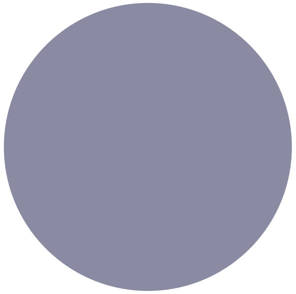
LIGHTING AND SHADING

Dr. Chao Peng

Reference: E. Angel and D. Shreiner: Interactive Computer Graphics 6E © Addison-Wesley 2012

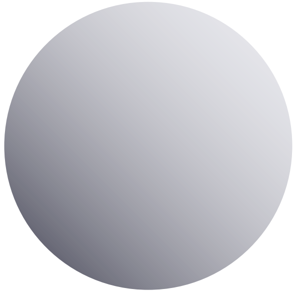
Three-Dimensional Appearance

- When rendering a 3D geometry, you might be disappointed to see the image which looks flat.
- Flat appearance of a geometry fails to show its three-dimensional nature.
- Suppose we build a colored sphere with many polygons, a rendered image would be like



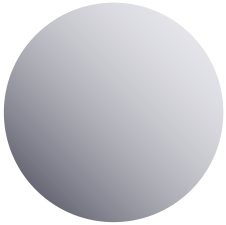
The sphere appears as a uniformly colored circle.

- But we want



In a photograph, the sphere should appear as a circular shape with many gradations, or **shades**, of color.

Shading

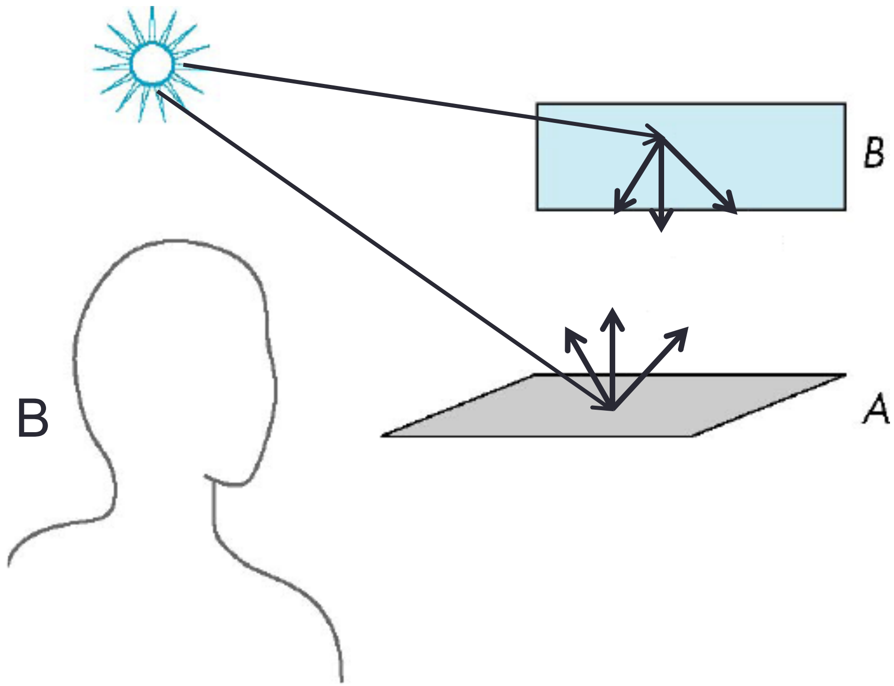
- Why does the image of a real sphere look like  ?
- Considering the interaction between light and the material of surfaces.
  - Light-material interactions cause each point on the surfaces to have a different color or shade.
- To calculate shades, we need:
  - Light sources (e.g., the position of a light source, color of the light, light intensity, etc.)
  - Transformation
  - Material properties of the surface
  - Location of viewer

Physical Process

- How light is reflected from a surface?
  - Light strikes A
    - Some scattered
    - Some absorbed
  - Light strikes B
    - Some scattered
    - Some absorbed
  - Some of scattered light from A strikes B
    - Some scattered
    - Some absorbed
  - Some of this scattered light strikes A

…

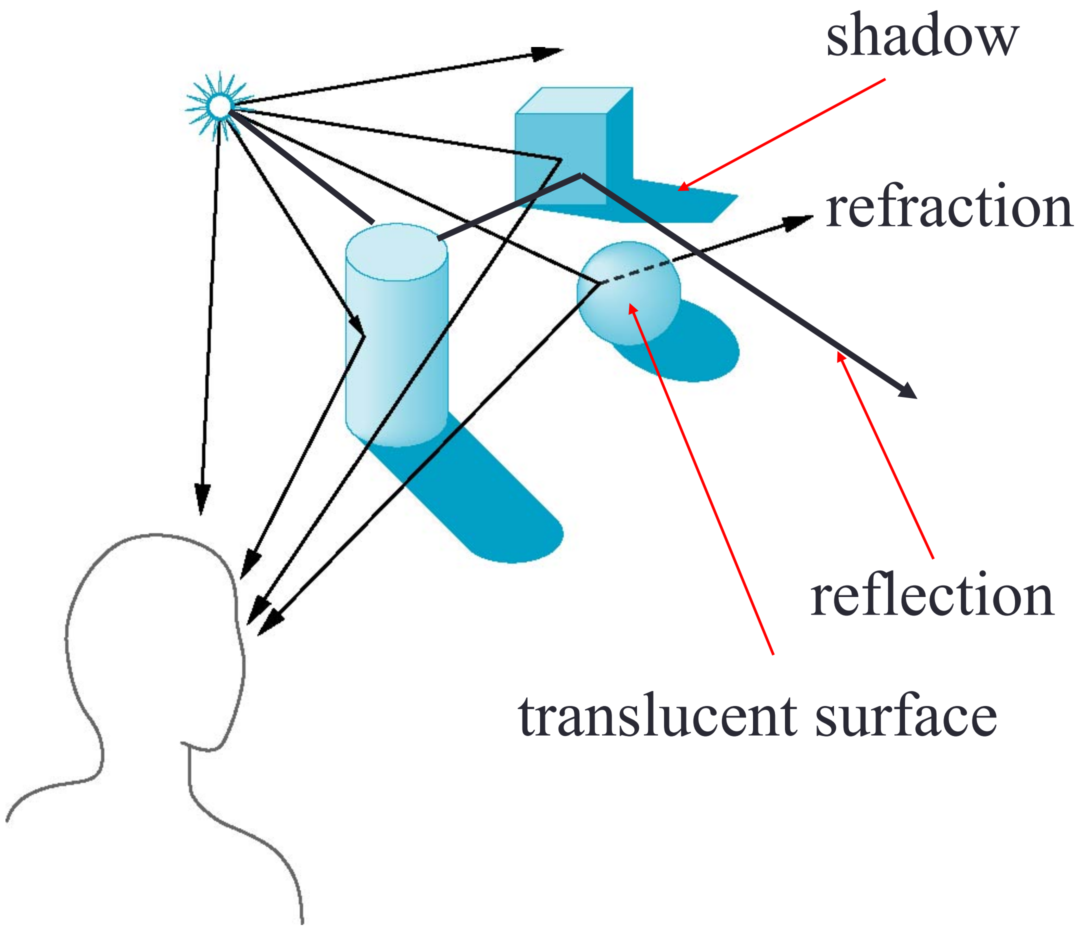
and so on



Challenging

- The infinite scattering and absorption of light can be described by the *rendering equation, but* cannot be solved in general.
- Rendering equation is global and includes:
  - Shadows
  - Multiple scattering from object to object

Global Effects



Local vs Global Calculation for Rendering

- If we want the correct shading, a global calculation is required, involving all objects and light resources in the scene.
- However, the global calculation is incompatible with the standard graphics pipeline, where each geometry primitive should be shaded independently (local calculation).
- In computer graphics, we are satisfied if the shades of an object "looks right".
    - We are interested in the techniques that can approximate the global shading effects.

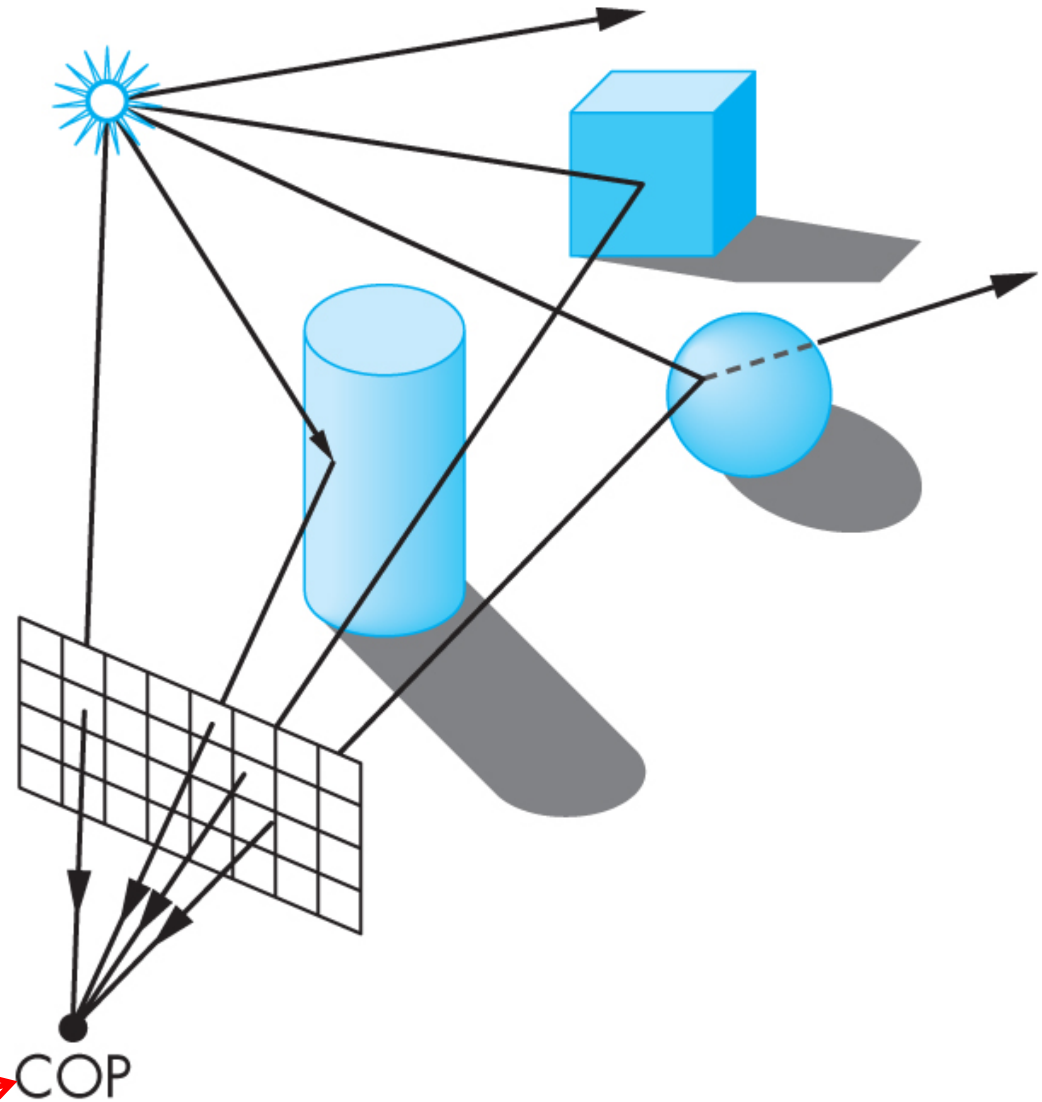
Local Rendering

- Rather than modeling a global energy balance, we consider only single interactions between light sources and surfaces.
- So, we need to do two things:
  - We model the light sources in the scene.
  - We build a reflection model that deals with the interactions between light and surfaces' materials.

Local Rendering

As we introduced in a previous lecture, the viewer sees the scene on the near plane. The viewer is replaced by the near plane (projection plane).

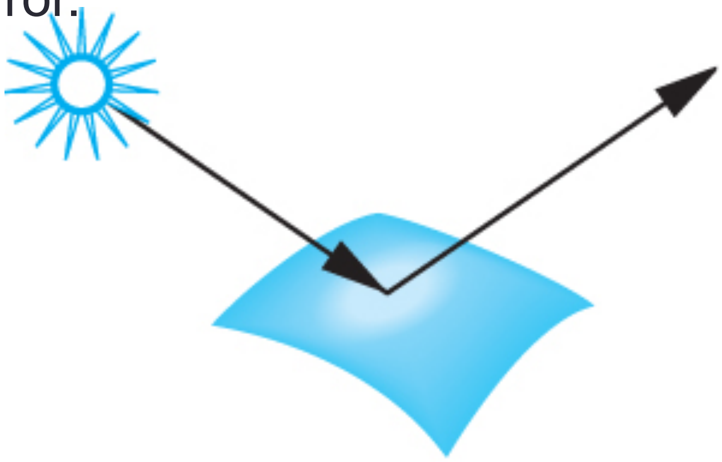
We build light rays between COP and each pixel on the near plane, and trace them back to the light source.



Center of Projection
(Eye position of camera)

Light-Material Interactions

- When the light strikes a surface, if the surface is opaque and smooth, the surface appears shinny.
  - Usually, we can see a highlight spot on the surface.
  - Reflected light is scatted in a narrow range of angles close to the reflection direction.
  - We call it **specular surface**.
  - i.e. metal surfaces, aluminum alloys, mirror.

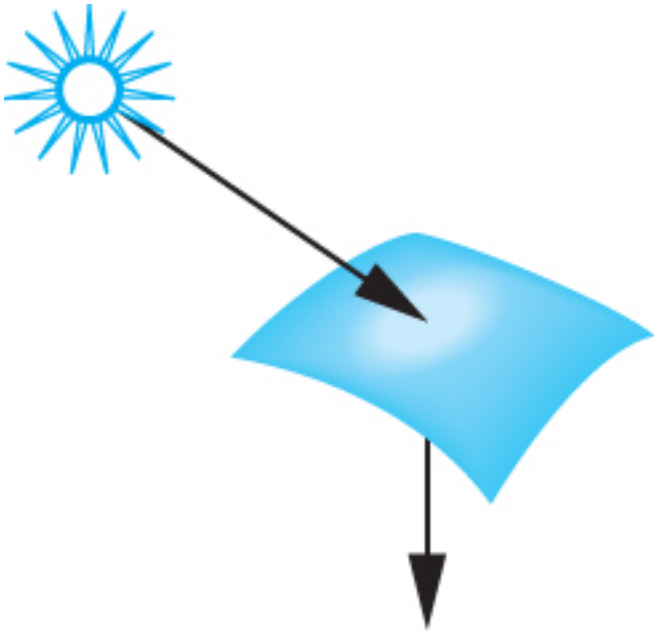


Light-Material Interactions (cont.)

- If the surface is opaque and dull, the surface appears rough.
  - No highlights, it appears as smooth falloff from bright to dark.
  - The reflected light being scatted in all directions.
  - We call it **diffuse surface.**
  - i.e. paper, carpet, etc.

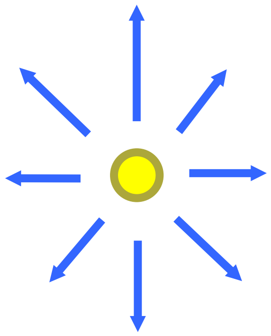
Light-Material Interactions (cont.)

- If the surface is translucent, the surface appears rough.
  - it usually appears in some level of transparent because some of the light is transmitted through the surface.
  - This process is known as **refraction**.
  - We call it **translucent surface.**
  - i.e. glass, water, etc.

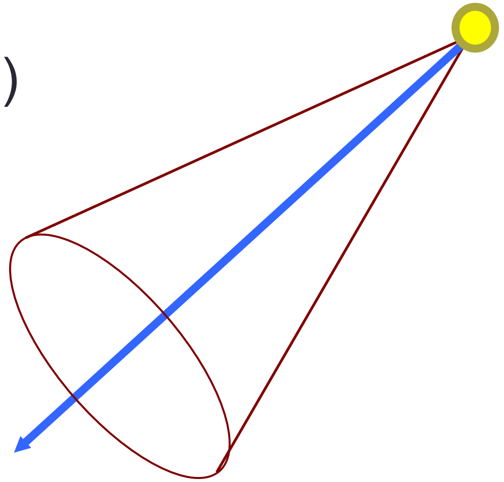


Light Source Types

- Light source has position and color.
- In computer graphics, we usually treat lights as rays emitting from a source.
- The direction of these rays can either be:
  - Omni-directional (e.g., **a point light source** that emits light equally in all directions.)
  - Directional (e.g., **a spotlight source**)



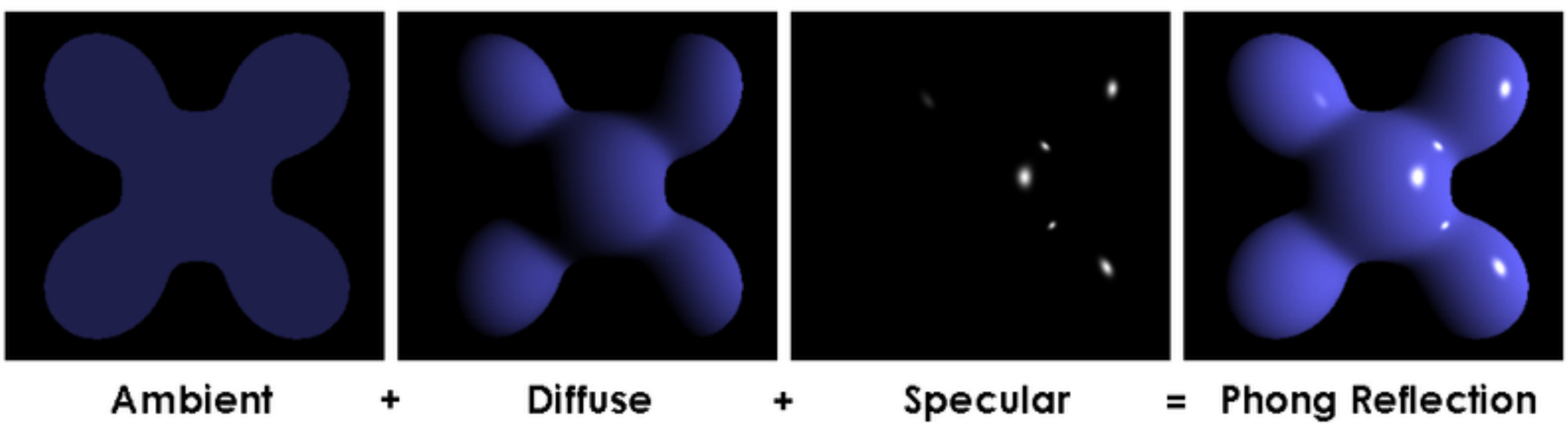
Point Light



Spotlight

Contributions from a light source

- When we generate the shade of an object with a light source, we will first breakdown what a light source does to the object into three different components:
  - Ambient component ($\boldsymbol{I_a}$)
  - Diffuse component ($\boldsymbol{I_d}$)
  - Specular component ($\boldsymbol{I_s}$)

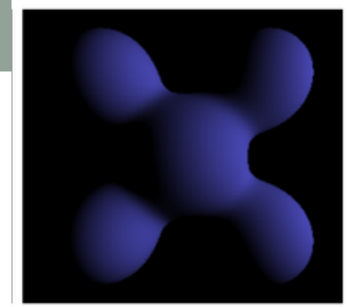


http://en.wikipedia.org/wiki/File:Phong_components_version_4.png

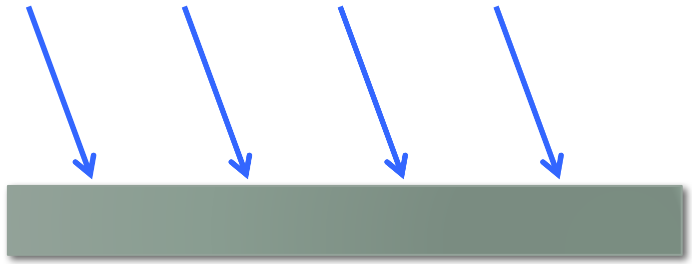
Ambient Component

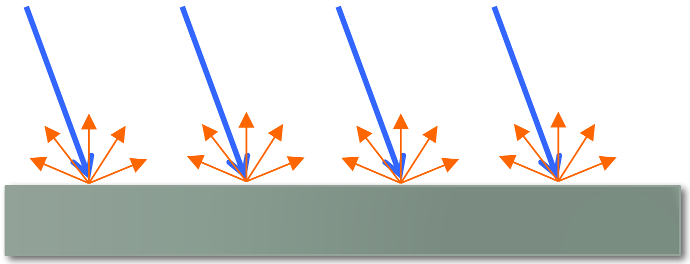
- Let's imagine an example first:
  - In a classroom, the light sources are positioned to provide uniform illumination throughout the room.
  - To achieve such illumination, inter-reflections among all objects and light resources in the room are simulated.
  - Basically, one object affects the rendering of another object, which is known as **global illumination**.
- Rendering a scene with the simulation of global illumination is extremely time-consuming!
- To achieve a uniform light level in the room, we introduce the **Ambient light**, also known as **Background light**.
- Ambient light is a HACK, and represents the approximate contribution to the rendering of the entire scene, regardless of the positions of light sources and objects.

Diffuse Component



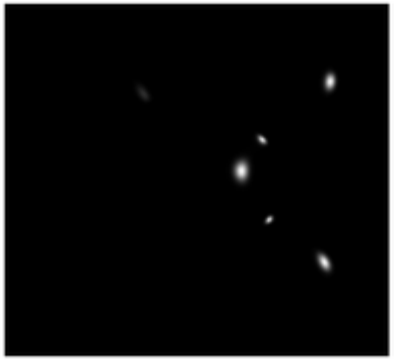
- Diffuse component describes the interaction between light source and the surface, and its color is usually affected by the nature of the surface material.
- At a microscopic level, diffuse surfaces are very rough. This means that a light ray coming in has an equal chance of being reflected in any direction.
- Diffuse component represents **the amount of light that illuminates the surface, regardless of the direction from a point of the surface to the location of the eye** (viewing direction).





Specular Component

- The contribution of the specular component can be thought of "shinny highlight".
- It is more mirror-like and is highly directional, where light rays do not scatter but instead is reflected directly from the surface.
- Usually, the color from the specular component is the same as the light source.
- The specular component is **view-dependent**.

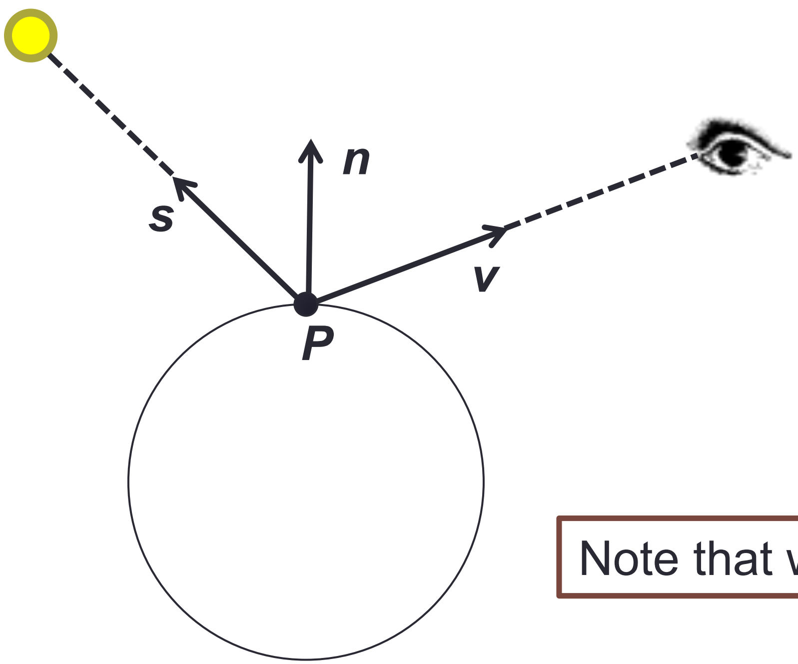


Local Illumination Model

- Now, we learn how to use a light source to render an object.
- The light striking at a point on a surface is a combination of *ambient* + *diffuse* + *specular*
- Correspondingly, the material properties of the surface consist of three **reflection coefficients**, so that the shade of a point on the surface has:
  - *Ambient* = $I_a \rho_a$ **---** ρ_a is the ambient reflection coefficient
  - *Diffuse* = $I_d \rho_d$ **---** ρ_d is the diffuse reflection coefficient
  - *Specular* = $I_s \rho_s$ --- ρ_s is the specular reflection coefficient

$$\rho_a, \rho_d, \rho_s \in [0,1]$$

Local Illumination Model



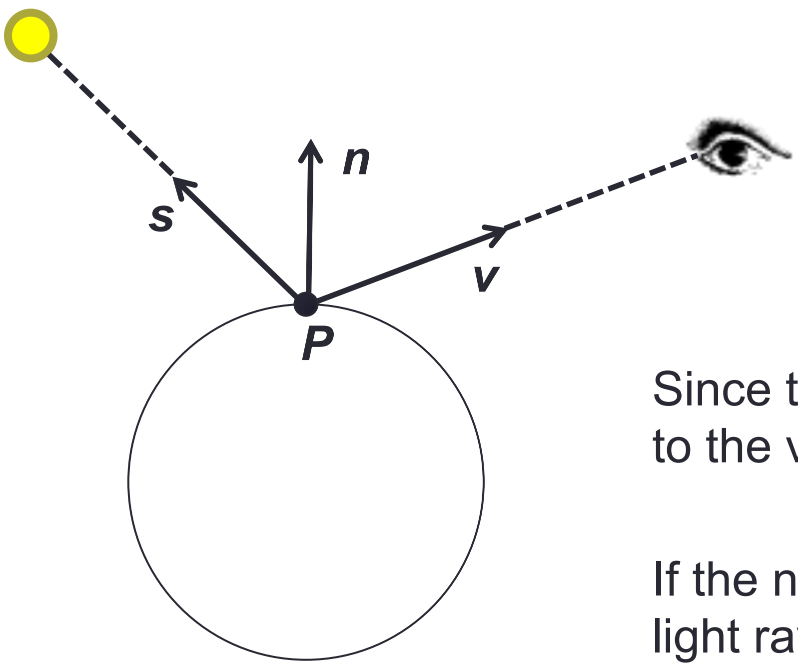
In order to find the amount of light that reaches the eye from the point ***P***, three vectors are required:

1. The normal vector ***n*** to the surface at ***P***
2. The vector ***v*** from ***P*** to the eye
3. The vector ***s*** from ***P*** to the light source

Note that we assume ***s***, ***n*** and ***v*** are normalized.

Local Illumination Model: Diffuse

- Now, let's see how the diffuse component contributes to the shading?



The contribution of diffuse component is calculated according to **Lambert's Law** (after Johann Heinrich Labmert 1728 - 1777)

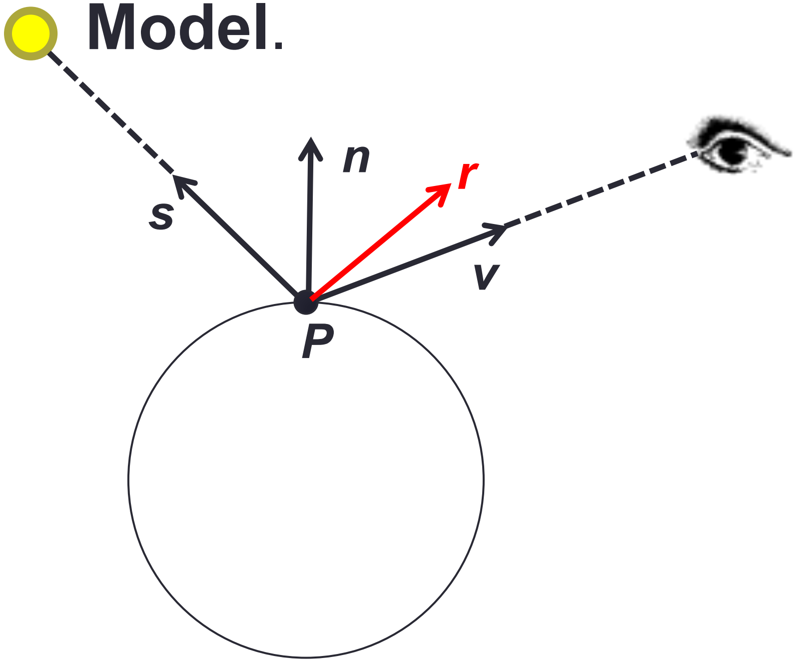
Since the diffuse is independent to the viewing direction, we have $I_d \rho_d (s \cdot n)$

If the normal vector of point P is away from the light ray, we want to evaluate the contribution to 0.

The contribution of diffuse component = $I_d \rho_d \max\left(0, (s \cdot n)\right)$

Local Illumination Model: Specular

- The contribution of specular component represents the mirror-like reflection. It is calculated according to **Phong Model**.

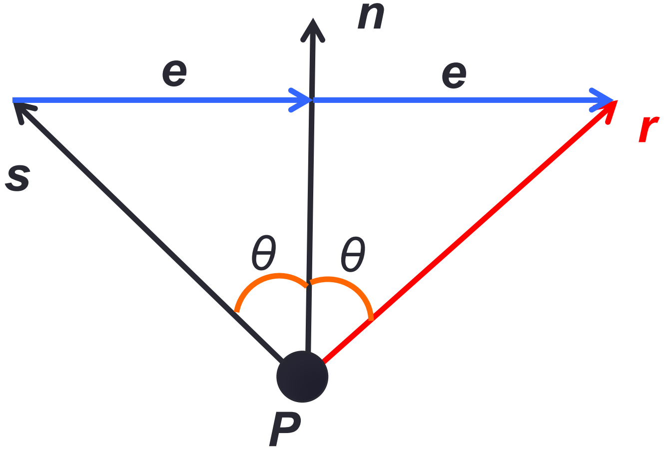


The amount of light reflected is greatest in the direction of *perfect mirror reflection* (vector ***r*** in the figure).

If we can know the amount of light on vector ***r***, we then can calculate its contribution to ***v***.

So, let's take a look how to get ***r***.

Local Illumination Model: Specular (cont.)



We have:

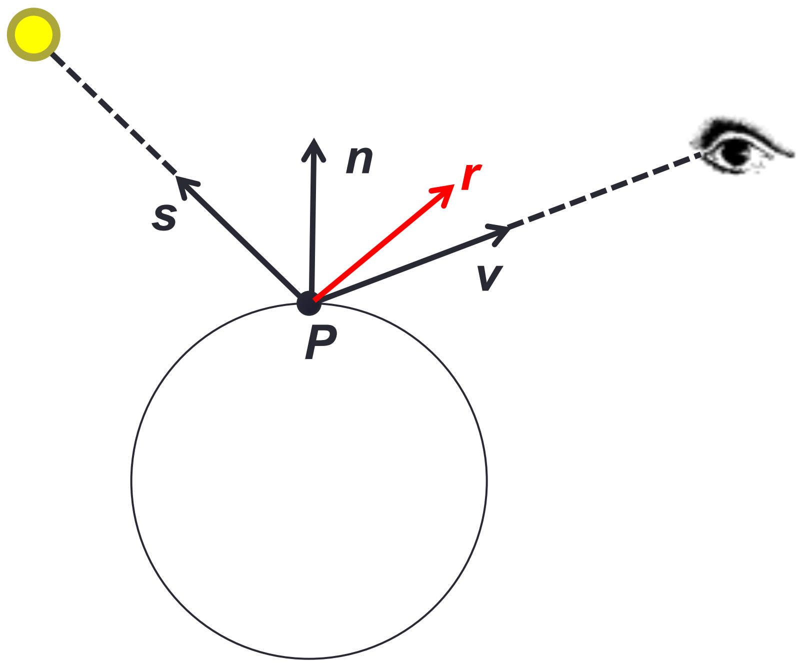
$$r = (n \cdot s)n + e$$

$$e = (n \cdot s)n - s$$

So:

$$r = 2n(n \cdot s) - s$$

Local Illumination Model: Specular (cont.)

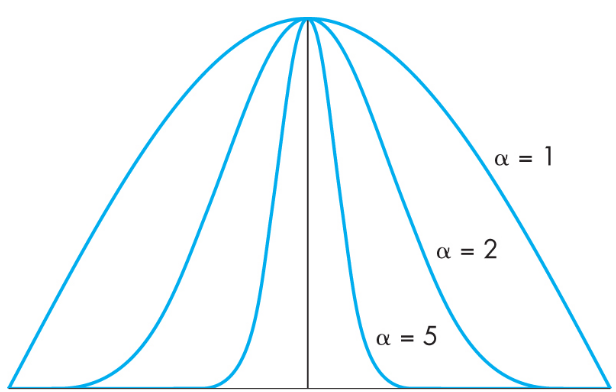


The contribution of specular component =

$$I_s \rho_s \max\big(0, (r \cdot v)\big)^{\alpha}$$

For the surface that are shinny but not true mirrors, the amount of light reflected falls off as the angle ***φ*** between ***r*** and ***v*** increases.

The fall-off amount is approximated as some power ***α*** of the cosine of ***φ***.



Combine Light Contributions

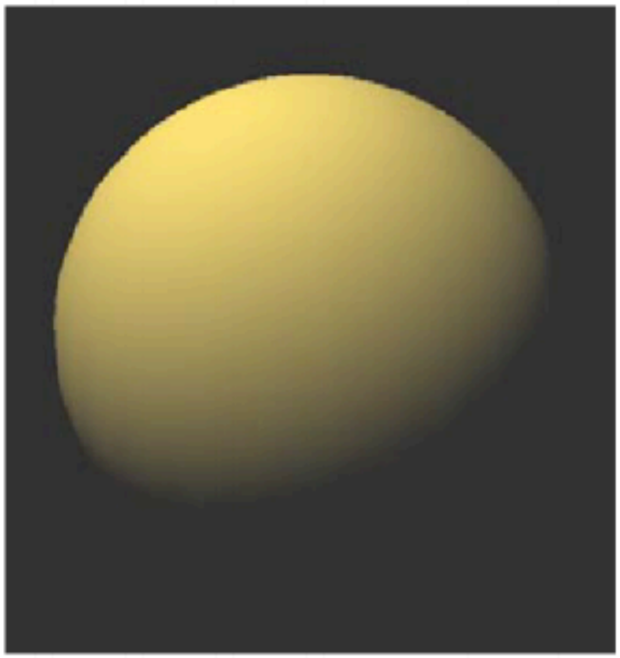
- Now, let's sum the contributions of ambient, diffuse and specular, to the total amount of the light ***I*** captured by the eye at the point ***P***.

$$I = ambient + diffuse + specular$$

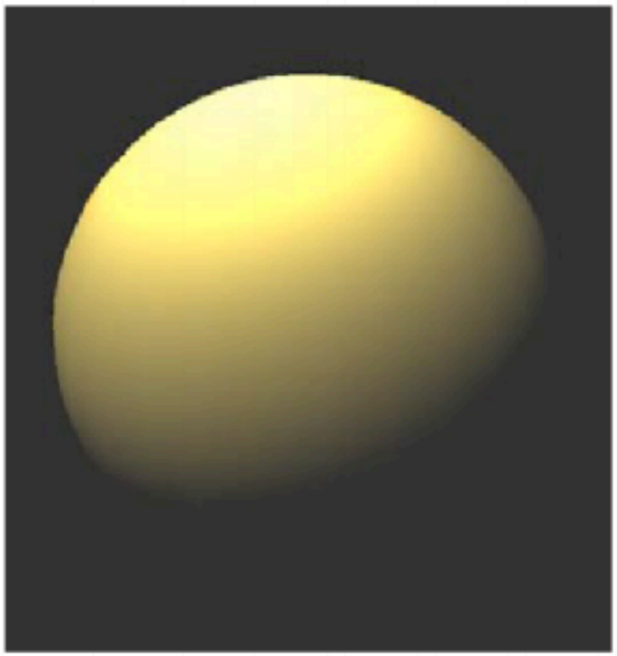
$$I = I_a \rho_a + I_d \rho_d \times lambert + I_s \rho_s \times phong^{\alpha}$$

$$lambert = \max\left(0, (s \cdot n)\right) \qquad phong = \max\left(0, (r \cdot v)\right)$$

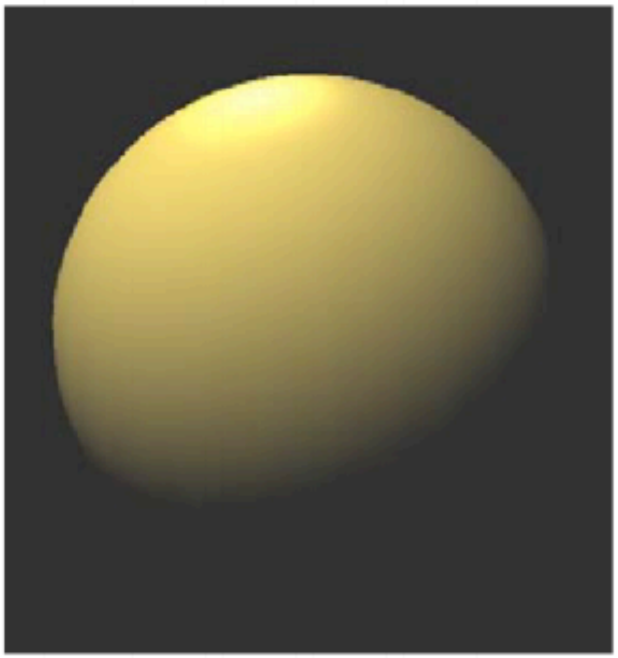
Example of the Local Illumination Model



Diffuse



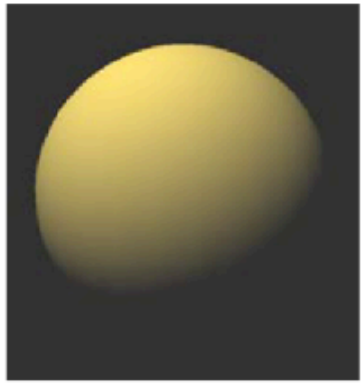
Diffuse + Specular with α=5



Diffuse + Specular with α=50

Example of the Local Illumination Model (cont.)

Diffuse



Specular

Ambient

Combination of the three

Add Colors

- We know that a light source should have intensity for the independent red, green and blue components.
- The intensity should be defined as

$$I = \begin{bmatrix} I_r \\ I_g \\ I_b \end{bmatrix}$$

- Same to the colored surfaces, where a surface should have three color component, red, green and blue.

Add Colors (cont.)

- When dealing with the colored light sources and surfaces, we calculate each color component individually and simply ad them to form the final color.

$$I_r = I_{a\_r}\rho_{a\_r} + I_{d\_r}\rho_{d\_r} \times lambert + I_{s\_r}\rho_{s\_r} \times phong^{\alpha}$$

$$I_g = I_{a\_g}\rho_{a\_g} + I_{d\_g}\rho_{d\_g} \times lambert + I_{s\_g}\rho_{s\_g} \times phong^{\alpha}$$

$$I_b = I_{a\_b}\rho_{a\_b} + I_{d\_b}\rho_{d\_b} \times lambert + I_{s\_b}\rho_{s\_b} \times phong^{\alpha}$$