

Exercícios Revisão 03 - Computação Gráfica

Gustavo Lopes Rodrigues

29 de novembro de 2021

1)

- **Cor física** é o processo físico que ocorre para a formação da cor, ou seja as ondas eletromagnéticas
- **Cor do objeto** é o resultado das ondas eletromagnéticas refletindo sobre um objeto e chegando aos olhos humanos
- **Cor percebida** refere-se a fração do espectro magnético de cores que podem ser captadas pelos olhos humanos, também conhecido como o espectro visível

2)

- **RGB** - Red(Vermelho), Green(Verde) e Blue(Azul), é um dos principais sistemas de representação de cores, é um sistema aditivo e tem uma utilidade muito grande na exibição de cores em telas de dispositivos eletrônicos.
- **CMYK** - Cyan(Ciano), Magenta(Magenta), Yellow(Amarelo) e Black(Preto) é uma outra forma de representar as cores, porém em um sistema subtrativo. É um sistema mais ideal para fazer impressão, já que este sistema possui uma maior abrangência de cores.

3)

- **Figure-ground relationship** - segmentação do mundo entre figura, o objeto que está em foco em nosso campo de visão e o fundo(background), tudo aquilo que está no nosso campo de visão, mas que não está em foco.
- **Grouping by similarity** - tendência humana em agrupar coisas que estão próximas umas das outras(proximidade também pode ser similaridade)
- **The law of good continuation** - sugere que somos capazes de enxergar melhor linhas contínuas e suaves do que aquelas que são distorcidas
- **Closure** - organizamos nossa percepção em objetos completos, em vez de partes separadas. Em outras palavras, se vemos um conjunto de linhas que parecem formar uma figura geométrica(ex: retângulo), então nossa visão irá completar isso de forma natural.

4)

Os elementos utilizados em uma cena para cálculo de iluminação é: uma fonte de luz(Light Source) e um objeto.

5)

A ideia de usar a câmera como ponto de partida para a renderização de raycast é simples. Para otimizar o custo de processamento do computador na construção da cena, a partir de uma referência do que será mostrado na tela final, o feixe da câmera é emitido para a frente até que o primeiro objeto seja encontrado e construído na cena. Desta forma, objetos atrás do objeto em colisão ou objetos que não aparecerão na imagem final não serão construídos. **6)**

- **Volumes limítrofes** - Encapsular objetos por regiões com seções planas como retângulos(paralelepípedos), circunferências(esferas), elipses(elipsóides). Isto simplifica os cálculos das retas, com as seções do objeto.

- **Hierarquia de volumes limítrofes** - Agrupam-se outros volumes dentro de um volume maior. O método de agrupamento é arbitrário, pode utilizar a distribuição dos objetos na cena.
- **Grid uniforme** - Seção plana da cena dividida em uma matriz, onde cada célula possui o mesmo tamanho.
- **Octree** - Aspecto da cena como critério de homogeneidade, flexibiliza células (regiões das seções planas) de tamanhos variados
- **BSP** - Possibilidade de divisões em regiões mais otimizadas para reduzir volume de cálculos das interseções.

7)

- **Iluminação direta** - Objetos são apenas iluminados por fontes diretas de iluminação.
- **Iluminação global** - Objetos não apenas refletem luz para o observador, mas também para outros objetos.

8)

- **Point** - Ponto que emite luz para todas as direções, com a intensidade da luz diminuindo com a distância;
- **Directional** - Luz uniforme que ilumina uma cena em uma única direção, diferente do **Point**, a intensidade da luz não altera com a distância.
- **Spotlight** - Produz um cone de luz, a luz fica mais intensa quanto mais perto da fonte de luz;
- **Ambient** - Fonte de luz que ilumina objetos, mesmo quando não tem nenhuma outra fonte de luz, Luz ambiente tem uma intensidade que é independente de direção, distância e outros objetos.

9)

- **Difusa** - Iluminação direta, onde o raio incidente é decomposto em uma superfície refletora, e é refletido em várias direções igualmente espaçadas.
- **Ambient** - Iluminação sem direção (global), interage de forma uniforme com todas as superfícies;
- **Specular** - Iluminação que dá a objetos pontos de brilho e realce

10)

$$I_L = \sum_{i=1}^n (K_{Es} * I_{Es} + K_D * I_D) + K_{Em} * I_{Em} + I_A \quad (1)$$

$K_D \rightarrow$ coeficiente de atenuação

$\propto \frac{1}{c+bd+ad^2}$

$n \rightarrow$ número de fontes de luz

11)

Flat shading é mais utilizado quando precisamos de um método de shading com baixo custo computacional, já que este precisa calcular a luz para cada polígono apenas uma vez por render de forma uniforme.

Flat shading é geralmente utilizado para visualizações rápidas (preview), ou quando temos objetos muito distantes.

12)

- **Gouraud** - Sobreamento interpolado onde a luz é calculado por vertex(Vertex shader) de uma primitiva, e então interpola a luz para os fragmentos da primitiva
- **Phong** - Outro tipo de sobreamento interpolado, porém a luz é calculado separadamente para cada fragmento de shader. Em outras palavras: diferente do Gouraud, onde o polígono todo é usado para o cálculo, no Phong, apenas as normais das vertexs são usadas.

13)

O método **texture scanning** apresenta o problema de resultar em um mapeamento parcial das texturas, fazendo com que as bordas não fiquem corretamente "costuradas". Uma maneira de resolver esse problema seria através do mapeamento inverso, permitindo o mapeamento por completo, e da escolha de qual RGB usar.

14)

Para o cálculo de iluminação, o programa precisa levar em conta os seguintes passos: primeiro precisa estabelecer o(s) objeto(s) emissor(es) da cena, a forma como a fonte de luz é formada e o ângulo de incidência que é formado no objeto que queremos iluminar, depois temos a técnica de sobreamento, que influenciará como a intensidade de cada uma das reflexões(Especular,Difusa e Ambiente)será calculada, com estas considerados, o cálculo poderá ser feito.

15)

- **Mapeamento Procedural** - Vértices em cada ponto definida previamente que pode ser alterado para incluir detalhes ao longo da superfície com variações do contraste das intensidades(claro x escuro, mais profundo x menos profundo)
- **Bump Mapping** - Vetores normais ao longo da superfície são definidas previamente e alterados posteriormente com base em uma função que distorce estes vetores. Bump maps não causam modificações na superfície do modelo, apenas causa essa ilusão por meio da textura.

16)

- **key frames** - Definir frames(imagens) fixos para determinado instante do tempo
- **in-betweens** - Também chamado de tweening, é o processo de gerar frames intermediários entre dois keyframes. O objetivo é criar a ilusão de movimento, a partir de uma transição suave entre os frames.

17)

O maior custo do key framing, se deve ao custo elevado pela mão de obra, para definir individualmente quais serão os keyframes.

18)

Vértices

$$V_{min} = 5 \quad V_{max} = 6$$

$$N_s = (V_{max} - 1) \% (V_{min} - 1) \rightarrow \text{arestas}$$

$$N_p = \left(\frac{V_{max} - 1}{V_{min} - 1} \right) \rightarrow \text{vértices}$$

Arestas

$$A_{min} = 5 \quad A_{max} = 6$$

$$N_a = A_{max} \% A_{min}$$

$$N_s = (int) \frac{A_{max}}{A_{min}}$$

19)

- **Cinemática** - Reposicionamento do objeto ao longo do tempo por um movimento que é realizado por ele. As forças que geraram o movimento e sua variação ao longo do tempo são desprezadas no cálculo da cinemática.
- **Dinâmica** - Nos cálculos da cinemática acrescenta-se a informação da variação das forças que geraram o movimento.

20)

Na Cinemática Direta, entramos com a posição inicial(em um tempo qualquer), a velocidade e aceleração de um objeto, e queremos como saída a posição nos próximos tempos.

Já na Cinemática Inversa, incluímos a posição inicial(e seu tempo), e então colocamos a segunda posição(e seu tempo), com isso, a saída será a velocidade e aceleração necessária para completar essa travessia.