

# Colorização

# Colorização

- A função de iluminação produz valores de cor para qualquer ponto de uma superfície no espaço da cena.
- Durante o processo de rasterização, as cores dos pixels devem ser determinadas a partir da projeção dos objetos da cena sobre a tela virtual.

$$I_c(x, y) = I(x', y', z')$$

# Tipos de Colorização

- O processo de colorização interpola os valores da função de colorização nos vértices dos polígonos para os pixels no interior do polígono.
  - ◆ Aumenta a velocidade de renderização.
- Assim, existem três tipos principais de colorização:
  - ◆ Constante (*flat*)
  - ◆ Gouraud
  - ◆ Phong

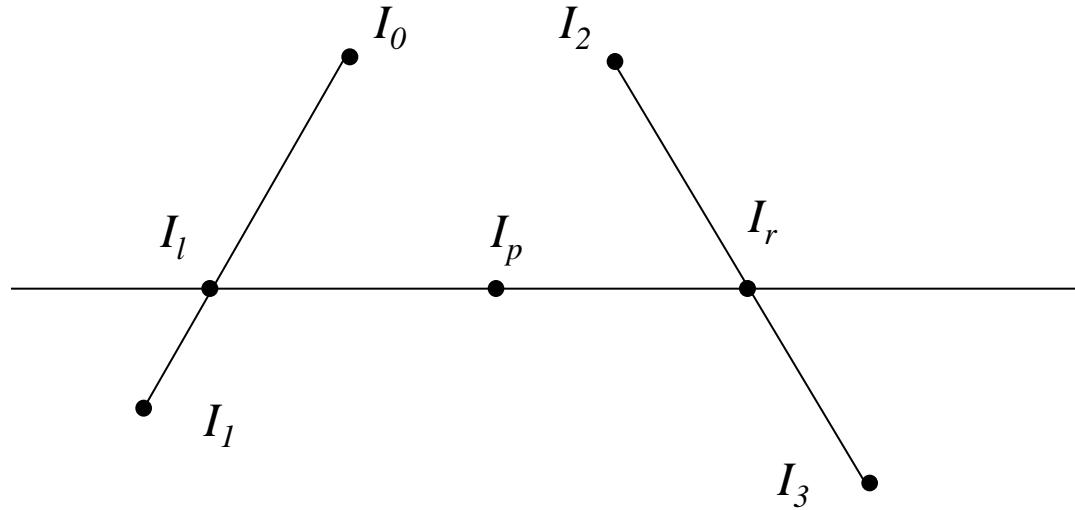
# Colorização Constante

- Assume que:
  - ◆ A fonte de luz está no infinito.
  - ◆ Cada polígono possui uma cor constante, dada pela sua normal.
- Extremamente rápida, mas produz imagens facetadas, pois a transição de um polígono para outro adjacente, é marcante.

# Colorização de Gouraud

- Em geral, um modelo B-rep é apenas uma aproximação poligonal de um objeto.
- Supõem que cada vértice do modelo possui uma normal associada.
- Intensidades nos vértices são interpoladas.
  - ◆ O que acontece com os *highlights* de Phong?
- Despreza a variação da normal no interior dos polígonos.
- Silhueta ainda parece estar linearizada.

# Interpolação Poligonal



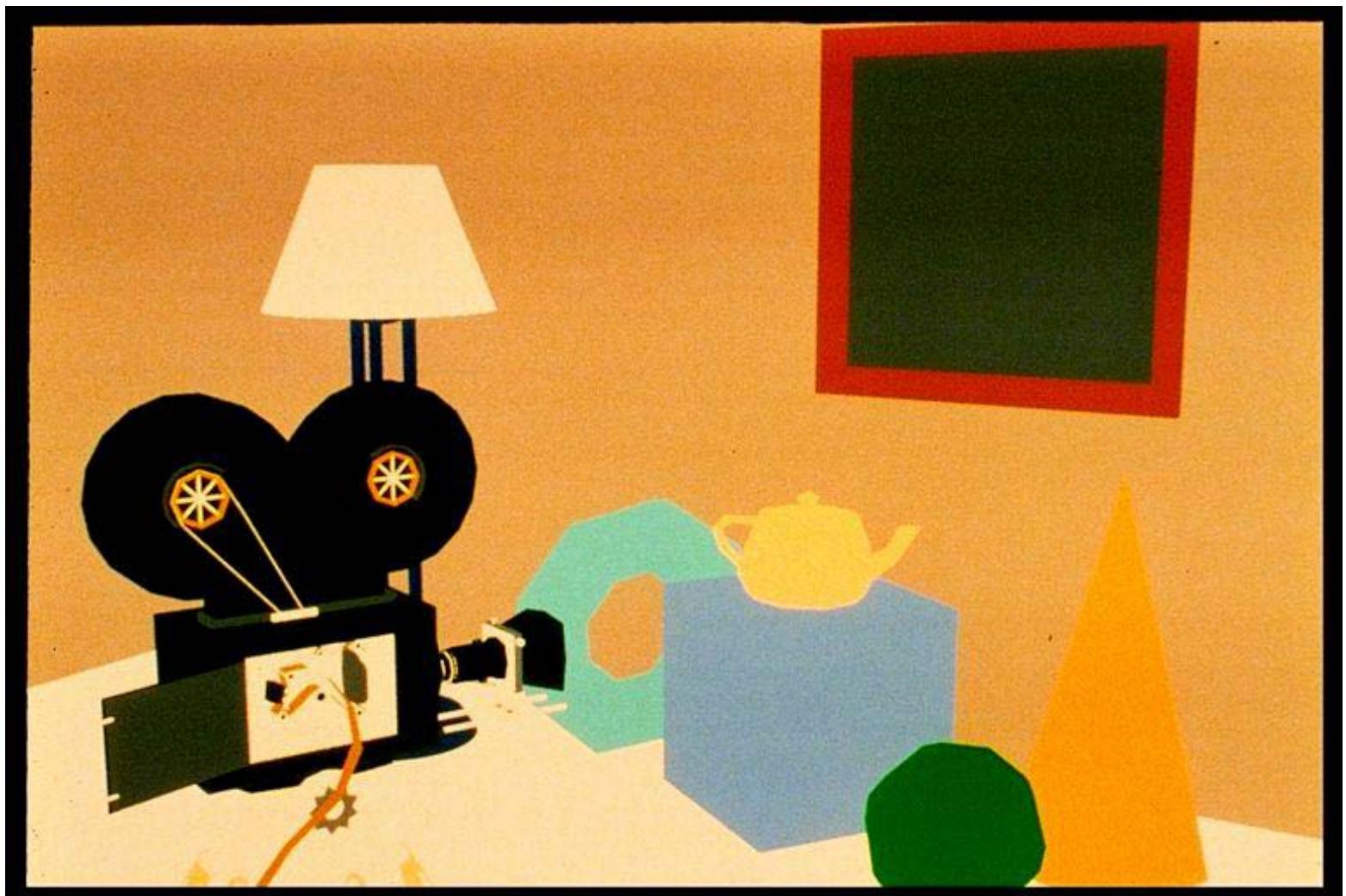
$$I_l = \left( \frac{y - y_0}{y_1 - y_0} \right) (I_1 - I_0) + I_0 \quad I_r = \left( \frac{y - y_2}{y_3 - y_2} \right) (I_3 - I_2) + I_2$$

$$I_p = \left( \frac{x - x_l}{x_r - x_l} \right) (I_r - I_l) + I_l$$

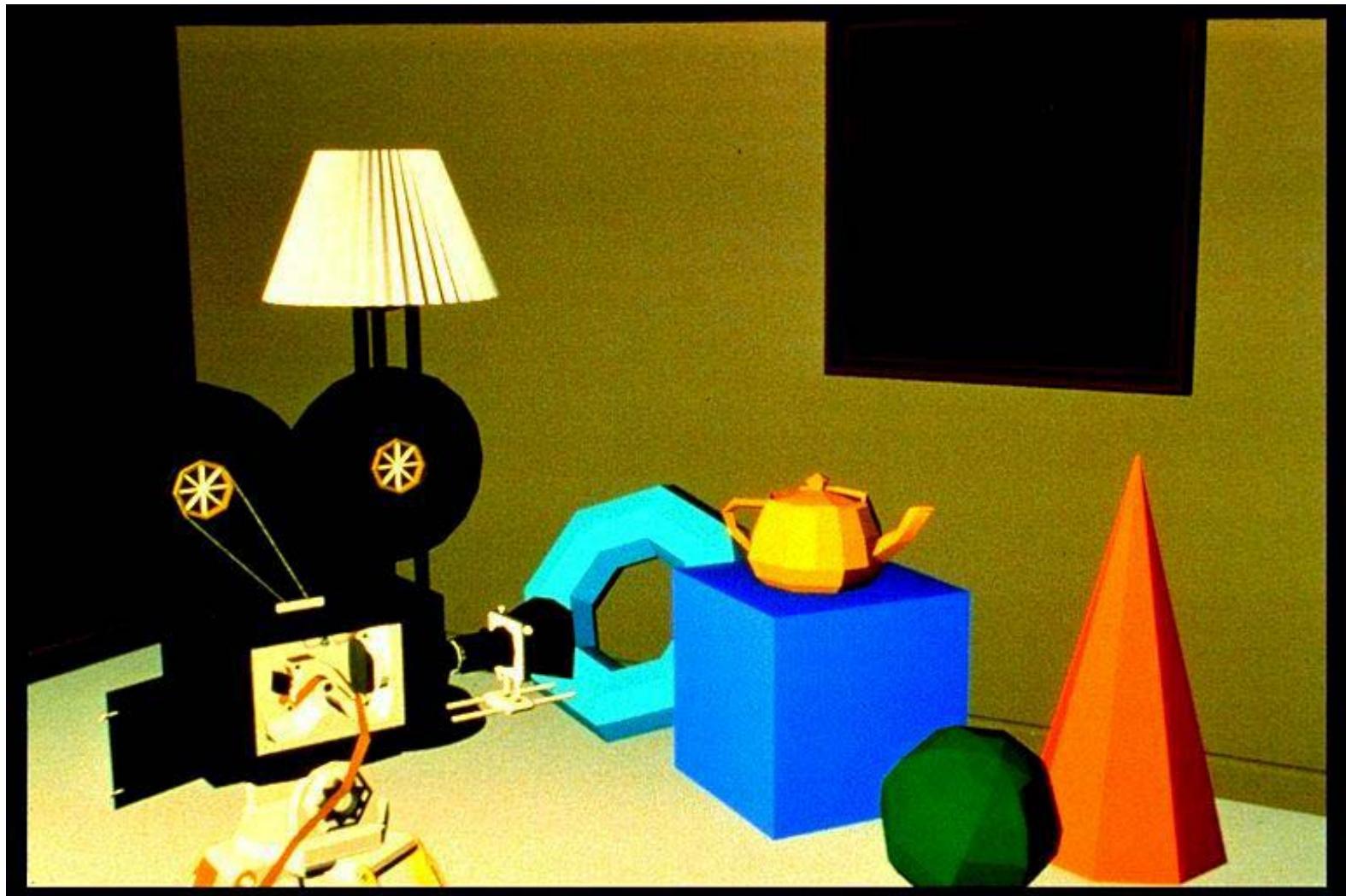
# Colorização de Phong

- Interpola as normais nos vértices.
- A função de iluminação deve ser avaliada para cada pixel.
- Transformações perspectiva não preservam ângulos.
  - ◆ Logo, a função de iluminação deve ser avaliada no espaço da cena ou num espaço isométrico a ele.
  - ◆ Usa-se a inversa da transformação de visualização para voltar ao espaço da cena.

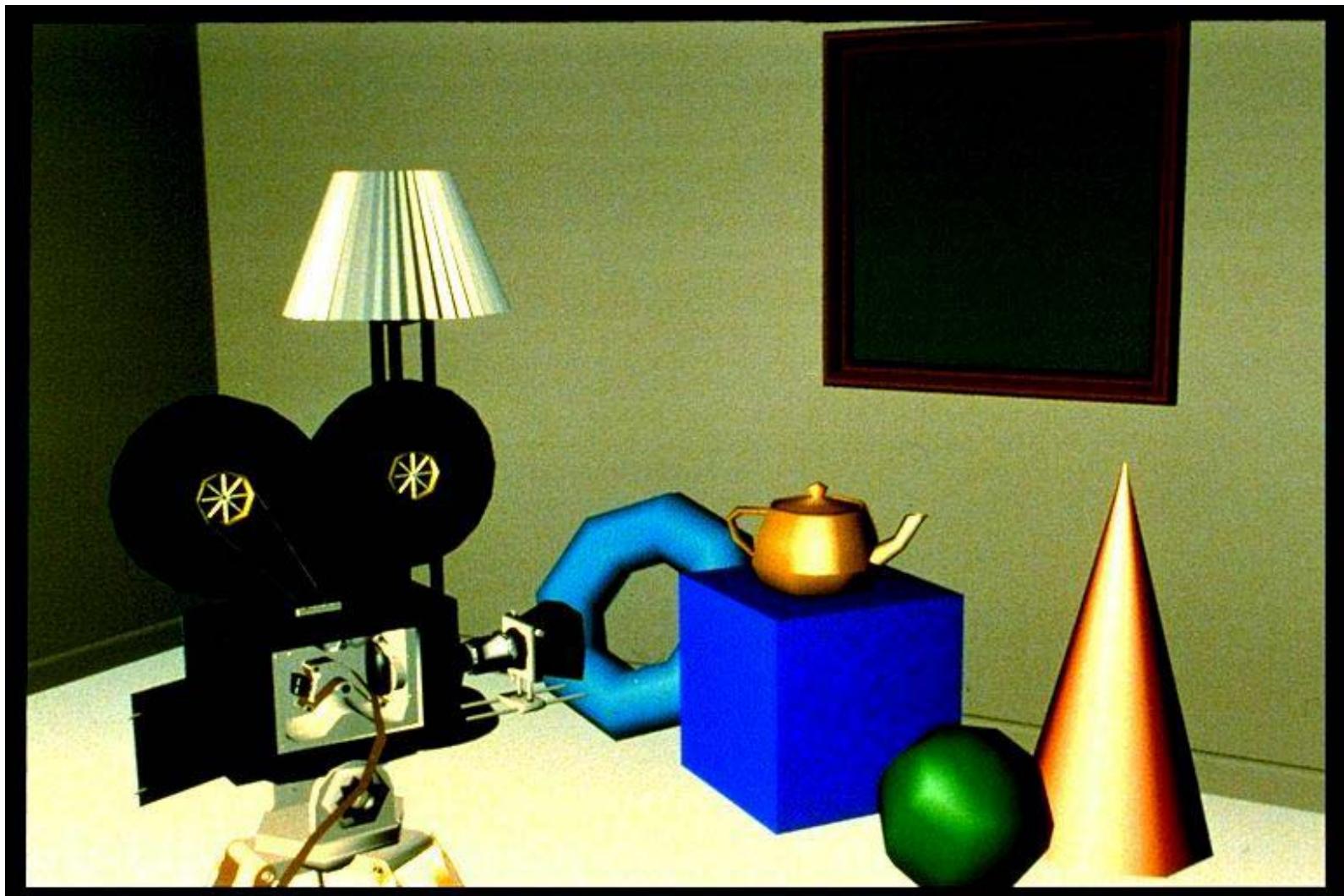
# Sem Iluminação



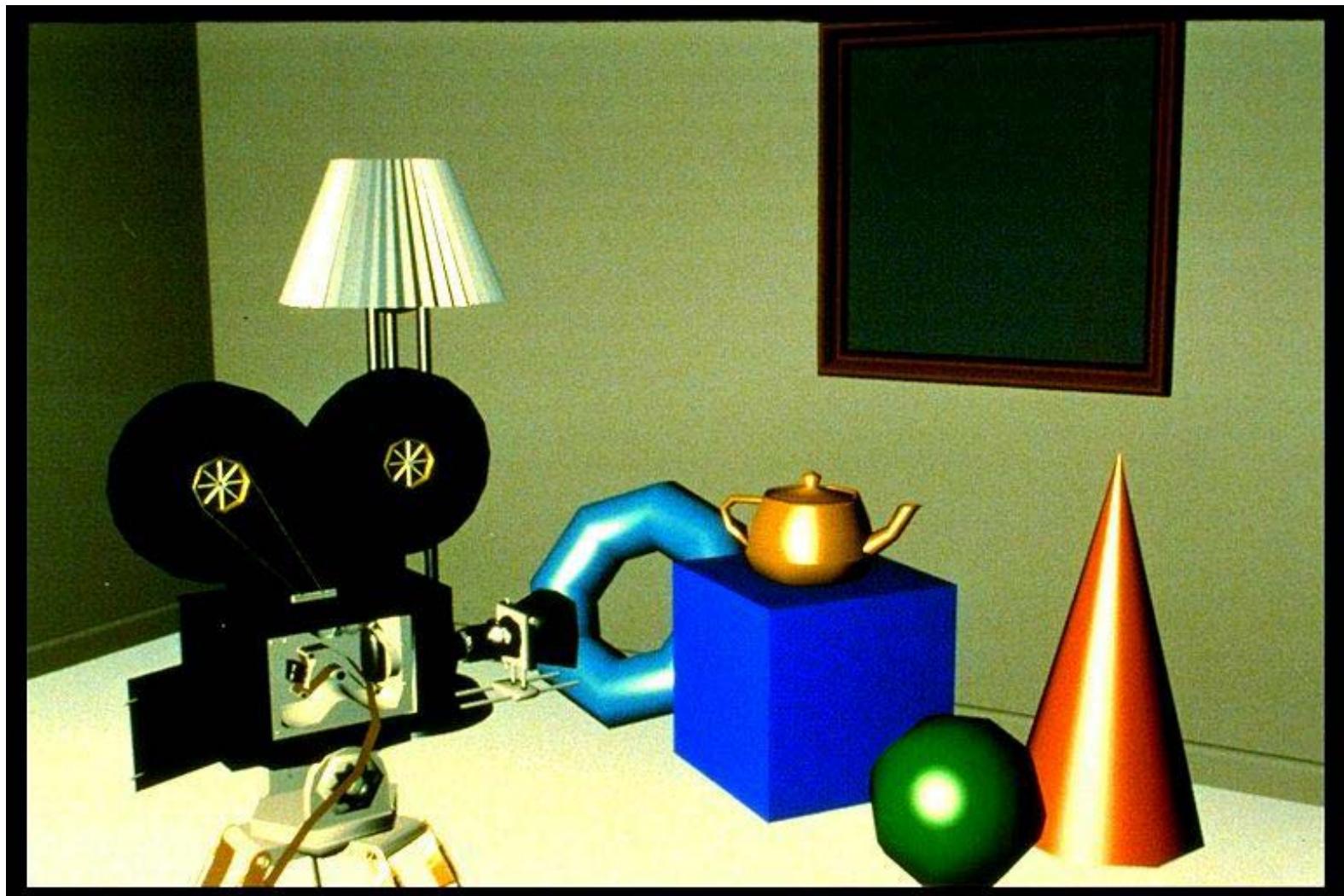
# Flat Shading



# Interpolação de Gouraud

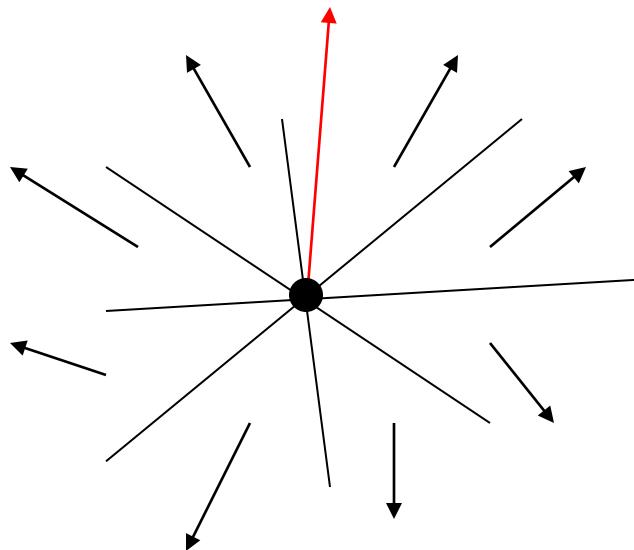


# Interpolação de Phong - Por pixel



# Estimando Normais

- As normais nos vértices podem ser estimadas acumulando-se as normais de todos os polígonos incidentes em cada vértice.
  - ◆ A circulação dos polígonos deve ser consistente.



# Interpolação

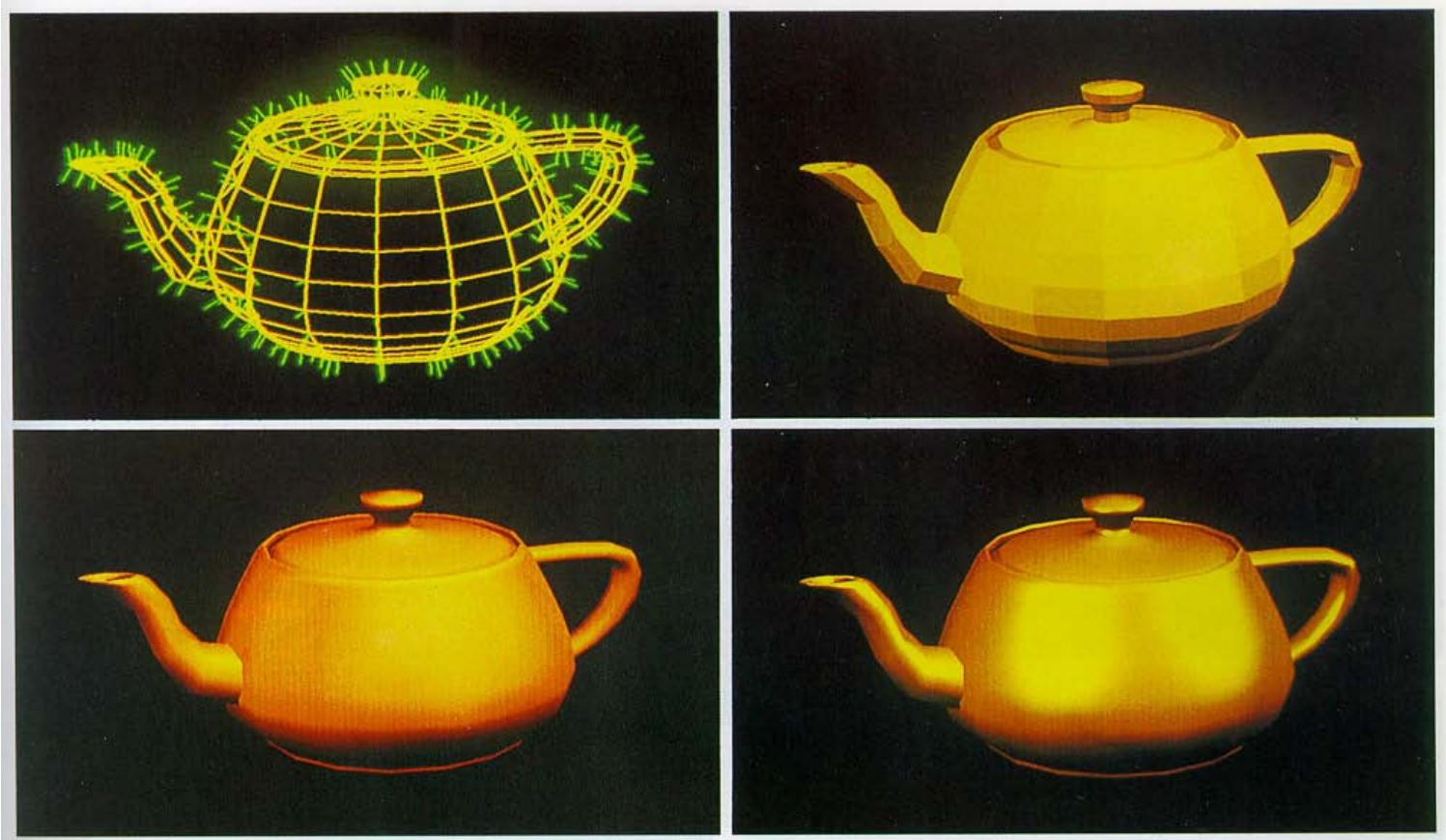


Image courtesy of Watt & Watt, Advanced Animation and Rendering Techniques

# Considerações Finais

- Interpolação poligonal só independe da orientação do polígono para o caso de triângulos.
  - ◆ Depende da orientação do polígono em relação à linha de varredura.
  - ◆ Cria efeitos indesejáveis em animações.
- Rasterização é executada no espaço normalizado, após a transformação perspectiva.
  - ◆ Incrementos entre linhas de varredura consecutivas são considerados constantes na interpolação poligonal.
  - ◆ Na realidade, eles aumentam na direção Z, quando caminha-se na direção do centro de projeção.

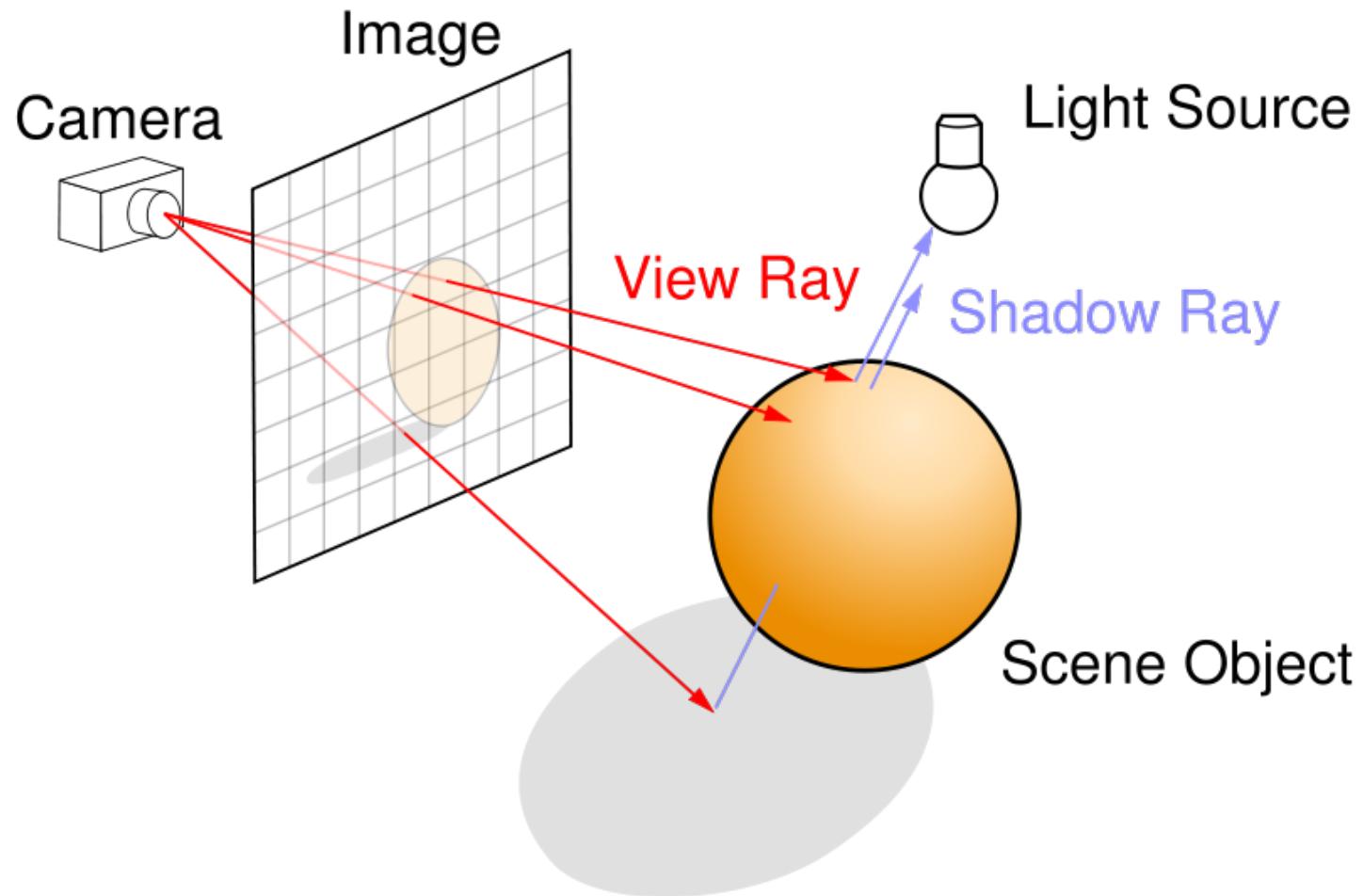
# Traçadores de Raios

- Executam de forma integrada, a projeção, visibilidade, iluminação e rasterização.
- Integram o fluxo de energia ao longo de um número finito de raios.
- Produzem imagens bastante realistas, se forem implementados de forma distribuída.
  - ◆ Utiliza super-amostragem, com distribuição aleatória, para diminuir *aliasing*.

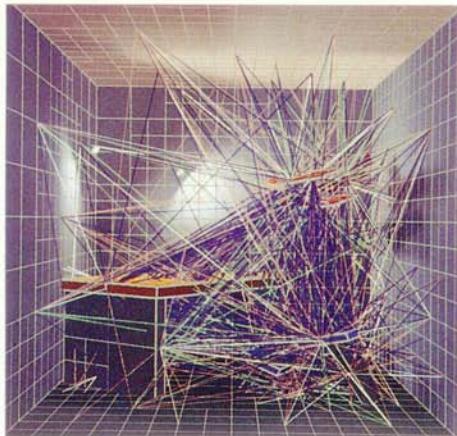
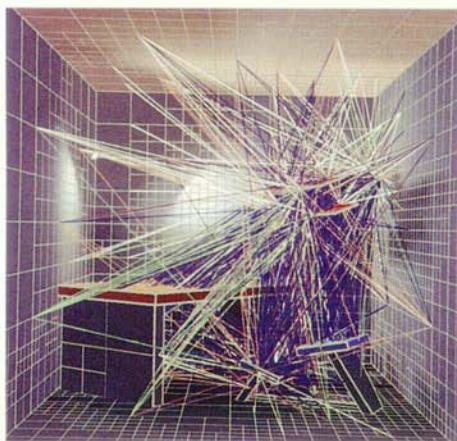
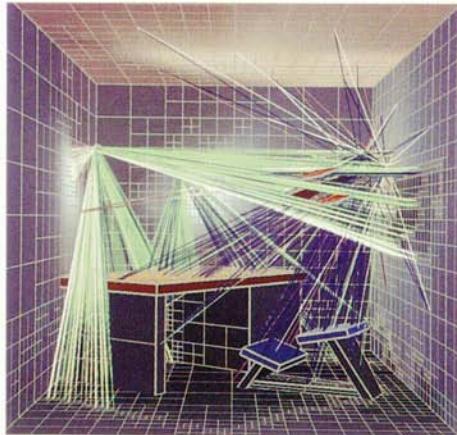
# Parece Real?



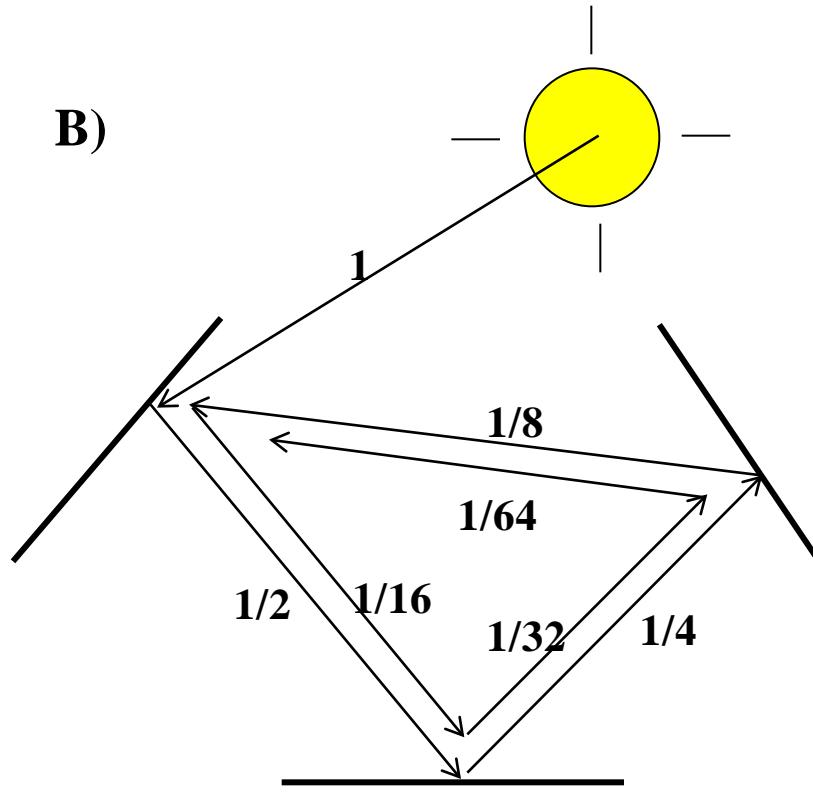
# Princípio Básico



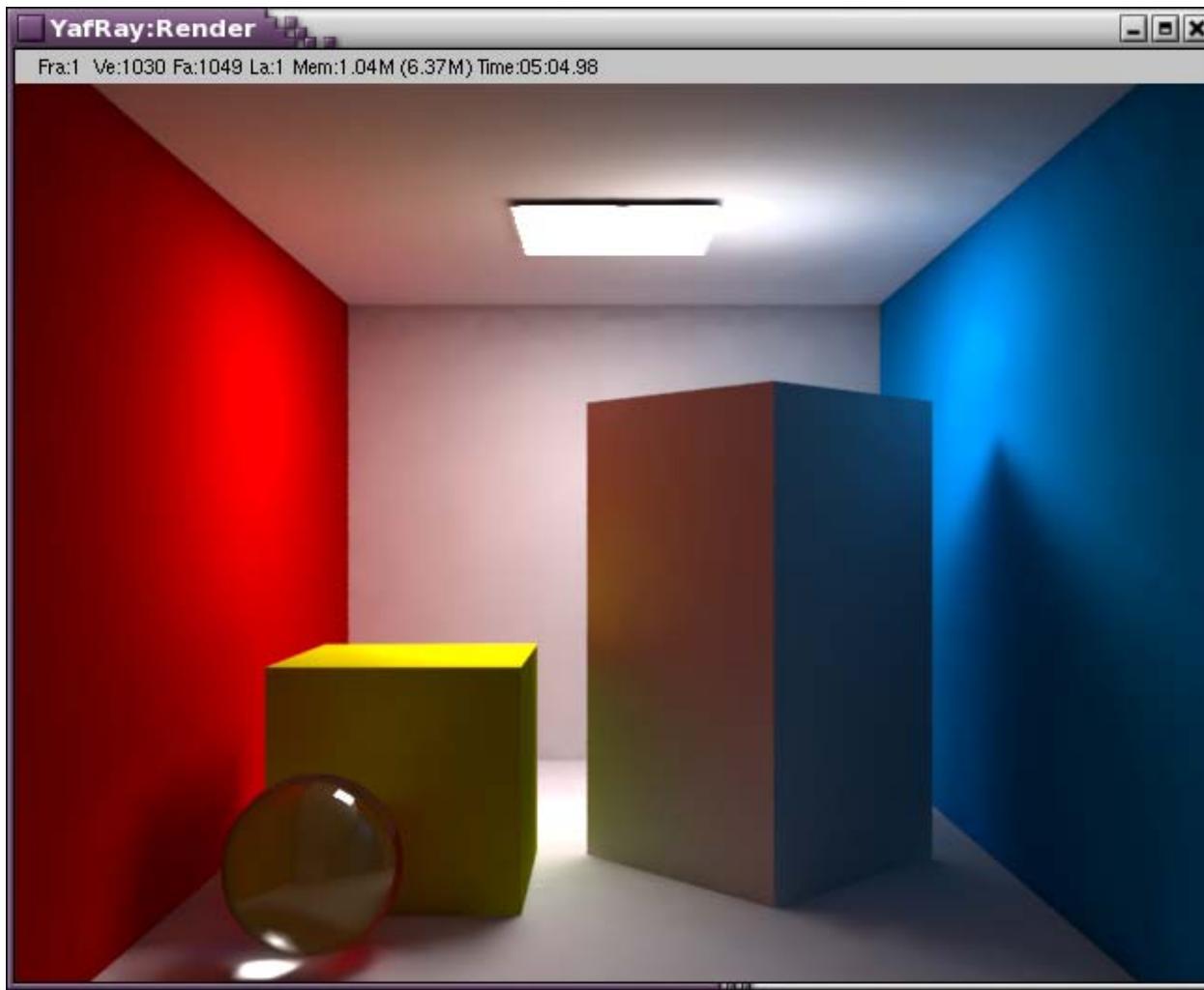
# Problemas com o *Ray-tracing*



A)

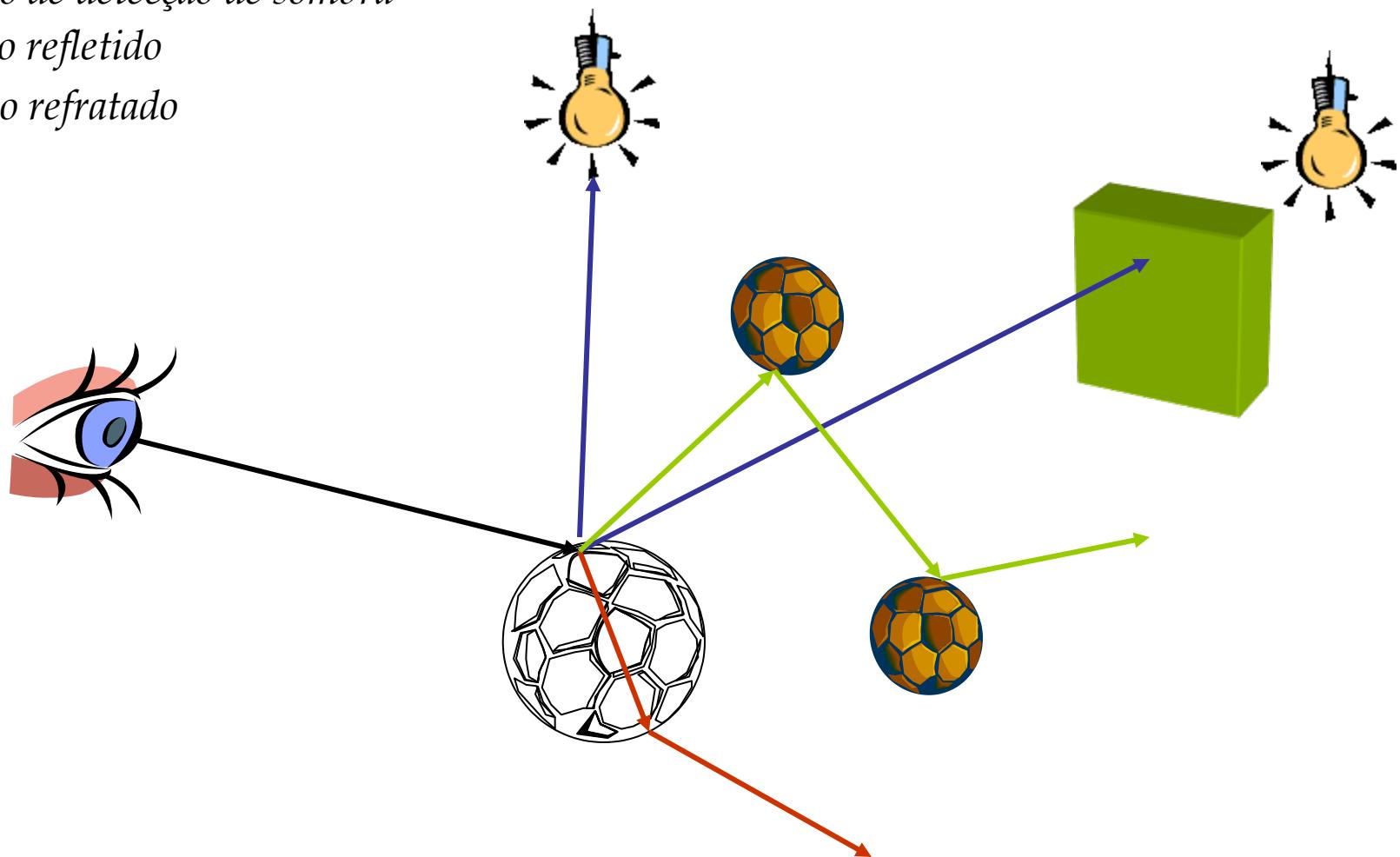


# Reflexão Difusa

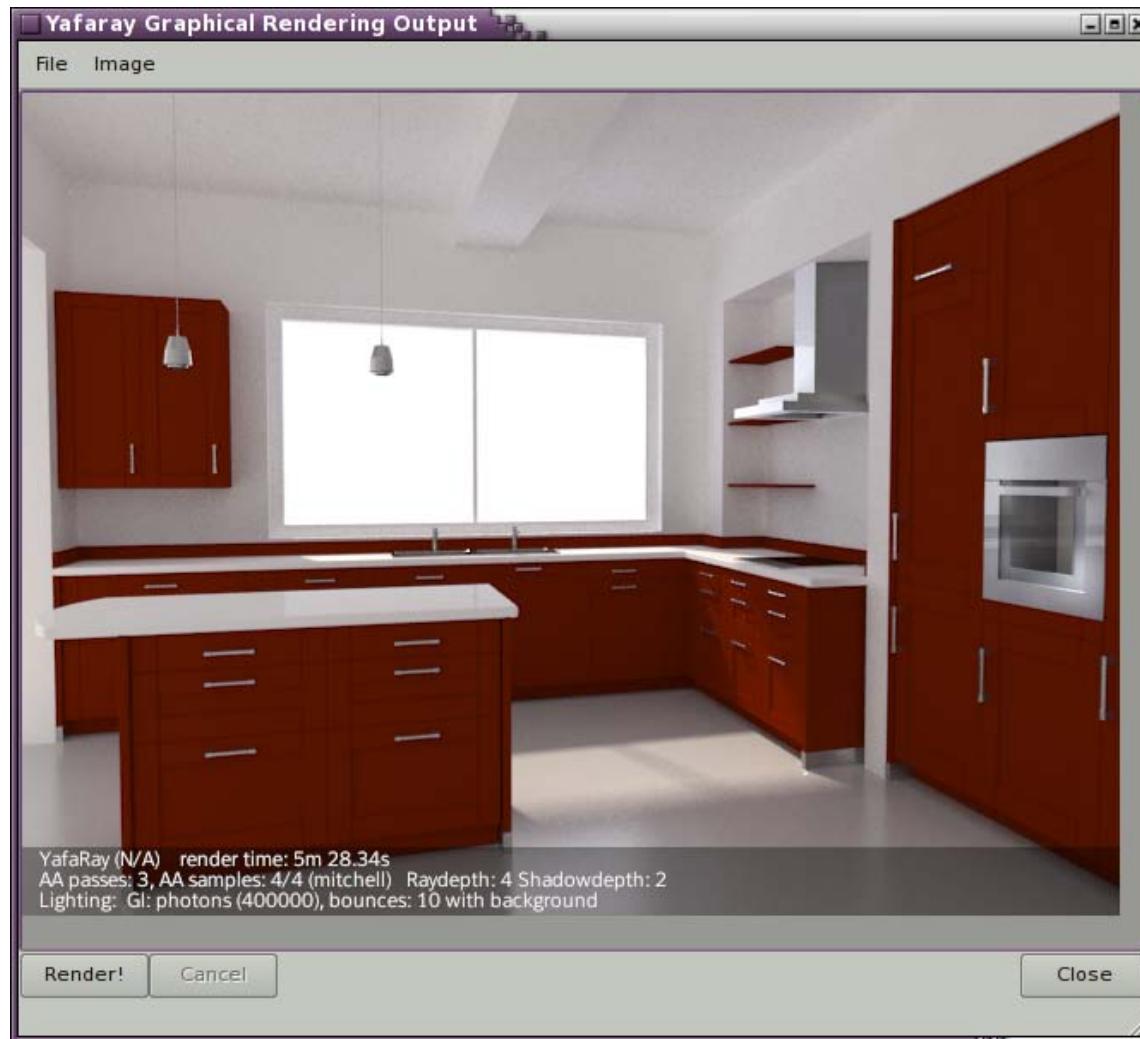


# *Ray Tracing Recursivo*

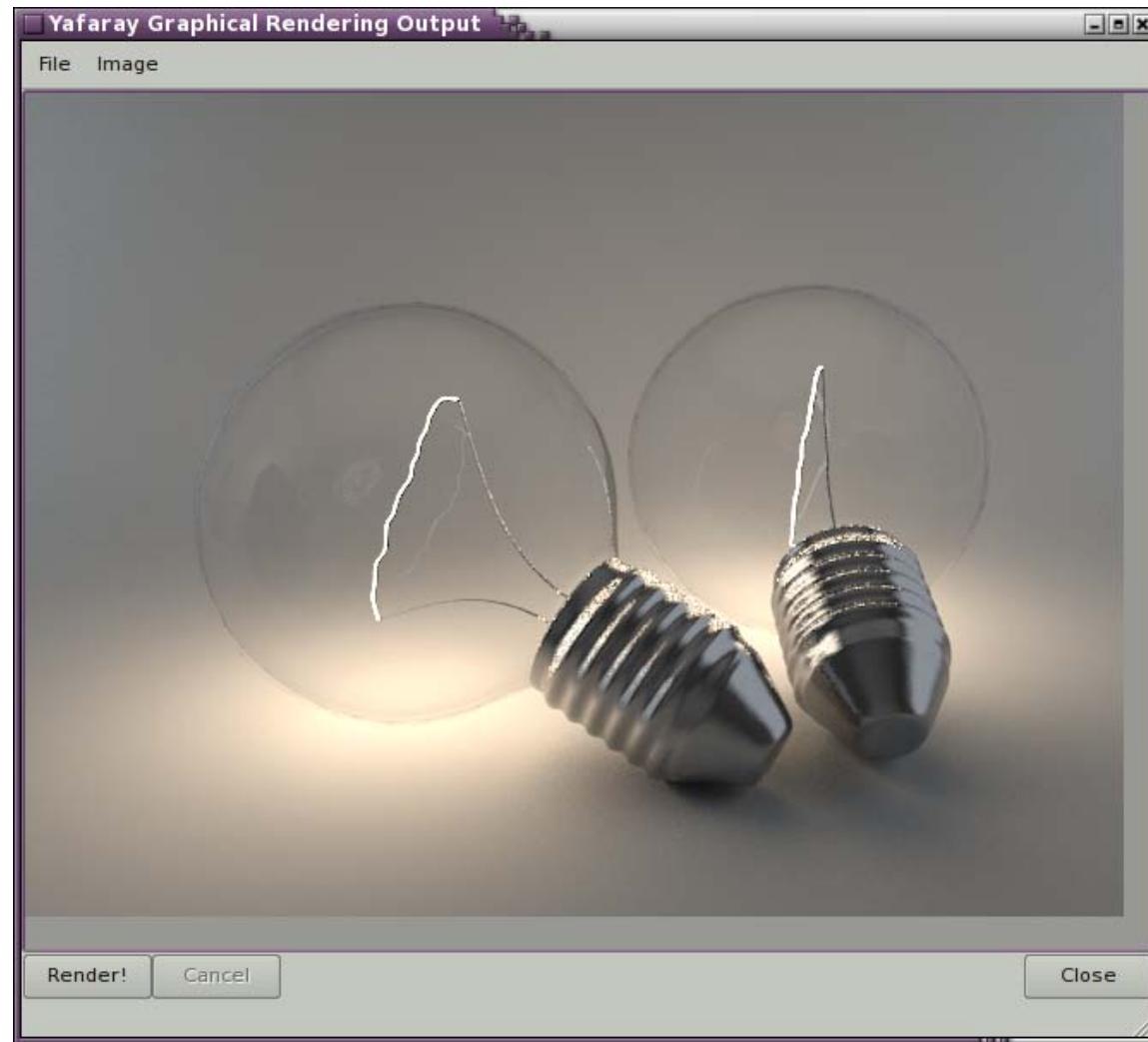
- Raio de visibilidade
- Raio de detecção de sombra
- Raio refletido
- Raio refratado



# Exemplos Criados com o Yafaray (Blender)



# Reflexão



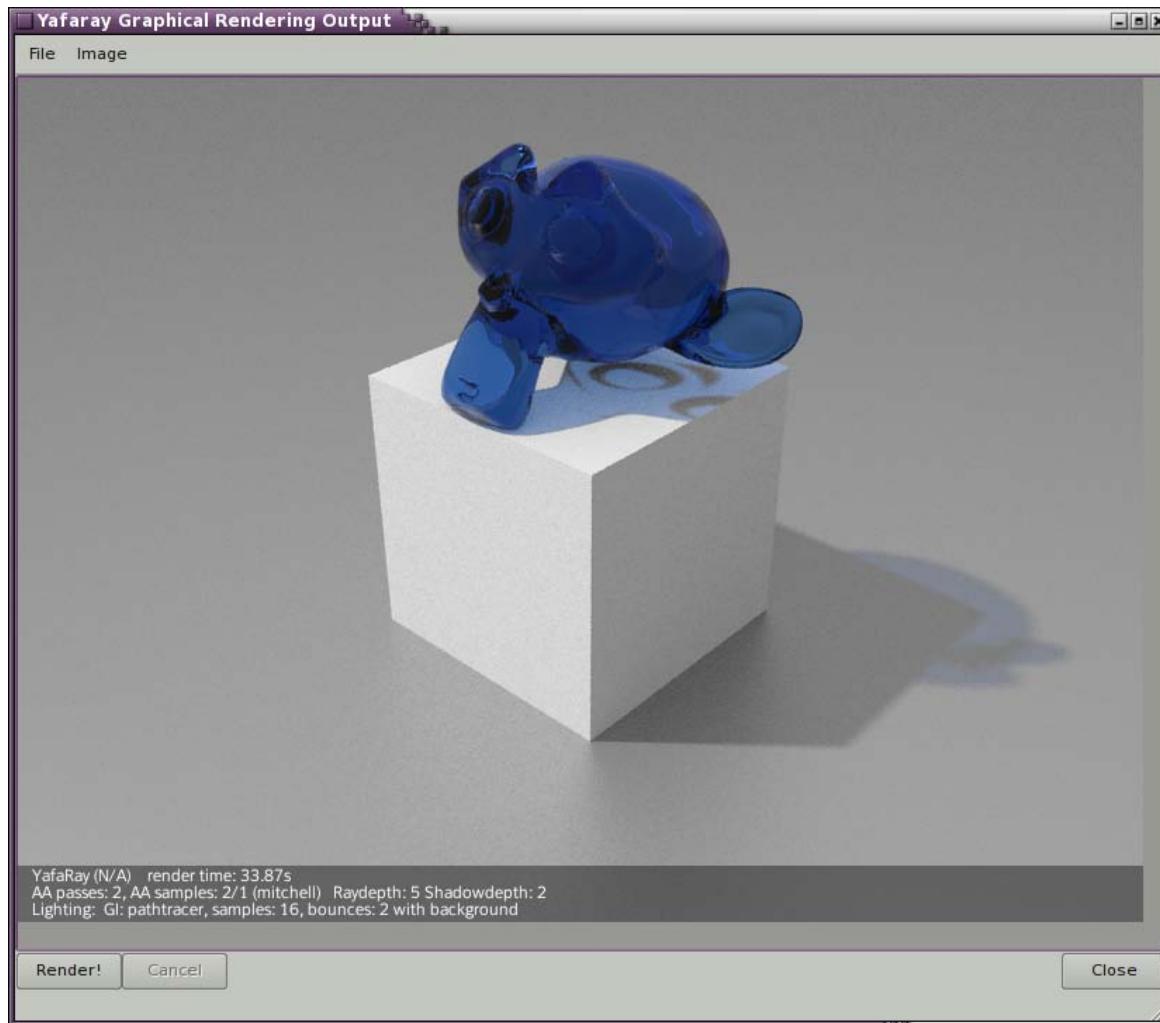
# Illuminação Exterior



# Illuminação Interior



# Transparência



# Objeto Fosco

