



UNIVERSIDAD  
DE LA REPUBLICA  
URUGUAY



# Tesis de grado

Iluminación global con superficies especulares

Bruno Sena

Ingeniería en Computación  
Facultad de Ingeniería  
Universidad de la República

Montevideo – Uruguay  
Junio de 2019



UNIVERSIDAD  
DE LA REPUBLICA  
URUGUAY



FACULTAD DE  
INGENIERIA

# Tesis de grado

Iluminación global con superficies especulares

Bruno Sena

Tesis de grado presentada en la Facultad de Ingeniería de la Universidad de la República, como parte de los requisitos necesarios para la obtención del título de grado en Ingeniería en Computación.

Directores:

José Aguerre

Eduardo Fernández

Montevideo – Uruguay

Junio de 2019

Sena, Bruno

Tesis de grado / Bruno Sena. - Montevideo: Universidad de la República, Facultad de Ingeniería, 2019.

VII, 18 p.: il.; 29, 7cm.

Directores:

José Aguerre

Eduardo Fernández

Tesis de Grado – Universidad de la República, Ingeniería en Computación, 2019.

Referencias bibliográficas: p. 16 – 18.

1. iluminación global, 2. radiosidad, 3. reflexión especular. I. Aguerre, José, Fernández, Eduardo, . II. Universidad de la República, Ingeniería en Computación. III. Título.

## INTEGRANTES DEL TRIBUNAL DE DEFENSA DE TESIS

---

NombreTribunal1 ApellidoTribunal1

---

NombreTribunal2 ApellidoTribunal2

---

NombreTribunal3 ApellidoTribunal3

Montevideo – Uruguay  
Junio de 2019



## ABSTRACT

Aquí va el abstact

Keyword:

# Tabla de contenidos

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Motivación y problema . . . . .	1
1.2	Objetivos . . . . .	1
1.3	Resultados esperados . . . . .	1
1.4	Estructura del documento . . . . .	1
<b>2</b>	<b>Estado del arte</b>	<b>2</b>
2.1	Modelos de iluminación . . . . .	2
2.1.1	Iluminación Local . . . . .	3
2.1.2	Iluminación Global . . . . .	4
2.2	Radiosidad . . . . .	4
2.2.1	Radiosidad en superficies lambertianas . . . . .	4
2.3	Métodos de cálculo de la matriz de Factores de Forma . . . . .	6
2.3.1	Rasterización . . . . .	6
2.3.2	Raytracing . . . . .	8
2.4	Cálculo del vector de radiosidades . . . . .	8
<b>3</b>	<b>Alcance y diseño</b>	<b>10</b>
3.1	Alcance . . . . .	10
3.2	Diseño . . . . .	10
<b>4</b>	<b>Implementación</b>	<b>11</b>
4.1	OpenGL . . . . .	11
4.1.1	Cálculo de factores de forma de la componente difusa . . . . .	11
4.1.2	Cálculo de factores de forma de la componente especular . . . . .	11
4.2	Embree . . . . .	11
4.2.1	Cálculo de factores de forma de la componente difusa . . . . .	11
4.2.2	Cálculo de factores de forma de la componente especular . . . . .	11

4.3	OpenGL y Embree . . . . .	11
4.3.1	Cálculo de factores de forma de la componente difusa . .	11
4.3.2	Cálculo de factores de forma de la componente especular	11
<b>5</b>	<b>Experimental</b>	<b>12</b>
5.1	Hardware . . . . .	12
5.2	Escenas . . . . .	12
5.3	Casos de prueba . . . . .	12
<b>6</b>	<b>Conclusiones y trabajo futuro</b>	<b>13</b>
6.1	Conclusiones . . . . .	13
6.2	Trabajo futuro . . . . .	13
	<b>Lista de figuras</b>	<b>14</b>
	<b>Lista de tablas</b>	<b>15</b>
	<b>Apéndices</b>	<b>16</b>
	Referencias bibliográficas . . . . .	18







# Capítulo 1

## Introducción

1.1. Motivación y problema

1.2. Objetivos

1.3. Resultados esperados

1.4. Estructura del documento

# Capítulo 2

## Estado del arte

### 2.1. Modelos de iluminación

El proceso de dibujado de gráficos por computadora comprende la generación automática de imágenes a partir de tres parámetros principales, un observador, superficies geométricas que representan objetos de la realidad y las propiedades físicas de los materiales que los componen.

Trivialmente, esto puede ser reducido al problema de cálculo del valor de intensidad lumínica observada en un punto  $x$  y proveniente de otro punto  $x'$ . Matemáticamente, este problema fue planteado por Kajiya en 1986, comúnmente denominado «la ecuación del rendering»:

$$I(x, x') = g(x, x') \left[ \epsilon(x, x') + \int_S \rho(x, x', x'') I(x', x'') \delta x'' \right] \quad (2.1)$$

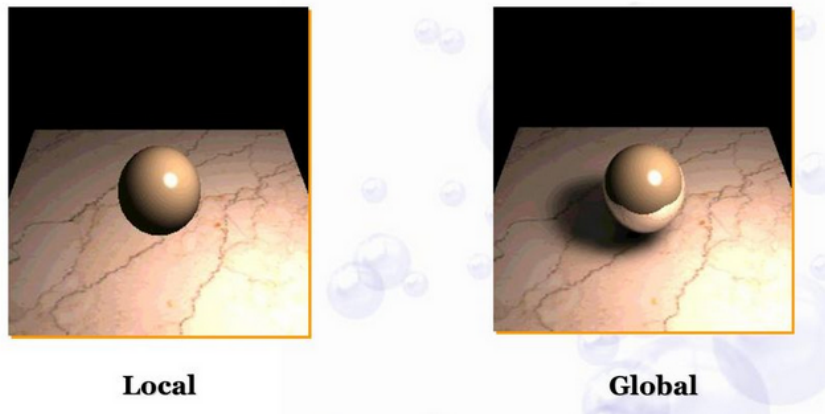
donde:

- $I(x, x')$  describe energía de radiación lumínica observada en el punto  $x$  proveniente de  $x'$
- $g(x, x')$  es un término geométrico, toma el valor de 0 si existe oclusión entre  $x'$  y  $x$  en otro caso su valor es  $\frac{1}{r^2}$  donde  $r$  es la distancia entre  $x'$  y  $x$
- $\epsilon(x, x')$  mide la energía emitida por la superficie en el punto  $x'$  a  $x$
- $\int_S \rho(x, x', x'') I(x', x'') \delta x''$  está compuesta por dos términos:
  - $\rho(x, x', x'')$  es el término de dispersión de la luz que llega desde  $x''$  a  $x$  desde el punto  $x'$

- $I(x', x'')$  describe energía de radiación lumínica observada en el punto  $x'$  proveniente de  $x''$

por lo que este término refiere a la intensidad percibida desde  $x$  considerando todas las reflexiones de luz posibles para el espacio  $S$ .

Existen distintos métodos de resolución de la ecuación del rendering, la mayoría implican aproximaciones dado el gran costo computacional requerido para calcular el valor exacto de  $I(x, x')$ . Estos métodos balancean el costo computacional de los algoritmos utilizados y la fidelidad con el valor final de la función. Dependiendo de las decisiones y simplificaciones consideradas existen dos clasificaciones posibles para el modelo: local y global 2.1.



**Figura 2.1:** Dibujado utilizando iluminación local y global

### 2.1.1. Iluminación Local

Los modelos de iluminación local como el propuesto por Phong en 1975 tienen en cuenta las propiedades físicas de los materiales y las superficies de cada uno de los objetos de la escena de forma individual. Es decir, al dibujar uno de los objetos no se toman en cuenta las posibles interacciones de los haces de luz con los objetos restantes.

En referencia a la ecuación del rendering, el término geométrico nunca toma el valor 0 es decir, no se toma en cuenta las colisiones de los haces de luz con otros objetos,  $\epsilon(x, x')$  toma un valor constante (independiente de  $x'$ ) y  $\int_S \rho(x, x', x'') I(x', x'') \delta x''$  toma el valor constante 1.

### 2.1.2. Iluminación Global

El término iluminación global refiere a una modelo de computación gráfica que simula completamente las interacciones de la luz con todos los objetos que se encuentran en la escena. Es decir, en contraposición a la iluminación local, se consideran los fenómenos de reflexión y refracción de la luz.

## 2.2. Radiosidad

El transporte de la luz entre superficies puede ser modelado a través de la propiedad física conocida como la radiosidad, definida como el flujo de energía irradiada por unidad de área.

Originalmente, este modelo de iluminación global fue propuesto por Goral et al. en 1984 se basa en modelos matemáticos similares a los que resuelven el problema de la transferencia de calor en sistemas cerrados, conocidos como métodos de elementos finitos.

### 2.2.1. Radiosidad en superficies lambertianas

La solución propuesta por Goral et al. implica que todas las superficies son idealmente lambertianas. Además considerara que cada superficie irradia energía lumínica en todas direcciones en un diferencial de área  $\delta A$ , para una dirección de vista  $\omega$  puede ser definida como:

$$i = \frac{\delta P}{\cos \phi \delta \omega} \quad (2.2)$$

donde:

- $i$  es la intensidad de la radiación para un punto de vista particular
- $\delta P$  es la energía de la radiación que hemana la superficie en la dirección  $\phi$  con ángulo sólido  $\delta \omega$

En superficies perfectamente lambertianas, la energía reflejada puede ser expresada como:  $\frac{\delta P}{\delta \omega} = k \cos \phi$ . Donde  $k$  es una constante. Sustituyendo en (2.2) se obtiene:  $\frac{\delta P}{\delta \omega} = \frac{k \cos \phi}{\cos \phi} = k$ , esto implica que la energía percibida de un punto  $x$  es constante, independientemente del punto de vista.

Es por esto que la energía total que deja una superficie ( $P$ ) puede ser calculada integrando la energía que deja la superficie en cada dirección posible,

esto es, se integra la energía saliente en un hemiesfera centrada en el punto estudiado:

$$P = \int_{2\pi} \delta P = \int_{2\pi} i \cos \phi \delta \omega = i \int_{2\pi} \cos \phi \delta \omega = i\pi \quad (2.3)$$

Por tanto, dada una superficie  $S_i$ , es posible calcular la energía lumínica que deja la superficie utilizando (2.3). Resta definir la *cerradura* de una superficie, definiremos la cerradura de una superficie como los límites que definen los puntos internos y externos de esta, llamaremos parche a cada una de estas superficies cerradas. Esto hace que el problema sea resoluble utilizando métodos de elementos finitos, con esta re-formulación del problema, es fácilmente trasladable a la ecuación (2.1).

$$B_j = E_j + \rho_j \sum_{i=1} N B_i F_{ij} \quad (2.4)$$

donde:

- $B_j$  es la intensidad lumínica (radiosidad) que deja la superficie  $j$ .
- $E_j$  es la intensidad lumínica directamente emitida por  $j$ .
- $\rho_j$  es la reflectividad del material para la superficie  $j$ .
- $F_{ij}$  se denomina *factor de forma*, un término que representa la fracción de energía lumínica que deja la superficie  $i$  y llega a  $j$ .

Cabe destacar que la naturaleza recursiva de la ecuación anterior, implica que se toman en cuenta todas las reflexiones difusas que existan en la escena. Como puede observarse, resolver el sistema de  $N$  ecuaciones lineales bastaría para conocer la energía emitida por cada superficie cerrada.

$E$ ,  $\rho$  dependen de los materiales que compongan la escena, son parámetros dados. Sin embargo, resta computar la matriz de factores de forma  $\mathbf{F}$  para finalmente obtener el vector de radiosidades  $B$ . Para determinar una entrada de la matriz  $F_{ij}$  involucrando a las superficies  $i$  y  $j$  de área  $A(i)$ ,  $A(j)$ , considerando los diferenciales infinitesimales de área  $\delta A_i$ ,  $\delta A_j$  el ángulo sólido visto por  $\delta A_i$  es  $\delta \omega = \frac{\cos \phi_j \delta A_j}{r^2}$ . Sustituyendo en (2.3) se obtiene:

$$\delta P_i \delta A_i = i_i \cos \phi_i \delta \omega \delta A_i = \frac{P_i \cos \phi_i \cos \phi_j \delta A_i \delta A_j}{\pi r^2} \quad (2.5)$$

Considerando que  $P_i A_i$  es la energía que deja  $i$ , y que el factor de forma  $F_{ij}$  representa el porcentaje de dicha energía que llega a  $j$  podemos observar

que:

$$F_{\delta A_i - \delta A_j} = \frac{\frac{P_i \cos \phi_i \cos \phi_j \delta A_i \delta A_j}{\pi r^2}}{P_i \delta A_i} = \frac{\cos \phi_i \cos \phi_j \delta A_i}{\pi r^2} \quad (2.6)$$

Integrando, para obtener el factor de forma para el área total:

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \phi_i \cos \phi_j \delta A_i \delta A_j}{\pi r^2} \quad (2.7)$$

De (2.7) se obtienen las siguientes propiedades:

1.  $A_i F_{ij} = A_j F_{ji}$
2.  $\sum_{j=1}^N F_{ij} = 1$
3.  $F_{ii} = 0$
4.  $F_{ij}$  toma el valor correspondiente a la proyección de  $j$  en una hemiesfera unitaria centrada en  $i$ , proyectándola a su vez en un disco unitario.

## 2.3. Métodos de cálculo de la matriz de Factores de Forma

### 2.3.1. Rasterización

El cálculo de la matriz de factores de forma  $\mathbf{F}$  supone la proyección de los parches, de aquí en más se podrá asumir que estos parches son polígonos. Por tanto, es posible aprovechar la rasterización para calcularlos.

La rasterización es una técnica que consiste en la conversión de gráficos vectoriales (en este caso, polígonos) para convertirlos en imágenes compuestas por píxeles.

El proceso de dibujado consiste en un conjunto de etapas bien definidas, a efectos de este informe, se considerarán aquellas que apliquen al trabajo realizado.

El pipeline gráfico consiste en el procesamiento de primitivas geométricas proveídas por el usuario, la rasterización y procesamiento de píxeles.

1. Procesamiento de geometría:

Esta técnica es extremadamente rápida, además la mayoría de dispositivos contienen hardware especializado capaz de acelerar estos cálculos, comúnmente



conocidos como Unidades de Procesamiento Gráfico (o GPU en sus siglas en inglés). Es por este motivo que existe el método del hemicubo para el cálculo de factores de forma.

## El método del hemicubo

El hardware optimizado para realizar operaciones de rasterización tiene la capacidad de proyectar escenas tridimensionales en imágenes bidimensionales a gran velocidad.

Para utilizar el hardware eficientemente consideraremos que se calculará una fila completa de  $\mathbf{F}$ , esto implica que dada  $S_i$ , una superficie, calcularemos simultáneamente los factores de forma para las superficies restantes.

El método original propone la proyección de la escena una hemiesfera centrada en  $S_i$ , sin embargo los modelos de proyección utilizados no lo permiten. Por esto es necesario proyectar la escena a un hemicubo centrado en  $S_i$ , esto supone el dibujado de cinco superficies bidimensionales, y por tanto puede ser realizada utilizando la rasterización.

Este método aprovecha el buffer de profundidad (Z-buffer), tomando en cuenta los píxeles proyectados para los elementos que se encuentren más cercanos al parche  $S_i$ .

El algoritmo, propuesto originalmente por Cohen and Greenberg en 1985, propone rasterizar la escena tridimensional en cinco texturas correspondientes al hemicubo, para cada pixel renderizado se sumará un valor diferencial del factor de forma, que dependerá de la posición del píxel en el hemicubo en relación a la hemiesfera que este aproxima. Esta suma genera una fila de la matriz  $\mathbf{F}$ , específicamente la fila  $\mathbf{F}_i$ .

Por tanto, podremos definir:

$$\mathbf{F}_{ij} = \sum_{q=1}^R \delta F_q \quad (2.8)$$

donde:

- $R$  es la cantidad de píxeles correspondientes a la superficie  $S_j$  que cubren el hemicubo.
- $\delta F_q$  el diferencial de factor de forma asociado al píxel del hemicubo  $q$ .

Los diferenciales de factores de forma deben corregir la deformación introducida con el cambio de proyección desde una hemiesfera a un hemicubo,

para ello, para cada píxel que compone el hemicubo es necesario calcular la proporción de área que este término ocupa en la hemiesfera unitaria.

Para la cara superior, los diferenciales se calculan como:

$$\delta F_q = \frac{\cos \phi_i \cos \phi_j}{\pi r^2} \delta A = \frac{\delta A}{\pi(x^2 + y^2 + 1)} \quad (2.9)$$

Para las caras laterales, la fórmula dada es:

$$\delta F_q = \frac{\cos \phi_i \cos \phi_j}{\pi r^2} \delta A = \frac{z \delta A}{\pi(x^2 + z^2 + 1)} \quad (2.10)$$

### 2.3.2. Raytracing

Embree

## 2.4. Cálculo del vector de radiosidades

Luego de computar la matriz  $\mathbf{F}$  y dado los vectores de emisiones  $E$  y reflexiones  $\rho$ , resta computar el vector de radiosidades correspondiente para cada parche, denominado  $B$ .

Recordando (2.4), es posible deducir el problema al sistema de ecuaciones dado por:

$$E = (\mathbf{I} - \mathbf{R}\mathbf{F})B \quad (2.11)$$

Los estudios de álgebra lineal modernos permiten la resolución de sistemas de ecuaciones de forma optimizada, dependiendo de las propiedades observadas.

Recordando las propiedades en 2.2.1, podemos observar que:

- $\sum_{j=1}^N \mathbf{F}_{ij} \leq 1 \forall i \in [1, N]$
- $\rho_i \leq 1 \rightarrow \sum_{j=1}^N \mathbf{R}_{ij} \leq 1 \forall i \in [1, N]$

Esto implica que las entradas de  $\mathbf{R}\mathbf{F}$  son siempre menores a 1, por tanto,  $(\mathbf{I} - \mathbf{R}\mathbf{F}) = M$  es diagonal dominante ya que  $\sum_{j=1}^N |R_{ij}F_{ij}| \leq 1 \forall i \in [1, N]$  y  $R_{ii}F_{ii} = 0 \forall i \in [1, N]$ . Esto garantiza la convergencia del uso de métodos de resolución iterativos, como el algoritmo de Gauss-Seidel.

Si bien existen maneras alternativas de resolución del sistema planteado con consideraciones específicas del problema a efectos de esta investigación se

considerarán los métodos de resolución de ecuaciones lineales que requieran a lo sumo esta propiedad.

Cabe aclarar, que el método planteado hasta el momento resuelve la radiosidad en un único canal. Es decir, no se toma en cuenta todo el espectro electromagnético de la luz, es por ello que puede establecerse una extensión del método. Esta extensión implica la existencia de tres vectores de reflexión, uno para cada canal *RGB*. Por tanto es necesario que se resuelvan tres y no un único sistema de ecuaciones, aunque cabe destacar que la matriz  $\mathbf{F}$  permanece constante, ya que las oclusiones geométricas no dependen de la longitud de onda.

## Capítulo 3

### Alcance y diseño

#### 3.1. Alcance

#### 3.2. Diseño

# Capítulo 4

## Implementación

### 4.1. OpenGL

4.1.1. Cálculo de factores de forma de la componente difusa

4.1.2. Cálculo de factores de forma de la componente especular

### 4.2. Embree

4.2.1. Cálculo de factores de forma de la componente difusa

4.2.2. Cálculo de factores de forma de la componente especular

### 4.3. OpenGL y Embree

4.3.1. Cálculo de factores de forma de la componente difusa

4.3.2. Cálculo de factores de forma de la componente especular

## Capítulo 5

# Experimental

5.1. Hardware

5.2. Escenas

5.3. Casos de prueba

## Capítulo 6

### Conclusiones y trabajo futuro

#### 6.1. Conclusiones

#### 6.2. Trabajo futuro





# Lista de figuras

2.1	Dibujado utilizando iluminación local y global . . . . .	3
-----	--	---

## Lista de tablas

# APÉNDICES

## Apéndice



# Referencias bibliográficas

- [1] James T Kajiya. The rendering equation. In ACM SIGGRAPH computer graphics, volume 20, pages 143–150. ACM, 1986.
- [2] Bui Tuong Phong. Illumination for computer generated pictures. Communications of the ACM, 18(6):311–317, 1975.
- [3] Cindy M Goral, Kenneth E Torrance, Donald P Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. In ACM SIGGRAPH computer graphics, volume 18, pages 213–222. ACM, 1984.
- [4] Michael F Cohen and Donald P Greenberg. The hemi-cube: A radiosity solution for complex environments. In ACM SIGGRAPH Computer Graphics, volume 19, pages 31–40. ACM, 1985.