



Cap. 7

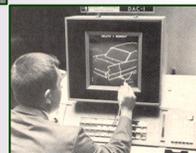
Coloração (Shading) e

Iluminação Global



Engenharia Informática (5385)

- 2º ano, 2º semestre

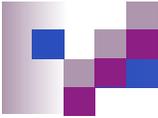




Revisão sobre Modelos de Iluminação Dependentes da Luz

■ Modelos dependentes da luz

- Ambiente
 - Normais à superfície não são importantes
- Lambert/Difusa
 - Ângulo entre a normal à superfície e o vector director da fonte de luz
- Phong/Especular
 - Normal à superfície, fonte de luz e, ainda, ponto de vista (observador)



Aplicação da Iluminação

- Temos agora um modelo de iluminação directa para um simples ponto na superfície
- Assumindo que a nossa superfície é definida como uma malha de faces poligonais, **que pontos devemos nós usar?**
 - Computar estes modelos para todos os pontos é dispendioso
 - As normais podem não estar explicitamente definidas para todos os pontos
- Há que ter em conta que:
 - A iluminação envolve um processo de cálculo bastante pesado se for aplicado a todos os pontos da superfície dum objecto
 - Há várias soluções possíveis, cada uma das quais tem implicações diferentes na qualidade visual da cena



Modelos de Coloração

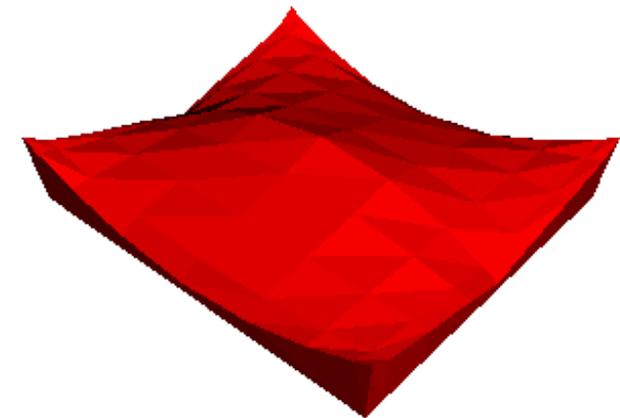
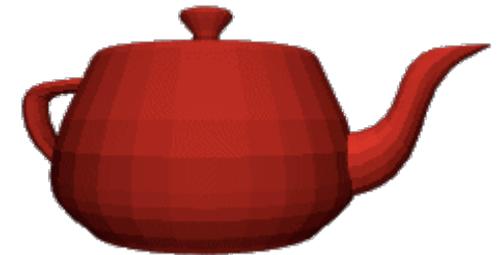
■ Várias opções:

- Coloração **constante** (*flat shading*)
- Coloração de **Gouraud** (interpolação)
- Coloração de **Phong** (interpolação)

■ Novo hardware gráfico faz coloração programável por pixel (*per-pixel programmable shading*)!

Coloração Constante

- É o método mais simples, pois calcula a iluminação num só ponto de cada polígono.
 - OpenGL usa um dos vértices do polígono
- A intensidade de iluminação (cor) é a mesma para todos os pontos de cada polígono.
- Vantagens:
 - Rápido - um valor de cor calculado por polígono
- Desvantagens:
 - Impreciso
 - Descontinuidades nas fronteiras dos polígonos



A coloração constante será realística para objectos facetados?

NÃO!

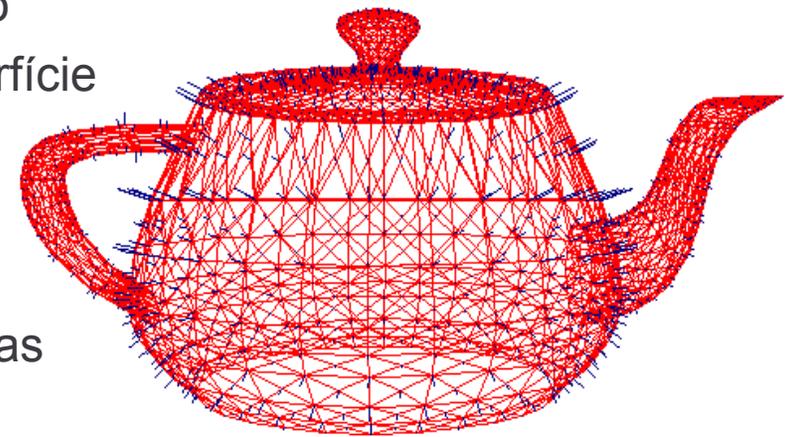
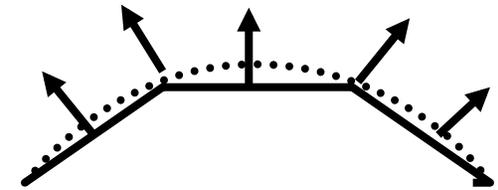
- Para fontes de luz pontuais, a direcção à fonte de luz varia para cada ponto da faceta.
- No caso da reflexão especular, a direcção ao observador varia para cada ponto da faceta.



Como tornar a coloração mais suave?

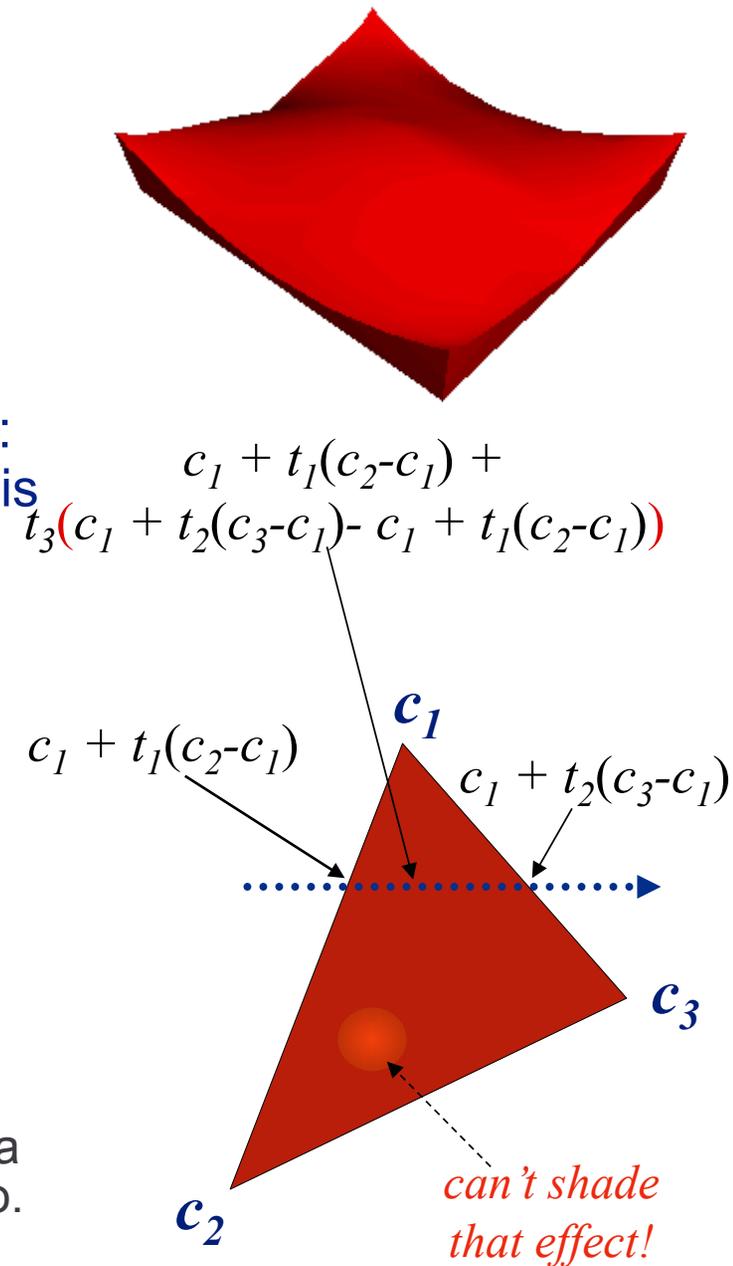


- Pode refinar-se um pouco este tipo de coloração através do cálculo da iluminação de Phong em cada pixel de cada polígono, mas o resultado continua a ser claramente facetado.
- Para obter superfícies visualmente suaves, há que usar **vectores normais** nos vértices dos polígonos
 - normalmente diferentes das normais às faces
 - são usados **somente** para efeitos de coloração
 - Faz pensar numa melhor aproximação à superfície **real** que os polígonos aproximam
- As normais aos vértices podem ser
 - fornecidas com o modelo
 - aproximadas pela média das normais às facetas que partilham cada vértice



Coloração de Gouraud

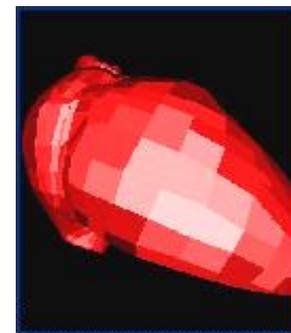
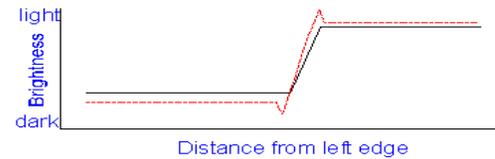
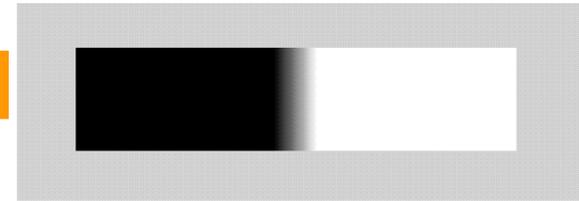
- Ilumina or colora directamente cada vértice pela utilização da sua posição e da sua normal.
- Faz interpolação linear das cores sobre as faces: primeiro, ao longo das arestas de fronteira; depois ao longo das *scanlines* dentro do seu interior.
- Vantagens:
 - Rapidez - cálculos incrementais aquando da rasterização
 - Maior suavidade de cor- usa 1 normal por vértice partilhado de forma a obter continuidade de cor entre faces
- Desvantagens:
 - Ainda impreciso. Polígonos parecem bacentos e pouco brilhantes.
 - Tende a eliminar a componente especular. Se esta for incluída, será ponderada sobre todo o polígono.
 - Mach banding.



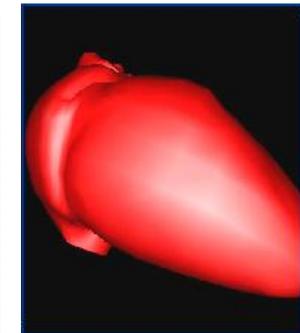
Coloração de Gouraud

Mach banding

- Artificial nas discontinuidades da intensidade da luz.
- O efeito de Mach banding descreve como a mente humana aumenta o contraste subconscientemente entre duas superfícies com luminâncias diferentes.
- A diferença entre duas cores é mais pronunciada quando elas estão lado-a-lado e a fronteira é suave.
- Isto realça as fronteiras entre cores, mesmo se a diferença de cor é pequena.
- Fronteiras grosseiras são “ponderadas” pelo nosso sistema de visão por forma a fornecer uma variação suave.



flat shading



Gouraud shading

*banded
along
edges*



floor appears banded



Coloração em OpenGL

■ A OpenGL define dois modelos de coloração:

□ Controlam como as cores são atribuídas aos *pixels*

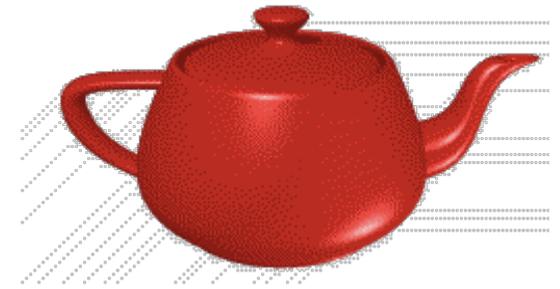
□ **Coloração constante:** usa uma cor *constante* em todo o polígono

```
glShadeModel(GL_FLAT)
```

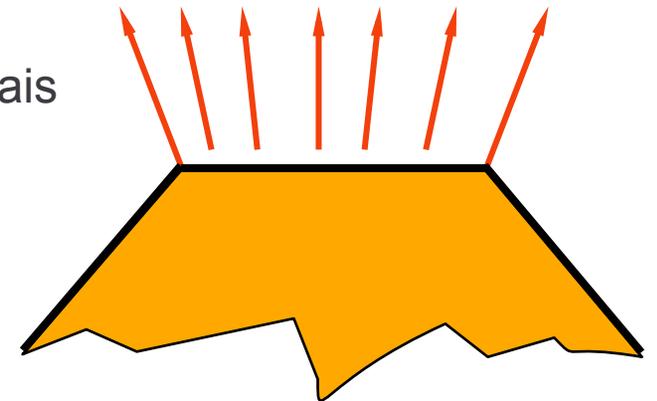
□ **Coloração de Gouraud:** *interpola* as cores entre os *vértices* (coloração por defeito)

```
glShadeModel(GL_SMOOTH)
```

Coloração de Phong



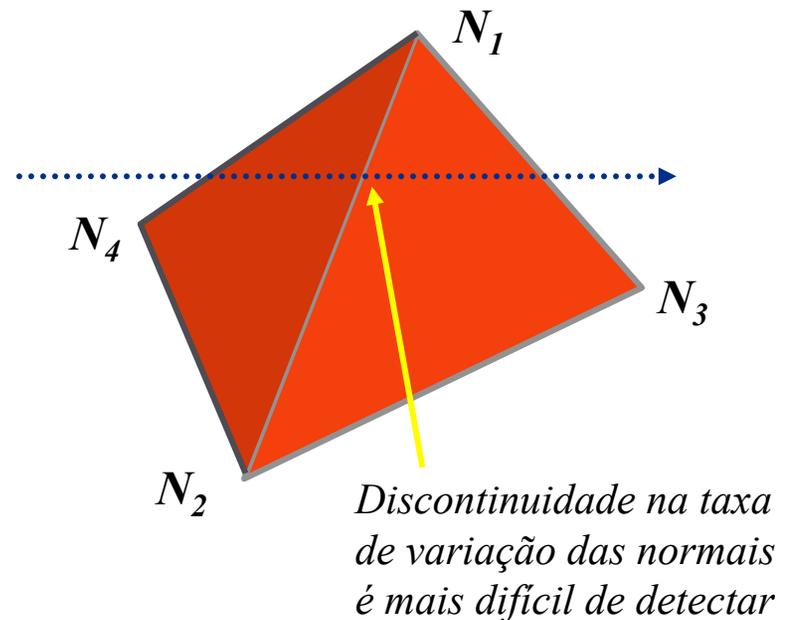
- *Coloração de Phong* não é o mesmo que iluminação de Phong, embora sejam conceitos às vezes confundidos um com o outro
 - **Iluminação de Phong:** o modelo empírico que discutimos no capítulo anterior para calcular a iluminação num ponto da superfície dum objecto.
 - **Coloração de Phong:** interpola linearmente as normais em toda e qualquer faceta, aplicando o modelo de iluminação de Phong em cada pixel
- **Vantagens:**
 - Obtém-se resultados visualmente muito suaves
 - Elevada qualidade, especularidades estreitas
- **Desvantagens:**
 - Mas é bastante mais dispendioso nos cálculos
 - Continua a ser uma aproximação para a maior parte das superfícies



Coloração de Phong

■ Interpola linearmente as normais aos vértices

- Calcula as equações de iluminação em cada pixel
- Pode usar a componente especular
- Note-se que as normais são usadas para computar as componentes difusa e especular



$$I_{total} = K_A I_A + \sum_{i=1}^{\# lights} I_i (K_D (\vec{N} \cdot \vec{L}_i) + K_S (\vec{V} \cdot \vec{R}_i)^n)$$

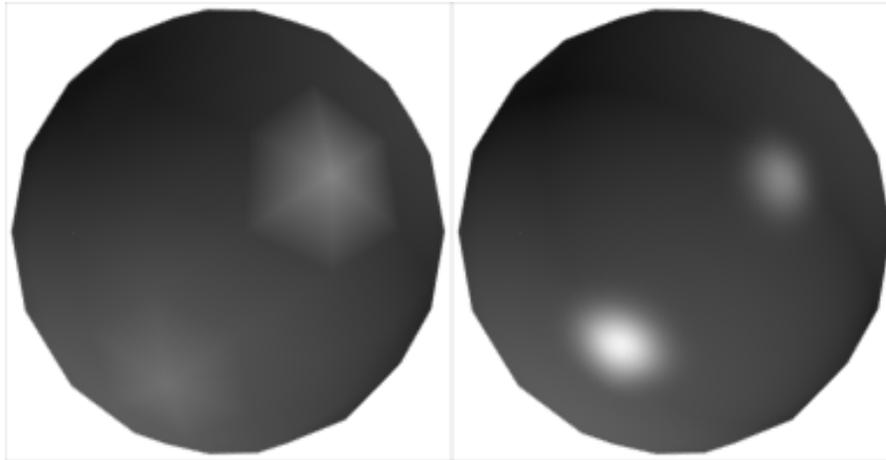


Problemas na Coloração

- Silhuetas poligonais continuam presentes
- Distorção em perspectiva
- Interpolação dependente da orientação dos pológonos
- Problemas nos vértices partilhados
- Má média nos vértices

Problemas na Coloração (1)

- Presença de silhuetas poligonais



Gouraud

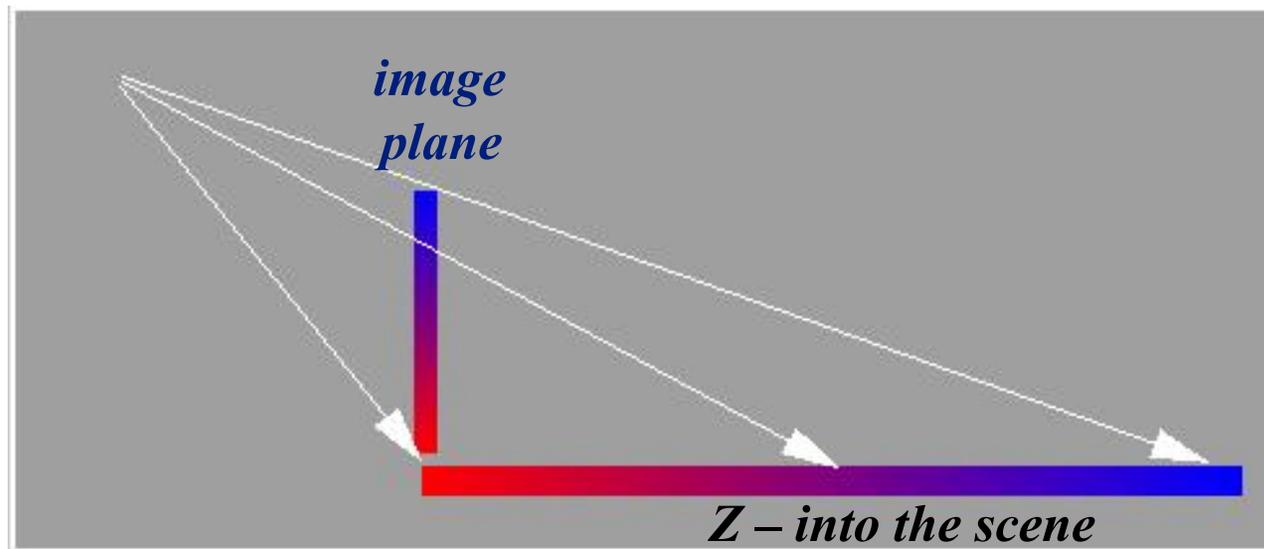
Phong

Problemas na Coloração

(2)

■ Distorção em perspectiva

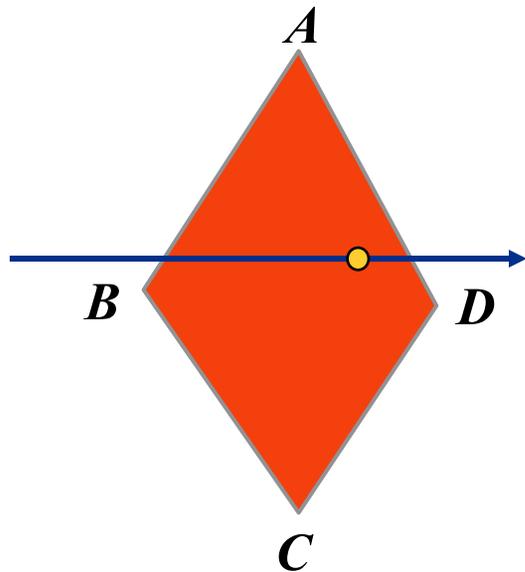
- Note-se que a interpolação linear no espaço de ecrã não está alinhada com a interpolação linear no espaço do domínio da cena.
- Particionam-se polígonos grandes em polígonos mais pequenos para reduzir a distorção.



Problemas na Coloração

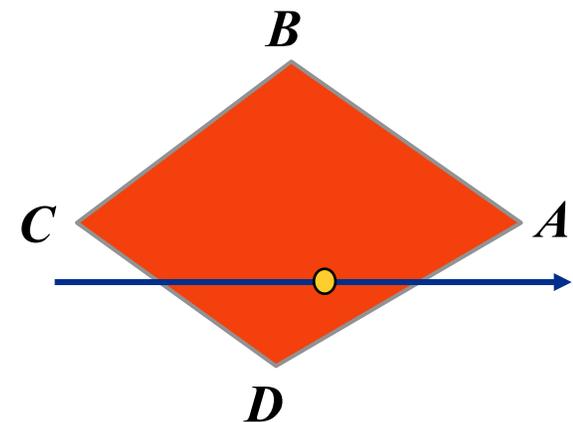
(3)

- Interpolação dependente da orientação dos polígonos



*Interpola entre
AB e AD*

*Roda -90°
e colora o
mesmo ponto*



*Interpola entre
CD e AD*

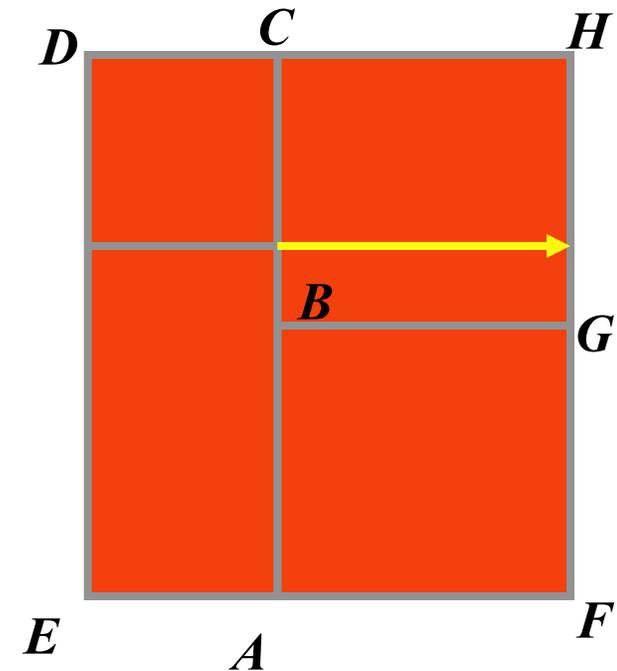
Problemas na Coloração

(4)

■ Problemas nos vértices partilhados

□ Exemplo ao lado:

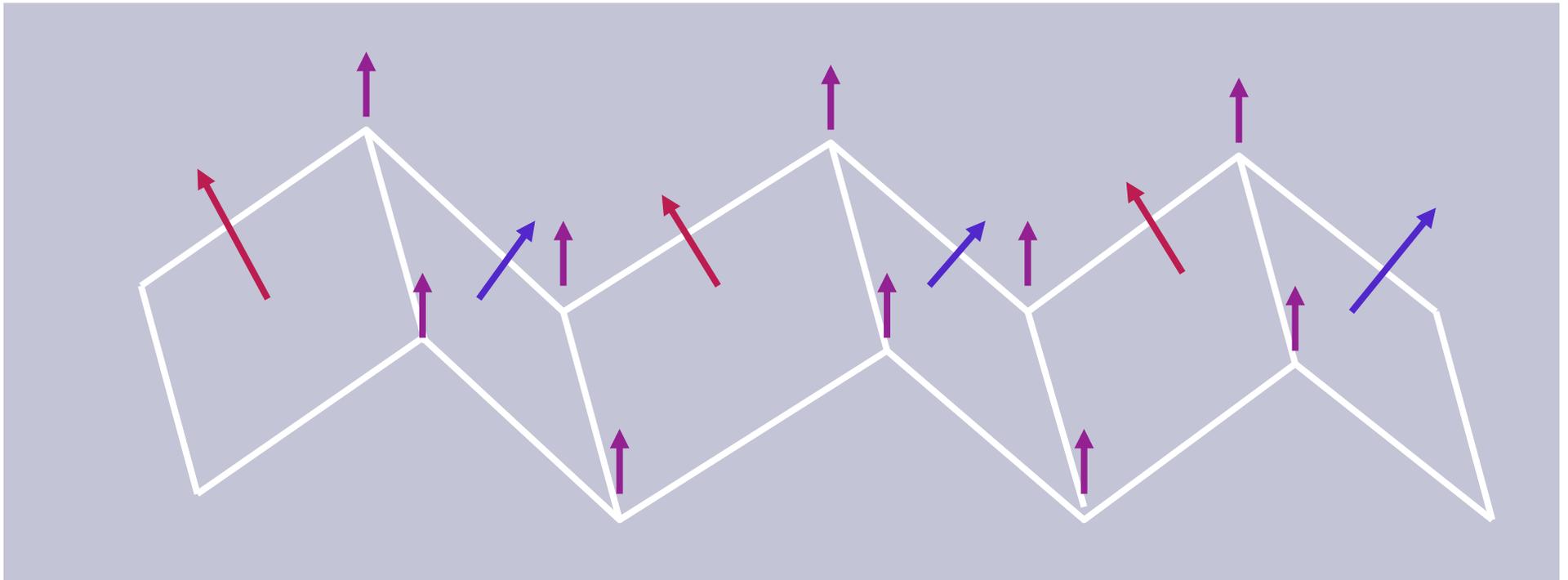
- O vértice B é partilhado por dois rectângulos à direita, mas não por aquele à esquerda
- O primeiro segmento da *scanline* é interpolada entre DE e AC
- O segundo segmento da *scanline* é interpolada entre BC e GH
- Uma discontinuidade pode surgir



Problemas na Coloração

(5)

- Má média nos vértices





Modelos de Coloração (por Iluminação Directa)

Sumário:

- Coloração Constante
 - Computa-se a iluminação de Phong uma só vez para cada polígono
- Coloração de Gouraud
 - Computa-se a iluminação de Phong nos vértices e interpola os valores de iluminação sobre o polígono
- Coloração de Phong
 - Computa-se as médias das normais das faces nos vértices
 - Interpola-se as normais sobre a face e faz-se a iluminação de Phong sobre ela



Geração Actual de Coloradores

- O hardware actual permite-nos ir para além do modelo de iluminação estandardizado.
- Os *Coloradores de Vértices Programáveis* permitem-nos escrever um pequeno programa que determina como é que a cor dum vértice é calculada
 - O nosso programa tem acesso à normal e à posição da superfície, assim como algo mais que lhe queiramos fornecer (como a luz)
 - Podemos adicionar, subtrair, calcular produtos internos, etc



Geração Actual de Coloradores

(1)

- Só abordámos superficialmente a iluminação de superfícies
 - O modelo comum é claramente inadequado para iluminação precisa, mas tem a vantagem de ser rápido e simples
- Tem em conta dois sub-problemas de iluminação
 - Para onde vai a luz? *Transporte de Luz*
 - O que acontece nas superfícies? *Modelos de Reflexão*
- Outros algoritmos endereçam o transporte ou a reflexão, ou ambos
 - A abordar talvez no mestrado...

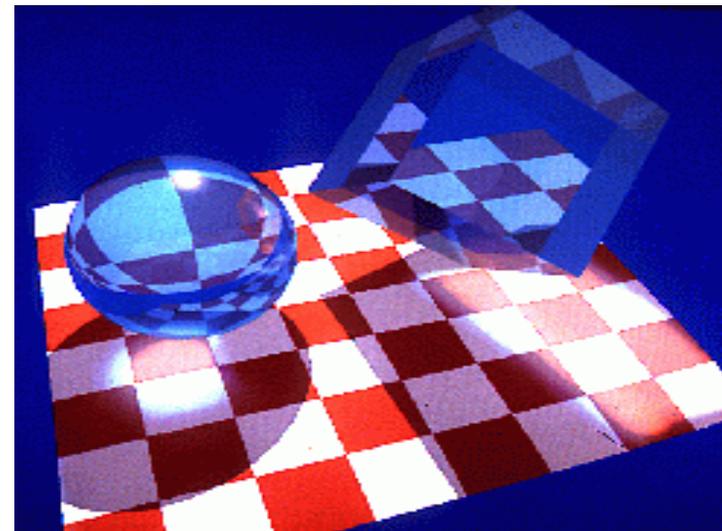
Sobrevisão: modelos baseados na iluminação

■ Iluminação Directa ou Local

- Tipos de Luz
- Fontes de Luz (emissão)
- Materiais da superfície dos objectos (reflexão)

■ Iluminação Indirecta ou Global

- Sombras
- Refracções
- Reflexões entre objectos



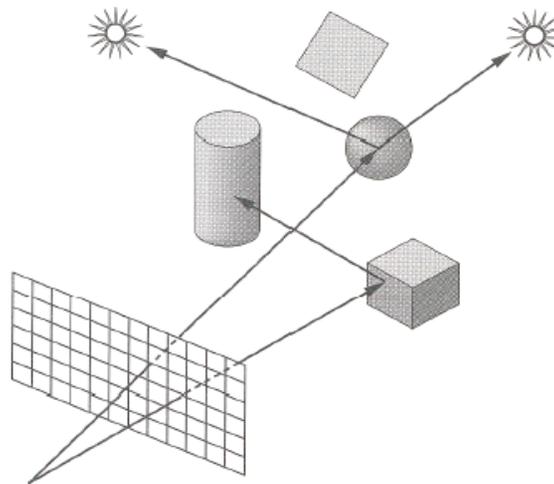


Iluminação Global

- Temos visto dum forma aproximada como é que a luz funciona
- E assim continuaremos...
- Mas daremos agora um passo mais para a frente
- Iluminação global
 - A noção de que um ponto é iluminado por mais luz do que aquela que vem das fontes de luz locais; a iluminação é feita por todos os emissores e reflectores na cena

Sombras (shadows)

- Os termos relativos a sombras dizem-nos que fontes de luz estão obstruídas
 - Emite-se raios visuais em direcção a cada fonte de luz L_i
 - $S_i = 0$ se o raio é obstruído, $S_i = 1$ no caso contrário

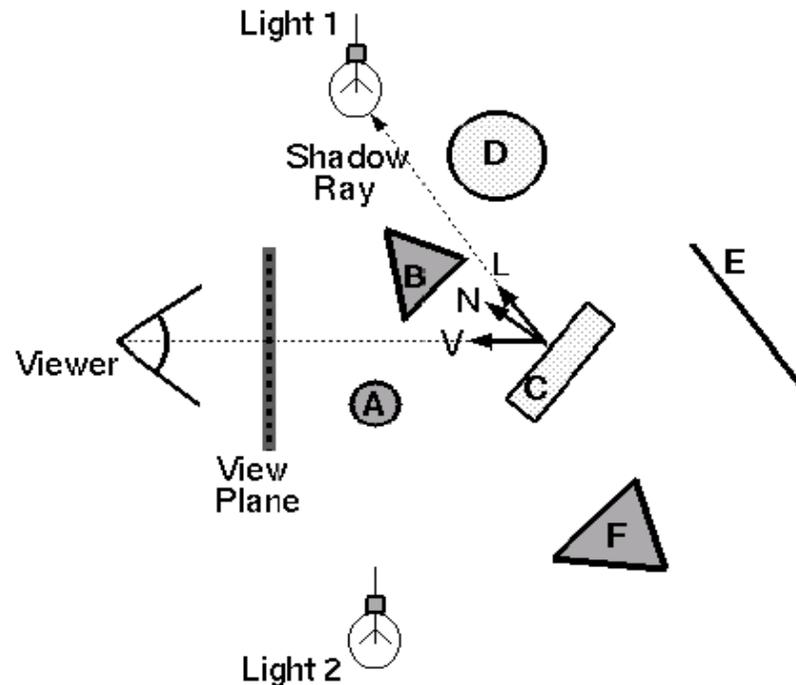


Shadow
Term

$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{ lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i$$

Emissão de Raios Visuais (Ray Casting)

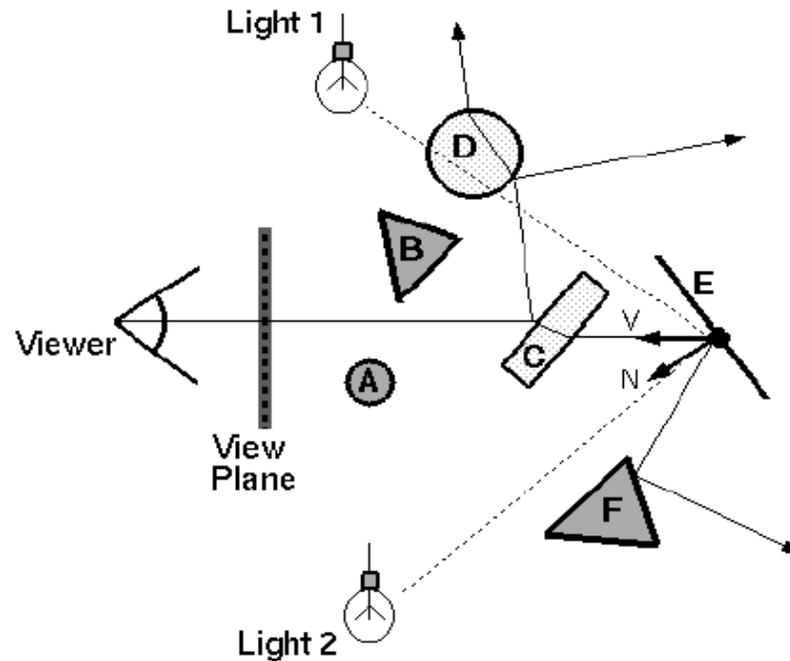
- Traçar raios primários a partir da câmara
 - Iluminação directa a partir de luzes não obstruídas somente



$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{ lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i$$

Traçagem Recursiva de Raios (Recursive Ray Tracing)

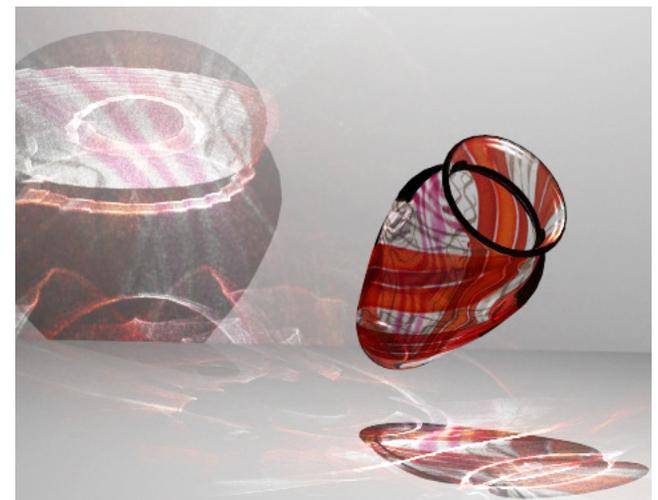
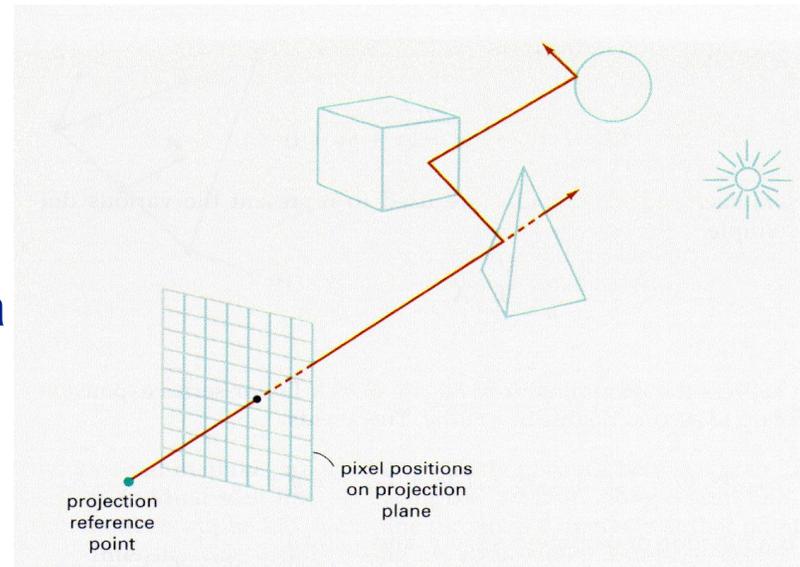
- Também traça **raios secundários** a partir das superfícies atingidas
 - Iluminação global a partir de **reflexão simétrica e transparência**



$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i + K_S I_R + K_T I_T$$

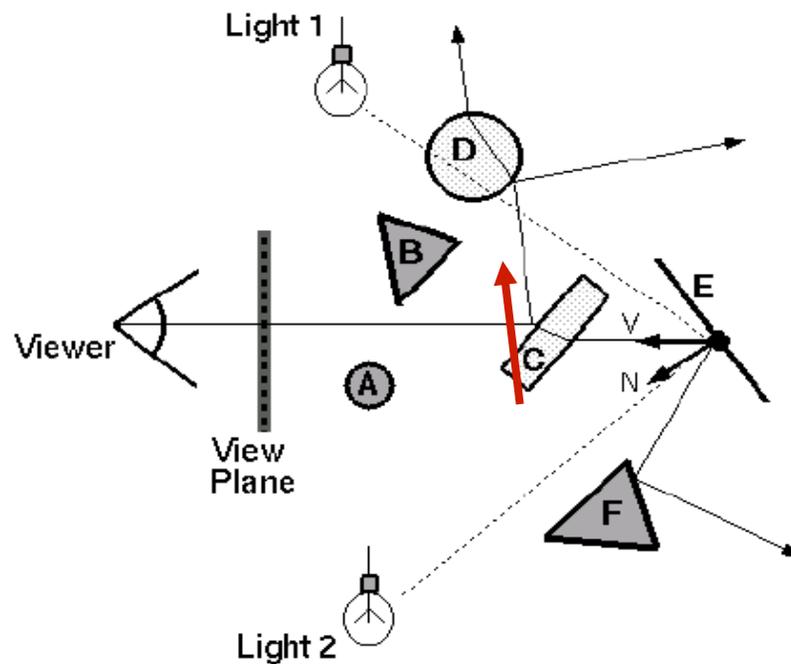
Traçagem Recursiva de Raios: sobrevisão

- **Raios primários.** Emissão dum raio a partir do olho do observador através de cada pixel, e depois a partir do objecto intersectado para as fontes de luz de forma determinar as condições de sombreamento/iluminação
- **Raios secundários.** Ray tracing também gera raios secundários
 - Raios de reflexão e raios de refração
 - Usa normais à superfície como guia (ângulo de incidência é igual ao ângulo de reflexão)
 - Se outro objecto é atingido, determina-se a luz que ele ilumina recursivamente por ray tracing
- **A recursão pára quando:**
 - Raio falha a intersecção com um objecto
 - A profundidade máxima especificada pelo utilizador é atingida
 - A máquina fica sem memória disponível



Reflexão simétrica (mirror reflection)

- Traça raio secundário na direção da reflexão simétrica (mirror reflection)
 - Calcula a radiância ao longo do raio secundário, incluindo-a no modelo de iluminação

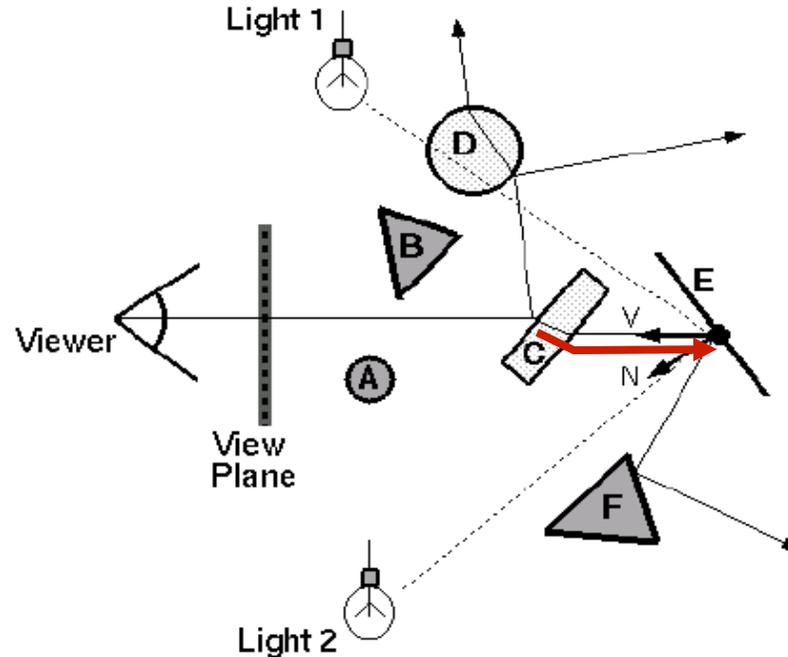


*Radiance
for mirror
reflection ray*

$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{ lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i + K_S I_R + K_T I_T$$

Transparência

- Traça raio secundário na direcção da refração
 - Calcula a radiância ao longo do raio secundário, incluindo-a no modelo de iluminação

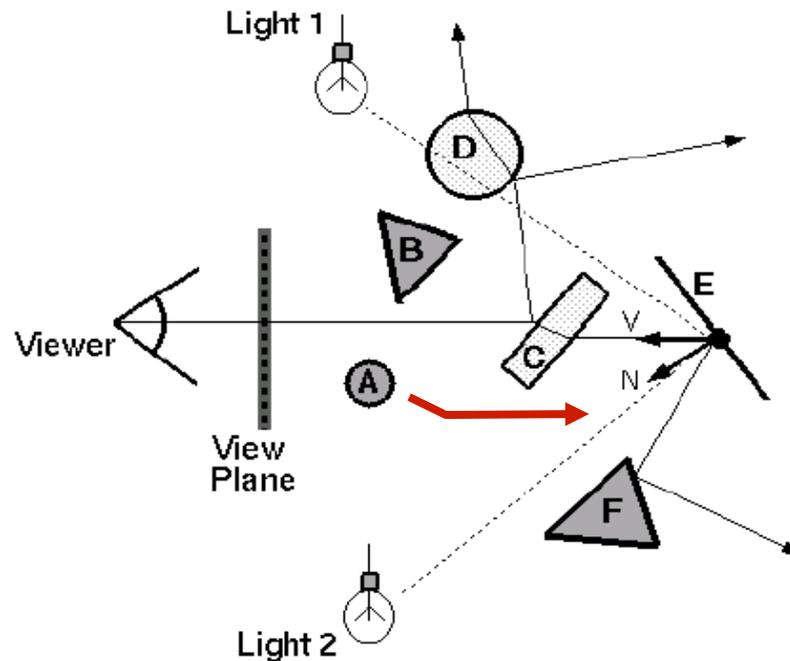


Radiance for refraction ray

$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{ lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i + K_S I_R + K_T I_T$$

Transparência

- O coeficiente de transparência é a fracção que é transmitida
 - $K_T = 1$ se objecto é translúcente, $K_T = 0$ se objecto é opaco
 - $0 < K_T < 1$ se objecto é semi-translúcente

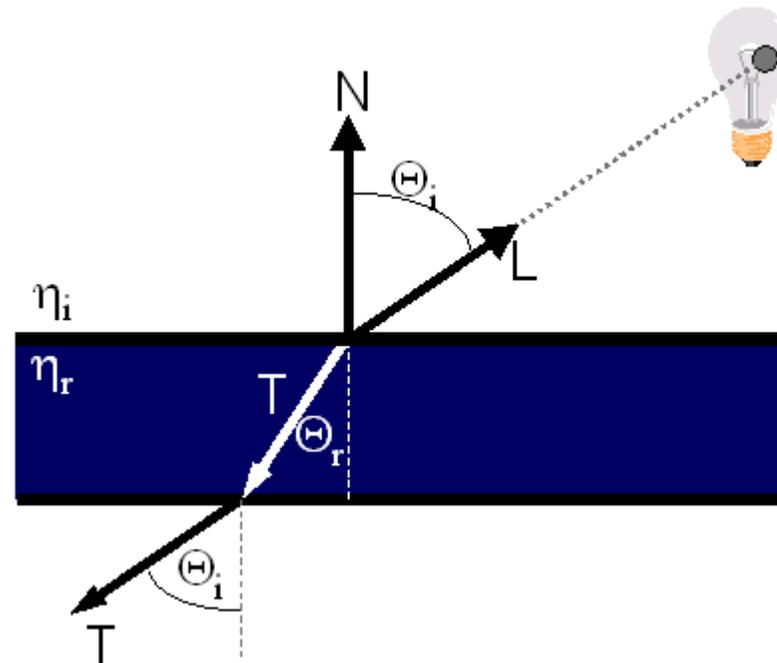


Transparency
Coefficient

$$I = I_E + K_A I_A + \sum_{i=1}^{\# \text{ lights}} [K_D (\vec{N} \cdot \vec{L}) + K_S (\vec{V} \cdot \vec{R})^n] S_i I_i + K_S I_R + K_T I_T$$

Transparência Refractiva

- Para superfícies finas, podemos ignorar a mudança de direcção
 - Assume-se que a luz viaja em linha recta através da superfície

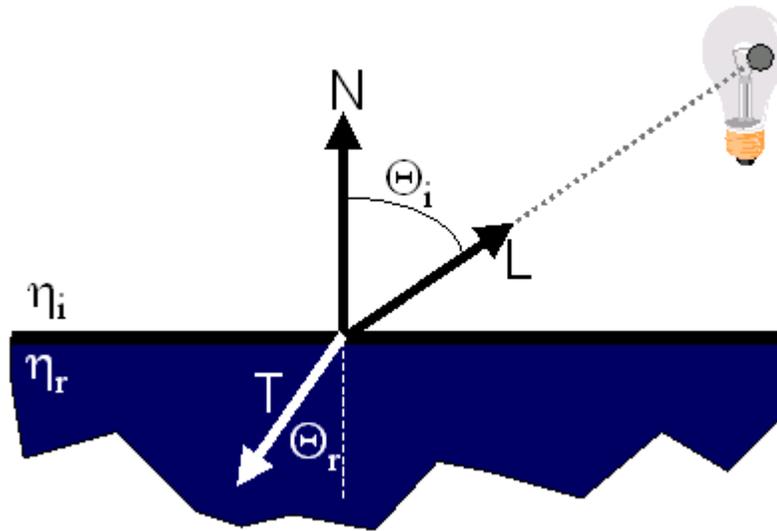


$$T \cong -L$$

Transparência Refractiva

■ Para objectos sólidos, aplica-se a Lei de Snell:

□ $\eta_r \sin \Theta_r = \eta_i \sin \Theta_i$



$$\mathbf{T} = \left(\frac{\eta_i}{\eta_r} \cos \Theta_i - \cos \Theta_r \right) \mathbf{N} - \frac{\eta_i}{\eta_r} \mathbf{L}$$

Radiosidade



- Ray tracing modela a reflexão especular e transparência refractiva, mas ainda usa uma componente ambiente nos efeitos de luz
- Radiosidade é a taxa à qual a energia é emitida ou reflectida pela superfície
- Pela conservação da energia da luz num volume, estes efeitos de radiosidade podem ser traçados





Sumário

- Sombreamento baseado na Iluminação Directa
 - Emissão de Raios (Ray casting)
 - Usualmente usa-se aproximações analíticas simples para a emissão das fontes de luz e para a reflexão nas superfícies

- Sombreamento baseado na Iluminação Indirecta
 - Traçagem recursiva de raios (Recursive ray tracing)
 - Incorpora sombras, reflexões simétricas e refrações
 - Radiosidade
 - Usa lei da conservação de energia