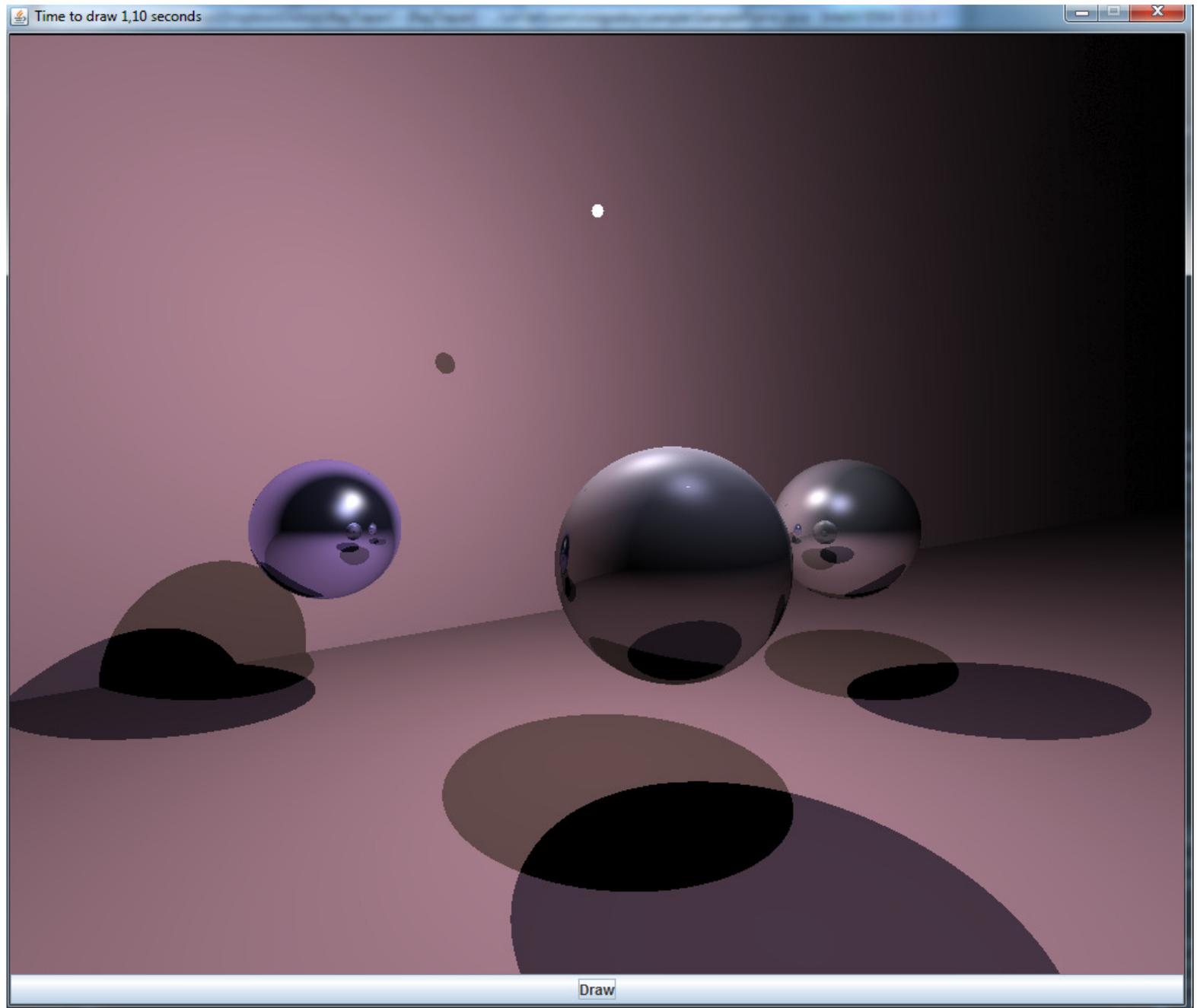


aula 21
Técnicas
de
Iluminação
Global
IC/UFF –
2017

Aura



Modelos de iluminação globais

Ao contrario dos modelos locais que consideram a superfície a luz e o observador, os globais consideram todos os objetos da cena, precisam ter toda a base de dados dos objetos

Principais: Raytracing e radiosidade

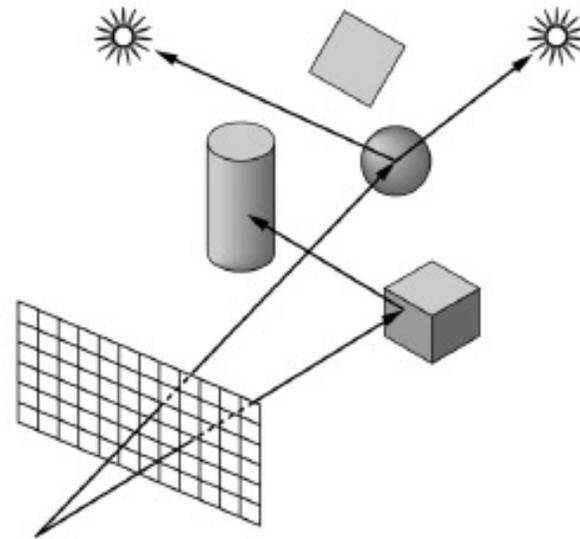
Não produzem os mesmo efeitos nem são adequados pra as mesmas coisas!

Lentos para real time!

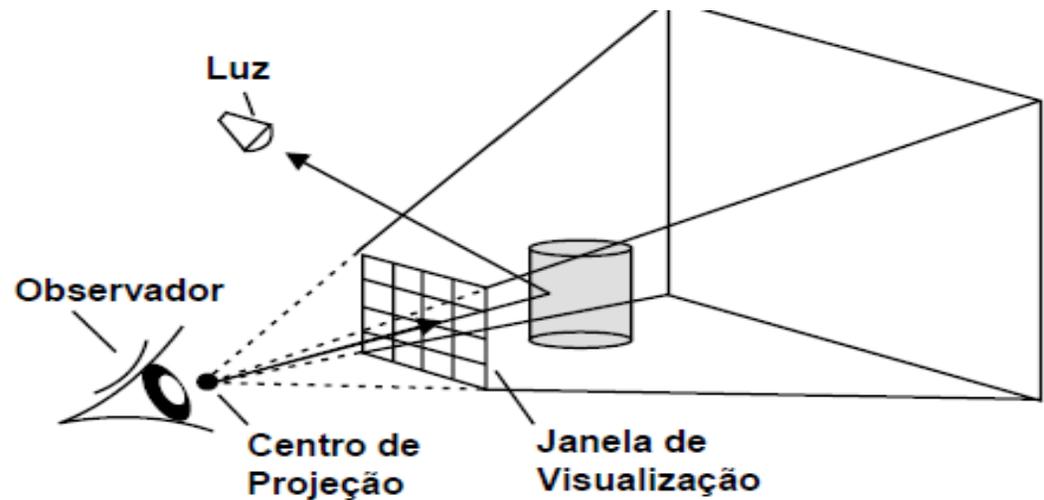
Raytracing

Bom para:

reflexões,
transparências,
objetos fáceis de
calcular interseções
(superfícies,
planas, esférica,
E cilíndricas)



Ray tracing



é uma técnica para gerar uma imagem, seguindo o caminho da luz através de pixels em um plano de imagem e simulando os efeitos de seus encontros com objetos.

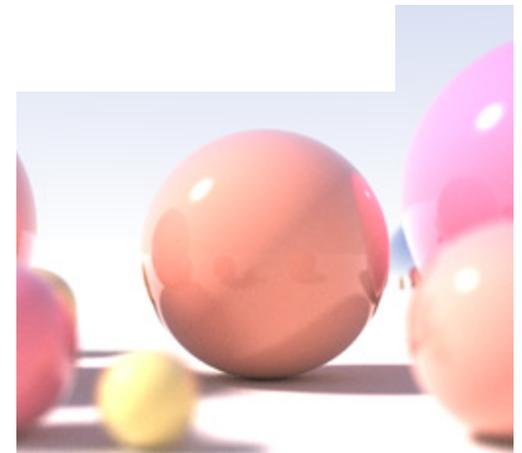
é capaz de produzir um elevado realismo visual, geralmente maior do que o dos métodos de processamento locais típicos, mas em um maior custo computacional.

Isso faz com ray tracing mais adequado para aplicações em que a imagem pode ser renderizada lentamente, como em imagens de cinema e televisão, efeitos visuais, e pouco adequada para aplicações em tempo real, como jogos, onde a velocidade é fundamental.

é capaz de simular uma variedade de efeitos ópticos, como os fenômenos de dispersão, reflexão e refração.

O algoritmo de Ray tracing considera os seguintes pontos:

- Os raios são disparados de forma sistemática, de modo que cada um deles corresponda a um pixel na tela.
- Após o disparo, o raio percorre o espaço podendo atingir um objeto ou sair da cena.
- Se atingir algum objeto, o ponto de intersecção é calculado. As contribuições das fontes de luz para cada ponto, levando em conta a sombra de outros objetos, também são calculadas.
- Se o objeto for opaco, a soma dessas contribuições será a intensidade luminosa total naquele ponto.
- Caso contrário, as contribuições devidas aos reflexos e refrações, serão também computadas. O pixel correspondente pode, então, ser exibido.
- Se não houver intersecção, o pixel terá a cor de fundo.



Algoritmo clássico

Para cada pixel da tela:

1. Trace um “raio” a partir do observador até a cena a ser representada através de um pixel da tela;
2. Determine qual o primeiro objeto a interceptar esse raio;
3. Calcule a cor ambiente da superfície do objeto no ponto de interseção baseado nas características do objeto e na luz ambiente;
4. Se a superfície do objeto for reflexiva, calcule um novo raio a partir do ponto de interseção e na “direção de reflexão”;
5. Se a superfície do objeto for transparente, calcule um novo raio a partir do ponto de interseção.
6. Considere a cor de todos os objetos interceptados pelo raio até sair da cena ou atingir uma fonte de luz, e use esse valor para determinar a cor do pixel e se há sombras.



Espelhos:

O ray tracing deve considerar os raios refletidos toda vez que o coeficiente de reflexão de uma superfície for diferente de zero. O coeficiente de reflexão varia entre 0 e 1, determinando que quantidade de energia do raio de luz deve ser considerada como absorvida pelo objeto em questão, compondo uma soma ponderada das componentes de cor para o pixel na tela. Um espelho possui um coeficiente de reflexão próximo de 1, ou seja, nessa superfície todos os raios incidentes devem ser refletidos com o mesmo ângulo de incidência em relação à direção da reta normal à superfície. Além do coeficiente de reflexão, as superfícies também apresentam um coeficiente de refração que expressa a maneira pela qual a luz passa através de um meio para outro.



Ray tracing *simplificado ou aproximado* *ou*

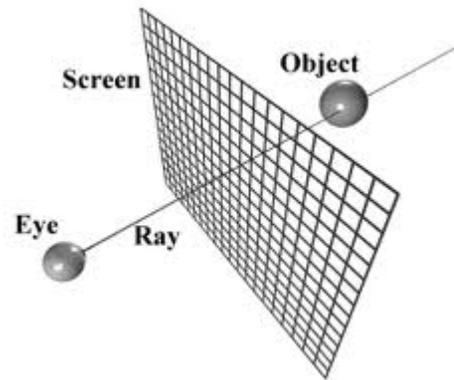
Ray casting lança raios a partir do observador de forma a perceber a distância dos objetos que compõem a cena.

Os raios são emitidos **a partir do observador**, (no sentido inverso do que acontece na natureza), para reduzir recursos computacionais (pois a maior parte dos raios de luz que partem da fonte não chegam ao observador).

Ray casting

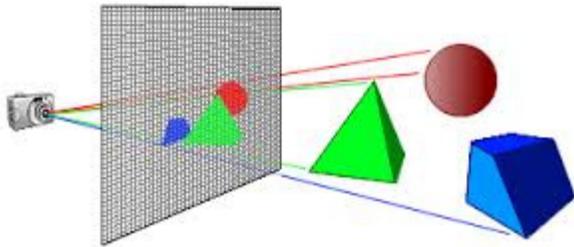
Supõe-se um raio do olho do observador passando por **cada ponto da tela** a ser desenhada. O ponto da tela receberá a cor do objeto que for atingido na cena pelo raio.

O calculo das interseções é o ponto chave do algoritmo.



Ray casting

permite remover as superfícies escondidas utilizando as informações obtidas a partir das primeiras intersecções encontradas pelos raios lançados a partir do observador.



Ray tracing (rastreamento)

Método recursivo, onde recorre ao lançamento de raios secundários a partir das interseções dos raios primários com os objetos.

Ray casting é apropriado para a renderização de jogos 3D em tempo-real.

Durante a viagem do raio pode acontecer: absorção, reflexão ou refração. A superfície pode refletir toda ou apenas uma parte do raio numa ou mais direção. A soma das componentes absorvidas, refletidas e refratadas tem que ser igual ao inicial.

Cálculo de interseções:

A tarefa principal do ray tracing consiste no cálculo da interseção de um raio com o objeto. Para essa tarefa, utiliza-se normalmente a representação paramétrica de um vetor ou reta. Cada ponto (x, y, z) ao longo de um raio com origem no ponto (x_0, y_0, z_0) e direção do ponto (x_0, y_0, z_0) para o ponto (x_1, y_1, z_1) é definido em função do parâmetro t , (com valores no intervalo $[0,1]$ pelas equações paramétricas da reta):

$$x = x_0 + t(x_1 - x_0);$$

$$y = y_0 + t(y_1 - y_0);$$

$$z = z_0 + t(z_1 - z_0);$$

$$x = x_0 + t\Delta x; \Delta x = x_1 - x_0$$

$$y = y_0 + t\Delta y; \Delta y = y_1 - y_0$$

$$z = z_0 + t\Delta z; \Delta z = z_1 - z_0$$

(x_0, y_0, z_0) for considerado o centro de projeção, ou o olho do observador

(x_1, y_1, z_1) for o centro de um pixel na “janela”

t varia de 0 a 1 entre esses pontos.

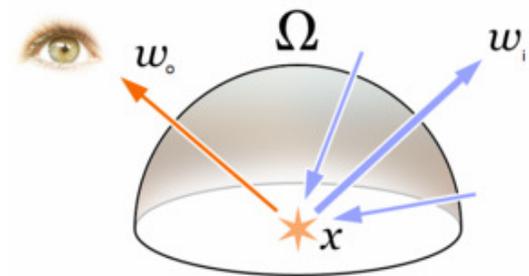
valores de t maiores que 1 correspondem a pontos depois da janela

Radiosidade

considera a solução da integral de rendering (equilíbrio da radiância em um ponto ou a conservação da energia) para modelar a iluminação.

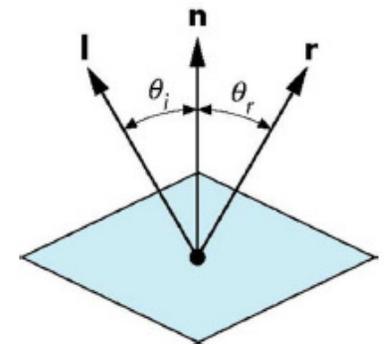
O nível de realismo da modelagem é muito maior.

Considera a função bidirecional de distribuição da reflectancia-**bidirectional reflectance distribution function (BRDF)**.



Os anteriores todos consideram
Que os 3 vetores estão no mesmo plano
(reflexão ideal)

$$\mathbf{r} = 2 (\mathbf{l} \cdot \mathbf{n}) \mathbf{n} - \mathbf{l}$$



Radiosidade é:

- uma aplicação do método de elementos finitos para resolver a equação de renderização para cenas com superfícies que refletem a luz de forma difusa.
- um algoritmo de iluminação global: a iluminação não vem apenas a partir das fontes de luz, mas todas as superfícies de cena interagindo uns com os outros.
- independente do ponto de vista**, o que aumenta o volume dos cálculos envolvidos, mas torna-os **úteis para todos os pontos de vista**.
- inicialmente uma aplicação desenvolvidos na área de transferência de calor, posteriormente adaptada para a aplicação de computação gráfica (1984 na Universidade de Cornell).

Color bleeding

Em rendering , **color bleeding** é a ocorrência de colorização de um objeto ou superfície pela cor refletida de superfícies próximas.

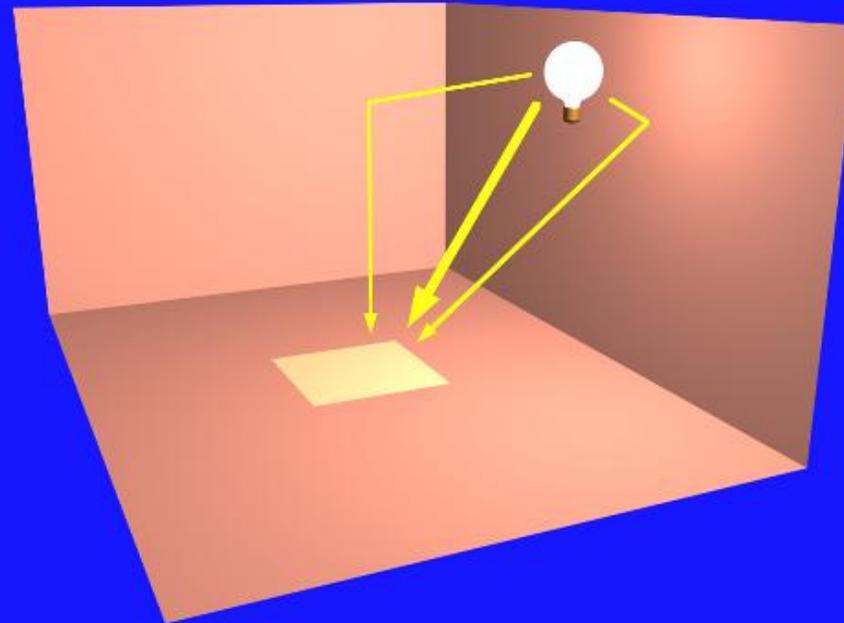
Ocorre principalmente quando se usa Radiosity para a cena 3D.



Radiosidade: discretiza o ambiente em um malha

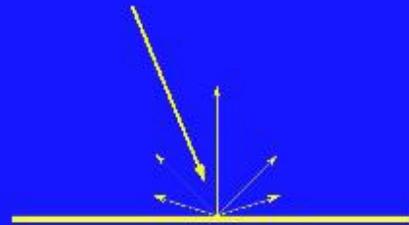


Os limites da malha devem coincidir com os limites das zonas de diferença de iluminação

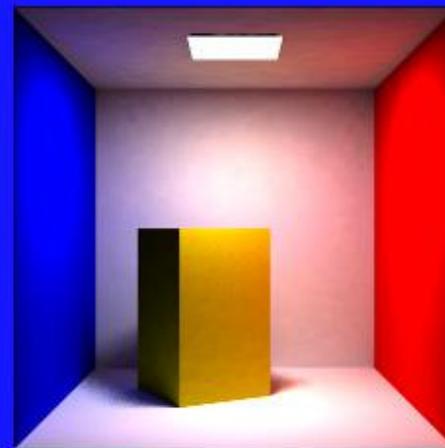


The illumination at a given point in the environment is a combination of the light received directly from a light source and the light which is reflected one or more times from the surfaces of the environment.

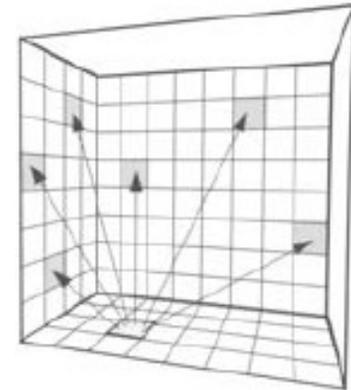
Balanço ou equilíbrio de energia radiante



Light striking a surface is reflected in all directions, following the Lambertian reflection model. This diffuse reflection of light leads to color bleeding, as light striking a surface carries that surface's color into the environment.



Radiosidade:



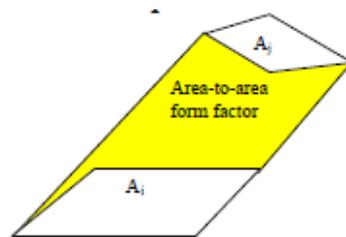
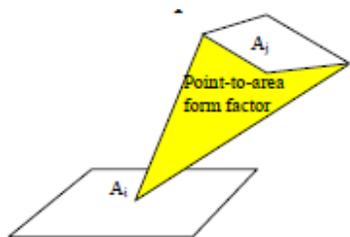
O método da radiosidade é baseado em um modelo simples de balanço de energia. Na sua origem, o cálculo da radiosidade empregado em Transmissão de Calor não é mais do que a aplicação da lei da conservação da energia a cada uma das superfícies de um recinto ou cena, e pressupõe a existência de equilíbrio térmico. Em cada superfície de um modelo, a quantidade de energia emitida é a soma entre a energia que a superfície emite internamente mais a quantidade de energia refletida. A quantidade de energia refletida pode ser caracterizada pelo produto entre a quantidade de energia incidente na superfície e a constante de reflexão da superfície.

$$B_j = \rho_j H_j + E_j$$

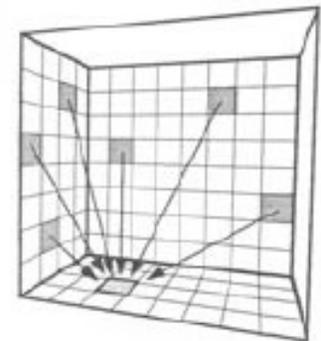
onde B_j é a radiosidade da superfície j , ρ_j sua reflectividade, H_j a energia incidente nesta superfície e E_j a energia emitida pela superfície j

Radiosidade

A radiosidade de uma superfície é a energia dissipada. Isso é usado para determinar a intensidade luminosa da superfície. A quantidade de energia emitida por uma superfície deve ser especificada como um parâmetro do modelo, como nos métodos tradicionais onde a localização e a intensidade das fontes de luz devem ser especificadas. A reflectividade da superfície também deve ser especificada no modelo, como nos métodos de iluminação tradicional. A única incógnita da equação é a quantidade de luz incidente na superfície. Esta pode ser encontrado somando-se todas as outras superfícies à quantidade de energia refletida que contribui com a iluminação dessa superfície:



$$H_j = \sum_{i=1}^n B_j F_{ij}$$



onde H_j é a energia incidente na superfície j , B_j a radiosidade de cada superfície i da cena e F_{ij} uma constante $i j$.

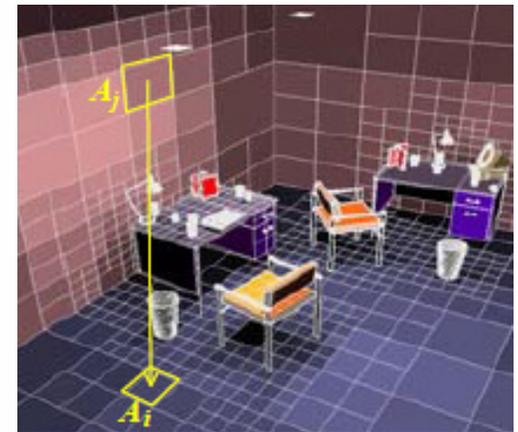
A constante dessa equação é definida como a fração de energia que sai da superfície i e chega na superfície j , e é, portanto, um número entre 0 e 1. Essa constante pode ser calculada por métodos analíticos, ou através de semelhança geométrica.

A equação da radiosidade fica assim:

$$B_j = E_j + \rho_j \sum_{i=1}^n B_i F_{ij}$$

A consideração de todas as superfícies da cena forma uma seqüência de N equações lineares com N incógnitas, o que leva a uma solução matricial:

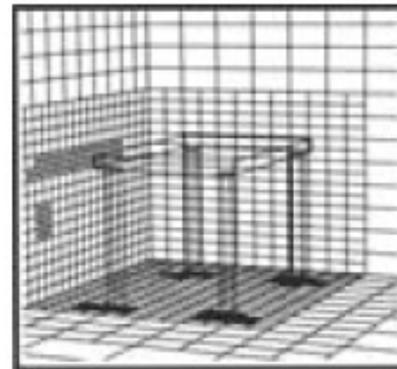
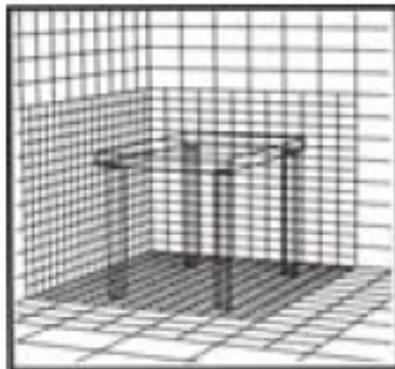
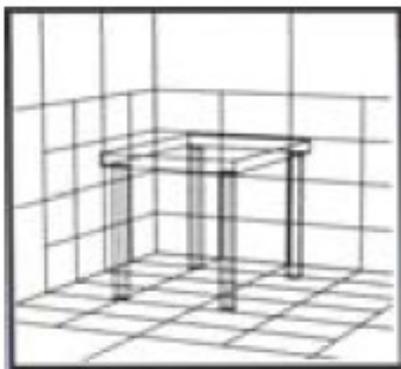
$$\begin{bmatrix} 1 - \rho_1 F_{11} & 1 - \rho_1 F_{12} & \Lambda & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1 - \rho_2 F_{22} & \Lambda & -\rho_2 F_{2n} \\ M & M & O & M \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \Lambda & -\rho_n F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ M \\ B_3 \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ M \\ E_3 \end{bmatrix}$$



Essa matriz tem duas propriedades interessantes: é diagonalmente dominante e, portanto, converge quando usada no método iterativo de Gauss Seidel, [GOR, 84]. Métodos alternativos para o cálculo dessa matriz já foram propostos por Cohen et al. [COH, 88]. Alguns permitem uma convergência para a solução correta mais rápida que o algoritmo iterativo de Gauss Seidel.

Refinamentos progressivos

Alterando o numero de elementos da malha:



Coarse patch solution
(145 patches)

Improved solution
(1021 subpatches)

Adaptive subdivision
(1306 subpatches)

Sombreamento anisotrópico

Isotrópico x ortotrópico



Photon mapping

Algoritmo de **iluminação global** em 2 passadas (two-pass) que considera modelos de radiância para maior realismo na simulação da refração e reflexão da luz em superfícies transparentes

É capaz de simular a refração da luz em meios transparentes tal como o vidro ou a água, interreflexões difusas entre objetos iluminados, a dispersão da luz sob a superfícies de materiais translúcidos, e efeitos causados por partículas, tal como o **fumaça ou a água de vapor**.

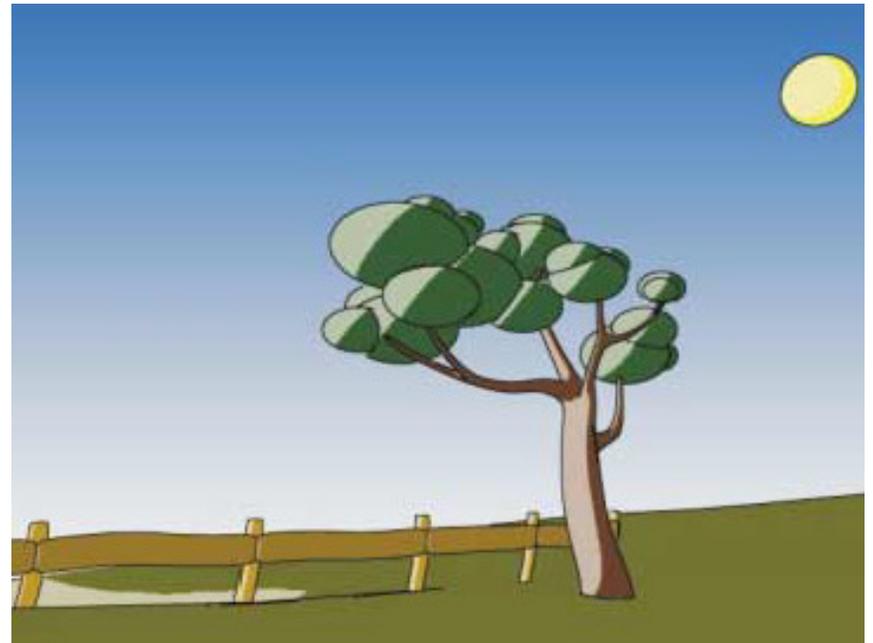
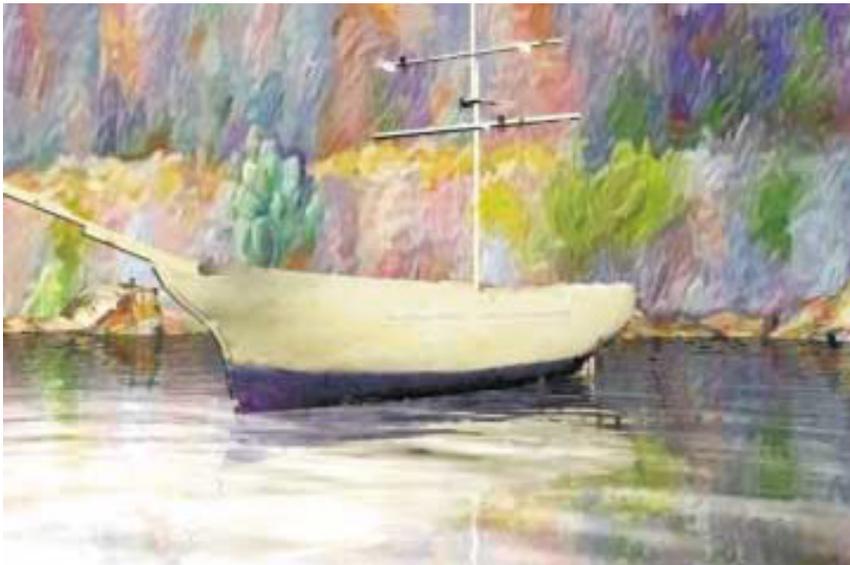


Há muito mais do que isso!

Vimos aqui apenas sobre um realismo fotográfico das imagens, mas há diversas outras formas e esse assunto está sempre em constante evolução. Assim depois desta leve introdução continue na área! Você já tem a bagagem teórica que precisa para agora descobrir o resto sozinho!

Toon Shading

Stylistic rendering



Bibliografia:

E. Azevedo, A. Conci, *Computação Gráfica: teoria e prática*, Campus ; - Rio de Janeiro, 2003

J.D.Foley, A. van Dam, S.K. Feiner, J.F. Hughes. *Computer Graphics- Principles and Practice*, Addison-Wesley, Reading, 1990.

H. Watt, F. Policarpo - *The Computer* , Addison-Wesley Pub Co (Net); 1998

http://en.wikipedia.org/wiki/Shadow_mapping

https://noppa.oulu.fi/noppa/kurssi/521493s/luennot/521493S_3-d_graphics_vi.pdf

<http://graphics.stanford.edu/papers/rad/>