

Visualização de Imagem Volumétrica

Alexandre Xavier Falcão

Instituto de Computação - UNICAMP

afalcao@ic.unicamp.br

- Já sabemos traçar raios do plano de visualização em direção à cena, após aplicar a transformação $\phi^{-1}(q + \lambda\vec{n})$ a cada raio que parte de $q \in D_J$.

- Já sabemos traçar raios do plano de visualização em direção à cena, após aplicar a transformação $\phi^{-1}(q + \lambda\vec{n})$ a cada raio que parte de $q \in D_J$.
- Ao encontrar os pontos, p_1 e p_n , de intersecção entre os planos das seis faces da cena e um raio de visualização, sabemos executar o algoritmo DDA para visitar os pontos p_i , $i = 1, 2, \dots, n$ do segmento $\overline{p_1 p_n}$.

- Já sabemos traçar raios do plano de visualização em direção à cena, após aplicar a transformação $\phi^{-1}(q + \lambda\vec{n})$ a cada raio que parte de $q \in D_J$.
- Ao encontrar os pontos, p_1 e p_n , de intersecção entre os planos das seis faces da cena e um raio de visualização, sabemos executar o algoritmo DDA para visitar os pontos p_i , $i = 1, 2, \dots, n$ do segmento $\overline{p_1 p_n}$.
- Nesta aula, vamos primeiro aprender a interromper o algoritmo DDA quando um ponto p_i deste segmento encontra a superfície **opaca** de um objeto da cena.

- Neste momento, vamos projetar a luz $J(q) \leftarrow L(p_i)$ (**modelo de iluminação de Phong**) refletida em p_i na direção e sentido do observador (*rendering de superfície*).

- Neste momento, vamos projetar a luz $J(q) \leftarrow L(p_i)$ (**modelo de iluminação de Phong**) refletida em p_i na direção e sentido do observador (*rendering de superfície*).
- Depois, vamos considerar k superfícies visíveis de objetos semi-transparentes com opacidades $\alpha_j \in [0, 1]$, $j = 1, 2, \dots, k$, sendo perfuradas pelo raio em pontos $p^{(j)} \in \overline{p_1 p_n}$.

- Neste momento, vamos projetar a luz $J(q) \leftarrow L(p_i)$ (**modelo de iluminação de Phong**) refletida em p_i na direção e sentido do observador (*rendering de superfície*).
- Depois, vamos considerar k superfícies visíveis de objetos semi-transparentes com opacidades $\alpha_j \in [0, 1]$, $j = 1, 2, \dots, k$, sendo perfuradas pelo raio em pontos $p^{(j)} \in \overline{p_1 p_n}$.
- Neste caso, as luzes refletidas $L(p^{(j)})$ nas superfícies visíveis são **combinadas** usando as respectivas opacidades até saturar a opacidade acumulada no raio.

$$J(q) \leftarrow \alpha_1 L(p^{(1)}) + \sum_{j=2}^k \alpha_j L(p^{(j)}) \prod_{i=1}^{j-1} (1 - \alpha_i),$$

onde o termo $\prod_{i=1}^{j-1} (1 - \alpha_i) < \epsilon$ indica a saturação.

- O termo

$$\prod_{i=1}^{j-1} (1 - \alpha_i)$$

corresponde ao percentual de luz transmitida até a superfície j , em função das reflexões em superfícies anteriores, que estão entre j e o observador, e desta parcela, a superfície j reflete $\alpha_j L(p^{(j)})$.

- O termo

$$\prod_{i=1}^{j-1} (1 - \alpha_i)$$

corresponde ao percentual de luz transmitida até a superfície j , em função das reflexões em superfícies anteriores, que estão entre j e o observador, e desta parcela, a superfície j reflete $\alpha_j L(p^{(j)})$.

- Finalmente, vamos calcular $\vec{L}(p^{(j)})$ como a **cor refletida** em $p^{(j)}$ quando atribuímos cor a cada objeto, gerando uma projeção colorida em $\hat{J} = (D_J, \vec{J})$.

- O termo

$$\prod_{i=1}^{j-1} (1 - \alpha_i)$$

corresponde ao percentual de luz transmitida até a superfície j , em função das reflexões em superfícies anteriores, que estão entre j e o observador, e desta parcela, a superfície j reflete $\alpha_j L(p^{(j)})$.

- Finalmente, vamos calcular $\vec{L}(p^{(j)})$ como a **cor refletida** em $p^{(j)}$ quando atribuímos cor a cada objeto, gerando uma projeção colorida em $\hat{J} = (D_J, \vec{J})$.
- No caso de não existir uma definição de objetos na cena, vamos ainda estudar modelos para cômputo de $L(p_i)$ e α_i , $i = 1, 2, \dots, n$, capazes de gerar uma visualização semi-transparente da cena (*rendering de volume*).

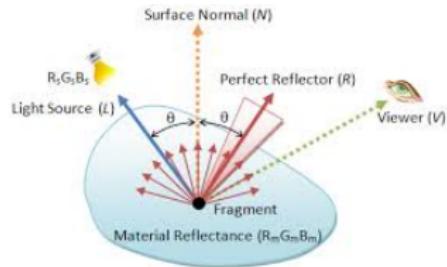
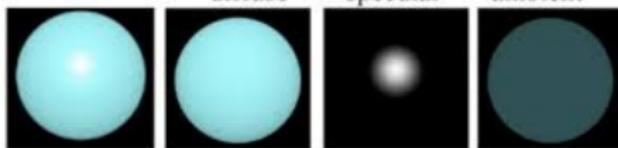
Modelo de Phong

A luz que chega aos olhos do observador combina a reflexão uniforme da luz ambiente com as reflexões difusa e especular da superfície visível do objeto.

Phong Lighting Equation

$$I = K_d L_d(l \cdot n) + K_s L_s(r \cdot v)^a + K_a L_a$$

diffuse specular ambient



Vamos adotar um modelo simplificado em que observador e uma única fonte de luz se encontram na mesma posição, longe o suficiente do plano de visualização para obter uma projeção é ortogonal da cena.

Modelo de Phong

O modelo simplificado fica

$$L(p) = k_a L_a + D(p) (k_d \cos(\theta) + k_s \cos^{n_s}(2\theta)), \quad (1)$$

onde $k_a + k_d + k_s = 1$ são as constantes ambiente, difusa, e especular; $D(p)$ é uma tonalização baseada em profundidade; $L(p)$ só é calculado se $0 \leq \theta \leq \frac{\pi}{2}$ (ângulo entre o vetor $-\vec{v}$ e o vetor $\vec{N}(p)$ normal à superfície em p); e o componente especular só é calculado se $0 \leq \theta \leq \frac{\pi}{4}$.



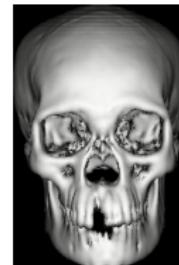
$L(p)$



$k_a L_a$



$D(p)$



$D(p)k_d \cos(\theta)$



$D(p)k_s \cos^{n_s}(2\theta)$

Cômputo do vetor normal $\vec{N}(p)$

- Baseado na cena: vetor gradiente de intensidades da cena.
- Baseado no objeto: vetor gradiente da TDE sinalizada em torno da borda do objeto.
- Baseado na projeção: produto externo entre vetores obtidos do *index buffer*.

Modelo de Phong com cor e opacidade

A reflexão é uma cor no espaço RGB.

$$\vec{J}(q) \leftarrow \alpha_1 \vec{L}(p^{(1)}) + \sum_{j=2}^k \alpha_j \vec{L}(p^{(j)}) \prod_{i=1}^{j-1} (1 - \alpha_i),$$



Um rendering de superfície é gerado com a visão de cada olho (como se um olho estivesse encoberto e o outro não). As imagens podem ser apresentadas ao observador de diversas formas, de modo a enganar o cérebro, dependendo do tipo de dispositivo (óculos) usado para a visualização. Por exemplo, podemos apresentar uma imagem com linhas pares da visão do olho esquerdo e linhas ímpares da visão do olho direito.

Rendering de volume

- Funções de transferência.
- Estimativa do vetor normal $\vec{N}(p)$.