<기초컴퓨터그래픽스 HW4>

컴퓨터공학과 / 20181611 / 김시은

1. 퐁의 조명 모델

물체 표면의 주어진 지점으로 빛이 들어왔을 때, 우리가 바라보고 있는 시선의 방향으로 반사되는 빛의 색을 계산하기 위한 방식 중 하나로 퐁의 조명 모델(Phong's reflection model)을 사용한다. 퐁의 조명 모델은 물리학적으로 정확한 모델은 아니지만, 비교적 계산량이 적고 성능이 좋기 때문에 렌더링 시 기본 모델로 사용한다. 퐁의 조명 모델에서는 엠비언트 반사(ambient reflection), 난반사(diffuse reflection), 정반사(specular reflection) 이렇게 세 가지 형태의 반사를 고려한다.

(1) 엠비어트 반사 (ambient reflection)

엠비어트 빛이란 광원에서 직접 들어오는 빛이 아니라 간접적으로 들어오는 빛을 모델링하기 위하여 사방에 고르게 퍼져 있다고 가정하는 빛을 의미한다.

1) 앰비어트 반사 공식 $$I_\lambda = I_{a\lambda} \cdot k_{a\lambda}$$

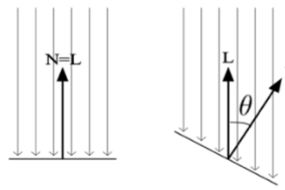
$I_{a\lambda}$는 앰비언트 광원의 색을 의미한다. 광원의 색은 파장 λ의 가시광선 영역에 대하여 설정할 수 있으며, 이 경우 빛의 색은 $I_{a\lambda} = (I_{aR} I_{aG} I_{aB})^t$와 같이 표현된다. $k_{a\lambda}$은 앰비언트 반사 계수를 의미하는데, 이는 물체가 들어오는 앰비언트 빛을 어떻게 반사를 시킬 것인가를 결정하는 물질의 고유한 성질이다.

(2) 난반사 (diffuse reflection)

난반사는 입사 광선을 사방으로 고르게 동일한 밝기로 반사시키는 형태의 반사를 말한다. 물체가 순수하게 난반사만 한다면, 관찰하는 시점이 어디에 있건 같은 밝기로 보인다. 이러한 물체를 이상적인 난반사체(ideal diffuse reflector) 또는 램버트 반사체(Lambertian reflector)라고 한다. 이러한 물체들은 표면에서 빛을 반사시킬 때 램버트의 코사인 법칙을 따른다.

1) 램버트의 코사인 법칙(Lambert's cosine law)

램버트의 코사인 법칙에 따르면 두 벡터 L과 N 사이의 각도를 θ라고 할 때 반사되는 빛 에너지의 양은 $cos\theta$에 비례한다. θ값이 커질수록 $cos\theta$값이 작아지므로, 이 법칙에 따르면 빛이 수직으로 내려 쪼일 때 가장 세게 반사가 되고, 비스듬하게 빛을 비출수록 반사되는 빛의 밝기가 약해진다. 하지만 어디에서 바라보든지 동일하게 보인다.



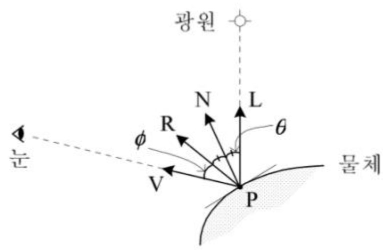
2) 난반사 공식

$$I_\lambda \;=\; I_{l\lambda} \cdot k_{d\lambda} \cdot \cos\theta = I_{l\lambda} \cdot k_{d\lambda} \cdot (N \cdot L)$$

여기서 $I_{l\lambda}$는 광원의 색을 의미하고, $k_{d\lambda}$는 물체가 난반사를 할 때 각 채널의 빛을 어떠한 비율로 반사를 시킬 것인지 결정하는 난반사 계수로, 0과 1 사이의 값을 가진다. 그리고 (N·L)는 N 벡터와 L 벡터의 내적을 의미하는데, 이 둘은 길이가 1인 벡터이므로 둘의 내적값은 $cos\theta$와 같다.

(3) 정반사 (specular reflection)

정반사는 난반사와는 다르게 물체 표면으로 들어오는 빛을 특정 방향을 중심으로 집중적으로 반사시키는 형태의 반사를 말한다. 정반사는 바라보는 지점을 고정한 상태에서 시점을 옮기면 반사되는 빛의 밝기 세기가 급격히 변한다.



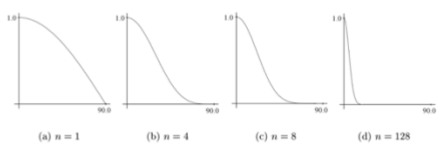
1) 정반사 공식

정반사 공식은 다음과 같다.

$$\begin{aligned} I_\lambda &= I_{l\lambda} \cdot w(\theta, \lambda) \cdot \cos^n \phi \\ &\approx I_{l\lambda} \cdot k_{s\lambda} \cdot \cos^n \phi = I_{l\lambda} \cdot k_{s\lambda} \cdot (R \circ V)^n \end{aligned}$$

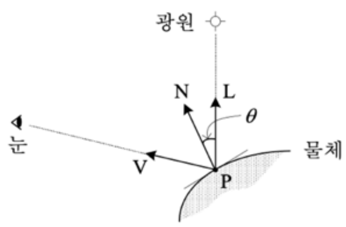
여기서 $I_{l\lambda}$는 광원의 색이고, $w(\theta,\lambda)$는 정반사 계수이다. 정반사 계수는 물질의 고유한 성질로서 들어오는 빛을 어떤 비율로 반사를 시킬 것인가를 결정한다. 일반적으로 $k_{s\lambda} = (I_{sR} I_{sG} I_{sB})^t$로 근사화한 모델을 사용한다.

ϕ값은 정반사 방향 R과 V 사이의 각도인데, R과 V를 단위 벡터라 할 경우 $cos\phi$는 R과 V의 내적과 같은 값을 갖는다. 그리고 n은 정반사 지수를 의미한다. n 값을 통해서 시점 방향이 정반사 방향에서 벗어남에 따라 반사되는 빛의 세기가 약해지는 속도를 계산하는데 사용한다. $cos\phi$값에 n 제곱을 함으로써 약해지는 속도를 조절할 수 있게 된다.



지금까지 **앰비어트 반사, 난반사, 정반사**에 대해 알아보았다. 일반적으로 물체는 순수하게 난반사, 정만사만 일어나는 것이 아니라 두가지 모두를 적절히 혼합하여 빛의 반사형태를 나타낸다. 따라서 **기본적인 퐁의 조명 모델의 식**은 다음과 같다.

$$I_\lambda = \underbrace{\boxed{I_{a\lambda} \cdot k_{a\lambda}}}_{\text{ambient reflection}} + \underbrace{\boxed{I_{l\lambda} \cdot k_{d\lambda} \cdot (N \circ L)}}_{\text{diffuse reflection}} + \underbrace{\boxed{I_{l\lambda} \cdot k_{s\lambda} \cdot (R \circ V)^n}}_{\text{specular reflection}} \quad \ldots (1)$$



퐁의 조명 모델에서 변형을 통해 해프웨이 벡터, 빛의 감쇠 효과, 다중 광원을 나타낼 수 있다.

여기서 **해프웨이 벡터(Halfway Vector)**는 퐁의 조명 모델을 구현할 때 계산량을 줄이기 위해 사용된다.$(R \cdot V)^n$을 .$(N \cdot H)^n$으로 대치해서 계산을 한다. 여기서 H는 해프웨이 벡터(halfway vector)로, $H = \frac{L+V}{|L+V|}$를 의미한다. 해프웨이 벡터는 L과 V의 중간 방향으로서, 물체 표면의

법선 벡터 방향이 이 방향과 일치할 때 가장 강하게 정반사를 한다. 다음으로 **빛의 감쇠효과(Lighting Attenuation Effect)**는 광원과 물체간의 거리에 따른 밝기 조절을 원하는 경우 사용될 수 있다. 점 광원의 밝기$I_{l\lambda}$에 $f_{att}(d_i)$ 를 곱하면 이러한 효과를 낼 수 있다. 따라서 빛의 감쇠 효과를 고려하는 퐁의 조명모델은 $I_\lambda = I_{a\lambda}\cdot k_{a\lambda} + f_{att}(d)\cdot I_{l\lambda}\cdot\{k_{d\lambda}\cdot(N\cdot L) + k_{s\lambda}\cdot(N\cdot H)^n$ 로 나타낼 수 있다. 많은 그래픽스 시스템에서는 $f_{att}(d) = \frac{1}{k_0+k_1\cdot d+k_2\cdot d^2}$를 사용한다.

마지막으로 **다중 광원(Multiple Light Sources)**은 광원이 한 개가 아니라 여러 개가 있을 경우를 처리하기 위한 방법이다. 광원이 m 개가 있을 때의 퐁의 조명 모델은 다음과 같다.

$$I_\lambda = I_{a\lambda}\cdot k_{a\lambda} + \sum_{i=0}^{m-1} f_{att}(d_i)\cdot I_{l_i\lambda}\cdot\{k_{d\lambda}\cdot(N\cdot L_i) + k_{s\lambda}\cdot(N\cdot H_i)^n\} \quad ...(2)$$

여기서 첨자 i 가 붙은 값들은 모두 광원의 색이나 위치, 방향에 영향을 받는 값들이다.

2. OpenGL 에서의 조명 공식

위에서 살펴보았던 퐁의 조명 모델에 기반을 둔 OpenGL 의 조명 공식을 살펴보면 다음과 같다.

$$a_{cm} * a_{cli} + (n \odot \overrightarrow{VP}_{pli})d_{cm} * d_{cli} + (f_i)(n \odot \hat{h}_i)^{S_{rm}} s_{cm} * s_{cli} \quad ...(3)$$

OpenGL 에서도 위에서 설명한 것과 같이 한 광원이 물체에 직접 영향을 미치는 과정을 엠비언트 반사, 난반사, 정반사 등으로 나누어서 생각을 한다. 이러한 조명 공식을 사용하면 그 결과로 RGBA 색이 구해지는 것이다. 각각 하나씩 살펴보면, 먼저$a_{cm} * a_{cli}$ 는 i 번째 광원에 대한 지역 앰비언트 반사 색을 나타낸다. OpenGL 에서는 각 광원에 대한 엠비언트 반사를 전역적인 것과 지역적인 것으로 구분한다.

다음으로 $(n \odot \overrightarrow{VP}_{pli})d_{cm} * d_{cli}$는 식 (1)에서 난반사 색 $I_{l_i\lambda}\cdot k_{d\lambda}\cdot(N\cdot L_i)$에 대응되는 것이다. 우선 $\overrightarrow{VP}_{pli}$를 보면, V 는 현재 조명을 계산하려는 꼭지점의 좌표이고, P_{pli}는 광원의 위치 또는 방향을 나타낸다. 따라서 $\overrightarrow{VP}_{pli}$는 광원에서 빛이 들어오는 방향의 반대 방향에 대해 길이 1 인 벡터를 나타내며, 이는 (1) 식에서 구한 식의 L 에 해당이 된다. $(n \odot \overrightarrow{VP}_{pli})$은 n 과 $\overrightarrow{VP}_{pli}$의 내적 결과가 양수일 때에만 그 값을 사용하고, 그 이외의 경우에는 0 값을 취하는 방식을 사용한다는 의미이다.

$(f_i)(n \odot \hat{h}_i)^{S_{rm}} s_{cm} * s_{cli}$ 는 식 (1)에서 정반사 색 $I_{l_i\lambda}\cdot k_{s\lambda}\cdot(N\cdot H_i)^n$ 과 대응되는 것이다. 여기서 해프웨이 벡터 h_i는 다음과 같이 정의된다.

$$h_i = \begin{cases} \overrightarrow{VP}_{pli} + \overrightarrow{VP}_e, & v_{be} = TRUE, \\ \overrightarrow{VP}_{pli} + (0\ 0\ 1\ 0)^t, & v_{be} = FALSE \end{cases}$$

만약 조명 모델에서 지역 관찰자를 사용한다면, 이는 관찰자가 눈 좌표계의 원점 $P_e = (0\ 0\ 0\ 1)^t$ 에 있는 상황이므로 꼭지점 좌표 V 에서 원점으로 향한 벡터 $\overrightarrow{VP}_e$가 관찰자 방향이 되고, 만약 무한 관찰자를 선택하면 눈 좌표계에서 양의 z_e축 방향인 $(0\ 0\ 0\ 1)^t$이 관찰자 방향으로 사용이 된다. 해프웨이 벡터는 단위 벡터 이어야 하므로, $\hat{h}_i$가 실제로 사용하는 해프웨이 벡터가 된다.

f_i는 0 또는 1 값을 가지는 변수로서, $(n \odot \overrightarrow{VP}_{pli})$가 0 보다 크면 1 값을, 아니면 0 값을 가진다. 여기서 내적 값이 0 보다 크다는 것은 n과 $\overrightarrow{VP}_{pli}$ 사이의 각이 90 도보다 작다는 것을 의미한다.

그리고 i 번째 광원에 대해 앰비언트 반사, 난반사, 정반사 값을 모두 더하면 식 (3)이 된다. 하지만 이 값은 물체에 대한 광원의 직접적인 효과라고 하기에는 불충분하다. 따라서 두개의 값 att_i과 $spot_i$를 곱해 i 번째 광원이 물체에 직접적으로 영향을 미치는 반사 색으로 사용한다.

$$(att_i)(spot_i)[a_{cm} * a_{cli} + (n \odot \overrightarrow{VP}_{pli})d_{cm} * d_{cli} + (f_i)(n \odot \hat{h}_i)^{S_{rm}} s_{cm} * s_{cli}] \quad ...(4)$$

먼저 att_i는 빛의 감쇠 효과를 위한 값으로, OpenGL 에서는 다음과 같이 정의된다

$$att_i = \begin{cases} \dfrac{1}{k_{0i} + k_{1i}\|VP_{pli}\| + k_{2i}\|VP_{pli}\|^2}, & P_{pli}'s\ w \neq 0, \\ 1.0, & otherwise \end{cases}$$

여기서 두번째 경우, 즉 P_{pli}의 w 좌표가 0 이라는 것은 이 광원이 평행 광원이라는 것을 의미한다. 이런 경우에는 att_i를 1 로 설정해 주며, 따라서 식 (4)에 아무런 영향을 미치지 않는다. 왜냐하면 무한 거리만큼 떨어진 평행 광원에 대해서는 빛의 감쇠 효과를 내지 못하기 때문이다. 반면 점 광원을 사용할 때에는 거리에 대한 2 차식의 역수 $\frac{1}{k_{0i}+k_{1i}\|VP_{pli}\|+k_{2i}\|VP_{pli}\|^2}$ 를 사용하여 점 광원에 대한 감쇠 효과를 낸다.

$spot_i$는 i 번째 광원이 스폿 광원일 경우에 이를 처리하기 위한 것이다.

$$spot_i = \begin{cases} (\overrightarrow{P_{pli}V} \odot \hat{s}_{dli})^{S_{rli}} & c_{rli} \neq 180.0 \ \&\ \overrightarrow{P_{pli}V} \odot \hat{s}_{dli} \geq \cos c_{rli}, \\ 0.0, & c_{rli} \neq 180.0 \ \&\ \overrightarrow{P_{pli}V} \odot \hat{s}_{dli} < \cos c_{rli}, \\ 1.0, & c_{rli} = 180.0 \end{cases}$$

c_{rli}는 스폿 광원의 절단 각도로, 디폴트 값은 일반 점 광원을 사용하는 것에 해당하는 180.0 이고, 이때의 $spot_i$값은 1.0 이다. 프로그램에서 이 값을 0.0 과 90.0 사이의 값으로 사용하여 스폿 광원을 사용하겠다고 명시적으로 밝히지 않는 한 스폿 광원 효과는 나타나지 않게 된다. 스폿 광원을 사용하겠다고 설정을 했을 때에는($c_{rli} \neq 180.0$), 점 V 가 절단 각도 범위 내에 들어올 경우에만 적용이 된다. 조명의 위치에서 꼭지점을 향한 방향에 대한 단위 벡터인 $\overrightarrow{P_{pli}V}$와 스폿 조명의 중심축 방향에 해당하는 단위 벡터 $\hat{s}_{dli}$의 내적을 취한 값이 $\cos c_{rli}$ 보다 작은 경우는 점 V 가 스폿 조명 범위 밖에 있는 상황을 의미하기 때문에 는 0.0 값을 갖는다. V 가 스폿 조명의 범위 내에 들어오면 주변으로 갈수록 어두운 효과를 내기 위하여 $(\overrightarrow{P_{pli}V} \odot \hat{s}_{dli})^{S_{rli}}$ 을 사용하여 스폿 조명 효과를 내준다. 지금까지 설명한 식은 한 개의 광원이 물체 표면에 직접적으로 미치는 영향을 표현한 수식이다. n 개의 광원이 있을 때로 확장해보면, 각 광원에 대한 반사 색을 다 더한 후 전역 앰비언트 반사와 물질의 방사 색을 더하면 다음의 OpenGL 기본 조명 공식을 얻게 된다.

$$c = e_{cm} + a_{cm} * a_{cs} + \sum_{i=0}^{n-1} (att_i)(spot_i)[a_{cm} * a_{cli} + (n \odot \overrightarrow{VP}_{pli})d_{cm} * d_{cli} + (f_i)(n \odot \hat{h}_i)^{S_{rm}} s_{cm} * s_{cli}]$$

여기서e_{cm}은 물질의 방사 색을 의미하고, $a_{cm} * a_{cs}$은 전역 앰비언트 반사를 의미한다.