

Trazado de rayos estocástico en la GPU

Joaquim Romo

Treball Final de Grau / 2014

THESIS ADVISOR
Javier Agenjo Departament GTI



Resumen

El presente trabajo tiene como objetivo el estudio y la implementación de un algoritmo de renderizado con raytracing estocástico en la unidad de procesamiento de gráficos (GPU). Se ha elegido realizar la implementación en una arquitectura de este tipo debido a las ventajas que ofrece en cuanto a tiempo de ejecución, gracias a la gran capacidad de cómputo en paralelo que ofrecen las arquitecturas de GPU actuales.

La primera parte del trabajo se dedica al estudio teórico del algoritmo de pathtracing, se comentan algunos conceptos físicos básicos relacionados con el transporte de luz y su interacción con los materiales así como las ecuaciones matemáticas y las técnicas estadísticas necesarias para la comprensión y correcta implementación del algoritmo.

En una segunda parte se discuten las tecnologías involucradas, concretamente el uso que se hace de la arquitectura CUDA, la librería OptiX y su funcionamiento y la implementación del algoritmo que se ha realizado sobre estas.

Índice

Index of figures	VII
Index of tables	IX
1. INTRODUCCIÓN	1
1.1. Contexto	1
1.2. Algoritmos de iluminación global	3
1.2.1. Radiosity (Goral et al. 1984)	3
2. FUNDAMENTOS TEÓRICOS	5
2.1. Unidades Radiométricas	5
2.1.1. Flujo	5
2.1.2. Irradiancia	5
2.1.3. Angulo solido	5
2.1.4. Radiancia	5
2.2. BRDF	6
2.3. Ecuación de renderizado	7
2.4. El metodo de montecarlo	8
3. RESEARCH OBJECTIVES	9
3.1. A Section	9
3.1.1. A Subsection	9
3.2. Another Section	9
4. RESEARCH PLAN AND METHODOLOGY	11
5. SUMMARY OF PRIOR WORK (IF ANY)	13
6. SIGNIFICANCE	15

Índice de figuras

3.1. Exemple de grafic	10
----------------------------------	----

Índice de cuadros

3.1. Prova de taula	9
-------------------------------	---

Capítulo 1

INTRODUCCIÓN

1.1. Contexto

En el entorno de la imagen generada por computador siempre ha sido un reto tratar de generar imágenes lo más realistas posibles. Para ello un gran número de investigadores se han dedicado a diseñar algoritmos que simulan o imiten el comportamiento y la interacción de la luz con los materiales. Estos algoritmos que tratan de simular de forma realista el comportamiento de la luz son generalmente conocidos como algoritmos de iluminación global.

Estos algoritmos, por lo general, suelen tener una complejidad computacional muy elevada y el tiempo de cómputo necesario para obtener un resultado satisfactorio en escenas complejas era un factor limitador en su aplicación práctica. Por ello las aplicaciones que hacen uso de gráficos 3D en tiempo real típicamente se centran en la iluminación local o directa de los objetos de la escena y simulan la iluminación indirecta mediante técnicas que aun sin tener un fundamento físico ofrecen una mayor credibilidad para el ojo humano. Estas técnicas suelen ser algoritmos de postprocesado que se aplican en espacio de pantalla, por ejemplo “ambient occlusion” o “directional occlusion”.

Sin embargo en los últimos años se han realizado grandes avances en las arquitecturas de las unidades de procesamiento de gráficos (GPUs), en especial la gran capacidad de cómputo en paralelo debido al elevado número de microprocesadores que forman estos dispositivos. Con tal de aprovechar estos avances en el hardware, los fabricantes de GPU han desarrollado librerías de computo generico (OpenCL, CUDA) que ofrecen gran libertad al programador para implementar sus propios algoritmos.

Estas mejoras han permitido realizar implementaciones de algoritmos de iluminación global en las GPUs que son mucho más rápidos que las implementaciones típicas en la CPU permitiendo reducir el tiempo de cómputo de varias horas o

días a minutos e incluso a ratios interactivos dependiendo de la GPU y algoritmos utilizados.

1.2. Algoritmos de iluminación global

Se conoce como algoritmos de iluminación global aquellos que tratan de simular distintos aspectos del comportamiento de la luz en su interacción con los objetos de una escena tridimensional. Algunos de ellos están pensados y optimizados para fenómenos concretos mientras que otros tratan de recrear fielmente todos los aspectos del transporte de luz.

En esta sección revisaremos por encima algunos de los algoritmos clásicos.

1.2.1. Radiosity (Goral et al. 1984)

Radiosity fue el primero de los algoritmos de iluminación global que se desarrollaron. Inicialmente el algoritmo fue desarrollado en los años 1950 para aplicarlo al problema de la transferencia de calor. En 1984 fue modificado y adaptado por Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg y Bennett Battaille, investigadores de la universidad de Cornell para su aplicación en la generación de imagen sintética.

Este algoritmo trata de resolver el problema de la iluminación indirecta entre superficies puramente difusas o Lambertianas.

Capítulo 2

FUNDAMENTOS TEÓRICOS

2.1. Unidades Radiométricas

2.1.1. Flujo

2.1.2. Irradiancia

2.1.3. Angulo solido

2.1.4. Radiancia

2.2. BRDF

La función de distribución de reflectancia bidireccional (de ahora en adelante BRDF, por sus siglas en inglés), definida por primera vez por Nicodemus (1965), es un función que define la respuesta a la luz de una superficie opaca, tomando como parámetros dos vectores unitarios que definen las direcciones de entrada y salida de la luz. Más formalmente, la BRDF mide la relación entre la radiancia diferencial reflejada en la dirección de salida y la irradiancia diferencial entrante en el ángulo sólido diferencial alrededor del vector de entrada

$$f(x, l, v) = \frac{dL(x \rightarrow v)}{dE(x \leftarrow l)} \quad (2.1)$$

donde l es el vector unitario que apunta en la dirección opuesta a la de entrada de la luz y v es el vector unitario que apunta en la dirección de salida de la luz.

La BRDF solo esta definida para vectores l y v tales que $n \cdot v > 0, n \cdot l > 0$, siendo n la normal de la superficie.

2.3. Ecuación de renderizado

James T. Kajiya (1986) propuso una ecuación integral que unifica y formaliza los distintos algoritmos de renderizado, ya que hasta ese momento no existía un marco de trabajo teórico común.

Esta ecuación se puede encontrar en muchas formas distintas según el autor que la cite pero en la forma propuesta por el autor original es la siguiente:

Donde $I(x, x')$ es la intensidad de la luz que llega del punto x' al punto x , (x, x') es la intensidad de la luz emitida en el punto x' hacia el punto x . (x, x', x'') es una función de distribución que determina qué proporción de la luz incidente en x' proveniente de x'' es rebotada hacia x . Esta función depende de las características de cada material y es comúnmente conocida como BRDF de sus siglas en inglés Bidirectional Reflectance Distribution Function.

El dominio de la integral, S , es la unión de todas las superficies de la escena $S = \bigcup g(x, x')$ es un término geométrico que determina la visibilidad entre los puntos x, x' .

A primera vista esta ecuación puede parecer imponente pero si nos olvidamos por un momento de los formalismos matemáticos y nos centramos en su significado, lo que viene a decir es que la luz que llega al punto x en la dirección $x'x$ es igual a la luz emitida en el punto x' en la dirección $x'x$ más la integral de toda la luz que llega al punto x' y es dispersada en la dirección $x'x$.

Lo significativo de esta ecuación es que resulta muy intuitivo derivar algoritmos de renderizado de la misma: se evalúa para cada punto a pintar y se evalúa $I(x', x'')$ recursivamente hasta que se cumpla determinada condición.

2.4. El metodo de montecarlo

Capítulo 3

RESEARCH OBJECTIVES

UbiCALab.

3.1. A Section

UbiCALab.

0	0
0	0

Cuadro 3.1: Prova de taula

3.1.1. A Subsection

UbiCALab.

3.2. Another Section

UbiCALab..

Incorpora una figura amb el logo de la UPF (Figura 3.1) perquè surti a l'índex de figures.



Figura 3.1: Exemple de grafic

Capítulo 4

RESEARCH PLAN AND METHODOLOGY

Lorem ipsum dicam tation maiorum vel ea, an nam minimum pertinax vituperatoribus, mea no illud tibiue nostrum. Odio aperiri dolorem in sit, utinam labore adipiscing cu vel. Nam porro choro cu, nam virtute maluisset voluptatum in, per te vero iuaret praesent. Eos ut volumus pertinacia interesset, tota debitis abhorreant mei et, clita electram pri at. Ea fabulas admodum theophrastus sea, id nostro omnium periculis sed. Vim id nulla aliquando. Id mel habeo nostrud mandamus, laudem legimus fastidii quo ea, ut iriure utamur quo.

Cu sit nobis omnesque scripserit, nec albucius perfecto salutatus ex. Novum debet propriae vis cu, sed ea suscipit accusata, nam minim salutandi voluptatum ut. Quaerendum definitiones est ut, no quo postea prompta scaevola. Nec at choro soleat causae, per ut docendi denique, discere veritus no sed. Ei sed sale delicatissimi mediocritatem, eligendi deterruisset ut mei, ignota offendit occurreret at mel. Numquam blandit gubergren an pro, mei cu quot dolore recteque. Suscipit aliquyam principes eam ut, id error dicit perpetua qui.

Eam mentitum noluisse instructor cu, ad scripta nonummy aliquyam eos. Augue possit gloriatur mel an, mea et dolorum suavitate assueverit. At ius dolores persecuti moderatius, id qui justo tantas dissentiet. Integre invenire imperdiet ut mea, harum alienum officiis te nam, integre qualisque conceptam in vis. Viderer consetetur vim id, sumo tempor complectitur vim an. His te aeterno tamquam integre, nec agam aliquando repudiandae id, te fabellas verterem sea.

Per dicit veniam delicata ut, dicunt corpora mei eu. His no alienum accusata accommodare, eu nam clita audire invidunt. Mutat debitis mediocrem duo te, ex vis prima labore inermis. Usu eu dictas possim vocent. Suas meliore ancillae ad eum, eam ei cibo viderer similique.

Usu ne apeirian deserunt expetendis, sed in iisque abhorreant definiebas, quo nulla inermis appareat cu. Ei nam possim docendi pertinax, ut duis veritus detracto

pri. Animal percipit ius ne, ne iisque quaeque philosophia pro, ut sed nobis nullam adolescens. Persius principes omittantur per an. Mel quot nostrud labores at, ad vis latine persius petentium. Cum partiendo neglegentur mediocritatem ut, per simul aliquando tincidunt no, falli nominavi molestiae ex nam.

Saperet inciderint in vim, vis quod civibus accusata in. Et ullum facilisi accommodare nec, decore utroque definitionem an vel, id duo kasd facer latine. Ea sumo ullamcorper per, ex pro nobis efficiendi concludaturque, vix eu verear lupatum occurreret. Occurreret intellegat te eum, ius ea cibo epicurei dignissim. Diam elit quaeque cum ut. Est dicat summo prodesset te, no vel harum saepe sensibus, et ignota altera doming qui.

Sit altera regione feugiat in, vim ad augue mediocritatem, duo reque choro theophrastus ea. Ipsum apeirian voluptatum duo no, nam ut ornatus accumsan percipitur, est no posse suscipiantur vituperatoribus. Errem legendos partiendo est cu, est clita signiferumque cu. Ea nulla scripta signiferumque eam, sea vidit dolores officiis ei, nibh saepe in nam. Ad vim justo dolor reprimique, eum ad labitur dissentias scriptorem, cu tempor mentitum assentior eum. Id vis mucius percipit invidunt.

Vel at ullum intellegam constituam, mea dicta affert perpetua ea. Eripuit blandit iracundia nec ea, tale timeam molestie cum te. Cum ei kasd mazim, per ei delenit ullamcorper, ad sale vocibus temporibus usu. An nonummy oporteat phaedrum sed, eu unum deserunt incorrupte mei. His an possim delicata, ei accumsan constituam mea. Ei viris constituam pro, cu inimicus argumentum duo.

Cu regione veritus persequeris vis, tation nusquam percipit in pri, vis no labitur deseruisse. Ius in reque fierent. Quo et erant recusabo honestatis. Iusto accusam liberavisse ex mei, ut cum nobis primis accommodare, viderer vocibus assueverit ut nec. Id odio oporteat instructor pri. Nam sint illud vituperatoribus no, zzril ubique dissentias id nec.

Cu est legimus reformidans. Nam cibo mucius dolores ad, prima posse similique vix te. Mea at nemore mandamus assueverit, sit viris option scaevola cu. At omnis decore nostro pro, aliquyam explicari salutatus ei mea, sea no falli incorrupte. Graeco bonorum lobortis ut sed, ut pro discere legimus, an sea volumus consetetur.

Capítulo 5

SUMMARY OF PRIOR WORK (IF ANY)

Capítulo 6

SIGNIFICANCE

