

# Relatório Exercício Prático 8

Gustavo Ciotto Pinton 117136  
EA979 - Processamento de Imagens

## Explicação

O modelo de iluminação proposto por Phong é composto pela soma de três componentes, sendo elas, ambiente, difusa e especular. A iluminação **ambiente** atinge todos os objetos em todas as direções de maneira similar e é refletida, consequentemente, em todas elas também. A luz **difusa**, por sua vez, é proveniente de uma fonte pontual e atinge cada objeto  $i$  da cena com um ângulo  $\theta_i$ , que consiste no ângulo entre a normal da respectiva superfície e a direção com que a luz atinge  $i$ . Essa luz é refletida em todas as direções com intensidades variando em função de  $\cos \theta_i$ . Assim, esse tipo de iluminação depende da posição da fonte em relação aos objetos. Por fim, a luz **especular** também provém de uma fonte pontual, porém depende do tipo de material com que o objeto é feito e da posição do observador. O primeiro determina uma espécie de campo de reflexão, isto é, um intervalo de ângulos em que o observador visualizará a luz refletida, dependendo de sua posição. Materiais como o espelho possuem um intervalo bem pequeno de reflexão e, em oposição, objetos feitos de plástico, por exemplo, um grande campo. Em outras palavras, tal modelo aproxima-se do conceito de brilho, caracterizado pelos tipos de materiais.

Para ativar a iluminação, a biblioteca *OpenGL* disponibiliza a função `glEnable`. *OpenGL* oferece suporte a várias fontes de iluminação, porém vamos utilizar somente uma delas, identificada por `GL_LIGHT1`. O código utilizado para habilitar a iluminação é, portanto:

```
glEnable(GL_LIGHTING);  
glEnable(GL_LIGHT1);
```

A atribuição das três iluminações é realizada por uma mesma função, chamada de `glLightfv`, recebendo como parâmetro a identificação da fonte (neste caso, sempre `GL_LIGHT1`), o tipo de iluminação (ambiente, difusa ou especular) e as características relacionadas a ela (cor ou posição, por exemplo). As 3 figuras abaixo representam cada um dos três tipos de iluminação utilizados **separadamente**, para um fonte luminosa posicionada no eixo  $z$ , nas coordenadas  $(0, 0, 5)$ , isto é, no mesmo eixo e local em que o observador se encontra.

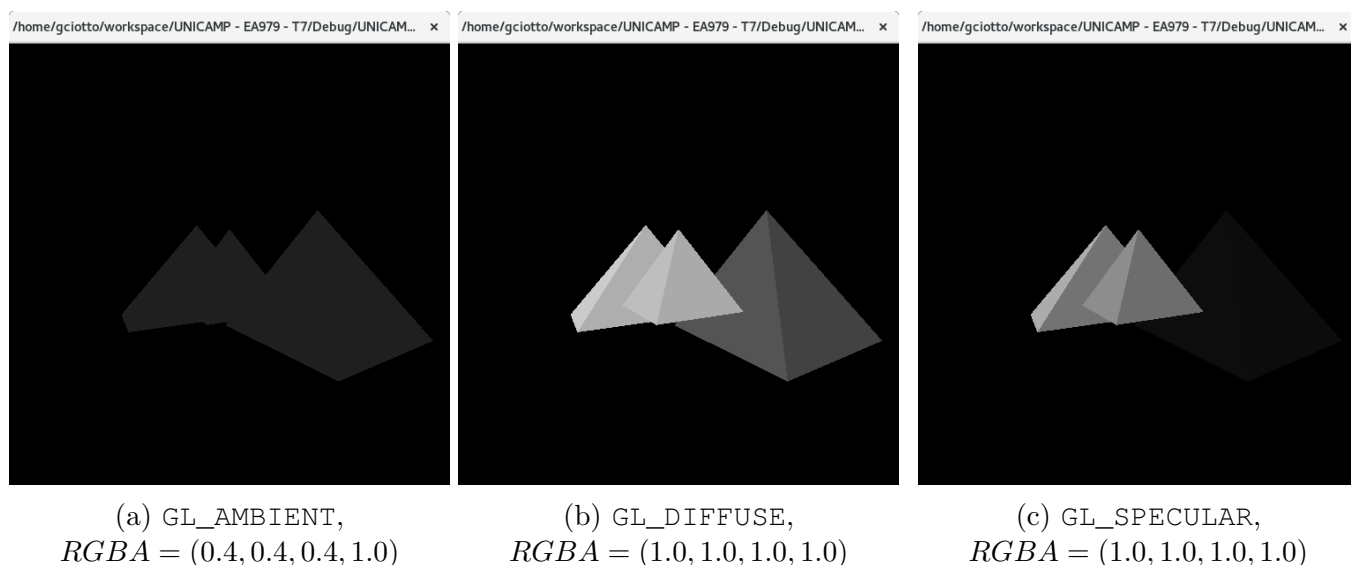


Figura 1: Tipos de iluminação utilizados **separadamente**.

Observa-se que

- i. A iluminação ambiente, figura 1a, não acrescenta a noção de profundidade ao objeto, uma vez que ilumina todas as faces de uma maneira similar.
- ii. A iluminação difusa, figura 1b, depende da posição da fonte. Objetos posicionados mais próximos ao eixo  $z$ , isto é, pirâmides do centro e à esquerda, são iluminados mais fortemente. A pirâmide mais à direita, estando mais distante de  $z$ , reflete menos luminosidade, visto que o cosseno para ângulos próximos de  $90^\circ$  é pequeno.
- iii. A iluminação especular, figura 1c, depende do material e posições do observador e fonte. Para a especificação do material, utilizamos o trecho abaixo. A função `glMaterialfv` especifica os parâmetros do material para o modelo de iluminação. Em especial, o parâmetro `GL_SHININESS` especifica o expoente  $ns$  da fórmula  $(\cos \phi)^{ns}$ .

```
GLfloat mat_specular[] = { 1.0, 1.0, 1.0, 1.0 };
GLfloat mat_shininess[] = { 05.0 };
glMaterialfv(GL_FRONT, GL_SPECULAR, mat_specular);
glMaterialfv(GL_FRONT, GL_SHININESS, mat_shininess);
```

A figura abaixo representa a soma das três figuras acima.

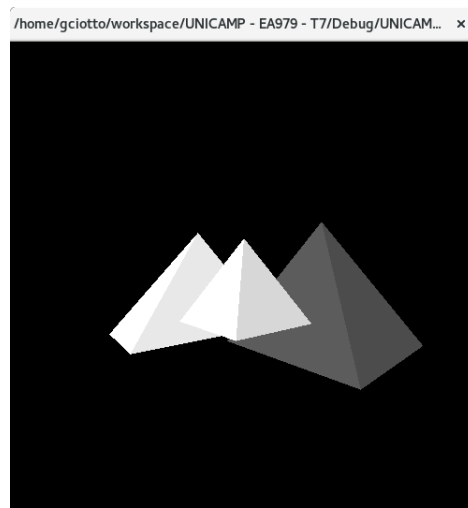
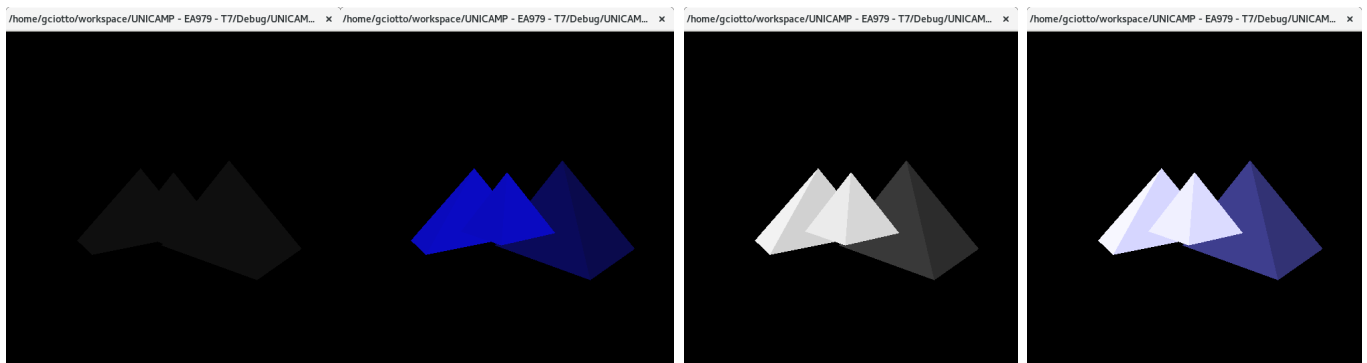


Figura 2: Renderização de 3 pirâmides com iluminação.

As figuras abaixo foram geradas para diferentes parâmetros das iluminações. Para a luz ambiente, utilizamos  $RGBA_{ambient} = (0.1, 0.1, 0.1, 1.0)$ , para a difusa usamos somente a componente azul, isto é,  $RGBA_{diffuse} = (0, 0, 1.0, 1.0)$ , e, para a especular, diminuimos o expoente para  $ns = 2$ . Além disso, posicionamos a fonte à esquerda dos objetos, na posição  $(5, 2, 0)$ .



(a) GL\_AMBIENT

(b) GL\_DIFFUSE

(c) GL\_SPECULAR

(d) Todas as iluminações.

Figura 3: Tipos de iluminação utilizados separadamente.

## Referências

1. *OpenGL Programming Guide Chapter 5*, disponível em <http://www.glprogramming.com/red/chapter05.html>.