

## Shading - 블린-퐁 쇼이딩(Blinn-Phong Shading)

쉐이딩은 조명과 재질의 상호작용을 고려해 색을 정밀하게 계산이다.

방법 1) vertex shader에서 vertex의 색을 interpolation하여 Pixel Shader에서 Interpolation한 후 방법

2) shading algorithm 자체를 Pixel Shader에서 계산하는 방법

1번 방법이 계산 수가 적어 더 빠르고 품질은 2번 방법이 좋다.  
최근에 GPU의 발전으로 2번 방법을 많이 사용한다.

Blinn-Phong Shading은 Phong Shading의 속도를 블린시킨 Shading이다.

Ambient Color = 조명과 상호작용하는 물체의 색

전역조명 흐름을 냄 때 모든 빛은 추적하는 것과 비슷이 매우 균일  
Ambient Color는 사용하면서 이를 단순화해 전역조명 흐름을 냄 수 있다.  
전역조명 흐름이 좋아야 하는데 Ambient Color는 빛을 추적하는 것보다 좋다.  
3D 물체의 입체감을 표현하는데 Diffuse와 Specular가 더 중요하다.

물체를 블라인 것은 광원으로부터 나온 빛이 물체에 부딪히어서 눈에 들어온  
물체의 색을 볼 수 있게 된다. 그래서 조명의 세기(strength)로 색을 연관된다.  
즉, 색은 조명에서 볼 때 색이다.

Light = Vec3(1.0f, 1.0f, 1.0f)

Diffuse가 조명과 관련이 있는데 만약 흰색 물체에 광원의 세기인  
 $\text{Vec3}(1.0f, 1.0f, 0.0f)$ 을 하면 Bluer색이 빠지고 노란색이 된다.

DiffuseColor = Light.R \* Diffuse.R + L.G \* D.G + L.B \* D.B

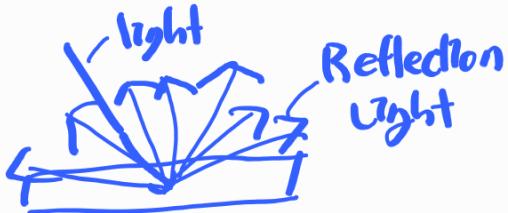
Directional Light는 빛의 세기와 방향만 가진 조명이다.

Directional Light는 위치 대비로 같은 역할을 하는 조명이다.

빛의 방향 =  $\text{Vec3}(0.0f, -1.0f, 0.0f)$  // 위에서 아래로 빛을 낸 조명

## Lambertian Surface

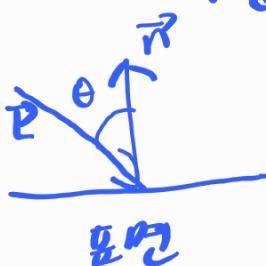
빛이 윤동불동한 표면에 빛이 퍼진다 (반사)



즉, 물체 표면의 빛기는 물체의 표면의  
기울기로 빛의 방향, 2가지로 결정된다.

이 기준은 Normal Vector를 통해

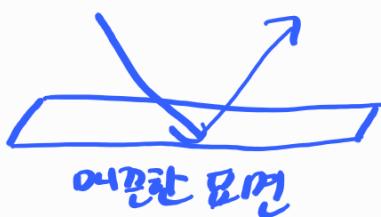
$$I_{\text{diffuse}} = K_d I_{\text{light}} \cos \theta, \quad \text{즉각 각의 빛을 많이 받고 } 90^\circ \text{에 가까울수록 } 0^\circ \text{에 가까워진다.}$$



표면에 수직인 Normal Vector를 사용해  
어떤 면이 빛을 많이 받을 수 있는지를 판별한다.

물체의 색은 빛이 표면에 반사되어서 보인다. 이렇게 반사되어서  
보일지는 물체의 성질이 결정된다

## Specular

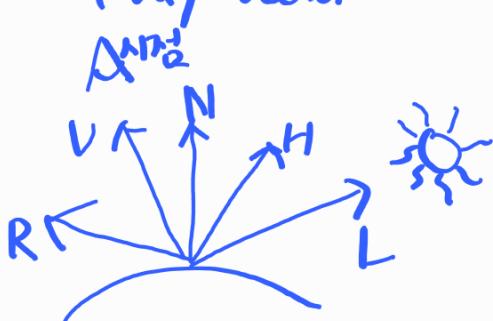


## Diffuse



## Blinn-Phong Model

### Halfway Vector



$V$  = 표면에서 시점으로의 방향  
(증강)

$N$  = Normal Vector

$L$  = 표면에서 조명으로의 방향

$R$  = 원점반사방향

### Phong Model의 R·V 대신

$N \cdot H$  사용해 속도를 올린다

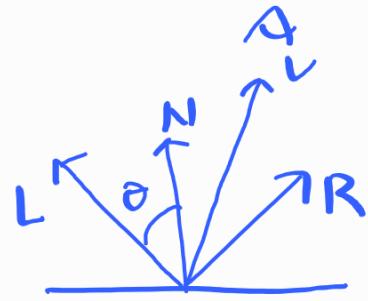
$$H = \frac{L + V}{\|L + V\|}, \quad V는 L의 중간에 있다.$$

## Phong specular Model

$$\text{Specular} = K_s I_{\text{light}} (V \cdot R)^n, n = \text{shiny degree}$$

$$2(N(N \cdot L)) - L = R, K_s = \text{mat. Specular}$$

$$I_{\text{light}} = \text{Light Strength}$$

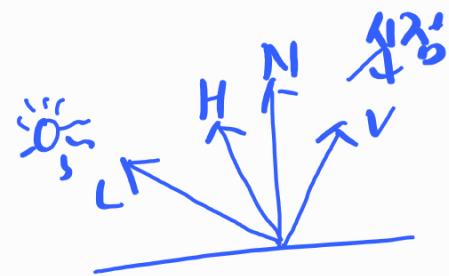


## Blinn-Phong Specular Model

$$\text{Specular} = K_s I_{\text{light}} (H \cdot N)^n, n = \text{shiny degree}$$

$$H = (L + V) / 2$$

$$H_{\text{normalize}} = \text{normalize}(L + V)$$



! 둘째로 회전 시 Normal Vector도 같이 회전해야 되지.

## Blinn-Phong Shader

$$\text{Normal} \rightarrow \text{Normal vector} \text{의 } 90^\circ \text{ 깊이 } = \max(\text{dot}(\text{light}, \text{normal}), 0, \text{of})$$

$90^\circ$ 를 넘어지면 음수가 되는게, 뒤집어 이걸 더해도 0이 되도록 한다.

$$\text{빛의 세기} = \text{Light Strength} \times (\text{빛과 Normal Vector } \geq 0)$$

$$\text{Halfway} = \text{normalize}(L + V)$$

$$\text{Specular} = \text{mat. specular} \times \max((H \cdot N), 0, \text{of})^n, H \cdot N \geq 0 \text{ 일 때}$$

$$\text{pixel} = \text{mat. ambient} + (\text{mat. diffuse} + \text{specular}) \times \text{빛의 세기}$$

Pixel 단위의 Position과 Normal을 구할 때는 모계 중심좌표系를 사용해 Interpolation 해준다.

$$P = w_0 P_0 + w_1 P_1 + w_2 P_2$$

$$N = w_0 N_0 + w_1 N_1 + w_2 N_2$$