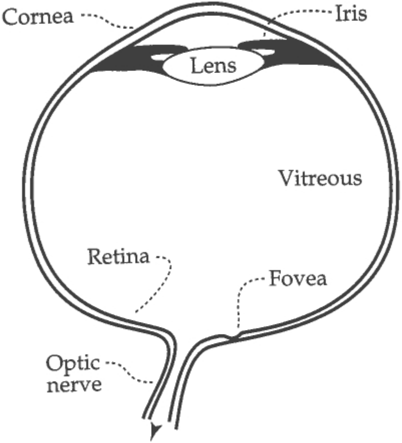
Appearance of Objects

- object appearance depends on three factors:
  - lighting
  - material properties
  - viewer properties
- for the most part, graphics techniques do not account for the properties of the viewer
  - example: for the synthetic camera, properties of film are not modeled
  - example: for a human observer, properties of human visual system (eye and brain) are not modeled

Human Vision

- it is useful (and interesting) to study human vision to understand the generation and appearance of computer images
- vision is the inverse problem of graphics
  - graphics: how do we describe the 3D (4D if we consider time) world to produce a 2D image?
  - vision: given a 2D image, what can we infer about the 3D/4D world?
- the eyes and brain comprise the human visual system
  - we will only study the eye

Structure of the Eye



Structure of Eye (cont)

- cornea
  - clear coating over front of eye
  - two major purposes:
    - protects internal structure
    - focusing of light (cornea is strongest focusing element in the eye)
- iris
  - colored annulus between cornea and lens
  - changes the size of the pupil to allow more or less light into the eye

Structure of the Eye (cont)

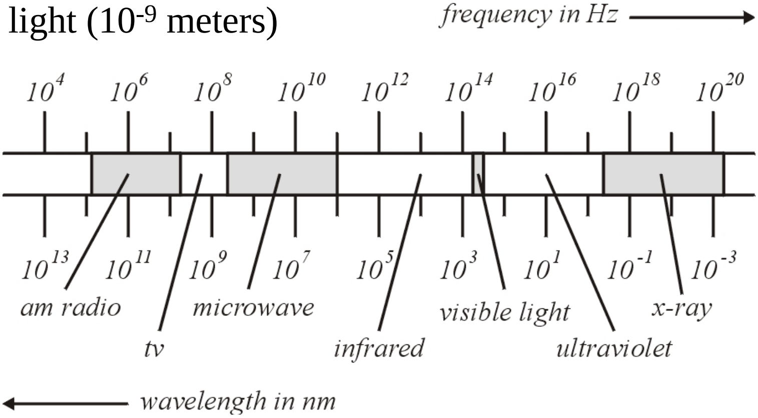
- lens
  - clear elastic focusing element
  - muscles stretch and compress the lens to help focus light (elasticity diminishes with age)
- retina
  - thin layer of cells covering approximately 200 degrees on the back of the eye
  - two types of photosensitive cells in retina:
    - cones: sensitive to color light
    - rods: sensitive to light intensity only (not color) but 10 times more sensitive than cones

Structure of the Eye (cont)

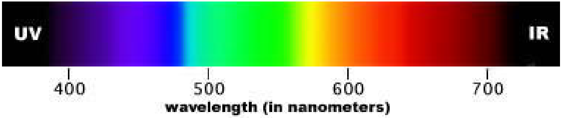
- fovea
  - very small region of the retina with the densest collection of cone cells (147,000 cones/mm)
    - some hawks have 1,000,000 cells in the same area (can see a small animal at a distance where a human could not even see the hawk)
  - visual field is centered on fovea
  - rods start to appear at the edge of the fovea and increase rapidly in density away from fovea
    - night vision is often better slightly away from the center of the visual field

The Nature of Light

- light is an electromagnetic phenomenon (like radio waves, microwave, x-rays, etc)
- waves are characterized by wavelength (or frequency) usually measured in nanometers for light (10^{-9} meters)



Visible Spectrum

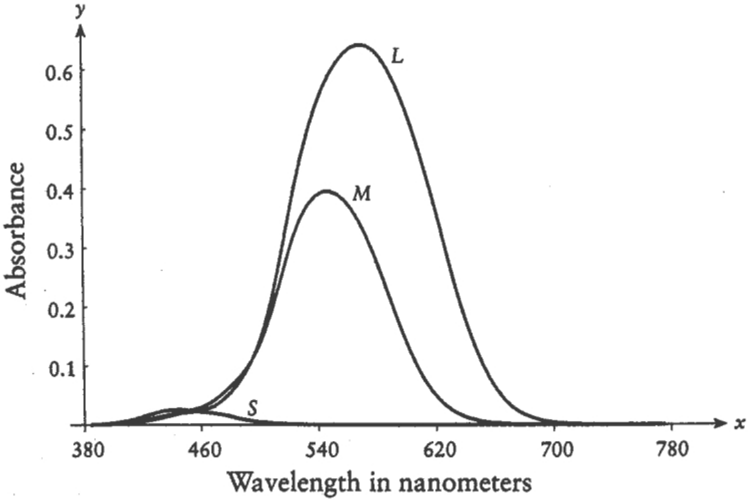


www.handprint.com/HP/WCL/color1.html

- visible spectrum approximately 400-700nm
- light does not have color
  - the sensation of color is perceived
- color perception starts with cone cells

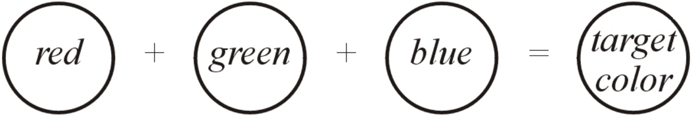
Tristimulus Theory

- 3 different cone cells respond to certain regions of the visible spectrum



Tristimulus Theory (cont)

- only have 3 (?) different types of cone cells
  - this suggests that a properly blended combination of three different colors can reproduce any color light we perceive
  - mantis shrimp has 10 different color receptors
- a good choice of colors is red, green, and blue
- if you take a red, green, and blue light can you match any color light?
  - C = rR + gG + bB ?

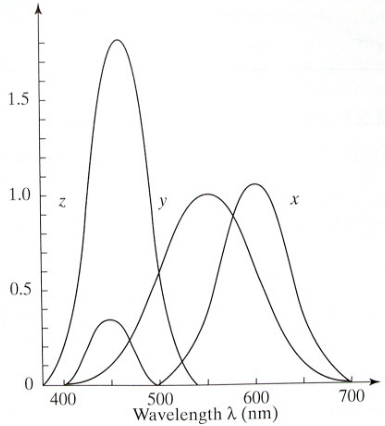


Tristimulus Theory (cont)

- many target color lights cannot be matched
  - what if we add red light to the target light?
    - C + rR = gG + bB
      - this works!
  - mathematically same as adding a negative amount of red light
    - C = -rR + gG + bB
- picture of color-matching functions r, g, b in Hill Figure 12.6

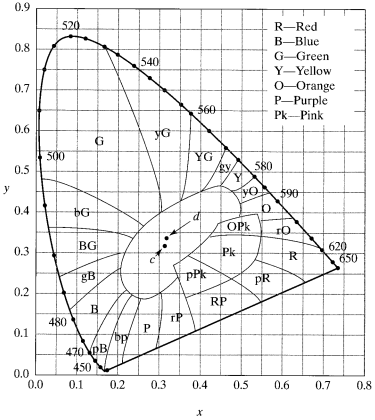
CIE Color Matching Functions

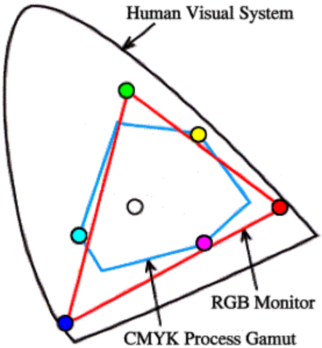
- Commission Internationale de L'Eclairage (CIE) defined the standard observer (1931)
- invented three primary color lights (X, Y, and Z) that when added in positive amounts can match any perceivable color light
  - C = xX + yY + zZ
  - Hill Figure 12.8



CIE Chromaticity Diagram

- coefficients x, y, z define a 3D color space
- a 2D slice of this space yields the CIE chromaticity diagram (Hill Figure 12.10)





viz.cac.psu.edu/sem_notes/color_2d/html/working_with_color.html

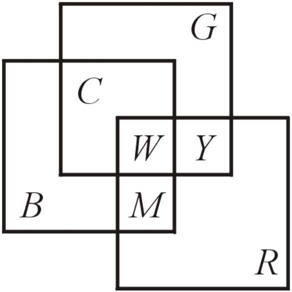
RGB Color Space

- most common color space in graphics is red-green-blue (RGB) color space
  - reason: easy to display on color monitors (which use red, green, and blue phosphors)
- C = rR + gG + bB where

$$0 \le r \le 1, \quad 0 \le g \le 1, \quad 0 \le b \le 1$$

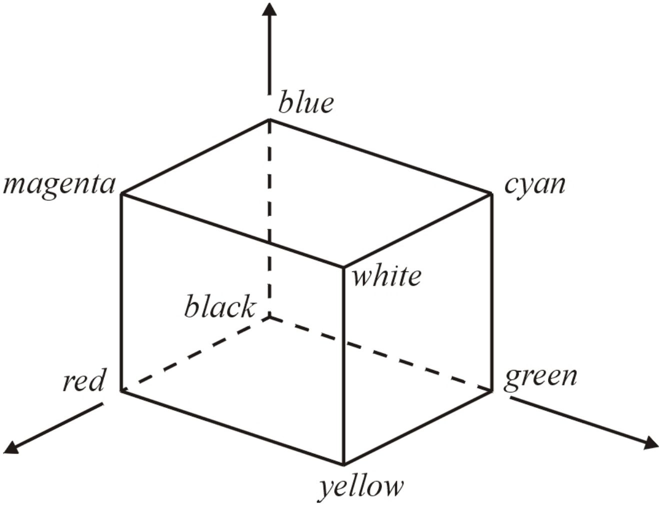
- additive color space

C cyan
Y yellow
M magenta
W white



RGB Color Space (cont)

- in 3D r, g, b form a color cube



RGB Color Space (cont)

- some rgb values for colors

color	r	g	b	color	r	g	b
black	0	0	0	cyan	0	1	1
white	1	1	1	magenta	1	0	1
gray	0.5	0.5	0.5	orange	1	0.65	0
red	1	0	0	navy	0	0	0.5
green	0	1	0	sky blue	0.53	0.81	0.98
blue	0	0	1	khaki	0.94	0.90	0.55
yellow	1	1	0	maroon	0.69	0.19	0.38

- note that this is not an intuitive color space!

CMY and CMYK Color Spaces

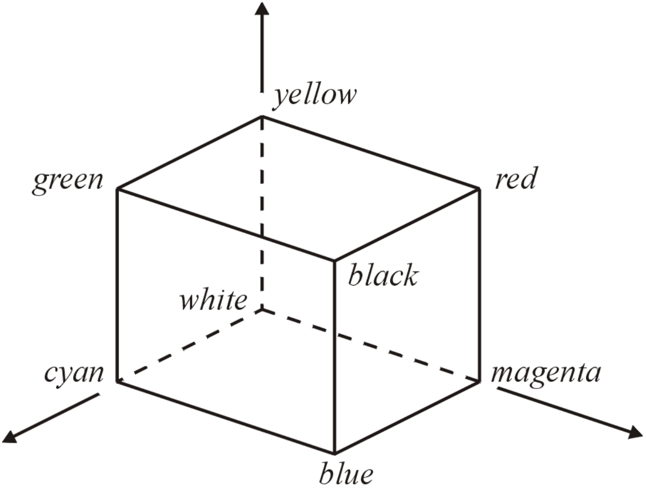
- most common printer color spaces are cyan-magenta-yellow (CMY) and CMYK (CMY plus black)
- C, M, Y, and K are not lights but filters of light
  - cyan filters out red
  - magenta filters out green
  - yellow filters out blue
  - (c, m, y) = (1, 1, 1) – (r, g, b)
- subtractive color space
  - start with white light and subtract red, green, and blue light using cyan, magenta, and yellow filters

CMY and CMYK Color Spaces (cont)

- your printer deposits tiny dots of transparent cyan, magenta, and yellow ink
  - each of these dots acts like a filter
  - printed images only look correct if printed on white paper and illuminated with white light
- equal amounts of cyan, magenta, and yellow can be replaced with black
  - conserves the more expensive color inks
    - k = min(c, m, y)
    - (c - k, m - k, y - k, k)

CMY and CMYK Color Spaces (cont)

- in 3D c, m, y form a color cube



HSV Color Space

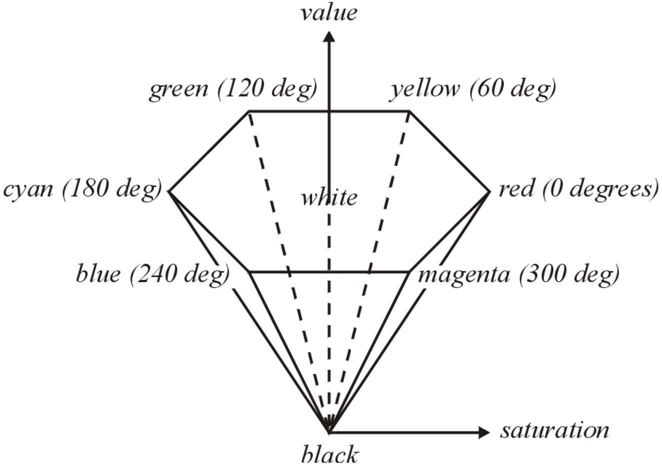
- hue, saturation, value (HSV) is a more intuitive color space than RGB
- hue
  - the different color sensations
    - red, green, and blue are different hues
- saturation
  - purity of color or how far from gray a color is
    - red is fully saturated (saturation = 1)
    - pink is less saturated (saturation < 1)
    - white is zero saturation (saturation = 0)
  - no mixture of three primaries is fully saturated

HSV Color Space (cont)

- value
  - the sensation of light and dark colors
  - white has a value of 1 and black has a value of 0
- easier for a human to choose colors
  - pick the color family (red, green, yellow, etc)
  - pick the purity or strength of the color
  - pick the lightness of the color

HSV Color Space (cont)

- hue is measured in degrees around the circle
- forms a hexcone in space



Color in OpenGL

- OpenGL only supports RGB and RGBA
  - we'll study RGBA a little later
- whenever an object is drawn, it is drawn with the current color
  - set color, draw, set color, draw, etc
- specify colors using

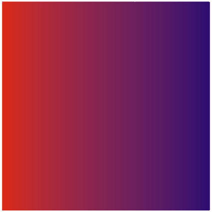
```
void glColor3f(float red, float green, float blue)
```

- sets the current color to (red, green, blue) where the values of red, green, and blue are clamped to between 0.0f and 1.0f

Color in OpenGL (cont)

- can set the color per vertex
  - OpenGL will interpolate color between vertices

```
glBegin( GL_QUADS );
  glColor3f( 1.0f, 0.0f, 0.0f );  // red
  glVertex2f( 0.0f, 1.0f );
  glVertex2f( 0.0f, 0.0f );
  glColor3f( 0.0f, 0.0f, 1.0f );  // blue
  glVertex2f( 1.0f, 0.0f );
  glVertex2f( 1.0f, 1.0f );
glEnd();
```

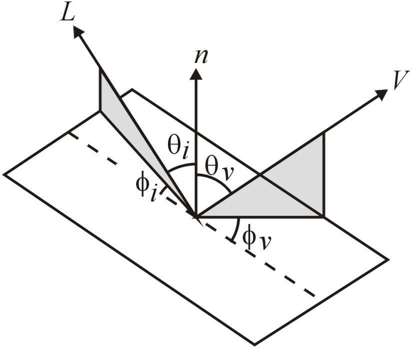
red  blue

Interaction of Light With Matter

- interaction of light with matter is generally not well understood
- a simplified approach is the bidirectional reflection distribution function (BRDF)
  - an even simpler approach is taken by traditional computer graphics (we’ll study this shortly)
- BRDF assumes that light striking a point on the surface leaves the surface from the same point
  - idea: for *every* direction incident on a point, measure the amount of light leaving the point in *every* direction

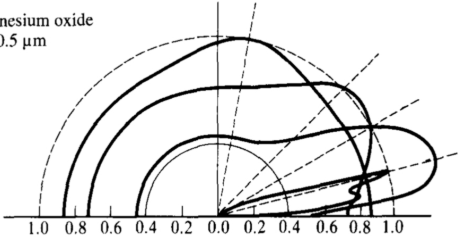
BRDF

- the BRDF is often written as $R(\lambda, \phi_i, \theta_i, \phi_v, \theta_v)$
  - λ is the wavelength (hue) of incident light
  - (ϕi, θi) defines the direction to the light source L
  - (ϕv, θv) defines the direction to the viewer V
- the BRDF tells us about the ratio of the incoming and reflected light



BRDF (cont)

- for real materials BRDF is usually very complex
  - need lots of samples from a BRDF to accurately model a surface
  - from "3D Computer Graphics" by Alan Watt



- need simpler models for most graphics applications

Phong Reflection Model

- most common model in computer graphics
- Hill uses “shading model” which is confusing
- model not based on physical principles
  - but looks good for plastic-like surfaces
- aside:
  - physically-based illumination models
    - Cook-Torrance (see Hill)
    - He (SIGGRAPH’91)
    - Oren and Nayar (SIGGRAPH’94)

Phong Reflection Model

- total light intensity at a surface is sum of three components:

$$I_{total} = I_{amb} + I_{diff} + I_{spec}$$

I_{amb}	ambient intensity
I_{diff}	diffuse intensity
I_{spec}	specular intensity

Reflected Ambient Intensity

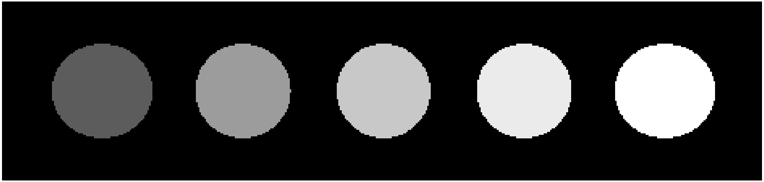
- why can you see the bottom of things when light comes from above? why are shadows not absolute black?
  - because light is reflected from other surfaces
    - called global illumination
- global illumination is very difficult to model accurately
- ambient intensity is crude approximation of effect of global illumination

Reflected Ambient Intensity (cont)

- assume ambient intensity is constant
- depends on:
  - amount of ambient illumination I_a
    - property of light source
  - material property ρ_a
    - property of object
    - called ambient reflection coefficient
- ρ_a is fraction of ambient intensity reflected by surface
  - $0 \leq \rho_a \leq 1$
- yields the reflected ambient intensity $I_{amb} = I_a * \rho_a$

Reflected Ambient Intensity (cont)

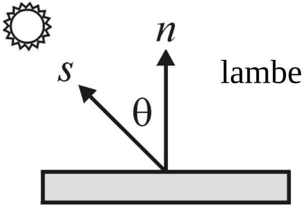
- picture of spheres lit with ambient light only
  - ambient reflection coefficient increases from left to right



- makes objects look flat

Reflected Diffuse Intensity

- a diffuse reflector reflects incident light equally in all directions
- obey Lambert's Law:
  - reflected intensity proportional to cos(θ)



$$\text{lambert} = \max(0, \cos(\theta)) = \max\left(0, \frac{\vec{s} \cdot \vec{n}}{|\vec{s}||\vec{n}|}\right)$$

- independent of where the viewer is
  - reflected intensity is the same in all directions

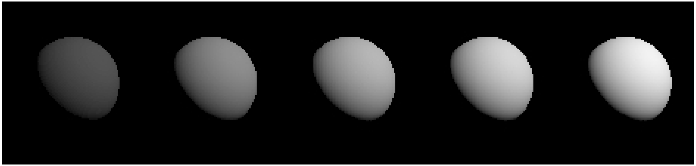
Reflected Diffuse Intensity (cont)

- depends on:
  - light source intensity I_s
    - property of light source(s)
  - material property ρ_d
    - property of object
    - called diffuse reflection coefficient
- ρ_d is fraction of diffuse intensity reflected by surface
  - $0 \le \rho_d \le 1$
- yields the reflected diffuse intensity

$$I_{diff} = I_s * \rho_d * \text{lambert}$$

Reflected Diffuse Intensity (cont)

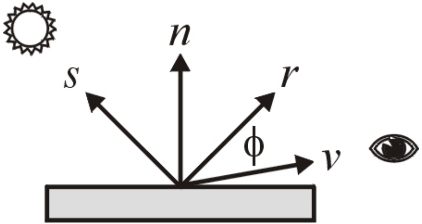
- examples of mostly diffuse surfaces:
  - roughened plastic, chalk, writing paper
- picture of spheres lit with diffuse intensity only
  - diffuse reflection coefficient increases from left to right



- provides information about shape

Reflected Specular Intensity

- specular intensity models shininess
  - results in highlights
- most of incident intensity reflected in mirror direction
  - but some is reflected around the mirror direction
- Phong approximation is pure hack
  - reflected intensity proportional to $\cos^f(\varphi)$

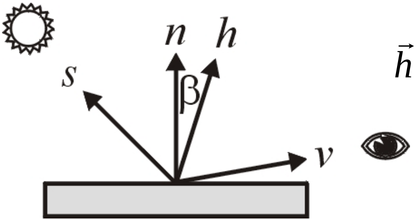


Reflected Specular Intensity (cont)

- how do we compute the mirror direction?
  - mirror direction

$$\vec{r} = -\vec{s} + 2\frac{\vec{s}\cdot\vec{n}}{|\vec{n}|^2}\vec{n}$$

- mirror direction is a bit expensive to compute
  - we can use the angle β between the normal vector and the halfway vector instead



$$\vec{h} = \vec{s} + \vec{v}$$

Reflected Specular Intensity (cont)

- depends on:
  - light source intensity I_s
    - property of light source(s)
  - two material properties
    - specular reflection coefficient ρ_s
    - ρ_s is fraction of specular intensity reflected by surface
      - $0 \leq \rho_s \leq 1$
    - specular reflection exponent f
    - f controls how fast the highlight decreases
    - big highlight $1 \leq f \leq 200$ small highlight

Reflected Specular Intensity (cont)

- using the mirror direction we get:

$$\text{phong} = \max(0, \cos(\phi)) = \max\left(0, \frac{\vec{r} \cdot \vec{v}}{|\vec{r}||\vec{v}|}\right)$$

- using the halfway vector we get:

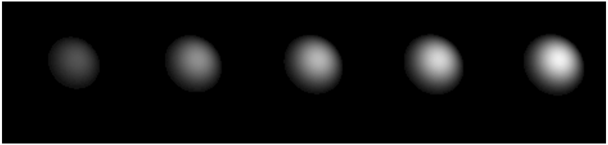
$$\text{phong} = \max(0, \cos(\beta)) = \max\left(0, \frac{\vec{h} \cdot \vec{n}}{|\vec{h}||\vec{n}|}\right)$$

- yields the reflected specular intensity

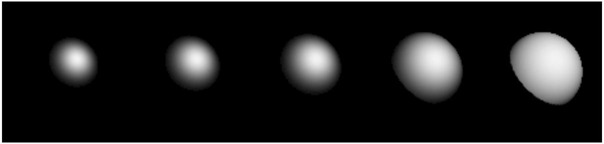
$$I_{spec} = I_s * \rho_s * \text{phong}^f$$

Reflected Specular Intensity (cont)

- examples of specular surfaces
  - smooth metal, smooth glass, smooth plastics
- specular reflection coefficient increases left to right



- specular exponent decreases left to right



Putting It All Together

- the total reflected intensity is

$$I_{total} = I_{amb} + I_{diff} + I_{spec}$$
$$= I_a * \rho_a + I_s * \rho_d * lambert + I_s * \rho_s * phong^f$$

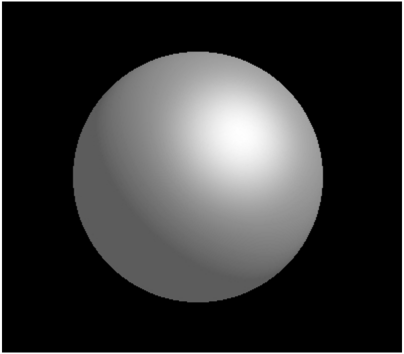
- Hill writes I_{total} a little differently

$$I_{total} = I_a * \rho_a + I_d * \rho_d * lambert + I_{sp} * \rho_s * phong^f$$

- I_d is the diffuse intensity of the light source
- I_{sp} is the specular intensity of the light source

Putting It All Together (cont)

- picture of spheres lit with Phong model



Adding Color

- to add color
  - source intensities are in (r, g, b)
  - all reflection coefficients are in (r, g, b)
    - curiously, f is a constant (not in (r, g, b))
- $I_{total,r} = I_{a,r} * \rho_{a,r} + I_{d,r} * \rho_{d,r} * lambert + I_{sp,r} * \rho_{s,r} * phong^f$
- $I_{total,g} = I_{a,g} * \rho_{a,g} + I_{d,g} * \rho_{d,g} * lambert + I_{sp,g} * \rho_{s,g} * phong^f$
- $I_{total,b} = I_{a,b} * \rho_{a,b} + I_{d,b} * \rho_{d,b} * lambert + I_{sp,b} * \rho_{s,b} * phong^f$

- notice that it is possible for $I_{total} > 1$
  - usually I_{total} is clamped to the range [0, 1]
- if there are multiple lights, we compute I_{total} for each light and add up all of the contributions