

Aula – Computação Gráfica

Iluminação

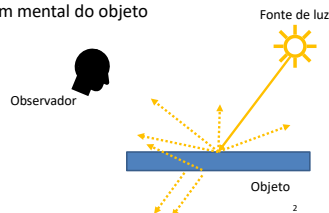
Slides para uso pessoal e exclusivo durante o período de aula. Distribuição ou qualquer uso fora do escopo da disciplina é expressamente proibido.

1

1

O Papel da Iluminação

- Percebemos objetos através da luz captada pelos nossos olhos
- Fontes de luz emitem raios de luz
- Raios de luz interagem com os objetos do ambiente
- Alguns raios são refletidos e chegam aos olhos do observador
- Cérebro forma a imagem mental do objeto



2

Modelagem da Iluminação

Física

- Luz é vista como uma radiação eletromagnética
 - Raios seguem um espectro de cores
- Fontes de luz
 - Emanam raios em todas as direções
- Raios de luz
 - Interagem com todos os objetos do seu entorno
 - Modelagem considera todos os caminhos da luz

Problema

- Altíssimo custo computacional (intratável)

3

3

Modelagem da Iluminação

Computacional em Computação Gráfica

- Luz é vista como um modelo simplificado
 - Utiliza número discreto de primárias
- Fontes de luz
 - Contribuem de forma passiva para o cálculo da iluminação
- Raios de luz
 - Modelos locais
 - Apenas o caminho *fonte de luz* -> *superfície* -> *observador*
 - Simplificado: usado no OpenGL
 - Modelos globais
 - Muitos caminhos são considerados
 - Complexo: usado em *Ray Tracing* (visto em aulas futuras)

Problema

- Quanto mais simplificado, menos realista

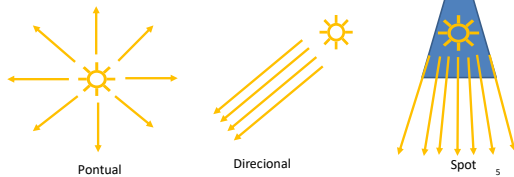
4

4

Modelagem da Iluminação

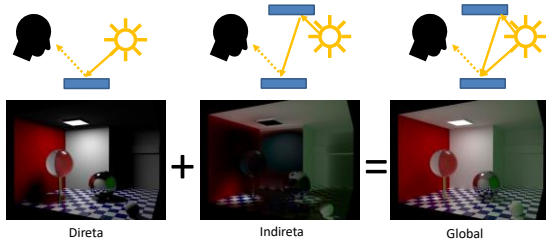
Fontes de Luz

- Permitem Visualizar o Sombreamento do Objeto
- “Emitem” raios de luz para a cena
- Diferentes tipos
 - Exemplos: Pontual, direcional e spot



5

Modelo Global de Iluminação



Barabga (figura do meio gerada pela subtração das outras duas)
https://commons.wikimedia.org/wiki/File:Direct_lighting.png https://en.wikipedia.org/wiki/File:Global_illumination1.png
 CC BY-SA 4.0

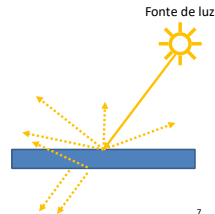
6

6

Componentes da Iluminação

Modelagem por Componentes

- Cada componente modela um efeito
- Junção das componentes fazem o resultado final
- Exemplo de componentes
 - Ambiente
 - Difusa
 - Especular
 - ...



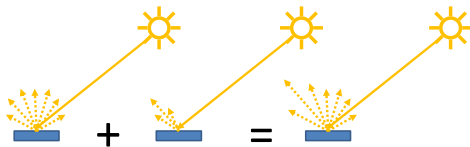
7

7

Componentes da Iluminação

Modelagem por Componentes

- As componentes definem as propriedades dos materiais
- Os materiais guiam a aparência dos objetos
- Combinações de componentes com valores diferentes
 - Definem diferentes materiais



8

8

Geometria dos Objetos

Normal

- A iluminação é calculada para um ponto em uma cena
- A geometria dos objetos no ponto é definida pela normal
- A normal é uma peça fundamental no cálculo da iluminação
- A normal, geralmente, não é calculada automaticamente
 - Deve ser definida na construção do objeto
 - Deve ser inferida a partir de outras normais
- A normal de um triângulo pode ser calculada como:
 - Por exemplo, saindo da tela

$$N = \frac{(C - A) \times (B - A)}{\|(C - A) \times (B - A)\|}$$



9

9

Iluminação e *Shading* (Sombreamento)

Iluminação

- Cálculo da intensidade e cor de um ponto da cena
- Considera vários componentes da cena
 - Geometria da cena
 - Posição do observador, objeto, fonte de luz
 - Propriedades do material dos objetos

Shading (Sombreamento) Não confundir com sombra!

- Cálculo da cor dos pixels
- Parte de um modelo de iluminação conhecido
- Foco em acelerar o processo com simplificações nos cálculos

10

10

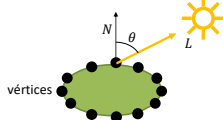
Iluminação e *Shading* (Sombreamento)

Iluminação

- Cálculo feito no universo 3D
- Geralmente, feito de forma completa nos vértices dos objetos

Shading (Sombreamento)

- Cálculo feito após o rastreamento para 2D
- Geralmente aproximado pelos valores de iluminação
 - Considerando o mapeamento dos vértices mais próximos



11

11

Modelos de Iluminação

Exemplos

- Phong
- Blinn-phong

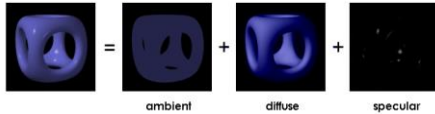
12

12

Modelos de Iluminação Phong

Componentes

- Ambiente
 - Não depende da geometria da cena
- Difusa
 - Espalhamento uniforme da reflexão (lambertiana)
- Especular
 - Espalhamento direcionado da reflexão (superfícies polidas)



https://commons.wikimedia.org/wiki/File:Phong_components_revised.png
CC BY-SA 3.0

13

13

Modelos de Iluminação Phong

Componentes

- Ambiente
 - Simplifica a componente indireta de iluminação
 - Modelos globais são computacionalmente caros
 - Raios devem interagir com objetos ao seu redor
 - Modela esse efeito como uma constante
 - Contribuição dada por:

$$A = I_A M_A$$

14

14

Modelos de Iluminação Phong

Componentes

- Difusa
 - Representa o espalhamento da luz na superfície
 - Característica de objetos foscos
 - No mundo real, objetos tem espalhamento não uniforme
 - Porém, a modelagem é computacionalmente cara
 - Modela esse efeito como superfície lambertiana
 - Espalhamento uniforme para todas as direções
 - Objeto é permanece igual com a variação do ângulo do observador



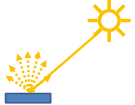
15

15

Modelos de Iluminação Phong

Componentes

- Difusa
 - Representa o espalhamento da luz na superfície
 - Característica de objetos foscos
 - No mundo real, objetos tem espalhamento não uniforme
 - Porém, a modelagem é computacionalmente cara
 - Modela esse efeito como superfície lambertiana
 - Espalhamento uniforme para todas as direções
 - Objeto é permanece igual com a variação do ângulo do observador



16

16

Modelos de Iluminação Phong

Componentes

- Difusa
 - Representa o espalhamento da luz na superfície
 - Característica de objetos foscos
 - No mundo real, objetos tem espalhamento não uniforme
 - Porém, a modelagem é computacionalmente cara
 - Modela esse efeito como superfície lambertiana
 - Espalhamento uniforme para todas as direções
 - Objeto é permanece igual com a variação do ângulo do observador



17

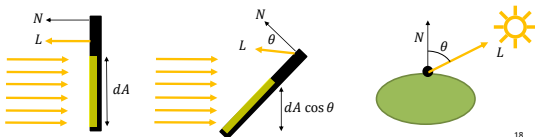
17

Modelos de Iluminação Phong

Componentes

- Difusa
 - Luminosidade aparente depende do ângulo de incidência do raio
 - Não depende do observador
 - Contribuição dada por:

$$D = I_D M_D \cos \theta = I_D M_D \left(\frac{L \cdot N}{\|L\| \|N\|} \right)$$



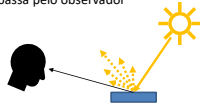
18

18

Modelos de Iluminação Phong

Componentes

- Especular
 - Representa o reflexo da luz na superfície
 - Característica de objetos polidos (tendendo ao espelho)
 - Depende também da posição do observador
 - Espelhos perfeitos exigem a posição perfeita do observador
 - Portanto, a modelagem é feita como um espelho imperfeito
 - A luz se espalha seguindo um cone
 - Centro passa pelo observador



19

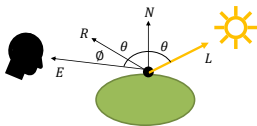
19

Modelos de Iluminação Phong

Componentes

- Especular
 - Contribuição é dada por:

$$S = I_S M_S \cos^n \phi = I_S M_S (R \cdot E)^n$$



20

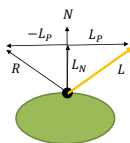
20

Modelos de Iluminação Phong

Componentes

- Especular
 - Cálculo de R (vetor da luz refletida)

$$R = L_N - L_P = (N \cdot L)N - (L - (N \cdot L)N) = 2(N \cdot L)N - L$$



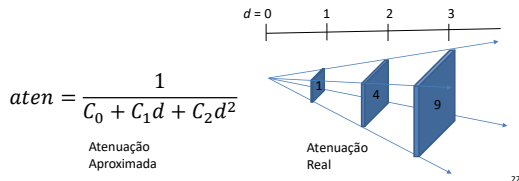
21

21

Modelos de Iluminação Phong

Atenuação

- Para fontes de luz pontuais e spot, é possível definir atenuação
- Ela considera a distância entre a fonte de luz e o objeto
- Objetos mais próximos são mais iluminados
 - Pois a área a ser iluminada cresce com o quadrado da distância



22

Modelos de Iluminação Phong

Atenuação

- Aplicada somente nas componentes difusas e especular
- Onde,
 - A é a componente ambiente
 - D_i é a componente difusa da fonte de luz i
 - S_i é a componente especular da fonte de luz i

$$C = A + \sum aten(D_i + S_i)$$

23

23

Modelos de Iluminação Phong

Componentes

- Especular
 - Coeficiente especular n indica o grau de polimento
 - Espelho perfeito tem grau infinito
 - Valores entre 5 e 100 são bons candidatos em geral

24

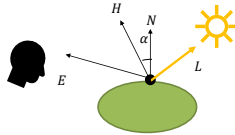
24

Modelos de Iluminação Blinn-Phong

Modelagem

- Igual ao Phong, porém com cálculo simplificado
- Usado no OpenGL
- Utiliza o ângulo entre a normal e o *halfway*

$$S = I_S M_S \cos^n \alpha = I_S M_S (H \cdot N)^n$$



$$H = \frac{E + L}{\|E + L\|}$$

25

25

Iluminação em OpenGL

Formulação

- Contribuição ambiente global (*Amb*) independente das fontes
- Emissão *E* independente das fontes
 - Luz que emana do objeto
 - Equivalente ao *glColor* do 2D
- Componentes de cada fontes de luz *i*
 - Ambiente A_i
 - Difusa D_i
 - Especular S_i
- Atenuação *aten*
- Decaimento do cone *spot*

$$C = Amb + E + \sum aten * spot * (A_i + D_i + S_i)$$

26

26

Iluminação em OpenGL

Fonte de Luz

- `glEnable (GL_LIGHTING);`
 - Habilita o cálculo da iluminação
 - Não usará mais os valores de cores definidos por `glColor`
- `glEnable (source);`
 - Liga uma fonte de luz
 - *source* é uma constante que indica a fonte
 - Exemplo, `GL_LIGHT0`
 - Implementação padrão permite pelo menos 8 fontes

27

27

Iluminação em OpenGL

Fonte de Luz

- `glLightfv(source, property, value);`
 - `source` é uma constante que indica a fonte
 - `property` é uma constante que indica a propriedade modificada
 - Componentes do modelo de iluminação:
 - `GL_AMBIENT`, `GL_DIFFUSE`, `GL_SPECULAR`
 - Geometria
 - `GL_POSITION`, `GL_SPOT_DIRECTION`, `GL_SPOT_CUTOFF`, `GL_SPOT_EXPONENT`
 - Coeficientes de atenuação
 - `GL_CONSTANT_ATTENUATION`, `GL_LINEAR_ATTENUATION`, `GL_QUADRATIC_ATTENUATION`

28

28

Iluminação em OpenGL

Propriedades do Material

- `glMaterialfv(face, property, value);`
 - `face` indica o lado da superfície a se configurar
 - `GL_FRONT`, `GL_BACK`, `GL_FRONT_AND_BACK`
 - `property` é uma constante que indica a componente modificada
 - `GL_AMBIENT`, `GL_DIFFUSE`, `GL_SPECULAR`, `GL_EMISSION`, `GL_SHININESS`

29

29

Iluminação em OpenGL

Normal

- `glNormal3f(x, y, z);`
 - Define a normal de um vértice
 - Geralmente chamada junto com a `glVertex3f` do vértice

30

30

Modelos de *Shading* (Sombreamento)

Exemplos

- Flat
- Gouraud
- Phong **Não confundir com o modelo de iluminação Phong!**

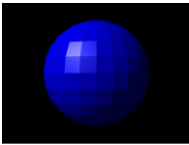
31

31

Modelos de *Shading* (Sombreamento)

Flat

- Normal definida por polígono
 - Exemplo, um triângulo ou retângulo
- Iluminação é calculada por polígono
- *Flat Shading* replica o valor da iluminação por polígono
 - Para cada pixel que pertença a ele



<https://commons.wikimedia.org/wiki/File:Phong-shading-sample.PNG>
Public Domain

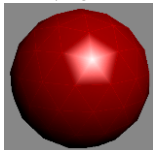
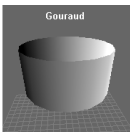
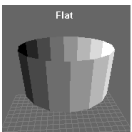
32

32

Modelos de *Shading* (Sombreamento)

Gouraud

- Normal definida por vértice
 - Exemplo, um triângulo ou retângulo
- Iluminação é calculada por vértice
- *Gouraud Shading* interpola os valores da iluminação do polígono
 - Utiliza os valores de cor calculados para os vértices do polígono



https://commons.wikimedia.org/wiki/File:D3D_Shading_Modes.png
Public Domain

https://commons.wikimedia.org/wiki/File:Gouraud_low_gnm.tif
CC BY 2.0

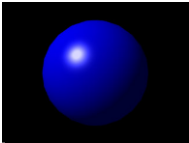
33

33

Modelos de *Shading* (Sombreamento)

Phong

- Normal definida por vértice
- *Phong Shading* calcula iluminação por pixel
 - Interpola normal dos vértices para obter normal do pixel



<https://commons.wikimedia.org/wiki/File:Phong-shading-sample.PNG>
Public Domain

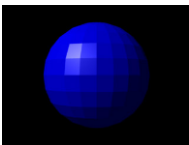
34

34

Modelos de *Shading* (Sombreamento)

Phong

- Normal definida por vértice
- *Phong Shading* calcula iluminação por pixel
 - Interpola normal dos vértices para obter normal do pixel



<https://commons.wikimedia.org/wiki/File:Phong-shading-sample.PNG>
Public Domain

35

35

Perguntas ?????

36

36