

Algoritmos de Iluminação Global

Sistemas Gráficos/

Computação Gráfica e Interfaces

Algoritmos de Iluminação Global

Objetivo: calcular a cor de cada ponto a partir da iluminação direta de uma fonte de luz, mais a soma de todas as reflexões das superfícies próximas.

Nos modelos de **iluminação local**, vistos anteriormente, a cor de cada ponto é definida somente pela intensidade luminosa que chega diretamente das fontes de luz.

A iluminação Global respeita a Equação de *Rendering*:

$$I(x, x') = g(x, x') \cdot \left[\varepsilon(x, x') + \int_S \rho(x, x', x'') \cdot I(x', x'') \cdot dx'' \right]$$

$I(x, x')$

Iluminação de x' sobre x

$g(x, x')$

Termo geométrico:

$=0$, se x e x' não se vêem mutuamente

$=1/r^2$, se x e x' se vêem (r : dist. entre ambos)

$\varepsilon(x, x')$

Emissão de luz de x' para x

$\rho(x, x', x'')$

Perc. de Iluminação oriunda de x'' e que é refletida em x' na direção de x

Algoritmos de Iluminação Global a estudar:

Ray Tracing

Radiosity

Algoritmos de Iluminação Global

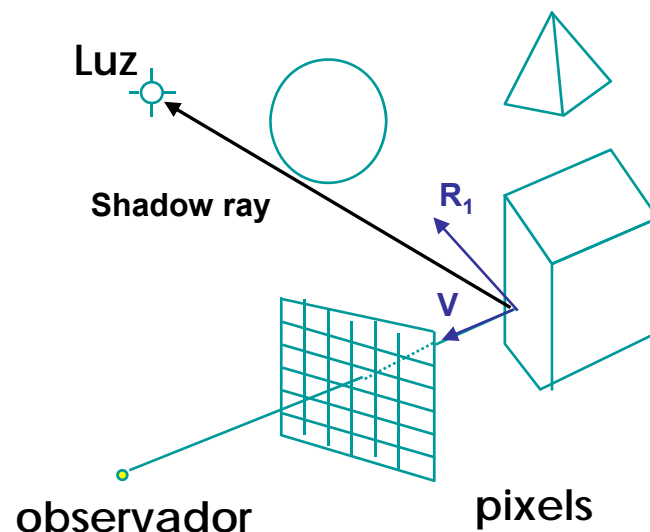
Ray Tracing

Ray Tracing

O algoritmo é uma extensão ao algoritmo *Ray-Casting* visto anteriormente.

O algoritmo depende da posição do observador (*view dependent algorithm*).

- O plano de visualização é discretizado em pontos de amostragem (pixels ou...);
- Faz-se passar, por cada ponto de amostragem, um raio luminoso que parte do observador em direção ao interior da cena.
- O rasto (*tracing*) de cada raio vai permitir somar as contribuições de reflexão entre faces próximas.



R_1 é o vetor de reflexão máxima:

$$R_1 = V - 2 (V \cdot N) N$$

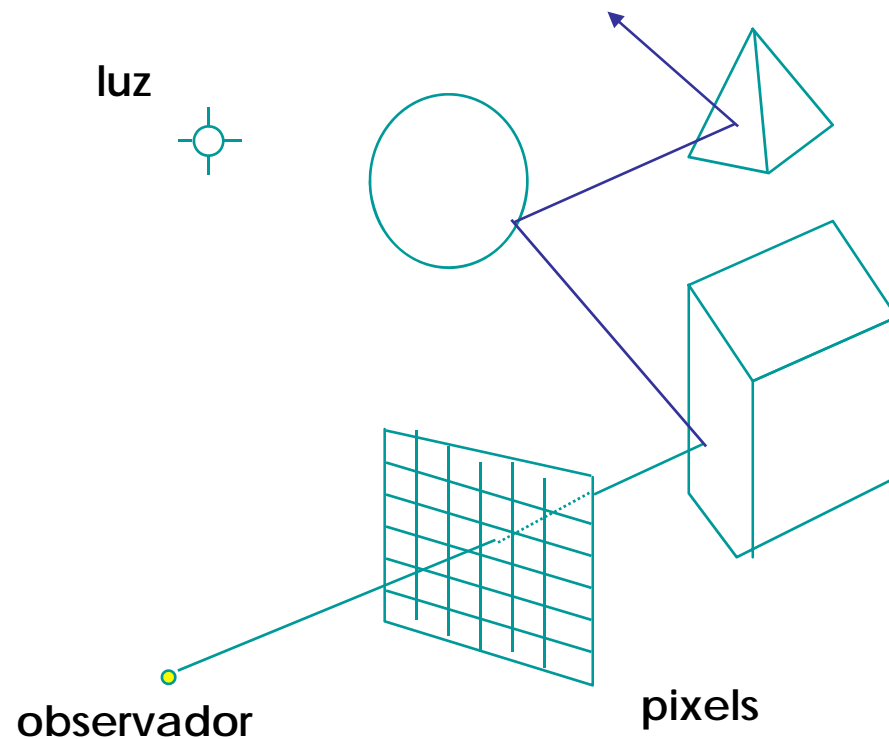
A intensidade luminosa inicial é:

$$I = I_{\text{local}} = k_a I_a + k_d \cdot I_p \cdot (N \cdot L)$$

Algoritmos de Iluminação Global

Ray Tracing

O interceção do raio refletido com os restantes objetos é registado para obter as contribuições destes na iluminação do ponto. A atenuação devido à distância da face pode ser considerada. O processo é recursivo.



A intensidade luminosa agora é:

$$I = I_{\text{local}} + k_r * I_{\text{reflexão}}$$

$I_{\text{reflexão}}$
é calculada recursivamente

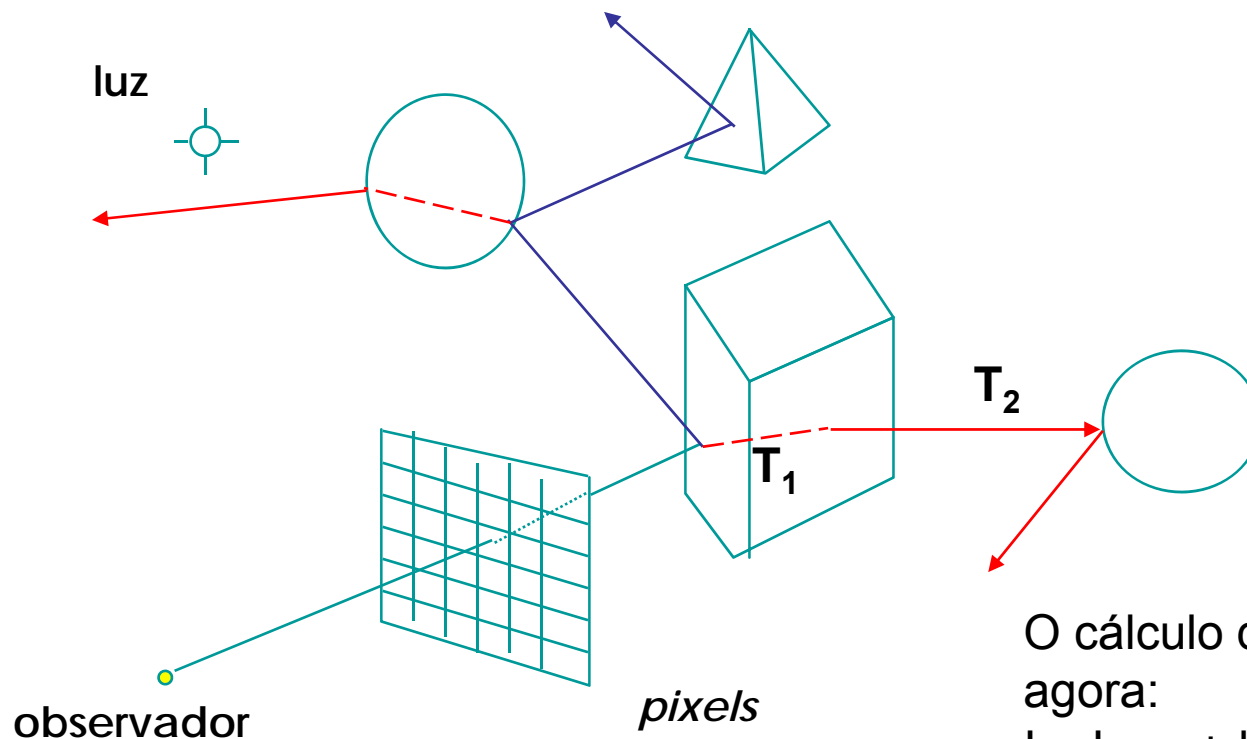
k_r é um coeficiente de Reflexão (semelhante a k_s)

Nota: em cada interceção é necessário determinar qual o objeto mais próximo.

Algoritmos de Iluminação Global

Ray Tracing

Se os objetos forem transparentes ou semitransparentes é necessário considerar os raios transmitidos para o interior do objeto (ou exterior). Por exemplo, os raios T_1 e T_2 .



O cálculo da intensidade é agora:

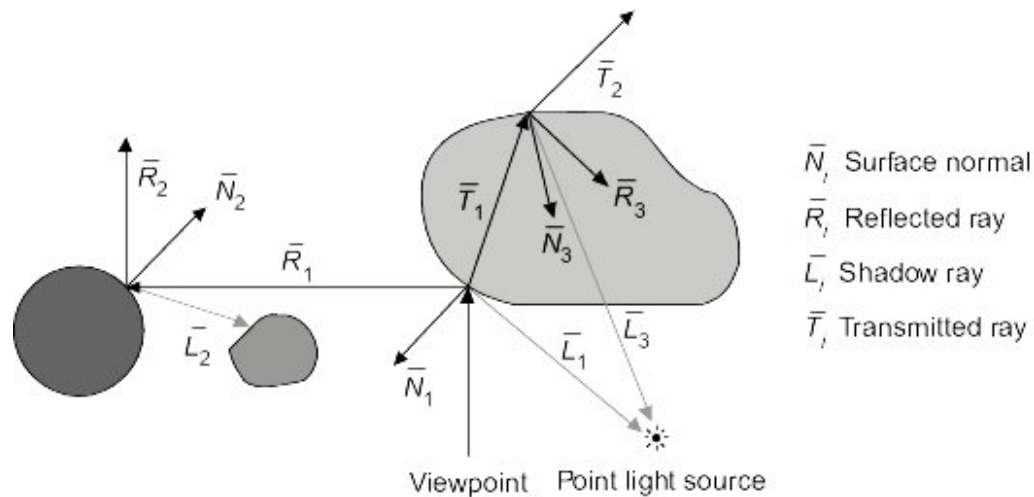
$$I = I_{\text{local}} + k_r * I_{\text{refletida}} + k_t * I_{\text{transmitida}}$$

$I_{\text{refletida}}$ e $I_{\text{transmitida}}$
são calculadas recursivamente

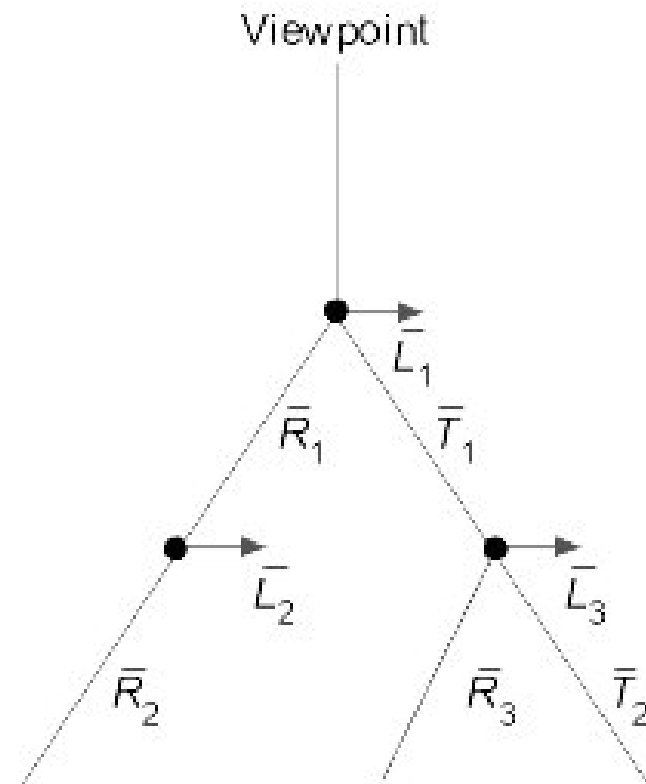
Algoritmos de Iluminação Global

Ray Tracing

Para cada pixel constrói-se uma árvore de interseções. A cor final do pixel determina-se percorrendo a árvore das folhas para a raiz e calculando as contribuições de cada ramo de acordo com o modelo de reflexão.



Nos objetos opacos não existe o raio transmitido.
O ramo da árvore termina quando o raio atinge um objeto não refletor ou o ramo atinge uma determinada profundidade pré-estabelecida



Algoritmos de Iluminação Global

Ray Tracing

O algoritmo de Ray Tracing é vantajoso porque:

sombras, reflexões e refrações são facilmente incorporadas
simula razoavelmente bem os efeitos especulares

O algoritmo de Ray Tracing tem custos computacionais elevados porque:

o custo de cálculo das intersecções é elevado
não simula bem os efeitos de iluminação difusa

(necessidade de outras variantes, mais complexas)

A otimização faz-se em duas áreas:

1. Diminuição do número de raios a processar.
2. Diminuição do número de intersecções a testar

Software *freeware* de Ray-Tracing: <http://www.povray.org>

Algoritmos de Iluminação Global

Ray Tracing

- **Diminuição do número de raios a processar**
 - **"Item Buffers"** - determinam-se quais as áreas do ecrã onde se situam os objetos (pré-processam/, ZBuffer)
 - **"Adaptive Tree-Depth Control"** - não é necessário levar todos os ramos da árvore de shading à sua profundidade máxima: usa a importância de um raio luminoso sobre o pixel a que pertence; esta importância diminui a cada reflexão ou transmissão
 - **"Light-Buffers"** - a cada fonte de luz associam-se listas com os objetos que a rodeiam; os *shadow feelers*, uma vez definida a sua direção, são primariamente testados com os objetos que se encontram na lista respetiva.

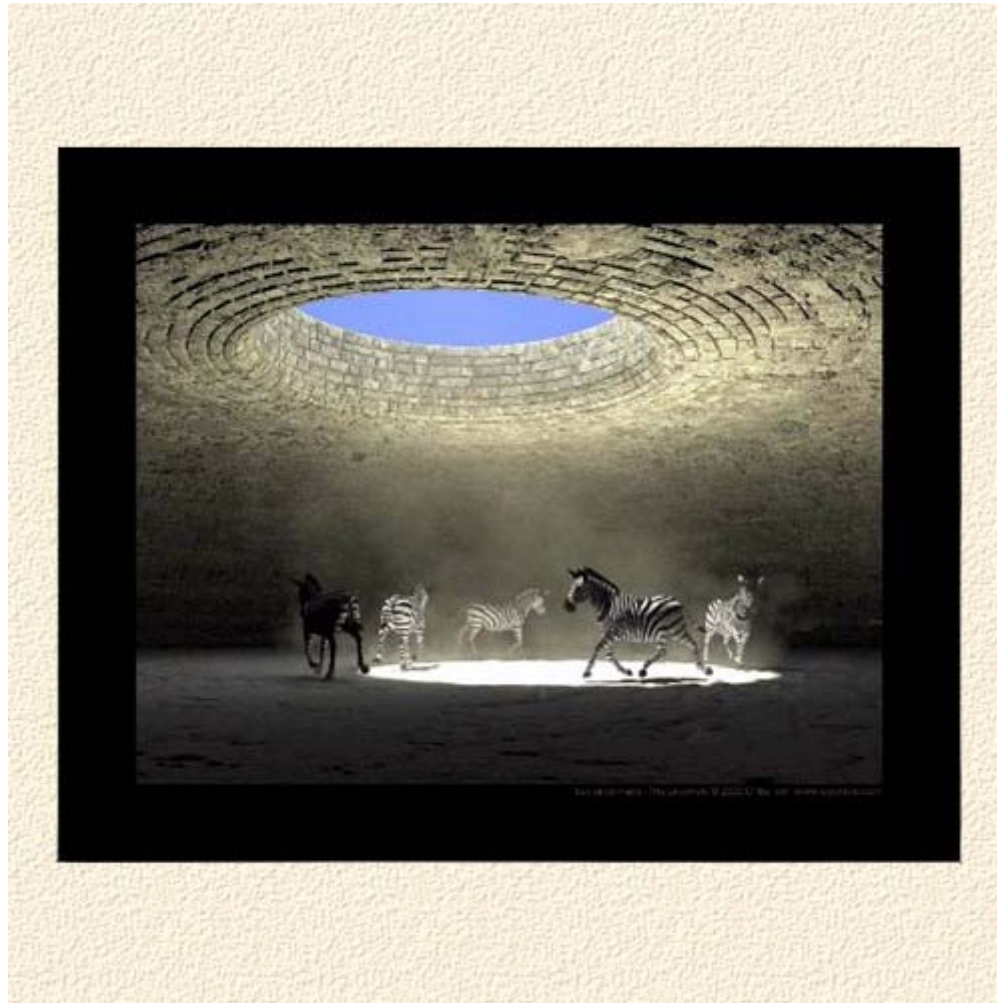
Algoritmos de Iluminação Global

Ray Tracing

- **Diminuição do número de intersecções a testar**
 - **Volumes Envolventes** - antes de efetuar o teste de intersecção de um raio com um objeto, tenta-se a sua intersecção com um volume simples (vulgarmente uma caixa) envolvente do objeto. Este teste prévio é muito rápido (a caixa tem as faces alinhadas com os três eixos) e exclui imediatamente muitos testes de intersecção mais complexos.
 - **Organização Hierárquica dos Volumes Envolventes** - a utilização de volumes envolventes de outros volumes envolventes permite economizar muitos testes de intersecção: se um raio não intersecta um volume, então também não intersecta os volumes nele contidos.
 - **Divisão Espacial em Grelhas Tridimensionais** - cada célula resultante desta divisão conhece os objetos que contém, total ou parcialmente. De acordo com a posição e a direção do raio em questão, só determinadas células são visitadas e, deste modo, só os objetos nelas contidos são testados. Dado que a ordem de progressão nas células é definida pelo sentido do raio, a primeira célula onde se detete uma intersecção termina o processo de visita do raio às células.

Algoritmos de Iluminação Global

Ray-Tracing



Algoritmos de Iluminação Global

Radiosity

O algoritmo é independente do ponto de observação. O algoritmo só efetua, realmente, o cálculo de iluminação; trabalha no espaço objeto. É complementado por um algoritmo de cálculo de visibilidade para a produção da imagem final.

Fases do processamento:

1. Modela as interações de luz entre objetos e fontes de luz, sem considerar a posição do observador.
2. Cria a imagem considerando o observador, efetua cálculo de visibilidade (ex: Z-buffer).

Nos modelos anteriores de iluminação, as fontes de luz foram tratadas de forma diferente das superfícies que iluminam. Pelo contrário, **os métodos de radiosidade** consideram que todas as superfícies podem produzir (emitir) luz. Assim, as fontes de luz são modeladas como superfícies normais, com uma dada área.

O método assume que os processos de emissão e reflexão são **difusos ideais**. Necessita das faces discretizadas em ***patches*** de forma a garantir que a **radiosidade se mantém constante** na área correspondente a um ***patch***.

Algoritmos de Iluminação Global

Radiosity

A radiosidade (B_i) é definida como a energia expelida, por unidade de tempo e de área, de um **patch**, sendo composta por duas partes:

$$B_i A_i = E_i A_i + \rho_i \sum_j (F_{j-i} B_j A_j)$$

energia
expelidaenergia
emitida
(produzida)energia
refletida

Por unidade de área:

$$B_i = E_i + \rho_i \sum_j (F_{j-i} B_j A_j / A_i)$$

B_i - radiosidade, energia expelida do **patch** em Watt/m²

E_i - emissão de luz (auto-emitida) pelo **patch i**

ρ_i - refletividade, percentagem da energia incidente que é refletida pelo **patch i**

F_{j-i} - fator de forma, percentagem de energia que abandona o patch **j** e atinge **i**

Algoritmos de Iluminação Global

Radiosity

Em ambientes difusos, existe a seguinte relação de reciprocidade entre fatores de forma:

$$A_i \cdot F_{i-j} = A_j \cdot F_{j-i}$$

Que aplicada na expressão anterior da radiosidade resulta em:

$$B_i = E_i + \rho_i \sum_j B_j F_{i-j}$$

Ou:

$$B_i - \rho_i \sum_j B_j F_{i-j} = E_i$$

Assim, a interação de luz entre **patches** pode ser representada por um sistema de equações lineares:

$$\begin{bmatrix} 1 - \rho_1 F_{11} & -\rho_1 F_{12} & \dots & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1 - \rho_2 F_{22} & \dots & -\rho_2 F_{2n} \\ \dots & \dots & \dots & \dots \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \dots & 1 - \rho_n F_{nn} \end{bmatrix} \times \begin{bmatrix} B_1 \\ B_2 \\ \dots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \dots \\ E_n \end{bmatrix}$$

Algoritmos de Iluminação Global

Radiosity

Criação da imagem:

1. Resolvendo o sistema de equações, por eliminação Gaussiana, obtém-se a radiosidade para cada **patch**.
 2. Definir a posição do observador.
 3. Aplicar um algoritmo de visibilidade, por exemplo, Z-buffer.
 4. Calcular a radiosidade dos vértices de cada polígono.
 5. Aplicar a interpolação de cor (Gouraud).
- A mesma solução do sistema é usada para qualquer posição do observador.
 - É necessário resolver novamente o sistema de equações se houver alteração de coeficientes de reflexão ρ ou de valores de emissão E .
 - É necessário recalcular os fatores de forma se a geometria da cena for alterada (mudança de posição de algum objeto)

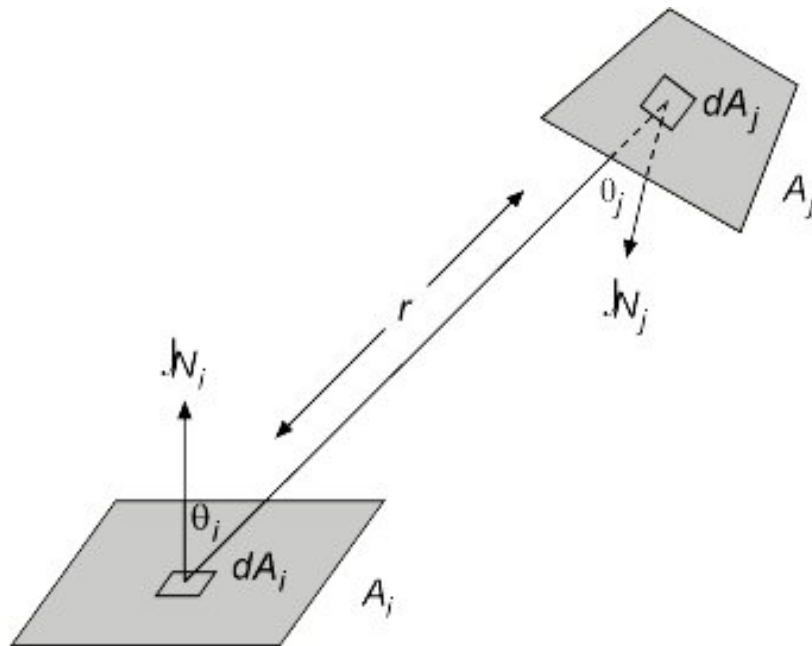
Fatores de Forma: A complexidade do método está no cálculo dos fatores de forma.

Algoritmos de Iluminação Global

Radiosity

Fatores de Forma

O fator de forma **F_{ij}** representa a fração (em percentagem) da energia total expelida pelo **patch “i”** que atinge o **patch “j”**, tomando em consideração a forma, orientação relativa e distância entre ambos os patches, bem como os obstáculos que obstruam o caminho.



O fator de forma da área diferencial dA_i para a área diferencial dA_j é dada por:

$$dF_{di-dj} = \frac{\cos \theta_i \cos \theta_j}{\pi r^2} H_{ij} dA_j$$

H_{ij} é 1 ou 0, dependendo de dA_j ser visível ou não a partir de dA_i .

Algoritmos de Iluminação Global

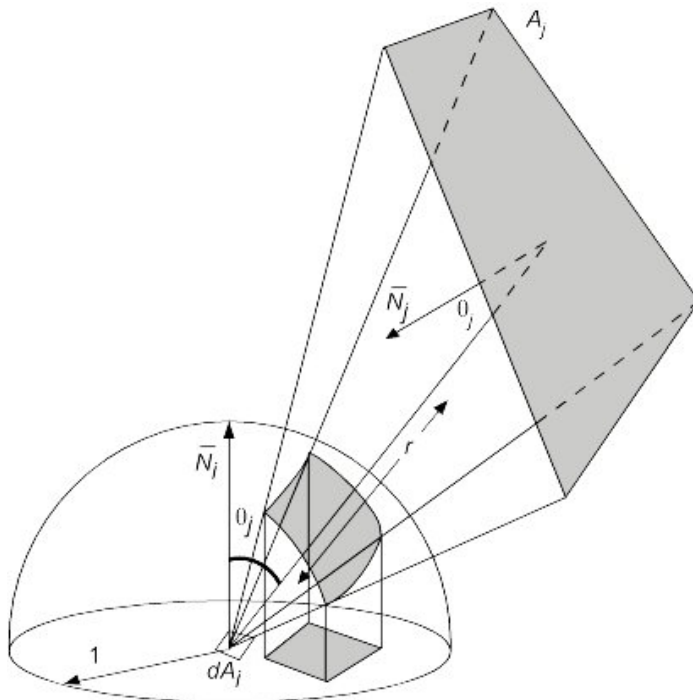
Radiosity

Para determinar F_{di-j} , o fator de forma da área diferencial dA_i para a área finita A_j , integramos a área da **patch j**:

$$F_{di-j} = \int_{A_j} \frac{\cos \theta_i \cos \theta_j}{\pi r^2} H_{ij} dA_j$$

Finalmente o fator de forma da área A_i para a área A_j é dado por:

$$F_{i-j} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \theta_i \cos \theta_j}{\pi r^2} H_{ij} dA_j dA_i$$



Verifica-se que o calculo do Fator de Forma F_{di-j} corresponde a projetar as partes de A_j visíveis de dA_i num hemisfério centrado em dA_i , projetando depois esta projeção de forma ortográfica na base do hemisfério e dividindo pela área do círculo. (Analogia de Nusselt)

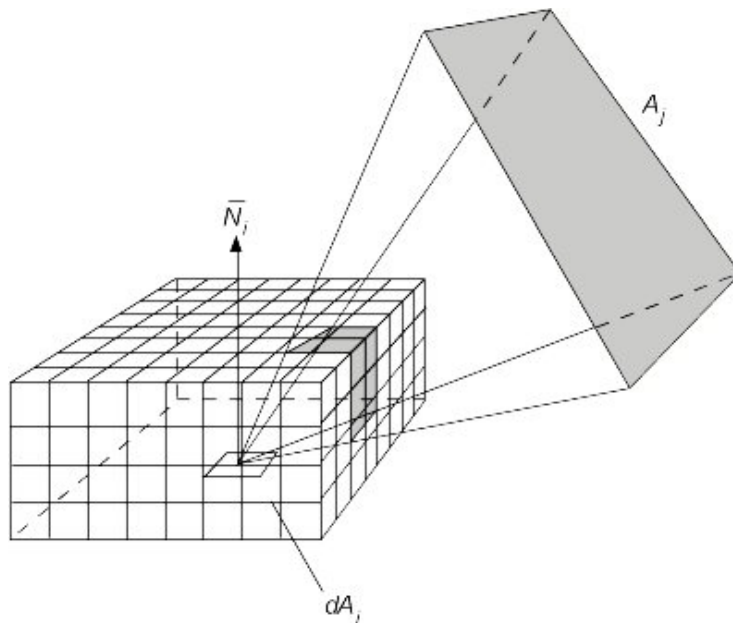
O cálculo é complexo.

Algoritmos de Iluminação Global

Radiosity

Simplificação de Cohen e Greenberg: método do hemicubo

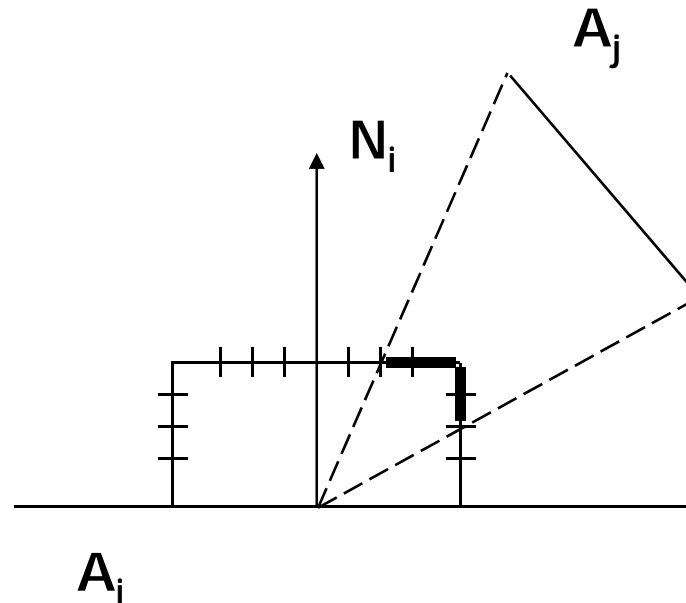
Em vez de usar a projeção num hemisfério, projeta na parte superior de um cubo centrado em dA_i , sendo a parte superior do cubo paralela com a superfície.



Cada face do hemicubo é dividida num conjunto de células quadradas de igual dimensão (ex: 50 por 50)

Algoritmos de Iluminação Global

Radiosity

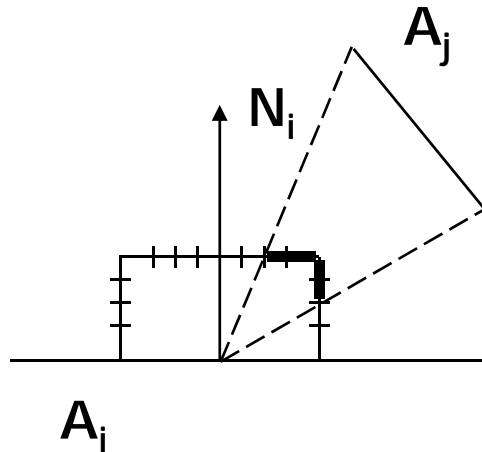


Project A_j no hemicubo, registrando os quadrados (mini-patch) que são cobertos.
Para cada quadrado registrar quais as **patches visíveis** A_j e a sua distância.

Guardar apenas a patch mais próxima, uma vez que as outras serão invisíveis: usar algoritmo de visibilidade no espaço imagem, eventualmente (normalmente...) o Z-Buffer!

Algoritmos de Iluminação Global

Radiosity



São calculados fatores de forma elementares F_q para cada quadrado/célula do hemicubo q .

O fator de forma F_{i-j} é então obtido somando todas as contribuições das células cobertas pelo patch j .

$$F_{i-j} = \sum F_q$$

Problemas do algoritmo de radiosidade:

- *Algoritmo computacionalmente pesado em processamento e utilização de memória.*
- *Para obter precisão é necessária a divisão dos objetos em patches de pequena dimensão. Implica N^2 fatores de forma para calcular.*

Radiosity



Radiosity

- *Progressive Refinement Radiosity*
 - Resolução do sistema de equações lineares...
 - Métodos iterativos com convergência para a solução final
 - Aproveitamento dos resultados intermédios como sendo “provisórios”
 - Imagem é apresentada desde o início dos cálculos
 - Usando os resultados intermédios
 - Qualidade dos resultados vai melhorando com o tempo de processamento
- *Junção Ray-tracing + Radiosity*
 - Exploração do que cada um processa melhor...
 - Ray-Tracing: reflexão especular
 - Radiosity: reflexão difusa