [3].

El algoritmo de *Ray Tracing* de Whitted se basa en el algoritmo de *Ray Casting* de Appel [4], el cual consiste en “lanzar” rayos (segmentos de línea dirigidos) desde el punto de visión hacia la escena, detectando intersecciones entre los rayos y la escena para realizar el ordenamiento de las superficies visibles y cálculos de iluminación y sombreado. Por su parte, el algoritmo de *Ray Tracing* realiza un trazado recursivo de los rayos para calcular efectos visuales como la reflexión y refracción de la luz, de ahí su nombre [3]. El algoritmo de *Ray Tracing* se caracteriza por ser capaz de

producir efectos visuales de alta complejidad y realismo y por su relativa sencillez de implementación [1].

Existen múltiples extensiones y variantes al algoritmo de *Ray Tracing*, una de las cuales es el algoritmo de *Path Tracing* o *Ray Tracing* de Monte Carlo, introducido originalmente por James Kajiya en [5], el cual utiliza técnicas de muestreo estadístico basadas en métodos de Monte Carlo para simular el transporte de luz entre distintas superficies.

Las técnicas de generación de imágenes sintéticas basadas en el algoritmo de *Ray Tracing* son ampliamente utilizadas para la generación de imágenes de alto nivel de realismo visual en campos como el cine de efectos especiales, la animación por computador, la visualización arquitectónica y la visualización de conjuntos de datos volumétricos [1], ya sea utilizando el algoritmo básico o variantes del mismo como lo son el *Path Tracing* [5], el *Photon Mapping* [6], el *Ray Marching* [2] y el *Volume Ray Casting* [7].

El presente tutorial pretende introducir a los participantes en el campo de la generación de imágenes sintéticas con los algoritmos de *Ray Tracing* y *Path Tracing*, practicando los conceptos fundamentales de trazado de rayos, intersección de rayos con figuras geométricas, iluminación y sombreado y muestreo aleatorio, utilizando el lenguaje C++.

### II. OBJETIVOS

A continuación se presentan los objetivos general y específicos del Tutorial planteado.

#### II-A. Objetivo general

Introducir y practicar los fundamentos de generación de imágenes sintéticas con los algoritmos de *Ray Tracing* y *Path Tracing*.

#### II-B. Objetivos específicos

- Presentar el algoritmo de *Ray Tracing* de Whitted.
- Practicar la implementación de técnicas de intersección rayo-objeto.
- Presentar los fundamentos de iluminación y sombreado con funciones BRDF.
- Presentar el algoritmo de *Path Tracing*.

- Practicar la implementación de un método de muestreo aleatorio de Monte Carlo.

### III. CONTENIDOS

Durante el Taller se desarrollará el siguiente contenido temático:

#### III-A. Introducción a la generación de imágenes sintéticas

Fundamentos de generación de imágenes sintéticas. Comparación de varios algoritmos de generación de imágenes sintéticas: *Z-Buffer*, *Scanline* y *Ray Tracing*.

#### III-B. Introducción al Algoritmo de Ray Tracing

Definición de rayo. El algoritmo de *Ray Tracing* de Whitted. Generación de los rayos primarios. *Aliasing* y técnicas de *Supersampling*.

#### III-C. Intersección de Rayos con Figuras Geométricas

Enfoques de intersección de rayos: geométrico y algebraico. Intersecciones: rayo-esfera, rayo-plano.

#### III-D. Iluminación y sombreado

El concepto de *Bidirectional Reflectance Distribution Function* (BRDF). La BRDF de Phong.

#### III-E. Introducción al algoritmo de Path Tracing

El *Path Tracing*. Muestreo aleatorio de Monte Carlo. Generación de muestras sobre una semi-esfera. Iluminación global.

### IV. METODOLOGÍA

La metodología del Taller está basada en una presentación de conocimiento teórico apoyada por una serie de ejercicios prácticos de programación. Para cada tema listado en la Sección III se comenzará con una presentación breve del contenido teórico necesario, seguida por la implementación guiada de los contenidos presentados. Los ejercicios de programación serán realizados con plantillas de código provistas por el instructor, de forma que los participantes puedan concentrarse en los detalles de implementación de los algoritmos y fórmulas presentadas sin tener que distraerse con detalles foráneos al contenido del tutorial.

### V. DURACIÓN

Se estima una duración de 8 horas para el Taller (un día completo).

### VI. PÚBLICO OBJETIVO

Estudiantes, profesionales y personas interesadas en conocer el funcionamiento y detalles de implementación del algoritmo de *Ray Tracing* y derivados.

Los participantes deben tener un dominio básico a intermedio del lenguaje C++ en ambientes Linux, especialmente en lo referente a las capacidades de programación orientada a objetos del lenguaje.

### VII. REQUERIMIENTOS

Para poder llevar a cabo el tutorial se requiere lo siguiente:

- Un proyector.
- Acceso a Internet en la sala.
- Computadoras donde puedan trabajar hasta dos personas por computadora, las cuales deben poseer lo siguiente:
  - Sistema operativo basado en Linux.
  - El compilador g++.
  - Editor de texto con resaltado de sintaxis.
  - Las bibliotecas GLM y JSON-Spirit y dependencias (paquetes `libglm-dev` y `libjson-spirit-dev` en sistemas basados en Debian).

### VIII. IDIOMA

La exposición, dictado, ejercicios y ejemplos serán dados en español.

### REFERENCIAS

- [1] J. D. Foley, A. Van Dam, S. K. Feiner, J. F. Hughes, y R. L. Phillips, *Introducción a la Graficación por Computador*. Addison-Wesley, 1996.
- [2] H. W. Jensen, *Realistic image synthesis using photon mapping*. Ak Peters Natick, 2001, vol. 364.
- [3] T. Whitted, "An improved illumination model for shaded display," in *ACM SIGGRAPH Computer Graphics*, vol. 13, no. 2. ACM, 1979, p. 14.
- [4] A. Appel, "Some techniques for shading machine renderings of solids," in *Proceedings of the April 30-May 2, 1968, spring joint computer conference*. ACM, 1968, pp. 37-45.
- [5] J. T. Kajiya, "The rendering equation," in *ACM Siggraph Computer Graphics*, vol. 20, no. 4. ACM, 1986, pp. 143-150.
- [6] H. W. Jensen, "Global illumination using photon maps," in *Rendering Techniques' 96*. Springer, 1996, pp. 21-30.
- [7] B. Preim y D. Bartz, "Visualization in medicine," *Theory, Algorithms, and Applications*. Amsterdam: Morgan Kaufmann, 2007.

PRESENTADOR

*Miguel Angel Astor Romero*

Profesor investigador, categoría Instructor, de la Escuela de Computación, de la Universidad Central de Venezuela (UCV). Es Licenciado en Computación de la Universidad Central de Venezuela (2014). Actualmente cursa estudios de Maestría en Ciencias de la Computación en la UCV bajo la tutoría del Dr. Wílmer Pereira Gonzales.

Dirección: Paseo los Ilustres, Urb. Valle Abajo. Facultad de Ciencias, Cdad. Universitaria de Caracas, Venezuela.

Tlf: +58-212-6050984. Correo: miguel.a.astor@ucv.ve