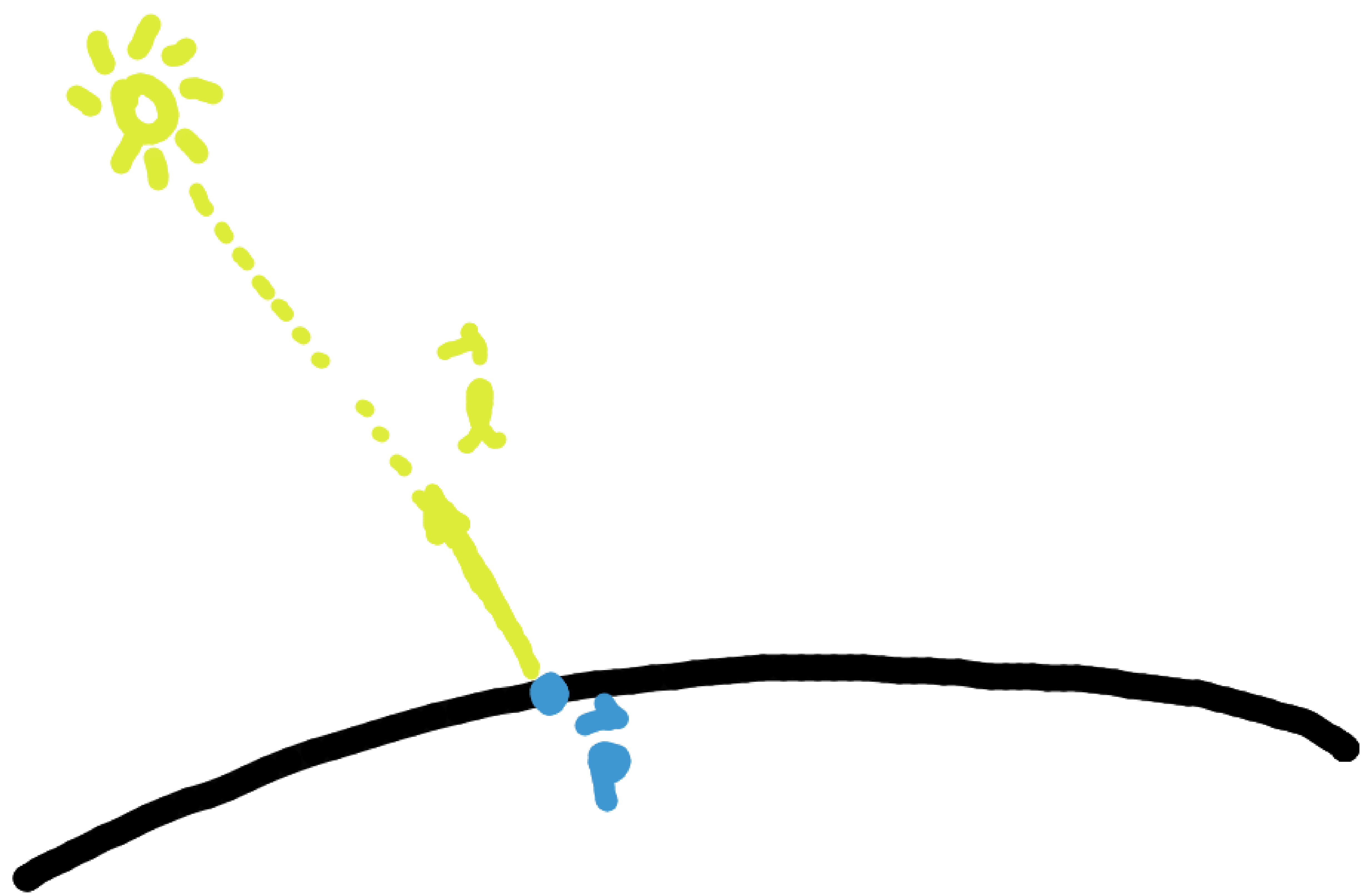
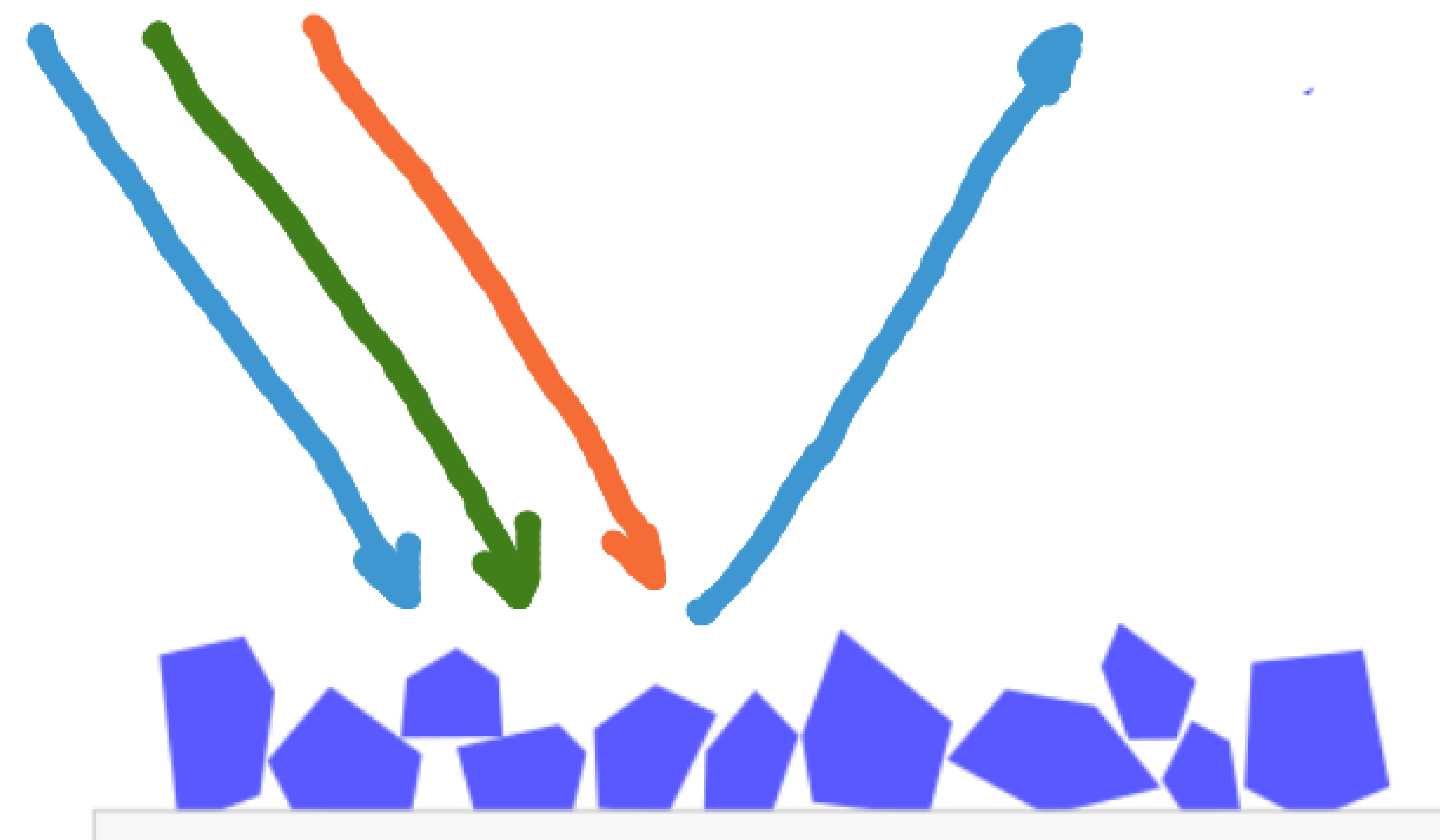
We'll mostly work with a mix of "point" and "directional" lights.



Ingredient #2: Materials reflect certain light components, absorb others.

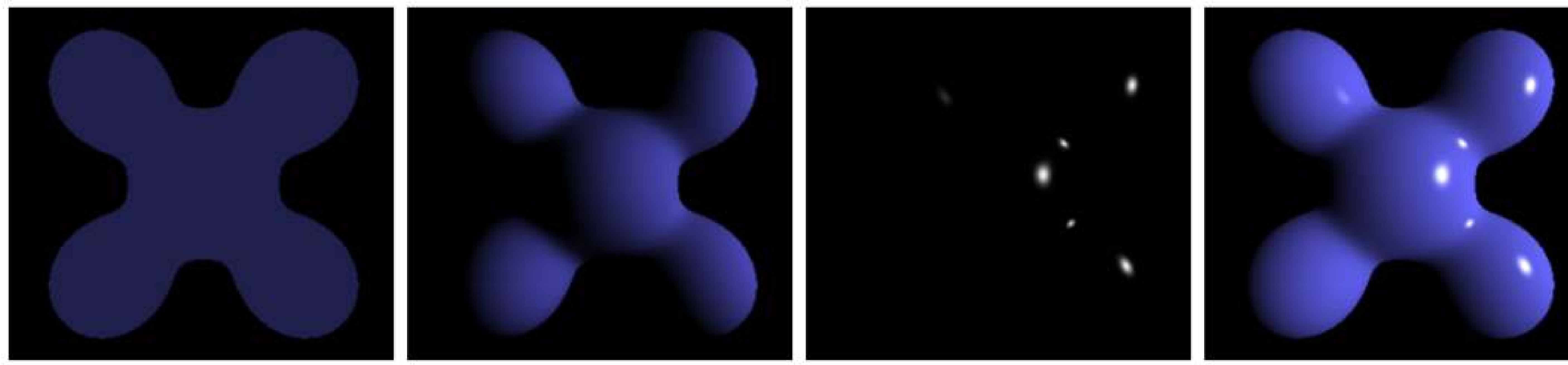


Things we need to consider:

- **albedo:** fraction of incoming light diffusely reflected,
- **properties:** in what direction is light reflected?

We will look at matte-like, plastic-like, mirror-like and also translucent materials.

Our shading equation (*Phong Reflection Model*): ambient + diffuse + specular contributions.



Ambient + Diffuse + Specular = Phong Reflection

$I_a + I_d + I_s = I$

Works really well if we want to render plastic!

illumination color (r,g,b)

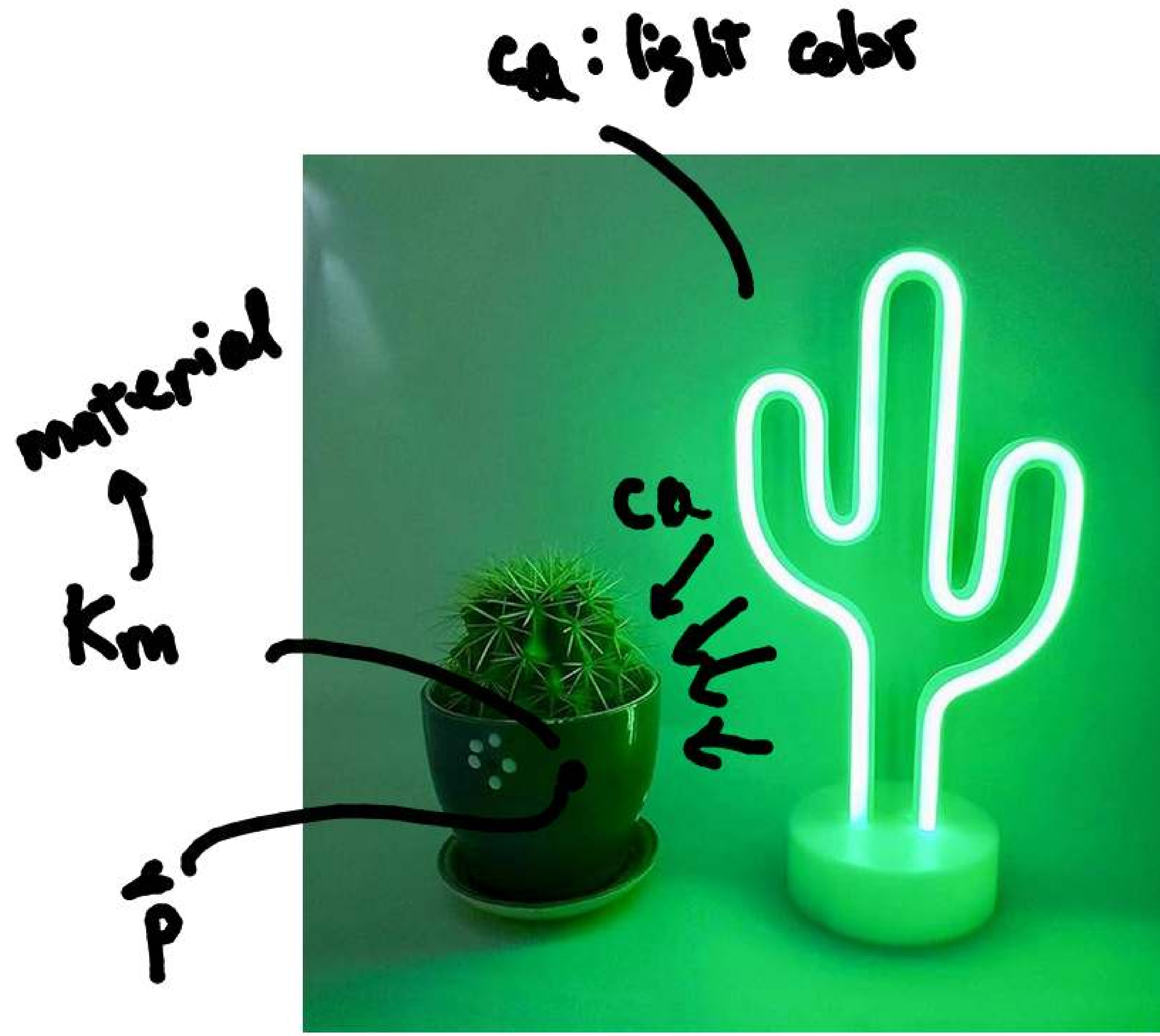
The ambient contribution (I_a) provides background lighting.

$$I_a = c_a : K_m$$

componentwise multiplication

$$\begin{pmatrix} I_{a,r} \\ I_{a,g} \\ I_{a,b} \end{pmatrix} = \begin{pmatrix} c_{a,r} \cdot K_{m,r} \\ c_{a,g} \cdot K_{m,g} \\ c_{a,b} \cdot K_{m,b} \end{pmatrix}$$

vec3.multiply



Light scatters in all directions across diffuse (matte-like) surfaces (I_d).

$\vec{a}\cdot\hat{b} = \|\vec{a}\|\|\vec{b}\|\cos\theta$ always use unit vectors!!!!!

Lambert's law:

$I_d \sim \cos\theta$

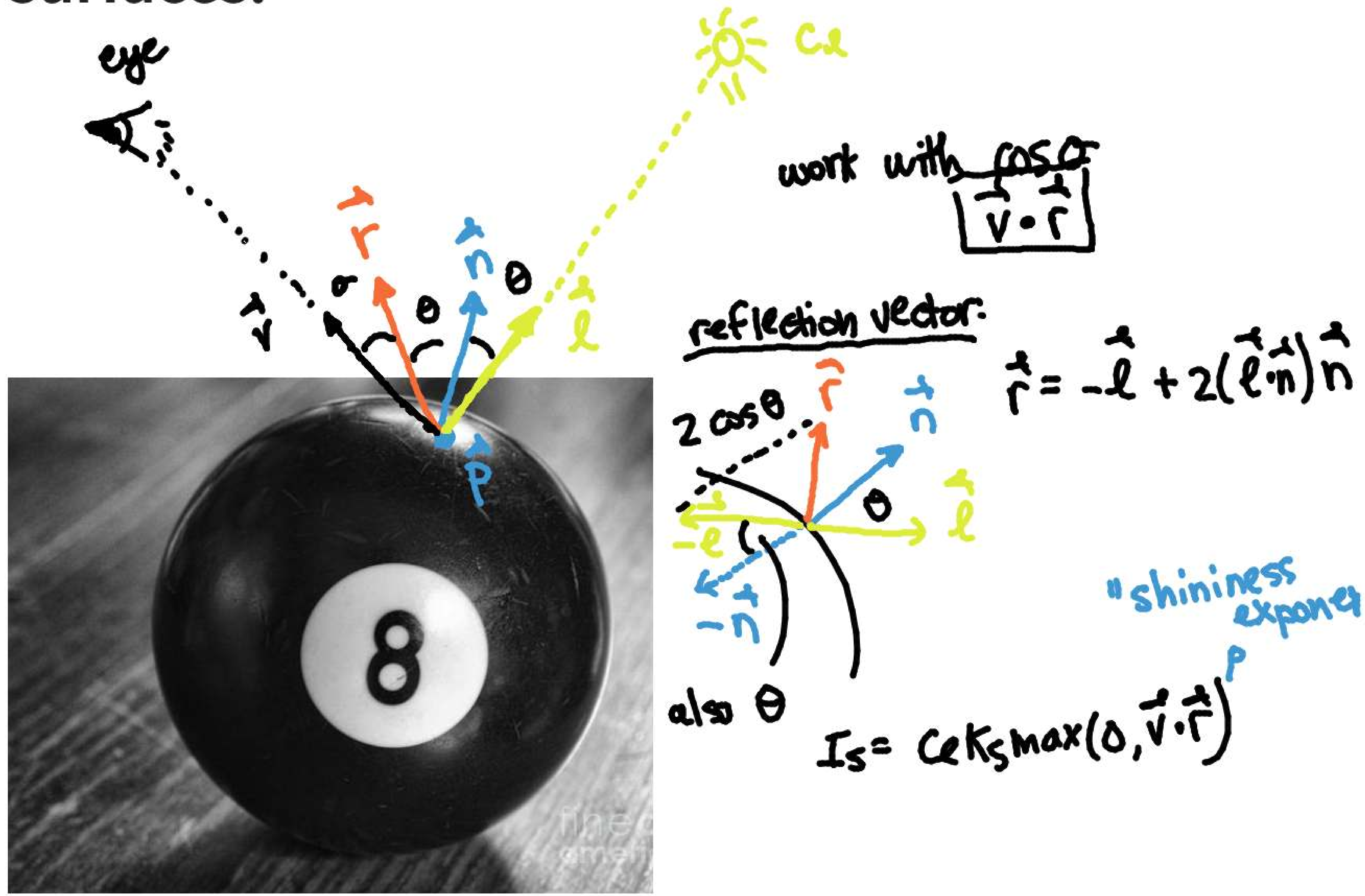
$I_d \sim \vec{n}\cdot\vec{l}$

$I_d = c_l k_m \max(\vec{n}\cdot\vec{l}, 0)$

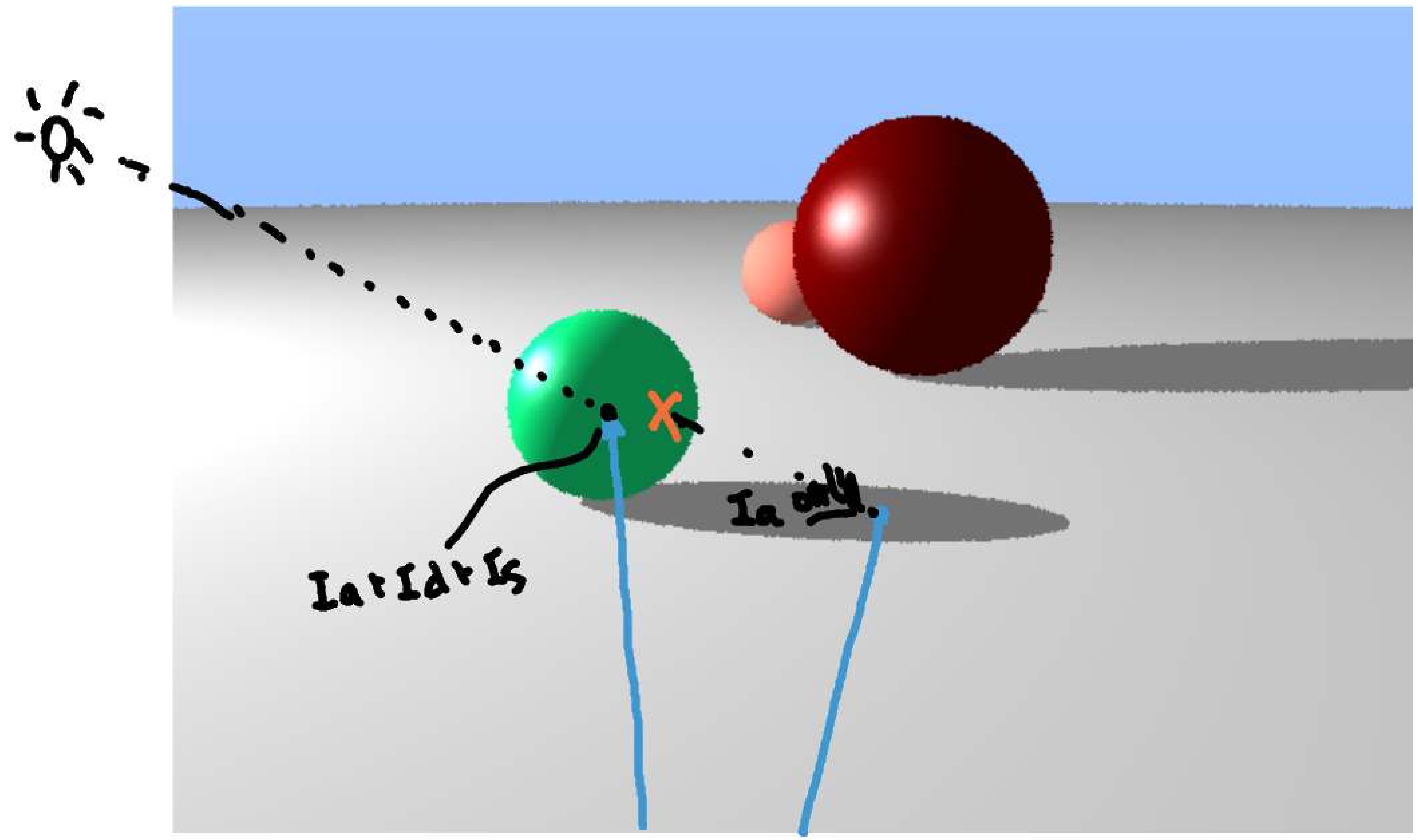
or

$I_d = c_l k_m |\vec{n}\cdot\vec{l}|$

Specular term (I_s) adds a highlight to glossy surfaces.



What about shadows?



Cast a ray from intersection point to light source.
No intersection? use I_a, otherwise add I_d and I_s.

Computing the reflection direction.