Programação 3D - Assignment I

Grupo 02
Francisco Campaniço 83463
João Rafael 83482
Rodrigo Oliveira 83558

Março 2019

Parser NFF

O programa começa por ler o ficheiro NFF definido pelo utilizador no início do código. Seguindo a especificação do formato NFF, adiciona uma câmara, luzes e objetos à cena. A cena (classe `Scene`) contêm uma câmara, um número de luzes definido pelo ficheiro NFF, e objetos também definidos pelo mesmo.

Um objeto (classe `SceneObject`) contém uma classe `Material` e métodos genéricos de interseção e cálculo de normais, que são *overriden* pelas classes que herdam desta.

Dado que o formato NFF não têm uma opção para *Bounding Boxes*, foi adicionada uma opção para tal: **`aabb x0 x1 y1 y2 z1 z2`**, que cria uma AABB com os limites em cada eixo declarados pelas variáveis seguintes.

Interseções

O raio verifica interseções com objetos ao chamar o método `intersect` de cada objeto, que devolve um *boolean* e a distância t da origem do raio ao objeto. Depois disto, o objeto com menor t é o mais próximo, logo os cálculos seguintes de iluminação aplicam-se a este.

A tabela abaixo contém os resultados de tempo para cada um dos testes disponibilizados. Estes resultados foram obtidos num portátil com um CPU Intel Core i7-8750H (6 *cores*, 2.2GHz) e uma GPU NVIDIA GTX 1060, correndo Kubuntu 18.04 e usando o programa Unix `time` para determinar o tempo de execução.

Teste	Low	Medium	High	Very High
Balls	0m11s	0m18s	12m14s	-
Mount	0m15s	-	8m58s	2h21m3s

Raio - Esfera

O cálculo da interseção raio - esfera contém várias otimizações. Primeiro, se a distância entre a origem do raio e o centro da esfera for maior que o raio da mesma, calcula-se

$$B = x_d(x_c + x_o) + y_d(y_c + y_o) + z_d(z_c + z_o).$$

Se B for negativo, o raio aponta para a direção oposta da esfera pode-se concluir que não há interseção.

Depois, calculamos

$$R = B^2 - C = B^2 - d_{OC}^2 + r^2$$

e se B for negativo, conclui-se também que não há interseção.

Finalmente, há interseção. Finalmente, calculamos

$$t_i = \begin{cases} B - \sqrt{R}, & \text{se } d_{OC}^2 > r^2 \\ B + \sqrt{R}, & \text{se } d_{OC}^2 \leq r^2 \end{cases}$$

para determinar o ponto de interseção e a sua normal, para efeitos de cálculo de cor.

Raio - Plano

A interseção raio - plano é simples; calcula-se o produto interno da normal do plano e a direção do raio. Se esta for zero, o raio e o plano são paralelos e como tal não se podem intersetar.

Depois, calculamos

$$t_i = -\frac{(o - a) \cdot n}{n \cdot d}.$$

Se t for negativo, rejeita-se os cálculos. Caso contrário, usa-se o t para determinar o ponto de interseção e a sua normal.

Raio - Triângulo

Para calcular a interseção entre raios e triânglos, utiliza-se o algoritmo de Möller-Trumbore. Primeiro, calculamos o determinante da matriz com os 3 pontos do polígono, que também pode ser calculada através de

$$det = a_{01} \cdot (d \times a_{02}),$$

a_{01} e a_{02} sendo arestas do triângulo. Se este determinante for zero, raio e triângulo são paralelos e abandonam-se os cálculos de interseção.

Depois, calcula-se coordenadas baricêntricas u e v. Se u não estiver no intervalo (0,1), v for negativo ou $u + v > 1$, abandonam-se os cálculos.

Finalmente, calculamos

$$t_i = \frac{1}{det}(a_{02} \cdot ((o - v_0) \times a_{01})).$$

Se t for negativo, a direção do raio é oposta ao vetor entre raio e triângulo, por isso abandonam-se os cálculos. Caso contrário, há uma interseção e usamos t para calcular o ponto de interseção e a normal correspondente.

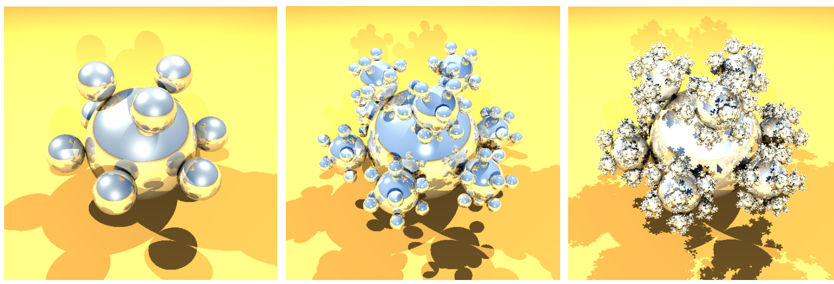
Raio - Cone (Extra)

Raio - AABB (Extra)

Iluminação Blinn-Phong

Sombras

Reflexão



Refração