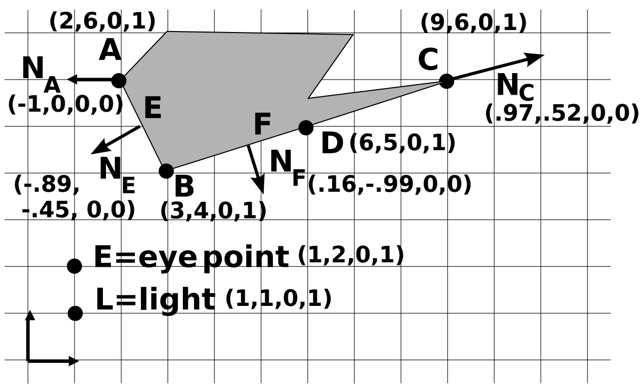
CPSC 314, Written Homework 3

solution and grading criteria by Yufeng Zhu
Value: 2% of final grade
Total Points: 50

Lighting and Shading (50 pts)

For the following questions, refer to the figure and parameters below. Show your work. Remember to normalize all vectors used in lighting calculations!



- ambient light color I_a is (.1,.1,.2)
- light color I_L is (1.0, 1.0, 1.0)
- diffuse material color k_d is (.2, .8, .3)
- ambient material color k_a is (.9, .5, .5)
- specular material color k_s is (1, .5, 1)
- shininess exponent is 10

1. (2 pts) Compute the normal at point B using per-vertex normals, interpolating between the provided normals for face E and face F.

Solution

There are several different ways to compute vertex normal depending on different context. In this problem, we introduce the simplest way which is equal weight average normal. Here 'equal weight' means the multiplier factor we choose for each face normal is the same.

$$\vec{\mathbf{N}}_{V_i} = \frac{\sum\limits_j \omega_j \vec{\mathbf{N}}_{F_{ij}}}{\sum\limits_j \omega_j} \tag{1}$$

$\vec{\mathbf{N}}_{V_i}$ is the vertex normal of vertex i. F_{ij} is the jth face surrounding vertex i. And $\vec{\mathbf{N}}_{F_{ij}}$ is its face normal while ω_j is its weight contributing to the vertex normal. As we only consider equal weight case and just have two face in this problem. Equation 1 can be reduced to the following,

$$\vec{\mathbf{N}}_B = \frac{\omega\vec{\mathbf{N}}_E + \omega\vec{\mathbf{N}}_F}{2\omega} = \frac{1}{2}(\vec{\mathbf{N}}_E + \vec{\mathbf{N}}_F) \tag{2}$$

Plug in the normal vector value given in the problem, we have

$$\vec{\mathbf{N}}_B = \begin{pmatrix} -0.45 \\ -0.89 \\ 0 \\ 0 \end{pmatrix} \tag{3}$$

[0.5 pt for each entry of normal vector]

Note for Q2 to Q4

Phong lighting model and Phong shading model are totally different concept. The former is a local lighting model producing visible color values given light direction, view direction, normal direction, material properties, etc. You can apply it to vertex, patch, curve or any defined primitives. Phong shading model as well as flat shading and Gouraud shading are instead more like strategies deciding how coloring (i.e. applying Phong lighting model) is organized over a given surface (i.e. triangle, polygon, etc).

- Flat shading: we color one point on the surface and use it as the color for the whole surface.
- Gouraud shading: we color every vertex of the surface and then **interpolate the color** for every other point or fragment on the surface.
- Phong shading: we first **interpolate the normal** for every point or fragment on the surface depending on the given vertex normal. Then color every point or fragment on the surface as we have all the information required for coloring at each point after normal interpolation.

Two more issues need to be clarified:

- When coloring, we should always use local information like normal, vertex position, light direction, material property, etc to avoid ambiguity. However, this time I will still give credit to students who use face normal for flat shading.
- Some students asked me what interpolation method they should use. Well, in general, you can use any interpolation method for your own project as long as it satisfies your requirement. In this homework, we use 1D linear interpolation for simplicity.

2. (16 pts) Compute the ambient, diffuse, specular, and total illumination at points B, C, and D using Phong lighting and the flat shading model. For flat shading, use the rightmost point on each face.

Solution

We use the rightmost point C for face F. Thus we just need to color point C and set it to both point B and D under flat shading.

$$k_a I_a = \begin{pmatrix} 0.09 \\ 0.05 \\ 0.1 \end{pmatrix}, k_d I_L = \begin{pmatrix} 0.2 \\ 0.8 \\ 0.3 \end{pmatrix}, k_s I_L = \begin{pmatrix} 1 \\ 0.5 \\ 1 \end{pmatrix}$$
$$\vec{N}_C = \frac{\vec{N}_C}{\|\vec{N}_C\|} = \begin{pmatrix} 0.88 \\ 0.47 \\ 0 \\ 0 \end{pmatrix}, \vec{L}_C = \frac{\vec{L} - \vec{C}}{\|\vec{L} - \vec{C}\|} = \begin{pmatrix} -0.85 \\ -0.53 \\ 0 \\ 0 \end{pmatrix}, \vec{N}_C \cdot \vec{L}_C = -0.997 < 0 \tag{4}$$

The dot product is less than 0 implies that the light source is on the backside of the surface. Thus by default there is no diffuse and specular contribution from this light source even though the view point is also on the backside of the surface in this case. However, here I still give the step for computing the specular component, especially for reflection direction.

$$\vec{R}_C = 2(\vec{N}_C \cdot \vec{L}_C)\vec{N}_C - \vec{L}_C = \begin{pmatrix} -0.90 \\ -0.41 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_C = \frac{\vec{E} - \vec{C}}{\|\vec{E} - \vec{C}\|} = \begin{pmatrix} -0.89 \\ -0.45 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_C \cdot \vec{R}_C = 0.99$$
$$I_{diff} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, I_{spec} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{5}$$
$$I_B = I_D = I_C = \begin{pmatrix} 0.09 \\ 0.05 \\ 0.1 \end{pmatrix}$$

Alternative Solution

Using $\vec{N}_F$, the face normal, as normal vector when computing the color at vertex C,

$$I_B = I_D = I_C = \begin{pmatrix} 1.0 \\ 0.81 \\ 1.0 \end{pmatrix} \tag{6}$$

3. (16 pts) Do those computations using the Gouraud shading model.

Solution

Applying the same procedure to compute the color of vertex B while color of vertex C stays the same as in Q2. Finally we get color of D by linear interpolate between I_B and I_C,

$$
\begin{aligned}
I_C &= \begin{pmatrix} 0.09 \\ 0.05 \\ 0.1 \end{pmatrix} \\
\vec{N}_B = \frac{\vec{N}_B}{\|\vec{N}_B\|} &= \begin{pmatrix} -0.45 \\ -0.89 \\ 0 \\ 0 \end{pmatrix}, \vec{L}_B = \frac{\vec{L}-\vec{B}}{\|\vec{L}-\vec{B}\|} = \begin{pmatrix} -0.55 \\ -0.83 \\ 0 \\ 0 \end{pmatrix}, \vec{N}_B \cdot \vec{L}_B = 0.99 \\
\vec{R}_B = 2(\vec{N}_B \cdot \vec{L}_B)\vec{N}_B - \vec{L}_B &= \begin{pmatrix} -0.34 \\ -0.93 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_B = \frac{\vec{E}-\vec{B}}{\|\vec{E}-\vec{B}\|} = \begin{pmatrix} -0.71 \\ -0.71 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_B \cdot \vec{R}_B = 0.90 \\
I_{diff} &= \begin{pmatrix} 0.20 \\ 0.79 \\ 030 \end{pmatrix}, I_{spec} = \begin{pmatrix} 0.33 \\ 0.17 \\ 0.30 \end{pmatrix}, I_B = \begin{pmatrix} 0.62 \\ 1.0 \\ 0.73 \end{pmatrix} \\
I_D = \frac{1}{2}(I_B + I_C) &= \begin{pmatrix} 0.36 \\ 0.53 \\ 0.41 \end{pmatrix}
\end{aligned} \tag{7}
$$

4. (16 pts) Do those computations using the Phong shading model.

Solution

The color of vertex B and C stay the same as in Q3. Then we compute the normal of vertex D by linear interpolating between normals of B and C. After that we apply the same procedure shown in Q2 to compute the color of vertex D.

$$
\begin{aligned}
I_B &= \begin{pmatrix} 0.67 \\ 1.0 \\ 0.78 \end{pmatrix}, I_C = \begin{pmatrix} 0.09 \\ 0.05 \\ 0.1 \end{pmatrix} \\
\vec{N}_D = \frac{\vec{N}_B + \vec{N}_C}{\|\vec{N}_B + \vec{N}_C\|} &= \begin{pmatrix} 0.72 \\ -0.70 \\ 0 \\ 0 \end{pmatrix}
\end{aligned} \tag{8}
$$

Be careful when you average the normal. Always normalize each normal first and then average them. Otherwise, you are introducing different weights into Equation 1.

$$
\begin{aligned}
\vec{L}_D = \frac{\vec{L}-\vec{D}}{\|\vec{L}-\vec{D}\|} &= \begin{pmatrix} -0.78 \\ -0.62 \\ 0 \\ 0 \end{pmatrix}, \vec{N}_D \cdot \vec{L}_D = -0.12 < 0 \\
\vec{R}_D = 2(\vec{N}_D \cdot \vec{L}_D)\vec{N}_D - \vec{L}_D &= \begin{pmatrix} 0.61 \\ 0.80 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_D = \frac{\vec{E}-\vec{D}}{\|\vec{E}-\vec{D}\|} = \begin{pmatrix} -0.86 \\ -0.51 \\ 0 \\ 0 \end{pmatrix}, \vec{V}_D \cdot \vec{R}_D = -0.93 < 0 \\
I_{diff} &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, I_{spec} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\
I_D &= \begin{pmatrix} 0.09 \\ 0.05 \\ 0.1 \end{pmatrix}
\end{aligned} \tag{9}
$$

Marking Checklist for Q2 to Q4

- **[4 pts clamping dot product to 0 for diffuse part]**
- **[4 pts clamping dot product to 0 for specular part]**
- **[4 pts clamping color to 1]**
- **[8 pts 1 vertex color calculation for flat shading]**
- **[2 pts off for each vector used in lighting without normalized]**
- **[4 pts for correct viewing direction vector]**
- **[4 pts for correct lighting direction vector]**
- **[4 pts for correct reflection vector]**
- **[2 pts off for forgetting shinness exponentiation]**
- **[4 pts for correct diffuse calculation]**
- **[4 pts for correct specular calculation]**
- **[4 pts for correct ambient calculation]**

[marks will be deducted only once for the same kind of errors]