Name: Catherine Nguyen

Lab 3: Shading and Illumination

1. Phong illumination:

Phong illumination model is implemented as the Illumination class.

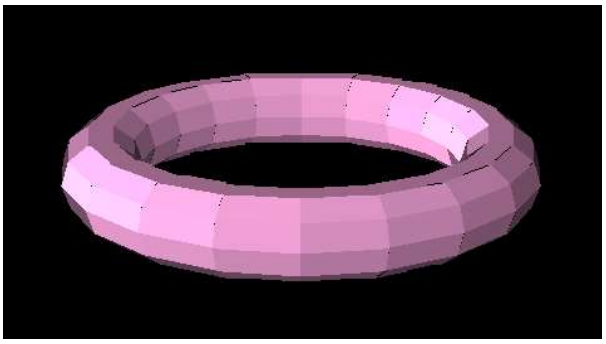
```
class Illumination:
   def __init__(self, ka, kd, ks, L, V, sourceLight):
      self.ka = ka
      self.kd = kd
      self.ks = ks
      self.sourceLight = sourceLight
      self.iAmbient = np.array([0.4, 0.4, 0.4])

      self.L = unitVector(L)
      self.V = unitVector(V)
      self.h = L + V
      self.h = unitVector(self.h)
```

The Phong Illumination is implemented using this formula:

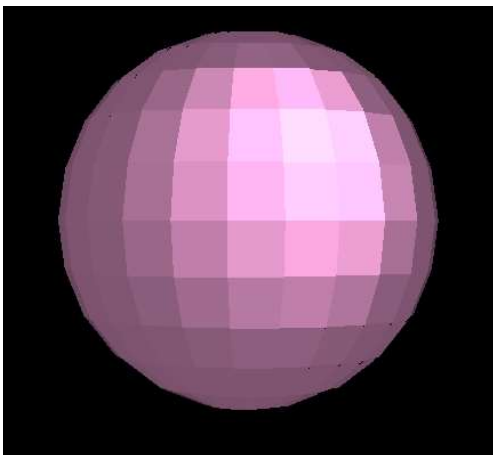
$$I = I_E + K_a I_{ambient} + K_d (N \bullet L) I_{Light} + K_S (V \bullet R)^n I_{Light}$$

2. Constant Shading

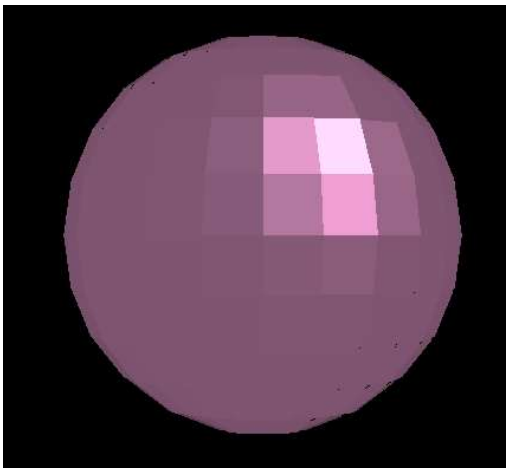
Constant shading is just project 2 implemented with Phong Illumination.



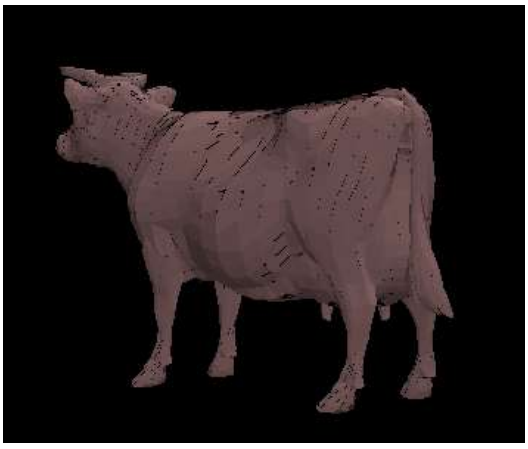
```
#Transformation constants
cam = np.array([0, 1, -3])
pRef = np.zeros(3)

#Illumination constants
n = 100
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([-15, 5, -50])
```

<table>
<tr>
<td>  </td>
<td>

```
#Transformation constants
cam = np.array([0, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 350
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
```

</td>
</tr>
<tr>
<td>  </td>
<td>If we increase n to 2000 to see the difference between Phong and Gouraud shading

```
#Transformation constants
cam = np.array([1, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 2000
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
```

</td>
</tr>
<tr>
<td>  </td>
<td>

```
#Transformation data
cow_pRef = np.array([10, 0, 20])
cow_cam = np.array([-20, 0, -20])

#Illumination constants
n = 500
color = [0.4, 0.3, 0.3]
ka = 0.4
kd = 0.4
ks = 0.95
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 23, -450])
```

</td>
</tr>
</table>

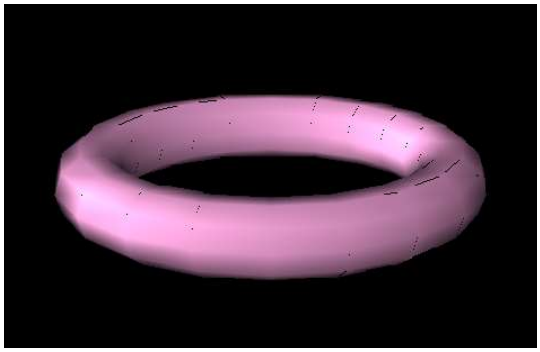
3. Gouraud Shading

The Gouraud Shading is implemented by interpolating the intensity of each pixel from the intensity of the vertices:

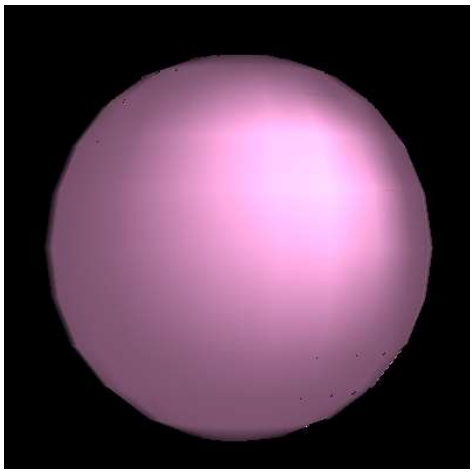
$$l_a = l_1 \frac{y_s - y_2}{y_1 - y_2} + l_2 \frac{y_1 - y_s}{y_1 - y_2}$$

$$l_b = l_1 \frac{y_s - y_3}{y_1 - y_3} + l_3 \frac{y_1 - y_s}{y_1 - y_3}$$

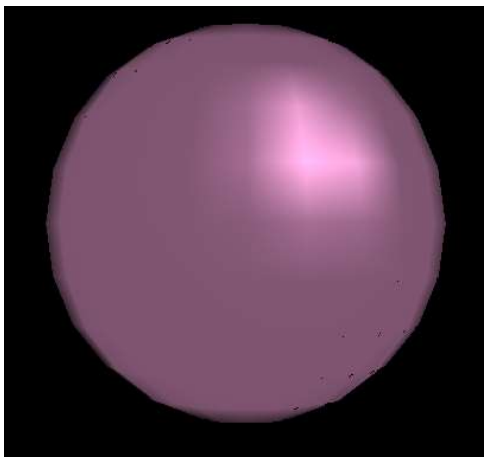
$$l_p = l_a \frac{x_b - x_p}{x_b - x_a} + l_b \frac{x_p - x_a}{x_b - x_a}$$

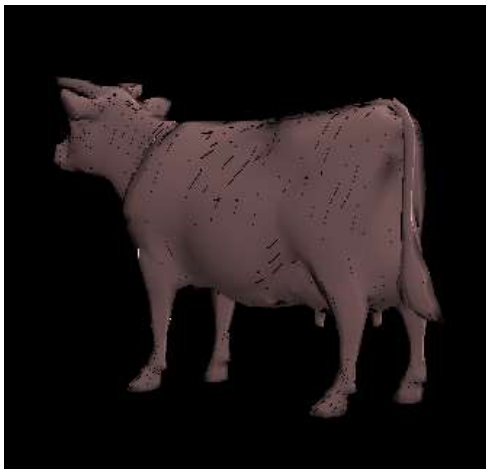


```
#Transformation constants
cam = np.array([0, 1, -3])
pRef = np.zeros(3)

#Illumination constants
n = 100
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([-15, 5, -50])
```



```
#Transformation constants
cam = np.array([1, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 350
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
```

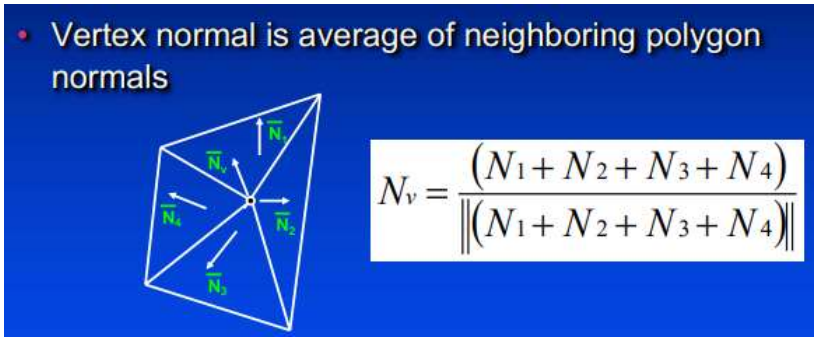
<table>
<tr>
<td>  </td>
<td>If we increase n to 2000 to see the difference between Phong and Gouraud shading
<pre>
#Transformation constants
cam = np.array([1, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 2000
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
</pre>
</td>
</tr>
<tr>
<td>  </td>
<td>
<pre>
#Transformation data
cow_pRef = np.array([10, 0, 20])
cow_cam = np.array([-20, 0, -20])

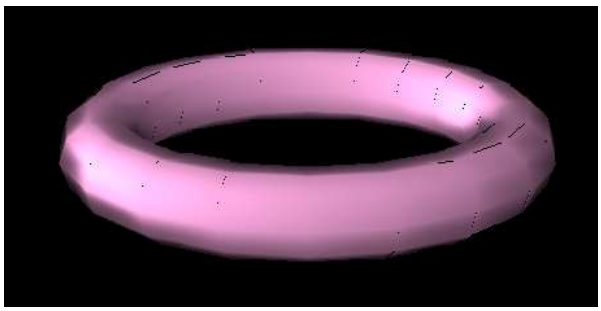
#Illumination constants
n = 500
color = [0.4, 0.3, 0.3]
ka = 0.4
kd = 0.4
ks = 0.95
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 23, -450])
</pre>
</td>
</tr>
</table>

4. Phong Shading

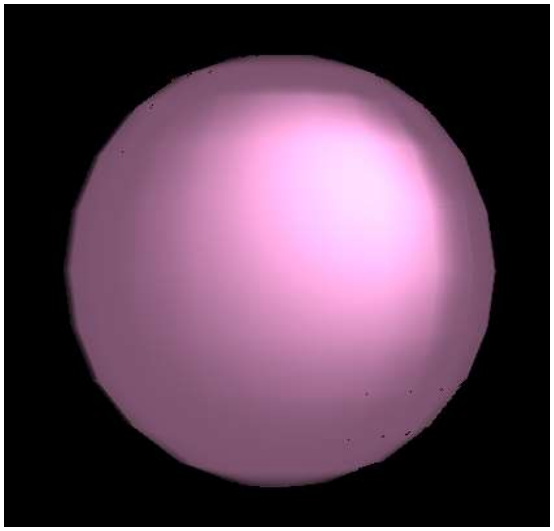
Phong Shading includes finding the vertex normal according to this formula:



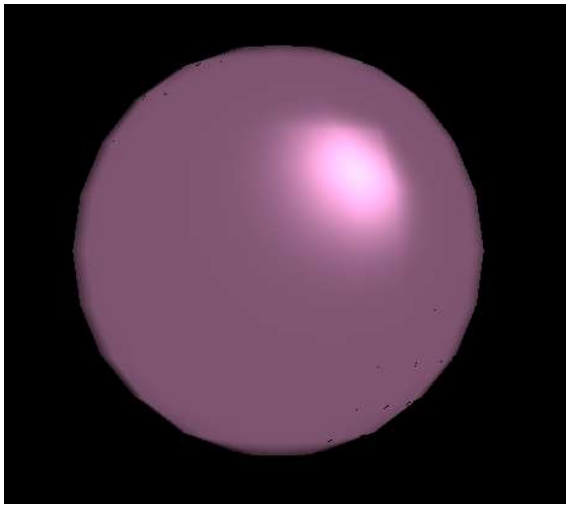
Then the vertex normals are interpolated in the same way as the intensity interpolation for Gouraud.



```
#Transformation constants
cam = np.array([0, 1, -3])
pRef = np.zeros(3)

#Illumination constants
n = 15
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([-15, 5, -50])
```

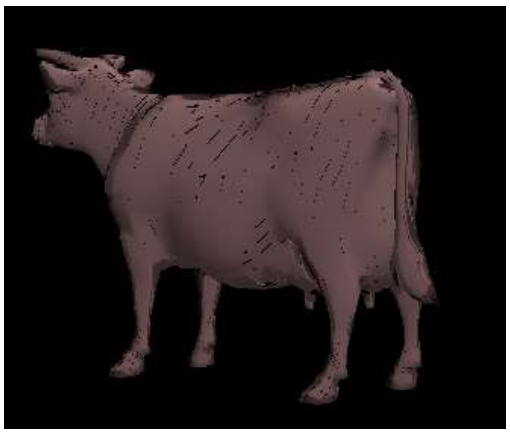


```
#Transformation constants
cam = np.array([1, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 350
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
```



If we increase n to 2000 to see the difference between Phong and Gouraud shading

```
#Transformation constants
cam = np.array([1, 0, -6])
pRef = np.zeros(3)

#Illumination constants
n = 2000
color = [0.9, 0.6, 0.8]
ka = 0.4
kd = 0.4
ks = 0.9
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 10, -250])
```



```
#Transformation data
cow_pRef = np.array([10, 0, 20])  # cow,
bench
cow_cam = np.array([-20, 0, -20])  #
back

#Illumination constants
n = 500
color = [0.4, 0.3, 0.3]
ka = 0.4
kd = 0.4
ks = 0.95
V = np.array([0, 0, 15])
sourceLight = np.array([1, 1, 1])
L = np.array([10, 23, -450])
```