

*"To Realism and Beyond!"*



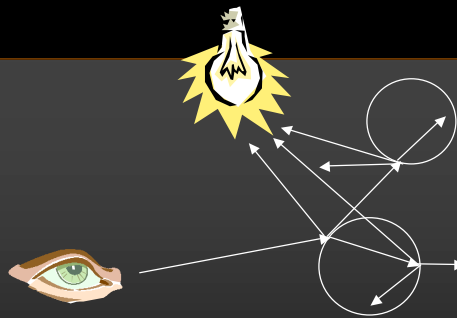
## Iluminação Global Além de RT e Radiosidade

Marcelo Walter  
UFPE

atualização maio/2009

## Algumas considerações sobre Ray Tracing

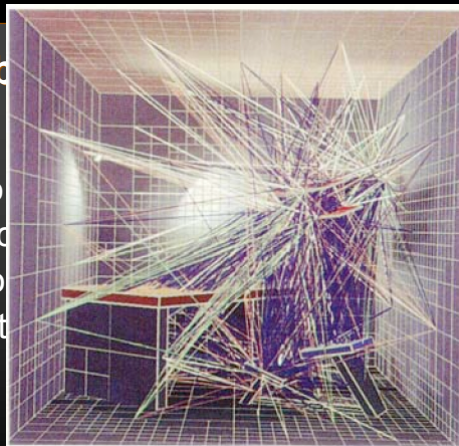
- Raios enviados do olho para a cena
  - Porque?



**Porque a maioria dos raios originados nas fontes de luz nunca atingem o olho!**

## Traçado de Raios

- **Má distribuição**
  - Porquê?
  - Porque o expo...
  - resultado...
  - important...



alho

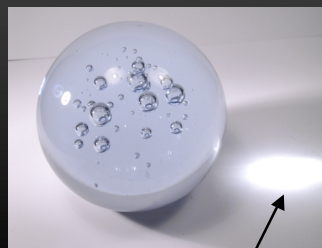
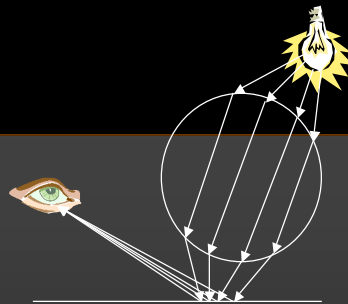
ento, o

D

3

## Casos Complicados I

- **Cáusticas**
  - Luz foca através de uma superfície especular numa superfície difusa
  - Qual direção devem ser os raios secundários enviados para detectar as cáusticas?



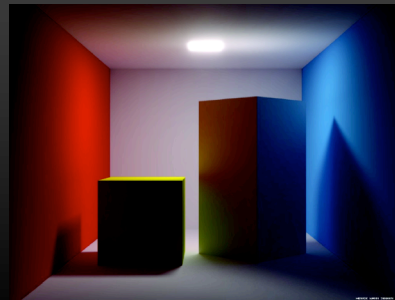
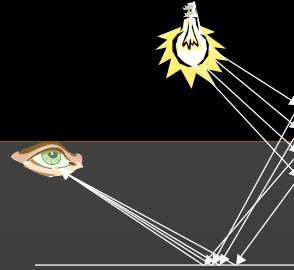
Afinal é difuso aqui...

4

## Casos Complicados II

- **Bleeding**

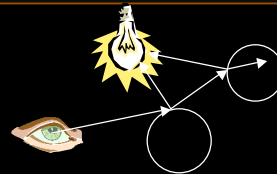
- Cor de uma superfície difusa refletida em outra superfície difusa
- Em que direção emitir os raios secundários?



5

## Path Tracing

- Uma extensão de ray tracing [Kajiya, SIGGRAPH 86] Mesmo paper da *Rendering Equation*
- Reflexão difusa gera número infinito de raios
- Seleciona um raio aleatoriamente



6

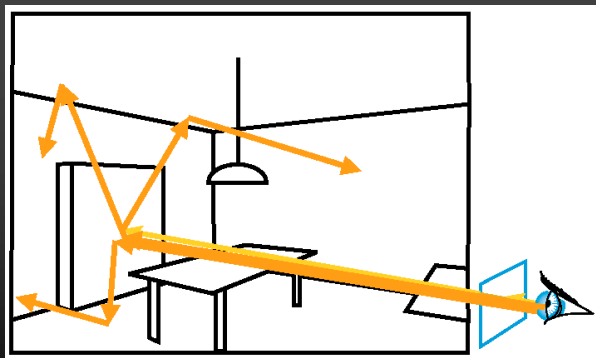
## Path Tracing

- A distribuição de luz é amostrada através do envio de raios **aleatórios** ao longo de todos os caminhos de iluminação possíveis
- Uma abordagem **Monte Carlo** para o problema de iluminação global
- A média de várias amostras provê uma estimativa da luz total que chega no pixel

7

## Monte Carlo (MC) Path Tracing

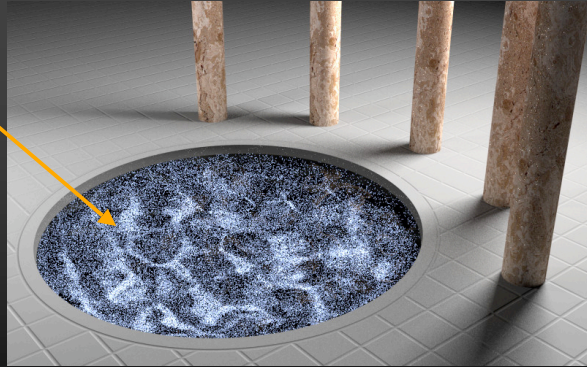
- Envia **apenas um raio** secundário por recursão
- Mas **envia muitos raios** primários por pixels



8

## Principal problema da Integração MC

- Ruído



- Diminui com número de amostras

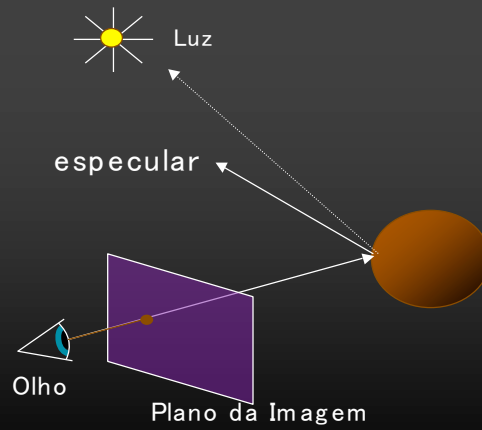
9

## Algoritmo para MC *Path Tracing*

- Envia o raio e encontra intersecção
- Do ponto de intersecção enviar
  - Um raio para cada fonte de luz
  - Um raio adicional (critério para este raio mais adiante)
    - Reflexão difusa, OU
    - Reflexão especular, OU
    - Um raio transmitido
- Produz um ray “path” – não uma árvore de raios

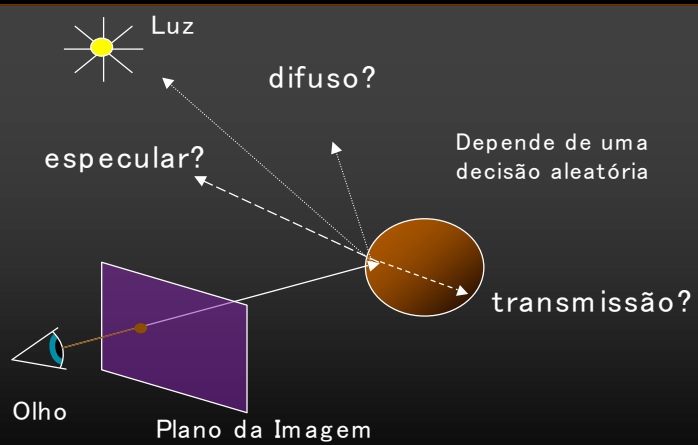
10

## Traçado de Raios



11

## Path Tracing



12

## Estratégias de Amostragem

- **Como escolher a distribuição dos raios aleatórios**
- **Duas possibilidades:**
  - Amostragem por regiões (*stratified sampling*)
    - Divide as direções possíveis em sub-regiões e envia uma amostra por subregião
  - *Importance sampling*
    - Amostragem de acordo com a BRDF
  - Uniforme
    - Amostragem de acordo com parâmetros do objeto

13

## Escolhendo o tipo de raio

- **Como selecionamos qual raio enviar?**
- **Cada material tem um  $k_d$ ,  $k_s$ , e  $k_t$** 
  - Seja  $k_{tot} = k_d + k_s + k_t$
  - Selecionar um número aleatório  $R$  no intervalo  $(0, k_{tot})$ 
    - Se  $(R < k_d)$  então dispara difuso
    - Senão, se  $(R < k_d + k_s)$  dispara especular
    - Senão dispara transmitido

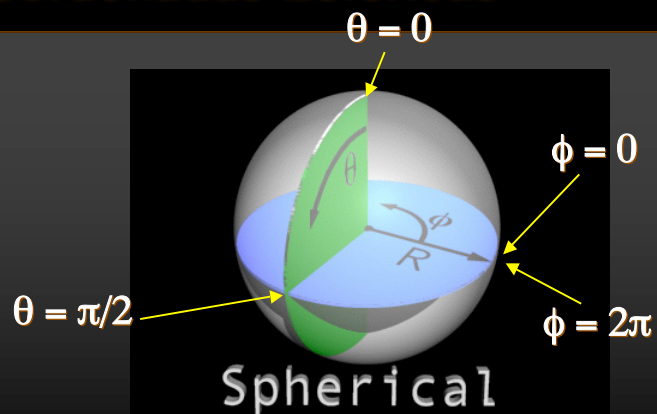
14

## Exemplo

- Sejam  $k_d=0.5$ ,  $k_s=0.3$ ,  $k_t=0.2$ 
  - Valores aleatórios ate 0.5 envia raio difuso
  - Valores aleatórios maiores do 0.5 e menores do que  $0.5+0.3$  envia raio especular
  - Valores maiores do que 0.8 envia raio transmitido

15

## Para cobrir o semi-hemisfério Coordenadas Esféricas



16



## Calculando a Reflexão Difusa

- Podemos calcular uma direção aleatória com o seguinte:
  - Dados dois números aleatórios  $\xi_1$  em  $[0,1]$  e  $\xi_2$  em  $[0,1]$ , a direção  $\omega_d$  refletida aleatoriamente é dada por

$$\omega_d = (\theta, \phi) = (\cos^{-1}(\sqrt{\xi_1}), 2\pi\xi_2)$$

17

## Exemplo

$$\xi_1 = 0.3 \quad \xi_2 = 0.67$$

$$\omega_d = (\cos^{-1}(\sqrt{0.3}), 2\pi \cdot 0.67)$$

$$(\theta, \phi) = (56.78^\circ, 241^\circ)$$

18

## **Vantagens do *Path Tracing***

- **Simula iluminação global**
- **Não desperdiça tempo em coisas não visíveis**
- **Custo constante e proporcional ao número de paths**

19

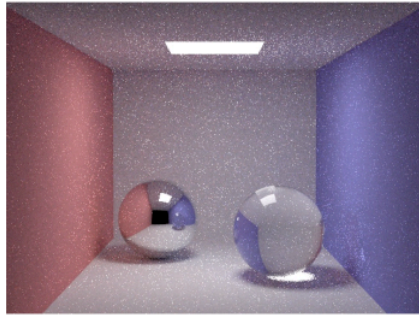
## **Problemas**

- **Necessário traçar muitos raios para conseguir uma “boa” imagem**
- **Tipicamente 100 – 1000 raios por pixel**

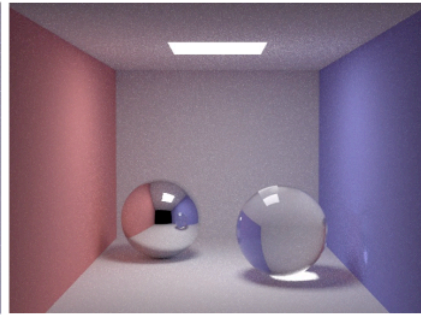
20

## Cornell Box: Path Tracing

---



10 rays per pixel



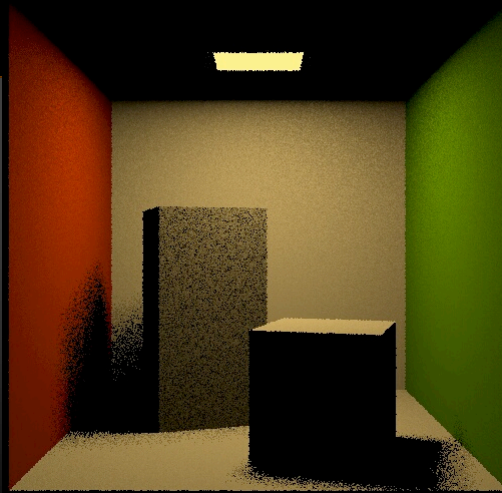
100 rays per pixel

From Jensen, Realistic Image Synthesis Using Photon Maps

2

CS348B Lecture 15

Pat Hanrahan, Spring 2002



22

1 path per pixel

