

1. 공식의 각 변수 의미 및 Phong 모델 확장 요약

• OpenGL 기본 조명 공식

$$\mathbf{c} = \mathbf{e}_{cm} + \mathbf{a}_{cm} * \mathbf{a}_{cs} + \sum_{i=0}^{n-1} (\mathbf{att}_i)(\mathbf{spot}_i) [\mathbf{a}_{cm} * \mathbf{a}_{cli} + (\mathbf{n} \odot \overrightarrow{\mathbf{VP}}_{pli}) \mathbf{d}_{cm} * \mathbf{d}_{cli} + (f_i)(\mathbf{n} \odot \hat{\mathbf{h}}_i)^{s_{rm}} \mathbf{s}_{cm} * \mathbf{s}_{cli}]$$

1. \mathbf{e}_{cm} : "재질의 방출 색상"이다. 즉, 해당 재질 자체에서 스스로 발광하는 색상 성분을 나타낸다.
2. \mathbf{a}_{cm} : "재질의 주변광 반사 계수"이다. 즉, 전역 Ambient 빛을 얼마나 반사하는지를 나타낸다.
3. \mathbf{a}_{cs} : "장면에 전체에 깔린 주변광의 색상"이다.
4. $\sum_{i=0}^{n-1}$ 식: "장면 내에 활성화된 최대 n 개의 광원 각각에 대해 계산하여 합산"한다는 뜻이다.
5. \mathbf{att}_i : "광원 i와 픽셀(=fragment) 사이의 거리 기반 감쇠 인자"이다.
6. \mathbf{spot}_i : 광원 i가 스포트라이트 형태일 때, 원추 내부 여부와 중심부 강도를 동시에 반영하는 계수이다.
7. \mathbf{a}_{cli} : "광원 i가 가지는 주변광 색상"이다. 즉, 해당 광원에서 발산되는 주변광 성분의 색상을 나타낸다.
8. \mathbf{n} : "픽셀 표면의 법선 벡터"로, 빛과 표면이 이루는 각도를 계산할 때 사용한다.
9. $\overrightarrow{\mathbf{VP}}_{pli}$: "광원 i에서 픽셀 위치로 향하는 단위 벡터"이며, 길이는 광원과 픽셀 간 거리 d_i 이다.
10. \mathbf{d}_{cm} : "재질의 난반사 반사 계수"로, 빛이 표면에 부딪혀 확산될 때 재질이 얼마나 반사하는지를 나타낸다.
11. \mathbf{d}_{cli} : "광원 i가 가지는 난반사 색상"이다. 빛이 표면에 부딪혀 확산될 때 해당 광원이 주는 색상의 강도를 나타낸다.
12. f_i : " $\mathbf{n} \cdot \overrightarrow{\mathbf{VP}}_{pli} > 0$ "일 때만 스페큘러 계산을 허용하기 위한 스위칭 변수이다.
13. $\hat{\mathbf{h}}_i$: "광원 방향 벡터와 뷰어(카메라) 방향 벡터를 더한 뒤 정규화한 하프 벡터"이다. 정반사 계산 시, 반사 벡터 대신 하프 벡터를 사용하여 연산량을 줄이고 GPU 내장 조명 함수와 호환성을 높인다.
14. s_{rm} : "재질의 광택도"이다. 숫자가 클수록 하이라이트가 작아지고 날카로워진다.
15. \mathbf{s}_{cm} : "재질의 스페큘러 반사 세기"이다. 표면이 금속처럼 얼마나 빛을 반사하는지를 RGB 값으로 나타낸다.

16. s_{cli} : "광원 i 가 가지는 스페큘러 색상"이다. 일반적으로 광원이 반사광으로 만들고자 하는 색상이 이 값이 된다.

• Phong 모델 대비 확장 요약

- Phong 모델은 Ambient, Diffuse, Specular 세 가지 항만 다루며 거리 감쇠나 Spotlight 를 고려하지 않는다.
- OpenGL 공식은 e_{cm} 항을 추가하여 재질 자체 방출 색상을 반영한다.
- $a_{cm} * a_{cs}$ 를 통해 장면 전역 Ambient 와 재질 반사 계수를 분리하였으며, $\sum a_{cm} a_{cli}$ 로 광원별 Ambient 성분을 개별 계산한다.
- Diffuse 항은 $(n \cdot \overrightarrow{VP_{pli}}) d_{cm} d_{cli}$ 형태이고 스페큘러 항은 $(n \cdot \hat{h}_i)^{s_{rm}} s_{cm} s_{cli}$ 로, Phong 모델의 반사 벡터 대신 하프 벡터를 사용한다.
- 스페큘러 계산 전처리 스위치 f_i 를 도입하여 $n \cdot \overrightarrow{VP_{pli}} \leq 0$ 일 때 스페큘러를 0 으로 처리한다.
- 거리 기반 감쇠(att_i)와 스포트라이트($spot_i$)를 추가하여, Phong 모델보다 현실감 있는 조명 효과를 구현한다.

2. Spotlight 효과 값 $spot_i$ 계산

- Spotlight 효과 $spot_i$ 는 광원 i 가 "특정 원추 모양으로만 빛을 비추는" 경우, 픽셀이 그 원추 내부에 있는지 확인하고 중심부와 가까울수록 밝기를 증가시키는 계수이다. 값을 구하는 방법은 다음과 같다.

- 1) 광원 위치를 P_{li} 라고 하고, 픽셀 위치를 P_f 라고 한다.
- 2) "광원→픽셀 방향 벡터"는 $\overrightarrow{P_{li}P_f} = P_f - P_{li}$ 이며, 길이 $\|\overrightarrow{P_{li}P_f}\|$ 로 나눈 단위벡터를 L_i 라고 한다.
- 3) Spotlight 축 방향 단위벡터를 \hat{s}_i 라고 정의하며, 이는 조명 빔을 비추고자 하는 주축이다.
- 4) 두 단위벡터 사이 내적 $L_i \cdot \hat{s}_i = \cos \theta$ 를 계산하면, 각 θ 를 알 수 있다.
- 5) 스포트라이트 컷오프 반각을 c_i 라고 부르고, 이 값이 180° 가 아닌 한 " $\cos \theta \geq \cos c_i$ "를 만족하면 픽셀은 원추 내부에 속한다. 그렇지 않으면 원추 밖이어서 $spot_i$ 를 0 으로 설정한다.
- 6) 원추 내부 픽셀에 대해서는 " $\theta \leq c_i$ "일 때 $(L_i \cdot \hat{s}_i)^{r_i}$ 를 계산하여, 빔 중심에 가까운 정도에 따라 밝기를 지수 형태로 줄인다. 여기서 r_i 는 Spotlight 집중도이다.
- 7) 즉,

$$spot_i = \begin{cases} (L_i \cdot \hat{s}_i)^{r_i}, & \text{if } L_i \cdot \hat{s}_i \geq \cos(c_i) \text{ and } c_i \neq 180^\circ \\ 0, & \text{if } L_i \cdot \hat{s}_i < \cos(c_i) \text{ and } c_i \neq 180^\circ \\ 1, & \text{if } c_i = 180^\circ \end{cases}$$

8) 결과적으로 " $\theta \leq c_i$ "인 원주 내부 픽셀은 $(L_i \cdot \hat{s}_i)^{r_i}$ 값을, 원주 바깥 픽셀은 0 을, 그리고 컷오프 반각이 180° 라면 모든 픽셀에 1 을 부여한다.

3. 빛의 감쇠 효과 값 att_i 계산

- 감쇠 인자 att_i 는 광원 i와 픽셀 사이 거리에 따라 빛의 세기가 자연스럽게 감소하도록 만드는 계수이다.

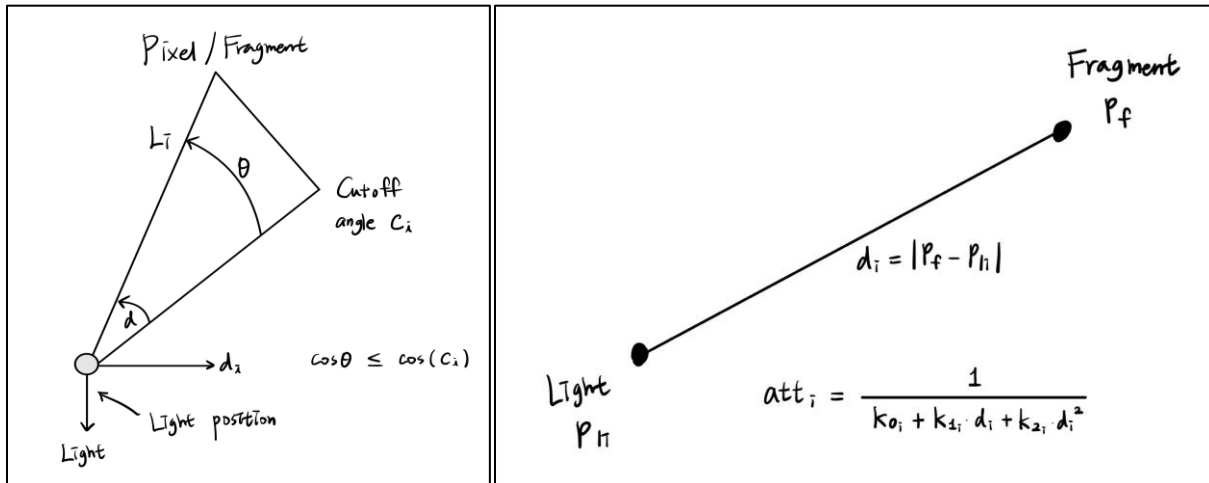
- 1) 광원 위치를 P_{li} , 픽셀 위치를 P_f 라고 한다.
- 2) 두 점 사이 직선 거리 $d_i = \|P_{li} - P_f\|$ 를 구한다.
- 3) 세 가지 감쇠 계수 k_{0_i} (*constant*), k_{1_i} (*linear*), k_{2_i} (*quadratic*)을 정의한다.
- 4) "원근 투영"인 경우에는 아래 식으로 계산한다.

$$att_i = \frac{1}{k_{0_i} + k_{1_i}d_i + k_{2_i}d_i^2}$$

이 때 거리가 커질수록 분모가 커지므로 att_i 가 작아져 픽셀이 어두워진다.

5) "정투영"인 경우에는 $att_i = 1.0$ 으로 설정하여 거리와 상관없이 밝기를 일정하게 유지한다.

6) 예를 들어 $k_{0_i} = 1.0, k_{1_i} = 0.1, k_{2_i} = 0.05$ 이면 거리가 0 일 때 $att_i = \frac{1}{1.0+0+0} = 1.0$ 이고, 거리가 2 일 때 $att_i = \frac{1}{1.0+0.2+0.2} = \frac{1}{1.4} \approx 0.71$ 이 되어 빛이 약해진다.



[2 번 계산 관련 그림]

[3 번 계산 관련 그림]